1

EPT

Encyclopedia Processor Technica

History and Basic Operation of the Sol

ENCYCLOPEDIA PROCESSOR TECHNICA

Volume 1

History and Basic Operation of the Sol

The Complete Guide to
Processor Technology Corporation
Products

Published by

Proteus
The Processor Technology Users' Group
1690 Woodside Road, Suite 219
Redwood City, California 94061

ENCYCLOPEDIA PROCESSOR TECHNICA

The complete guide to
Processor Technology Corporation's Products

Edited by Stanley M. Sokolow, B.A., D.D.S.
Sonya Sokolow, Ph.D.

Published by Proteus, Processor Technology Users
Copyright (C) 1982 Proteus

CONTENTS
Revised 4/19/82

Volume 1 -- History and basic operation of the Sol. Preface to the Encyclopedia Processor Technica. Includes the semi-technical Sol Users Manual that came with later Sols, functions of the Sol's buttons and switches, hookups to common printers, cassette operations, use as a dumb terminal, minor troubleshooting. Also a historical scrapbook of Processor Technology Corporation and of the Sol: reprints of articles which describe the way the Sol came to be, copies of old advertising, product catalogs, price lists, dealer list, historical accounts from insiders, Personal Computer Retailer newsletter for dealers, all issues of Access newsletter.

Volume 2 -- Technical manual on the Sol computer. It incorporates material from the big black looseleaf notebook that came with early Sol's, plus material taken from the PTC Sol Service Manual, from Access (PTC's newsletter to users), from Proteus, and from other sources. Covers topics such as: theory of operation, schematics, assembly drawings, parts lists, troubleshooting, testing, hardware updating (rev D to rev R), constructing and using the ParaSol Debugger, input/output hookups, timing diagrams, testpoint voltages, hardware interfacing, user-originated modifications.

Volume 3 -- Programming the Sol. Solos user's manual, the 8080 microprocessor from the programmer's view, the 8080 instruction set, assembly language programming, communications interface programs, the Sol system architecture.

Volume 4 -- PTC Programming Manuals.

Volume 5 -- ALS-8, Assembly Language System. Including the user's manual, users' group newsletters, internal routines.

Volume 6 -- PTC memory boards. From the 2KRO ROM board to the 64KRA-1 RAM board. Schematics, assembly drawings, theory of operation, switch settings, engineering changes, upgrades, troubleshooting, ROM listings, parts lists, etc.

Volume 7 -- PTC input/output interfaces. The 3P+S, the CUTS board, the VDM.

Volume 8 -- Helios Disk System Manual. Installation, testing, light maintenance, theory of operation, drawings.

Volume 9 -- Helios Service Manual. Sol/Helios System troubleshooting. Updates, Controller/formatter troubleshooting, PerSci Drive Maintenance.

Volume 10 -- Programming the Helios disk system. Users' manuals on the following: PTDOS 1.5, Extended Disk BASIC, Optional Precision BASIC, Level I Business BASIC (Demo version), Disk Basic/5.

Volume 11 -- SolPrinters and Hytype interface.

Volume 12 -- Sol a la carte. Using the Sol with other equipment, such as interfacing to other disks, modifying memory boards to work properly in Sol, customizations for the Sol with CP/M, NorthStar, Micropolis, etc. (This volume to be published at a later date.)

ENCYCLOPEDIA PROCESSOR TECHNICA

Foreword to the Encyclopedia

by Stanley M. Sokolow, B.A., D.D.S.

Since August, 1977, I have been the de facto Executive Director of the organization for owners of Processor Technology Corporation Sol computers. Originally, we called ourselves "Solus" (pronounced like "solice"), a name coined by Bill Burns, the founder of the group. The name represented "SOL Users Society," but it really had a double meaning since the early Sol users actually needed a lot of solice to cope with Processor Technology Corporation's growing pains.

Later, I changed the name to "Proteus" (for PROcessor TEchnology USers) to indicate a broader scope, covering all of the Processor Technology product line, not just the Sol. Proteus has continued to be the primary focal point for communication among Sol owners, mainly through our newsletter.

This Encyclopedia represents the accumulation of information which Proteus has been able to gather on Processor Technology products. It is envisioned as a living resource which will be updated as new information becomes available. Any reader who has information or literature on any Processor Technology hardware or software product is requested to submit this material to Proteus for inclusion into future editions of the Encyclopedia and its supplements. (The original documents will be returned to you if requested.)

I also urge all owners of Sol computers to subscribe to our newsletter, called Proteus News. This has been an invaluable source of practical information about the Sol and other Processor Technology products. Our hope is that the organization will continue to keep the Sol a viable computer as long as there is interest in it, and even far into the future when it is an antique.

Proteus's address is Proteus, c/o Stanley M. Sokolow, 1690 Woodside Road, Suite 219, Redwood City, California 94061.

EXPLANATION OF PAGE NUMBERING

Since the material in the Encyclopedia has come from many

sources, the original page numbering is out of sequence. Instead, we refer to new page numbers at the top of each page. They have this form:

5 EPT 7 - 123

where "EPT" means Encyclopedia Processor Technica, the number before it means the volume number, the number following is the chapter number, and the number after the hyphen is the new page number. We left the old page numbers intact so that references to them in the body of the text could be found.

ENCYCLOPEDIA PROCESSOR TECHNICA

Copyright (C) 1982 Proteus
1690 Woodside Road
Redwood City, California 94061

DISCLAIMER

The information presented in this and other volumes of the Encyclopedia Processor Technica is published in the interest of the owners of Processor Technology Corporation products. The publisher assumes no liability for the accuracy of the content presented, although we have done our reasonable best to include only accurate information. It is assumed that the reader will use due care in utilizing the information, especially with regard to suggested modifications to products.

1 EPT 3/3/82

ENCYCLOPEDIA PROCESSOR TECHNICA

Volume 1

History and Basic Operation of the Sol

CONTENTS

Page

Foreword to the Encyclopedia.

Preface to Volume 1.

Chapter 1. Historical Scrapbook of Processor Technology... 1
"VDM-1" (Jan 1976)..... 2
"BYTE's Audio Cassette Standards Symposium (Feb 1976).. 4
"4KRA" (March 1976)..... 6
"Altair" (April 1976)..... 8
"8080 Expansion" (June 1976).....10
"Popular Electronics" (July 1976).....11.1
"BYTES'S Altair 8800 Users!" (September 1975.....11.6
"8080 Compatible" (Sept 1976).....12
"Sol-20" (Nov 1976).....14
"Sol" (Dec 1976).....16
"Intelligent-Terminal Designers Opt for 8080-Compatible Circuitry," by Lee Felsenstein and Robert Marsh (Feb 1977).....22
Personal Computer Retailer, Vol. 1 No. 1 (March 1977)..23
The Small Computer Catalog, May 1977.....31
"Sol: The Inside Story," by Lee Felsenstein (July 1977).....53
"The Sol-20 Computer Terminal," by Rudolf Hirschmann (Aug 1977).....59
Sol System Suggested Retail Price List (Sept 1, 1977)..61
"Why Sol Should be your Small Computer Choice."....63
"Helios II Disk Memory System." (Product description)..64
Sol Small Computer Systems (Catalog).....70
Personal Computer Retailer, May 3, 1978.....78
"Build an all-in-one home computer," by J. Free (June 1978).....84
Sol System Suggested U.S. Retail Price List (July 1, 1978).....85
Personal Computer Retailer, July 18, 1978.....87
Technical Training Seminars Diploma.....93
Sol Dealer List, August 1978.....94
Personal Computer Retailer, Sept 7, 1978.....95
Product Description: "WordWizard Electronic Typing System".....101
Sol System Suggested U.S. Retail Price List (Oct 1, 1978).....105
Retailer Memo, October 12, 1978.....109
Personal Computer Retailer, November 20, 1978.....113

Personal Computer Retailer, December 18, 1978.....119
Personal Computer Retailer, January 26, 1979.....125
Processor Technology 90 Day Limited Warranty, 2/6/79..129
Personal Computer Retailer, April 18, 1979.....130
"Processor Technology Goes Under," by Tom Williams
6/25/79.....136
"A User's Reaction to the SOL-10 Computer" by Robert
Bumpous, BYTE, January 1978.....137
"User's Report: The SOL-20" by Dennis Barbour, BYTE,
April 1978.....143

Chapter 2. Processor Technology ACCESS Newsletter
Vol 1, No 1, February 1977..... 2
Vol 1, No 2, April 1977.....14
Vol 1, No 3, June 1977.....30
Vol 1, No 4, November 1977.....50
Vol 2, No 1, March 1978.....86

Chapter 3. Sol Users Manual.
Section 1. Exploring the Sol System..... 8
Section 2. At the Controls.....18
Section 3. Introduction to Software.....22
Section 4. Loading Programs from Cassette.....28
Section 5. At the Keyboard (Key Functions).....36
Section 6. Unpacking and Installation.....45
Section 7. TLC for Your Sol (Light Maintenance).....54
Appendix 1. Products Available from Processor Tech.....59
Appendix 2. Cassette Recorder Supplement.....65
Selecting a Cassette Recorder.....65
Interconnect Requirements for Two Recorders.....66
Appendix 3. Internal Controls (Switch Settings).....68
Appendix 4. Keyboard Tables.....72
Appendix 5. Sol Specifications.....76

VOLUME 1

PREFACE

This first volume of the Encyclopedia Processor Technica is intended to provide a general introduction to the Sol computer and its manufacturer, Processor Technology Corporation. The history is sketchy because when the corporation went out of business the records were mostly all discarded and the employees scattered as they went on to new projects. As more information is uncovered, we will publish it in our annual supplements to the Encyclopedia. We hope that readers who have something of value to add will contact Proteus and help us fill in the gaps.

In addition to this volume, historical information can be gleaned from back issues of Proteus News, the newsletter of the Processor Technology Users, publisher of this Encyclopedia. These back issues are available in a separate volume.

Stanley M. Sokolow,
Editor

Chapter 1. Historical Scrapbook of Processor Technology... 1
"VDM-1" (Jan 1976)..... 2
"BYTE's Audio Cassette Standards Symposium (Feb 1976).. 4
"4KRA" (March 1976)..... 6
"Altair" (April 1976)..... 8
"8080 Expansion" (June 1976).....10
"Popular Electronics" (July 1976).....11.1
"8080 Compatible" (Sept 1976).....12
"Sol-20" (Nov 1976).....14
"Sol" (Dec 1976).....16
"Intelligent-Terminal Designers Opt for 8080-Compatible Circuitry," by Lee Felsenstein and Robert Marsh (Feb 1977).....22
Personal Computer Retailer, Vol. 1 No. 1 (March 1977)..23
The Small Computer Catalog, May 1977.....31
"Sol: The Inside Story," by Lee Felsenstein (July 1977).....53
"The Sol-20 Computer Terminal," by Rudolf Hirschmann (Aug 1977).....59
Sol System Suggested Retail Price List (Sept 1, 1977)..61
"Why Sol Should be your Small Computer Choice."....63
"Helios II Disk Memory System." (Product description)..64
Sol Small Computer Systems (Catalog).....70
Personal Computer Retailer, May 3, 1978.....78
"Build an all-in-one home computer," by J. Free (June 1978).....84
Sol System Suggested U.S. Retail Price List (July 1, 1978).....85
Personal Computer Retailer, July 18, 1978.....87
Technical Training Seminars Diploma.....93
Sol Dealer List, August 1978.....94
Personal Computer Retailer, Sept 7, 1978.....95
Product Description: "WordWizard Electronic Typing System".....101
Sol System Suggested U.S. Retail Price List (Oct 1, 1978).....105
Retailer Memo, October 12, 1978.....109
Personal Computer Retailer, November 20, 1978.....113
Personal Computer Retailer, December 18, 1978.....119
Personal Computer Retailer, January 26, 1979.....125
Processor Technology 90 Day Limited Warranty, 2/6/79..129
Personal Computer Retailer, April 18, 1979.....130
"Processor Technology Goes Under," by Tom Williams 6/25/79.....136

BYTE January 1976 pp. 20,21

NEW!... VDM-1

1 EPT 1 - 2

features-

- ultra high speed intelligent display
- generates 16, 64 character lines of alpha-numeric data
- displays upper and lower case characters
- full 128 ascii characters
- single printed circuit card
- standard video output

- $160.00 -

SPECIAL FREE OFFER!

Scientific Notation Software Package with Formatted Output

The floating point math package features 12 decimal digits with exponents from +127 to −127; handles assigned and unassigned humbers. With it is a 5 function calculator package: + − X ÷ & sq. root. It includes 3 storage and 3 operating memories and will handle chain and column calculations.

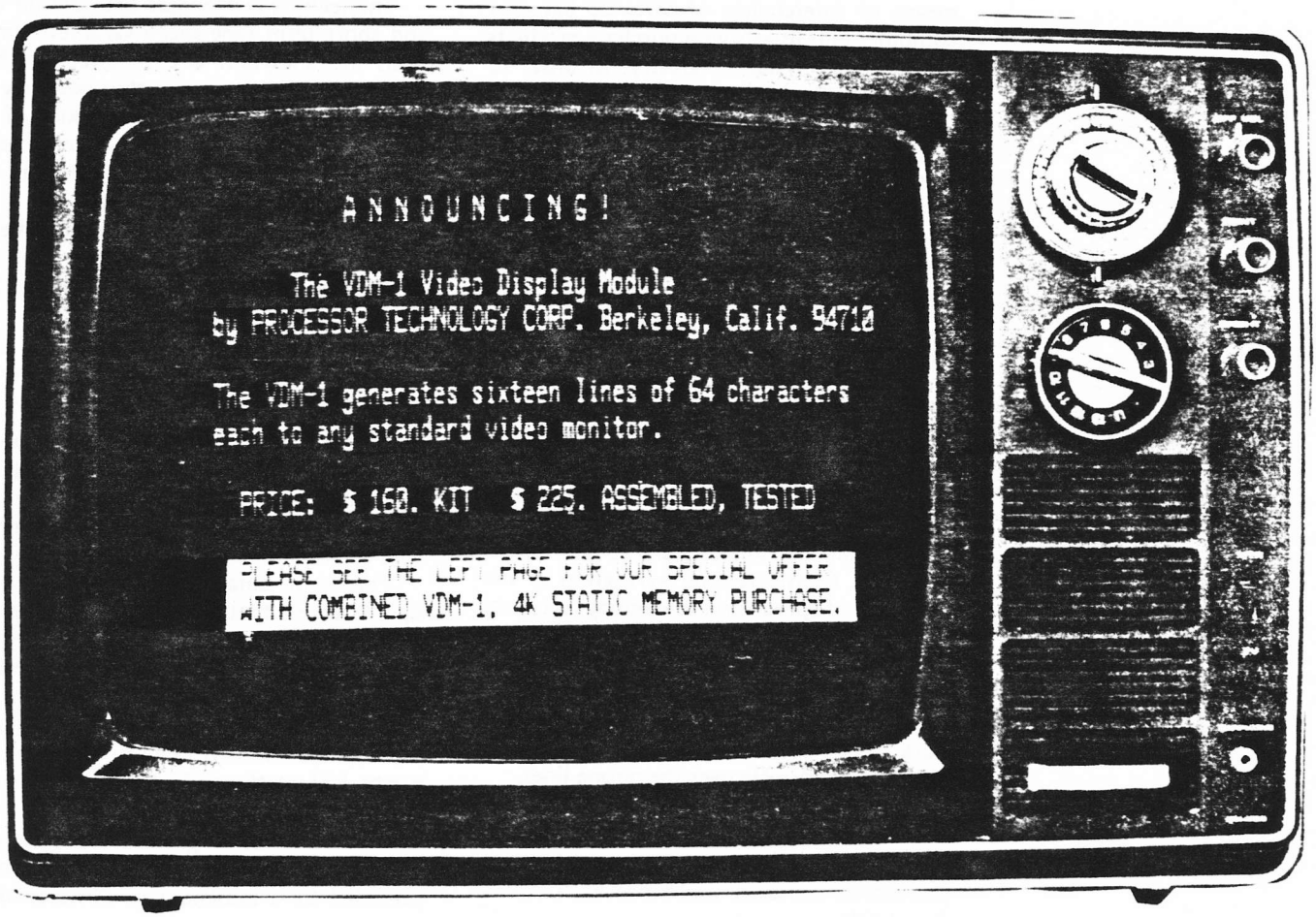
With the purchase of (1) VDM-1 and (1) 4KRA-4 Memory:

Just $299.00 (Offer expires 2-1-76)

from-

Processor Technology

VIDEO DISPLAY MODULE



Corporation

2465 Fourth Street
Berkeley, Ca. 94710

BYTE February 1976 pp. 72,73 1 EPT 1 - 4

Report:

BYTE's Audio Cassette Standards Symposium

Manfred and Virginia Peschke

BYTE Magazine sponsored a symposium on November 7 and 8, 1975 in Kansas City MO regarding the interchange of data on inexpensive consumer quality audio cassette drives. These drives may be used as one of the mass storage devices in the first generation of personal computers, and will retain importance for some time to come as a means of interchange of software between computer enthusiasts who purchase products of the small systems industry. In order to promote the growth of the industry, BYTE sought to achieve an industry standard on audio cassette data interchange through a working conference.

We extend our greatest appreciation to the 18 people who worked very hard until late Friday night and Saturday morning to discuss the multitude of problems and solutions associated with digital recording on auto cassettes. The names of the participants are listed in table 1. In spite of the short time available, the participants were able to draft a set of provisional standards which seems to promise great reliability and is rather inexpensive to implement; implementations may be entirely in hardware, or may require a mix of software and some minimal hardware. Considerations were given to the problems of speed variation among recorders and playback equipment, start and stop delays, recording density (or speed) versus reliability, and recording frequencies to avoid interference with the telephone network in case some users plan to transmit the tones of the cassette over the phone lines.

On Saturday afternoon, Mr. Felsenstein and Mr. Mauch volunteered to write up the consensus among the participants as to a provisional standard which has been reproduced below.

Provisional Audio Cassette Data Interchange Standard

The consensus among the participants of the audio cassette standards symposium at Kansas City MO sponsored by BYTE Magazine is as follows:

The proposed standard centers around the use of a frequency shift modulation method from which serial clock data can be extracted at rates of up to 300 baud. The system is intended to be used with low to medium cost cassette recorders incorporating electrical stop and start capability which may be operated under program control.

The technique proposed provides for long and short term tape speed variation, limitations in bandwidth due to effects such as tape misalignment, and the necessity to retain low cost and low complexity of the hardware. The technique allows for potential operation at higher tape speed than the nominal 1.875 inch/s (4.75 cm/s).

A mark (logical one) bit consists of eight cycles at a frequency of 2400 Hz.

A space (logical zero) bit consists of four cycles at a frequency of 1200 Hz.

A recorded character consists of a space as a start bit, eight data bits, and two or more marks as stop bits.

The interval between characters consists of an unspecified amount of time at the mark frequency. In this respect the data format is similar to that of asynchronous data communication.

The eight data bits are organized least significant bit first, most significant bit last, and followed (optionally) by a parity bit. The total number of significant bits and the parity bit cannot exceed 8.

Where less than eight data bits are used, the unused bits (following the optional parity bit) at the end of the character are mark bits (2400 Hz).

Data will be organized in blocks of arbitrary and optionally variable length, preceded by a minimum of five seconds of marks.

To avoid errors due to splice and wrinkle problems common at the begining of tape, the beginning of the first data block will occur no sooner than 30 seconds from the beginning of clear leader.

The contents of the data block are not specified.

The data block ends after the stop bits of the final character.

Bit clocking information may be extracted from the recorded waveform, which is always an integer multiple of the bit rate, regardless of tape speed. This permits the recovery and retiming of data by means of a UART, which requires a clock of sixteen times the bit rate, although other simple circuitry may be used.

A reliable bandwidth of 3000 Hz was assumed in choosing mark and space frequencies due to the head misalignment expected between various cassette recorders. The recording technique is a redundant form of Manchester or bifrequency code which has a long history of reliability in the computer industry. In its present form it was proposed by three independent manufacturers at the conference. One cited reliability rates of one error in 10**7 characters for 200 passes. ■

Table 1: Participants at Audio Cassette Symposium.

Ray Borrill, 1218 Prairie Dr, Bloomington IN

Hal Chamberlin, The Computer Hobbyist, P O Box 5985, Raleigh NC 27607

Tom Durston, MITS, 6328 Linn NE, Albuquerque NM

Lee Felsenstein, LGC Engineering, 1807 Delaware St, Berkeley CA 94703

Joe Frappier, Mikra-D, 32 Maple St, Bellingham MA

Bill Gates, MITS

Gary Kay, Southwest Technical Products Corp, 219 W Rhapsody, San Antonio TX 78216

Bob Marsh, Processor Technology, 2465 Fourth St, Berkeley CA 94710

Harold A Mauch, Pronetics, 4021 Windsor, Garland TX 75042

Bob Nelson, PCM, San Ramon CA

George Perrine, HAL Communications Corp, Box 365, Urbana IL 61801

Ed Roberts, MITS

Richard Smith, The Computer Hobbyist, P O Box 5882, Raleigh NC 27607

Les Solomon, Popular Electronics, 1 Park Av, New York NY 10016

Michael Stolowitz, Godbout Electronics, Box 2355, Oakland Airport CA 94614

Paul Tucker, HAL Communications Corp

Mike Wise, Sphere, 791 S 500 W, Bountiful UT 84010

Bob Zaller, MITS

1K 475 ns STATIC RAM
$4.25 for one
$4.00 each for eight

SIGNETICS 2602-1

all orders shipped postpaid and insured. Mass residents add 3% sales tax

$3.75 each for 32

WHY PAY FOR BEING SMALL?

Centi-Byte is a new source of memory components and other necessary items for the computer hardware builder. Our function is to be a voice to the manufacturing companies representing you, the modest volume consumer of special purpose components. *Centi-Byte* brings you this special introductory offer of fast memory chips, chips fast enough to run an MC6800 or 8080 at maximum speed. These 2602-1's are new devices purchased in quantity and fully guaranteed to manufacturer's specifications.

Centi-Byte works by concentrating *your* purchasing power into quantity buys of new components. Let us know what you need in the way of specialized components and subsystems for future offerings. With *your* purchasing power concentrated through us, together we will lower the cost of home computing.

Centi-Byte

PO BOX 312
BELMONT, MASS. 02178

OSI MODEL 300 COMPUTER TRAINER

THE MODEL 300 IS A COMPLETELY ASSEMBLED AND TESTED, READY TO USE, COMPUTER DESIGNED TO "BOOTSTRAP" THE STUDENT, HOBBYIST, AND ENGINEER INTO THE MICROPROCESSOR WORLD. THE UNIT IS BUILT AROUND THE MOS 6502 8 BIT MICROPROCESSOR AND USES A 128 WORD RAM. ITS CONTROLS INCLUDE 8 DATA SWITCHES, 7 ADDRESS SWITCHES, MEMORY LOAD, PROCESSOR RESET, PROCESSOR RUN, INTERRUPT, AND MEMORY WRITE PROTECT. THE DISPLAYS INDICATE DATA, ADDRESS, AND PROGRAM EXECUTION. TWO INPUT LINES AND ONE OUTPUT LATCH ARE PROVIDED FOR I/O PROGRAMMING. THE MODEL 300 COMES COMPLETE WITH A LABORATORY MANUAL WITH 20 EXPERIMENTS STARTING WITH SIMPLY LOADING AND READING MEMORY AND FINISHING WITH OUTPUTING TO A TELETYPE.

MODEL 300 COMPUTER TRAINER COMPLETELY ASSEMBLED WITH LAB MANUAL (REQUIRES +5VDC AT 500MA)..... $ 99.00

THE 6000 SERIES COMPUTER FAMILY (SUPERBOARD)

A COMPLETE MINICOMPUTER PC BOARD (DOUBLE SIDED EPOXY) WHICH ACCEPTS ANY 6000 SERIES PROCESSOR, SYSTEM CLOCK, 2- 1702 TYPE ROMS, 1K X 8 RAM (2102 TYPE), 1 PIA, 1ACIA, CURRENT LOOP AND PARALLEL INTERFACES AND HAS BUS EXPANSION CAPABILITIES. EACH SUPERBOARD COMES COMPLETE WITH DOCUMENTATION.

SUPERBOARD BARE WITH MANUAL 29.00
6800 MICROPROCESSOR AND SUPERBOARD 69.00
6501 MICROPROCESSOR AND SUPERBOARD 49.00
6502 MICROPROCESSOR AND SUPERBOARD FEATURES INTERNAL CLOCK 54.00

ALSO AVAILABLE :
ALL SYSTEM SUPPORT PARTS
RAM - ROM MEMORY EXPANDER BOARD
SUPERI/O BOARD CONTAINING CASSETTE INTERFACE, X, Y DISPLAY AND A/D CONVERTER.
VIDEO GRAPHIC BOARD

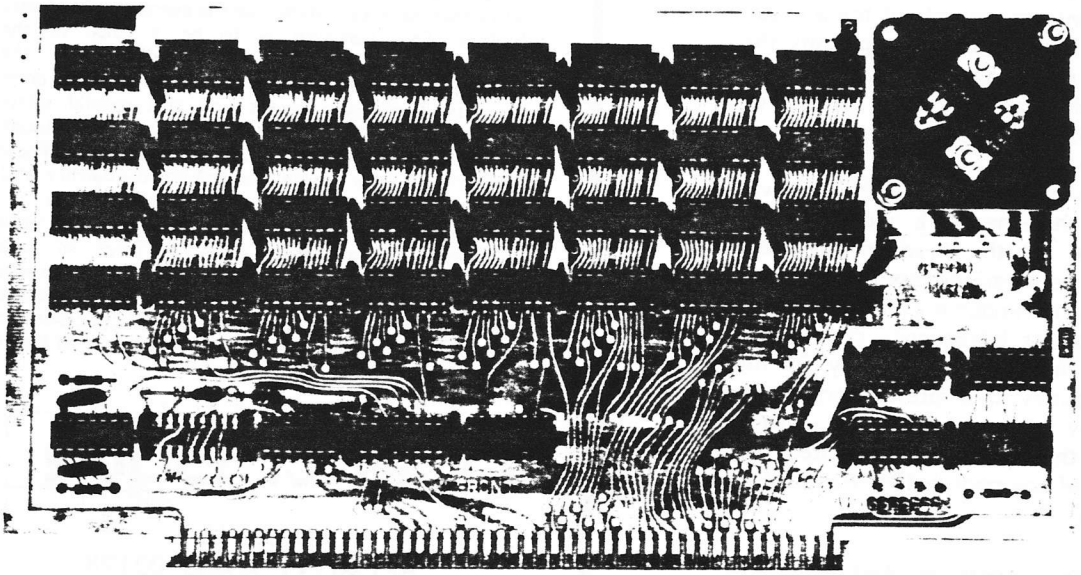
COMING SOON:
FIRMWARE BASIC BOARD (USES ROM AND CALCULATOR CHIP)

CALL (216) 653-6484 OR WRITE TODAY FOR OUR COMPLETE INFORMATION PACKAGE.

OSI OHIO SCIENTIFIC INSTRUMENTS
P.O BOX 374 HUDSON, OHIO 44236

BYTE March 1976 pp. 48,49 1 EPT 1 - 6

4KRA
LOW POWER MEMORIES AT A NEW LOW PRICE!



Want Details?

The Processor Technology 4KRA-4 is the fastest, most reliable and yet, least expensive read/write memory module on the market today. The 4KRA-4 is the only memory available of any kind that can be powered by two "D" size batteries for up to ten (10) hours. (Ask for our TB-101 technical bulletin on back-up battery operation of the 4KRA-4).

Don't be mislead by "undirected" statements on the subject of memory power consumption. All RAM's used in the 4KRA-4 consume typically 1/3 the power of 8101 or 2102 type memories. Under absolute worst case conditions our RAM's require only 30% more power than the "typical" consumption of any 4K dynamic memory. Remember that dynamic memories use three power supplies, but static memories only one.

It's time to clear the air of any confusion anyone might have about memory speed. In any 8080 system, all memories with access times between 50 and 520 nanoseconds are the same speed! Access time alone is not a valid indicator of speed unless it is greater than 550 nanoseconds, thereby requiring slow-down "wait" states. However, two other factors affect overall system speed.

1) Dynamic memories must refresh themselves periodically, slowing down the micro processor. In a well designed dynamic memory system refresh slows the processor by a few percent. Static memories do not require any refreshing. When our 4KRA-4 memories are used in the Altair 8800 the "wait" light goes out, indicating maximum speed operation.

2) Long cycle times can slow the system down during critical operations such as Direct Memory Access. DMA is used by most disk memories and by such devices as a soon to be announced color graphics generator. Most dynamic memories now on the market have a 1500 nanosecond cycle time, about three times that of our 4KRA-4. This longer cycle time can slow down the DMA device by at least 33%!

Most important, our 4KRA-4 Static Memories work, and keep on working! Processor Technology has four 8800 computers, each using at least 32K Bytes of 4KRA-4 memory. We use these machines heavily for program development and product testing. We have yet to lose a single bit in normal operation! Reports from our customers confirm our experience and indicate that we have one of the lowest failure rates in the industry.

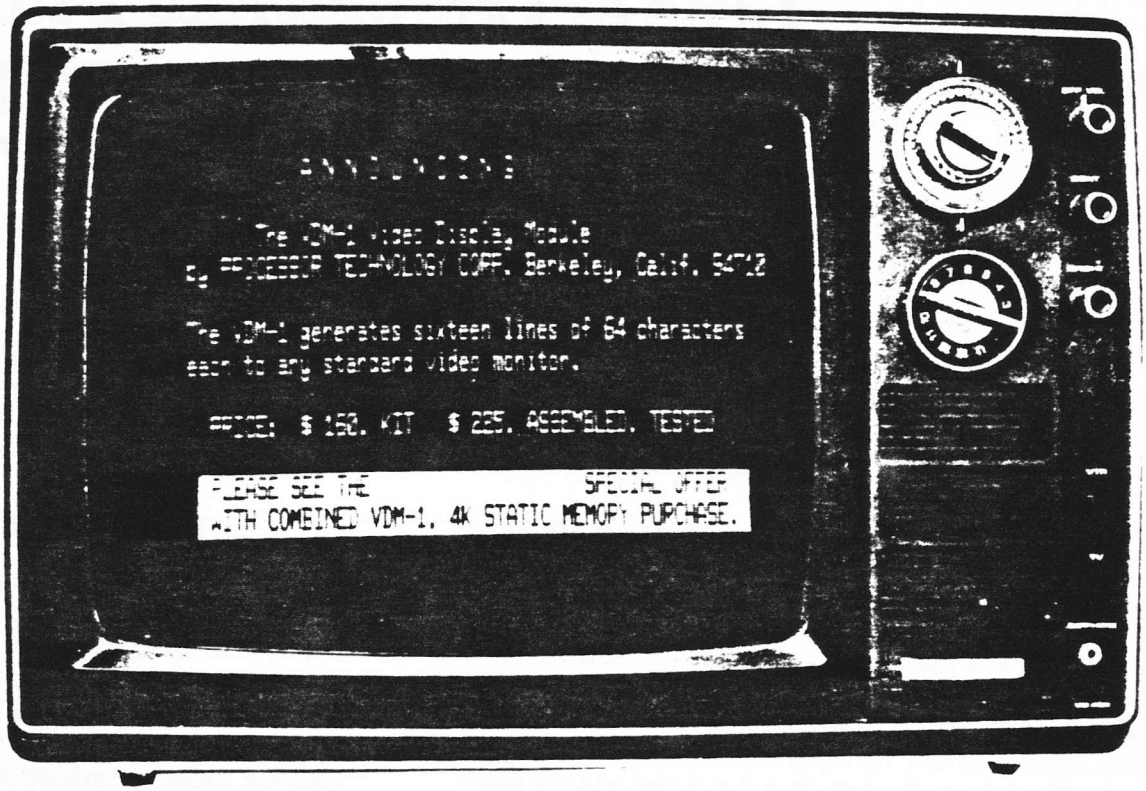
Frankly, we have done everything we could to make the best read/write memory around, because, after all, memory is the most important part of any computer.

SPECIFICATIONS: 4KRA-4

Maximum capacity: 4096 eight bit bytes
Operating mode: Static
Access time: 520 nano-seconds, worst case maximum
Cycle time: 520 nano-seconds maximum, read or write
Bus Pinout: Plug-in compatible with Altair 8800 Bus
Edge contacts: Gold plated, 100 pins (dual 50) on .125" centers
Power requirements, operating: +7.5 to +10VDC at 1.0A maximum (0°C), 0.8A typical at 25°C.
standby: +1.6 to +2.5VDC at 0.5A maximum worst case, 0.4A typical
Dimensions: 5.3" x 10.0" (13.46cm x 25.4cm)

$139 KIT FORM **$195 ASSEMBLED**

VDM-1
VIDEO DISPLAY MODULE



Please don't think of the VDM-1 as an ordinary TV typewriter! The VDM-1 is an intelligent display whose capabilities are limited only by your imagination! All cursor and display formations are fully programmable — there are very few hardware limitations inherrent in the design.

The VDM-1 can be used as a terminal when running BASIC or our Resident Assembler using the FREE software drivers included with every kit.

The VDM-1 contains 1024 bytes of low power high speed RAM memory which can be directly accessed by the computer as any 1K segment within its normal 65K address range. The VDM-1 is a single pc card and is plug-in compatible with the Altair 8800 bus. Multiple cursors are possible, each fully programmable. The display can be black on white, white on black, or both simultaneously. Output is standard EIA video with a signal bandwidth of 7 Mhz, compatible with any video TV monitor. The VDM-1 is so fast, efficient, and powerful we think it will soon become the standard against which other displays must be compared.

SPECIAL FREE OFFER!

Scientific Notation Software Package with Formatted Output

The floating point math package features 12 decimal digits with exponents from +127 to −127; handles assigned and unassigned numbers. With it is a 5 function calculator package: + − X & sq. root. It includes 3 storage and 3 operating memories and will handle chain and column calculations.

With the purchase of (1) VDM-1 and (1) 4KRA-4 Memory: $299.00 (Offer Expires April 1, 1976)

Please write for details on these and other Altair bus compatible modules

Processor Technology

2465 Fourth Street Berkeley CA 94710 (415) 549 0857

BYTE April 1976 pp. 48,49

Watch Your

With Help, Of Course, From Line of Compatible Plug-in

All our products are compatible with the 8800 — they just plug in to become an integral part of your system. Each card can be used in the widest range of applications, giving maximum versatility to your Altair. Our "no compromise" philosophy assures you the highest possible quality. We use a conservative thermal design to provide for long life and reliable operation. We've also put hysteresis bus receivers on noise prone high-speed inputs, giving you maximum noise immunity. And, finally, we cover it with a full six- month warranty.

Video Display Module

Your Altair already has the intelligence, so let us provide the display module. This is **not** a limited "TV Typewriter", but an ultra-high speed computer terminal built into your computer. The **VDM-1** generates sixteen 64-character lines from data stored in the 1K Byte on-card memory. Alpha-numeric data is shown in a 7 x 9 dot matrix with a full 128 **upper and lower case** ASCII character set. The VDM-1 features EIA Video output for any standard video monitor. (Your TV set can be easily modified at your local television repair shop.) Multiple programmable cursors, automatic text scrolling and powerful text editing software are included **free.** Kit Price, $160.

A Versatile I/O Card

Just one **3P+S** card will fulfill the Input/Output needs of most 8800 users. There are two 8-bit parallel input and output ports with full handshaking logic. There is also a serial I/O using a UART with both teletype current loop and EIA RS-232 standard interfaces provided. The serial data rate can be set under software control between 35 and 9600 Baud. You can use your models 15, 28 or 33 TTY! This module gives you all the electronics you need to interface most peripheral devices with the 8800. And, should you decide to buy a 3P+S, we'll be happy to advise you on the best way to implement your system with our module. Kit Price, $125.

FOCAL!

Get a full 8080 implementation of 8K FOCAL* (including the game of Lunar Lander). It's now at your local Processor Technology dealer. Object tape is available for the copying charge only; complementary source listing available with minimum purchase.
(*® Digital Equipment Corp.)

An EPROM Module

Read Only Memories do not lose their stored data when power is removed. Thus, they have an advantage when used in stored program applications. Some ROM's, called EPROM's, are both erasable and reprogrammable. Our **2KRO** module will accept up to eight EPROM's, providing the user with up to 2048 eight bit words of non-volatile storage for monitor, executive, loader, and other programs. (We recommend the use of 1702A and MM5203 EPROM types. They are not included, but are readily available for reasonable prices on the industrial and surplus markets.) Kit Price, $50.

Altair Grow!

Processor Technology's Growing Hardware and Software.

Low Power, Plus!

The **4KRA** is a 4096 word read/write static memory module. It provides **faster, less expensive, lower power, and more reliable operation** than any comparable memory module sold today. The static memories don't need refreshing, so the result is faster speed in actual operation. Lower power does **not** mean decreased reliability. All RAM's used in the 4KRA are 91L02A's by Advanced Micro Devices. These RAM's typically require one-third the power of standard 2102 or 8101 types, and, even under worst case conditions, draw only 30% more than any currently available dynamic memory. Each RAM is manufactured to military specification. Since our module draws so little standby current, **memory can be maintained using a battery back-up,** in case of power failure, allowing long term retention of data. Kit Price, $139.

Software

Our Assembly Language Listing (Package #1) is $3, and includes the source listing with hexidecimal object code. Each command is described and six pages of sample use are included. Paper tapes of this system are available from many Computer Clubs throughout the country.

Turn-On-The-Switch Power

With our ALS-8, the full power of your computer is available at the instant you turn the switch. It provides 6K of PROM's, pre-programmed with an **expanded** version of our Software Package #1, including advanced file management capabilities. Files can be appended together, re-numbered, moved, or taken apart . . . all the features you'd expect from a company willing to practically give away Software #1. We include a manual which explains the use of the program (with examples), each routine of the system, and how to call these routines from other programs you've written. You won't be locked into a ROM version -- you'll be able to customize your ALS-8 to your individual needs. The ALS-8 Firmware includes a module with all components capable of holding 8K of "turn-on-the-switch" program, so it will be upward compatible with future software developments. And, two of these are up and running now. Kit Price, $250.

An Interpretive Simulator...

Our SIM-1 Expansion Firmware runs 8080 programs, in an interpretive mode, on the same 8080 that contains the Simulator. This isn't just a de-bug program, because the program actually "thinks" it's an 8080! A complete brochure explaining all its capabilities is available from Processor Technology. Price, $95.

And, A Text Editor!

Every ALS-8 includes the edit command. This command combined with our **Expansion Firmware #2** (Text Editor) and the VDM adds the world of text editing to your system. Single characters, entire lines, portions of lines can be inserted, deleted, moved and, ultimately, printed out. The addition of Firmware #2 to your ALS-8 will give you text editing capability equivalent to systems selling for $30,000 just a few short years ago. It's been running at Processor Technology since January, and it's ready for shipment **now.** Price, $95.

Write us for complete specifications on these and other compatible plug-in products: Our single-piece Mother Boards will give you 16-card capacity in one installation. A Wire Wrap Board, to help you do your own wire wrap prototyping, creating custom interfaces. An Extender Board, which allows accessibility in servicing any 8800 compatible module.

All items postpaid if full payment accompanies order. COD orders must include 25% deposit. Master Charge (minimum $25) accepted, but please send an order with your signature on it. Discounts on orders over $375.

Processor Technology

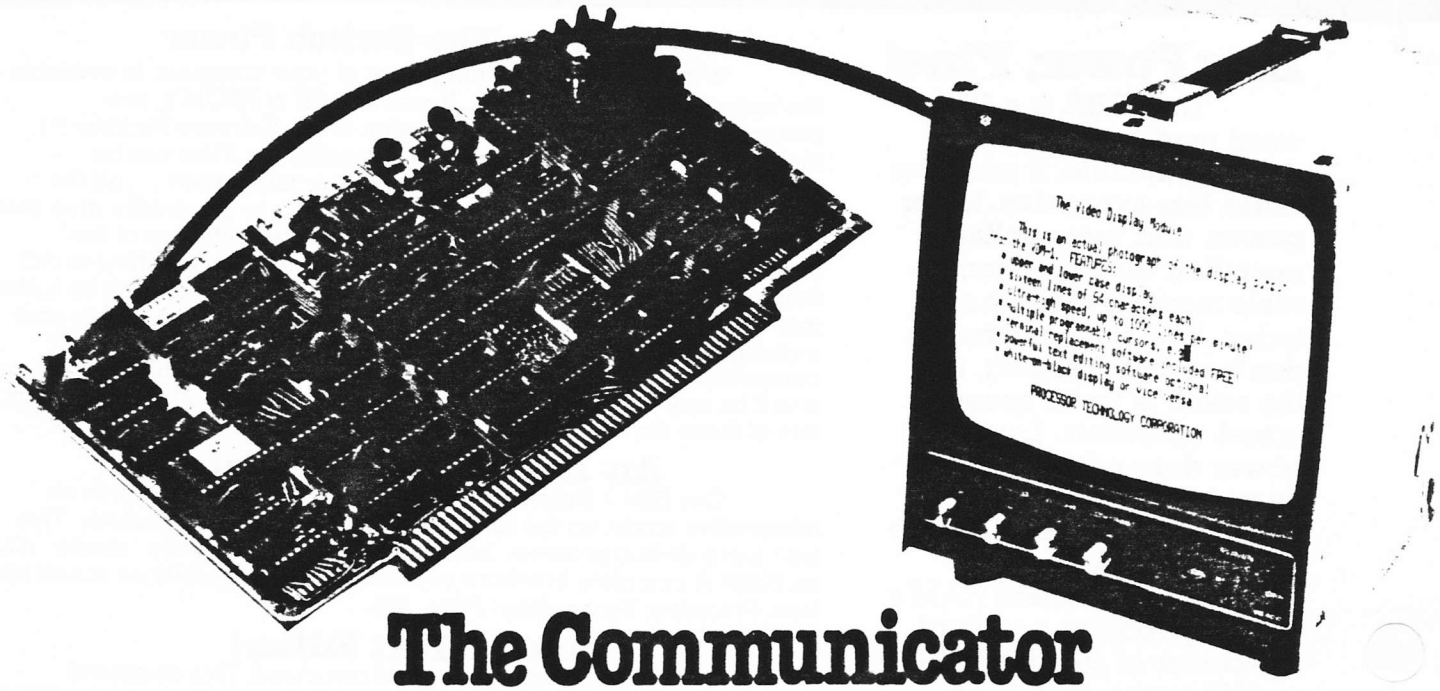
6200-B Hollis Street
Emeryville, CA 94608

BYTE June 1976 pp. 56,57

1 EPT 1 - 10

The 8080 Expa

Let Processor Technology expand your Altai[illegible] and reliable plug-in modules.



The Communicator

Your Altair already has the intelligence, so let our **VDM-1 Display Module** make the best of its capacity to communicate. This is **not** a limited "TV Typewriter." The VDM-1 is an **ultra-high speed output device, built right into your 8800 system.**

The VDM-1 generates **sixteen 64-character** lines in a large easy-to-read font with both **upper and lower case letters.** It contains 1K (1024) bytes of random access memory, to which the processor can read or write, just as though the memory were an integral part of the system. As the information is written in, contents of the on-card memory are displayed instantly without interrupting the operation of the processor.

Once the processor provides the display status parameters, the VDM-1 can be made to **"scroll" its display upwards or downwards.** A built-in timer allows scrolling at about 4 lines per second, eliminating complicated timing program routines. At top speed, the display scrolls through a dump of 65K of memory in two minutes; that's about 1000 lines per minute!

Multiple programmable cursor circuitry is built in. All 1024 cursors can be displayed at one time or begin anywhere in the display. Thus, the VDM-1 can display white-on-black or black-on-white—perfect for many video games! The VDM-1 also features EIA Video output for any standard video monitor, or a television repair shop can easily modify your own TV set.

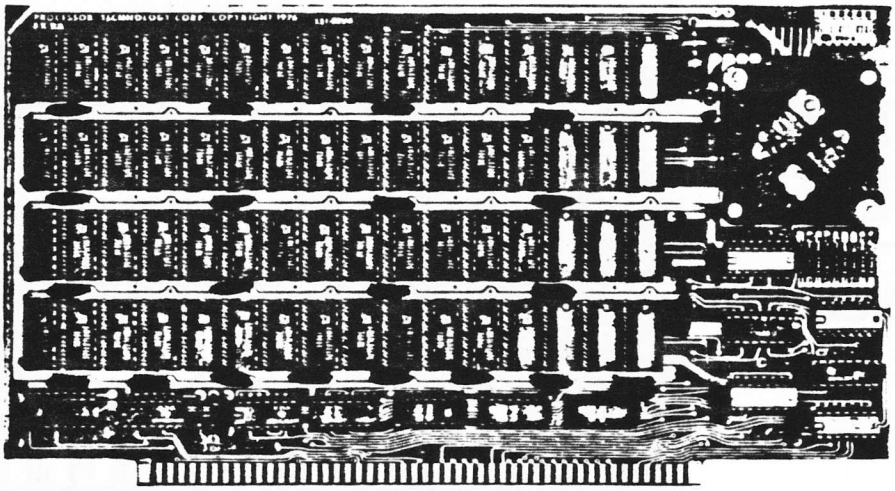
The VDM-1 comes with **free terminal mode software,** designed for teletype replacement when used with BASIC or our own Resident Assembly system. (Powerful text editing software and various game packages are also available from Processor Technology Corp.)

Our detailed VDM-1 Owner's Manual is available for $4.00, refundable with purchase of the VDM-1.

Kit Prices, $179 (premium grade, low profile IC sockets included).

nsion Principle.

or IMSAI system—with innovative, excellent quality es. [At most agreeable prices, too!]



Sharpen your Memory!

Processor Technology's new **8KRA Static Memory Module** will add double-capacity, lower power, and greater efficiency to your Altair or IMSAI system. We've actually improved on our already popular 4KRA board, by doubling the capacity **(8192 eight-bit bytes!)** and lowering the power-per-bit.

The 8KRA uses full speed, low power static RAM's—our prime supplier is AMD (91L02A's). These RAM's are manufactured to stringent military specification, MIL STD-883A, to insure reliability. **The 8KRA uses less power-per-bit than any other true static memory module**—requirements so low, that two "D" flashlight cells will retain data for 4-5 hours. (Recharging circuitry for Ni-Cad cells and battery connectors are on the card.) Also, unlike dynamic memories, static memories do not require periodic refreshing, allowing them to run at the processor's **maximum speed.**

The 8KRA occupies any 8K segment at 1K intervals within the 8080 addressing range (for example, from 3K to 11K, not just 0-8K). Card address is selected by a DIP switch, located conveniently at the top of the card. Each of the 76 Integrated Circuits has its own **premium grade, low-profile IC socket.** We've included this special feature to increase reliability, and to make assembly, testing and repairs many times easier.

8KRA 8192-word Static Memory Module **$295** (all sockets
4KRA 4096-word Static Memory Module **$154** included)

Processor Technology manufactures a full line of plug-in modules compatible with the Altair and IMSAI minicomputers. **Write us now**—we have a lot to tell you!

Processor Technology
6200-L Hollis Street
Emeryville, CA 94608

EASY-TO-BUILD BURGLAR ALARM FOR APARTMENT USE

Popular Electronics®

WORLD'S LARGEST-SELLING ELECTRONICS MAGAZINE JULY 1976/$1

MOBILE COMMUNICATIONS: CB vs. 2-METER FM

Microwave Ovens for the Home

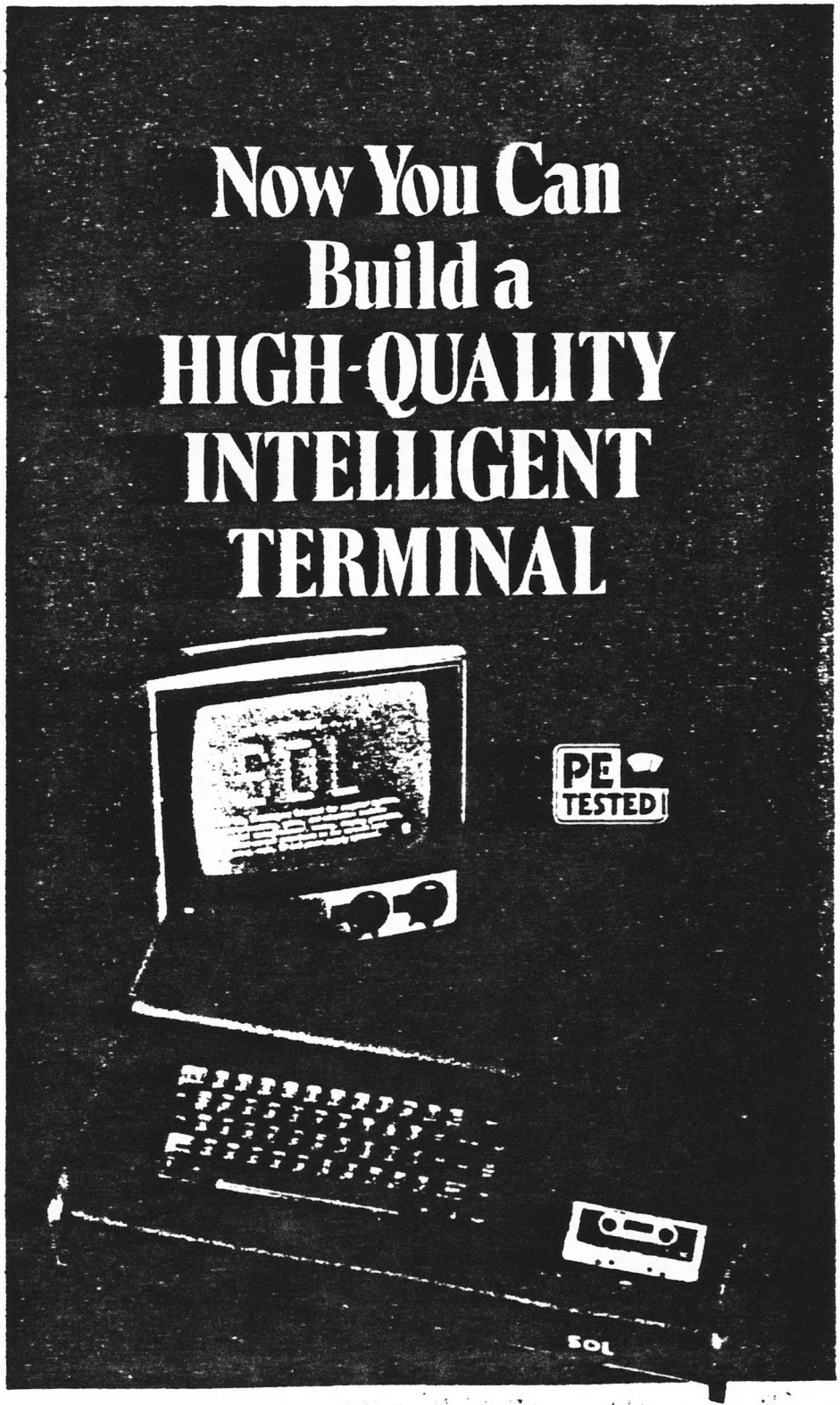
CMOS Probe Extends Multimeter Use

Guide to Choosing TV & FM Antennas

Learning Electronic Theory With Hand Calculators

TEST REPORTS:

- Nikko 7075 AM/FM Stereo Receiver
- MXR Stereo Equalizer
- SBE "Opti/Scan" Scanner
- Hickok 370 Analog Multimeter



Popular Electronics
JULY 1976

1 EPT 1 - 11.2

BUILD

SOL An Intelligent Computer Terminal

Based on an 8080 MPU, this hobbyist's computer terminal can compete with most commercial units

BY ROBERT M. MARSH
AND LEE FELSENSTEIN

THERE are essentially two types of video computer terminals in common use. The "dumb" terminal—little more than a "glass Teletype"—is a simple data transmitter receiver whose only stand-alone function is its use as a TV typewriter. The "intelligent" (also known as "smart") terminal, on the other hand, offers powerful stand-alone features. Built around a sophisticated microprocessor, intelligent terminals allow you to write, store, and edit programs for transmission to a computer or a hard-copy device. It also provides very powerful word processing at relatively low cost.

The SOL video terminal project presented here is one of the most advanced of intelligent terminals. It can interface with any mini- or microcomputer via its built-in RS-232 or 20-mA current-loop interfaces, in either serial or parallel format. It can also tie into a time-sharing computer via a telephone line and a modem (such as the Pennywhistle described in the March 1976 issue of POPULAR ELECTRONICS). In fact, it is even possible for two SOL terminals to communicate with each other without human supervision.

The key to SOL's versatility is its integral 8080 microprocessor (μP) chip. The μP operates on instructions stored in PROM's (programmable

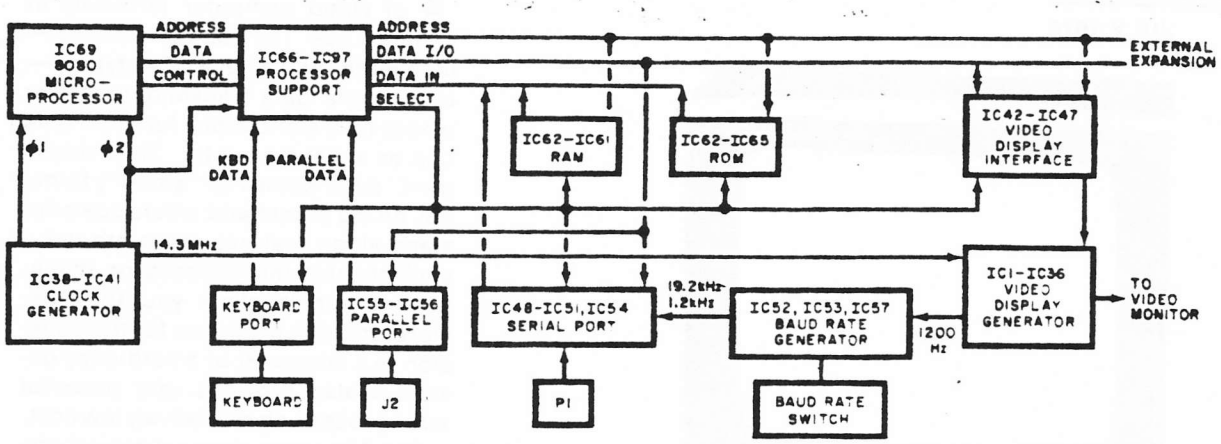


Fig. 1. Terminal accepts data from keyboard, parallel port, and RS-232 or 20-mA serial port. Output is 1 volt p-p for conventional TV requirements. Memory can be externally expanded to 65 k.

PARTS LIST

C1—10-pF disc capacitor
C2, C11, C21, C22—0.001-μF disc capacitor
C3, C7, C15, C16, C17, C18, C20, C27 through C63—0.1-μF disc capacitor
C4, C5—680-pF mica capacitor monolithic
C6—1.5-μF, 25-volt ceramic capacitor
C8, C13—1-μF, 35-volt dipped tantalum capacitor
C9, C10—15-μF, 20-volt dipped tantalum capacitor
C12—0.01-μF disc capacitor
C14, C23—680-pF disc capacitor
C19—100-μF, 16-volt upright aluminum electrolytic capacitor
C24—0.1-μF Mylar tubular capacitor
C25—0.001-μF Mylar tubular capacitor
C26—0.01-μF Mylar tubular capacitor
D1, D2, D4 through D9—1N4148 diode
D3—5.1-volt, 1-watt zener diode (1N5231B or similar)
IC1, IC8, IC11, IC12, IC23, IC39, IC73, IC74—74LS175N quad latch IC
IC2, IC79—74LS20N dual 4-input NAND gate IC
IC3—74LS86N quad exclusive-OR gate IC
IC4, IC42, IC45, IC47, IC71, IC72, IC94—74LS02N quad 2-input NOR gate IC
IC5, IC51—7406N open-collector hex inverter IC
IC6—DC4049AE CMOS hex inverter IC
IC7, IC14, IC25, IC26, IC53, IC57—74LS161 or 74LS163 4-bit synchronous counter IC
IC9—6575 MOS character generator IC
IC10—74166N 8-bit parallel-in shift register IC
IC13, IC24—CD4029AE 4-bit up/down counter IC
IC15—74L161, 74L163, or 93L16 4-bit synchronous counter IC (do not substitute)
IC16, IC93—74LS10N triple 3-input NAND gate IC
IC17—CD4001AE CMOS quad 2-input NOR gate IC
IC18 through IC21, IC29 through IC32—21L01-1 or 91L02APC MOS 1024-bit RAM IC
IC22, IC33, IC40, IC46, IC66, IC67, IC68, IC75, IC80, IC81, IC82—8T97 hex tristate buffer IC
IC27, IC48, IC78, IC95—74LS109N dual JK flip-flop IC
IC28, IC50, IC89, IC96—74LS04N hex inverter IC
IC34, IC35, IC36—74LS157N quad 2-input data selector IC
IC37—74HOOON high-speed quad 2-input NAND gate IC (do not substitute)
IC38—74SO4N Schottky hex inverter IC (do not substitute)
IC41—MH0026P MOS clock driver IC
IC43, IC87, IC90—74LS74N dual D flip-flop IC
IC44, IC83, IC86—74LS00N quad 2-input NAND gate IC
IC49, IC88—74LS08N quad 2-input AND gate IC
IC52—CD4046AE CMOS phase-locked loop IC
IC54—TR1602B, AY-5-1013, or S1883 UART IC
IC55, IC56—74173N quad tristate latch IC
IC58 through IC61—21L01 or 91L01PC 256 × 4 MOS RAM IC
IC62, IC63, IC64, IC65—S5204A or MM5204Q 512 × 8 MOS erasable PROM IC (optional; write to address below for details)
IC69—8080, 8080A, or 9080A microprocessor IC
IC70, IC77, IC84, IC91—74LS253N dual 4-input tristate data selector IC
IC76—DM8836N quad 2-input NOR gate IC
IC85, IC92—74LS155 dual 2-to-4 line decoder IC
J1—Right-angle PC mount (AMP206584-1 or DB25S)
Q1, Q2, Q3—2N2907 transistor
The following resistors are ¼-watt, 10% tolerance:
R1, R2—330 ohms
R3, R9, R10, R21, R23 through R30, R80—10,000 ohms
R4, R5, R6, R14 through R20, R22, R31 through R35, R37, R39, R41, R43, R45, R46, R48, R52, R56, R57, R58, R65, R72, R73, R74, R76, R77, R78, R79, R82, R83, R84, R89 through R98, R100, R101—1500 ohms
R7, R8—47 ohms
R36, R67, R68, R99—4700 ohms
R38, R40, R42, R47, R49, R53, R55—2200 ohms
R44, R60, R81—3300 ohms
R50, R54, R64, R87—100 ohms
R51—200 ohms
R59, R63—33,000 ohms
R61, R62, R66—1000 ohms
R69—15,000 ohms
R70, R71, R113—100,000 ohms
R75, R88—3.3 megohms
R85—75 ohms
R103, R105, R107, R109, R110, R111, R112—8200 ohms
R106, R108—39,000 ohms
R11, R12, R13—100-ohm 1-watt, 10% tolerance resistor
R86—330-ohm, ½-watt, 10% tolerance resistor
R102, R104—50,000-ohm trimmer potentiometer (Bourns No. 3352-1-503 or similar)
S1 through S4—Four-position dual in-line switch
S5—Momentary-action spst switch
S6—Single-pole, seven-position rotary switch
S7, S8, S9—Spst switch
S10—Spdt switch
XTAL—14.318-MHz, 0.01% or better tolerance, series-resonant crystal in HC18U case
Misc.—Two 40-pin, five 24-pin, 54 16-pin, and 31 14-pin IC sockets (optional); 75-ohm coaxial cable; TV monitor; ASCII keyboard; power supply; suitable chassis; mounting hardware; hookup wire; solder; etc.

Note: The following items are available from Processor Technology Corp., 6200 Hollis St., Emeryville, CA 94608: Complete SOL-PC kit of parts (does not include case, power supply, or keyboard) for $297.00. Available separately are SOL-PCB etched and drilled printed circuit board for $40.00; SOL-SS set of IC sockets for $40.00; and SOL-FAN fan for $20.00. A complete kit that includes all parts, pc board, power supply, ASCII keyboard, all cables and plugs, and a case is available for $497.00; specify kit SOL-1. **Free copies of the complete schematic, etching and drilling guide, and component placement guide are available from the same source on request when accompanied by a self-addressed stamped (26¢) envelope (9" × 12").**

read-only memories). In its basic configuration, the SOL terminal consists of a printed circuit assembly that contains the μP, 512 eight-bit bytes of PROM, 2048 eight-bit words of RAM (random-access memory), 1024-character video display generator, keyboard interface, serial and parallel interfaces for connection to external devices, and an edge connector for memory expansion. All you add are a power supply, TV receiver or video monitor, ASCII keyboard, and a case.

Since the SOL terminal is 8080 based, its memory capability can be expanded to 65k bytes. Hence, one might ask, is the SOL an intelligent terminal or a powerful microcomputer? In essence, it is both.

How It Works. The complete schematic diagram for the SOL terminal is much too large to be reproduced in this article. Therefore, a complete schematic, an etching and drilling guide, and component layout diagram for the printed circuit board are available on request simply by sending a self-addressed stamped (26¢) envelope (9" × 12") to the source given in the Parts List.

The block diagram shown in Fig. 1 will be used to explain circuit operation. Notice the similarity of this diagram to that of a conventional 8080 microcomputer. The 8080 (or 8080A or 9080A) microprocessor, *IC69*, is the "heart" of the terminal. It is supported by *IC66* through *IC97*, which include address and data line drivers and selectors; "wait state" timers; flag latches for data ports, and partial address decoding. Both address and data I/O (input output) ports are available for expansion using currently available 8080-type memory cards.

As many as four PROMS (*IC62* through *IC65*) allow up to 2048 bytes of program to be installed in the terminal. Up to 512 bytes of RAM can also be installed and are designated *IC58* through *IC61*.

SOL TERMINAL SPECIFICATIONS

Display: 16 lines of 64 characters per line. Black characters on white background or reverse.

Character set: 96 printable ASCII upper-and lower-case characters. Plus 32 control characters (optional).

Display position: Continuously adjustable both horizontally and vertically.

Cursor: Solid video inversion (switch selectable blink), cursors are programmable.

Serial interface: RS-232 and 20-mA current loop, 75 to 9600 baud, synchronous.

Parallel interface: Eight data bits for input and output; output bus is tristate for bidirectional interfaces; levels are standard TTL.

Keyboard interface: Seven-level ASCII encoded, TTL levels; requires strobe pulse with data stable for approximately 100 μs following positive edge.

Microprocessor: 8080, 8080A, or 9080A.

On-card memory: 512 bytes PROM (expandable to 2048 bytes), 1280 bytes RAM (expandable to 1560 bytes).

External Memory: Expandable to 56k bytes total ROM, PROM, and RAM.

Signal output: 1.0 to 2.5 volts peak-to-peak with composite negative sync; nominal bandwith is 7 MHz.

Power required: +5 volts at 2.5 amperes, +12 volts at 150 mA, and −12 volts at 200 mA; all buses must be well regulated.

The heart of the video display section is character generator ROM *IC9*. The generator provides both upper- and lower-case characters in a 7 × 9 dot matrix format. Descenders on lower-case characters g, j, p, q, and y go below the base line to provide true typewriter character formatting. The remainder of the IC's in the video section (*IC1* through *IC36*) produce the horizontal and vertical sync, cursor options, video inversion (black characters on white background) and all video "handshake" requirements.

The video output has a maximum bandwidth of 7.15 MHz. It contains a composite sync to allow operation with any conventional video monitor or monochrome TV receiver converted for video input (Fig. 2). Color TV receiver CRT's may not be capable of providing the resolution required for a clean video display, although the authors have obtained acceptable results using a type-approved r-f modulator to feed color receivers through the antenna input. (CAUTION: Do *not* use a transformerless video monitor or TV receiver unless a line-isolating transformer is installed.)

The *IC54* UART is used in the terminal for data transmission and reception. It is supported by *IC48* and *IC51*. Clock pulses for the UART are provided by the baud-rate generator made up of *IC52*, *IC53*, and *IC57*. Phase-locked loop *IC52* operates with dividers to produce the required clock signals. A switch is provided for setting the baud rate for 75, 110, 150, 300, 600, 1200, 2400, 4800, or 9600 baud (data bits per second). The serial port has both RS-232 and 20-mA current-loop provisions.

The parallel port consists of an eight-bit latch made up of *IC55* and *IC56*. These IC's have tristate outputs that enable their use with a bidirectional parallel data channel if desired. Signals are eight data bits wide at standard TTL levels at the input and output.

The ASCII keyboard connects to the main terminal board by a single connector that provides power to the keyboard and accepts signals from the keys. The interface requires seven-level ASCII at TTL levels and a strobe pulse with the data stable for approximately 100 μs following the positive edge.

Power for the main board must be 5 volts dc at 2.5 amperes, +12 volts at 150 mA, and −12 volts at 200 mA. The power bus lines must be well regulated.

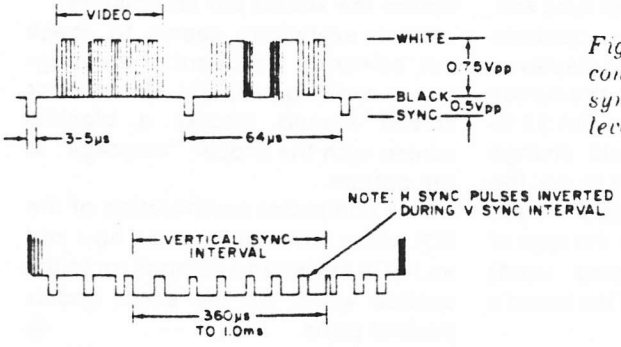
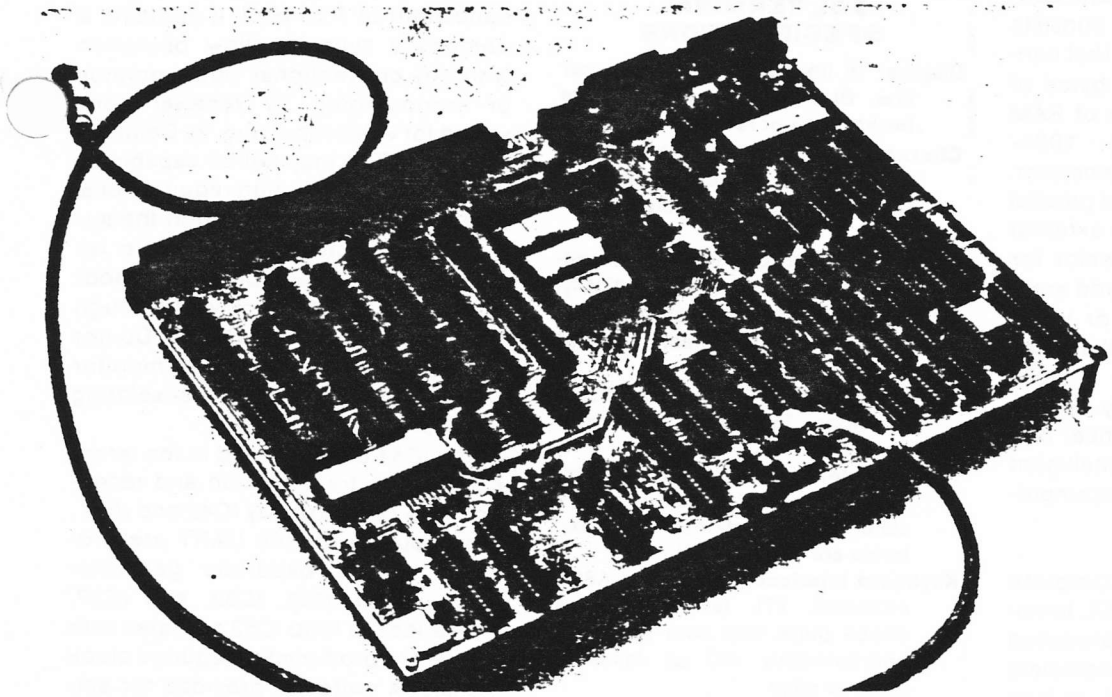


Fig. 2 Video output is conventional with negative sync and 1-volt p-p signal level.

Construction. Since the printed circuit board measures 13" × 11" (33 × 28 cm) and has numerous traces and pads that require careful registration, home fabrication of the board is not recommended unless you are highly experienced in making complex double-sided boards. Once you have the board and are ready to start mounting components, save IC installation for the last.

Start wiring the board by mounting



Large connector at left rear of assembled circuit board assembly is for external memory; coax cable is for composite video output. All input and output connectors are on rear edge.

IC sockets (recommended for all IC's to make removal and replacement easy) in place. Next, mount and solder into place the resistors, capacitors, diodes, and transistors. Then mount the baud rate switch and connectors flush to the surface of the board; make sure they do not sit askew after soldering them down.

Once the crystal is mounted and soldered into place, pass a length of bare hookup wire over its case and into the holes flanking the case. Solder the wire to the crystal's case and board pads. Install and solder into place the coaxial cable for the terminal's output.

Carefully check the board assembly for poor soldered connections, solder bridges between closely spaced pads and traces, and proper polarization of diodes and capacitors and basing of the transistors.

Checkout. Before installing any IC's, power up the circuit board assembly to verify that no short circuits exist. Measure the potential across zener diode *D3*; it should be −5 volts. Check the fine foil traces near *R85* (at the video output) for short circuits on the +12-volt line. If everything checks out, turn off the power.

Insert *IC37* through *IC41* in their sockets, making sure you properly orient them. Install jumpers from pad A to pad B and pad D to pad E (next to *IC37*). Turn on the power and use an oscilloscope to check the 47-ohm resistors next to *IC41* for the clock pulses. When you obtain the pulses, turn off the power.

Install *IC1* through *IC36*. Be particularly careful when handling *IC9* to avoid static discharges. After removing this IC from its protective foam carrier, be sure to touch the pc board with your other hand *before* bringing the IC into contact with its socket. Seat the IC carefully in its socket and gently press it home. (Note: If you encounter excessive resistance when trying to install *IC9*, replace the IC in its foam carrier. Then loosen the socket pin receptacles by repeatedly inserting and removing a non-MOS IC or piece of bare 24 gauge wire.) Install *IC9*.

Set horizontal and vertical sync controls *R104* and *R102* to midposition and the four-position dual in-line switch so that *S1* and *S4* are off and *S2* and *S3* are on. Connect SOL's video output cable to the video monitor and turn on the power to both monitor and terminal board. Displayed on the screen should be at least one line of random characters and white cursor blocks. Adjust the V and H controls on the terminal board for proper sync and the contrast and brightness controls on the monitor for the best display.

Set *S3* to off and *S4* to on; the cursor should flash at a slow rate. Set *S2* to off; the background should change from black to white. Set *S1* to on; the control characters (symbols or abbreviations, depending on the type of character generator being used) should disappear. Turn off the board's power supply

Install *IC42* through *IC50* and *IC66* through *IC97*. Practice the same precautions for *IC69*, the microprocessor chip, that you took for *IC9* above. Connect the "wait state" jumper at *IC71* from pad W to pad 1.

With the video monitor still connected to the terminal and operational, turn on the board's power. The CRT screen should display one or more lines of alternating 9 and "null" characters and should flicker every few seconds. This indicates that the µP is working. If there is any doubt, briefly operate the RESET switch. If you observe no activity on the screen, turn off and remove power from the board and check that all IC's are in their proper sockets and properly oriented.

Install *IC52* through *IC61* and program PROM *IC62*. Use the same precautions detailed above for *IC9* and *IC69* when handling and installing *IC54*, *IC58* through *IC61*, and *IC62*. Make sure that the socket for UART *IC54* is not too tight. If you encounter difficulties during insertion, use a non-MOS IC or 24 gauge wire to loosen the socket pin receptacles.

Once everything seems to check out, power up the board. If the program is running properly, the monitor screen should display a blanked screen with the proper "message" at the bottom.

This completes construction of the SOL video terminal. You can now add an ASCII keyboard and hook up to the outside world via the serial and/or parallel ports. ◈

Advertisement from Byte #1 Sept. 1975

1 EPT 1 - 11.6

ALTAIR 8800 USERS!

Did you know...

- That all our modules are 100% compatible with the Altair 8800 computer, NO modifications necessary!
- That our 4KRA Static Read/Write Memory module doesn't have to lose it's data when you pull the plug!
- That our 3P+S Input/Output module will fully interface two TV Typewriters with keyboards and a modem or teletype at the same time!
- That we make the most powerful alphanumeric Video Display module anywhere!
- That our software is FREE, or close to it!
- That all our modules are truly high quality, computer grade, but that our prices are the lowest in the industry!
- That we have already shipped hundreds of modules on time, and we will continue to deliver what we promise, FAST!

CHECK THE SPECS:

4KRA Static Read/Write Memory

This 4096 word STATIC memory provides faster, more reliable and less expensive operation than any currently available dynamic memory system. The 4KRA permits Altair 8800 operation at absolute top speed continuously. All RAM's (Random Access Memories) used in the 4KRA are 91L02A's by Advanced Micro Devices, the best commercial memory IC on the market today. 91L02A's require typically 1/3 the power of standard 2102 or 8101 type RAM's and each one is manufactured to military specification MIL STD-883 for extremely high reliability. These memories can be operated from a battery backup supply in case of power failure with very low standby power consumption. (Ask for our technical bulletin TB-101 on power down operation.) In short we have done everything we could to make the best 4K memory module in the computer field, and because we buy in large quantity, we can make it for a very reasonable price. Available now.

2KRO Erasable Reprogrammable Read Only Memory Module

With this module the Altair 8800 can use 1702A or 5203 type Erasable Reprogrammable ROM's. The 2KRO accepts up to eight of these IC's for a capacity of 2048 eight bit words. Once programmed this module will hold its data indefinitely whether or not power is on. This feature is extremely useful when developing software. All necessary bus interfacing logic and regulated supplies are provided but NOT the EPROM IC's. Both 1702A and 5203 PROM's are available from other advertisers in this magazine for well under $25. Available now.

3P+S Input/Output Module

Just one 3P+S card will fulfill the Input/Output needs of most 8800 users. There are two 8-bit parallel input and output ports with full handshaking logic. There is also a serial I/O using a UART with both teletype current loop and EIA RS-232 standard interfaces provided. The serial data rate can be set under software control between 35 and 9600 Baud. You can use your old model 19 TTY! This module gives you all the electronics you need to interface most peripheral devices with the Altair 8800, it's really the most useful and versatile I/O we've seen for any computer. Available now.

MB-1 Mother Board

Don't worry any more about wiring hundreds of wires in your Altair to expand the mainframe. Our single piece 1/8-inch thick, rugged mother board can be installed as one single replacement for either three or four 88EC Expander cards, so you don't have to replace your already installed 88EC card if you don't want to. The MB-1 has very heavy power and ground busses and comes with a piece of flat ribbon cable for connection to the front panel board of the 8800. Available now.

VDM-1 Video Display Module

This module is the first real computer terminal display in kit form. Under software control the VDM-1 displays sixteen 64 character lines to any standard video monitor. Characters are produced in a 7x9 dot matrix, with a full 128 character set, upper and lower case plus control characters. Data is accessed by the VDM as a block from any 1K segment within the 65K address range of the 8800 computer. Multiple cursors are completely controlled by software and the display can begin anywhere on the screen (this is great for many video games). When the last line is filled the display scrolls up a line. Powerful editing capabilities are provided with the FREE software package included in every VDM-1 kit. Available in September '75.

SOFTWARE

Our Assembler, Text Editor and System Executive is being [illegible]. This software package gives you very powerful [illegible] language capability in the Altair 8800. The Executive and Editor allow you to call programs by name (including BASIC) and then add, delete, change, or list programs by line number. The Assembler provides a formatted symbolic mnemonic listing as well as octal or binary object code from Assembly Language programs written using the Editor. The Assembler also gives valuable error messages to help in debugging those inevitable errors. The Assembler, Editor, Executive Package No. 1 will be available in read only memory along with an expanded Executive and a powerful Interpretive Simulator by October or November of 1975.

We are working on two BASIC Language packages which should be ready by October. One will be a basic BASIC needing about 8K of memory as a minimum and the other will be an Extended version with additional string manipulation, matrix operations and double precision arithmetic capabilities requiring about 12K. Both these packages will be available in Read Only Memory for a reasonable price.

PRICE LIST

Item	Kit	Assembled	Delivery
2KRO EPROM module	$ 50.	$ 75.	2 weeks ARO
3P+S I/O module	125.	165.	3 weeks ARO
4KRA-2 RAM module w/2048 8-bit words	135.	185.	2 weeks ARO
4KRA-4 w/4096 8-bit words of RAM	215.	280.	2 weeks ARO
RAM only, AMD 91L02A 500n sec low power	8/$40	–	2 weeks ARO
MB-1 Mother Board	35.	–	2 weeks ARO
VDM-1 Video Display module	160.	225.	Sept. 30, '75 then 3 weeks ARO

Send for our FREE flyer for more complete specifications and for pricing on additional items.

TERMS: All items postpaid if full payment accompanies order. COD orders must include 25% deposit. MasterCharge gladly accepted, but please send us an order with your signature on it.
DISCOUNTS: Orders over $375 may subtract 5%, orders over $600 may subtract 10%.

Processor Technology Corporation
2465 Fourth Street
Berkeley, Ca. 94710 (415) 549-0857

1 EPT 1 - 11.7

Processor Technology
2465 Fourth Street
Berkeley, Ca. 94710

BERKELEY, CALIF.

FIRST CLASS MAIL
U.S. POSTAGE
PAID 1 oz.
PERMIT 1460

First literature
Processor Technology
ever sent out!!

T.L. Todd
Box 319
Platteville, WI 53818

REQUESTED INFORMATION

Could you use a Resident Assembler, Editor and Monitor for the 8800?

We have them!

GET TO KNOW THE FULL POWER OF YOUR ALTAIR!

PROCESSOR TECHNOLOGY CORP. provides the hardware AND the software you really need to accomplish things with your Altair 8800.

Want specifics?

- The only CRT terminal display module ever offered that is actually part of the computer
- A reasonably priced, LOW POWER, 4K Memory
- A versatile input/output module designed to meet most common I/O requirements with *one card*
- A reprogrammable read only memory (EPROM) module
- A rugged full length mother board with improved power distribution and integral bus terminator and card cage
- FREE SOFTWARE, or close to it!
- Firmware ROM modules containing our Software programs

FEATURES:

- Full 8800 hardware and software compatibility: JUST PLUG IN!!
- Maximum versatility: each card can be used in the widest range of applications.
- Conservative thermal design for long life and reliable operation.
- Maximum noise immunity: hysteresis bus receivers on noise prone high speed inputs.
- Highest possible quality: No compromise design!
- 100% Guaranteed: six month warranty.

READ ON . . .

Each of our kits contains all necessary parts including double sided epoxy PC board with plated through holes and gold contacts, all IC's, voltage regulators, massive heatsink, and mounting hardware. Complete instructions are included for easy asser and checkout. Every kit and assembled unit also includes information on how to use the module most effectively, covering soft programming considerations as well as hardware connections.

THE DETAILS . . .

FAMILY SPECIFICATIONS:

Bus connection: Fully 8800 compatible, electrically and mechanically. Outputs to the 8800 bus are three state, standard TTL le
Storage temperature range: −40 to +85° C
Operating environment: 0 to 50° C, humidity to 95%, non condensing.

ALS-8

Ever dream of just turning on your computer and having its full power available instantly? The ALS-8 Firmware Module gives you that kind of "turn on the switch" system with 4K of PROM's pre-programmed with a greatly *expanded* version of our Software Package No. 1.

This powerful file oriented package is the finest Resident Assembly Language Operating System available today. 8080 assembly language programs can be stored as named files (5 letter names) and called at will to be assembled, edited, or listed by line number. Six files can reside in memory at once, or, they can be stored in either source or object code on paper tape, cassette or any other form of storage medium for re-input at a later time. Assembly of the source file can take place from a memory file or from any input device you choose.

The assembler converts the mnemonic codes and labels of the sembly source language program to object code at any address choose. (The run address can be different from the location re the code is placed.) Symbolic addressing includes the ty to chain common symbols from one program to another though the other program was assembled long ago. Fea- of the assembler include labels, comments, expressions, ants, relative symbolic addressing . . . in short a profes- system just made for assembly language work.

Input/output for the program is so versatile that it defies imagination, the driver software can even change the configuration under program control. That's right . . . a pr can output one batch of information to one port and a diffe batch to another port, each seemingly at the same time with human intervention. The system can handle any I/O config tion you can hook up and will call to any driving software wish to implement for the I/O.

Also, if you are new to computers, the manual included w each module not only explains how to use the program (w examples), but goes on to explain each routine of the Syst and how to call these routines from other programs you h written. This combined with the I/O handling along wit custom "command table" allows every ALS-8 to be customi within your system to your individual needs. You don't have worry about being locked into a ROM version, this one designed with your expansion and creativity in mind.

The ALS-8 Firmware comes complete with a module with components capable of holding 8K of TURN ON THE SWIT program. So why 8K? Because we wrote this system to upward compatible with future software developments and first of these, an interpretive simulator, is up and running no

LOW POWER . . . yes and don't be fooled by "undirected" statements about this subject. All RAM's used in the 4KRA are 91L02A's by Advanced Micro Devices (AMD). 91L02A's require typically 1/3 the power of standard 2102 or 8101 type RAM's and under worst case conditions draws only 30% more than any currently available dynamic memory. In addition each RAM is manufactured to military specification MIL STD-883. This all adds up to low power and reliability, notice . . . we're the ones who publish the maximum worst case power required by our memory.

Plus, our memory draws so little standby current that they can be operated from a battery back up in case of power failure. This allows long term retention of data under loss of power conditions. Our TB-101 illustrates how to add a few simple components to implement this feature and is shipped with every 4KRA sold. In short we've done everything we could to make this the best 4K memory module in the computer field.

SPECIFICATIONS: 4KRA

Maximum capacity: 4096 eight bit words
Operating mode: Static
Access time: 520 nsec. maximum
Cycle time: 520 nsec. maximum, read or write
Bus pinout: Plug-in compatible with Altair 8800 bus
Edge contacts: Gold plated, 100 pins (dual 50) on .125" centers
Power requirements: +8 to +10 VDC, 1.2 A Max, at 0°C, 0.9 A Typical.
Dimensions: 5.3" x 10.0" (13.46cm x 25.4cm)

2KRO EPROM MEMORY MODULE

Read only memories (ROM's) do not lose their stored data when power is removed and thus have an advantage (we specialize in understatements) when used in stored program applications. Some ROM's, called PROM's, are user programmable, and some PROM's, called EPROM's, are erasable and reprogrammable as well. The most common and least expensive types of EPROM's in use today are the 1702A and MM5203; both are organized as 256 x 8 bit arrays. Our 2KRO module will accept up to eight of these IC's, providing the user with up to 2048 eight bit words of non-volatile storage for monitor, executive, loader, and other programs.

Each 2KRO module is jumper selectable to fit any one of thirty-two 2K segments within the 65K addressing range of the 8800 computer. Additional jumpers are provided to select the appropriate number of "wait" states as determined by the access time of the EPROM's used. The EPROM's are not provided (everything else is) but are readily available for reasonable prices on the industrial and surplus markets. The module will accept either of two types of EPROM: the 1702, 1702A, MM5202 or 9702 made by AMD, Intel, MIL and National, or the MM5203 made by National. However, both 1702 and 5203 types cannot be used at the same time.

Erasing and programming services for the 1702, 1702A or 02 will be provided by Processor Technology Company. A ly printed or typewritten listing in octal (base 8) code of program or data must be submitted along with address orr There will be a charge of $5.00 per EPROM ogr For additional information and pricing on our RC for our programming bulletin.

SPECIFICATIONS: 2KRO

Maximum capacity: 2048 eight bit words
Operating Mode: Static
Access & Cycle Time: Dependent of EPROM used, logic on card will work over an access time range of 30 to 2500 nsec. Number of "wait" states jumper selectable from zero to four.
Voltage Requirements: +8 to +10VDC, −15 to −18VDC
Power Requirements: 10W max., 8W typical at 25°C with 8 1702A's. Replacement transformer available for full negative supply power (see price list).
Bus Pinout: Plug-in compatible with Altair 8800 bus
Edge contacts: 100 pins (dual 50) on .125" centers, gold plated
Dimensions: 5.3" x 10.0" (13.46cm x 25.4cm)

3P+S INPUT/OUTPUT MODULE

Getting data into and out of a computer can be one of the most difficult and expensive tasks in bringing up a working system. Our 3P+S module was designed to provide maximum versatility to allow this one card to meet all the I/O needs of most 8800 system users. For example, one teletype and two TV Typewriters with keyboards can operate simultaneously with the 8800 via one 3P+S module; or, one TV Typewriter, an EIA RS-232 modem, a teletype and another parallel data device can be fully interfaced at the same time.

In addition, one parallel output port is available to be used for setting up control conditions for both parallel and serial ports, as well as to set the serial I/O baud rate under program control. The Baud rate can be set between 35 and 9600 Baud and the module is the only one available that will allow 1.5 stop bits as required by the old model teletypes that are available at such low cost.

Also, one parallel input port is available for polling the Input Data Available flags and External Device Ready flags, as well as for checking the serial I/O error flags. Full handshaking with both input and output peripherals can be implemented with these provisions.

Interfacing to the 8800 vectored interrupt bus is provided on the card as a jumper selectable option, allowing any of the UART (Universal Asynchronous Receiver Transmitter) error flags or handshaking signals to be used to generate interrupts. The Vectored Interrupt Module is required for this purpose.

Addressing of the module is jumper selectable to any one of 64 four address segments within the 8800 range of 256 I/O addresses. Additional flexibility allows either the UART and control port, or the two parallel ports to occupy the lower two relative addresses.

Complete information on each of the options available is included with each 3P+S sold. In addition, a letter to us describing your system configuration will be returned illustrating the best way to implement the system with our module.

SIM-1 EXPANSION FIRMWARE

INTERPRETIVE SIMULATOR????? Yes, an interpretive simulator that runs 8080 programs, in an interpretive mode, on e same 8080 that contains the Simulator. This isn't just a ebug program, because the program thinks it's an 8080! That's right, all registers, flags, program counter, and stack are SIMULATED! Breakpoints are in the simulator program, not a jump to a restart location. (How else would you debug a program that needed the restart location?) Also, when a call or jump instruction is encountered, full control remains with the simulator. This is the only system that will not lose control if a program error is made on this type of instruction.

All registers, flags, program counter, stack and memory contents can be changed at all times and there is even a single ep mode that allows full printout of all the registers, flags, after each instruction is executed.

Input instructions within the simulated program can be handled three ways: Real Time, Simulated, and Pre-Set. In the simulated mode the simulator stops and asks "what input for input port X." You input from your regular input port for any of the 255 other input ports! Output instructions are also simulated directly to you from the simulator. The output to your regular output port identifies the data output as well as the port it would have been output to. Again, don't mistake this for a simple program that handles breakpoints, this simulator actually gets each instruction and runs it under simulator control.

The SIM-1 comes pre-programmed and ready to plug into the ALS-8 module. The ALS-8 is an integral part of the simulator and must be available for its operation. (It would have been a 6K program if the customizing power of the ALS-8 hadn't been there.)

As you can see, this without a doubt is the most powerful program development tool that has ever been resident on an 8080 to this day. It's running now at Processor Technology and a sample print-out illustrating its operation is available on request.

Will paper tapes of the ALS-8 software be available? We imagine they will appear but . . . using 8K of RAM to receive a paper tape that takes the better part of a half hour to load each time a power glitch occurs just doesn't make a real system. Using this module, loaded with the ALS-8 "PERMANENT PROGRAM" will save that expensive RAM for data and development programs as well as giving the true convenience of a "turn-on-the-switch" system.

SOFTWARE NO. 1 FIRMWARE

We have had a number of inquiries about providing Software No. 1 as a PROM Module, so we decided to provide one. The Software No. 1 Firmware Module is the ALS-8's little brother. Software No 1, as we call it, contains the basic resident assembler, system monitor, and line oriented text editor. It has these features: 1) Teletype oriented I/O (i.e. only one I/O device permitted). 2) RAM resident files, perfect for small ograms, but difficult for files larger than about 1K Byte. 3) tware No. 1 Firmware cannot assemble from files not in 1. 4) It is not upward compatible with Disc or Cassette rat... Systems or the Simulator.

SPECIFICATIONS: ALS-8, SIM-1, & SW1F

MEMORY TYPE: 4096 word Erasable PROM's
CYCLE TIME: 1.0 micro-second worst-case
ACCESS TIME: same as cycle time
POWER REQUIREMENT: +8 to 10 VDC at 300 Ma maximum, −14 to −17 VDC at 300 Ma maximum.
BUS PINOUT: Plug-in compatible with Altair 8800 bus.
EDGE CONTACTS: 100 pins (dual 50) on .125" centers
DIMENSIONS: 5.3" x 10.0" (13.46 x 25.4 cm)

Do you get Keyboard?

VDM-1 VIDEO DISPLAY MODULE

Build a smart terminal into your Altair! Your Altair already has the intelligence, we provide the display module. This is not a limited "TV TYPEWRITER" but an ultra-high speed computer terminal built into your computer. The VDM-1 generates sixteen 64 character lines from data stored in the 1K Byte on-card memory. Alphanumeric data is shown in a 7x9 dot matrix with a full 128 upper *and* lower case ASCII character set. The VDM-1 features EIA Video output for any standard video monitor. (A TV set can be easily modified at your local TV repair shop.) Multiple programmable cursors, automatic text scrolling and powerful text editing software are included FREE!

SPECIFICATIONS: VDM-1

CHARACTER SET: 96 character ASCII, plus control characters 7 x 9 matrix with descending lower case.
DISPLAY FORMAT: 16 lines, 64 characters per line.
DISPLAY MEMORY: 1024 8-bit Bytes LOW POWER Static Read/Write memory
DISPLAY POSITION: Continuously adjustable, horizontally and vertically
I/O, DATA: Addressable as a 1K page of memory, Read or Write.
CURSOR: Solid video inversion (blinking optional) appears at all character positions when bit 7 is high.
I/O, CONTROL: Output from CPU is one 8-bit Byte. The lower four bits control Beginning Line Address; the upper four bits control Beginning Display Offset. Input to CPU is a one bit (DI), Parameter Change Ready flag.
BLANKING CHARACTERS: CR (octal 015) blanks test, except cursor, to end of line. VT (octal 013) blanks text, except cursor, to end of screen.
SIGNAL OUTPUT: 1.0 to 2.5 VP-P video composite, negative sync
MONITOR BANDWIDTH: 6.0 MHz (at −3dB) required video BW.

4KRA LOW POWER STATIC MEMORY MODULE

This 4096 word read/write memory module provides faster, less expensive, more reliable and lower power operation than any comparable memory module sold today.

Faster speed in actual operation results because static memories don't need periodic refreshing. Dynamic memories require a refresh period at least 500 times per second. During this period no data can be written or read from the memory and the computer must sit idle in a "wait" condition.

SPECIFICATIONS: 3P+S

Outputs: Two 8-bit parallel ports, standard TTL levels, relative addresses at 0 & 1, or 2 & 3.
One Teletype 20mA current loop output.
Four EIA RS-232C outputs for serial transmit data and/or control signals.
One Peripheral Interface Control driver (PIC) 50mA current source for paper tape reader control or cassette recorder control. Jumper selectable to control port output.

Inputs: Two 8-bit parallel ports, standard TTL voltage levels, input current is 0.36mA max.
One Teletype 20mA current loop receiver for UART data input.
Four EIA RS-232C receivers for received serial data and/or control signals.
Control: One 8-bit output port, relative card address selectable as 0 or 2. Lower four bits for baud rate control and/or EIA control outputs and/or PIC driver.
Upper four bits for UART control, i.e. word length, parity, and number of stop bits. Control conditions can be strapped on, off or to software controlled, latched output bits.
One 8-bit input port, relative card address selectable as 0 or 2. Bits selectable with jumpers to read UART error flags, i.e. parity, overrun, and framing errors, and/or EIA control inputs, and/or Data Available flags for parallel input ports, and/or External Device Ready flags for parallel output ports.

Interrupt Control: Any control input, status flag, or UART output may be jumpered to the Interrupt Bus Driver. Interrupt operation requires use of a Vectored Interrupt Module to gate the Restart instruction to the processor.

Bus Pinout: Plug-in compatible with Altair 8800 bus

Edge Contacts: Gold plated, 100 pins (dual 50) on .125" centers.

Voltage Requirements: +8 to +10VDC, +15 to +18VDC, −15 to −18VDC

Power Requirements: 7.0W maximum, 5.5W typical at 25°C

I/O connection: Two standard 44 pin (dual 22) edge connectors, .156" centers

Dimensions: 5.0" x 10.0" (12.7cm x 25.4cm)

MB-1 MOTHER BOARD

Don't worry any more about wiring hundreds of wires in your Altair to expand the mainframe. Our single piece 1/8-inch thick, rugged mother board can be installed as one single replacement for either three or four 88EC Expander cards, so you don't have to replace your already installed 88EC card if you don't want to. The MB-1 has very heavy power and ground busses and comes with a piece of flat ribbon cable for connection to the front panel board of the 8800, a built-in bus terminator, and card guide cage for sixteen plug-in slots.

Specify MB-12 for replacement of three 88-EC's, or MB-1 for placement of four 88-EC's. Both are the same price. Available v.

ensions: .125" x 11" x 14.5" (0.32cm x 27.94cm x 3€

WWB – WIRE WRAP BOARD

This is the card for all of you who do wire wrap prototyping. Now you can easily create your own custom interfaces or strange "Kluges" of any kind. The WWB has a "universal" pattern of seven rows of pads on .3" centers, so that standard 14, 16, 24, and 40 pin DIP IC sockets can be plugged in. Power and ground are dedicated to pins 16 and 8 respectively (i.e. for 16 pin DIP's) but the layout is designed for fast conversion of each position to other IC sizes. Up to 62 sixteen pin DIP IC's can be used. An extra six wire wraps socket positions have been set aside for connections to the 8800 bus.

Each WWB Kit comes with one 5 volt regulator, a heat sink, and decoupling capacitors. Space is reserved on the card for two more regulators for positive or negative supplies.

Dimensions: 5.3" x 10.0" (13.46 x 25.4 cm)
Contacts: Gold plated, 100 pins (dual 50) on .125" centers
Board Materials: 1/16" G-10 glass-epoxy, plated through holes, 2 oz. copper: solder plated.

EXB – EXTENDER BOARD

This Extender Board makes troubleshooting and servicing of any 8800 compatible module much easier. With the EXB you can plug in a troublesome module five inches above the Mother Board so that both sides of the Board can be reached easily with an oscilloscope, VTVM, or logic probe.

Dimensions: 5.3" x 10.0" (13.46 x 25.4 cm)
Contacts: Gold plated, 100 pins (dual 50) on .125" centers.
Edge Connector: Viking 3VH50/1CV (one included).

Software

Package No. 1: This is the self contained Assembly Language Operating System developed and distributed by Processor Technology for an "almost free" $3.00. The $3.00 package includes the source listing with hexidecimal object code and ample comments throughout. Each command is described and six pages of sample use are included. Paper tapes of this System were sent to Computer Clubs across the nation and are available from them on request. Get the listing and the tape and you will be running one of the finest Assembly Language Systems developed to date.

Package No. 2: We are working on two BASIC Language packages which should be ready by December. One will be a basic BASIC needing about 8K of memory as a minimum and the other will be an Extended version with additional string manipulation, matrix operations and double precision arithmetic capabilities requiring about 12K. Both these packages will be available in Read Only Memory for a reasonable price.

write for free listing

Up & Coming

THE NEWS IN A NUTSHELL!

The last edition of Nutshell produced an overwhelming stream of input to our product development department. This input did not go to thin air as you see.

MASS STORAGE

We have always wanted a low cost, reliable, fast access storage device using standard Phillips cassettes (we bet you have too), so we got to work and designed one – here it is! With the CDS-VIII Cassette Data System you have computer controlled access to 128K bytes of data within 20 seconds when using C-30 cassettes. We provide read/write electronics and transport controller, Altair interface, a case and power supply, and one or two multiple motor cassette transports plus FREE driving software! Yes, up to two cassette drives! Two drives provide much more powerful file handling and copying capabilities as well as, of course, twice the storage capacity. Data can be written and/or read asynchronously at any transfer rate up to 150 bytes/sec: at this rate 8K BASIC can be loaded in about 50 seconds! We have also included provision for use of any read/write electronic plug-in section so that tapes using HIT, Computer Hobbyist or Digital Group formats may be read at lower data rates. Full details will be available in November, 1975.

PTCOS

What is PTCOS you may ask? It stands for Processor Technology Cassette Operating System and it means a real Operating System program based around our CDS-VIII dual Cassette Data transport System. When operating under this program you have true file handling power to create, delete, edit, relocate, and copy all kinds of files (e.g. BASIC and programs written in BASIC). PTCOS can handle multiple I/O devices using a special type of file and suitable small driving routines. At last an integrated system concept for the 8800 is a reality! PTCOS is devilishly similar in its basic operation to an FDOS and is upward compatible with future software developments from Processor Technology.

TTK

The True Time Keeper is alive and well but had to take lower priority than the VDM & CDS. It will be ready about December or January and will include standby power. Complete facilities for vectored interrupt and the very necessary functions of a real time clock will be included along with the month, day, hour, minutes, seconds. . . . We're sorry it takes so long but this one is well worth waiting for. I WANT ONE, anyone?

MMM-1

Everyone was so firm about the CDS and VDM that the MMM's received hardly a mention. Our business guy insists on seeing the "I WANT ONE'S!" before he cuts us loose. Remember these modules really will play honest music under program control. Two tones on the first module with full program control of amplitude and frequency. Any other ideas?

PPM PROM PROGRAMMING MODULE

Did we really ask if you wanted both or "just" one or the other. We will make one of these as soon as we know the way pricing is going to go on the 5203 and 1702A's.

Warranty

PROCESSOR TECHNOLOGY COMPANY, in recognition of its responsibility to provide quality components and adequate instruction for their proper assembly, warrants its products as follows:

All components sold by Processor Technology Company are purchased through normal factory distribution and any part which fails because of defects in workmanship or material will be replaced at no charge for a period of 6 months following the date of purchase. The defective part must be returned postpaid to Processor Technology Company within the warranty period.

Any malfunctioning module, purchased as a kit and returned to Processor Technology within the warranty period, which in the judgement of P.T.Co. has been assembled with care and not subjected to electrical or mechanical abuse, will be restored to proper operating condition and returned, regardless of cause of malfunction, with a minimal charge to cover postage and handling. (This is really a good deal, we think.)

Any modules purchased as a kit and returned to P.T. Co. which in the judgement of P.T. Co. are not covered by the above conditions will be repaired and returned at a cost commensurate with the work required. In no case will this charge exceed $20.00 without prior notification and approval of the owner.

Any modules, purchased as assembled units are guaranteed to meet specifications in effect at the time of manufacture for a period of at least 6 months following purchase. These modules are additionally guaranteed against defects in materials or workmanship for the same 6 month period. All warranted factory assembled units returned to P.T.Co. postpaid will be repaired and returned without charge.

This warranty is made in lieu of all other warranties expressed or implied and is limited in any case to the repair or replacement of the module involved.

Processor Technology
2465 Fourth Street
Berkeley, Ca. 94710

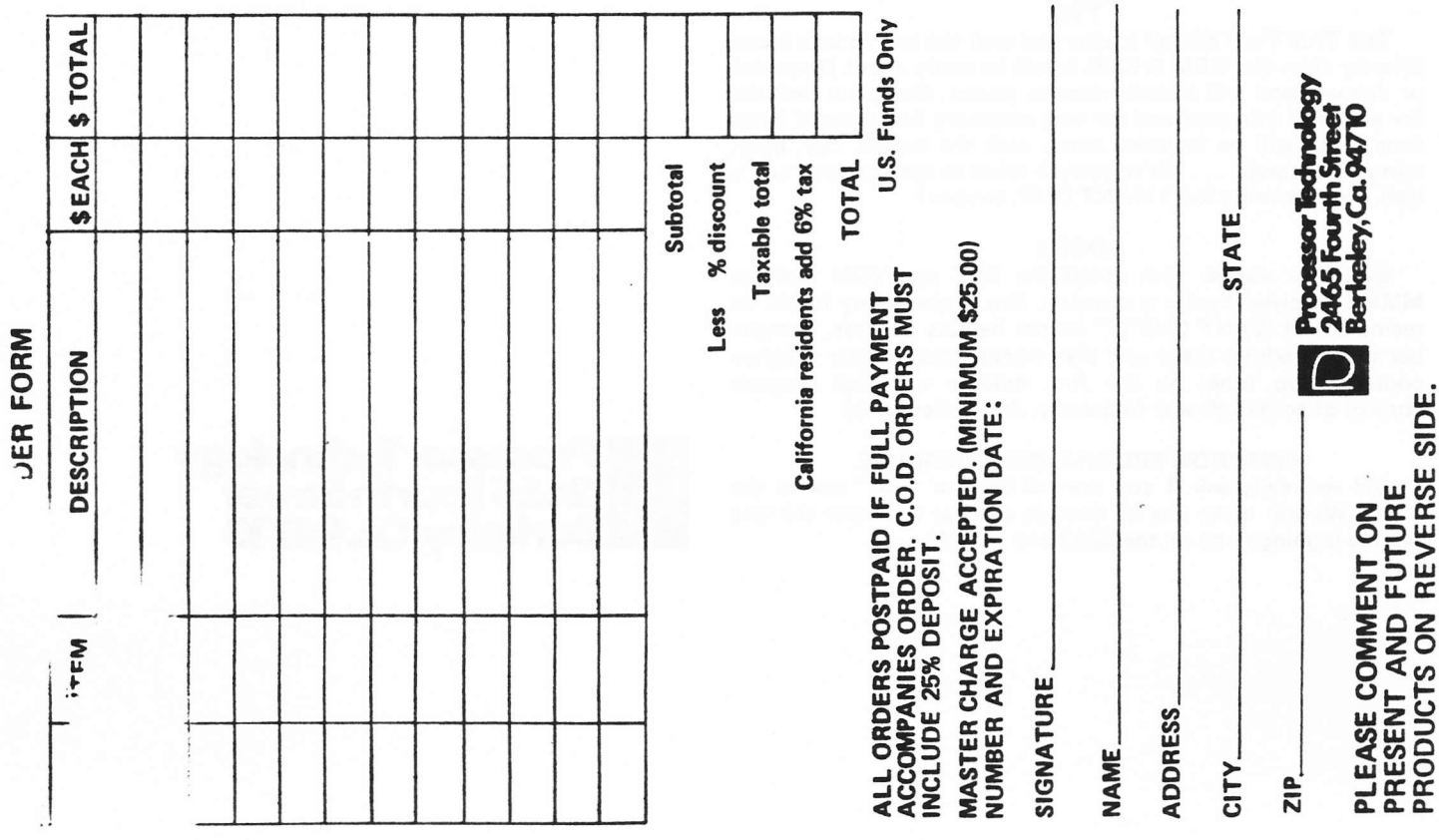
PRICE SCHEDULE effective December 1, 1975

ITEM	DESCRIPTION	KIT PRICE	ASSEMBLED, TESTED	AVAILABILITY
2KRO	Erasable PROM Module	$ 50.00	$ 75.00	NOW
3P+S	Three Port Input/Output Module	125.00	175.00	NOW
4KRA-4	4096 Word Low Power, Static RAM Memory RAM only, 1024 x 8	139.00 25.00	195.00 –	NOW
SW1F	Software No. 1 Firmware	189.00	245.00	NOW
ALS-8	Assembly Language Operating System Firmware Module	250.00	350.00	DEC. 15
SIM-1	Simulator Expansion Module	95.00	100.00	DEC. 15
VDM	Video Display Module	160.00	225.00	DEC. 15
EXB	Extender Board Module	35.00	45.00	NOW
WWB	Wire-Wrap Prototyping Module	40.00	–	NOW
MB-1/CC	16-slot Mother Board with Bus Terminator and Card Cage (includes BT-1 & CC 16)	70.00	–	NOW
CC-16	16-slot Card Cage for MB-1	20.00	–	NOW
44PC	Dual 22 pin edge connector (.156"), two required for use with 3P+S	3.00	–	NOW
100PC	Dual 50/50 pin edge connector (.125" centers) ONE REQUIRED FOR EACH MODULE TO PLUG INTO 88EC or MB-1	8.50	–	NOW
DB25S	Socket, 25 pins for I/O connections on 8800 rear panel	4.50	–	NOW
DB25P	Plug and hood, 25 pins for cables plugging into DB25S	4.50	–	NOW
TRF-1	High current transformer for PROM module replaces T3 in 8800	5.00	–	NOW

DELIVERY: From stock to 3 weeks after receipt of order.
TERMS: All items postpaid to U.S. and Canada if full payment accompanies order. COD orders must include 25% deposit.
DISCOUNTS: Prepaid and COD orders over $375 subtract 5%; orders over $600 subtract 10%.

Prices, specifications, and delivery subject to change without notice.

PROCESSOR TECHNOLOGY CO., 2465 Fourth Street, Berkeley, Ca. 94710 Phone: (415) 549-0857



BYTE September 1976 pp. 24,25 1 EPT 1 - 12

How Far Can You Go

Find out–by adding 8080 compatibl

3P+S Input/Output Module

Our 3P+S is a simple inexpensive answer to handling a variety of peripheral devices. It has **two 8-bit parallel I/O ports,** with full handshaking logic. Plus, it has a **serial I/O port,** with a data rate range from 35 to 9600 Baud. Simultaneously handle a paper tape reader, a keyboard, a TV terminal, a tape punch, **and** a telephone coupler! Available with premium grade low-profile IC sockets, **$149.** Owner's Manual, **$4.95.***

ALS-8 Assembly Language Operating System

Just turn on the switch, and instantly you have the power to **write, edit, assemble, de-bug, and run your own programs.** The **ALS-8** is the most **useful** software development tool available today. Optional firmware includes **SIM-1,** an interpretive simulator, that simulates programs without running them in real time. Thus, errors encountered during testing do not endanger your entire system. The **TXT-2** firmware adds the dimension of text editing. Insert, delete, move entire lines or single characters, and much more! **ALS-8** (assembled only), **$425. SIM-1, $95. TXT-2, $95.**

VDM-1 Video Display Module

If you're setting up just the system you really want, don't fall short by limiting its communicating ability. The **VDM-1** is an **ultra-high speed** video output device. Its **16 display lines** have **64 characters each, upper and lower case.** 1024 bytes of random access memory are on the card. It scrolls up or down, **even to 2000 lines per minute!** Any combination of the 1024 cursors can be displayed as black-on-white or vice versa. **Free terminal mode software is included,** along with premium grade, low-profile IC sockets. **$199.** Owner's Manual, **$4.95.***

8KRA Static Memory Module

We now offer a low-power static memory module, with a **full 8192 bytes of memory.** It has an exclusive KSET™ address selection DIP switch, so you can conveniently set address boundaries in increments of 1K. **Our low-power RAM's typically require one-third less power** than those commonly used by our competitors. They'll even retain memory for 4-5 hours when powered by two "D" flashlight cells. **On-board recharging circuitry and battery connectors** make it possible to protect your data against sudden power loss. Each RAM has its own IC socket, too, for easier assembly and repair. **$295.** Owner's Manual, **$4.95.***

2KRO EPROM Module

The **2KRO Read Only Memory** will accept up to eight 1702A or 5203 EPROM's (not included), providing 2048 eight-bit words of non-volatile storage for monitor, executive, loader and other programs. Programming services available from your dealer or write us for details. **$65.** Owner's Manual, **$4.95.***

4KRA Static Memory Module

The 4KRA (4096 bytes) was our first static memory module. It's still very popular, and uses the same **low-power static RAM's** as the 8KRA. Plus, we've added a **DIP switch,** and **every RAM now has its own premium grade, low profile IC socket.** On-board **recharging circuitry** (with battery backup) makes it possible to retain memory for 8-10 hours during power failure. **$159.** Owner's Manual, **$4.95.***

MB-1 Mother Board

Our single piece **Mother Board** for the Altair 8800 gives you 16-card capacity in one single installation. Available with 12 **(MB-12)** or 16 **(MB-1)** slots. **$70.** (Discontinued; limited to stock on hand.)

Wire Wrap Board

Do your own wire wrap prototyping with the **WWB Wire Wrap Board.** Up to 62 16-pin sockets or various combinations of 14, 16, 24 and 40-pin sockets. **$40.**

EXB Extender Board

The **EXB Extender Board** allows accessibility in servicing any 8080-compatible module. **$35.**

*Fully descriptive Owner's Manual available separately. Price refundable with purchase of kit.

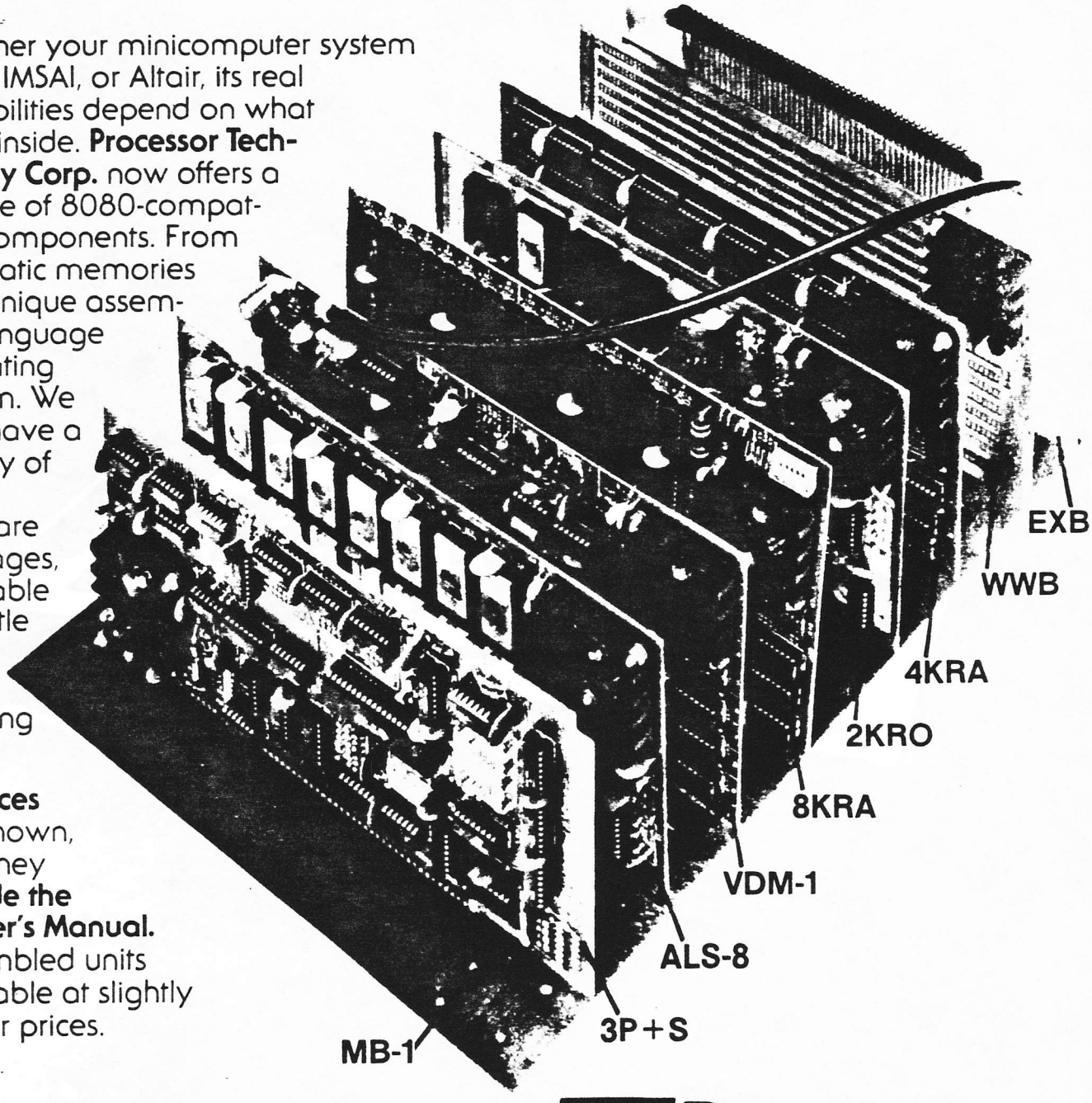
THIS PAGE LEFT BLANK INTENTIONALLY.

On The 8080 Bus?

products from Processor Technology.

Whether your minicomputer system is Sol, IMSAI, or Altair, its real capabilities depend on what goes inside. **Processor Technology Corp.** now offers a full line of 8080-compatible components. From fine static memories to a unique assembly language operating system. We also have a variety of useful software packages, available for little more than copying costs.

Kit Prices are shown, and they **include the Owner's Manual.** Assembled units available at slightly higher prices.

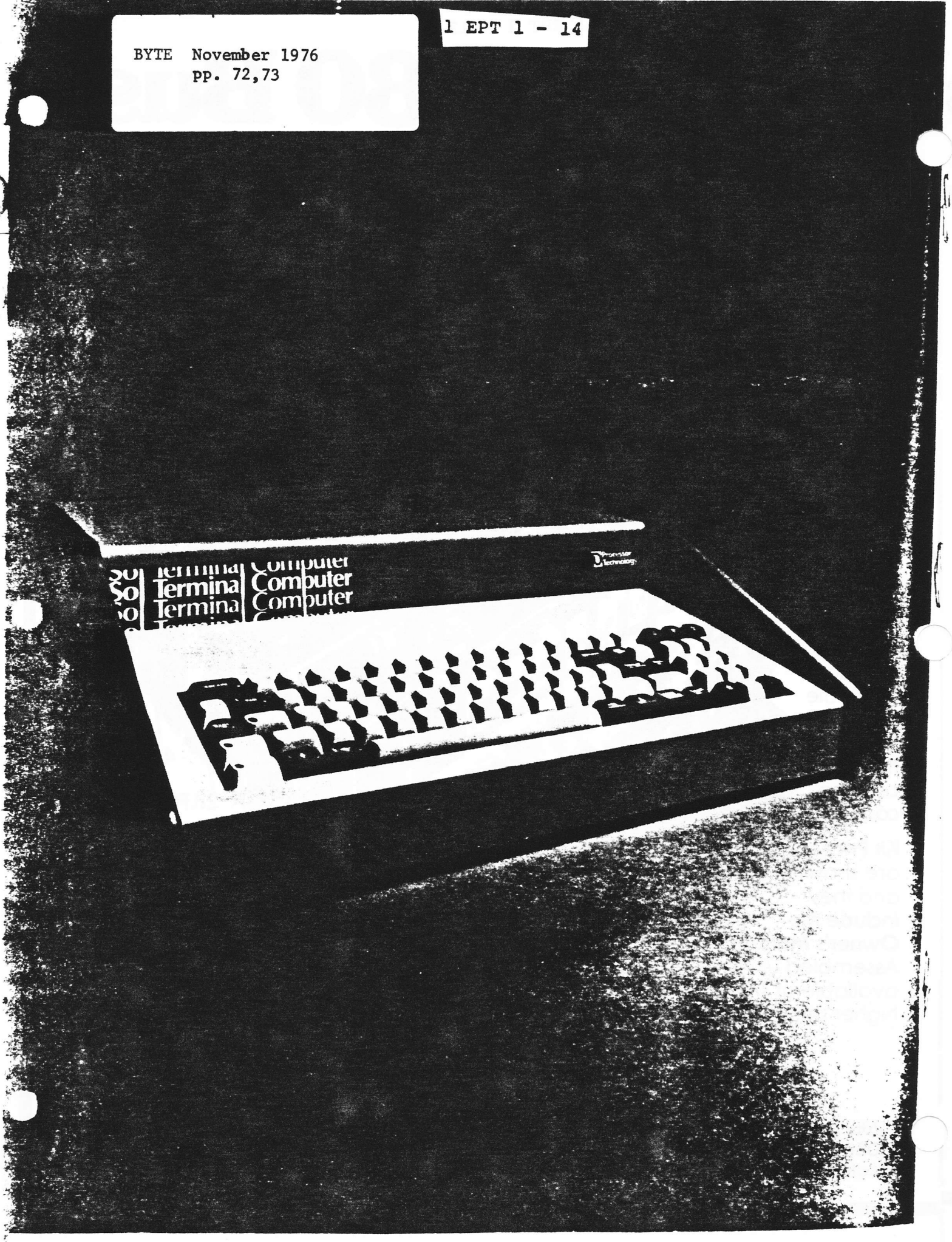


Write Us for further details on all our 8080-compatible products, and for the dealer nearest you.

Processor Technology
6200-V Hollis Street
Emeryville, CA 94608

BYTE November 1976
pp. 72,73

1 EPT 1 - 14



The first complete small computer.

As you thumb through this magazine, you'll see a lot of ads for small computers. For $600 you can find a pretty good box with a power supply, four slot mother board, CPU module, and all the expected lights and switches.

But you know what? It won't work.

That's because in order to make the computer go you have to buy memory—normally both read-write (RAM) and read-only (PROM), interfaces to the outside world (parallel, serial, and cassette), keyboard, video display module, and software.

Add this all up and it's going to cost you at least $1,400 for a complete system.

Got the picture?

Now listen to this. The remarkable new Sol-20 Terminal Computer will give you all of the above... plus more!... as *standard equipment* for just $995, in kit form. This is because the Sol-20—like no other small computer—was designed from the ground up to be complete.

Here's what the Sol-20 includes as standard equipment:

- 8080 microprocessor.
- Video display circuitry.
- 1024 words of static, low-power RAM.
- 512 words of preprogrammed PROM.
- Custom 85-key solid-state keyboard.
- Audio cassette tape interface.
- Both parallel and serial interfaces and connectors.
- Power supply.
- A beautiful case with solid walnut sides.
- Software that includes a preprogrammed PROM personality module and a cassette with BASIC-5 language plus two sophisticated computer video games.
- Full expansion capability with all S-100 bus (Altair/IMSAI/PTC bus) products.

It's a handsome terminal or computer that will even look good in your living room or office. Small systems start at just $475.

Full expansion capability

The Sol-20 system can be tailored to your applications using the complete line of peripheral products from Processor Technology. These include the video monitor, audio cassette and digital tape systems, dual floppy disc system, memories, and interfaces... plus all other peripherals compatible with the S-100 bus.

The Sol-20 greatly simplifies the computer-buying process. It's a splendid package that will excite both present and prospective computer owners.

Our brochure tells all. Write for it today.

Processor Technology, 6200 Hollis Street, Emeryville, CA 94608.
(415) 652-8080.

The Sol-20 expandable system.

Processor Technology Corporation

1 EPT 1 - 16

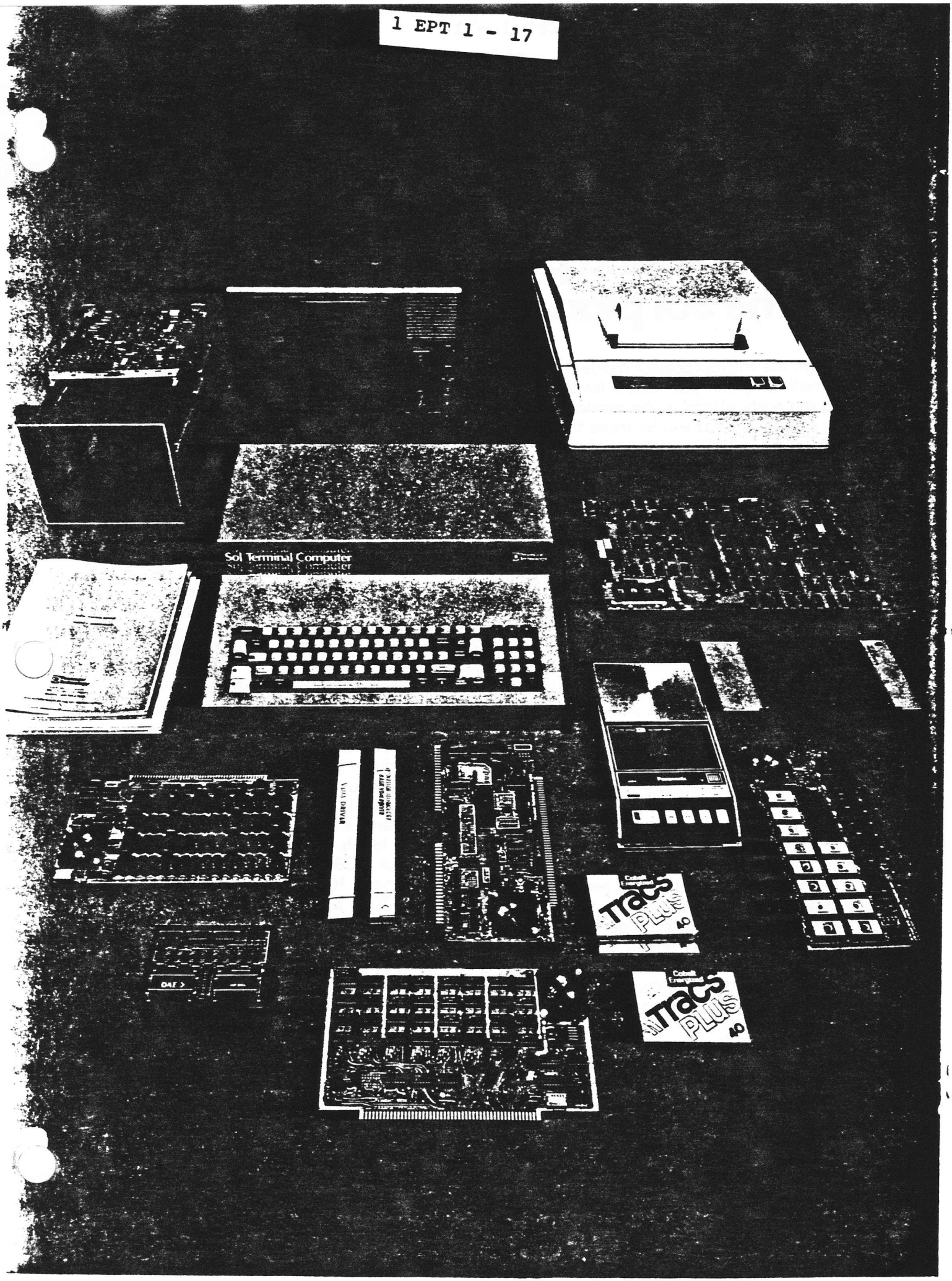
BYTE December 1976
pp. 71-76

Introducing Sol Systems

A complete computer/terminal concept with all the standard features, software and peripheral gear you want in your personal computer.

Processor Technology Corporation

1 EPT 1 - 17



Sol Systems put it all together. One source for hardware and software. One source for engineered compatibility of computer and peripherals. That's the Sol plan.

Though the microprocessor made the powerful small computer possible, a lot of folks found out early efforts in the marketplace were selling the sizzle a lot more than the steak. After an initial investment of several hundred dollars, you ended up with some nice parts, but no memory of any kind, no I/O devices or interfaces, no display, printout or software.

The Sol plan ends all that. Processor Technology takes the position that it's far better to be right than first. So let's get down to the Sol no tricks plan.

For $995 in kit form, the first complete small computer

Standard is a basic word at Processor Technology. The Sol-20 has more standard features than any other small computer we know of. Here's what you get.

8080 microprocessor* 1024 character video display circuitry* 1024 words of static low-power RAM* 1024 words of preprogrammed PROM* a custom, almost sensual 85-key solid-state keyboard* audio cassette interface capable of controlling two recorders at 1200 baud* both parallel and serial standardized interface connectors* a complete power supply* a beautiful case with solid walnut sides* software which includes a preprogrammed Prom personality module and a cassette with Basic-5 language plus two sophisticated computer video games* the ability to work with all S-100 bus (Altair 8800/IMSAI/PTC) products.

There are no surprises. Everything you need to make it work is here. In kit form, nominal assembly time from our fully documented instructions is four to seven evenings.

Or start with the Sol-PC for just $475

You can begin your Sol system with the all on one board Sol-PC kit. It has all the memory and interface electronics including video display, keyboard interface, audio cassette interface, all necessary software and the ability to accept the full Processor Technology line of memory and interface modules. Use the Sol-PC as the basis of a microcomputer, low cost CRT terminal or editing terminal

And these specs are standard

Display: 16 lines of 64 characters per line.
Character set: 96 printable ASCII upper and lower case characters plus 32 selectable control characters.
Display position: Continuously adjustable horizontally and vertically.
Cursor: Selectable blinking. Solid video inversion. Programmable positioning standard.
Serial interface: RS-232 and 20-mA current loop, 75 to 9600 baud, asynchronous.
Parallel interface: Eight data bits for input and output; output bus is tristate for bidirectional interfaces; levels are standard TTL.
Keyboard interface: Seven-level ASCII encoded, TTL levels.
Microprocessor: 8080, 8080A, or 9080A.
On-card memory: 1024 bytes PROM (expandable to 2048 bytes), 2048 bytes RAM.
External Memory: Expandable to 65,536 bytes total ROM, PROM, and RAM.
Video signal output: 1.0 to 2.5 volts peak-to-peak. Nominal bandwidth is 7 MHz.
Power required (±5%): +5 volts at 2.5 amperes, +12 volts at 150 mA, and −12 volts at 200 mA.

The Sol plan, completely expandable.

By filling the basic main frame with tailor made Processor Technology plug-in PC boards, you can really expand the computing power and flexibility of your Sol-20 Personal Computer.

New items are being announced frequently, but right now, here are some of the

things you can add to your Sol-20. The ALS-8 Firmware module is an assembly language operating system to give you the power to develop and run programs. Use it to quickly write, edit, assemble, de-bug and run your own programs. Some say it's the most useful software development on the market today, but modesty prohibits.

And when it comes to add-on memory boards, you've come to the right place. We've probably got more than anyone else. Choose from 2K ROM or 4, 8 or 16K RAM (read all about the 16KRA board on the last page of this ad). The PT 2KRO will accept up to eight 1702A or 5203Q erasable, reprogrammable memories (EPROM's) with the ability to store in a non-volatile fashion up to 2048 eight-bit words.

Our read/write memories are the industry standards for high reliability. We know, because we have literally scores of customer letters saying "Your memory modules work and keep on working."

To help you solve additional interfacing problems, add the 3P+S I/O module. Here's a board with two 8-bit parallel I/O ports with full handshaking logic and a serial data rate that can be set anywhere between 35 and 9600 baud. Set up control conditions for both parallel and serial ports. Data and error flags can be polled.

A full line of Sol-20 tailored peripherals

No computer can do the full job without the right set of peripheral gear. PT has sought out the best manufacturers of peripheral equipment and worked with them to give you a choice of quality so you can get the most out of your Sol-20. Choose from line and serial printers, perforated tape readers and punches, floppy disk memories, black and white or color graphics displays, A/D, D/A converters and more.

Software, the Computer Power Essential

A big part of making the first complete small computer is providing you with a wide range of easy to use, easy to obtain, low cost software. For the Sol-20, we've developed a whole group of offerings. And more are on their way.

TREK 80

Based on the NBC television series STARTREK, this 8K assembly language program uses the VDM graphics capability for real time war with the Klingons. No holds barred, they're out to get you from each of the 100 quadrants. TREK 80 resides and runs in 8K of memory and requires the PTC Sol or VDM-1.

New PT 8K Basic

Processor Technology has the fast new BASIC you've needed for so long. Using our superior BCD math, the speed of the new language is double that of our own fast BASIC-5. To multiple program capability, we've added strings, multidimensional arrays and multi-line, multi-variable, user functions. This is the BASIC for full capability systems. Look at the BUSINESS ANALYSIS program example in the manual to find out how PT 8K BASIC gives you more while using less memory for the working program.

Five reasons why it's so good

1. Strings are not limited to a length of 256 characters and can extend to the bounds of memory.
2. Renumbering of lines with full gosub, etc. updating. Also EXAM and FILL allow for direct memory operations while IN and OUT provide direct I/O capability.
3. Every statement is fully implemented. RESTORE, for instance, restores the data pointer as usual. BUT, with PT 8K BASIC, RESTORE 100 will set the pointer to the data located at line 100.
4. Fully implemented string and math functions include all of the standards — VAL, STR, ASC . . . . EXP and LOGI and LOG. Also, the more advanced statements such as ON-GOTO and IF THEN ELSE along with a loop EXIT are provided.
5. PT 8K BASIC has a 'perfect' implementation of PRINT USING which saves program memory space while still providing more capability than the usual PRINT USING.

The new PT 8K BASIC is similar to the version we're developing for ROM. You use it here before buying the more expensive ROM.

You'll find your PT 8K BASIC also includes both a built-in VDM driver and special editor. The cassette version also includes named program SAVE and LOAD for the CUTS Cassette interface or Sol.

New 8080 FOCAL™ DEC

8080 FOCAL has been updated to include operator precedence and all other standard FOCAL conventions. It also has a driver for VDM-1 display and PT Cassette program SAVE and LOAD This version is available only on CUTS Cassette and resides in 8K of memory.

GAMEPAC 1 to entertain family and friends

Show off your VDM-1 and computer with this lineup of video games. Each is included on the cassette or paper tape.

TARGET keeps track of your hits and misses while you blast away at the moving target. You and your family can get together for whole evenings at a time with this one.

ZING. Learn hexidecimal arithmetic fast with this VDM game as two players keep the five balls in the air. If both of you get too good . . . ZING, of course, will make it harder.

LIFE. The Sol or VDM makes a good display for the game of life and this version allows two modes of operation. The universe can be flat or wrapped around on itself. The real meaning of life we'll leave to you but it's fun to watch.

PATTERN. We haven't figured this one out ourselves but it's sure nice to have your computer doing it. You choose the geometric design and how rapidly it changes.

Sol Systems Price List

(prices are net, effective Dec. 1, 1976)

SOFTWARE

ITEM with manual	Source	CUTS cassette	Paper tape
BASIC 5 software #2	yes	**	$19.50
8K BASIC	no	$29.00	$37.00
New 8080 Focal	no	$14.50	N/A
TREK 80 video game	no	$ 9.50	$14.50
GAMEPAC 1 video games	no	$ 9.50	$14.50
MATHPACK video calculator	yes	$14.50	$19.50
ASSEMBLER software #1	yes	$14.50	$19.50

**CUTS cassette of BASIC 5 is included FREE with all orders for Sol units or CUTS cassette interfaces. Additional cassettes available for $14.50.

Sol system owners be sure to note Sol system on your order. These special versions use less code and provide easier loading along with more convenient operation. SOLOS, SOLED and CONSOL all have provision for the special versions.

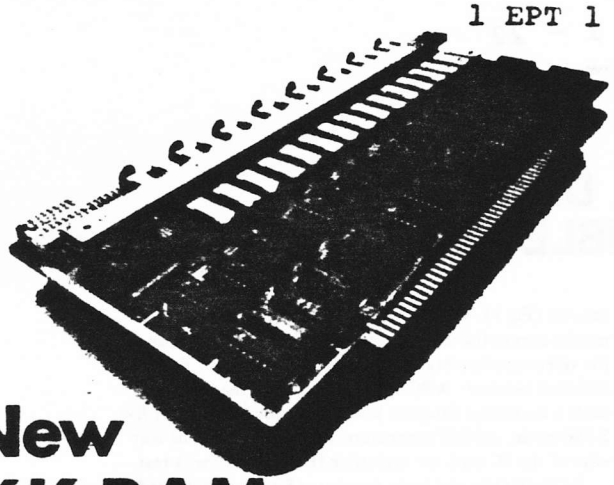
All Processor Technology software is distributed on an individual sale basis for personal use. No license to copy, duplicate or sell is granted with this sale. Each software package has been copyrighted by Processor Technology and all rights therein are reserved.

Sol Terminal Computers	Kit Price
SOL-PC SINGLE BOARD TERMINAL COMPUTER™	$475.*
SOL-10 TERMINAL COMPUTER™ Sol-PC with case, power supply and 70 key solid state keyboard.	$795.*
SOL-20 TERMINAL COMPUTER™ all features of Sol-10 with larger power supply, 85 key solid state keyboard, fan, and five slot expansion backplane.	$995.*

*Sol prices include CONSOL Personality Module. If SOLED Intelligent Editing Terminal Module or SOLOS Stand-alone Operating System Module is desired instead, add $100. If ordered separately, personality modules are $150 each.

Memory Modules	Kit	Asmbld.
ALS-8 PROM Resident Assembly Language Operating System	—	$425
SIM-1 Interpretive Simulator add-on option for ALS-8	—	$ 95
TXT-2 Text Editing add-on option for ALS-8	—	$ 95
2KRO Erasable PROM module	$ 65	$ 89
4KRA 4096-word Low Power Static RAM	$159	$195
8KRA 8192-word Low Power Static RAM	$295	$375
16KRA 16384-word Dynamic RAM	—	$529
Interface modules		
3P+S Parallel, Serial I/O module	$149	$199
CUTS Computer Users Tape System cassette interface	$ 87	$119
VDM-1 Video Display Module	$199	$295
Mass Storage Systems		
Helios II Disk System™ includes dual PerSci 270 floppy disk drive, cabinet, fan, S-100 bus compatible controller, power supply, system diskette with complete PTDOS software	$1895	$2295
Misc.		
EXB Extender Board	$ 35	$ 45
WWB Wire Wrap Board	$ 40	—

Prices, specifications and delivery subject to change without notice. Please allow up to two weeks for clearance of personal checks. Mastercharge accepted. All orders amounting to less than $30 must include $3 for handling.



New 16K RAM, fully assembled, $529

More bits per buck than ever before on a fully burned in and tested board unconditionally guaranteed for one year.

Processor Technology made the first 4K static RAM modules for the home computer market. Now in a price performance breakthrough we offer you a 16,384 byte dynamic memory module assembled, tested and burned in. Not a kit—and at $529 who'd want to build it from scratch?

Processor Technology gives you the features to make 4K dynamic RAMS work for you.

- Invisible refresh, no waiting while CPU is running.
- High speed 400 μsec access time worst case Z-80 and 8080 compatible.
- Versatile addressing, each 4096 byte segment is individually addressed to any of the sixteen available 4K segments.
- Low power—typically 5 watts when running—the same as most 4K memory modules.
- BATTERY BACKUP capability built-in for standby operation.
- IMPORTANT NOTICE—No 16K memory module available is fully, truly static. 4200/4402 type "static" RAM's have high level, high current clocks with high transient power levels. Any RAM with 12 volt 30 mA clock pulses should not be called "STATIC" just because each memory cell is a flip-flop.

Specifications

Access Time	400 nsec max
Cycle Time	500 nsec max
Rams Used	Intel 2104 or Mostek 4096 types
Capacity	16384 8-bit bytes
Memory Protect	standard on card
Addressing	each 4096 byte page addressable
Operating Power	+7.5 to 10 VDC at 0.4 A typical +15 to +18 V at 10 mA typical −15 to −18 VDC at 20 mA max

The new Processor Technology 16K board is available for immediate delivery. See your nearest dealer listed below or contact us directly. Address Processor Technology, 6200 Hollis Street, Emeryville CA 94608, Phone 415/652-8080.

Processor Technology Dealers

CALIFORNIA

The Byte Shop
1514 University Ave.
Berkeley CA 94703

The Byte Shop
2559 South Bascom Ave.
Campbell CA 95008

The Computer Mart
624 West Katella #10
Orange CA 92667

The Byte Shop
2227 El Camino Real
Palo Alto CA 94306

The Computer Center
8205 Ronson Road
San Diego CA 92111

The Computer Store
of San Francisco
1093 Mission Street
San Francisco CA 94103

The Byte Shop
509 Francisco Blvd
San Rafael CA 94901

The Byte Shop
3400 El Camino Real
Santa Clara CA 95051

The Byte Shop
2989 North Main St.
Walnut Creek CA 94596

FLORIDA

Microcomputer
Systems Inc.
144 So. Dale Mabry Hy.
Tampa FL 33609

GEORGIA

Atlanta Computer Mart
5091-B Buford Hwy.
Atlanta GA 30340

ILLINOIS

The Numbers Racket
518 East Green Street
Champaign IL 61820

itty bitty
machine co., inc.
1316 Chicago Ave.
Evanston IL 60201

The Chicago
Computer Store
517 Talcott Road
Park Ridge IL 60068

INDIANA

The Data Domain
111 South College Ave.
Bloomington IN 47401

MICHIGAN

The Computer Store
of Ann Arbor
310 East Washington
Ann Arbor MI 48104

NEW JERSEY

The Computer Mart
of New Jersey
151 Kline Boulevard
Colonia NJ 07067

Hoboken Computer Works
56 Second Street
Hoboken NJ 07030

NEW YORK

Audio Design Electronics
487 Broadway, Ste. 512
New York NY 10013

The Computer Corner
200 Hamilton Ave.
White Plains NY 10601

The Computer Mart
of Long Island
2072 Front Street
East Meadow, L.I. NY 11554

The Computer Mart
of New York
314 Fifth Ave.
New York NY 10001

Synchro Sound Enterprises
193-25 Jamaica Ave.
Hollis NY 11423

OREGON

The Real Oregon
Computer Co.
205 West 10th Ave.
Eugene OR 97401

RHODE ISLAND

Computer Power, Inc.
M24 Airport Mall
1800 Post Road
Warwick RI 02886

TEXAS

The Micro Store
634 South Central
Expressway
Richardson TX 75080

WASHINGTON

The Retail Computer Store
410 N.E. 72nd
Seattle WA 98115

WISCONSIN

The Milwaukee Computer Store
6919 W. North Ave.
Milwaukee WI 53213

CANADA

The Computer Place
186 Queen St. West
Toronto, Ontario M5V 1Z1

Trintronics
160 Elgin St.
Ottawa, Ontario

Processor Technology Corporation

DESIGNING CONSUMER-SYSTEM PERIPHERALS

INTELLIGENT-TERMINAL DESIGNERS OPT FOR 8080-COMPATIBLE CIRCUITRY

by Lee Felsenstein and Robert Marsh

Because the "home computer" serves a consumer market, both it and the peripherals designed for use with it must exhibit high-volume production and adequate customer support. With these fundamental requirements in mind, we designed the Sol product line to simultaneously meet the needs of two applications. Both Sol-10 and Sol-20 function as intelligent terminals—each unit lacks only a CRT monitor. Additionally, Sol-20 incorporates a power supply and expansion chassis, which with adequate memory allow it to operate as a stand-alone computer. And the system's basic electronics, housed on one board and designated Sol-PC, serves OEM applications that require a single-board computer.

To provide Sol with the required customer support, we developed Basic and Focal language packages as well as application and game programs. We also developed two ROM-resident programs—Solos, which optimizes Sol-20 functions for stand-alone computer applications, and Soled, which implements the functions of an intelligent terminal on either Sol-10 or Sol-20.

Lee Felsenstein *is the founder of LGC Engineering, Berkeley, CA, and a consultant to Processor Technology, Emeryville, CA.* **Robert Marsh** *is vice president of Processor Technology.*

All of these actions hinged on one major development in the hobby-computer market. Within the past year, several peripheral-interface and memory kits have appeared, each of which utilizes the "hobbyist," or S-100, bus structure used in several 8080-based computers. To take advantage of such kits, we decided to design the Sol family around this same S-100 bus structure. In essence, Sol is the combination of a microprocessor circuit with several S-100 peripheral modules.

implementing the design

As initially conceived by one of us (Marsh), Sol consists of a typewriter-sized cabinet on whose flat top a video monitor can rest (Fig 1). One 10" x 16" PC card contains all electronics except the unit's keyboard and power supply, and the video signal generated by the device serves any EIA-standard monitor. A PC edge connector on the board accepts a backplane daughter board that holds as many as five S-100 cards. All I/O connectors, also available on the rear edge of the PC card, are accessible from the cabinet's rear.

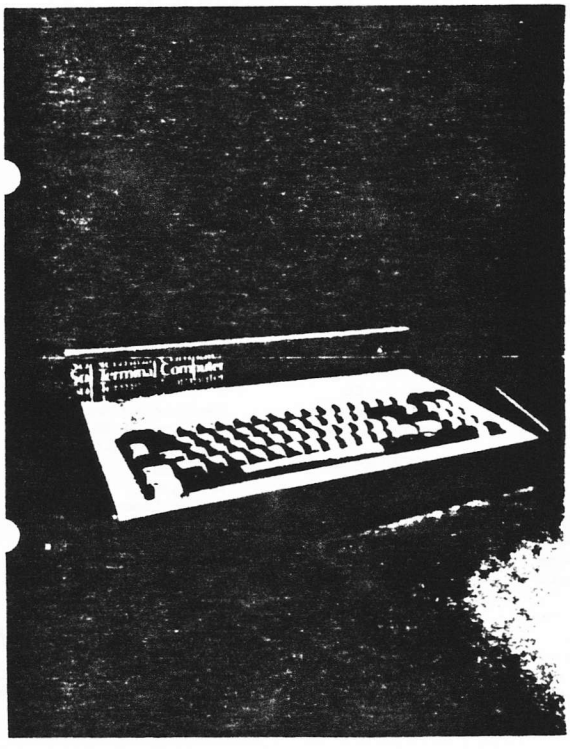


Fig 1 Housed in a typewriter-sized cabinet on which a CRT monitor can sit, Sol-20 contains all circuitry necessary to function either as an intelligent terminal or a stand-alone computer. The system's designers chose to configure it around the "hobbyist," or S-100, bus structure used in several 8080-based computers; Sol is basically the combination of several S-100 peripheral modules with a microprocessor circuit.

To implement this basic structure (Fig 2), we buffered the system's 8080 address and data lines to the circuit and bus connector through tri-state drivers, much as do other S-100-type processors. Next, we paralleled two unidirectional data buses to form a bidirectional bus and thereby eliminated the need to run eight additional lines around what promised and proved to be a very crowded card. We also connected a 4-input multiplexer to select data input to the processor from the keyboard port, the parallel port and an internal data bus, as well as from the external data bus.

We created the internal bus as a unidirectional circuit for low-drive, on-card memory and I/O devices that cannot meet the heavy drive requirement of the full external data bus; it allows maximum utilization of the tri-state capabilities of the universal asynchronous receiver/transmitters (UARTs) used in the serial and tape channels. The flag and status outputs of these devices arrive in parallel at the 8080 chip; port addresses from the board's address decoder section enable the outputs. The decoder also controls the multiplexer's switching; default direction comes from the system's external bus. We defined the 4K bytes of memory in Page C (high-order hex digit of the address) as on-card memory, divided between 2K of ROM, 1K of RAM and 1K of "visible RAM" in the video display circuit.

The display section treats its RAM as 2-port memory; the processor has the highest priority. We placed the second port under control of the screen refresh circuitry, which calls up data as required for conversion by the character-generator ROM into video signals for display. We didn't connect the video display section to the internal bus because we felt we had to allow for its being loaded directly from an external DMA device, which can gain control of the data bus but not the internal bus. We made the data bus the source of all data fed to memory and I/O, both on-card and off; the only other data input to the processor from on-card circuitry through the data bus comes from the sense switches, an 8-wide DIP array that lets the CPU sense an alterable parameter byte under program control through input port FF.

We derived board timing from a 14.31818-MHz crystal oscillator; the frequency, four times that of the NTSC color burst, provides compatibility with color video graphics devices. This "dot clock" goes to an external connector and feeds the output shift register and character divider of the video display section as well as the 8080 clock divider. We configured the clock circuit using MSI and SSI TTL chips rather than the Intel 8224 LSI chip to allow for several selectable microprocessor clock rates; that way, we can retrofit higher-speed processor chips into the same board. When designing the clock, we took care to ensure non-overlapping

THE PERSONAL COMPUTER RETAILER

VOL.1 NO.1 PUBLISHED BY PROCESSOR TECHNOLOGY CORP., 6200 HOLLIS ST, EMERYVILLE, CA 94608 MARCH, 1977

This is the first issue of our new newsletter distributed solely to retail computer stores. PCR will be mailed monthly and is free to bona fide retail computer stores. In the newsletter we will try to give you advance notice of all Processor Technology ads in national magazines as well as the latest dope on new products. We will use PCR as a medium for discussion of detailed specifications, selling points, competitive comparisons, price changes, and special promotions as well as for quasi philosophical diatribes on business policy and general P.R.

We believe you will find the PCR an invaluable source of information whether or not you are a Processor Technology dealer. Your responses are welcomed and encouraged since we hope to make PCR an interactive medium of communication. In this vein take note of our Q & A column on page 7 where we will gladly respond in print to your questions about PTC products and policies. Yes, we'll even answer the nasty ones!

SOLOS?- CUTER?- GPM? What Is Going On?

Many of the PT dealers have long ago seen most of the software packages we recently advertised. Each package, in one form or another, has been fully implemented and essentially ready for delivery for a number of months. Why then have we created the two month void by not delivering? Quite simply the answer is: Real System Thinking.

The advent of the Sol-20 and CUTS cassette board opened an opportunity to at long last provide a software configuration that allows for a major step forward in standardization. Think about it.

Someone using a parallel output keyboard with the 3P+S had a hard time getting status at port Ø and data at port 1.

Bootstrap loaders for paper tape had to assume the user had memory at some location outside of the area required by the program. (For the paper tape version of TREK 80 we put the loader on the VDM screen!!)

The Sol System, however, has clearly defined I/O.

Ø for Keyboard and Display
1 for Serial IN and out
2 for Parallel
3 for user defined

These numbers are used with the SOLOS "SET" commands to dynamically determine I/O. And, the Sol System always has available memory (1K worth) at C8ØØ. In addition, the Sol System has a cassette tape interface ready and willing to load or save programs.

As SOLOS became a reality, we discovered a method of using the CUTS board and CUTER software to make the "other" brands' mainframes almost as easy to use as the Sol.

CUTER defines the 3P+S Ports in a manner similar to the Sol I/O. CUTER allows the user to have keyboard, serial and parallel I/O on the 3P+S just as with the Sol-20.

(cont. on page 2)

cont.

What Is Going On?

But, more important, CUTER allows our software to run on any machine even though we designed it for the Sol System running SOLOS!!

Was it worth the delay?? Yes, indeed!

Now for the Important Details

In case you haven't received it yet, SOLOS (CUTER, too) features 18 commands to provide super-convenient operation of the computer. Standardized Input/Output, Variable Display Speed and complete tape load and store provisions are some of the highlights. And, buried deep within the code are routines that provide BYTE BY BYTE access to the cassette tape data. SOLOS handles two buffers to read blocks of tape while the user calls for data one byte at a time!

Using this feature, Sol BASIC-5 allows the user to:

```
10 READ #1; A,B,C
20 PRINT #2; A*2, B/C
```

But the "other brand" user must have CUTER to use these features.

CUTER is provided on a cassette tape for $11.00 with each CUTS purchase. It loads to ANY selected memory address using a short bootstrap loader. Then, when the user loads a P.T. program he places his selected address into the loaded program. (We assume address CØØØ and the programs are provided with this address.)

CUTER assumes I/O is provided by a 3P+S but can be modified for less conventional boards. A very new good VDM-1 driver is also resident in CUTER.

Wouldn't it be nice if everyone had a 3P+S, VDM-1 and CUTER IN ROM at CØØØ? Well....the March issues of the magazines will announce the Processor Technology GPM (General Purpose Memory) in a very special way. Here's how it goes.

The GPM has space for ROM, PRO and RAM. The GPM is given FREE with CUTER in ROM right at CØØØ and 1K of RAM starting at C8ØØ.

Have we lost our bananas?

Actually, we lost them a long time ago but the GPM is FREE only with the purchase of a SUBSYSTEM "B" and the SUBSYSTEM "B" is the keynote of our forthcoming ads.

The user buys a 3P+S, VDM-1, CUTS and any one of the PTC Memory Boards as a SUBSYSTEM "B". The total price is exactly the sum of the individual prices. The GPM, CUTER ROM and 1K of RAM is then absolutely FREE.

The March ads for the Subsystem will create a little confusion because they refer to a "bootstrap" loader instead of CUTER. CUTER is provided!! This error is corrected in the April '77 issue.

If the entire Subsystem "B" is not needed, the price of the GPM with CUTER AND RAM is $129.00 Kit or $169.00 assembled.

The GPM can be ordered without CUTER and RAM for $89.00 Kit, $100.00 assembled for use in Sol Systems.

About Deliveries

Delivery of the GPM is solely dependent on our receipt of the CUTER ROM. For this reason each SUBSYSTEM "B" will be shipped with a mailing card for the buyer to receive his ROM. We have been promised 5-week delivery on the ROM's so please expect at least seven weeks per the usual.

The GPM is general purpose for another very special reason. The ALS-8 ROMS are now available and they reside on the GPM right along with CUTER.

The ROM's are sold separate of the GPM and are priced as follows:

ALS-8	$159.00	6144 bytes of ROM
SIM-1 & TXT-2	60.00	2048 bytes of ROM

(cont. on page 3)

What Is Going On?

Thus the price for the ALS-8, SIM-1, TXT-2, CUTER and 1K of RAM is $348.00 when not purchased with Subsystem "B", but for use in NON-Sol main frames.

Note: Subsystem "B" purchasers need only buy the ROM's. The GPM is free with their Subsystem purchase.

The GPM module will fit in the Sol-10 cabinet without a backplane, as well as into all other S-100 computers. Don't forget the Sol version GPM is $40 less than the Subsystem "B" version.

The First Issue Of Processor Technology ACCESS Is In The Mail

This is the "technical newsletter" we referred to in our Fall '76 new product flyer. We think you will find the ACCESS an extremely valuable information source. Yearly subscriptions are $4.00 for 8 issues. The order of distribution will be as follows:

1) paid subscribers
2) complimentary copies to retailers and trade press, simultaneous with No. 1
3) newsstand copies to retailers, about one week after Nos. 1 and 2

Newsstand copies are distributed free to Processor Technology dealers to be sold for 75¢ per copy. Other retail computer stores or suitable outlets may purchase the ACCESS from us for 30¢ each--minimum order $30.00, i.e., 100 copies.

New Small Computer Catalog Available - 22 Pages!

PTC's new color catalog is currently being distributed to all of our retail outlets. You should find these beautiful catalogs more than helpful in explaining the features of PTC equipment and software. We do not charge stores for these catalogs. The decision whether or not to charge customers for these catalogs is left up to each individual store owner, but we recommend that they be given out free to "qualified" customers. Initially we will be quite liberal with the number sent to each store. Expect us to get "uptight", however, when you have distributed 1000 catalogs, sold only three Sol-20's, and want more catalogs.

The March-April issue of PERSONAL COMPUTING MAGAZINE will contain the entire PTC catalog. Since most present personal computer owners get a copy of PERSONAL COMPUTING one way or another, the demand for actual catalogs from retail stores will be somewhat reduced.

TV/Monitor

The new catalog announces for the first time our PT-872 Video Monitor/TV Set. This is a Panasonic TV modified for selectable Monitor or Television operation. We added the unit to our line as a customer service to provide an excellent quality monitor at $199.00. Because of the Customer Service aspects and because $199.00 is very close to the retail price, the net discount for P.T. Dealers has been set at 25% regardless of quantity. Also, to limit our processing costs further, please place your PT-872 TV/Monitor orders along with regular orders.

Diatribe Department

You too can have your name in print when you become the famed author of a PCR Diatribe. This month we will monopolize the column, but we would really enjoy printing guest Diatribe editorials written by you. Send your double-spaced typewritten manuscript to PCR's Robin Content, Diatribe Dept.

Real System Thinking - This Month's Diatribe

A wise old sage named Anon once said "A mess of parts doth not a computer make." Anon was once also heard to utter the timely phrase neither do working hard, firm, or soft-ware a Real System make."

Real Systems are more than a stack of different modules plugged into a main frame. Real Systems are more than a consistently interfaced and reliably connected group of peripherals. Real Systems are more than a series of different debugged programs that run on a given set of hardware.

So far the personal computer industry and the "movement" as a whole have shown a decided lack of system thinking. Home computers typically grow erratically and contain an eclectic blend of modules and programs from many sources that rarely work together without substantial modification. Usually the modification process is ongoing and continuous so that never is one guy's rig completely compatible with another's.

Much of the incompatibility problem has been created by manufacturers, PTC among them, who have not been able to get together with their competitors to agree on standardization of I/O or software. The problem derives though, not from this seemingly inevitable inability to standardize, but from non-system thinking.

What is system thinking? System thinking's number one principle is "All system elements must always be compatible with all other elements." To the system owner and user this principle implies that his system should be:

1) consistent and reliable
2) easy to operate
3) easy to expand (or reduce)
4) easy to repair and maintain
5) easy to link to other systems

To the system designer and manufacturer this principle implies that his system should be:

1) conservatively rated, software too!
2) designed with strong user input and feedback
3) modular
4) pluggable, unpluggable, as simply as possible
5) conventionally interfaced whenever possible

What makes a Real System? Real Systems, i.e., real Computer systems always have these five elements:

1) Human Being(s) for and with whom the computer is working--the most important element
2) Hardware, everything you can touch, even if you shouldn't
3) Software, everything you can't actually touch that must be there for the system to function
4) Manufacturer(s) and Designer(s), they've got to support Nos. 1 and 5 as well as crank out Nos. 2 and 3 in volume
5) Sales and Service people, hopefully they will be organized and take prime responsibility for solving No. 1's problems.

Notice that most of these system elements are people not products. Real Systems must have organized people backing them up. The people that made the machine had better be capable of fixing it or you will certainly never be able to fix it. The people who wrote the systems software had better be able to listen to you and maintain and improve their programs or you won't be able to adapt, not even if you have a rare source listing. In general, they had better be able to help you solve your problems.

As for the hardware, all elements including CPU, memory, input/output, expansion interfaces, keyboard, and power supply must be designed together in as simple a manner as possible without sacrificing versatility and the virtually certain necessity of future expansion. All of these elements must be both simple to operate and simple to build, test and repair. External peripheral devices should always be "plug-in-able" using commonly available connectors and cables. Keyboards are critical elements! Real Systems never scrimp on keyboard costs since this is the first place the user will discover the results of a cheap design.

As for software: rule No. 1--Real System software designers must talk to the hardware designers. Both designers should even ask each other for advice fairly often. This would be an unusual situation in the computer business, to say the least, but it is a very important principle. As an example of hard/software interaction, commonly used housekeeping programs should be accessible to all programs that run on the system. Accessibility should imply also that the whole works should run as soon as it is turned on.

Rule No. 2--System software must be modular and expandable, very much as the hardware must be. For example, a 4K BASIC program has got to run in a 12K BASIC system without modification.

The author recommends the preceding ideas as guidelines to be followed by both computer owners and makers when considering buying, modifying or expanding a computer system. Whatever the application problems, if we can all direct our thoughts and energies toward integrating the multitudinous available pieces and parts into <u>one systems package</u> we will have taken a giant step toward eliminating the threat of the creeping Kludge.

CONSOL Update

All standard PTC software is designed around CUTER and SOLOS ROM's. Normally these ROM's reside at CØØØ to C7FF (hex) on either a Sol personality module or GPM board. CONSOL, since it is a temporary introductory package, has its I/O routines in non-standard locations. Therefore such programs as BASIC-5 <u>will not work</u> properly with CONSOL. PTC will be glad to reprogram without charge CONSOL modules to standard I/O locations if customers or retailers return them to us. However, we strongly recommend replacing the CONSOL module with SOLOS. ONLY SOLOS has the subroutines which permit cassette write operations, dynamic port switching and BASIC file manipulation.

Module Department

16KRA

PTC's new 16KRA module is just about the lowest price per bit, high quality S-100 RAM module available. At $529 the 16KRA is a memory that is fully assembled, burned-in (for 12 hours at maximum rated temperature) and thoroughly tested. Only highest quality premium grade parts are used.

The Intel 2104-6 or Mostek MK4096-11 RAM's used are spec'd at 350 nanosec. access time and 500 nanosec. cycle time worst case. [NOTE: we can get faster RAM's--down to 150 nsec. If you need them, let us know right away.] The S-100 bus interface circuiting is fully asynchronous and uses the latest low power Schottky and integrated Schottky delay time technology. There are no "one shots" in critical timing paths and refresh is automatic and invisible to the CPU. Each 4096 byte page is individually addressable, using four of the 16 DIP switches on the card, conveniently placed at the top edge. A connector is provided for optional battery backup during power failures.

The 16KRA is guaranteed compatible with Sol, IMSAI 8080 and Altair 8800 mainframes as well as the Helios II DMA disk controller. The 16KRA is the perfect high density memory since the Sol has a limited number of expansion slots.

Important: Phantom, pin 67 in the S-100 bus is recognized by the 16KRA and is required for proper operation of Sol's and ALS-8.

Deliveries: First deliveries of 16KRA modules to retailers are now scheduled for late March with our large current order backlog eliminated in April. As is usual at PTC once this product is in production, we will be stocking it in depth!

CUTS

We will start delivering CUTS modules to all those with current back orders on March 9, 1977. The demand for this product has far exceeded our expectations so we expect it will take another month to eliminate our backlog.

CUTS, the Computer Users Tape System, is hardware, the key element required to make full use of our many systems software packages. The CUTS module is compatible with the built-in tape interface in Sol Systems and can read and write at either 300 bits per second, pure "Byte/Kansas City standard," or 1200 bits per second. All PTC CUTS cassettes are recorded at 1200 baud and cannot be read at 300 baud. Cassettes recorded with SWTPC, Poly 88, or Morrow boards can be read at 300 baud by CUTS or Sol hardware but the software in CUTER, CONSOL or SOLOS requires a specific data header format not used by these suppliers. So far as we know, Tarbell cassettes are not CUTS compatible.

Once PTC software is in your stores in volume, every S-100 bus computer owner will find the CUTS module indispensable--That is a promise!

Memories--4KRA, 8KRA, 16KRA

All three PTC RAM modules incorporate circuity which disables any RAM at location zero during power-on start up. This feature, called PHANTOM, is mandatory with Sol systems and only PTC memory modules have it! ALS-8's and GPM modules can generate PHANTOM for auto-start up operation in Altair and IMSAI mainframes.

Sol Department

We hardly need say that the Sol Terminal Computer is the highest quality and most cost effective Real System available in the small computer field. Inventories of Sol-20 in particular are being built up by PTC to meet the quick turn around needs of your store. Once the SOLOS personality modules and Sol software are in the field in quantity we expect to see another surge in demand. We urge all PTC retail dealers to place realistic orders for Sol Systems soon (for up to three month's needs) to eliminate the possibility of demand exceeding our ability to supply retailers quickly. If you order Sol Systems for a two or three month period, we will allow you to schedule shipments with reasonable minimum quantities per shipment. Be advised: be accurate--we will ship the numbers you request.

SOL POSTERS!

PTC still has a good number of beautiful Sol-20 color posters. If you would like more of these posters to decorate your retail store or computer repair shop or to give to Sol buyers as a "freebie," let us know. They're off the shelf.

Q A

Q: Can a customer trade in his CONSOL module when upgrading to SOLOS?

A: Yes--PTC will credit the customer $30 for a returned CONSOL module in working condition.

Q: What is the expansion power supply capacity of the Sol-20?

A: The Sol-20 can supply +8VDC unregulated at 8A and ± 17VDC at .9A unregulated. These are conservative ratings for the five slot backplane only. The Sol-PC and keyboard supplies are separate from the backplane.

Q: How do I add more slots to my Sol?

A: One way is to add a Helios II disk system. The Helios cabinet has space for an additional optimal 10 slots. We have not yet announced price or availability of this option to the Helios system.

Q: Why do we advertise before products are ready?

A: In the past we were forced to give in to that unfortunate temptation because of intense competitive pressure in an expanding marketplace. We want you to know that we are committed to avoiding this practice in the future. Our goal is to have products on the dealers' shelves when the new announcements appear.

Subsystem "B" makes the computer you already have work almost as well as a new Sol-20.

Add it now, and for a limited time we'll throw in a new module with 2048 bytes of preprogrammed PROM or ROM and 1024 bytes of RAM memory free!

Processor Technology's Subsystem "B" puts together the major system elements you need to get your Altair or IMSAI up and running.

You get both RAM and PROM memory, parallel, serial, cassette and video display interfaces, and software. Software includes a bootstrap loader program so you can load any Processor Technology CUTS ("Byte/Kansas City") 1200 BAUD cassette tapes. Standardized subroutines in ROM, similar to those in a Sol personality module, are used by many Processor Technology software packages to improve program efficiency. You'll find you rarely need to touch your front panel switches. With our Subsystem "B" you are up and running as soon as you turn on the power.

Three subsystems are available, depending on your memory requirements. Each Subsystem "B" includes five S-100 bus compatible modules as listed below. Each is dependent upon our new GPM module (GPM = General Purpose Memory) which provides 1024 bytes of low power static RAM and 2048 bytes of preprogrammed ROM or EPROM as well as space for up to 8192 bytes more of ROM or 2708 type EPROM. The GPM module accepts the new ROM version of our well known ALS-8 Editor/Assembler software package.

With Subsystem "B" you have guaranteed compatibility with all Processor Technology software and hardware products.

Subsystem Model	B70	B110	B190
Total memory provided (bytes)	7168	11264	19456
Display I/O	VDM-1	VDM-1	VDM-1
Parallel, Serial I/O	3P+S	3P+S	3P+S
Tape Cassette I/O	CUTS	CUTS	CUTS
Memory	**4KRA**	**8KRA**	**16KRA**
Price	$594	$730	$964

NOTE: The GPM module with both RAM and PROM is included FREE in all the above subsystems. The GPM kit is $129 if purchased separately.

The ALS-8 ROM chip set is $159. SIM-1 & TXT-2 ROM add-on set is $60 and requires both GPM and ALS-8 ROM.

I want Subsystem "B" to get my Altair or IMSAI up and running.

☐ Enclosed is a check for $_____
California residents add 6% sales tax.
No shipping charge.

☐ Mastercharge # _____
(include Interbank No. and expiration date)

☐ Send more information

I want:
☐ B70 at $594
☐ B110 at $730
☐ B190 at $964

Name _____

Address _____

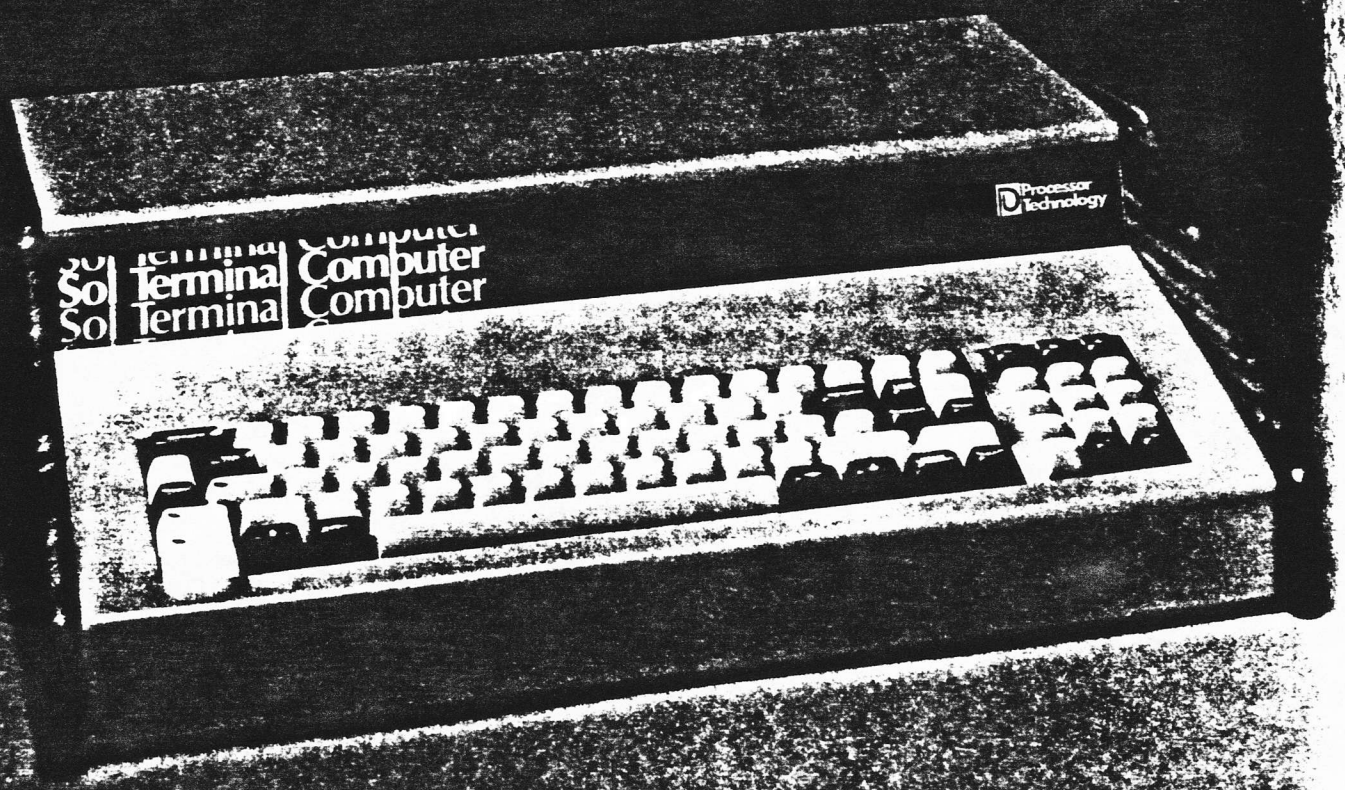
City _____ State _____ Zip _____

Processor Technology Corporation

6200 Hollis Street, Box B
Emeryville, CA 94608 (415) 652-8080

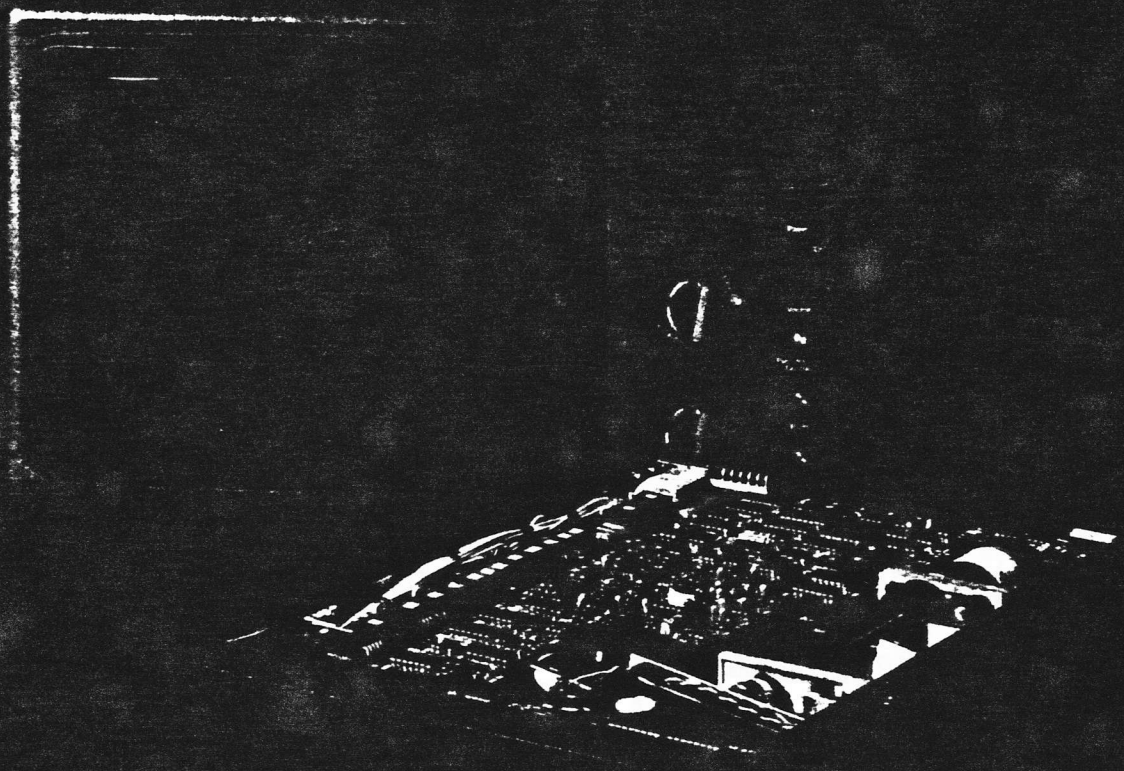
1 EPT 1 - 31

The Small Computer Catalog.

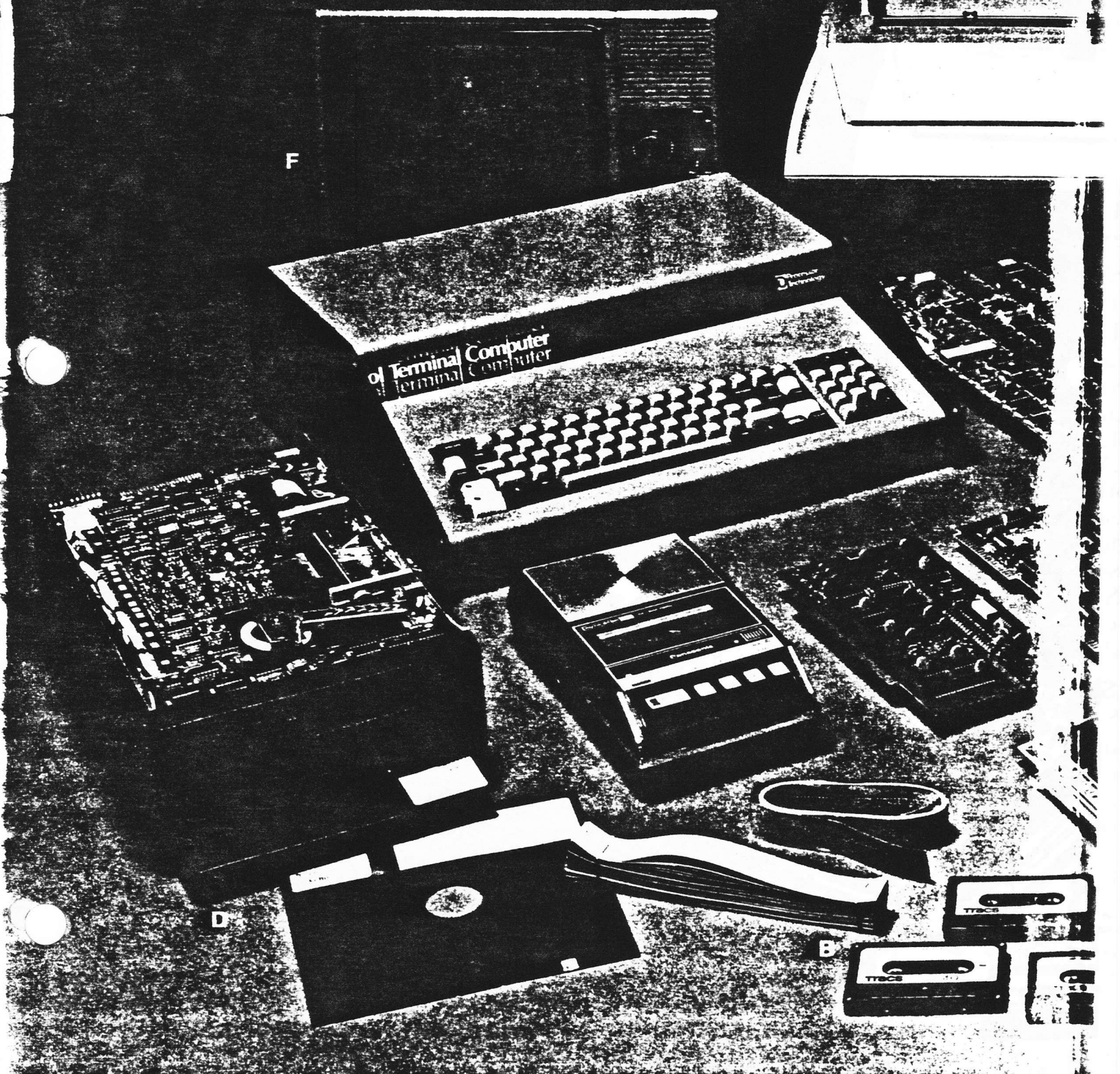


1 EPT 1 - 32

And Sol Solution chart showing applications in business, science, home entertainment and management, art, law, medicine and education.



One source for quality hardware, software, and peripherals.



That's the Sol plan.

The new Sol-20 is unique.

It's the first small computer designed as a complete system.

Most small computers simply "grew like Topsy" — a memory here, an expansion module there. They weren't conceived or integrated to provide maximum efficiency at lowest possible cost.

Sol-20, a true breakthrough in small computer systems, includes all the essential elements as *standard* equipment — central processor, memory, keyboard and display, software, a power supply, and appropriate packaging.

There are no "surprises." You don't have to buy expensive peripheral equipment to make it run. Its own keyboard and "smart" terminal are built-in.

Use it without being a programming expert.

In fact, you can operate it efficiently without any prior computer experience.

Unlike other small computers, Sol is already programmed to receive your commands the moment it's turned on, thanks to Sol plug-in Personality Modules.

And Sol systems are supported in depth by extensive software and additional peripherals — such as flexible disk memories — so it's appropriate for more sophisticated applications.

Sol computer systems never grow old. Add new modules to update and expand your computer's power.

Sol is easy to use

Sol operates like a typewriter so many applications require no special programming. Packaged in handsome cases with solid walnut sides, Sol computers look good in the living room, office or lab. Sol computers come in kit or fully assembled form.

Sol-20 is a scaled-down big computer system

Use Sol in a variety of applications.

In the home. Home uses are limited only by your imagination. Regulate heat and light to save fuel. Run a complex model railroad. Compute taxes. Play a variety of TV games, not only computer hockey and tennis, but more interesting, more complex games such as TREK-80, where your starship takes on a whole fleet of Klingons. Several sophisticated TV games come with the Sol-20. And you can even design your own.

At the office. Use it as a full-fledged business computer. Use it to compose and edit letters electronically, store and retrieve mailing lists, process orders, maintain journals and general ledgers, and produce statements and reports.

In the lab. Use Sol to reduce and analyze data statistically, control lab equipment, prepare graphics, and fit curves. Sol-20 frees your time and expands your overall capability.

In schools and universities. Use Sol-20 to teach computer programming. Use it for computer-aided instruction. Use it for notes, records and sorting.

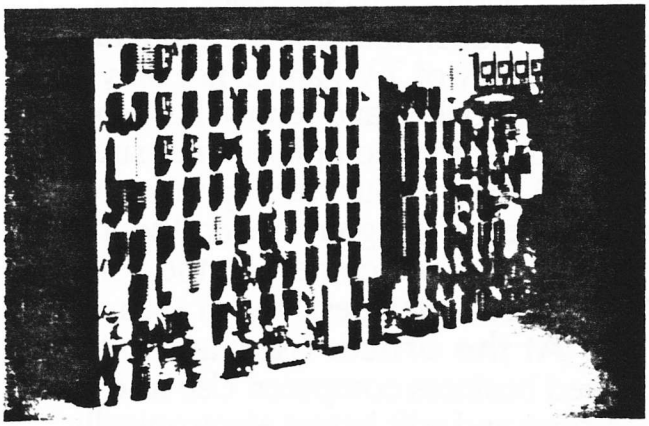
So much is standard

Here's the computer with a microprocessor, display and input/output circuitry, memory, full alpha-numeric keyboard, big power supply, handsome cabinet, and software.

Add extras for more power

Extras include a module to help write, edit, assemble, de-bug and run your own programs. There's no better collection of add-on memories anywhere... up to 16,384 words per module. Solve additional interfacing problems with our I/O module. Get big system performance with our Helios II "floppy" disk system. Display results on our video monitor. Output on line or serial printer. Other peripherals include joysticks, paper tape readers, A/D and D/A converters, and PROM programmers.

Sol Computers



Sol computers are currently offered in three forms: the Sol-20, Sol-10, and Sol PC.

The Sol-20 Stand Alone Computer

Sol computer systems are currently offered in three forms: the Sol-20, Sol-10, and Sol PC.

Sol 20 is the most complete and sophisticated of the three packages, a fully contained "personal" computer able to take on an infinite variety of tasks. Sol-20 comes with:

- 8080 microprocessor, still the most sophisticated computer-on-a-chip available and the "brains" of the Sol-20.
- 1024-character video display circuitry. View your output on any standard video monitor or specially adapted TV.
- 1024 words of static low-power read/write memory (RAM) for program storage.
- 1024 words of static low-power, preprogrammed permanent memory (ROM) takes care of important system "housekeeping" chores. ROM memory automatically readies the computer for your commands as soon as the Sol is turned on.
- a custom designed, beautifully laid-out 85-key solid-state upper and lower case keyboard with cursor keys and arithmetic keypad.
- an audio cassette interface capable of controlling two recorders at 1200 bits per second. Store and retrieve programs and large amounts of data at very low cost.
- both parallel and serial standardized interfaces with connectors on card.
- a complete rugged power supply and quiet cooling fan.
- a handsome case of walnut and metal.
- software including a preprogrammed PROM personality module and a cassette with BASIC-5 language, plus two sophisticated computer video games.
- a design compatible with all S-100 bus products.
- a back plane capable of accepting five expansion modules.

The Sol-10 Terminal Computer

Sol-10 comes in the same handsome package as the Sol-20, but because of limited memory, it is designed more specifically for "smart" terminal applications. Price of the Sol-10 includes case, power supply, and 70-key solid state keyboard. A fifteen key arithmetic pad is optional.

Later you can upgrade to a Sol-20 by adding an expansion backplane, extra power supply, fan and keypad.

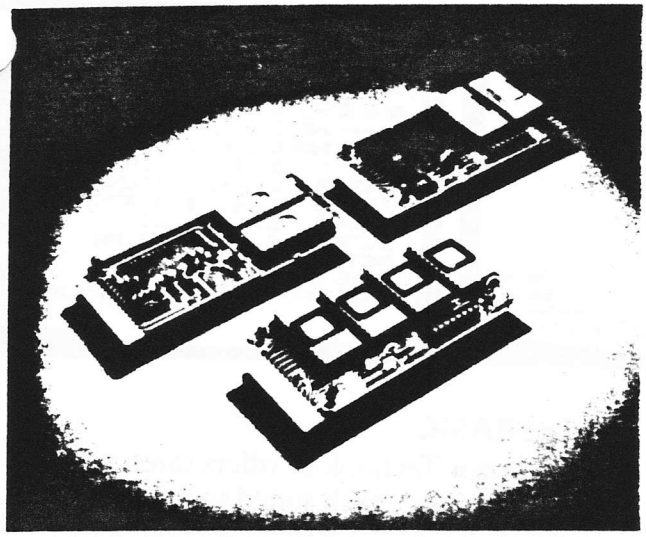
Sol-PC Single Board Terminal Computer

Here's the heart of the Sol system. The Sol-PC is a single printed circuit board with microprocessor, memory, display and interface electronics, and plug-in personality module that is fully compatible with our complete line of memory and interface modules.

The board comes in kit or fully assembled form with all of the following:

- Display: 16 lines of 64 characters per line.
- Character set: 96 printable ASCII upper and lower case characters plus 32 selectable control characters.
- Cursor: Selectable blinking. Solid video inversion. Programmable positioning standard.
- Serial interface: RS-232 and 20mA current loop, 75 to 9600 baud, asynchronous. 25 pin female "D-type" connector on card.
- Parallel interface: Eight data bits for input and output; output bus is tristate for bidirectional interfaces; levels are standard TTL. 25 pin male "D-type" connector on card.
- Keyboard interface: Seven-level ASCII encoded, TTL levels.
- Microprocessor: 8080, 8080A, or 9080A.
- On-card memory: 1024 bytes PROM (expandable to 2048 bytes); 2048 bytes low power static RAM.
- External Memory: Expandable to 65,536 bytes total ROM, PROM and RAM.
- Video signal output: 1.0 to 2.5 volts peak-to-peak. Nominal bandwidth is 7 MHz. Power required (±5%) : +5 volts at 2.5 amperes, +12 volts at 150 mA, and −12 volts at 200 mA.

A. Personality Modules



Sol Personality Modules allow you to choose three different levels of operation. Software in each module optimizes Sol for a particular application and at the same time provides a measure of general purpose capability. For custom applications and for volume OEM users personality modules are available without memory for three different types of EPROM and two types of factory-mask ROM.

SOLOS, the most popular module, optimizes the Sol for stand-alone computer applications. Choose SOLOS if you intend to use your Sol system to store and retrieve business or personal records, control electronic instruments, perform independent calculations for business, science or education, or any other application where the Sol system will be "on its own" operating independently of other computers.

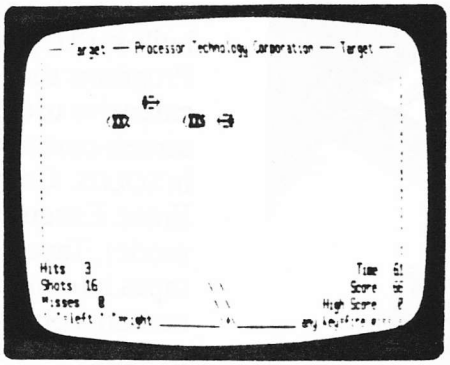
SOLOS is oriented around use of the Sol's built-in CUTS audio cassette data interface. Programs such as Sol-BASIC and ALS-8 can make extensive use of the cassette handling and screen-cursor manipulation routines contained in SOLOS. Commands included are: Dump, Enter, Execute Terminal (i.e. enter Terminal mode), Tape Load (reads CUTS format cassette tapes into memory), Tape Save (stores memory contents on CUTS tape) and Set I/O (permits dynamic switching of input and output devices under manual or program control). With SOLOS the Sol can also be used as a "smart" terminal in conjunction with other computer systems, but ordinarily the SOLED module is the better choice when the Sol system is often used as a terminal.

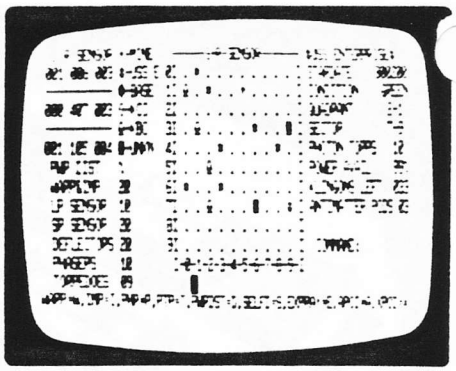
With the SOLED personality module installed the Sol becomes an advanced editing terminal system. Like SOLOS, SOLED uses the full 2048 byte capacity of its module. It contains programs and routines which allow remote direct cursor addressing and file and cassette tape editing. Data and text can be edited on or off-line and transmitted in blocks under local or remote control. Large cassette data files or text messages can also be transmitted and received automatically from remote locations.

SOLED has the ability to dynamically change input and output device assignments. Stored information can be transmitted via modem, printed on one of several printers or stored on additional cassettes or flexible disks.

CONSOL is a 1024 word low cost personality module which gives minimal capability to the system. Commands include Enter, Dump, Execute, Tape Load, and Terminal. CONSOL permits operation as a low level CRT terminal and is useful for simple stand-alone applications. Full keyboard cursor control, up, down, right, left, home, clear is provided. CONSOL is not needed with SOLOS or SOLED modules because its functions are duplicated.

B. Software





Software is the sine qua non of any computer system. It's the computer power essential. No computer can be more powerful than the software that goes with it.

That's exactly why Processor Technology has devoted more effort to the development of software than other small computer makers. Maybe that's why some of our worthy competitors have taken our source listings, added a few twists and taken title. But the truth will out.

All Sol systems software is designed to make full use of the routines and programs permanently stored in all Sol personality modules. User programs such as BASIC require less memory space, because personality module routines are called up whenever needed for functions such as keyboard input, screen formatting, and cassette tape storage operations. Interface with the user is straightforward and consistent because keyboard commands and control sequences are standardized for all Sol software.

Sol BASIC

Processor Technology offers three versions of BASIC language, each suited to a different application. BASIC-5 is a small version of this versatile language designed for applications requiring just mathematical manipulation without extensive processing of text. BASIC-5 is the perfect language for an introduction to computer programming because it's easy to learn and requires a small amount of memory storage. Many hundreds of programs already written in BASIC work with Sol BASIC-5 and our 8K BASIC as well.

Processor Technology 8K BASIC is a very high speed full function language with all the virtues of BASIC-5's multiple program capability and BCD floating point math. Speed is at least double that of the already fast BASIC-5. For even greater power, we've added strings, multi-dimensional arrays and multi-line, multi-variable, user functions. Here's the language for full capability systems. For instance, in our instruction manual, take a look at the *Business analysis* program. See how you get more power while using less memory for the working program.

Advantages of Sol BASIC

Processor Technology 8K BASIC offers several unique and unusual features. Versatile print statements provide fully formatted output to multiple devices, from CRT screen to teletype to line printer. Multi-dimensional arrays permit powerful fast processing of any data that can be organized graphically or in tabular form. Several statements are provided to give complete and direct high level language control over system memory and input/output channels. Full capability string functions simplify manipulation

and processing of text and alphabetic materials so they are more straightforward and easy to use than ever before. In short, with this BASIC, no effort has been spared to bring you high level problem solving power.

Extended Disk BASIC has all the powerful features of the 8K memory-resident version and includes disk commands and big system file handling capability. Disk BASIC is perfect for such complex applications as inventory control and payables-receivables accounting.

BASIC CHART

Commands:		BASIC-5	8K BASIC	Extended Disk BASIC
ASAVE	ASCII DISK SAVE			+
CONT	Continue		+	+
CLEAR		+	+	+
GET	tape or disk	+	+	+
KILL	delete file			+
LIST		+	+	+
MEM	multiple programs	+	+	+
NULL	for printers	+	+	+
RESAVE				+
RNUM	Renumber		+	+
RUN		+	+	+
SAVE	tape or disk	+	+	+
SCR	Scratch	+	+	+
XEQ	Get + Run	+	+	+
Statements:				
CALL	call machine subroutine	+	+	+
CLEAR			+	+
CLOSE	disk file			+
DATA		+	+	+
DEF	define function		+	+
DIM(X)		+	+	+
DIM(X,Y,Z,)			+	+
ELSE	if,then,else		+	+
END		+	+	+
EXAM	memory "dump"		+	+
EXIT		+	+	+
FILL	"deposit" memory		+	+
FOR . . . NEXT		+	+	+
FREE	free space		+	+
GOSUB		+	+	+
GOTO		+	+	+
IF . . . THEN		+	+	+
INP(X),Y	from inport x		+	+
INPUT		+	+	+
INPUT," "	suppress CRLF	+	+	+
LET		+	+	+
ON	ON . . . GOSUB		+	+
OPEN	disk file			+
OUT(N),	to out port N		+	+
PAUSE			+	+
PRINT		+	+	+
PRINT USING			+	+
READ		+	+	+
READ#N	read file	+	+	+
REM		+	+	+
RESTORE		+	+	+
RESTORE	with line #		+	+
RETURN		+	+	+
REWIND	rewind file pointer			+
SET I/O	for peripherals	+	+	+
STOP		+	+	+
WAIT	for input port bit(s)		+	+
WRITE	disk			+

BASIC Functions		BASIC-5	8K BASIC	DISK
ABS	absolute value	+	+	+
ARG	16 bit conversion	+	+	+
ASC	ASCII value		+	+
ATN	Arctangent		+	+
CHR	Decimal value of character		+	+
COS	Cosine	+	+	+
EOF	End of file			+
EXP	e^x		+	+
INT	Integer	+	+	+
LEN	String length		+	+
LOG	Natural logarythm		+	+
LOG10	LOG base 10		+	+
RND	Random number	+	+	+
SEARCH	Search string for string		+	+
SGN	Sign of number	+	+	+
SIN	Sine	+	+	+
SQR	Square root	+	+	+
STR	Convert no. to string		+	+
TAB	PrintTAB(X)	+	+	+
TAN	Tangent	+	+	+
VAL	Convert string to no.		+	+

The ALS-8 Program Development System

Applications with very high speed data manipulations or critical timing elements demand "custom fit" programs and subroutines. High level languages written for microprocessors such as FOCAL, BASIC or FORTRAN cannot always handle these assignments. In these cases the best solution is programs written in assembly language, a language much more closely related to actual real-time computer operations. Assembly language is easy to learn and, with either of our two assemblers, quite easy to use.

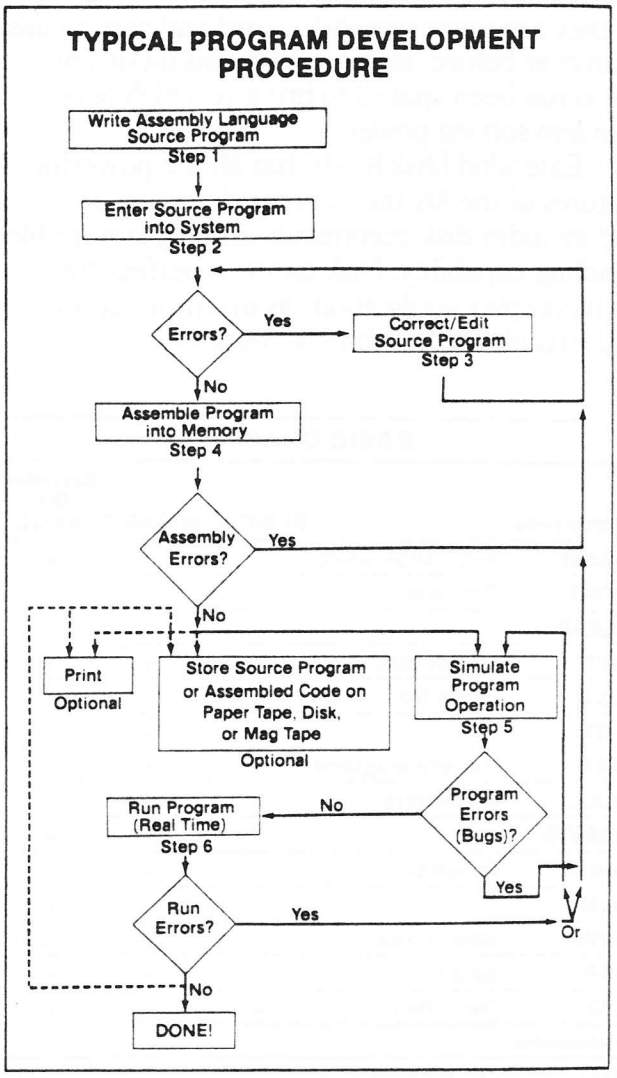
To simplify the development process as diagrammed on the right both Processor Technology assembler programs organize user programs as files.

Processor Technology's much imitated Software #1 package is a small assembler-monitor system designed for development of small to medium length programs which must be stored in system RAM memory for assembly. The ALS-8 is a more versatile and expanded development package with many additional powerful features.

With the ALS-8 up to six source programs can be stored in memory as named files and called at will to be listed, edited, assembled or simulated. Files may also be stored on tape or disk and can be assembled from any selected input device. Files can be appended, moved, re-numbered, taken apart or linked together. Using the FCHK command, crashed files can be restored.

Assembly language source programs are entered using line numbers from paper or mag tape, keyboard or disk. All editing is done by line number but with the TXT-2 Text Editing software, it becomes possible to automatically add line numbers to un-numbered text.

The Assembler includes labels, comments, expressions and constants, along with relative symbolic addressing, which gives you the ability to chain common symbols from one program to another (even if the other program was assembled at some other time). Also, various assembly error messages are provided to help you eliminate program bugs.



ALS-8, a powerful, new development procedure

ALS-8 has the unusual ability to dynamically adjust the system's I/O handling configuration. The system includes an I/O driver table accessible through use of three resident commands or the drivers themselves. I/O device driver routines may switch themselves on and off or transfer I/O control to a different device driver under program control.

Your development system might have a CRT terminal, a high speed line printer, paper tape reader/punch and a teletype. The System can print a listing to the line printer, then input from the paper tape reader and return console control to the CRT terminal or teletype, all under program control.

Up to 20 custom commands can be entered by the user and called in exactly the same way as the standard resident commands. With the

custom commands, I/O driver table, dynamic I/O switching capability and common symbol tables, you can change your system's configuration and operating modes at any time.

Resident commands are:

ASSM	CUST	ENTR	FIND	MOVE	SYME
ASSME	CUSTD	EXEC	FMOV	NFOR	SYML
ASSMI	CUSTE	FCHK	FORM	SIMU	SYSIO
ASSMX	DUMP	FILE	IODR	STAB	SWCH
AUTO	EDIT	FILES	LIST	SYMD	TEXT

Custom commands: Up to 20 specified by by user.

The ALS-8 requires 2048 bytes of random access memory (4096 is recommended) for symbol tables and system global area, addressed at D000 (hexidecimal).

The SIM-1: The SIM-1 Interpretive Simulator is a program that actually thinks it's an 8080! With the SIM-1/ALS-8 combination, simulate 8080 programs on your Sol, IMSAI, or Altair computer without actually running them in real time. All registers, flags, program counter, and stack are simulated. Try out programs with no fear of crashing your system if something goes wrong. The system doesn't lose control if a program error is encountered (e.g., an incorrect jump or call).

With SIM-1, you can set breakpoints, enable or disable register/memory content printout. I/O instructions can be run in real time, simulated from the system console, or set to predetermined values for any I/O port address.

SIM-1 is a powerful de-bugging tool for 8080 programming.

TXT-2, Text Editor

Adds the world of text editing to your system. Using TXT-2, insert, delete and move single characters, entire lines or portions of lines. Complete text files can be scanned at several user controlled rates, up to almost 2000 lines per minute when used with our VDM-1 Video Display Module.

Both ALS-8 and Software #1 packages are available on "CUTS" 1200 bps cassette or paper tape. The ALS-8 is also available preprogrammed into permanent ROM memory to provide "Instant-on" efficiency and speed.

TREK 80

Based on the NBC television series STARTREK, this machine language program uses 8K of memory and the VDM graphics capability for real time war with the Klingons. No holds barred, they're out to get you from each of the 100 quadrants. You can warp through hyperspace, fire phasers, photon torpedos or experimental rays, or if you just can't go on, self-destruct. TREK 80 resides and runs in 8K of memory and, if not used with a Sol, requires a Processor Technology VDM-1 Video Display Module.

New 8080 FOCAL (™ DEC)

FOCAL is a high level math language originally written for the PDP-8 minicomputer. Many thousands of FOCAL programs are in existence and now they can run in the Sol. Our original 8080 FOCAL has been updated to include operator precedence and all other standard FOCAL conventions. It also has a driver for VDM-1 or Sol displays and CUTS cassette program save and load. FOCAL is available only on CUTS 1200 bps Cassette and resides in 8K of memory.

Gamepac 1

Show off your Sol system with this line up of video games. Each is included on the CUTS cassette or paper tape.

TARGET — Keeps track of your hits and misses while you blast away at the numerous flying objects. Includes sound effects. You and your family will spend whole evenings at a time with this one.

ZING — Learn hexidecimal arithmetic fast with this video game as two players keep the five balls in the air. If both of you get too good... ZING of course, makes it harder.

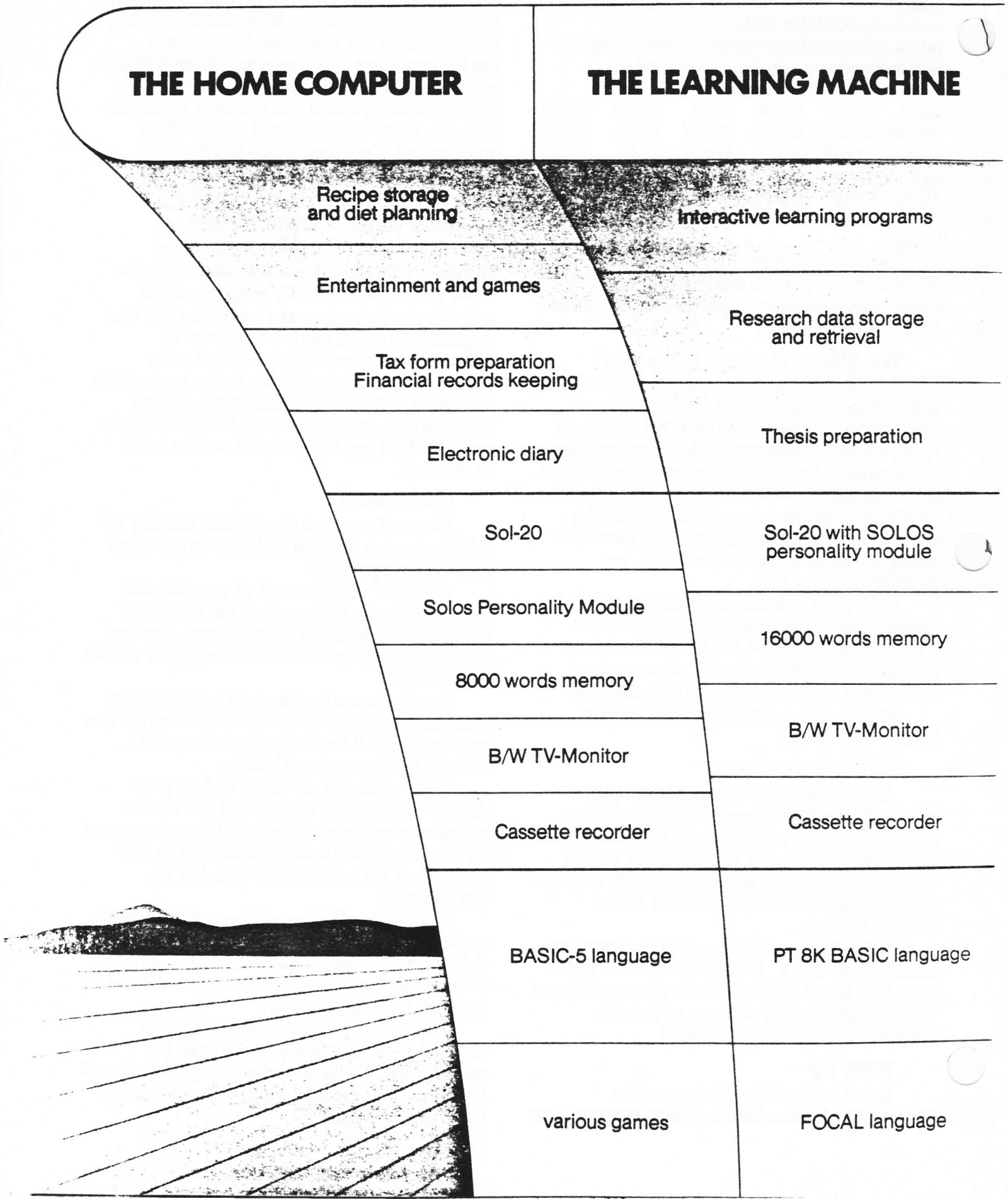
LIFE — The Sol or VDM-1 make a good display for the game of LIFE and this version allows two modes of operation. The universe can be flat or wrapped around on itself. The real meaning of life we'll leave to you, but it's fun to watch.

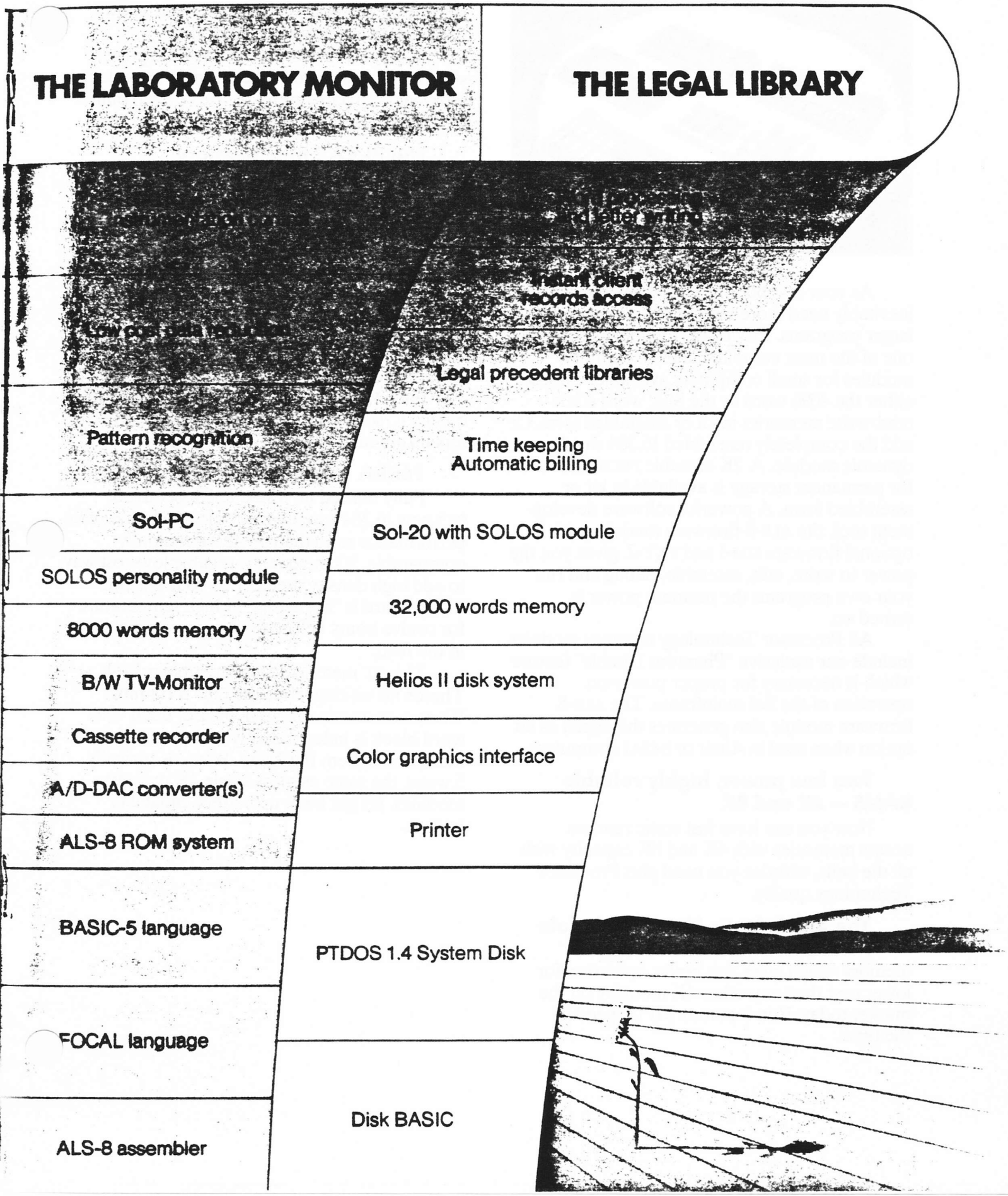
PATTERN — We haven't figured this one out ourselves, but it's sure fun to have your computer doing it. You choose the geometric design and how rapidly it changes. The computer dazzles you with its artistic genius.

All Processor Technology software is distributed on an individual sale basis for personal use. No license to copy, duplicate or sell is granted with this sale. Each software package has been copyrighted.

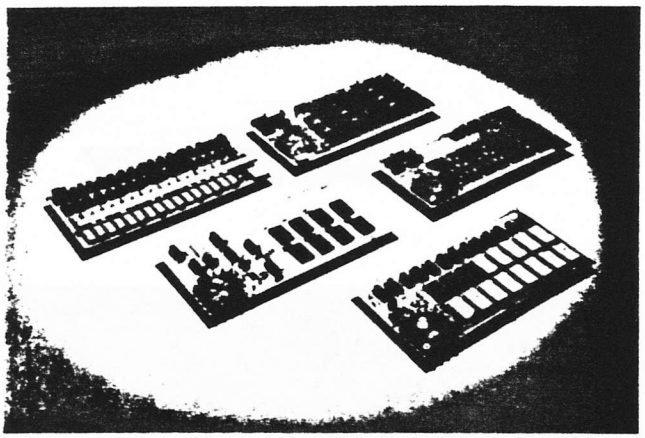
Sol Solution Chart

1 EPT 1 - 42




THE LABORATORY MONITOR
THE LEGAL LIBRARY
Instant client records access
Legal precedent libraries
Pattern recognition
Time keeping
Automatic billing
Sol-PC
Sol-20 with SOLOS module
SOLOS personality module
32,000 words memory
8000 words memory
B/W TV-Monitor
Helios II disk system
Cassette recorder
Color graphics interface
A/D-DAC converter(s)
Printer
ALS-8 ROM system
BASIC-5 language
PTDOS 1.4 System Disk
FOCAL language
Disk BASIC
ALS-8 assembler

C. Memories



As your computing needs grow you will inevitably need more memory for storage of larger programs. Processor Technology offers one of the most complete lines of memory modules for small computers available. Choose either the 4096 word or the 8192 word static read/write memories in kit or assembled form. Or add the completely assembled 16,384 word dynamic module. A 2K erasable PROM module for permanent storage is available in kit or assembled form. A powerful software development tool, the ALS-8 firmware module, with its optional firmware SIM-1 and TXT-2, gives you the power to write, edit, assemble, debug and run your own programs the moment power is turned on.

All Processor Technology memory modules include our exclusive "Phantom Disable" feature which is necessary for proper power-on operation of the Sol mainframe. The ALS-8 firmware module also generates this signal as an option when used in Altair or IMSAI computers.

Two low power, highly reliable RAMS — 4K and 8K

Now you can have fast static random access memories with 4K and 8K capacity with all the bells, whistles you need plus Processor Technology quality.

The 4KRA Static Memory Module

Here's a 4096 word read/write static memory which gives you better operation for lower cost than any other 4K memory on the market today. Run it at max MPU speed all the time.

Processor Technology uses only low power static RAM Integrated circuits. So you know you're getting outstanding reliability.

In fact our module draws so little power, you can use standard "D" cells to give you long term back up data retention. We've even built in a battery connector, and recharge circuitry.

The 8KRA Static Memory

PT's 8K memory gives you all the advantages of our 4K with twice the capacity and more flexible addressing circuitry. The 8KRA uses less power than two 4KRA memories.

All address and data lines are fully buffered. Noise immunity circuitry is built-in. The 8KRA has PT's exclusive built-in KSET switch giving you card address offset in 1K increments. Address is set by a dual inline switch easily accessible at the top of the PC board.

Each IC — all 76 of them — has its own top quality IC socket so that assembly, test and repairs are far easier.

16KRA Memory

Fully burned in, tested and assembled, PT's new 16,384 byte memory offers a better price performance ratio than anything remotely comparable. It's the quality, reliable low-cost way to add high density memory to your system. Every board is "burned in" at high temperature for twelve hours before test to insure reliability in the field.

This PT memory offers invisible refresh. There's no waiting while the CPU is running. Worst case access time is 400 nsec. Each 4096 word block is independently addressable for maximum system flexibility. Power is typically 5 watts, the same as most single 4K memory modules. It's got back-up battery capability built in.

2KRO Erasable Programmable Memory

Accepts up to 2048 bytes erasable programmable read-only memory. Stores data even when power is off. Great for your custom loader or monitor programs.

The 2KRO is jumper selectable to fit any one of thirty-two 2K segments within the 65K addressing range of the 8080. Additional jumpers select the appropriate number of "wait" states, determined by the access time of the EPROMs in use.

The 2KRO was designed for either the 1702A or MM5203 EPROMs. EPROMs are not included, but both are readily obtainable for reasonable prices on the industrial and surplus markets.

The ALS Firmware Module for fast software development

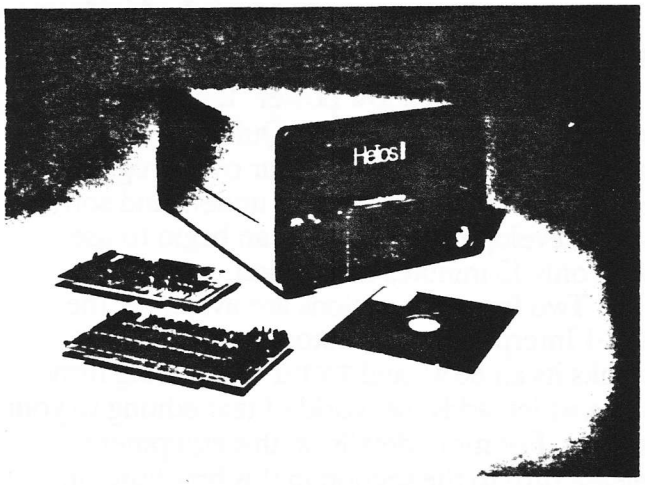
The ALS-8 is a low power "turn-on-the-switch" program developer. Quickly write, edit, assemble, de-bug and run your own programs. Here's an easy to use, easy to understand software development tool you can begin to use with only 15 minutes instruction.

Two firmware options are available, the SIM-1 Interpretive Simulator, a program that thinks its an 8080, and TXT-2 text editing firmware which adds the world of text editing to your system. For more details on this equipment please turn to the section in this brochure on software. The ALS-8 is only available factory assembled and tested.

PTC MEMORY MODULES

	4 KRA	8 KRA	16 KRA	2 KRO	GPM/ALS-8
Maximum Capacity (8-bit words)	4096	8192	16,384	2048	5120 to 8192 bytes ROM
RAMS used	91L02A or 2102LPC	91L02A or 2102LPC	Intel 2104 or Mostek 4096 types	1702A EPROM	9216B ROM
Operating Mode	Static	Static	Dynamic	Static	Static
Access and Cycle Time	520 nanoseconds worst case maximum. Typical 400 nanoseconds.	Same	400 nsec access 500 nsec cycle	Dependent on EPROM used. Works over range of 30 to 2500 nsec	450 nsec
Bus Pinout	Plug in compatible with Sol, Altair 8800 and IMSAI 8080 bus	Same	Same	Same	Same
Power: Operating	+7.5 to 10 VDC @ 1.0A max (0°C), 0.8A typical at 25°C. 0.8A typical, 1A max.	+7.5 to +10 VDC at 1.4A typical (25°C); 1.9A max (0°C to 70°C)	+7.5 to 10 VDC @ 0.4A typical, 0.8A max. +15 to +18 @100mA typical, 150mA max. −15 to −18 VDC @20mA max.	+8 to −10 VDC @ 0.6 max. −15 to −19 VDC @350mA max with 8 1702As installed. (Replacement transformer available for full negative supply in Altair 8800)	+7.5 to +10 VDC @ 600 max. +14 to 19 VDC@ 200mA max. (with SIM-1 and TXT-2 options installed)
Power: Standby	+1.6 to 2.5 VDC at 0.5A max worst case. 0.4A typical	+1.6v to 2.5 VDC typical; 0.9A max (power connector provided for battery connection)			
Address Selection	Dual in line switches	Dual inline switch at top of PC board allows manual selection of any 8K segment on 1K increments	Each 4096 byte page addressable with dual in line switches at top edge of PC board	Jumper selectable to any 2048 byte block of the 32 available.	Fixed at E000 to FFFF (hex)
Dimensions	5.3"x 10.0" (13.46 cm x 25.4 cm)	5.4"x 10.0"	5.4"x 10.0"	5.3"x 10.0"	5.3"x 10.0"
Phantom RAM (for Sol and ALS-8)	Yes	Yes	Yes	No	Yes

D. Disk Storage



Disk Storage

Every computer owner longs for all the advantages of fast random access memory. We're ready when you are to put big system disk memory power at your command. The new Helios II is more than just a floppy disk drive and controller. It's more than just scattered pieces of wire and patches of software. Helios II is a complete, integrated disk storage system which should meet every program and data storage requirement your system is likely to have. The Sol-Helios pair forms a cost effective, high performance system without equal.

Helios II gives you BIG SYSTEM PERFORMANCE

Big system performance is unique to the Helios II. Used in any application requiring manipulation of large data files, Helios II will outperform all other microprocessor based systems by a factor of at least 10 to 1.

Big system performance means all disk and memory buffer space allocation, all file management, all device interaction, comes from the system.

Big system performance means extended DISK BASIC, DISK FOCAL, and Processor Technology software support. DISK FOCAL is provided free on the system diskette and extended DISK BASIC is offered on a separate diskette for $50. Using these simple languages you can immediately write programs for any application you have in mind. The file operations include random byte or block access as well as update and rewrite in place of standard sequential files. Other application packages are under continuous development at Processor Technology. And in line with our basic software philosophy, each will reach the market at the lowest possible cost.

Helios II comes complete with dual drive, controller, system diskette with DOS, power supply, case, all necessary cables and full systems documentation. A 12K assembly language program to test and report on every aspect of your unit is included too.

Helios II loads an 8000 byte program with a look up in the system directory in 0.3 sec....a speed which becomes truly significant when you are working on two 100K source files to create a third, adding up to a total of 200,000 bytes.

"Firm sectored" Controller raises disk storage to 386,000 bytes per diskette

The Helios controller is a genuine performance breakthrough, increasing formatted data capacity per diskette surface to over 386,000 bytes and at the same time assuring higher reliability than the older IBM format. Standard Helios II storage capacity is over 750,000 bytes. With two dual drives, capacity can be doubled to 1.5M bytes.

Asynchronous data transfers are made directly to memory at an effective rate of one-half million bytes per second. A sixteen byte fifo memory accumulates the data to or from the drives, freeing the computer for useful work. A standard hardware CRCC error test is performed on each transfer of data and an optional read-after-write verification mode is easily selected. The controller requires at least one S-100 bus slot and is fully compatible with Sol, Altair or IMSAI systems.

No need to buy special, expensive diskettes – the controller will pre-format any standard 32 hole "floppy" diskette.

Software

PTDOS 1.4.0 is a proven disk operating system with total file and memory management. Features include:

Complete management of static, dynamic user buffers.

Device files for generality of input/output operations.

System calls for complete file operations from external programs.

Three level, triple option error handling/trapping.

Random/Indexed Files for direct positioning to any word of a file, anywhere on the disk(s).

Command Line Interpreter accepts and executes a string of commands from you or a file.

System utility call performs a random search to the utility operation of your choice.

Helios II can be configured and reconfigured for any size buffer area. Over 40 files can be open at one time. System calls provide standardized access for all file operations from external programs and routines.

The Command Interpreter accepts input from the current command input file to provide direct file operations from the keyboard or another file. Support program calls are identical to commands, but executed outside of the system area (e.g. in low memory).

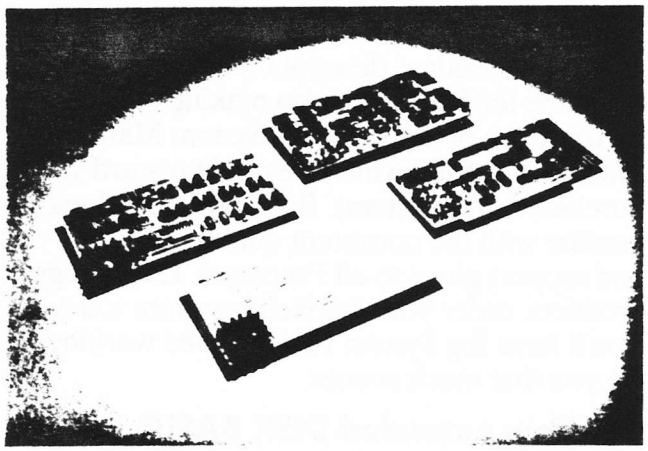
Want more information

A full product description of Helios II is available for $1. We are also making the PTDOS 1.4.0 portion of the Helios II System Manual available for $20. (which we credit toward your purchase of the system). But, if you are already familiar with the consistent quality, features and support given to all Processor Technology products, order your Sol-Helios system today. You'll have Big System Performance working for you that much sooner.

New extended DISK BASIC

Further increasing the value of your Helios II is our extended DISK BASIC. This powerful language offers advanced string and math functions plus direct commands (SAVE, RESAVE, ASAVE, KILL and XEQ) and program statements. DISK BASIC is the only available small computer BASIC with powerful disk file handling commands, statements and functions. These features make complex application programs for inventory control, data reduction and general accounting run ten times more efficiently on the Helios system.

E. Interfaces



When we talk about making the complete small computer, we mean interfaces, too. Nothing is left out. There's a video display module designed to work with computing equipment you may already have or auxiliary equipment you may need. There's the Computer Users Tape System so you can add additional audio cassette tapes for expanded program and data storage/interchange. There's a wire wrap extender board for anyone who does prototyping. If you're troubleshooting, you can see what you're fixing with Processor Technology's Extender Board. You can handle any additional input/output needs of your system with our 3 P+S Input Output Module.

In sum, Processor Technology has built every basic element you need into Sol for integral operation. And we have generated the extra equipment for use with peripheral devices or other existing computer you may have. Processor Technology is dedicated to helping you get optimum computer performance.

VDM-1 Video Display Module

We call it the communicator. It provides almost Sol-like performance for Altair and IMSAI computers. It's a high speed module which has 1024 bytes of random access memory, scrolling and multiple programmable cursor circuitry. Sixteen 64-character lines are generated in a large easy to read upper and lower case font. Data handled by the read/write on-card memory is displayed instantaneously with no interference to the processor. Top display scroll speed is 2000 lines per minute!

VDM-1 lets you display white on black or black on white. VDM-1 offers EIA video output. Terminal mode software comes with the module at no extra cost so you can use it with your existing programs. Most Processor Technology software packages already include versions of these display driver routines, so no time comsuming software patching is necessary. The VDM-1 can be used in Sol systems to add a second display output for expanded special applications.

CUTS: The computer users tape system

Here's the low cost high speed audio cassette interface for computer program and data storage interchange.

Operate at 300 bits per second or 1200 bits per second in the new Processor Technology CUTS format, upward compatible with the "Byte/Kansas City" standard. (see Popular Electronics, p. 86, March 1976)

Using CUTS you can load programs ten times faster than with a teletype paper tape reader. You can load Processor Technology BASIC in 58 seconds. There are no critical adjustments. Just about any ordinary cassette recorder will do. CUTS has AGC in both read and write modes. So you won't lose bits at 1200 or 300 baud.

Software on CUTS cassettes costs less than equivalent paper tape.

The following software for the CUTS module is available for $11, all on one cassette.

A. CUTER™— Computer Users Tape Entry and Retrieval monitor program.
B. BASIC-5 with CUTER compatible commands implemented.
C. Lunar Lander written in BASIC-5.

Many more programs are under development. You can reasonably expect a new one every few weeks.

Wire Wrap and Extender Boards

Wire wrap boards are designed for prototyping. Create custom interfaces or whatever your fancy dictates.

WWB has a "universal" seven-row pattern of pads on .3" centers, so standard 14, 16, 24 and 40 pin DIP IC sockets can be plugged right in. Power and ground are dedicated to pins 16 and 8 respectively (for 16 pin DIPs). Converts to other IC sizes easily. Use up to 62 sixteen-pin DIP ICs; six extra wirewrap socket positions have been set aside for wire wrap connections to any S-100 bus computer (Altair or IMSAI).

Use the Extender Board to help you troubleshoot any S-100 bus compatible module. Plug in a glitchy module 5" above the mother board for easy scope, VTVM or logic probe. Sol-20 systems already have a built-in extender connector on the back plane assembly.

3P+S Input Output Module

Processor Technology's 3P+S input/output module offers a low cost way to handle virtually all the I/O needs of any S-100 bus compatible computer system.

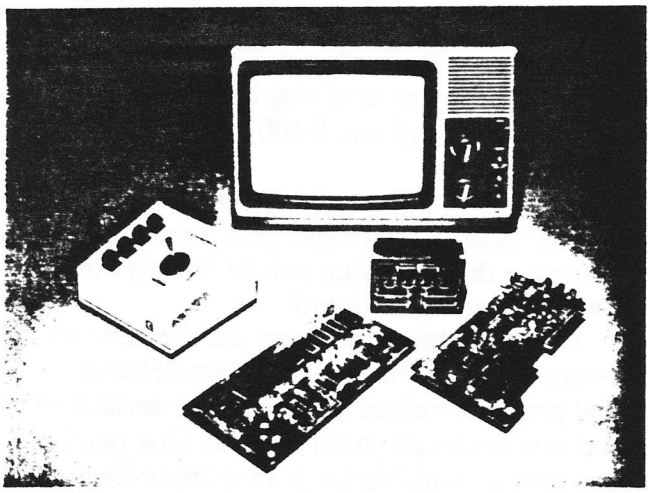
The 3P+S has two 8-bit parallel I/O ports, with full handshaking logic, plus a serial I/O port with a data rate that can be set anywhere between 35 and 9600 baud.

One parallel output port can be used to set up control conditions for both parallel and serial ports, as well as for setting the serial I/O baud rate under program control. One parallel input port is available for polling Input Data flags and External Device flags, and for checking the serial I/O error flags. You can implement full handshaking with both input and output peripherals.

Interfacing to the Sol System, Altair 8800, or IMSAI 8080 vectored interrupt bus is provided by a jumper selectable option which allows any of the UART (Universal Asynchronous Receiver Transmitter) error flags or handshaking signals to generate interrupts. (A Vectored Interrupt Module is also required for this mode of operation.)

Addressing of the module is selectable to any of 64 address segments within the range of 256 I/O addresses.

F. Peripherals



Peripherals

Processor Technology has selected a number of quality peripheral devices from other manufacturers to help you put complete systems together for many different applications. All these devices are only available factory assembled and tested.

TV-Monitor

Here's an 11" diagonal completely solid state black and white television specially modified for use with the Sol or VDM-1 units. A switch allows use as either a standard UHF/VHF television or as a video monitor. These units provide extremely crisp and stable displays and are fully grounded for safety. Manufactured by Panasonic.

High Speed Paper Tape Reader

All programs produced on paper tape for the 8080 may be loaded into Sol at up to 1000 characters/second with this handy low cost paper tape reader. The OP-80 is completely solid state and has no moving parts. The unit comes with a cable for plugging directly into the Sol Parallel Data Interface connector. Made by Oliver Audio Electronics.

PROM Programmer

The Bytesaver™ PROM programmer gives you two powerful features:

1) fast easy programming of 2708 Erasable Programmable Read-Only Memories (EPROMs).

2) Sol-bus compatible non-volatile storage of up to 8192 bytes of program.

The Bytesaver can be used to permanently store programs for use on custom application Sol personality modules or for expanded permanent storage.

Use the Bytesaver in any application where your special programs need to be permanently stored yet instantly accessed by the computer. Complete driving software included at no extra cost. Manufactured by Cromemco.

Multi-channel Analog Interface

The D+7AI/O™ module is the low cost efficient way to interface the Sol System's digital computer with the analog world. Use this module when joysticks, instruments and amplifiers, voltage and temperature sensors or any other analog device needs to be controlled or monitored by the Sol Computer.

Provided are:

- 7 multiplexed analog input channels for Analog to Digital conversion with 8 bit resolution and 5.5 microsecond conversion time.
- 7 Digital to Analog output channels with 8 bit resolution.
- 8 bit parallel interface post for digital control applications.
- +2.56 to −2.54 VDC input and output signal range (20mV monotonic increments).

The D+7AI/O is software compatible with the ALS-8 development system and PT8K BASIC language. Manufactured by Cromemco.

JS-1 Joystick

The Joystick is the fastest data entry method for interactive prompted programs and games. The JS-1 has both two axis analog outputs and four on-off switches. Requires D+7AI/O module. Manufactured by Cromemco.

See Sol now at your nearby dealer

ALABAMA

C.P. Computerland
1550 Montgomery Hwy
Birmingham, AL 35226
(205) 979-0707

ARIZONA

Byte Shop Tempe
813 N. Scottsdale Rd.
Tempe, AZ 85281
(602) 894-1129

Byte Shop Phoenix
12654 N. 28th Dr.
Phoenix, AZ 85029
(602) 942-7300

Byte Shop Tucson
2612 E. Broadway
Tucson, AZ 85716
(602) 327-4579

CALIFORNIA

The Byte Shop
1514 University Ave.
Berkeley, CA 94703
(415) 845-6366

Byte Shop Computer Store
6041 Greenback Lane
Citrus Heights, CA 95610
(916) 961-2983

Computer Center
1913 Harbor Blvd.
Costa Mesa, CA 92627
(714) 646-0221

Data Consultants, Inc.
2350 W. Shaw, Suite 114
Fresno, CA 93711
(209) 431-6461

Bits 'N Bytes
679 S. State College Blvd.
Fullerton, CA 92631
(714) 879-8386

The Byte Shop
16508 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-2421

The Byte Shop
1063 El Camino Real
Mountain View, CA 94040
(415) 969-5464

Digital Deli
80 W. El Camino Real
Mountain View, CA 94040
(415) 961-2828

The Computer Mart
624 West Katella #10
Orange, CA 92667
(714) 633-1222

The Byte Shop
2227 El Camino Real
Palo Alto, CA 94306
(415) 327-8080

Byte Shop
496 South Lake Ave.
Pasadena, CA 91101
(213) 684-3311

The Computer Store
of San Francisco
1093 Mission Street
San Francisco, CA 94103
(415) 431-0640

Byte Shop
321 Pacific Ave.
San Francisco, CA 94111
(415) 421-8686

The Byte Shop
2626 Union Avenue
San Jose, CA 95124
(408) 377-4685

The Computer Room
124H Blossom Hill Rd.
San Jose, CA 95123
(408) 226-8383

The Byte Shop
509 Francisco Blvd.
San Rafael, CA 94901
(415) 457-9311

The Byte Shop
3400 El Camino Real
Santa Clara, CA 95051
(408) 249-4221

Recreational Computer
Centers
1324 South Mary Ave.
Sunnyvale, CA 94087
(408) 735-7480

Byte Shop of Tarzana
18424 Ventura Blvd.
Tarzana, CA 91356
(213) 343-3919

The Byte Shop
2989 North Main St.
Walnut Creek, CA 94596
(415) 933-6252

Byte Shop
14300 Beach Blvd.
Westminster, CA 92683
(714) 894-9131

COLORADO

Byte Shop
2040 30th St.
Boulder, CO 80301
(303) 449-6233

FLORIDA

Sunny Computer Stores
University Shopping
Center
1238A S. Dixie Hwy.
Coral Gables, FL 33146
(305) 661-6042

Delta Electronics
2000 U.S. Hwy. 441 East
Leesburg, FL 32748
(904) 357-4244

Byte Shop of Miami
7825 Bird Road
Miami, FL 33155
(303) 264-2983

Microcomputer
Systems Inc.
144 So. Dale Mabry Hwy.
Tampa, FL 33609
(813) 879-4301

GEORGIA

Atlanta Computer Mart
5091-B Buford Hwy.
Atlanta, GA 30340
(404) 455-0647

ILLINOIS

The Numbers Racket
623½ South Wright St.
Champaign, IL 61820
(217) 352-5435

itty bitty machine co.
1316 Chicago Ave.
Evanston, IL 60201
(312) 328-6800

Reeves Communications
1550 W. Court St.
Kankakee, IL 60901
(815) 937-4516

itty bitty machine co.
42 West Roosevelt
Lombard, IL 60148
(312) 620-5808

INDIANA

The Data Domain
406 So. College Ave.
Bloomington, IN 47401
(812) 334-3607

The Byte Shop
5947 East 82nd St.
Indianapolis, IN 46250
(317) 842-2983

The Data Domain
7027 N. Michigan Rd.
Indianapolis, IN 46268
(317) 251-3139

The Data Domain
219 West Columbia
West Lafayette, IN 47905
(317) 743-3951

KENTUCKY

The Data Domain
3028 Hunsinger Lane
Louisville, KY 40220
(502) 456-5242

MICHIGAN

The Computer Store
of Ann Arbor
310 East Washington
Ann Arbor, MI 48104
(313) 995-7616

Computer Mart
or Royal Oak
1800 W. 14 Mile Rd.
Royal Oak, MI 48073
(313) 576-0900

Genral Computer Store
2011 Livernois
Troy, MI 48084
(313) 362-0022

NEW JERSEY

Hoboken Computer Works
No. 20 Hudson Place
Hoboken, NJ 07030
(201) 420-1644

The Computer Mart
of New Jersey
501 Route 27
Iselin, NJ 08830
(201) 283-0600

NEW YORK

The Computer Mart
of Long Island
2072 Front Street
East Meadow, L.I.,
NY 11554
(516) 794-0510

Synchro Sound
Enterprises
193-25 Jamaica Ave.
Hollis, NY 11423
(212) 359-1489

The Computer Shoppe
444 Middle Country Rd.
Middle Island, NY 11953
(516) 732-3086

Audio Design Electronics
487 Broadway, Ste. 512
New York, NY 10013
(212) 226-2038

The Computer Mart
of New York
118 Madison Ave.
New York, NY 10001
(212) 686-7923

The Computer Corner
200 Hamilton Ave.
White Plains, NY 10601
(914) 949-3282

OHIO

Cybershop
1451 S. Hamilton Rd.
Columbus, OH 43227
(614) 239-8081

OKLAHOMA

High Technology
1020 West Wilshire Blvd.
Oklahoma City, OK 73116
(405) 842-2021

OREGON

Byte Shop Computer Store
3482 S.W.
Cedar Hills Blvd.
Beaverton, OR 97005
(503) 644-2686

The Real Oregon
Computer Co.
205 West 10th Ave.
Eugene, OR 97401
(503) 484-1040

Byte Shop Computer Store
2033 S.W. 4th Ave.
Portland, OR 97201
(503) 223-3496

RHODE ISLAND

Computer Power, Inc.
M24 Airport Mall
1800 Post Rd.
Warwick, RI 02886
(401) 738-4477

SOUTH CAROLINA

Byte Shop
2018 Green Street
Columbia, SC 29205
(803) 771-7824

TENNESSEE

Microproducts & Systems
2307 E. Center St.
Kingsport, TN 37664
(615) 245-8081

TEXAS

Byte Shop
3211 Fondren
Houston, TX 77063
(713) 977-0664

Computertex
2300 Richmond Ave.
Houston, TX 77098
(713) 526-3456

Interactive Computers
7646½ Dashwood Rd.
Houston, TX 77036
(713) 772-5257

The Micro Store
634 So. Central
Expressway
Richardson, TX 75080
(214) 231-1096

VIRGINIA

The Computer Systems
Store
1984 Chain Bridge Rd.
McLean, VA 22101
(301) 460-3634

Media Reactions Inc.
11303 South Shore Dr.
Reston, VA 22090
(703) 471-9330

WASHINGTON

Byte Shop Computer Store
14701 N.E. 20th Ave.
Bellevue, WA 98007
(206) 746-0651

The Retail Computer Store
410 N.E. 72nd
Seattle, WA 98115
(206) 524-4101

WISCONSIN

The Milwaukee
Computer Store
6916 W. North Ave.
Milwaukee, WI 53213
(414) 259-9140

CANADA

Trintronics
160 Elgin St.
Place Bell Canada
Ottawa, Ontario K2P 2C4
(613) 236-7767

First Canadian Computer
Store, Ltd.
44 Eglinton Ave. West
Toronto, Ontario M4R 1A1
(416) 482-8080

The Computer Place
186 Queen St. West
Toronto, Ontario M5V 1Z1
(416) 598-0262

Pacific Computer Store
4509-11 Rupert St.
Vancouver, B.C. V5R 2J4
(604) 438-3282

All components sold by PROCESSOR TECHNOLOGY CORPORATION are purchased through normal factory distribution and any part which fails because of defects in workmanship or material will be replaced at no charge for a period of 3 months for kits, and one year for assembled modules, following the date of purchase. The defective part must be returned postpaid to PROCESSOR TECHNOLOGY CORPORATION within the warranty period.

Any malfunctioning module, purchased as a kit and returned to PROCESSOR TECHNOLOGY within the warranty 3 month period, which in the judgement of PTCO has been assembled with care and not subjected to electrical or mechanical abuse, will be restored to proper operating condition and returned, regardless of cause of malfunction, with a minimal charge to cover postage and handling.

Any modules purchased as a kit and returned to PTCO which in the judgement of PTCO are not covered by the above conditions will be repaired and returned at a cost commensurate with the work required. In no case will this charge exceed $20.00 without prior notification and approval of the owner.

Any modules, purchased as assembled units are guaranteed to meet specifications in effect at the time of manufacture for a period of at least one year following purchase. These modules are additionally guaranteed against defects in materials or workmanship for the same one year period. All warranted factory assembled units returned to PTCO postpaid will be repaired and returned without charge.

CONDITIONS and EXCLUSIONS

This warranty is made in lieu of all other warranties expressed or implied and is limited in any case to the repair or replacement of the module involved.

The warranty herein extends only to the original purchaser-user and is not assignable or transferrable.

Processor Technology Corporation is under no obligation to extend this warranty to any product for which a Warranty Registration Card has not been completed and mailed to Processor Technology Corporation within fifteen (15) days after date of delivery.

Processor Technology Corporation

6200 Hollis Street
Emeryville, CA 94608
415/652-8080

Printed in USA 5/77 100M

Sol: the inside story

by Lee Felsenstein

Lily Hou

"I designed the Sol!"

These words are made to be spoken from a pinnacle of technical authority, preferably by a gimlet-eyed Herr Doktor who pursues exact solutions to the nineteenth decimal place and who reigns over a limitless sea of subordinates slaving away over rows of drafting boards.

Or they could come from a furry little gopherlike creature with a piece of string for a belt who sleeps all day and occasionally surfaces to deposit a few dog-eared pages of scrawled diagrams with his custodians.

Since I fit neither of these descriptions, I hesitate to make that claim (except as part of a put-down), for it is only partly true. Besides, as I look over the reasons for making certain design decisions along the way, I am struck by the fact that most of those reasons had little to do with the ultimate advantages of the decisions.

The Sol, therefore, got designed—partly by me, partly by Bob Marsh, and partly by chance and circumstance. My description of that process is intended to instill confidence in those who feel that there are great secrets involved in the design of products and that mastery of most of the universe is a prerequisite to successful design. It is also intended as a warning to those who think that the design process is deterministic and uncomplicated.

WHAT IS THIS THING CALLED SOL?

For the benefit of future historians, I shall state that the Sol is a single-board computer built around the 8080 microprocessor and the S-100 bus structure. It incorporates an integral video alphanumeric display circuit, serial and parallel interfaces, and random-access and read-only memory on the board along with an audio cassette tape interface. A keyboard plugs into a connector on the board, and a video signal comes off through a coaxial cable. Regulated DC power is supplied to the board through another push-on connector, and that's all that is needed to make it compute.

Sol's main feature is a 100-pin edge connector that provides all the signals of the S-100 bus to any number of memory, I/O, or other peripheral cards available from different sources. An important secondary feature is the "personality module," a tiny (3-by-1½-inch) printed circuit card on which sits the ROM. The personality module plugs into a small edge connector on the Sol board. By this means the personality of the Sol can easily be changed without technical skill. Also, all the serial, parallel, and audio connectors are mounted along one edge of the card, requiring no external harness or connector assembly.

The Sol-10 and the Sol-20 both put this card, along with a power supply, into a metal chassis with walnut sides and a typewriter-style keyboard. The Sol-20 has a five-slot "daughter board" that plugs into the 100-pin edge connector and itself provides five more similar connectors, so that cards of the S-100 type can be plugged in within the cabinet. It has additional power supply capacity to feed these extra cards.

BEFORE THE BEGINNING

In 1974 I was helping to run a public-access "computerized bulletin board" system (called Community Memory), which was essentially a labor of love for me and several other people. It had two terminals in public where people could come in off the street and enter information items as well as search for them. Since the terminals were unattended, hardware reliability was an obvious problem, especially when we postulated much larger systems.

My way out of this future problem

was to design an all-purpose "convivial cybernetic device" as a terminal/concentrator/processor—in such a way that amateurs would be encouraged to get their hands on it. In theory, each place where one of these "Tom Swift Terminals" was installed would develop a computer club. Then, when a terminal broke down, relief would be a local matter, and people would not have to place their trust in a remote maintenance structure.

It was by placing a notice about this conceptual design on the system and inviting respondents to form a discussion group that I met Bob Marsh for the second time. (The first had been during our college days when we lived in the same co-op residence hall at Berkeley. We did not share many interests then.)

Now, however, Bob had raised himself to the state of an unemployed electronics engineer (self-taught) who had nearly won an encounter with a glorified version of Don Lancaster's TV Typewriter that he had built, improved, and fixed from scratch. After a while Bob suggested that I go in with him on the rental of a workshop. I agreed, and we signed a three-year lease on a garage in industrial Berkeley, commencing from January of 1975. I moved my workshop out of my living room, and Bob took up residence in the upstairs office, trying to find a product to manufacture. A plan to produce a limited-edition digital clock with a fancy wood case never materialized, which was a disappointment both for Bob and for Steve, a friend of his who did woodworking (we shall hear more of this later). Bob spent much time investigating the possibilities for a logic analyzer similar to others then available, doing much design before he gave it up as impractical.

In March 1975 I took Bob to the second meeting of the Homebrew Computer Club, where about twenty-five people stood around the first Altair 8800 to reach the area and watched it blink its front panel lights. That was all it could do, since it had no I/O circuitry and only 256 bytes of memory. There was a lot of empty space inside that cabinet.

Processor Technology was founded as a partnership in April of that year (I was not one of the partners) and began designing ROM, RAM, and I/O cards for the Altair. Incorporation followed in July, and I began to get bits and pieces of work—redrawing schematics, writing preliminary manuals, and other minor chores.

In July Bob finally got to me. He had a proposition: he would pay me to design the video display section of the Tom Swift Terminal. In October the VDM-1 video display module first saw Revision A. The Tom Swift Terminal never did get designed exactly as I dreamed it. But, as the hordes of amateur computer enthusiasts swarmed out of the woodwork, its purpose was being fulfilled.

THE SOL SOLUTION

I like to say that, in the process leading to the birth of the Sol, Leslie Solomon, technical editor of *Popular Electronics* magazine, performed the act equivalent to that of the male. I still don't know who solicited whom, but Les agreed to carry a construction article on "an intelligent terminal" on the cover if a working model could be supplied in thirty days. This proposal was made in the middle of November 1975. I was summoned to Bob's office.

Bob tactfully asked me if I thought such a project was impossible. I reluctantly admitted that I did not but strongly suggested that an unintelligent terminal be designed, mainly because I wanted badly to exercise the features designed into the VDM-1 for such an eventuality. Besides, I had already turned down an offer from Bob to design an 8080 CPU board for Processor Technology.

Alternate designs were roughed out and prices compared. Both were for terminals using the basic circuitry of the VDM-1. His had an 8080 thrown in, mine had decoders and counters. Mine was cheaper, but only by about ten dollars. Gradually I came to realize that if the inevitable were to happen, it would be better to be on the inside than on the outside. After a day or two of discussions, I agreed to contract to do the design. I pulled out of a volunteer project on the grounds that "the roof was about to fall in again." And somewhere in the midst of all this I looked up at Bob and said: "Let's advertise it as having 'the wisdom of Solomon.'" From the comment came the name Sol, which is meant to be written in biblical-movie-poster letters chiseled out of stone. Les will never live it down.

ROMtutorial ROMtutorial

Single Board Computer: *Electronic devices are built nowadays on "printed circuit cards" of fiberglass with patterns of copper foil instead of wires. In the old days, ten years ago, computers had hundreds of these boards, each containing a tiny portion of their electronics. Now things have shrunk so much that an entire computer can fit on one printed circuit board.*

Microprocessor: *the "thinking" section of a computer is called the central processing unit (CPU) or just the processor. If it's so small that you need a microscope to examine it, it's called a microprocessor.*

S-100 bus structure: *a bus is a wire connected to many places. Usually it's used to carry electricity for power, but, in a computer, very fast, low-power electrical impulses are sent between sections on buses. Since there are a lot of these signals happening at once, computer buses have a lot of wires in them. The S-100 is a 100-wire bus used by many personal computers. Because they all have the same pattern of four interconnections, the plug-in board from one will work (usually) when plugged into another S-100 machine. The S means standard.*

Integral video alphanumeric display circuit: *electronic circuitry which produces a signal that can be connected to a TV set and that causes the set to display letters and numbers on the screen. It is integral because it is built into the computer.*

ROMtutorialROMtutorial

Serial and parallel interfaces: *An interface is the dividing line between two electronic devices. A wire or cable usually goes across an interface carrying electrically-coded information. If the cable carries several different signals at once, it's a parallel interface. If the information moves in a sequence through a single wire the interface is serial.*

Random-access Memory: *memory like a set of pigeon-holes, into any of which the computer can put new information or from any of which it can read old information. The computer can choose any pigeon-hole (or address) at any time.*

Read Only Memory: *memory like a telephone directory which can only be read by the computer and not written in. It's used to hold instructions for the computer (the program).*

Push-on Connector: *a set of metal posts wired or soldered to electrical circuitry. A set of spring clips held in a plastic block can be pushed down onto these posts in order to make an electrical connection to a cable.*

Edge Connector: *a socket built as a long thin slot in a plastic block. A number of spring fingers are held in the block. If a printed circuit-board edge is pushed into the slot the spring fingers will "wipe" against both sides of the board and make contact with the copper foil patterns.*

I/O: *Input/Output. The electrical channels through which the computer moves information to and from the outside world.*

"Who, me?" Leslie Solomon, Technical Editor of Popular Electronics, *looks startled as the first Sol speaks its words of wisdom.*

In the process leading to the birth of the Sol, Leslie Solomon performed the act of the male. I still don't know who solicited whom.

Bob had already developed an architecture for the device. It involved taking the on-card memory and I/O devices, disconnecting the S-100 "DI" data input bus, and creating a separate low-drive "internal bus" that would get data to the CPU through a four-way data selector or multiplexer. One input of this would be the regular S-100 DI bus, another the internal bus, and the other two would be used for keyboard and parallel data inputs. The reasoning behind this was to minimize the need for high-power tristate drivers on the card and to allow the low-power tristate output features of the RAMs, ROMs, and UARTs to serve their intended functions.

I should explain here that designing for Bob Marsh can be somewhat of a trial. At least at that time, when he had little else with which to concern himself, he was continually turning up with new features and economies that he suddenly wanted incorporated in the design. He would explain the problem or opportunity and then preface his technical solution with an inevitable "All's ya got to do is...." This would be forgivable if he were not so often right and possessed of a truly useful and valuable idea. Were the designer a prima donna, the relationship would terminate after the second such incident, with the designer fuming about "professionalism" and "interference." Of course, since my workshop was in the same room as his, I could not have gotten very far if I had wanted to stamp out in a rage. The situation did, however, call heavily on my sense of futility, absurdity, and ultimate irrelevance.

A few days after I started on Sol, Bob had another idea. He wanted all clocks on the card derived from a single crystal. The VDM-1 had been using a 18.4784-Mhz crystal, and Bob had been doing some division problems with his calculator. It seemed to him that not one but three relevant clock speeds could be extracted from this frequency, allowing operation with 2.5-Mhz and 3-Mhz 8080 chips if

desired. I protested, but without -ounds, and very soon gave in and esigned a simple clock generator which allowed (through a variable-modulo flip-tail ring counter, if you must know) the kind of clocks Bob wanted.

bly MITS's) boards did not use the DBIN signal and were therefore useless with Sol, unless modified, is still a source of defensiveness in discussions of the design. I take the position that Bob made me do it, and he takes the position that history will absolve him.

Bob tactfully asked me if I thought the project was impossible.

After a decent interval Bob informed me that he would like to increase the crystal frequency to 14.31818, which would be necessary if color video peripherals were to operate with it. Please to redesign all relevant circuits accordingly. I fumed and grumbled but found a way out eventually. After all, the clock frequency to the 8080 would be of a 488-nanosecond period, a whole 1.8 percent shy of the spec sheet minimum! Why worry?

The biggest upset actually had the flimsiest reason. Bob wanted to be able ɔ extend the S-100 bus of the Sol to an .ternal cabinet through a flat ribbon-style cable, which came in 50-conductor maximum widths. With my avid agreement, Bob wanted to ensure that there were plenty of ground return lines sprinkled through this cable—a practice that had been neglected on the Altair and that probably caused many of the noise problems associated with it. To do this would require more than 100 wires.

Bob and I did some figuring while driving down to a Homebrew Club meeting, and he reached the conclusion that it would be permissible to take the DI and DO buses and connect them in parallel, making one single DIO bus. We assumed that every manufacturer who was anybody was doing the same as Processor Technology, using the DBIN signal from the 8080 as an enabling signal for data to the DI bus. DBIN could therefore be used at the Sol as a "direction signal" for the DIO bus and data could be sent both ways on one set of wires. Think of ll the cable you'd save!

As it turned out, those eight extra traces saved on the Sol board nearly made the difference between a buildable board and an impossible one. The fact that some manufacturers' (nota-

As the design progressed, we realized that we were building a general-purpose computer rather than just "an intelligent terminal," but the decision was made to soft-pedal the fact until the last possible moment. Once published, all the fuss possible was to be made about its general-purpose nature; but until it actually saw print, it was to be treated first as a terminal.

To jump ahead a bit, when I finally delivered the working prototype to Les Solomon's desk and pointed out its salient features, his eyebrows began twitching. Why couldn't he, he wanted to know, plug in a ROM board with BASIC burned in (as he could do with a Bytesaver) and run stand-alone? I smiled my blandest smile and muttered, "Beats me."

THE BIG PUSH

Originally I had been given to believe that I would be required by the terms of the contract only to provide a schematic diagram and to help interpret it for the benefit of the layout artist. When it came time to start lay-

Lee Felsenstein after the deed.

ROMtutorialROMtutorial

Multiplexer: *an electronic circuit which can choose one of several inputs and route the signals at that input to its output.*

Internal Bus: *Eight bus-type wires which are connected to most of the devices on the Sol which feed information to the processor. The information moves in eight-bit units (bytes), one wire for each bit.*

Tri-State Drivers: *electronic circuits used for feeding signals to a bus. They can either force the bus wire to a high voltage or a low voltage, or they can let the bus "float" at whatever voltage other drivers decree.*

Variable-modulo flip-tail ring counter: *an electronic worm continually chasing its tail. The worm's back is black and its belly is white. Each time around, it makes a half twist. A line of ants does close-order drill using the color of the worm as a cadence (clock signal). Since worms travel at a steady rate, the cadence can be changed by changing the length of the worm (variable-modulo).*

DI Bus: *Eight bus-type wires which carry electrical data signals from S-100 boards into the processor. Short for Data Input Bus.*

DO Bus: *Data Output Bus. Same as DI Bus, except that the electrical signals are carried out of the processor board to all of the S-100 boards.*

DBIN Signal: *Data Bus In. A timing signal which the microprocessor uses to tell anyone who cares that it can accept electrical data (one byte only) going in.*

ing tape, I discovered that I was expected to pitch in with a will and help stick down the many miles of crepe tape that would be required to produce the printed circuit artwork. Time was of the essence, as I agreed, and the layout artist who had been engaged was somewhat out of his depth with a 110-package layout to be done at four-to-one enlargement. Besides, he apparently didn't believe in making preliminary pencil sketches to test possible routings.

Steve the woodworker had built a large light table to Bob's specifications, and this was set up in the only available space in the now crowded garage – a loft above some offices that Processor had installed. I personally made sure that an electrical conduit running at forehead level was padded and tried to get someone to put up a fence to prevent one of us from stepping backwards off the edge of the loft. That, the knocking of heads against rafters, and the sticking or scratching of various parts of the head with needle-point X-acto knives, which we constantly had to hand, were the main hazards of working there.

In addition, there was the chill at night and the heat that accumulated during the day. People downstairs never had the same opinion about a comfortable temperature as we did. We soon settled into an insane schedule of fourteen- to seventeen-hour days, seven days a week. I kept going on orange juice, the younger layout man used Coke. He succumbed at the very end and I had to finish up alone. The task took almost three weeks of standing up looking down into a fluorescent-lit white background on which crawled worm tracks of black tape.

The crowning moment came when Bob changed his mind for the last time.

The crowning moment came when Bob changed his mind for the last time about which side he wanted the 100-pin connector to emerge from (this was determined by whether the S-100 boards were to mount above or below the Sol board). Layout had already started, and fortunately that connector lay in the other guy's area of concern. He had to cross over about seventy-five traces in order to flip the connector, and he did not succeed. Some traces were brought out to dead-end pads with no hole through them, leaving some of the pins to be connected by tack-soldered jumper wires later during assembly.

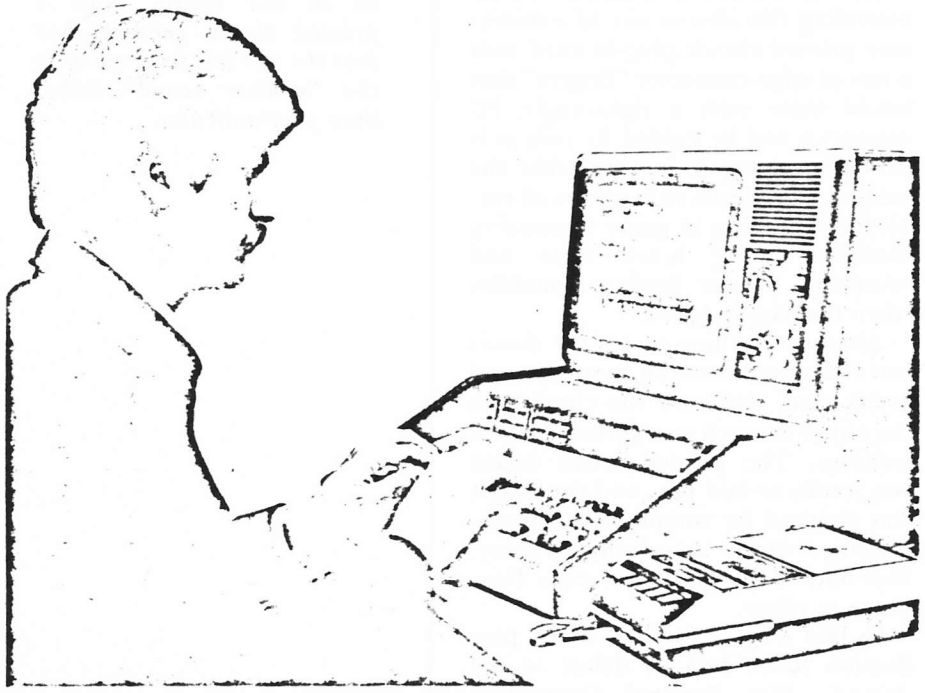
I should point out that this printed circuit board was the prototype; no wire-wrap or other version had been made. Only two boards were ultimately produced (by a specialty shop run by a work addict), and one of them sits on my window ledge at home to remind me of how rough things can become. Besides, I like to point out my half of the layout and how obviously superior was my technique compared with the other half.

The other board I loaded with parts, tack-soldered with almost a hundred jumper wires out of sight on the rear side where traces were incomplete due to the hasty layout, checked out, and presented to the software people to see what they could do. I have no recollection whether that was before or after Christmas of that year; I think it was before.

IN THE BAG

Bob had laid out the basic outlines of the cabinet and had patched together a power supply and a sample keyboard. One of his design criteria from the absolute beginning was that the cabinet have walnut sides of a certain height or less. Steve had told him of a great bargain to be had from center-cut pieces of walnut, which were ordinarily almost thrown away. There is some cause to believe that the primary reason for the existence of the Sol was to provide an outlet for this inexpensive wood. At any rate, Steve is now in the wilds of Wisconsin running a mill to make walnut Sol sides.

As the project proceeded, Bob was undecided as to whether a cassette interface or drive would be included in the production version. As photographic deadlines approached, he decided to put a dummy cassette drive in the cabinet. This was to symbolize the fact that cassette tape would ultimately be available in one way or another—how he did not know. The text of the article, when it appeared, mentioned nothing about it, and the mysterious tape deck has proved to be



Bonfield Associates

Bob Marsh with the Sol up and running. All's well that ends well—though in this case it seems to be only the beginning for Processor Technology.

Carrying the Sol shrouded in two paper bags, we embarked for the big-time world of New York.

the biggest single source of questions from the readership of that article.

The printed circuit board was ready forty-five days after the start of the project. About sixty days from the go-ahead, the cabinetry, power supply, and software were coming down the home stretch. Bob scheduled a night flight to New York, to save money, and informed me that I was coming; everybody went into a final home-stretch panic lasting the better part of twenty-four hours.

We made the helicopter only because my watch was fast. Bob forgot to bring his tickets to the heliport and broke numerous traffic laws going back to get them. Carrying the Sol shrouded in two paper bags, we embarked for the big-time world of New York.

Of course we got no sleep on the plane and arrived an hour too early for our purposes. We went directly to Les Solomon's home and shared breakfast with him, discovered that the Sol wouldn't work (you could see it trying behind a veil of "snow"), then took the train to Les's offices on Park Avenue, where we met the boss and everyone else. Leaving behind a trail of excuses, we emplaned for Boston, where the folks from *Byte* magazine took us to dinner and drove us up to Peterborough, New Hampshire to see the snow. I conked out on this last trip, after attempting to stay awake by gorging myself at every available opportunity. I fear that I made a poor impression by so doing.

Back at the workbench a day or so later, I traced the trouble to a tiny speck of wire obviously loosed from the shield braid of the coaxial cable as it was stripped. This crumb had been captured underneath a socket, where it had plainly stayed during all the checkout without causing mischief. In transit it had shifted to the worst possible position – shorting two obscure but critical traces together. Murphy's Law confirmed!

After that it was bundle up again, this time myself alone – but carrying my oscilloscope and tool kit as well as the Sol. An hour or two of demonstration in the offices of *Popular Electronics* convinced them that Sol would indeed work. Then it was back home directly, where I passed "Go" and collected the balance of my lump payment.

DEBUT

A great deal of work remained to be done after the prototype Sol was finished. Bob wasn't through with suggesting new things and better ways – I suspect he never will be. One major change in the design bears pointing out, though.

The personality module, which owes its name to Don Lancaster, came about as a result of unpredictable EPROM supplies. Some way had to be found to plug in three different EPROMs without taking up extra area. The first suggestion was for a "piggyback" card with upright pins that plugged onto a row of socket connectors on the board. I take credit for extending this idea to one of a miniature printed-circuit plug-in card with a row of edge-connector "fingers" that would mate with a right-angle PC connector and be guided by rails so it could be changed from outside the cabinet. This leads to fantasies of employees swapping in game personality modules during lunch time and returning to the business modules when the whistle blows.

Almost everything except the circuit was redesigned between prototype and production, and even the circuit was the object of much straightening out of wrinkles. The printed-circuit board was totally re-laid out, and the layout was digitized for computerized photoplotting. More expediting, twenty-four-hour days, and dead runs from plane to plane.

In late August the Sol, in its production form, had its debut at the Atlantic City Personal Computing Show. I understand the cocktail party was splendid. Personally, I spent the time trying to trace down a bug that I was later told was in software. ▼

ROMtutorial ROMtutorial

BASIC: *A computer program which allows anyone to use the computer and get instantaneous feedback as to whether they are doing OK or making a mistake.*

Bytesaver. *An S-100 plug-in board which can write information supplied by the processor into special read-only memory chips. Once written, the information cannot be erased by the computer but it can be read. A trademark of Cromemco.*

EPROM: *Erasable Programmable Read-Only Memory. Used on things like the Bytesaver. A chip which will hold information indefinitely after being written, but which will forget the information if exposed to ultraviolet light (the cover of the chip is clear quartz).*

Right-angle PC Connector: *A printed circuit edge-connector with pins bent 90 degrees so that once it's soldered down to its own board the slot is parallel to the board. Now a printed circuit card pushed into the slot will be parallel to the "mother board" rather than perpendicular.*

The Sol-20 Computer Terminal

By Rudolf Hirschmann

Rudy Hirschmann has written a review of the Sol computer. Rudy is a professor of German at the University of Southern California, where he is using microcomputer based systems for typesetting and for teaching.

The Sol-20 terminal computer from Processor Technology certainly makes a good first impression. It has the general appearance, shape and size of an electric typewriter with a full keyboard, attractive blue metal case and genuine walnut end plates. These would make it seem more at home in a modern office somewhere downtown rather than in my less elegant computer shack at home.

But beneath that sleek exterior lurks a genuine microcomputer that is capable of doing most everything that a state-of-the-art microcomputer ought to do. A fully operational Sol-20 consists of five circuit boards, namely the main circuit board, the plug-in personality board, the keyboard, the bus extender board and the power supply. I will discuss each of these in turn.

The main circuit board is by far the largest and most complex of the subassemblies for this computer. It combines the functions of several conventional S-100 plug-in boards, namely that of an 8080-based CPU board and a video display generator for sixteen lines of 64 characters each. In addition, it also provides a very useful combination of functions that are normally available only on several additional boards, namely one serial and one parallel I/O port, a reliable cassette I/O port that uses the "Kansas City/Byte" standard at either 30 or 120 characters per second, a modest amount of RAM *and* address decoding as well as a connector for up to 2K bytes of ROM. This ROM is actually held on a small board that plugs into the main board. A power-on-jump is also provided to this ROM, and for that reason this board is known as the personality module. More on that below.

The layout of the main board seems to be straightforward, logical and well executed. All features of this board worked as they are supposed to, although three errors in trace routing had to be corrected. This problem should be solved in boards marked Rev. E and beyond. The physical quality of this and the other boards is first rate. All traces and plated-through holes conducted properly, and a solder mask and parts labeling are provided.

Four sets of DIP switches are included on this board to allow easy selection of certain options. One switch allows changing the characteristics of the video display (which is virtually identical to Processor Technology's VDM-1), another selects the baud rate, while the third changes the word format for the serial I/O port. The final DIP switch is a thoughtful concession to those of us who have grown accustomed to a front panel, namely a set of eight sense switches connected as input port 255. While the location of these switches is a bit awkward (the cover has to be removed to get at them), it is nevertheless possible to run programs that depend upon the use of the front panel sense switches of an Altair or IMSAI system. It should be noted that the single step and the memory protect/unprotect functions are not provided for. This means that your memory boards must always go to the unprotect mode when power is turned on and then stay there. An easy modification circuit for this purpose appeared in *SCCS Interface* vol. 1, no. 4 (March, 1976), page 43, that will take care of any problems in this regard.

In order for you to assess other potential hardware/software incompatabilities with your present system, let me give you the port and memory locations implemented on the main board. Port F8 functions as a control and status port for serial I/O, F9 is the serial data channel, FA is the control and status port for parallel, cassette and keyboard I/O, FB is the cassette data port, FC is keyboard data, FD is parallel data, FE is status and control for the video display and FF is sense switch input. So far as memory allocation is concerned, personality ROM is from C000 to CFFF, system RAM is C800 to CBFF, and video display RAM is CC00 to CFFF.

Turning now to the personality plug-in module, this is available in several different implementations. As a CONSOL module it consists of a small plug-in board with two 5204 PROMs and space remaining for two more. In other words, 1K of firmware is implemented and space for an equal amount is left empty. As a SOLOS module, this same board is supplied with four 5204s. Presumably the SOLED will be supplied in the same form when it becomes available. Plug-in modules using other PROMs, such as the 2708, are also supposed to be in the works. That way all of us Bytesaver owners can implement our own firmware.

I had a chance to work with one CONSOL and two different SOLOS modules. Each of them has drivers for all I/O ports and the video display system as well as some very useful file handling firmware for the cassette I/O channel. When you record a section of memory onto tape, you must specify a starting address, a length and a name; you also have the option of specifying an execution address. All of this information is automatically stored as a header on the tape before the specified block of memory is stored. The beauty of this system becomes apparent when retrieving stuff from tape, because, although you could specify everything manually, you can

also simply specify the name mentioned above, and the firmware will then ignore all data on the tape until it encounters the same name in a header. It then transfers the block to the address that is also specified in the header. This may not be as convenient as a floppy disk, but it is a big step upward from systems that have no firmware support.

The keyboard is custom designed for the Sol terminal and is made up of high quality parts throughout. The keys feel solid and durable, and a good selection of special-function keys is included. The keys of the numeric pad (not included in all models of the SOL) are *not* simply connected in parallel with the corresponding keys on the main section of the keyboard. Instead, they are encoded separately, and the output code of the numeric pad has the eighth bit turned on, even though this bit is not used in the ASCII standard. It is easy enough to ignore this bit for most applications, but in special situations, that extra bit can be used to implement special control functions via the numeric pad. An application I am presently working on will have these keys control a set of cursors in a fairly complicated text editor.

A little quirk of this keyboard is that the key used for the backspace function does not generate the standard ASCII backspace code. This may not become a problem until you try to interface existing software with this system, but even then the solution will be straightforward. The time I found this anomaly disturbing was when I used the Sol as a remote terminal connected to a large computer over the telephone lines. In this situation it is necessary to transmit a control-H in order to have the large computer understand. This, however, shows up as a displayed character on the screen instead of an actual backspace. Moreover, the character that *is* used as a backspace in the terminal is blocked in firmware from being transmitted. In short, there is no way of implementing a satisfactory backspace in this situation. In addition, the firmware allows most (but not all) control characters coming from the large computer to be displayed on the screen. For those who will use the Sol as a remote terminal, these may be regarded as annoying factors; but they can surely be solved in firmware. Processor Technology, are you listening?

So far I have been talking about the basic components that make up many good terminal computers. What makes the Sol particularly noteworthy, however, is the inclusion of a bus extender board that adopts the essential conventions of the S-100 bus. That means that virtually all plug-in modules available for the IMSAI and Altair 8800 computers can be used in the Sol, and this shows the great advantage to the hobbyist (and manufacturer) for adopting some sensible standards. Are the other manufacturers listening?

The bus extender board plugs vertically into the main circuit board, it has plug-in positions with card guides for five horizontally oriented boards and an additional test position on top. This board has good large traces for power distribution and is double sided with plated through holes. Why all that trouble on a simple board like this? It allows for shielding, because little fingers of grounded conductors are placed between all the other bus conductors. I couldn't test them for their actual effect, and one might question their necessity on such a small backplane. But this type of design is also being adopted by others, and it is certainly indicative of fine engineering practice and is a step in the right direction.

I have saved the power supply for the end, because this is where a tendency that was evident throughout the entire machine was evident most clearly, namely the tendency to overdesign. This power supply is gutsy and will never have to be upgraded unless you somehow double or triple the size of that bus extender card.

There are two separate +8 V supplies, one supplying power only to the main mother board, and the other only to the bus extender board. The first of these is filtered by 18,000 ufd., regulated with discrete parts and protected by a crowbar circuit! It is normally called upon to supply about 1.5 A, but I loaded it down to twice this value, and everything looked fine even after an hour. The other +8 V supply is filtered by a capacitor of 54,000 ufd., and according to my measurements it can safely supply more than 10 A and still not run very hot. That translates to over 2 A per bus-extender slot, and that is certainly adequate. The + and − 16 V lines are hefty enough to handle any reasonable load.

All of these components are put together in an attractive case. The metal is fairly heavy gauge steel throughout, and everything has a very solid feel to it. The only thing I can quibble with is that in order to get at certain components, many others must first be removed. Some additional thought on this score, such as splitting the subassembly that holds the card guides in two, would make disassembly easier.

My overall impression is that the Sol-20 is a well designed machine and that many potential problems have been solved before releasing the product, a practice that should be more widely observed. As it stands, this machine still has a few problems as mentioned above, but they seem to be small ones that can either be solved in firmware or in slight adjustments in packaging.

There are a few other machines that have characteristics similar to those of the Sol-20, but none of the other single-package terminal computers presently on the market uses the 8080 and the S-100 bus. These factors alone should make the Sol-20 an attractive machine for many. The fact that the cost is relatively modest ($475 for the main circuit board only in kit form, $995 for the whole machine as a kit and $1395 assembled with the CONSOL module) should serve to add to its popularity.

SOL SYSTEM SUGGESTED RETAIL PRICE LIST (September 1, 1977)

I. Sol Systems		Order No.	Price
Complete systems are available from Processor Technology to fulfill the application requirements outlined on the Sol Solution Chart in our catalog. These systems include all necessary connecting cables and manuals.			
A. Sol System I includes Sol-20/8 with 8192-byte memory and SOLOS module, PT-872 monitor, RQ-413A cassette recorder, and BASIC 5 tape	Assembled/Tested	400100-01	$2129
	Kit	400100-02	$1649
B. Sol System II includes Sol-20/16 with 16,384-byte memory and SOLOS module, PT-872 monitor, RQ-413A cassette recorder, and BASIC 5 tape	Assembled/Tested	400200-01	$2283
	Kit	400200-02	$1883
C. Sol System III includes Sol 20/16 with 32,768-byte memory and SOLOS module, Helios II Disk System with DISK BASIC, and PT-872 monitor	Assembled/Tested	400300-01	$5450
	Kit	400300-02	$4750
D. Sol-20/8 Terminal Computer with 8KRA 8192-byte Memory Module and SOLOS Personality Module	Assembled/Tested	400400-01	$1850
	Kit	400400-02	$1350
E. Sol-20/16 Terminal Computer with 16 KRA 16,384-byte Memory Module and SOLOS Personality module	Assembled/Tested	400500-01	$1975
	Kit	400500-02	$1550
II. Sol Components			
A. Sol-PC Single Board Terminal Computer with SOLOS Personality Module	Assembled/Tested	101036-01	$745
	Kit	101036-02	$575
B. UGKPC-20 Sol cabinet, 85 key keyboard, fan, power supply and backplane expansion. Upgrades Sol-PC to Sol-20	Kit	101035	$675
C. Sol-KBD 85-key solid state Keyboard as used in Sol-20 series units	Assembled/Tested	104000	$230
D. PM 2708 Personality Module for use with 2708 EPROMs (does not include EPROMs)	Assembled/Tested	107000-01	$40
	Kit	107000-03	$30
E. SOLOS Personality Module	Assembled/Tested	107000-02	$90
	Kit	107000-04	$75
F. 220-Volt Transformer for all Sol-20 series units. Note: All Sol-20 series units and Sol Systems are available for 220 Volt, 50 Hz operation. Contact factory for pricing and delivery information.	Kit	105034	$50
III. Mass Storage Systems			
Helios II, Model 2 Disk System. Includes dual PerSci 270 floppy disk drive, cabinet, fan, S-100 bus compatible controller, power supply, system diskette with complete PTDOS software	Assembled/Tested	300000-01	$2695
	Kit	300000-02	$2395
IV. Subsystem B for all S-100 Bus Mainframes other than Sol			
A. Subsystem B70. Includes 4KRA memory, VDM-1 Video Display Module, 3P + S Parallel, Series I/O Module, CUTS Computer Users Tape System cassette interface and GPM General Purpose Memory Module	Assembled/Tested	406000-01	$829
	Kit	406000-02	$594
B. Subsystem B110. Includes 8KRA memory, VDM-1 Video Display Module, 3P + S Parallel, Series I/O Module, CUTS Computer Users Tape System cassette interface and GPM General Purpose Memory Module	Assembled/Tested	406100-01	$998
	Kit	406100-02	$730
C. Subsystem B190. Includes 16KRA memory, VDM-1 Video Display Module, 3-P + S Parallel, Series I/O Module, CUTS Computer Users Tape System cassette interface and GPM General Purpose Memory Module	Assembled/Tested	406200-01	$1095
	Kit	406200-02	$895

V. S-100 Bus Compatible Modules		Order No.	Price
Memory			
A. GPM General Purpose Memory Module with 1024-byte Read/Write memory and 2048-byte CUTER program on ROM	**Assembled/Tested**	210000-01	$169
	Kit	210000-02	$129
B. GPM-Sol. Same as GPM without Read/Write or ROM memory. These memories are included with the Sol.	**Assembled/Tested**	210000-03	$119
	Kit	210000-04	$89
C. ALS-8 ROM Resident Assembly Language Operating System with Interpretive Simulator (SIM-1) and Text Editor (TXT-2). For use with GPM or GPM-Sol. If purchased together, price includes assembly and testing. (ALS-8, SIM-1 and TXT-2 programs are copyrighted.)	**Assembled/Tested or Kit**	900014	$190
D. 2 KRO Erasable Programmable Read Only Memory Module	**Assembled/Tested**	204000-01	$89
	Kit	204000-02	$65
E. 4KRA 4096-byte Static Read/Write Memory Module	**Assembled/Tested**	201000-01	$150
	Kit	201000-02	$125
F. 8KRA 8192-word Static Read/Write Memory Module	**Assembled/Tested**	202000-01	$250
	Kit	202000-02	$225
G. 16KRA 16,384-word Dynamic Read/Write Memory Module	**Assembled/Tested**	203000-01	$399
	Semikit—assembled & inspected but not tested	203000-03	$369
Interface Modules			
H. 3P + S Parallel Serial I/O Module	**Assembled/Tested**	209000-01	$199
	Kit	209000-02	$149
I. CUTS Computer Users Tape System cassette interface (includes CUTER tape)	**Assembled/Tested**	207000-01	$140
	Kit	207000-02	$110
J. VDM-1 Video Display Module	**Assembled/Tested**	208000-01	$295
	Kit	208000-02	$199
K. WWB Wirewrap Prototyping Module	**Kit**	211000-02	$40
L. EXB Extender Board	**Kit**	212000-02	$35
VI. Peripherals			
A. PT-872 TV-Video Monitor by Panasonic		722016	$199
B. RQ-413A Cassette Recorder by Panasonic		722019	$85
VII. Software (including manual)	**Minimum Memory Required**		**CUTS cassette**
A. BASIC/5	10K	727000	$14.50
B. 8K BASIC	12K	727017	$29.00*
C. New 8080 FOCAL	8K	727024	$14.50*
D. TREK 80 Video Game	8K	727009	$14.50
E. GAMEPAC-1 Video Games	4K	727006	$14.50
F. MATHPACK Video calculator	4K	727020	$14.50
G. ASSEMBLERS: Software #1 Resident Assembler Package	8K	727022	$14.50
H. ALS-8 Resident Assembler, Simulator and Text Editor	12K	727012	$35.00
I. Software Technology Music System cassette	8K	727015	$24.50

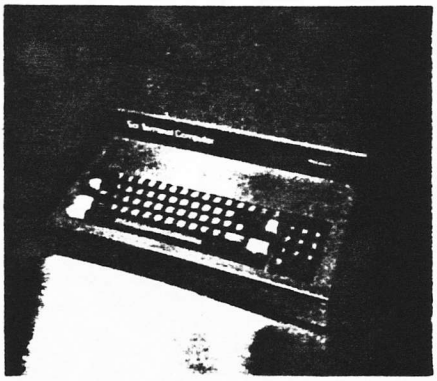
Attention

All listed software products require CUTER or SOLOS programs.

All Processor Technology and Software Technology software is distributed on an individual sale basis for personal use. No license to copy, duplicate or sell is granted with this sale. Each software package has been copyrighted by Processor Technology or Software Technology and all rights therein are reserved.

Prices and specifications are subject to change without notice.

*Available soon at your dealer.



Why Sol Should be your Small Computer Choice.

Sol-20 Terminal Computers are complete. You don't spend an extra penny for necessary interfaces. Sol computer systems start at $1350 in kit form.

We've done the software job. Only Processor Technology offers a fully implemented disk operating system for small computers: PTDOS. Our high level languages include Extended BASIC, FORTRAN* FOCAL, PILOT* and Assembler.

You can expand the Sol to handle business, engineering and research problems. Today's Sol can handle 65,536 bytes of memory and operate with a three megabyte on-line disk memory. S-100 bus compatibility means you can use a big group of standard peripherals.

Sol systems are conservatively rated. They won't quit on you. With over 5,000 in the field, we know the track record for reliable performance is outstanding.

We back Sol with the best documentation in the industry plus a factory support team to give you caring counsel by phone or letter. What's more, on site maintenance and service folks are located in over 50 cities throughout the U.S. and Canada.

So when you are shopping for a small computer, see your Sol dealer last. Then you can take your Sol with you!

For more information, please address Processor Technology Corporation, Box G, 7100 Johnson Industrial Drive, Pleasanton, CA 94566. (415) 829-2600.

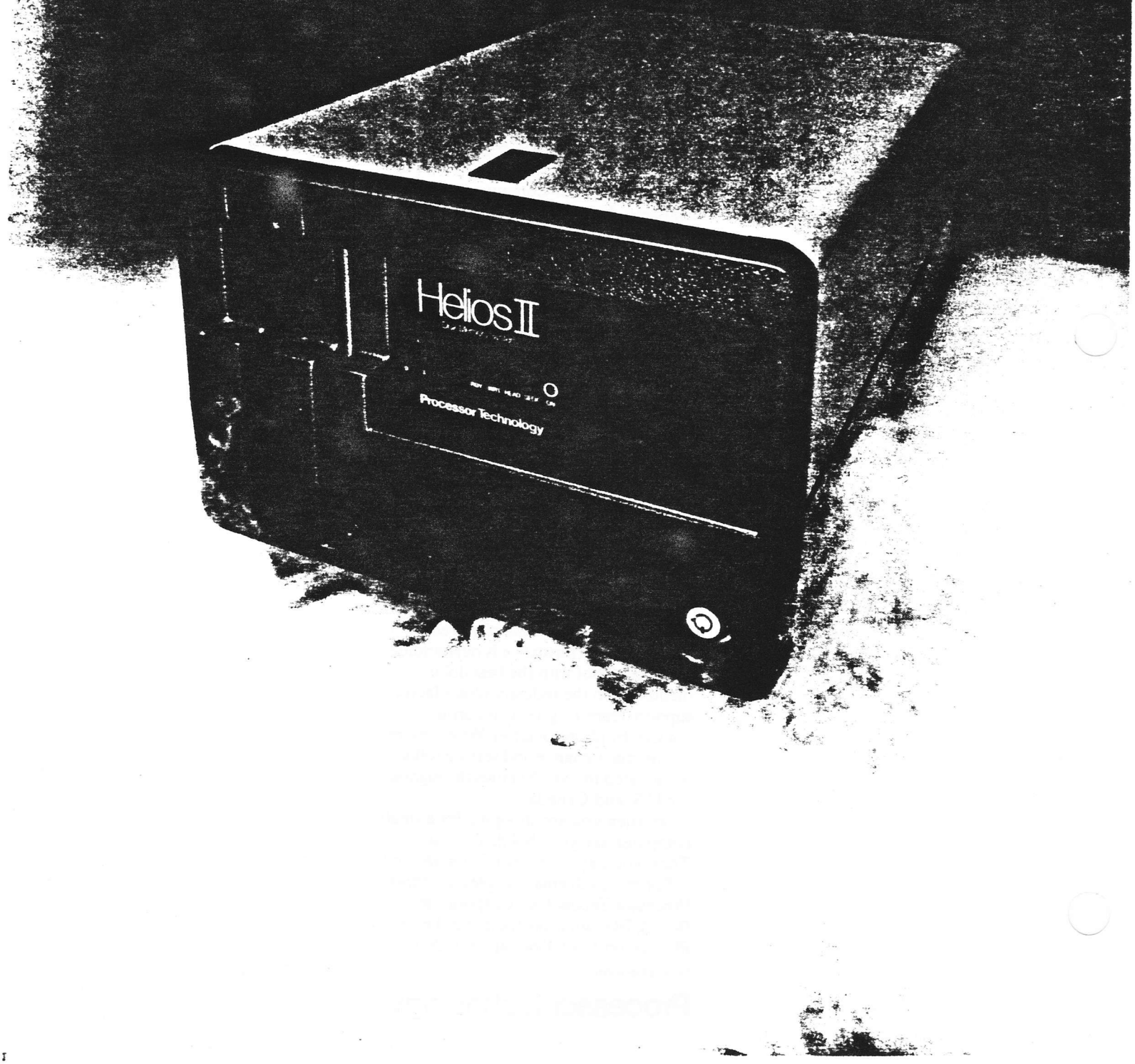
*available soon

Processor Technology

HELIOS II

Disk Memory System

1 EPT 1 - 64



When you want big system performance from your small computer, turn to Processor Technology's Helios II Disk Memory System. Helios II brings your small computer up to a level fully comparable and frequently superior to many mini-computer systems. Perhaps best of all, you can reach this point for thousands of dollars less than the typical mini-computer installation.

Today, Sol Systems using Helios II and PTDOS (Processor Technology Disk Operating System) are on line doing a variety of business, industrial, engineering, educational and scientific jobs.

A school system has replaced their time share terminals with our stand alone computer. A pharmacy system, specially configured for the Sol/Helios II combination, handles the complex and previously error prone task of prescription processing. Business firms use this powerful system for inventory control, invoicing and payroll. A small college stores the student-faculty directory, course requests, accounts receivable and student schedules on their system.

In the West, an advertising agency is using the system for media analysis while an astronomer plans to use the system to program astronomical observations.

A university reports the Sol System with Helios II provides superior performance for computer assisted and computer managed instruction.

Other applications are continuously under development by Sol/Helios II users everywhere.

Powerful software.

PTDOS 1.4 is a proven disk operating system with total file, device and memory management. Since the prototype was developed in 1975, PTDOS has undergone continuous expansion and refinement.

PTDOS increases the all-important throughput of your computer.

With the command interpreter as a convenient interface to the system, the user can type short commands which perform complex tasks. The powerful DO command provides a facility for editing and executing command procedure files.

Over 40 commands span from simple file management primitives to complex editors, assembler and other data management utilities.

PTDOS 1.4 System Commands

MAJOR SUBSYSTEMS	
EDIT	Video Display Text Editor
EDT3	Standard Text Editor
ASSM	8080 Assembler
DEBUG	8080 Program Debugging Aid
DO	Command Interpreter Macro Facility
DISK MAINTENANCE	
DISKCOPY	Disk to Disk Copy
RECOVER	Reclaim lost space on disk
INFORMATIVE	
FILES	List Information About Files
FREE?	List Remaining Free Space
SYST	List Configuration Information
OPEN?	List Names of Currently Open Files
FILE MANIPULATION COMMANDS	
COPY	Move Data Between Files
IMAGE	Save Memory Contents
BLDUTIL	Build Utility File
EXTRACT	Modify or List Image File Structure
KILL	Kill a file
RENAME	Change the name of a file
RETYPE	Change the type of a file
REATR	Change File Protection Attributes
DATA MANAGEMENT	
XREF	Assembly Language File Sort and Cross Reference
DUMP	Dump a file in Hex or ASCII
PRINT	Print ASCII contents of text file
RNUM	Renumber an Assembly Language File
SAVE	Create Archive Copy of Files
GET	Load SAVEd files or COPY from a Disk

FILE PRIMITIVES – LOW LEVEL
FILE MANIPULATION

CREATE	Create a new file
OPEN	Open a file
READ	Read a file into memory
WRITE	Write data into a file
CLOSE	Close a file
SPACE	Move File Cursor
RANDOM	Make a file randomly accessible
SEEK	Position the cursor of an indexed file
ENDF	End file at current cursor position

CONFIGURATION AND CONTROL COMMANDS

SETIN	Set CI input file
SETOUT	Set CI output file
SET	Set various miscellaneous parameters
EXEC	Begin execution at some address
ZIP	Set Memory
CONFIGR	Change System Configuration
OUT	Change Console Output Drive

LANGUAGES. PROCESSORS AND GAMES

BASC5	Small BASIC Interpreter
FOCAL	The FOCAL language
TREK80	A video Star Trek game

Languages.

Several high level languages are available. BASIC/5 and FOCAL come with the system. Low cost options include Extended Disk BASIC, Disk PILOT and a FORTRAN compiler extended to support all PTDOS operations.

PTDOS supports device-independent files.

Device files turn any peripheral, such as a printer or tape drive, into a "file." The device file then performs necessary manipulation of the device, for instance, reading a block of data or rewinding a tape mechanism. In this way the device can be accessed as a PTDOS file.

Easy interface at the assembly language level.

System calls provide an easy interface to PTDOS services from any assembly language program.

FORTRAN available March 1978. PILOT available mid-1978.

Memory management provides complete system management of static or dynamic buffers. Fully protected system memory area helps prevent inadvertent destruction of the system or managed data. For added flexibility, the user can manage his own file buffers as well as instruct the system to protect a specified block of memory.

The system utility call provides a powerful path to virtual memory with simple, fast program and data overlays.

Hardware interrupt speeds system operation and allows fast response to device requests for service.

Check these unique features.

Random indexed files offer direct, rapid positioning to any byte or block in a file, anywhere on the disk(s).

BOOTLOAD, a small bootstrap program, loads PTDOS in one simple command. BOOTLOAD, included with the system on a CUTS cassette, is also available as an optional Sol personality module.

A start-up file lets you run any sequence of commands automatically on a bootstrapped system.

Each individual diskette may be named and dated for easy identification.

A disk password with system lock secures your data.

As requirements change, the user can reconfigure the system.

Two drives...a must for any disk system.

The Helios II *dual drive* disk system uses inexpensive, widely available floppy disks for high speed data storage and retrieval. Two drives means you can have two floppy disks on line at all times. The Disk Operating System (PTDOS) resides on part of one diskette – the System Disk. A second diskette provides additional user space and also allows fast and easy production of backup copies to protect critical data.

Helios II uses a unique recording technique known as "firm" sectoring. Under optimum conditions each diskette can store 384,000 bytes of information. That's over 3/4 million bytes on line with each Helios dual

For a new dimension of small computer power, the Helios II Floppy Disk Drive and Processor Technology Disk Operating System. Software includes Extended BASIC, Assembler, PILOT, Extended FORTRAN and FOCAL.

drive. Firm sectoring also allows faster data access and variable block sizes.

Up to four dual drives (8 diskettes) can be accessed by the system so that over 3,000,000 bytes of data are on line.

Fast access.

Helios II is fast, with a typical access time of 33 thousandths of a second. Large block size allows programs and data to be transferred very quickly. A 4K block can be read from the disk to RAM computer memory in the proverbial blink of an eye.

Reliable performance.

Helios II was designed for system reliability using proven technology. No compromises are made nor chances taken with your data. Standard single sided diskettes store data using 77 tracks on each 8-inch diskette. Such data formatting has been used over the years for consistent, reliable storage. Helios II uses direct memory access (DMA) to asynchronously transfer data to and from memory. Data transfers are made at an effective rate of over 660,000 bytes per second. Data integrity is assured with a standard hardware CRC error test performed on each transfer to or from the diskette. In addition, an optional read-after-write verify mode can be selected to handle crucial data requirements. Three level, triple option error handling and trapping allows up to nine different ways to handle errors on every system call.

Helios II comes complete with dual drive, cabinet, controller, PTDOS System Disk, power supply, case, all necessary cables, and full systems documentation. A 12K program that tests and reports on every aspect on your disk system is included.

PTDOS Users Group.

HELIUM is an independent organization of HELIOS users. It publishes a newsletter with useful hints, systems support and updated PTDOS information. To join, write to HELIUM, c/o Ian Kettleborough, P.O. Box 9269, College Station, TX 77840.

HELIOS II SPECIFICATIONS

Electrical Power Requirements

117 Volts, 60 Hz AC
Controller: 7.25 VDC min.
8 VDC at 1600 ma typical
Formatter: 7.25 VDC min.
8 VDC at 600 ma typical
PerSci Model 270 Drive
+ 5 VDC 1.7 amps nominal
– 5 VDC 0.15 amps nominal
+24 VDC 1.0 amp nominal – seeking
0.2 amp nominal – not seeking

Operator Controls and Indicators

Locking power switch with key
Motorized disk eject switches – one for each drive
LED indicators: Unit, Ready, Write, Seek, Power

Physical Dimensions

Height	9.5 in	24.13 cm
Width	14 in	35.56 cm
Depth	21 in	53.34 cm
Weight	53 lbs	23.85 kg
Shipping weight	70 lbs	31.5 kg

Temperature Requirements

Operational	50 to 100°F	10 to 38°C
Non-operational	–20 to 120°F	–29 to 49°C

Head Positioning

Servo-controlled voice coil – 77 tracks

Head Positioning Times

Track to track, including settling time – 10 msec (maximum)
Track 0 to track 76, including settling – 100 msec (maximum)

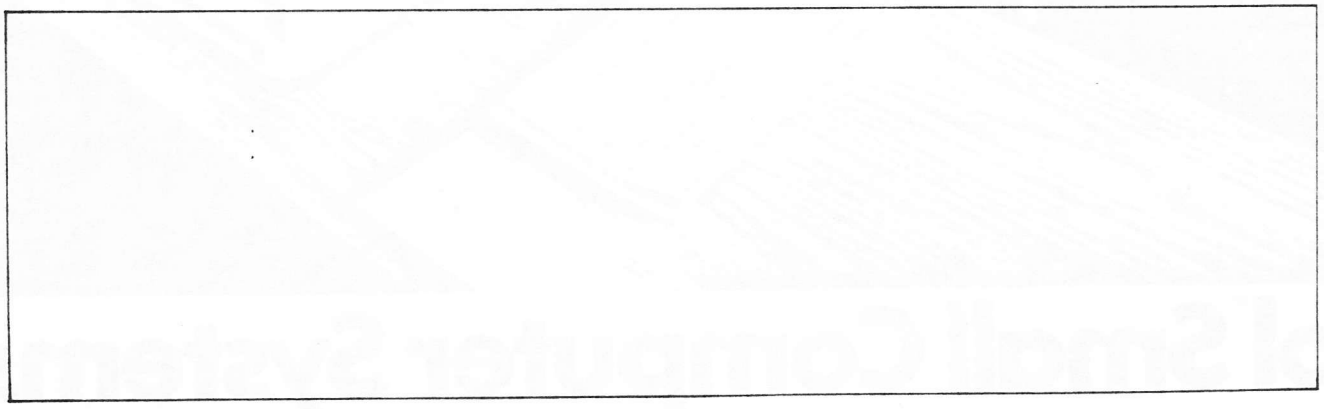
Miscellaneous

Cooling – forced air with filter
Direct-Coupled DC Spindle Motor – 360 RPM
Head Engage Time – less than 40 milliseconds
Read/Write Data Transfer Rate – 250 kilobits per second
Recording Capacity – Unformatted: 3.1 megabits per diskette
Actual data storage capacity: 384,000 bytes per diskette (maximum)

ProcessorTechnology

Processor Technology Corporation
7100 Johnson Industrial Drive
Pleasanton, CA 94566
(415) 829-2600

See it at your Processor Technology dealer



Copyright © 1978 2.78.20M

1 EPT 1 - 70

Sol® Small Computer Systems

ProcessorTechnology

Your key to effective, economical computer power.

1 EPT 1 - 71

If you ever wished to gain the force of a powerful general purpose computer operating with the simplicity and ease of ordinary office equipment, you can.

Now a scant year after the early 1977 introduction of the Sol* computer system, over 7,000 Sol's are on the job.

Their tasks are varied. Users all over the world are analyzing the stock market, tracking prescriptions, keeping the books, doing chemical, mathematical and physical research, controlling industrial processes, handling the payroll and profit and loss statements, correlating market research, analyzing product design and countless other tasks.

In a word, they are a working tool never before available to the average person or small business. They are saving time and earning money for lots of people like yourself. And they are doing all these things for a remarkably low investment!

Sol systems don't go out of date either. They were designed from the ground up to build on as newer memories, more powerful languages and more advanced peripheral equipment come along. As your needs for more computer force grow, no matter what Sol system you start with, you'll own a set-up beyond the dictates of the calendar. You'll know why you bought the best.

Processor Technology customers often tell us their original intent was to use a Sol small computer system at home or as a hobby. Sooner than you might believe, they were using the Sol system in their business, profession or education.

*Processor Technology Corporation

So by simply adding a disk memory system, the Helios II, a printer or other peripheral, perhaps putting one of several easy to learn languages to work, our users were solving routine, sometimes very complex, problems with amazing ease and speed. And solving them profitably.

Compare Sol to any other small computer

As the first totally integrated system in the small computer field, Sol has had imitators. None has matched Sol's high quality engineering and all around flexibility. No wonder it's the number one choice.

Notice the keyboard. Letters are positioned for standard touch typing. The key tops are contoured for comfort. When depressed, they respond with a resilient, springy touch. You can use the computer for hours without tiring. For extensive text editing, word processing, order entry, and similar continuous keyboard tasks, the Sol keyboard quality is unmatched.

Inside, the keyboard circuitry is non-contacting (capacitive). There are no contacts to wear out.

A sturdy metal cabinet protects Sol. A 200 pound person can stand on the top of it without damage. The wood sides are solid, hand-finished walnut.

Inside Sol, Processor Technology has taken some special pains. Engineers call the main motherboard "beautiful"—conservatively designed and rugged. No corners were cut.

Processor Technology paid particular attention to the video circuitry. Notice that the cursor block on the video screen (the spot that shows you where you are when entering text) surrounds the character completely leaving it still legible. All the characters on the screen are crisp and legible. They don't wobble or swim as you look at them. As you scroll the text up and down on the screen, the image moves smoothly without strobing or flashing. There's little eye fatigue from long time viewing.

Few wires are strung around on the main board. Connectors are mounted at the rear to plug in other electronic components, just like a hi-fi amplifier. No modifications are needed to plug in Teletype* machines, cassette recorders, tv monitors, telephone couplers (modems), and ASCII standard terminals (EIA & current loop). You can hook up a Sol computer system as easily as a home stereo system.

*Teletype Corp.

S-100 bus compatibility

Significantly, Sol systems are compatible with other S-100 bus products. The S-100 bus is a standard connecting configuration used by more than 100 manufacturers of small computer equipment. Many of these products can be easily plugged inside a Sol system when you need special hardware for voice input, control of electrical appliances elsewhere in the building, analog to digital conversion, or video graphic capability.

Another important feature unique among microprocessor systems is Sol's flexibility. It can switch in a moment from a stand alone computer to an intelligent terminal. Many people have access to large computers via telephone hookup. You can use the Sol as a computer to develop your program and reduce the data. Then switch to terminal mode to use it as a time-sharing terminal with a large computer.

Easy to use, the Sol is ideal for learning. Adults find hours of intellectual challenge and stimulating involvement. Kids enjoy the Sol. They start with games but quickly pick up the elements of programming. Sol's rugged construction is kidproof so you don't have to worry.

All Sol systems come with excellent, understandable documentation. Schematics and instructions are geared to learning. You don't have to be an electronic engineer to understand them. Even if you purchase a Sol fully assembled and factory tested, Processor Technology's instruction manuals will help keep your Sol up and running. Sol is easier to service. You'll have less down time and lower maintenance costs.

Factory Guarantee

Sol systems are backed by a factory warranty. Processor Technology guarantees assembled and tested products for one year (parts and labor). Kit products have a three month guarantee on parts. Any authorized Processor Technology dealer can service Sol systems, and many offer maintenance contracts.

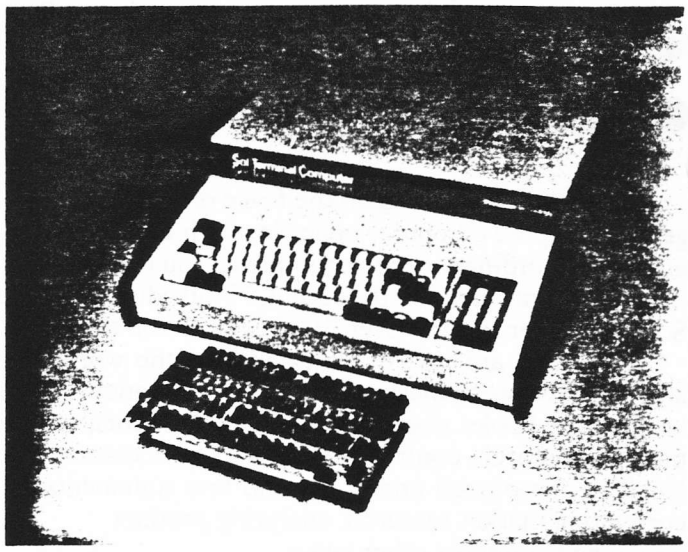
Sol's flexible

Change operating software quickly and easily. The operating system is preprogrammed on ROM (Read Only Memory) chips located on a small personality module inside. You can change the Sol's "personality" in one minute. The transformation requires no technical skill at all.

There are now two personality modules. SOLOS gears the computer for stand alone operation. It automatically readies the computer for your commands as soon as the Sol is turned on and controls the cassette, keyboard and video operations. SOLOS has terminal mode functions to switch the computer between stand alone and terminal operation.

BOOTLOAD personality module works with our Helios II Disk Memory System. We've replaced the terminal mode functions with a special loading program for the Disk Operating System. In other respects it is identical to SOLOS.

Additional personality modules are in development.



Sol small computer systems with the add on power of $40,000 mini's.

The Sol-20/16 Terminal Computer

The Sol-20/16, our basic computer, features the SOLOS personality module, 16,384 bytes of memory and BASIC/5 cassette plus a complete 300 page manual. It's available either fully assembled and factory tested or as a kit. A single, low power module carries the 16K dynamic memory.

BASIC/5 is a small version of the very versatile BASIC language. It is designed for applications requiring primarily mathematical manipulation without extensive processing of text. BASIC/5 is the perfect language for an introduction to computer programming because it's easy to learn and requires only a small amount of memory storage. Hundreds of published programs already written in BASIC work with Processor Technology's BASIC/5.

The 300 page Sol manual carries the most complete set of instructions, theory of operation, software information, drawings and operating information offered by any small computer manufacturer. And after you get it, we send you periodic updates. Many of our users also file their copies of ACCESS, the Sol users journal, in this handsome sturdy 3 ring binder.

You will need to add some device for storage and retrieval of data and programs. A standard cassette recorder will do this. Plug it directly into the back of the Sol. You will also need some means of visually displaying data — a tv monitor, teletype or similar device. These also plug into the back without special adaptors or modifications.

The Sol-20/16 offers the option of using equipment you have or selecting peripherals of your choice. It's an excellent low cost starting system.

Here are a few of the features of the Sol-20/16.

- 8080A microprocessor, a sophisticated computer-on-a-chip and the "brains" of the Sol.
- 1024 character video circuitry which displays 16 lines of 64 characters on a video monitor or suitably adapted tv.
- a custom designed, comfortable to use 85-key capacitive (non-contacting) keyboard with cursor keys and arithmetic keypad.
- an audio cassette interface capable of controlling two recorders (one read, one write) at 1200 bits per second. Store and retrieve large amounts of data on low cost, standard cassettes.
- parallel and serial standard interfaces with connectors permit a teletype and many other devices to plug directly into the back of a Sol.
- a complete rugged power supply and quiet cooling fan.
- a handsome case of walnut and metal.
- 16,384 byte dynamic RAM module.
- software on cassette with BASIC/5 language and two sophisticated computer games.
- a back plane capable of accepting five expansion modules.

Sol-20/32 Terminal Computer

Sol-20/32 offers all the features of the Sol-20/16 with twice as much memory. A single low power module carries 32,768 bytes of dynamic random access memory. Four slots remain open in the back plane for hardware additions.

Fully configured Sol Systems.

Presently four fully integrated Sol Systems with all the hardware, peripheral equipment, documentation and software are available. Depending on the complexity of your tasks, one will give you the kind of computer power you need now and still let you add more later.

All offer the best price/performance value on the market today. All integrated systems come with Processor Technology's new Extended BASIC. This new language is one of the fastest and certainly the most complete for the personal small computer on the market today.

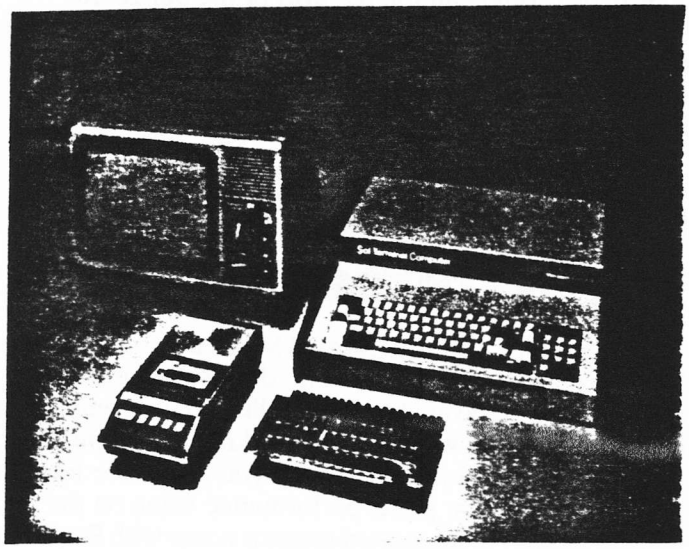
Extended BASIC

Extended Cassette BASIC features string and advanced file functions, special screen commands, timed input, complete matrix, logarithmic and trigonometric functions, exponential numbers, 8 digit precision, and advanced mathematical functions including log, natural log, sine, cosine, tangent, arc tangent and square root. Using this BASIC, you can do all of the mathematics needed to solve 5 equations with 5 unknowns in a matter of 4 seconds using just 2 lines of code.

Extended Cassette BASIC handles serial access files. It has provisions for tape rewind. Cursor control gives you some graphic abilities.

Extended Disk BASIC has all the number crunching talents of Extended Cassette BASIC plus almost instant access to data and programs on floppy disks. It has random as well as sequential files and a unique ability to update sequential data in place. It takes advantage of the powerful Processor Technology Disk Operating System and Helios II.

In sum, when you purchase a Sol system, you get real computer versatility. You can do the general ledger, the payroll, or engineering and scientific problems. You can display and print out the results. You can edit text. You can use it for computer aided instruction. You have computer power, not computer-like power.



Sol System I-A

Sol System I-A, offered in assembled form only, includes the Sol-20/16 Terminal Computer with 16,384-byte memory and SOLOS module plus PT-872 video monitor, RQ-413A cassette recorder, Extended Cassette BASIC, cables and the Sol system manual.

The PT-872 is a specially modified 11" Panasonic black and white TV set. It is switchable from the video display mode to the standard UHF/VHF television mode. Fully isolated for safety, this unit provides extremely crisp and stable displays. The cassette recorder is a standard Panasonic unit.

Sol System II-A

Sol System II-A is exactly the same as Sol System I-A except for a larger memory, 32,768 bytes (eight bit words). The memory is contained on a single dynamic RAM module, using only one slot of the back plane. Sold fully assembled and tested, it includes the Sol-20/32 Terminal Computer with 32K of memory and SOLOS personality module plus PT-872 video monitor, RQ-413A cassette recorder, Extended Cassette BASIC, cables and Sol System manual.

For serious business, you need a disk.

Cassettes are a convenient low cost way to store programs and data, but it takes time to wind and rewind the tape. For most business uses, the fast retrieval time of a disk is the way to go. With a Helios II Disk Memory System you can bring your Sol System up to a level fully comparable and frequently superior to many mini-computer systems. Helios II is fully supported by the most advanced disk operating system in the small computer field. It is a system under development since 1975 designed to make full use of the up to 1.5 million bytes of memory that are available to the user.

PTDOS offers total file, device and memory management. Add to, delete or edit files. List, copy or combine the files on any disk. Over 40 commands span from simple file management primitives to complex editors, assembler and other data management utilities.

Sol System III

Here's the system that stands up to the mini's. It includes the Sol-20 with 49,152 bytes of memory and BOOTLOAD personality module, Helios II Model 2 Disk System with Extended Disk BASIC, PT-872 video monitor, cables and documentation.

Sol System III comes assembled, burned-in and tested. Inclusion of Helios II brings this system up to a level fully comparable and frequently superior to many mini-computer systems. And, you reach this operational level for thousands of dollars less than typical mini-computer installations. A separate brochure on the Helios II and PTDOS is available on request.

Newest member of the Sol family, Sol System IV

Sol System IV consists of the Sol-20 with 49,152 bytes of memory and BOOTLOAD personality module, Helios II Model 4 Disk System with two dual drives, Extended Disk BASIC, PT-872 video monitor, cables and Sol System manual.

It comes completely assembled, tested and burned-in. Except for the larger data storage capacity, 1.5 million bytes, it is like Sol System III. Obviously with so much memory available the scope of problems and the amount of data that can be manipulated become much greater. Suitable for larger business applications.

Software.

1 EPT 1 - 75

Languages on both cassette and disk include BASIC/5. Extended BASIC. FOCAL.* FORTRAN* and PILOT.* ALS-8 is an "assembler" for program development which includes text editing and 8080 simulation. It is available either on cassette or on preprogrammed ROM (Read Only Memory) chips which mount on the Sol General Purpose Memory module. A variety of games and other special programs are also available.

Using one of these languages, you can program your Sol to perform a wide variety of tasks. Thousands of special programs have already been written in these languages. You will find many of them in readily available books and periodicals.

Processor Technology has devoted more effort to the development of software than any other small computer manufacturer. We are the first small computer company to offer a fully implemented disk operating system: PTDOS with over 40 major commands, several languages and numerous utilities. New programs are in development.

FOCAL

FOCAL is a math language originally written for the PDP-8 mini-computer. Many thousands of FOCAL programs exist and now can run in Sol. 8080 FOCAL includes "operator precedence" and all other standard FOCAL conventions. FOCAL is available on CUTS SOLOS compatible cassette. It is also included on the PTDOS system disk that comes with Sol Systems III and IV.

DEC

FORTRAN

FORTRAN is a high level language which includes the following functions:

- Explicit run time error comments during compilation and at run time
- Eight significant digits of precision
- String manipulation
- Cursor plot function
- Hexadecimal constants
- Direct in-line 8080 assembly language mnemonics accepted by the compiler.

Processor Technology FORTRAN is identical to FORTRAN IV except for the extensions and lack of COMMON and Double Precision statements. It is available on CUTS cassette or disk. The disk version interfaces well with PTDOS.

PILOT

Processor Technology PILOT, available in mid 1978, is offered on both disk and cassette.

PILOT is a string-oriented interactive language well suited to the applications of educators, educational institutions and psychologists. It is excellent for testing and computer aided instruction.

Custom tailored for Processor Technology by the creator Dr. John Starkweather of the University of California, PILOT runs on the Sol using the SOLOS I O with direct screen cursor positioning, program and data files.

And for fun

8080 CHESS:* Here's a wonderful way to improve your chess skills. Operating at a healthy level 5, the 8080 chess program is the finest available for microprocessor computers. Under development for less than a year, the program has already proved its mettle in competition against the truly big computers.

Offered in cassette form, and accompanied by a complete instruction manual, it gives education and delight to both the novice and the expert player.

TREK 80. Based on the NBC television series STARTREK, this machine language program uses 8K of memory for real time war with the Klingons. No holds barred, they're out to get you from each of 100 quadrants. You can warp through hyperspace, fire phasers, photon torpedos, or experimental rays. If you just can't go on, self-destruct.

GAMEPAC 1. Show off your Sol system with this line up of video games.

MATH PACK. Processor Technology floating Math Pack is a self contained five function calculator designed to run on any system using the 8080 code. The following functions are offered: addition, subtraction, multiplication, division and square root.

All Processor Technology software is distributed on an individual sale basis for personal use. No license to copy, duplicate or sell is granted with this sale. Each software package has been copyrighted.

*available soon.

See your Sol dealer for a demonstration.

Sol stores are located conveniently throughout the U.S. and Canada. Your Sol dealer has been carefully selected on the basis of his computer knowledge and ability to meet your needs. Not every store carries the Sol line. The dealers who do are the best in the world.

Sol Specifications

- Keyboard: 85 key upper/lower case with separate numeric keypad. Upper-case shift, shift-lock, cursor control and repeat keys provided. System reset performed by simultaneous depression of control keys. Indicator lights (LED) for local, upper case and shift.
- Character set: 96 printable ASCII upper and lower case characters plus 32 optionally displayable control characters.
- Cursor: Switch-selectable blinking. Block video inversion. Program controlled positioning standard. Cursors may exist at any or all character locations.
- CPU: 8080A. Uses same machine language as other 8080 systems. 2 MHz clock cycle time. 78 instructions.
- Cassette interface: 120 character per second CUTS format or 30 character per second Kansas City format, selected by software. Uses audio cassette function of microphone start-stop switches. AGC for level insensitivity. Phase-locked data recovery tracks with speed variations. Software performs CRC data integrity check each 256 characters.
- Serial interface: RS-232 and 20 mA current loop. 75 to 9600 baud, asynchronous. 25 pin female "D-type" connector on card.
- Parallel interface: Eight data bits for input and output: output bus is tristate for bidirectional interfaces: levels are standard TTL. 25 pin male "D-type" connector on card.
- External memory: Expandable to 65,536 bytes total ROM, PROM, and RAM. Uses S-100 standard modules.
- Video signal output: 1.0 to 2.5 volts peak-to-peak. Nominal bandwidth is 7 MHz.
- Power requirements: 117 volts AC 60 Hertz, 250 Watts. 220 Volt 50 Hertz option available.

Sol SMALL COMPUTER SYSTEMS

	Sol-20/16	Sol-20/32	Sol System I-A	Sol System II-A	Sol System III	Sol System IV
Mainframe	Sol-20	Sol-20	Sol-20	Sol-20	Sol-20	Sol-20
Personality Module (Operating System)	SOLOS	SOLOS	SOLOS	SOLOS	BOOTLOAD & PTDOS	BOOTLOAD & PTDOS
Random Access Memory (RAM) (8-bit words)	16K bytes	32K bytes	16K bytes	32K bytes	48K bytes	48K bytes
Language Included	BASIC/5	BASIC/5	Extended Cassette BASIC	Extended Cassette BASIC	Extended Disk BASIC FOCAL BASIC 5	Extended Disk BASIC FOCAL BASIC 5
Video Monitor	—	—	PT-872	PT-872	PT-872	PT-872
Cassette Recorder	—	—	RQ-413A	RQ-413A	—	—
Disk Memory System	—	—	—	—	Helios II Model 2	Helios II Model 4
Available Disk Memory (8-bit words)	—	—	—	—	750K bytes	1500K bytes
Number of Disks On-line	—	—	—	—	2	4

Proces Technology
Processor Technology Corporation
7100 Johnson Industrial Drive
Pleasanton, CA 94566

Bulk Rate
U.S. Postage
PAID
Permit #54
Pleasanton,
CA 94566

1 EPT 1 - 77

Sol Systems: Why they are first choice

THE PERSONAL COMPUTER RETAILER

Vol.2 No.3 Published by Processor Technology Corporation May 3, 1978
7100 Johnson Industrial Drive, Pleasanton, CA 94566

Teachers meet Sol in San Diego

Elementary and high school teachers are generally familiar with small computers but do not yet realize their potential as teaching aids.

Our marketing staff had the opportunity to learn more about this virtually untapped market at the National Council of Teachers of Mathematics annual convention in San Diego on April 13, 14 and 15. Processor Technology was the only small computer manufacturer represented.

More than 7000 teachers and administrators were on hand to look over what was new. The teachers were especially enthused by our demonstration of a computer-aided instruction program written in PILOT.

PILOT USEFUL AND EASY TO PROGRAM

Brett Bullington, dealer sales rep, commented, "PILOT is very useful for teachers because it is text- and string-oriented, yet versatile and easy to program. You don't have to be a programmer to use this language effectively."

Based on teachers' questions and responses, Ralph Palsson, also of our marketing staff, found that most instructors were interested in using small computers as aids in teaching mathematics and science. There was also interest in teaching programming to junior and senior high school students.

Both Ralph and Brett distributed dealer lists and referred many teachers to local Processor Technology dealers.

"It's usually inexpensive (around $60) for a dealer to rent a demonstration booth at a regional educators' conference. The local market exposure is great," says Ralph. "Once you get your name on the mailing lists of local school districts and community colleges, you will receive information about upcoming conventions and trade shows. You can then make your reservations early.

In this issue...

Sit back, relax...let your phone visit the typesetter for you.....story page 4

You want it when?!.....page 2

TV ads work for Maryland dealer.....page 3

Tips for troubleshooting the 16KRA.....page 5

Copyright © 1978, by Processor Technology Corporation. All rights reserved.

You want it when?!

NEW PRODUCT UPDATES

Item	Begin Shipping	Change
HARDWARE		
ParaSol Debugger	week of April 24	shipment has begun
HyType I	week of May 26	on schedule
HyType II	week of May 8	moved back one week
SOFTWARE		
8080 FOCAL	within the next 30 days	

SPARE PARTS

Spare Parts Kit (#900050). These kits are in stock and available for immediate shipment.

The kit contains a sampling of the various components, particularly integrated circuits, used in our product line. Dealer price is $400.

April FORTUNE, BYTE picture Sol

Both Fortune and Byte magazines picture the Sol's versatile personality in their April issues.

"Putting the Microcomputer to Work" (Fortune, April 24, 1978, page 54) features the Sol in use by "Big Brother Biasiotto" at the Franciscan Friars Provincial Headquarters in New York City.

On page 126 of the April '78 issue of Byte, the Sol is pictured enjoying life in a Victorian parlor. The accompanying article, "User's Report: The Sol-20", records the author's favorable impressions of the Sol.

Whodunnit

Editor.....Dottie Wanat

Graphics.....Lois Esser
Ray Oleson

General Input.....Rick Meyer
Chuck Rosas
Elizabeth Fairchild
Brett Bullington
Loren Willey
Ralph Palsson

TV advertising pays off for Maryland dealer

by Elizabeth Fairchild

"Dollar for dollar, television advertising has paid off for our store more than any other media," says Dave Gardner, partner and manager of COMPUTERS, ETC...in Towson, Maryland.

Undaunted by the high cost of video production and TV time, Dave experimented with producing his own ads. He kept costs down by writing the copy himself and selecting an inexpensive though effective format. Total production cost for two 30-second spots was only $230.

In each ad, a written message "crawls" across the screen one letter at a time at an average reader's pace. A voice in the background describes current sale items and special services offered by COMPUTERS, ETC... The description frequently ties in with the store's current newspaper ads. Sometimes business applications are emphasized, sometimes hobby uses. This format appeals to potential computer buyers without appearing obviously "low-budget."

The spots have been running on a Washington D.C. independent station in bursts Thursday through Sunday, one week per month. Dave spends approximately $1400 per weekend. He selects programs which appeal to his target audience: men ages 18-45 with medium incomes. To determine which shows are appropriate, he studies the rating guide provided by the station. So far he has used sports specials, a program about the movie GREASE, the Odd Couple, Perry Mason, and late movies.

The response has been so good, Dave is planning to expand to a UHF station which broadcasts to both Washington and Baltimore. (COMPUTERS, ETC... has a second store in the Baltimore area.)

Summing up his experience with media so far, Dave says, "I would rank TV #1, newspapers #2, and radio a poor third. Radio has only worked for us at Christmas time to advertise video games."

Low-hassle warranty cards coming soon

Beginning soon new warranty registration cards will accompany all hardware manuals. These cards are stamped and self-addressed--no envelopes, postage, glue, staples or tape is required. The customer can simply fill it, fold it, seal it, and send it to Processor Technology.

We require a separate card for each product purchased. For example, if a customer purchases a Sol System III, he should send us warranty registration cards for the Sol, the Helios, and each 16KRA board.

Please pass this information along to your customers.

SEE YOU AT NCC

IN JUNE.

How we typeset the new 16KRA and FOCAL manuals with a little help from Sol

by Rick Meyer

A Sol System III, assisted by special electronic word processing and typesetting equipment produced the new 16KRA and FOCAL manuals which you will receive soon.

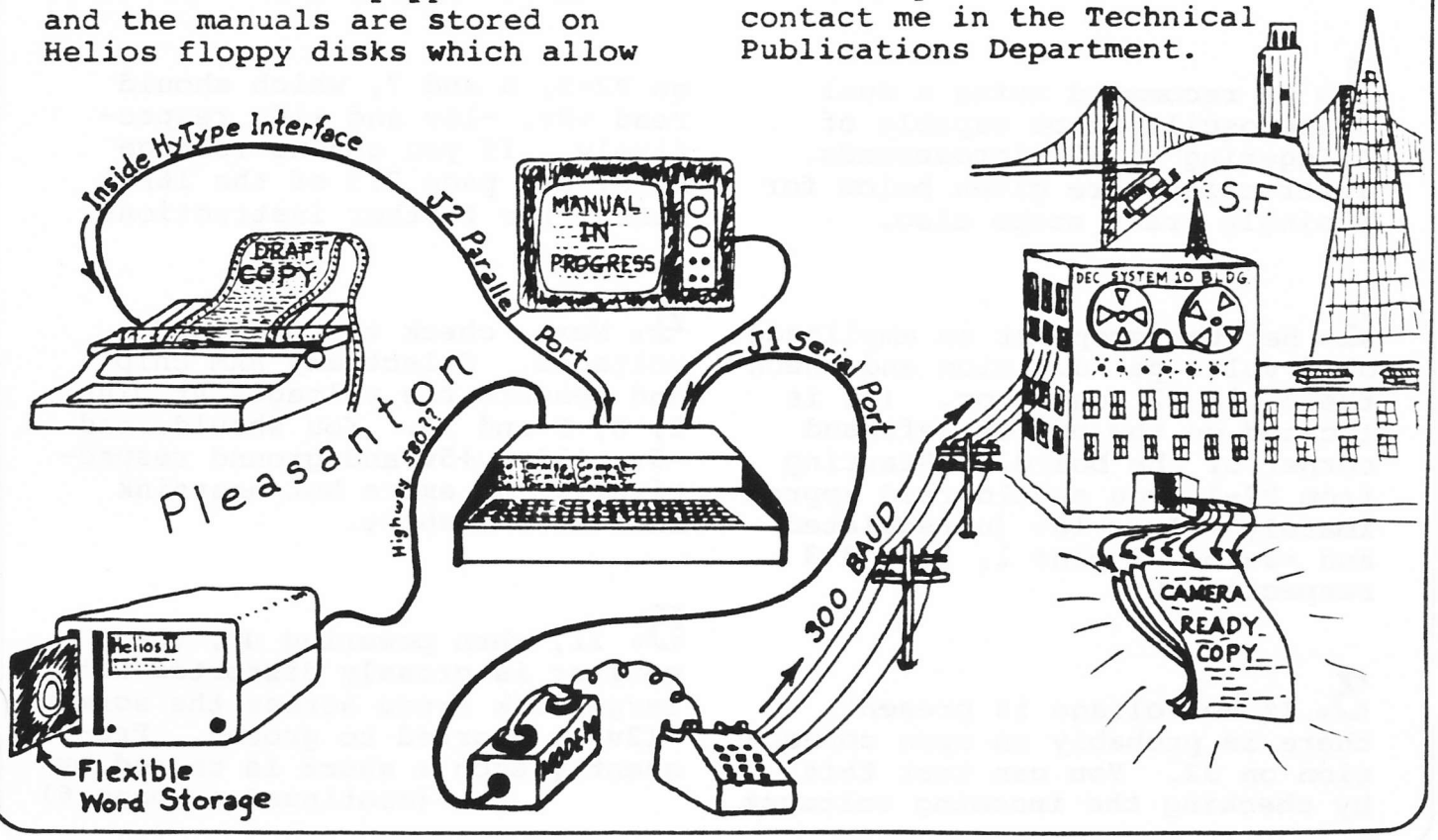
This small computer application allows a small staff in our publications department to produce more and better quality work. The new manuals are easier to read, more compact, and more professional in appearance. I'm quite excited by the results and believe you will find the procedure of interest.

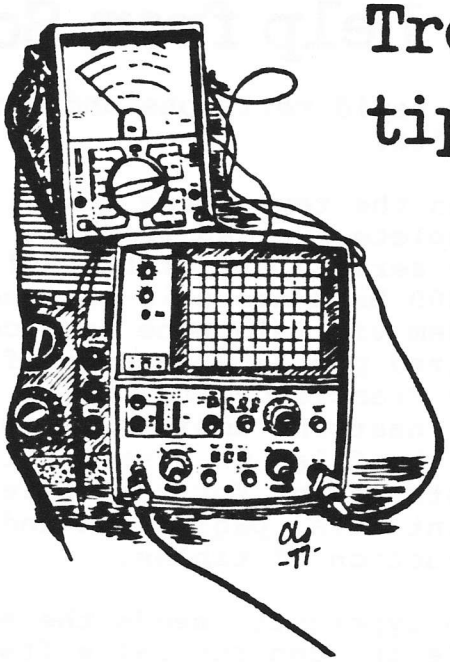
It begins when one of our technical writers enters his or her original draft into a Sol System III using a special word processing editor. The new Sol-HyType Interface prints the intermediate drafts on a HyType Printer and the manuals are stored on Helios floppy disks which allow for rapid revisions and additional drafts.

When the text for a manual is complete, we can transmit it out the serial port of the Sol through a 300 baud acoustic-coupled modem via telephone to a computerized photo-typesetting firm in San Francisco forty miles away. By inserting additional commands in the file, we give detailed instructions for type face and point size, pagination and construction of tables.

The typesetter sends the edited file through special software which drives the photo-typesetting machine and produces the camera-ready pages. In just a few days the copy is mailed to us ready for printing.

If you have any questions about the system, or would like further details, please feel free to contact me in the Technical Publications Department.





Troubleshooting tips for 16KRA

Excedrin headache number 16

When a 16KRA has a serious headache and you want a systematic treatment to give it fast, fast relief (before you get a headache), try this approach:

Check voltages

1. We recommend using a dual trace oscilloscope capable of triggering at .2 microseconds. Instructions are given below for a single trace scope also.

2. Set the scope at an amplitude of 5 volts per division and check the 7 pin J2 connector. (J2 is located on the bottom lefthand corner of the board.) Starting from P2-1, you should read approximately +16v, -16v (unregulated) and +8v on P2 pins 1, 2, and 3 respectively.

3. If no voltage is present, there is probably an open connection on J2. You can test this by checking the incoming voltages on P2-5, 6 and 7, which should read +8v, -16v and +16v respectively. If you get no voltage here, see page 7.3 of the 16KRA manual for further instructions.

4. Next, check the on-board voltages. Select any RAM chip and measure the voltages at pins 1, 8, 9 and 16. You should read -5v, +12v, +5v and ground respectively. An extra hot heatsink indicates a short.

5. If, when powering up, the monitor is grossly distorted with large dark areas across the screen, +12v is shorted to ground. Frequently such a short is caused by

(continued on page 6)

Troubleshooting tips...

(continued from page 5)
the socket pins of U29, 30 and 31 slicing into the insulation of the Rev J jumper and shorting either +12v or +5v to ground. Simply bend this insulation away from the pin. Please DO NOT use a soldering iron because this voids the warranty. If you still cannot locate the short, send the board back to the factory for repair.

Check delay lines

1. The delay line in socket U71 is vital to the board's operation. A failure (whether hard or intermitent) is often due to the socket itself. Often a simple dump command will clue you into the problem. Type in the command "DU Ø EF <cr>." If a dump does not occur, the delay line is a likely source of trouble.

2. To troubleshoot U71, temporarily remove the D to DN jumper in area D. The board is now in an undefined state; this condition is sufficient for testing the delay line. Another approach is to put the board into a wait state by grounding U68-11. In the wait state, pin 72 of the S-100 bus should be at ground potential.

3. The delay line determines the pulse widths of MC and RC. If you are using a single trace scope, check the signal at U61-8. Whenever it goes high or low, the delay line should follow it. A floating signal (+2 volts) at U57-5, 2, 4 or 1 indicates a bad delay line. The pulse out of the delay line should be a crisp 350 nsec long. Any variation indicates a defective delay line.

4. With a dual trace scope, set the sweep to 100 nsec per division. U61-8 should have a stable pulse of 300 to 350 nsec. Due to the output of NAND gate U57-6, there will be a 100 nsec delay relationship between U61-8 and U57-5.

Trigger on U61-8 to observe the timing relationships. The result should be:

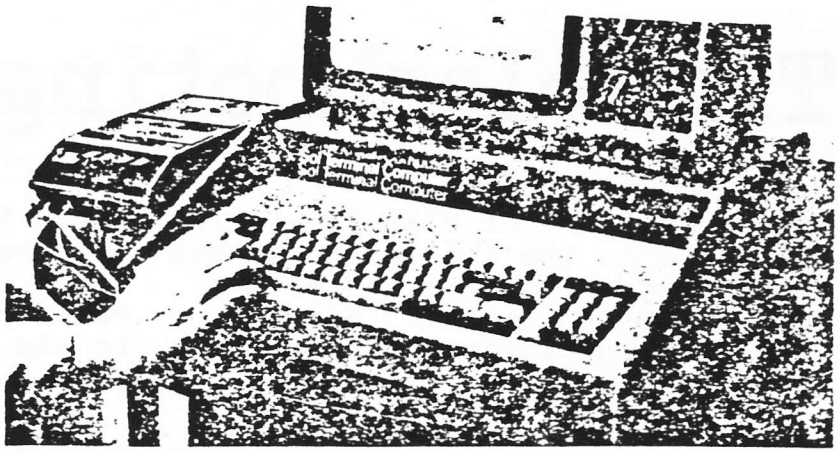
a. a 50 nsec delay between U57-5 and U57-2.
b. a 100 nsec delay between U57-2 and U57-4.
c. a 100 nsec delay between U57-4 and U57-1 with a pulse width of 300 to 350 nsec.

More tips later

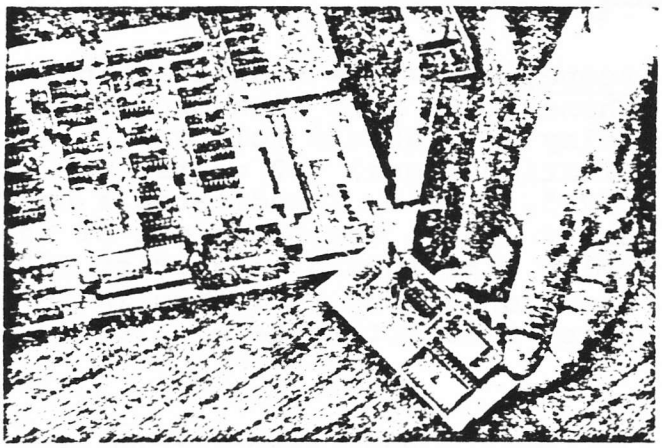
Another set of problems can occur on the address lines. I'll discuss these in a future article. 'Til then, take two aspirin and get plenty of rest.

--Chuck Rosas

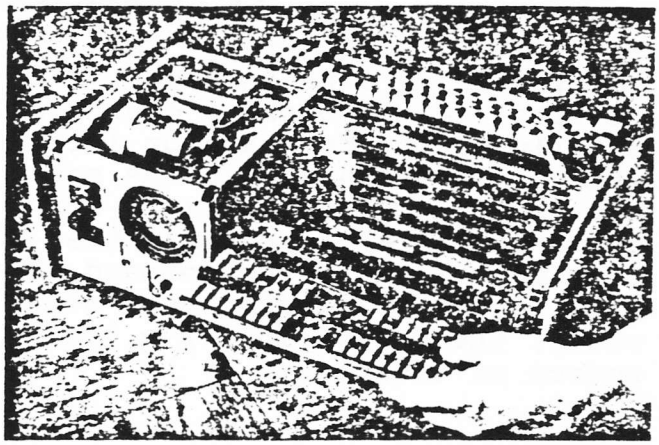
Build an all-in-one home computer



Arithmetic keypad (far right) on Sol 20 microcomputer simplifies entry of long data lists. For clear readouts, TV atop computer is modified to accept video signals.



Plug-in module carries the system monitor, a microcircuit with thousands of permanently stored instructions that regulate basic operations of the microcomputer.



Memory board slipped into Sol 20's five-slot backplane stores 16,384 bytes (16K of computer words). A quiet fan cools the power supply (left) and other circuit boards.

By JOHN FREE

Home-computer kits can be a hassle if you discover that several "optional" plug-in boards and accessories are needed to make your computer perform adequately. I built Processor Technology Corp.'s (PTC) Sol 20 Terminal Computer because video, memory, computing, input, and output circuits are all included. PTC (Pleasanton, Calif.) markets its computer separately or in complete systems: computer, 11-inch TV modified to accept non-RF video signals, plus a cassette recorder for loading and storing programs.

The Sol 20 is called a *terminal* computer because it has a built-in keyboard and functions that could enable you to communicate with other computers. PTC's computer uses the popular 8080 microprocessor, which means there are many programs around that you can use. It also accepts S-100 boards [PS, Nov. '77] that can play music, talk, listen, and do other amazing things. The computer has a trim, low silhouette, however, and room for only five boards.

I spent about 58 hours putting a Sol 20 together, and made only one blunder: A bridge rectifier installed backwards in the power supply shorted several components. Lesson learned: Since the manual has few step-by-step drawings, extra-careful reading is a must.

The computer displays 16 lines of 64 black characters, or white readouts on a black background. Compared to displays I've seen from some microcomputers, the system's Panasonic TV gives an exceptionally clear readout. Data for upper- and lower-case characters are stored in permanent memory.

The Basic 5 computer-language cassette included in the kit can run many programs listed in magazines and other sources. Basic 5 works with the 16K memory board included with the $1850 Sol 20-16 kit. A "brainer" $45 extended-Basic cassette requires either 15K or 12K of memory, depending on whether you load matrix capability or not. With extended Basic and more memory you can tackle virtually all Basic programs. PTC's $2495 System I-A (Sol 20, cassette recorder, TV extended Basic, 16K memory) is now sold only in assembled form. PS

Master circuit board for Sol 20 kit covers over half the chassis. Keyboard section and 100-pin memory boards are assembled and tested at the factory.

ProcessorTechnology

Sol® SYSTEM SUGGESTED U.S. RETAIL PRICE LIST (July 1, 1978)

Sol SYSTEMS		Order No.	Price ($US)
1. **Sol-20/16 Terminal Computer** with SOLOS personality module and 17,408 bytes of memory, includes BASIC/5 cassette and Sol System manual	Factory Assembled/Tested Sol Kit/16KRA A & T	400500 400502	$2095 $1850
2. **Sol20/32 Terminal Computer** with SOLOS personality module and 33,792 bytes of memory, includes BASIC/5 cassette and Sol System manual	Factory Assembled/Tested Sol Kit/32KRA A & T	400600 400601	$2395 $2150
3. **Sol System I-A** includes Sol-20/16 with 17,408 byte memory and SOLOS module, PT-872 video monitor, RQ-413A cassette recorder, Extended Cassette BASIC, cables and Sol System manual	Factory Assembled/Tested	400700	$2495
4. **Sol System II-A** includes Sol-20/32 with 33,792 byte memory and SOLOS module, PT-872 video monitor, RQ-413A cassette recorder, Extended Cassette BASIC, cables and Sol System manual	Factory Assembled/Tested	400800	$2795
5. **Sol System III** includes Sol-20 with 50,176 bytes of memory and BOOTLOAD module, Helios II Model 2 Disk System (stores 769,000 bytes) with Extended Disk BASIC, PT-872 video monitor, cables and Sol System manual	Factory Assembled/Tested	400300	$5995
6. **Sol System IV** includes Sol-20 with 50,176 bytes of memory and BOOTLOAD personality module, Helios II Model 4 Disk System (two dual drives store 1.5 million bytes) with Extended Disk BASIC, PT-872 video monitor, cables and Sol System manual Note: All Sol-20 series units and Sol Systems are available for 220 Volt, 50 Hz operation. Contact factory for pricing and delivery information.	Factory Assembled/Tested	400340	$7995

Sol COMPONENTS			
1. **Sol-PC Single Board Computer** with SOLOS personality module	Factory Assembled/Tested Kit	101036-01 101036-02	$745 $575
2. **BOOTLOAD Personality Module** with 2048 byte PROM stored program loads PTDOS directly into Sol main memory. (BOOTLOAD is included with Sol System III or IV.)	Factory Assembled/Tested	107015	$100
3. **Sol-Hytype I* Printer Interface** includes assembled and tested interface board, all cables and mounting hardware. Plugs directly from any Diablo* Series 1200 printer into Sol Parallel Data Interface connector. No modifications to Sol are required. Driving software included.	Installation Package	900023	$150
4. **Sol-HyType II* Printer Interface** like Sol-HyType I above for any Diablo Series 1300 printer. Driving software included.	Installation Package	900024	$150

MASS STORAGE SYSTEMS			
1. **Helios II, Model 2 Disk System** includes dual PerSci 270 floppy disk drive, cabinet, fan, S-100 bus compatible controller, power supply, and system diskette with complete PTDOS software and documentation. (Stores 769,000 bytes.)	Factory Assembled/Tested	300000	$2995
2. **Helios II, Model 4 Disk System** includes two dual PerSci 270 floppy disk drives, cabinet, fan, S-100 bus compatible controller, power supply, and system diskette with complete PTDOS software and documentation. (Stores over 1.5 million bytes.)	Factory Assembled/Tested	304000	$4995

S-100 BUS SUBSYSTEMS			
1. **Subsystem B110** includes 8KRA memory, VDM-1 Video Display Module, 3P + S Parallel/Serial I/O Module, CUTS Computer Users Tape System cassette interface and GPM General Purpose Memory and Extended Cassette BASIC	Factory Assembled/Tested Kit	406100 406101	$1095 $799
2. **Subsystem B190** includes 16KRA memory, VDM-1 Video Display Module, 3P + S Parallel/Serial I/O Module, CUTS Computer Users Tape System cassette interface and GPM General Purpose Memory and Extended Cassette BASIC	Factory Assembled/Tested Kit (16KRA A & T)	406200 406201	$1195 $995
3. **Subsystem B350** includes 32KRA memory, VDM-1 Video Display Module, 3P + S Parallel/Serial I/O Module, CUTS Computer Users Tape System cassette interface and GPM General Purpose Memory and Extended Cassette BASIC	Factory Assembled/Tested Kit (32KRA A & T)	406300 406301	$1495 $1295

*Hytype I, Hytype II and Diablo are registered trademarks of the Xerox Corporation.

ProcessorTechnology 1 EPT 1 - 86

Sol® SYSTEM SUGGESTED U.S. RETAIL PRICE LIST (July 1, 1978)

S-100 BUS COMPATIBLE MODULES (MEMORY)		Order No.	Price ($US)
1. **GPM General Purpose Memory Module** with 2048 byte CUTER masked ROM, space for 8192 bytes of 2708 type EPROM or ROM, and 1024 bytes RAM.	Factory Assembled/Tested Kit	210000-01 210000-02	$189 $149
2. **GPM-Sol** same as GPM without RAM or CUTER ROM. Includes space for 10,240 bytes of 2708 type EPROM.	Factory Assembled/Tested Kit	210000-03 210000-04	$129 $99
3. **ALS-8** ROM Resident Assembly Language Operating System with Interpretive Simulator (SIM-1) and Text Editor (TXT-2). For use with GPM or GPM-Sol. If purchased together, price includes assembly and testing. (ALS-8, SIM-1 and TXT-2 programs are copyrighted.)	Factory Assembled/Tested or Kit	900014	$190
4. **2KRO** Erasable PROM S-100 bus Module with capacity for up to 2048 bytes of EPROM-stored programs. (Does not include EPROM's.)	Factory Assembled/Tested Kit	204000-01 204000-02	$89 $65
5. **4KRA Memory Module** with 4096 words of Low Power Static RAM	Factory Assembled/Tested Kit	201000-01 201000-02	$150 $125
6. **8KRA Memory Module** with 8192 bytes of Low Power Static RAM	Factory Assembled/Tested Kit	202000-01 202000-02	$250 $225
7. **16KRA Memory Module** with 16,384 bytes of Dynamic RAM	Factory Assembled/Tested	203000-01	$429
8. **32KRA Memory Module** with 32,768 bytes of Dynamic RAM	Factory Assembled/Tested	203100	$750

S-100 BUS COMPATIBLE MODULES (INTERFACE)			
1. **3P + S Parallel/Serial I/O Module** features two 8-bit parallel I/O ports plus a serial port with RS-232C and 20 mA current loop I/O, 75 to 9500 baud.	Factory Assembled/Tested Kit	209000-01 209000-02	$199 $149
2. **CUTS Computer Users Tape System** audio cassette interface (includes CUTER tape) operates at 300 or 1200 baud. Panasonic RQ-413A cassette recorder recommended.	Factory Assembled/Tested Kit	207000-01 207000-02	$189 $149
3. **VDM-1 Video Display Module** offers EIA video output with either 625 or 525 vertical line resolution. 16 lines X 64 characters generated in upper and lower case font. High speed module has 1024 bytes of RAM, multiple programmable cursor control, selectable blinking and scrolling up to 2000 lines per minute.	Factory Assembled/Tested Kit	208000-01 208000-02	$295 $199
4. **WWB Wirewrap Prototyping Module**	Kit	211000-02	$40
5. **EXB Extender Board**	Kit	212000-02	$35

SOFTWARE (INCLUDING MANUAL)	Minimum Memory Required		
1. **Extended Cassette BASIC**	16K	727017	$45.00
2. **BASIC/5** cassette	10K	727000	$19.50
3. **New 8080 FOCAL*** cassette	10K	727024	$19.50
4. **TREK 80** Video Game cassette	8K	727009	$19.50
5. **GAMEPAC 1** Video Games cassette	4K	727006	$19.50
6. **MATHPACK** Video calculator cassette	4K	727020	$19.50
7. **Software #1** Resident Assembler Package cassette	6K	727022	$19.50
8. **ALS-8** Resident Assembler, Simulator and Text Editor cassette	12K	727012	$45.00
9. **Extended Disk BASIC**	32K	727036	$50.00
10. **PTDOS 1.4 System Disk** (replacement)	32K	727030	$50.00
11. **Extended Disk FORTRAN**	32K	727100	$50.00
12. **Cassette PILOT**	16K	727110	$24.50
13. **EDIT, 8080 Cassette Editor**	8K	727140	$19.50

* Registered trademark of Digital Equipment Corporation.

Attention

A SOLOS or CUTER monitor program is necessary to load and run cassette software products.

All Processor Technology software is distributed through authorized dealers or on an individual sale basis for personal use. Each software package has been copyrighted by Processor Technology and all rights therein are reserved. No license to copy or duplicate is granted with distribution or subsequent resale.

Prices and specifications are subject to change without notice. Prices are in U.S. dollars, apply in U.S.A. only and may be slightly higher elsewhere.

Processor Technology Corporation 7100 Johnson Industrial Drive, Pleasanton, California 94566 (415) 829-2600

THE PERSONAL COMPUTER RETAILER

Vol.2 No.6 Published by Processor Technology Corporation July 18, 1978
7100 Johnson Industrial Drive, Pleasanton, CA 94566

New extended BASIC option

A recent letter to all Processor Technology dealers announced the new family of BASIC's. Included is an option which converts any Extended BASIC disk (currently 8-digit precision) to 6, 10, 12, 14 or 16 digits of precision. This option is available to all authorized Processor Technology dealers for an initial flat fee of $350.

Dealers can customize the level of precision at the time of sale or retroactively for Sol users who would now like the advantages of greater precision, particularly for accounting applications.

Programs written in the original version of Extended BASIC will be fully compatible after the conversion.

New software to expand and protect YOUR programs.....story page 1

In this issue...

You want it when?!.....page 2

Editorial--Avoid unnecessary charges.....page 2

Upcoming tech training seminars emphsize day-to-day maintenance & standardized repair procedures.....page 4

Your free copy of revised SOLOS/CUTER Manual.....page 3

Memory board task force develops field quality assurance program.....page 3

Business BASIC protects YOUR valuable software

Also announced was the new Processor Technology Level I Business BASIC for software developers. For a flat fee of $2,000 you get a unique version which includes both development and runtime programs. Your BASIC programs will run but cannot be listed, thereby insuring you of privacy and added security.

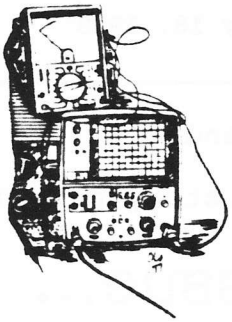
Your version of BASIC is unique and another developer cannot unscramble your programs with his version of Business BASIC. With this approach you get the major advantage of both interpreters and compilers, i.e. easy development and debugging plus runtime security.

Optional precision is included with Level I Business BASIC.

By paying this low, one-time-only fee you will never again have to pay royalities when using the programs.

Both Optional Precision and Level I Business BASIC will be available for delivery soon.

Memory board Task Force at work



We've had a group of our technical staff from engineering, production, and quality control working since May on the top-priority project of identifying all areas of difficulty in the 16KRA memory board and to insure that all such difficulties are eliminated.

According to the latest reports they are more than 90% there now. When the work is completed all hard failures and all timing discrepancies will be eliminated as well as most intermittent failures.

The Task Force has also developed a new complete test and verification procedure for use in the factory and field. The new Field Quality Assurance Program will guarantee 100% performance if followed carefully by your technicians. Included will be specs for test fixtures and equipment, comprehensive test and diagnostic software, and a fixed test procedure. We will also be covering these procedures in depth at our Technical Training seminars in August.

New SOLOS/CUTER Manual benefits all Sol users

A greatly expanded second edition of the SOLOS/CUTER Manual is now being shipped with all Sols. Included with this newsletter, free of charge, is your reference copy.

After looking it over I'm sure you'll agree with us that all Sol users, old and new, will benefit from the useful additional information contained in this new edition. Therefore, we encourage you to notify your customers of the availability of the new manual.

These manuals are available for immediate shipment at the normal dealer discount of 40%. The suggested retail price is $5.00.

The new issue of Access will also carry an article on the new edition. Manuals will be available for end-user purchase directly from the factory for $5.00. We will, however, make it clear that these manuals are also available through the local dealers.

KEY CHANGES IN THE MANUAL

1. All command descriptions are expanded and clarified.

2. Procedures for the use of the cassette recorder controls are now integrated with the command procedures.

(Continued on page 4)

Tech training seminars to introduce standardized repair procedures

"By popular demand the upcoming seminars have been designed to cater to practical applications with an emphasis on day-to-day maintenance and troubleshooting," says the coordinator, Aram Attarian II, formerly lead technician in Warranty Repair and Editor of Access.

Our main goal is to introduce to you a list of factory authorized repair procedures. These procedures will be identical to those used in-house. We are planning to share a coveted list of procedures and short-cuts which have been accumulated by our techs over the years. For the first time we will have a common set of terms and definitions from which to communicate.

You will also have hands-on experience by actually doing debugging, as well as a chance to work with the Helios and become more familiar with PTDOS.

You'll have a chance to meet and chat with your counterparts around the country. There is an hour set aside at the end of each day's session for an interchange of ideas and general discussion. Plus a reception is planned for the evening prior to the start of the seminar.

We are limited to around 30 participants at each seminar and registration must be made in advance. So be sure to set aside these days:

PLEASANTON, CA.	AUGUST 7, 8 & 9th
PHILADELPHIA, PENN.	AUGUST 28, 29 & 30th

Just fill out the registration form and return it to us as soon as possible in order to reserve your place. Then bring your manuals, questions and ideas and join us for this worthwhile experience.

Remember, whether you live on the East Coast, West Coast, or somewhere in between, there is a Tech Training Seminar just waiting for you. Don't miss it!

New SOLOS CUTER Manual

(Continued from page 3)

3. The use of typesetting allows for easier reading. Quicker reference also is possible because of a special "monospace" typeface which is used to denote all dialogue with SOLOS.

KEY ADDITIONS TO THE MANUAL

Several important new sections have been added:

1. Section 1.6. Deals with entering commands. Describes various functions in SOLOS and how they may be edited and used.

2. Section 5, System Interfacing. Instructs the user on how to call SOLOS sub-routines from other programs.

3. Appendix I. Gives general tips on using cassette recorders for data storage.

4. Appendix II. Contains a complete chart of ASCII Codes.

You want it when?!

NEW PRODUCT UPDATES

Item	Begin Shipping	Change
SOFTWARE		
Software #1, Resident 8080 Assembler	shipment has begun	on schedule
Extended Disk FORTRAN	shipment has begun	on schedule
Cassette PILOT	shipment has begun	on schedule
EDIT, Advanced 8080 Editor	shipment has begun	on schedule
ASSM, Advanced 8080 Assembler	week of July 31	
Math Pack Video Calculator	week of July 31	
8080 Chess Cassette	week of July 31	
Game Pac II	week of August 14	
Debug, Advanced 8080 Debugger	week of September 4	

Avoid unnecessary charges!

by Dottie Wanat

I'm sure you've all received the recent letter regarding the implementation of a 15% restocking charge for refused merchandise. I'm also pretty sure you're not jumping up and down! So, just a brief word in our defense. (Please don't stop reading).

First of all, rememeber that this charge is only made when a shipment is refused. Also, it only covers the additional costs that we incur by such a practice. As long as you give us reasonable notice that you wish to either reschedule or cancel part of your order, no charge will be made. This policy places the cost burdens on those who create them.

I'm sure that by working together in a cooperative manner we will all realize the benefits from these cost- and time-saving procedures.

Whodunnit

Editor.....Dottie Wanat

Graphics.....Lois Esser
Ray Oleson

General Input.....Loren Willey
Rick Meyer
Brett Bullington
Aram Attarian II
Elizabeth Fairchild

Copyright © 1978,
by Processor Technology Corporation.
All rights reserved.

Gentlemen, start your oscilloscopes!

Our new Technical Training Seminar will be presented for the first time on August 7, 8 and 9 at the factory in Pleasanton. Be sure to bring your manuals, questions and a PTDOS system diskette. During the seminar we will copy new diagnostic procedures for Sol and Helios onto your diskette.

At the completion of the session your tech will receive certificates which verify that he has completed a factory-sponsored technical seminar for either hardware, software or both.

Also, don't forget the get-acquainted reception planned for Sunday evening at 7 p.m. at the Howard Johnson's Motor Inn, 6680 Regional Street, Dublin, CA.

For those of you in the Midwest and on the East Coast, this same Seminar will be conducted again in Philadelphia on August 28, 29 and 30.

We hope you will send a representative to one of these valuable sessions.

Software Sentre

by Connie Hawkinson

This is the first in a series of columns about Helios software developed by Processor Technology dealers. My purpose is to serve as an information clearinghouse. I will briefly describe the Helios software package you offer and list your name, store, and phone number so that interested dealers may contact you directly for further information. Processor Technology will NOT be involved in any of these transactions. We do intend to encourage software exchange among dealers to benefit your Helios system sales as well as to prevent any duplication of efforts.

Below is the current listing. Due to time constraints I cannot contact all of our dealers. Therefore, if you are interested in participating in the "exchange", please contact me at the factory.

CURRENT LISTINGS:

COMPUTER EMPORIUM, Cherry Hill, New Jersey. Contact Joel Schusterman at (609) 667-7555. They have developed a standard system (stored on four diskettes) designed to provide a basic computerized package to meet the small to medium-sized company's information needs in the areas of sales order entry, invoicing, inventory control, accounts payable, accounts receivable, and sales analysis.

Sales tip

Here's an idea from Loren Willey, our Retail Marketing Manager: many dealers sell the Sol and Helios manuals separately. A potential customer who is technically inclined will often appreciate the opportunity to become familiar with the details of the system before he makes a purchase.

How about "loaning" the manual on a short term basis? Even if your customer is still undecided, he will have to come back to your store to return the manual, and you'll have another chance to present the product.

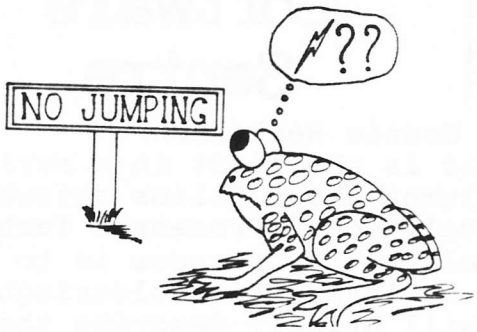
Both Sol and Helios manuals may be purchased separately. A scheduled order is not necessary. Just send a purchase order to your dealer coordinator. The Sol System manual has part number 730000 and retails for $25. The Helios manual, part number 730023, retails for $50.

by Ron Carpenter

"If anything can go wrong, it will. If there is a possibility of one of several things going wrong, the one that will happen is the one that will do the most damage."

Murphy's Law is particularly relevant when dealing with electronic systems, and, of course, microcomputers are certainly no exception. So, given this theoretical base, what's the best way to approach the inevitable?

A systematic approach to troubleshooting



We've designed some broad-based guidelines to give you a good overview of a systems approach to troubleshooting.

One of the most important things to remember when troubleshooting is NEVER JUMP TO CONCLUSIONS.

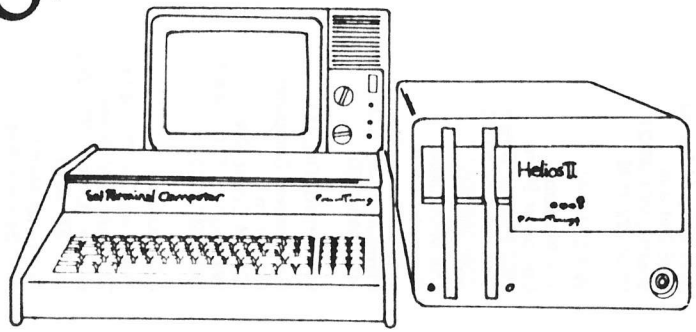
With any computer system, a combination of elements rather than just one may be at the root of a problem. Possible sources of difficulty are: 1) The Operator; 2) The Recording Media; 3) Mother Nature; and finally, 4) The Hardware. Start your analysis by first duplicating the symptoms. Then eliminate each possible cause one-by-one in the following order.

1. Check for Operator Error. If possible, let the customer operate the system. You may discover the problem lies right there. If not, go on to Step 2.
2. Recording Media Error (diskette or cassette). Be sure that you are using the same diskette or cassette used when the failure occurred. It's possible that a crashed diskette caused the problem. If not, continue on to Step 3.
3. Mother Nature. Was there some type of storm or electrical disturbance going on at the time of failure? Weather has been the cause in a surprising number of cases.
4. Hardware Error. If, after eliminating the first three possibilities the system is still not performing properly, check the hardware. Run an individual comprehensive test on each of the four subassemblies in this order: 1) Memory; 2) Helios cabinet: 3) Controller/Formatter: and 4) the Sol. Each test will isolate problems within the subassembly itself. As you discover and repair each problem, rerun the comprehensive test. Continue repeating the test for each subassembly until all problems are found and repaired.

Also, remember that the more information (detailed symptoms, test results, etc.) you have, the easier it is to locate and solve a problem. So be sure to keep records of your findings at each stage of the analysis.

Again, these are very general guidelines. A detailed set of Standardized Repair Procedures will be given to each attendee at our upcoming Tech Seminars. In combination with the detailed procedures, these general tips can improve the efficiency and effectiveness of your troubleshooting.

Processor Technology Corporation



FACTORY TRAINED

This certifies that

has completed a program
in Systems Diagnostics and Maintenance

Robert Nash
Vice-President of Marketing

Instructor

August 1978
Date

1 EPT 1 - 93

See it at your Sol dealer...

ALABAMA

ICP, Computerland
1550-D Montgomery Hwy.
Birmingham, AL 35226
(205) 979-0707

CALIFORNIA

The Byte Shop
1514 University Ave.
Berkeley, CA 94703
(415) 845-6366

Byte Shop Computer Store
6041 Greenback Lane
Citrus Heights, CA 95610
(916) 961-2983

Orange County
Computer Center
1913 Harbor Blvd.
Costa Mesa, CA 92627
(714) 646-0221

Computerland of Hayward
22634 Foothill Blvd.
Hayward, CA 94541
(415) 538-8080

The Byte Shop
16508 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-2421

Computer Magic
700 McHenry, Suite E
Modesto, CA 95350
(209) 527-5156

Digital Deli
80 W. El Camino Real
Mountain View, CA 94040
'415) 961-2670

Computer Center, Inc.
4014 Geary Blvd.
San Francisco, CA 94118
(415) 387-2513

The Byte Shop
509-B Francisco Blvd.
San Rafael, CA 94901
(415) 457-9311

The Byte Shop
2989 North Main Street
Walnut Creek, CA 94596
(415) 933-6252

COLORADO

Byte Shop
3101 Walnut Street
Boulder, CO 80301
(303) 444-6550

Byte Shop
E. 1st Ave. & University
Denver, CO 80206
(303) 399-8995

CONNECTICUT

Technology Systems
20 Chestnut Street
Bethel, CT 06801
(203) 748-6856

FLORIDA

Byte Shop of Ft. Lauderdale
1044 E. Oakland Blvd.
Ft. Lauderdale, FL 33334
(305) 561-2983

Byte Shop of Miami
7825 Bird Road
Miami, FL 33155
(305) 264-2983

Microcomputer
Systems, Inc.
144 S. Dale Mabry Hwy.
Tampa, FL 33609
(813) 879-4301

GEORGIA

Atlanta Computer Mart
5091-B Buford Hwy.
Atlanta, GA 30340
(404) 455-0647

ILLINOIS

Midwest Microcomputer
708 S. Main Street
Lombard, IL 60148
(312) 495-9889

IOWA

The Computer Store
of Davenport
4128 Brady Street
Davenport, IA 52806
(319) 386-3330

KENTUCKY

Martronix Associates
1235 E. Oak Street
Louisville, KY 40204
(502) 459-0500

MARYLAND

Computers Etc.
13A Allegheny
Towson, MD 21204
(301) 296-0520

MINNESOTA

Computer Depot
3515 W. 70th Street
Minneapolis, MN 55435
(612) 927-5601

MISSOURI

Computer Country
235 Dunn Road
Florissant, MO 63031
(314) 921-4434

NEW HAMPSHIRE

Computerland of Nashua
419 Amherst Street
Nashua, NH 03060
(603) 887-5238

NEW JERSEY

Computer Emporium
2438 Route 38
Cherry Hill, NJ 08002
(609) 667-7555

Computer Mart of
New Jersey
501 Route 27
Iselin, NJ 08830
(201) 283-0600

NEW YORK

The Computer Tree
409 Hooper Road
Endwell, NY 13760
(607) 748-1223

The Computer Mart of
New York
118 Madison Ave.
New York, NY 10001
(212) 686-7923

The Computer Corner
200 Hamilton Ave.
White Plains, NY 10601
(914) 949-3282

NORTH CAROLINA

ROMs 'N' RAMs
Crabtree Valley Mall
Raleigh, NC 27604
(919) 781-0003

OHIO

The Basic Computer Shop
Fairlawn Plaza
2671 W. Market Street
Akron, OH 44313
(216) 867-0808

Byte Shop
2432 Chester Lane
Columbus, OH 43321
(614) 486-7761

Computer Mart of Dayton
2665 S. Dixie Ave.
Dayton, OH 45409
(513) 296-1248

OREGON

Byte Shop Computer Store
3482 S.W. Cedar Hills Blvd.
Beaverton, OR 97005
(503) 644-2686

Byte Shop Computer Store
2033 S.W. 4th Ave.
Portland, OR 97201
(503) 223-3496

Computer Pathways
145 Alice Street So.
Salem, OR 97302
(503) 399-0534

PENNSYLVANIA

Computer Mart of
Pennsylvania
550 DeKalb Pike
King of Prussia, PA 19406
(215) 265-2580

RHODE ISLAND

Computer Power, Inc.
M24 Airport Mall
1800 Post Road
Warwick, RI 02886
(401) 738-4477

SOUTH CAROLINA

Byte Shop
1920 Blossom Street
Columbia, SC 29205
(803) 771-7824

TENNESSEE

Microproducts & Systems
2307 E. Center Street
Kingsport, TN 37664
(615) 245-8081

TEXAS

Computer Port
926 N. Collins
Arlington, TX 76011
(817) 469-1502

Micro Store
312 Randol Mill Road
Arlington, TX 75080
(817) 461-6081

Interactive Computers
7620 Dashwood
Houston, TX 77036
(713) 772-5257

Interactive Computers
16440 El Camino Real
Houston, TX 77058
(713) 486-0291

Neighborhood Computer
Store
4902-34th Street
Lubbock, TX 79410
(806) 797-1468

Micro Store
634 S. Central
Richardson, TX 75080
(214) 231-1096

UTAH

Home Computer Store
2343 East 3300 South
Salt Lake City, UT 84109
(801) 484-6502

VIRGINIA

The Computer
Systems Store
1984 Chain Bridge Road
McLean, VA 22101
(703) 821-8333

WASHINGTON

Byte Shop Computer Store
14701 N.E. 20th Ave.
Bellevue, WA 98007
(206) 746-0651

Byte Shop of Seattle
2605 2nd Ave.
Seattle, WA 98121
(206) 622-7196

WISCONSIN

The Madision
Computer Store
1863 Monroe Street
Madision, WI 53711
(608) 255-5552

The Milwaukee
Computer Store
4710 W. North Ave.
Milwaukee, WI 53208
(414) 445-4280

WASHINGTON, D.C.

Georgetown Computer
Emporium
3286 M Street NW
Washington, D.C. 20007
(202) 337-6545

CANADA

The Computer Circuit Ltd.
737 Richmond Street
London, Ontario N6A 3H2
(519) 672-9370

Computer Mart Ltd.
1543 Bayview Ave.
Toronto, Ontario M4G 3B5
(416) 484-9708

Basic Computer Group Ltd.
1548 West 8th Ave.
Vancouver, B.C. V6J 4R8
(604) 736-7474

ARGENTINA

Basis Sistemas Digitales
Colombres 849-1-A
1218 Buenos Aires
93-1988 or 57-7177

AUSTRALIA

Automation Statham
Pty. Ltd.
47 Birch Street
Bankstown 2200
Sydney
(02) 709.4144

BELGIUM

S.P.R.L. Finncontact
Square Larousse, 5
1060 Bruxelles
2/345-98-95

COLOMBIA

Video National
Diagonal 34 N. 5-62
Apartado Aeree 27499
Bogota
326650

GREECE

Attikos, Inc.
41 Char. Tricoupi T.T. 145
Athens
3604091

JAPAN

Japan Personal
Computers, Ltd.
Yamamoto Bldg. 11-18
2-Chome
Yoyogi, Shibuyaku
Tokyo 151
(03) 375-5078

MEXICO

Industrias Digitales
S.A. de C.V.
Plateros 19, D.F. Mexico
(905) 524-5132

PHILIPPINES

Integrated Computer
Systems, Inc.
Suite 118
LIM KET KAI Bldg.
Ortigas Avenue
San Juan,
Metro Manila 3113

SWEDEN

Wernor Elektronix
Torsvagen 61
Box 72
S-133 01 Saltsjobaden
Stockholm
(0) 8 717-62-88

UNITED KINGDOM

Comart, Ltd.
24A Market Square
St. Neots, Cambridgeshire
Huntingdon,
PE19 2AF England
(0480) 74356

VENEZUELA

Componentes Y Circuitos
Electronicos Ttlca
Ave. Fco.de Miranda
Res. Concordia-Local 4
Los Ruices, Caracas 101
355591

ProcessorTechnology

8/78

THE PERSONAL COMPUTER RETAILER

Vol.2 No.9 Published by Processor Technology Corporation September 7, 1978
7100 Johnson Industrial Drive, Pleasanton, CA 94566

Word processing debuts in Philadelphia

Processor Technology's new Word Processing software was greeted with enthusiasm by dealers at the Philadelphia Personal Computer show August 24-27.

Our first business application package uses PTDOS in a Sol System III with a Diablo printer. Basic Computer Group Ltd. of Vancouver, B.C., developed and has already installed the Word Processing system in a number of legal offices in the Vancouver area.

According to Brian Bristow, owner of Basic Computer Group, complete installation and training of operators can be accomplished in two weeks. His customers have found that the system actually improves morale in an office. Secretaries compete for time at the machine. The Sol is particularly popular because it makes less noise than an electric typewriter.

Installation, operator training and maintenance service are an integral part of Basic Computer Group's sales package. In most cases they repair or replace down equipment within four hours.
(Continued on page 3)

Rita & Larry Johnson (Microproducts and Systems, Kingsport, TN) discuss the new Word Processing system with Dottie Wanat (seated) at the Philadelphia Personal Computing show.....
.....see story page 1

Inventory flooring announced

Fifty dealers packed into our breakfast seminar held Friday morning August 25 in conjunction with the Philadelphia Personal Computing show. The star presentation came from Paul Cottone of FinanceAmerica who explained the details of our new retail inventory flooring program.

Offering up to 90-day financing on all purchases from Processor Technology Corporation, this program will free qualified dealers to schedule deliveries based on current need rather than cash flow considerations.
(Continued on page 5)

You want it when?!

Item	Shipment Begins	Change
8080 CHESS cassette	week of September 18	moved back 3 weeks
GAMEPAC 2	week of September 11	moved back 2 weeks
DEBUG	week of September 18	moved back 2 weeks

Meanwhile, back in Pleasanton...

I'm filling in for Dottie on this issue while she has all the fun in Philadelphia. It's sure been quiet in the Advertising Department since she left.

In addition to listening for the phone to ring with news about the show, we've been trying to get some work done here. Bob Marsh and Connie Hawkinson have hammered out the details for next quarter's order packet which includes a number of product changes.

Linda Allen has been organizing all the administrative details for the "Christmas in Hawaii" sales contest. She tells me that only 18 people have signed up so far. I'd say that gives them good odds for getting one of the twelve prizes. So if you want to be a winner, SIGN UP.

Linda Bilbrey tells me that Warranty Repair has borrowed techs from the production department to speed up the process of reducing the backlog. She would also like to remind you that after August 31 we will no longer accept any equipment returned for repair or for credit without an R.A. (return authorization).

Here in the advertising department, I'm working on a new ad, the latest in our series featuring successful Sol dealers for COMPUTER DEALER and COMPUTER RETAILING. In this one, Larry Stein (Computer Mart of New Jersey) talks about his approach to the business customer.

So, that's the latest.

--Elizabeth Fairchild

Whodunnit

Editor.....Dottie Wanat

Stand-in Editor.....Elizabeth Fairchild

Graphics.....Lois Esser
Ray Oleson

Photography...Brett Bullington

General Input.Connie Hawkinson
Edith Reisner
Terry Walters
Drew Rogge
Rick Meyer
Ellwood Douglas

Copyright © 1978,
Processor Technology Corporation.
All rights reserved.

Word processing debuts...

(Continued from page 1)
The most impressive feature of the Word Processing software, according to dealers who saw it in Philadelphia, is the ability to edit one document on the screen while the printer is putting out another document at 45 characters per second.

Text can be typed in at high speed since carriage returns are added by the software. Automatic justification creates an even right margin, or the text can be left ragged right. Since all formatting is done on the screen, lines of text will be printed exactly as they appear. Several modes of cursor control and scrolling allow rapid access to any position of the document.

Commands which affect format (headings, footings, character and line spacing, top and bottom margins, and widow elimination) can be imbedded in the document for dynamic control of the printed output. Non-printing comments can also be added.

With only a small portion of the document in memory at one time, a given document can be up to 110 pages in length. Disk access is handled automatically as the document is scrolled or edited. In the event of a power failure or crash, the file structure remains intact, and only a small segment of text is lost. Up to 20 different documents totalling 110 pages can be on one diskette.

Another useful feature is that the software permits line length up to 128 characters wide. The 64-character width on the video monitor is expanded by sideways scrolling commands which allow the operator to see what is on the right side of the copy.

Two letter quality printers and one dot matrix printer will be available from Processor Technology beginning November 15.

Details on price and availability of the Word Processing software will be announced soon.

Co-op ads hard at work

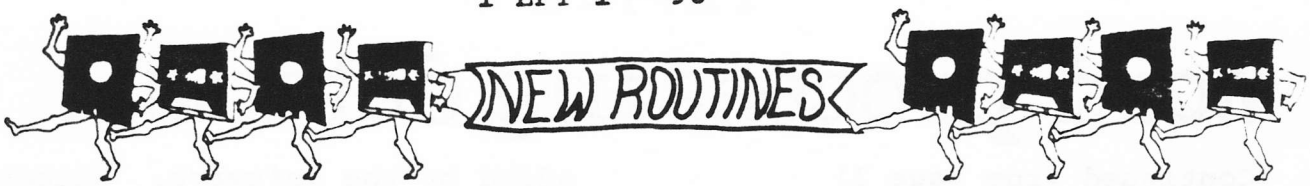
Two dealers have submitted co-op ads that are running successfully.

Bob Vipond (Computer Magic) has written and produced a first-rate radio commercial about Sol Systems to run on a local Modesto, California station. He says that store traffic has doubled as a result of the ad.

Hal and Harriet Shair (Computer Corner) ran the "Play a little Sol Music" ad in the business section of nine papers in Westchester County on Sunday, August 27. By Tuesday, Harriet reported that the ad was pulling more responses than any they had run previously in newspapers.

Due to the newspaper strike in New York City, the Westchester papers have tripled their circulation giving the ad extra exposure. Many people were calling and coming in who had not been aware of the Computer Corner before.

She added that the piano image has not created any confusion for her readers who were mostly business people looking for a business system.



New software introduced by Processor Technology this quarter includes two new video games called GAMEPAC 2. There's also three development programs on cassette which are modifications of portions of our PTDOS disk operating system. These three programs may be used in combination to create or modify large programs in a minimum amount of memory.

GAMEPAC 2

HANGMAN is an entertaining game that keeps children of all ages amused for hours. The program comes with a word list which you can add to or alter.

In addition to entertainment, HANGMAN is useful for teachers to make learning spelling, foreign language or any specialized terminology fun for students.

QUBIC is very much like a three-dimensional tic-tac-toe with a playing grid of 4 by 4 locations on 4 levels with 76 ways to win. You can play either with another player or against the computer. Rumor has it that the program sometimes cheats when the computer is about to lose.

EDIT

EDIT is a cassette based editor similar to EDT 3.0 on the PTDOS system disk. It allows you to create or alter source programs in a form usable by Cassette FORTRAN, ASSM, or Extended Cassette BASIC.

Editing may be done on character, string, line and page levels. At any of these levels additions, insertions, substitutions and deletions of text may be made. Additionally, EDIT offers the option to retain a command string as a macro and execute it repeatedly.

The EDIT program itself resides in low memory and requires approximately 4K.

When you use EDIT, you will most likely be using two cassette recorders, reading from one and writing to the other. EDIT saves and loads text as a multiple-block file on cassette.

DEBUG

DEBUG is an aid for debugging a

(Continued on page 5)

technical stuff

For those of you who missed the technical seminars where our new family of dynamic RAM memory modules were introduced, here's some details:

The 16KRA-1, 32KRA-1, 48KRA-1 and 64KRA-1 utilize either 8K or 16K dynamic RAM's. These new boards are designed to maximize reliability and to optimize manufacturing flexibility. Entirely new rigorous QC tests and procedures are being used to guarantee high reliability.

The boards are totally synchronous with the S-100 bus "Phase 2" signal and have access times of 400 nsec worst case. Address is switch-selectable in four blocks.

A variety of options for Extended Memory Address make the board very flexible through use of memory "Bank Select" techniques.

The 32KRA-1 version of the new design has already been shipped in Sol System III's and IV's. The 16, 48 and 64K versions will be on their way in September and October.

(Continued from page 4)
machine language program developed and assembled on a Sol or other 8080 computer system with CUTER. With this program, you can set as many as fifteen "breakpoints" in a program you want to debug or examine. It is capable of dumping memory in hexidecimal, ASCII or instruction modes.

When a program is executed under the control of DEBUG, it will stop at each of these addresses so that CPU registers, flags, and specified memory locations may be examined and modified. You can resume execution at a breakpoint or other specified memory location after making modifications. At the conclusion of debugging, you can save the modified version of your program.

There are five versions of the DEBUG program on the cassette tape. Each resides at a different memory location. Each occupies 4½K.

ASSM

The ASSM program on cassette tape translates symbolic 8080 assembly language source code into object code. It is designed for use with the Sol or other 8080 machine using CUTER. The assembler itself occupies almost 8K of memory. Additional memory is required for the symbol table and user program. Two cassette recorders are used.

Two other programs are recorded on the tape, PACK and UNPAC. These programs convert a cassette file from single-block format to multiple-block format and vice versa.

DELIVERY

ASSM and EDIT have already been shipped to dealers. GAMEPAC 2 is scheduled for shipment during the week of September 11; DEBUG during the week of September 18.

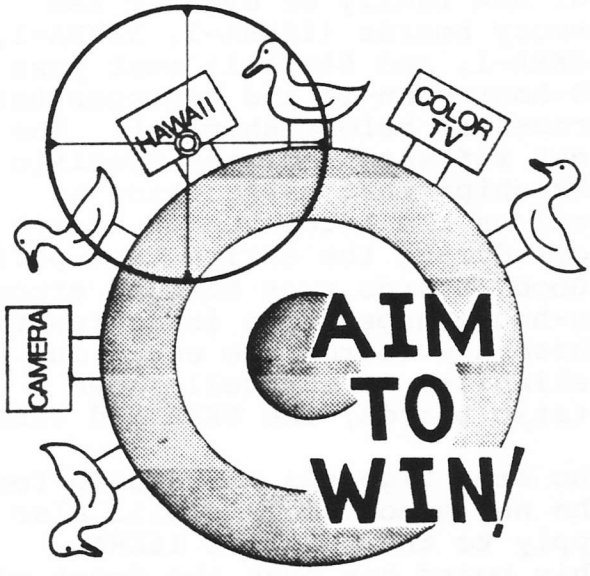
Flooring announced...

(Continued from page 1)
FinanceAmerica is a subsidiary of the Bank of America, the world's largest bank.

The flooring program will go into effect on October 1 with Fall Quarter scheduled orders and will be administered from the factory.

At the same time, Processor Technology is now able to extend its regular Net 15 terms to Net 30 for dealers who qualify.

Also on the program for this by-invitation-only seminar was a presentation by Loren Willey, our Retail Marketing Manager, with ideas for successful in-store merchandising. Dottie Wanat from our Advertising Department offered guidelines for developing your own ad campaign and utilizing the co-op advertising funds effectively.



Don't miss out. The "Christmas in Hawaii" sales contest is on. Register today. Full details available from Linda Allen in the Dealer Sales department at the factory.

Quality control tightens up

New and more rigorous test procedures will be reflected in the quality of Sol, Helios and memory board products we are now shipping. These higher standards are geared to the requirements of the small business system market.

We are testing all Sol Systems III and IV as complete systems after the individual units pass final tests. This system test procedure, in effect for the past two months, has already resulted in improved reliability in the field.

All incoming parts are now screened before they are put into stock. Critical components such as the 8080A are individually tested.

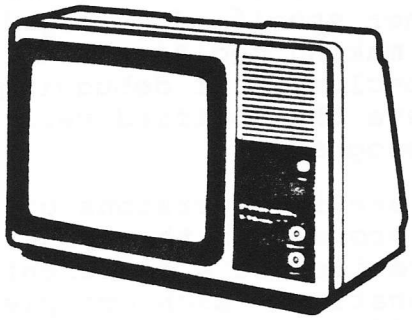
In addition to regular test procedures during construction, all Sol's must now pass a final 15-hour monitored test with no errors before shipment.

Our new family of dynamic RAM memory boards (16KRA-1, 32KRA-1, 48KRA-1, and 64KRA-1) must pass a 48-hour burn-in and rigorous test procedure before shipment. The test software isolates specific RAM chips that malfunction as well as the total number of errors during the entire test period. "Good" boards must have no errors. In-house experience indicates that these new boards are equal in reliabilty to our well known static boards, the 8KRA and 4KRA.

The same rigorous tests used for the new memory boards will also apply to the original 16KRA. This board has been the focus of a recent quality assurance task force which proposed some specific modifications to improve operation in the field.

As a further measure, Warranty Repair is running all 16KRA boards through the production tests before returning them to the dealers. According to Technical Support Manager Mike Mills, 200 16KRA's have passed through his department for repair in the past two months. Only three have come back a second time. And those three had mechanical problems rather than component or electronic failures.

Our highest priority is now focused on reducing any backlog of dealer equipment in our Warranty Repair Department and restoring a 5-day turnaround time.



LOOK, MA, NO KNOBS! Beginning October 1, our video monitors will have an attractive front panel eliminating the channel select and volume control knobs.

New 64KRA-1 goes to New Jersey

Alice Golembo of the Computer Mart of New Jersey won the door-prize at our open house held Thursday night, August 24, at the Philadelphia Hilton Hotel. Her prize--the new 64KRA-1 memory module. Her husband, Eri Golembo, the store's Administrative Director, says they will probably use the board in a Sol-20 they may buy through the salesperson's purchase plan.

WordWizard Electronic Typing System from Processor Technology

[B]ring the speed and efficiency of the newest technology [t]o your office without breaking your budget

1 EPT 1 - 101

Handle your typing load faster and more accurately.

Electronic typing used to be expensive... $20,000 and up for a typical word processing machine. Only large corporations and government agencies could afford these advanced systems with all the professional features.

Now, Processor Technology drastically cuts the cost of electronic typing to give big company typing power to even the smallest office. Here's how our WordWizard system works to cut your paperwork costs:

Your typist can type from dictation at top speed without stopping to check for errors. There's no need even to gauge the end of a line for the carriage return because WordWizard automatically moves to the next line.

Proofreading and corrections are done on the screen before anything goes down on paper. Change spelling, add or delete words, and move sentences or whole paragraphs with a few keystrokes. Once you have a good draft, the printer produces a single, double or triple spaced copy in minutes. WordWizard memorizes the original draft and stores it on a WordWizard diskette.

After you've reviewed the document and marked your additions or corrections, your typist can display the draft on the screen and enter your changes electronically. No retyping is necessary. Only the changes. WordWizard automatically makes room to add sentences or paragraphs and closes up space where you've made deletions.

If a word or phrase needs changing anywhere in the document, a simple operation called Global Search & Replace will locate every occurrence of the word or phrase and correct it in seconds.

When the document is complete, the SolPrinter will produce a copy at 480 wpm with single or double spacing, with even right-hand margins, or with bold headings. In minutes, you'll have a perfect, professional-looking report or letter without a single visible correction. Your secretary will be delighted to see an end to constant retyping.

One more thing. While the printer is running, your typist can work on the next project at the screen. No waiting for the printer to finish.

Your WordWizard electronic typing system pays for itself fast. You'll wonder how you ever did without it.

Attorneys and accountants say WordWizard is particularly cost effective.

Legal offices have been among the first to use WordWizard. Legal documents must be absolutely error-free and frequently contain standard paragraphs used over and over again. Such paragraphs are stored on diskettes and later called up by number or name. Your current numbering system for standard paragraphs will, in most cases, adapt to WordWizard without modification. Clerical time to produce wills and contracts can be cut drastically.

Accountants appreciate WordWizard's ability to type copy up to 128 characters wide. You can view half of a wide page at once on the screen. By using the horizontal scroll, you can move the image left or right to view the entire page.

A WordWizard feature especially useful in sales and customer service offices is its ability to type many original versions of a letter for different individuals quickly. Your mailing list is stored on a diskette which can be easily updated. To type a group of letters, call up the list on the screen, or any part of it. For example, you may want to write to customers in a particular zipcode area who own just one of your products. Then, the list or any part of it can be merged with the letter you want to send. A secretary who used to type 25 sales letters in a day now can do 200.

Accountants, medical offices and many other businesses take advantage of WordWizard's speedy form completion. As your secretary fills in a form, the system remembers the next space and jumps there directly with two keystrokes.

So easy to learn you can begin to use it with just a few minutes of instruction.

You don't have to be a technical genius to use WordWizard. Unlike many systems that require memorizing special "codes," WordWizard includes a set of keyboard labels to mark all the editing keys. Use two keystrokes to set a tab, jump to the beginning of the document, delete a line, close up a space, and so forth. WordWizard frequently prompts the operator with easy to understand, ordinary language instructions.

The manual is straightforward with special hands-on exercises to speed learning. It's comprehensive with complete descriptions of all the special functions.

Your typist can start typing electronically right away without worrying about setting margins or making other formatting decisions. WordWizard has useful standard values for an 8 1/2 x 11 page which take over automatically if you don't ask for a special format.

Design your own text format on the screen just as it will be printed.

When you do want a special page arrangement, WordWizard will show your instructions right on the screen. You can change the margins in the middle of a document, switch from double to single spacing, or ask for a header or footer of one or more lines to appear at the top or bottom of each page. You can create an even (justified) right-hand margin, develop a new form or reorganize a group of pages. Such changes appear on the screen as they will look on paper. No guesswork. WordWizard makes it all so easy.

Enter or retrieve up to 110 pages on each diskette.

Each diskette has an alphabetized index that appears on the screen when the disk is inserted. From this index you can locate documents, destroy documents and create new ones. As you make changes, the index is updated automatically. An erasure doublecheck prevents accidental deletion.

You can copy a document from one diskette to another quickly and easily. It's easy to make a duplicate or backup of all your most important documents to keep in an accident-proof location.

In addition to the WordWizard System Diskette, you get one Document Diskette which stores up to 20 different documents totalling as much as 110 pages. The diskette can be reused almost indefinitely. Because the system uses inexpensive, standard diskettes, additional supplies are readily available.

Three optional SolPrinters to choose from.

You can choose a letter-quality SolPrinter 2 with metal print wheel, an economy model SolPrinter 2E with plastic print wheel, or a draft-quality SolPrinter 3 with very fast dot matrix printing.

The SolPrinter 2 prints up to 480 wpm. It s rugged, long-lasting metal print element is the "Daisy Wheel" type. Several interchangeable typefaces are available to work with either carbon or cloth ribbons. The quality of the printed copy is equal to the output of the finest electric typewriter. An economy version, the SolPrinter 2E, uses plastic print wheels.

The much faster SolPrinter 3 prints up to 2400 wpm. Each character pattern of dots is called a dot matrix. This output copy is ideally suited to rough drafts, memos, and in-house documents.

The most efficient installation includes two printers — one Daisy Wheel machine for highest quality letters, reports or reproduction copy, and a dot matrix unit for rapid turnaround of drafts. Alternate between the two printers with a few strokes from the keyboard.

Programmability keeps your system up to date.

WordWizard electronic typing programs work with Processor Technology's multi-purpose small computer systems. The Sol Systems III and IV give you all the power of high-priced word processing machines plus the capability of programming for a wide range of business tasks.

As improvements and new features become available for WordWizard you take advantage of them simply by purchasing new System Diskettes. Many different business and accounting applications packages are available from your Sol Systems dealer. More are on the way. Because WordWizard and the Sol System are totally programmable, your system will stay up to date.

At the outset, you get two machines for the price of one — a full capability word processor *and* a general purpose business computer — for less than you would expect to pay for either one. Your office can take advantage of new, time- and money-saving programs to keep books, send invoices, manage inventory, or provide you with valuable management reports. All without the need for major additional capital investment. As time goes on, you'll be glad you planned for the future.

See your nearby Processor Technology dealer today and ask for a demonstration.

Compare the features. Then compare the price!

Enter up to 110 text pages per diskette.

- Store up to 20 different documents on each diskette
- Identify each document with any name or number
- Unlimited document storage on inexpensive diskettes
- Duplicate or format diskettes rapidly
- Select from menus on screen – no need to learn complex command codes
- Handle common operator mistakes with easy to understand instructions from the screen
- Alphabetized document index is continuously updated and displayed
- 64 character by 16 line video display
- Horizontal scrolling lets you compose and view lines up to 128 characters long
- Adjustable left and right margins
- Automatic margin readjustment by paragraph
- Unlimited adjustable tab stops
- Automatic carriage return, word wraparound, and justification with manual hyphenation
- Easy merging of frequently used phrases, paragraphs and mailing lists
- All documents may include comments which appear on the screen but not on the printout

Changes and corrections are easy with these editing features.

- Easy insertion, deletion or alteration of text
- Removable keyboard labels define edit keys – no need to memorize "edit codes"
- Instant positioning of cursor to start or end of document
- Automatic scrolling – horizontal or vertical, forward or reverse
- Single step or continuous cursor movement in four directions, home cursor, and jump to screen edges
- Automatic search and selective replacement of words, phrases, format statements or comments (Global Search & Replace)
- Copy, move or delete in blocks of up to 16 lines
- Merge standard paragraphs into any document
- Easy merging of mailing lists with letters
- Merge and assembly text can be organized in special library documents

Design your text format right on the screen.

- Margins, indents and justification appear on the screen as they will be printed
- Embedded format control statements are displayed on the screen
- Store many standard text formats to maintain document consistency
- Optional headers and footers on each page – versatile location and multiple lines
- Automatic pagination divides long documents into separate pages with forced page breaks and widow elimination
- Automatic page numbering from 0 to 99 beginning with any number anywhere in the document

Print one document at up to 2400 wpm while editing another on the screen.

- Three optional SolPrinters with speeds of 480 wpm (letter quality) to 2400 wpm (draft quality)
- Bidirectional printing
- Adjustable character and line spacing
- Boldface through multiple overstrike
- Underscore capability
- Copy up to 128 characters wide
- Quick printer selection for multiple-printer installations
- Printing options for skipping pages, page stops, and printing embedded statements or comments

Protect your confidential reports and documents.

- Lock and key power switch prevents unauthorized system use
- Erasure doublecheck protects documents from accidental deletion
- Continual transfer of the document to the diskette prevents loss of all but a small portion in case of power failure
- ASCII communications capability with special programming

Specifications subject to change without notice.

DOCUMENT ACTIVITIES

Create (start new document)
Remove (erase a document)
Edit
Select Merge (select a merge document)
Print
Stop Printing
Eject Page (paper feed)
Archive (copy document to System Disk)
Retrieve (copy archive document to Document Disk)

EDIT KEY FUNCTIONS

Cursor Control (up, down, right, left, or home)
Cursor to Edge of Screen (four directions)
Repeat (character or cursor)
Return (forced end of paragraph)
Left/Right Margin Set/Release
Tab Set/Clear
Jump to Start/End (of document)
Justify On/Off
Search
Continue Search
Replace
Continue Replace
Copy Block
Move Block
Insert Block
Delete Line
Insert Split
Close Paragraph
Call Merge
Reserve Blanks (through justification process)
Escape (go from editing to Document Activity menu)

FORMAT STATEMENTS

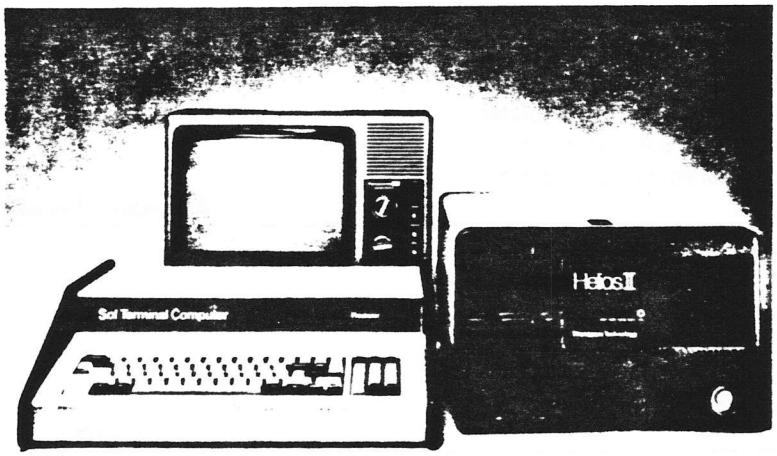
Pitch (horizontal space per character)
Vertical (spacing per line)
Spacing (double, triple, other multiples)
Define Form Length
Lines Per Page
Title (running header)
Footer (running footer)
Header Lines (header – margin)
Bottom Lines (footer – margin)
Number (set or reset page number)
Eject (force new page to start)
Widow (keep lines together on page)
Under (underline, boldface, overstrike)

ProcessorTechnology
7100 Johnson Drive, Pleasanton, CA 94566
(415) 829-2600

effective October 1, 1978

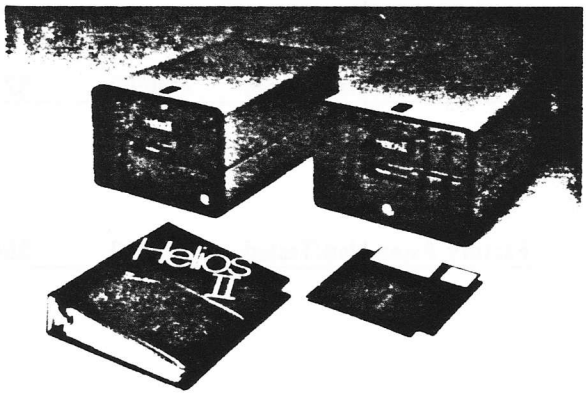
Sol® SYSTEM SUGGESTED U.S. RETAIL PRICE LIST

Sol TERMINAL COMPUTER and Sol SYSTEMS		Order No.	Price ($US)
1. **Sol-20/16 Terminal Computer** with SOLOS personality module and 16,384 bytes of memory, includes Extended Cassette BASIC and Sol System manual	Factory Assembled/Tested	400500	$2095
2. **Sol-20/32 Terminal Computer** with SOLOS personality module and 32,768 bytes of memory, includes Extended Cassette BASIC and Sol System manual	Factory Assembled/Tested	400600	$2395
3. **Sol System I-A** includes Sol-20/16 with 16,384 byte memory and SOLOS module, PT-872 video monitor, RQ-413A cassette recorder, Extended Cassette BASIC, cables and Sol System manual	Factory Assembled/Tested	400700	$2495
4. **Sol System II-A** includes Sol-20/32 with 32,768 byte memory and SOLOS module, PT-872 video monitor, RQ-413A cassette recorder, Extended Cassette BASIC, cables and Sol System manual	Factory Assembled/Tested	400800	$2795
5. **Sol System III-A** includes Sol-20 with 65,536 bytes of memory and BOOTLOAD module, Helios II Model 2 Dual Disk System with Extended Disk BASIC, PT-872 video monitor, cables and Sol System manual	Factory Assembled/Tested	400310	$6495
6. **Sol System IV-A** includes Sol-20 with 65,536 bytes of memory and BOOTLOAD personality module, Helios II Model 4 Quad Disk System (two dual drives) with Extended Disk BASIC, PT-872 video monitor, cables and Sol System manual	Factory Assembled/Tested	400350	$8750

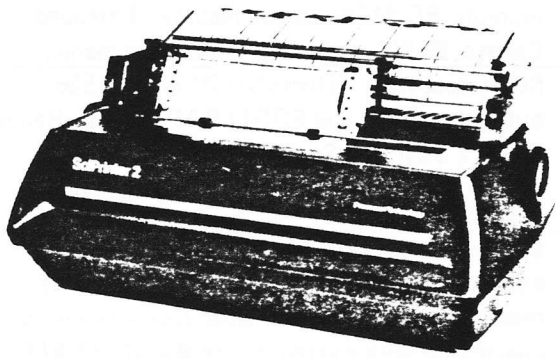


Sol System III-A

HELIOS DISK MEMORY SYSTEMS		Order No.	Price (SUS)
1. **Helios II, Model 2 Dual Disk System** includes dual PerSci 270 floppy disk drive, cabinet, fan, S-100 bus compatible controller, power supply, and system diskette with complete PTDOS software and Extended Disk BASIC	**Factory Assembled/Tested**	**300000**	**$3195**
2. **Helios II, Model 4 Quad Disk System** includes two dual PerSci 270 floppy disk drives, cabinet, fan, S-100 bus compatible controller, power supply, and system diskette with complete PTDOS software and Extended Disk BASIC	**Factory Assembled/Tested**	**304000**	**$5395**



Helios II Models 2 and 4



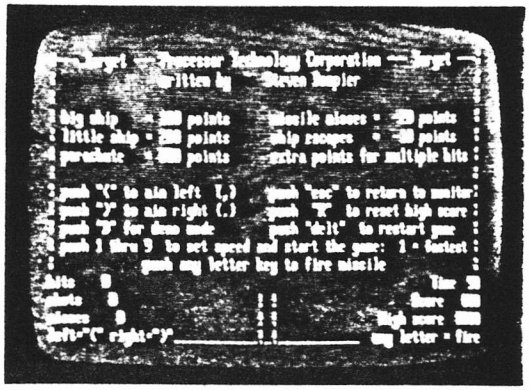
SolPrinter

Sol PRINTERS (available December 1, 1978)			
1. **SolPrinter ® 2,** Word Processing Printer includes Xerox/Diablo Model 1355WP printer mechanism, high-quality metal print wheel in Titan 10 font, carbon film ribbon, RO cover, power supply, friction platen, paper out, cover open and ribbon out detect options, and Sol-HyType interface installed with connecting cable for Sol-20 parallel port	**Assembled/Tested**	**910010**	**$3695**
2. **SolPrinter ® 2-E,** Economy Printer includes Xerox/ Diablo Model 1345A print mechanism, economical plastic print wheel in Courier 10 font, carbon film ribbon, power supply, friction platen, RO cover, paper out, cover open and ribbon out detect options, Sol-HyType interface installed with connecting cable for Sol-20 parallel port	**Assembled/Tested**	**910020**	**$3395**
3. **SolPrinter ® 3,** High speed 200 cps dot matrix printer with RS-232 serial interface (up to 9600 Baud), power supply, RO cover, friction platen, paper out, ribbon out and cover open detect options with connecting cable for Sol-20 serial port. Includes forms tractor	**Assembled/Tested**	**910030**	**$3695**
4. **Forms Tractor** for SolPrinter 2 and 2-E (SolPrinter 3 includes forms tractor)		**910100**	**$250**

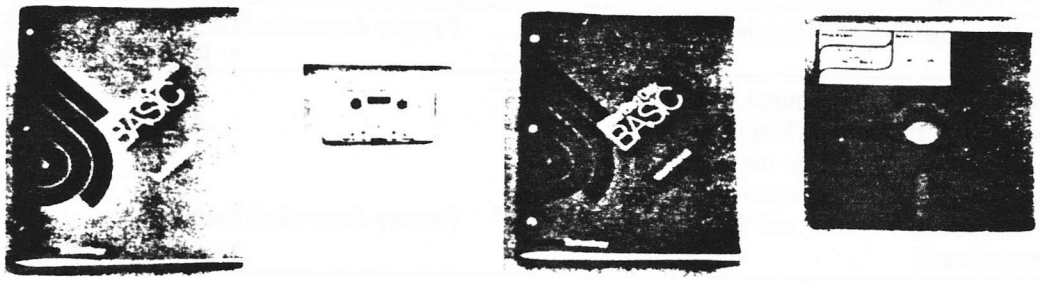
SYSTEM ACCESSORIES		Order No.	Price ($US)
1. **16KRA-1 Memory Module** with 16,384 bytes of Dynamic RAM	**Factory Assembled/Tested**	**214010**	**$429**
2. **32KRA-1 Memory Module** with 32,768 bytes of Dynamic RAM	**Factory Assembled/Tested**	**214020**	**$750**
3. **48KRA-1 Memory Module** with 49,152 bytes of Dynamic RAM (available November 1, 1978)	**Factory Assembled/Tested**	**214030**	**$1095**
4. **64KRA-1 Memory Module** with 65,536 bytes of Dynamic RAM (available November 1, 1978)	**Factory Assembled/Tested**	**214040**	**$1350**
5. **GPM-Sol**, General Purpose Memory Module	**Factory Assembled/Tested** **Kit**	**210000-03** **210000-04**	**$129** **$99**
6. **ALS-8** ROM Resident Assembly Language Operating System with Interpretative Simulator (SIM-1) and Text Editor (TXT-2). For use with GPM-Sol. If purchased together, price includes assembly and testing. (ALS-8, SIM-1, and TXT-2 programs are copyrighted.)	**Factory Assembled/Tested or Kit**	**900014**	**$90**

Sol System I-A

MISCELLANEOUS			
1. **Sol-PC Single Board Computer** with SOLOS personality module	**Factory Assembled/Tested** **Kit**	**101036-01** **101036-02**	**$745** **$575**
2. **PM 2708 Personality Module** without EPROMs	**Factory Assembled/Tested** **Kit**	**107000-01** **107000-03**	**$40** **$30**
3. **SOLOS Personality Module**	**Factory Assembled/Tested** **Kit**	**107000-02** **107000-04**	**$90** **$75**
4. **BOOTLOAD Personality Module**	**Factory Assembled/Tested**	**107015**	**$100**
5. **220/240 Volt Transformer** for all Sol-20 series units	**Kit**	**105034**	**$50**
Note: All Sol-20 series units and Sol Systems are available for 220 Volt, 50Hz operation. Contact factory for pricing and delivery information.			
6. **WWB Wirewrap Prototyping Module**	**Kit**	**211000-02**	**$40**
7. **Helios II Upgrade Option** provides all necessary hardware to upgrade Model 2 Helios Disk Systems to Model 4 including additional dual drive, front bezel, fan, cables and power supply. (Must be installed by factory.)	**Installation**	**304100**	**$2295**
8. **Sol Systems Technical Manual**		**730000**	**$25**
9. **Helios II and PTDOS Technical Manual**		**730009**	**$50**



This video game is included with GAMEPAC 1



Extended BASIC is available in both disk and cassette versions.

	SOFTWARE (including manual)	Minimum Memory Required**	Order No.	Price (SUS)
1.	**Extended Cassette BASIC**	16K	727017	$45.00
2.	**BASIC/5** cassette	10K	727000	$19.50
3.	**New 8080 FOCAL*** cassette	10K	727024	$19.50
4.	**TREK 80** Video Game cassette	8K	727009	$19.50
5.	**GAMEPAC 1** Video Games cassette	4K	727006	$19.50
6.	**GAMEPAC 2** Video Games cassette	4K	727160	$19.50
7.	**Software#1** Resident 8080 Assembler cassette	6K	727022	$19.50
8.	**ALS-8** Resident Assembler, Simulator and Text Editor cassette	12K	727012	$45.00
9.	**Extended Disk BASIC**	32K	727036	$95.00
10.	**PTDOS 1.4 System Disk** (replacement)	32K	727030	$95.00
11.	**Extended Disk FORTRAN**	32K	727100	$95.00
12.	**Cassette PILOT**	16K	727110	$24.50
13.	**8080 CHESS** cassette (available soon)	16K	727150	$19.50
14.	**ASSM**, Advanced 8080 Assembler cassette	10K	727120	$19.50
15.	**DEBUG**, 8080 Cassette Debugger	8K	727130	$19.50
16.	**EDIT**, Advanced 8080 Editor cassette	8K	727140	$19.50

* Registered trademark of Digital Equipment Corporation.
**If cassette CUTER is used, an additional 3K of memory is necessary.

Note:
A SOLOS or CUTER monitor program is necessary to load and run cassette software products.

All Processor Technology software is distributed through authorized dealers or on an individual basis for personal use. Each software package has been copyrighted by Processor Technology and all rights therein are reserved. No license to copy or duplicate is granted with distribution or subsequent resale.

Prices and specifications are subject to change without notice. Prices are in U.S. dollars, apply in U.S.A. only and may be slightly higher elsewhere.

Processor Technology Corporation 7100 Johnson Industrial Drive, Pleasanton, California 94566 (415) 829-2600

Processor Technology
Corporation

7100 Johnson Industrial Drive
Pleasanton, CA 94566

(415) 829-2600
Cable Address PROCTEC

October 12, 1978

Dear Retailer:

Due to temporary circumstances, I have suspended publication of the Personal Computer Retailer for a short period of time. A regular publication schedule will begin again soon.

In the meantime, here is some information you should know about right away!

I. Changes In The Marketing Department . . .

Our Marketing Department has been reorganized to improve its responsiveness to your needs. The following changes are effective as of October 10.

SALES DEPARTMENT

Loren Willey has been appointed Sales Manager, National and International Accounts. He will continue to manage all existing dealer accounts as well as established O.E.M. and international accounts.

Ralph Palsson becomes the Sales Manager for all new national and international accounts and will be working to develop new accounts in the retail, O.E.M., and international markets.

Brett Bullington and Debby Oester are our new Sales Representatives and as such will be handling all of your needs at the factory. Contact either one of them regarding the status of your account, orders, shipping information, etc.

SALES ADMINISTRATION DEPARTMENT

Connie Hawkinson becomes Manager of our newly formed Sales Administration Department. Connie will be in charge of co-ordinating a gamut of internal sales support functions.

ADVERTISING DEPARTMENT

Dottie Wanat is the new Advertising Manager and Sally Turner replaces Dottie as Advertising Assistant.

Elizabeth Fairchild moves from Advertising into a staff position in Marketing.

CUSTOMER SERVICE DEPARTMENT remains unchanged. Continue to contact:
Diane Ascher for Return Authorization requests and
Linda Bilbrey for Warranty Repair and parts requests.

TECHNICAL SUPPORT also remains unchanged, as do technicians' hours.

Processor Technology
Corporation

Page Two . . .

II. The CORONA...Color Graphics Board Announced . . .

You'll be getting inquiries shortly from your customers about the Processor Technology CORONA-1K color graphics board which is described in the cover story of the November issue of Popular Electronics magazine. I've attached a short article by Sally Turner which summarizes the Popular Electronics story in case you cannot get a copy.

The Popular Electronics story says that the product is still in development and will be available through computer stores in kit form for $395. The CORONA is indeed still in the design/development stage. It is currently scheduled to begin production in February and begin shipment in April. That projection assumes no obstacles arise to slow up the schedule. We'll keep you informed as the shipping date becomes more firm.

Meanwhile, you can assure your customers that the product will not be available until next year and that no other store has it either. We are not accepting prepayment or preliminary orders.

We'll be sending you sales literature and additional marketing information when we are closer to delivery.

III. Toll Free Dealer Locator Hotline . . .

Beginning in October, prospective customers can find the dealers nearest them by calling our toll-free hotline. The line will be answered between 8 and 5 Pacific Standard Time on weekdays. A friendly voice will take the callers name and address and give the names and phone numbers of nearby dealers. The names and addresses of callers will then be referred to dealers for followup.

The hotline phone is not connected to our regular switchboard. Calls cannot be transferred to other factory telephones.

The hotline numbers, one for California and one for the rest of continental USA, will appear in Processor Technology's smaller ads such as the one in November Popular Electronics. These ads do not include enough space for a dealer list. We will continue listing dealers in BYTE and Creative Computing.

I've included reprints of two new black and white ads with this letter.

IV. YOU WANT IT WHEN!?

Item	Shipment Begins	Change
8080 Chess cassette	week of 10/23	moved back 4 weeks
GAMEPAC 2	shipment begun	
DEBUG	week of 10/9	moved back 4 weeks
Faceplate for video monitors	week of 10/16	
16KRA-1,32KRA-1	shipment begun	
48KRA-1, 64KRA-1	shipment begun	

Processor Technology
Corporation

V. WordWizard SHOWN IN NEW YORK AND DALLAS . . .

Our new word processing software package has been christened the WordWizard. It will retail for $295 with a 50% Dealer discount. We should begin shipment in the next 30 days.

Diane Ascher from our Customer Service Department demonstrated the program at the New York Personal Computing show (Sept. 15-17) where she divided her time between three dealers' booths--Computer Mart of New Jersey, Computer Mart of New York, and the Computer Corner. Sales Rep. Debby Oester went to the Dallas show (Sept. 30-Oct. 1) where she presented the same demonstration at two dealer booths--Computer Port and Interactive Computers. (Debby, by the way, won a bottle of champagne for thinking up the name "WordWizard.")

Overall, supporting dealers at shows in this way seems to be very successful both for us and for you. We plan to do more of this in the future.

VI. NKRA . . .

Within the next week you will be receiving update information on the new NKRAs (16KRA-1, 32KRA-1, 48KRA-1 and 64KRA-1) which will enable you to make them compatible with all S-100 bus computer.

CHRISTMAS IN HAWAII . . .

Don't forget to keep sending in your sales verification forms for the "Christmas in Hawaii" contest. Thirty-nine salespeople have signed up so far.

Best regards,

Dottie Wanat

Dottie Wanat
ADVERTISING DEPARTMENT

CORONA: Color Graphics is Coming

by Sally Turner

The November Popular Electronics will feature a cover story about Processor Technology's innovative high resolution full color graphics accessory for microcomputers. The CORONA System is a graphic display memory capable of providing 256 different colors in a 256-by-256 display with both graphics and alphanumerics. These are mixed and overlaid to interleave graphic and text information anywhere desired on the screen.

CORONA is fully bus compatible with the Sol-20 microcomputer and VDM-1 video display module but can be modified to fit other S-100 bus formats. It operates both for color-picture storage or conventional black and white computer memory (256 shades of grey).

The CORONA's circuitry mounts on three printed circuit boards. Two boards plug directly into the S-100 bus of the computer. In Sol, only one slot, the "top slot", is needed. The third board is packaged externally with the analog circuitry and video connectors.

Memory access by the computer is handled in one of two modes: "bit mode", which simplifies interfacing with BASIC and FORTRAN, and allows direct access to individual points on the screen; and "byte mode" used for high speed data transfers from the CPU to the CORONA. Its color-video encoder transforms the red, green, blue and luminance data from the color map into NTSC or RGB color signals.

In addition to opening new visual frontiers for video games, CORONA is a powerful tool for business, artistic, scientific and educational applications. Full animation, vivid graphics, and poster-like displays can transform any presentation into a colorful multi-media event.

Many of your customers will undoubtedly be interested in this dynamic new product.

THE PERSONAL COMPUTER RETAILER

Vol.2 No.10 Published by Processor Technology Corporation November 20, 1978
7100 Johnson Drive, Pleasanton, CA 94566

Portland attorneys view WordWizard

Byte Shop of Portland introduced three new systems to an invited group of 130 attorneys and accountants at a special show on November 7. Debby Oester from our Sales Department represented Processor Technology with a demonstration of the WordWizard Electronic Typing System.

WordWizard-- Star of the Show

WordWizard turned out to be the star of the show. According to Debby, some of the guests already owned word processing systems from companies like Vydec, Word-Stream and Lexitron. They were very impressed with the capabilities of WordWizard compared to these more expensive systems. They also liked the flexibility of the Sol System for many other business applications.

The show was held in a banquet room of the Travel Lodge Inn from 10 in the morning to 7 at night. A hostess/receptionist greeted all prospective customers at the door. She made name tags and invited the guests to sign the register. Three systems were in operation with Debby and two Byte Shop salespeople always on hand to make presentations.

Marketing Staff looks on as Debby Oester demonstrates WordWizard.

Who's Who in the Marketing Dept.
Back Row, L to R: Bob Marsh, Bob Groppo, John Chong, Loren Willey, Don Regaller, Brett Bullington, Liz Fairchild, Dee Wunderlich, Mike Mills, Ralph Palsson
Front Row, L to R: Sally Turner, Freddie Jacobowitz, Denise Faria, Dottie Wanat, Linda Bilbrey, Connie Hawkinson, Diane Ascher and Debby Oester

Pat Terrell, co-owner of the store, was more than pleased with the successful event. He expects four to six word processing system sales to result directly from the show. His Seattle store will host a similar show very soon. He also plans variations on the show for insurance people and doctors.

Asked what he will do differently at the Seattle show, Pat said he wants to handle the people more efficiently. "Frequently there

(Continued on page 5)

You want it when?!

Item	Shipment Begins	Change
8080 CHESS Cassette	Shipment begun	
DEBUG	Week of November 27	Moved back 7 weeks
Faceplate for video monitors	Shipment begun	
SolPrinters 2 and 3	Week of December 4	
WordWizard	Week of November 27	

FinanceAmerica user happy with service

by Sally Turner

To get more information about the FinanceAmerica flooring program, I contacted a local appliance store owner who uses it.

Frank Lozano of Frank's TV & Appliances in Pleasanton told me, "No other finance company has ever served us so well. FinanceAmerica works WITH you. They eagerly pitch-in to help us with displays and special promotions."

Frank especially likes the flexibility of Finance America compared to other flooring companies. "They don't hold you to the bottom line," says Frank. "When we've needed extra merchandise for special occasions like Christmas, they stretch the budget to accomodate us."

When asked if he minded the monthly inventory check, Frank said, "Not at all. In fact, it's helped me keep my own books better. Besides, the benefits are so good, they far outweigh any disadvantages. All in all, the service offers more than I expected."

Frank generously consented to answer any questions you might have if you'd like to phone him at (415) 846-5505.

Whodunnit

Editor.....Dottie Wanat

Stand-in Editor...Sally Turner

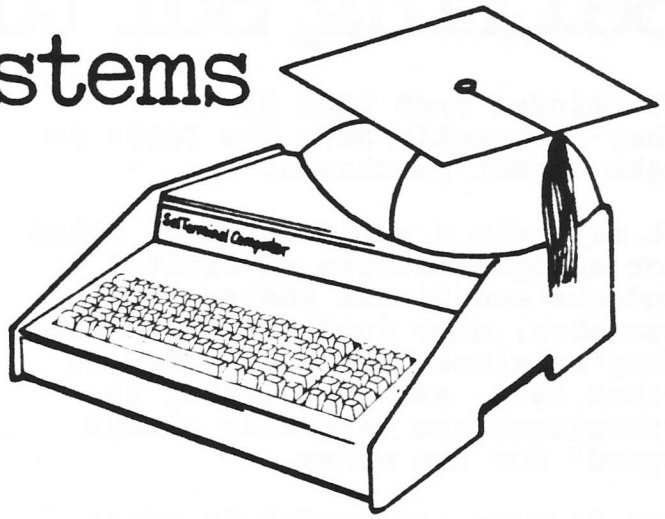
Assistant Stand-in Editor.....Elizabeth Fairchild

Graphics.....Lois Esser
Ray Oleson

General Input.....Diane Ascher
Connie Hawkinson
Rick Meyer
Terry Walters

Copyright © 1978,
Processor Technology Corporation.
All rights reserved.

Selling Sol Systems to educators



by Ralph Palsson

As you are aware, microcomputers are becoming increasingly popular as an educational tool in schools and universities at all levels. Some estimates predict that 25% of the dollars that will be spent on small computers in the next few years will come from the education segment.

How can you insure having a fair edge on this lucrative market? Virtually all public institutions purchase equipment on the basis of competitive bids. I'd like to discuss some aspects of bidding and a few points that will be helpful in winning contracts.

REGISTER YOUR PRODUCTS AND SERVICES

Many dealers shy away from proposing bids and contracts because of the red tape involved. I argue that this effort is worthwhile. Many times one winning bid will lead to the subsequent sale of scores of systems.

For bidding and contract proposals, the purchasing agencies in your locality must be aware of your products and services; so, you need to register with your state's (and neighboring state's) "Division of Purchasing" or the "Department of Education" in the state capitol. You will be sent an Application Form and Commodity Listing to indicate what products and services you're qualified to bid on. Needless to say, you also become eligible to bid on appropriate computer systems for other state agencies.

One form of bid you may receive is a Request for Quotation. Typically, RFQ will ask specific quote, delivery information, terms, etc. on an approved system. You are held to your submitted quote and may be issued a purchase order on the same document.

Another form you may encounter is a Request for Proposal; this is a contract proposal, used mainly for Multiple Systems Purchases and high-dollar volume potential, also for open number of systems over a given time period, e.g. one year.

SECRETS OF SUCCESSFUL BIDS

It is important to comply with the administrative requirements contained in bid specifications. For complicated proposals you might want to hire a professional contract writer. If you have specific questions on a proposal, phone me, I'll be glad to help you.

You can find out who your competition is and what they're offering from the bid-generating agency upon request. This will increase your chance of winning the contract.

I suggest you register with individual school districts and colleges.

(Continued on page 4)

Selling Sol to educators

(Continued from page 3)
They frequently have the funds to make direct purchases.

It may take a while to close sales for schools because of limited budgets and fiscal years. But remember, once funds are allocated, they're almost never diverted to other uses, and educational institutions are virtually always "good" for the money.

Sol Systems are useful in educational applications in a variety of rolls from classroom to administrative. They're useful in computer aided instruction from mathematics to bilingual learning.

Because of Processor Technology's comprehensive and high quality product line, we expect to hold a large share of this market.

We're fully committed to supporting the education market and your place in it. So, let's work together for a profitable future.

Ralph Palsson, educational marketing specialist for Processor Technology, has been writing sucessful bids for two years.

COMING IN JANUARY

Customized sales lit kits

We believe pictures generate excitement which in turn sparks enthusiasm...an effect that snowballs when sales literature is used right. We want you to have the tools that enable you to make well planned sales presentations. So, we designed a colorful 4 page folder and an array of data sheets describing our products, which you'll receive in January. By adding your own store's literature, you can customize to meet individual needs.

In the past sales literature consisted of brochures aimed at technical users. The new data sheets address a broader audience, particularly the business user.

The first 3 data sheets will be sent to you in early December. They consist of 2 pages on the nKRA memory boards, 4 pages on WordWizard electronic typing, 4 pages on Extended BASIC.

Our colorful 4 page folder will be a brochure explaining Processor Technology as a company; who we are, our history and our marketing intentions.

We'll continue to use the Sol and Helios brochures temporarily. The Helios brochure is being revised to include explanations and pictures of the Helios II model 4.

We developed this colorful set of materials to enable you to compliment and tie together sales promotion efforts.

Attorneys view WordWizard

(Continued from page 1)
were almost twenty people in the room at once, all trying to get their questions answered. One solution is to schedule demonstrations at definite times. Another approach is to give a fairly short formal demonstration, then qualify the prospects for a followup appointment with a complete demonstration later."

Store manager Tom Rich was responsible for most of the organization and planning of the show. He reported that total costs for the entire production were under $600 including the room rental and all advertising costs.

Primary advertising for the show was 500 formal, printed invitations to attorneys in the Portland area. In addition, he ran a small ad in the local paper for three days before the show.

As Tom pointed out, "A turnout of 130 people was excellent. We could easily have had more systems with more salespeople demonstrating them. I think Pat's idea of scheduled presentations would be helpful. It's also important to have hands-on time so people can see how easy the systems are to use."

"Another thing, I'd have the hostess do some qualification of the visitors. If she knew what their specific interests were, she could cue them to a particular system."

"And one last thing, I liked Debby's suggestion that we should have a quiet room, a closing room, where we could take people to discuss terms and sign papers. If we had had a room like that, we might have sold ten systems right at the show."

Tom Rich will be happy to talk with any dealer planning a similar event.

Participants in the August Technical Seminars received this attractive certificate!

All new ACCESS coming January 1

Diane Ascher from our Customer Service Department, recently volunteered (Army Style) to take over as editor of ACCESS magazine.

Her first official decision was to establish a regular bi-monthly schedule for the magazine. Her plans also call for some major format changes.

"My goal is to make ACCESS a magazine that appeals to both the technical and non-technical customer," says Diane. "Customer contributed articles will be featured, as well as regular articles about hardware and software updates. We are particularly interested in applications software. And, I encourage letters-to-the-editor," says Diane.

All previous subscribers will get the next three issues as part of their original subscription.

So here's your chance! See your name in print or encourage your customer to submit articles, Diane awaits your contributions.

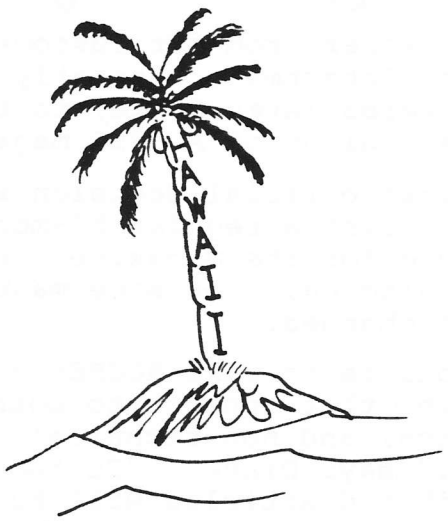
Helios II user's manual well received by dealers

"The new manual has a good Table of Contents so you can find what you're looking for right away," commented Paul Matzke of the Madison Computer Store. "The style makes it easy to understand, and I like the way it looks, too."

Our new Helios II User's Manual uses simple language to explain how to operate the Helios. All theoretical and service-related information is left out. We developed this manual to meet the needs of the business user who typically has no technical knowledge and depends entirely on the dealer to test and repair his equipment.

Dealers should be receiving the new PTDOS User's Manual soon, to be used in conjunction with the Helios User's Manual. It describes the operating system for the Helios.

Plans for the future call for both a User's Manual and a Technical Service Manual for each of our hardware products. The Sol is currently undergoing this change.



We know you're all anxious to find out who won the Christmas in Hawaii contest. We're going to tell you. But, all the verification forms aren't in yet. So, ...you'll just have to wait for the next issue. Keep Smilin'.

New price list "Best one yet"

This was the frequent reply from dealers and sales staff when asked how they liked our new Price List.

"It sure helps to have such a professional Price List," reported Gerald Wright of the Digital Deli (Mountain View, CA). "Before, I had to circle appropriate items for our customers. Now they can understand it and make selections themselves. The pictures really helped to tie it all together."

He liked the folder idea, too.

Several dealers commented that the new layout and pictures make this price list especially useful as a sales tool, unlike other price lists.

THE PERSONAL COMPUTER RETAILER

Vol.2 No.11 Published by Processor Technology Corporation December 18, 1978
7100 Johnson Drive, Pleasanton, CA 94566

New Jersey dealer ready for Hawaii

The news you've all been waiting for! Processor Technology proudly announces the 1978 Winner of the Christmas in Hawaii Contest: JOEL SHUSTERMAN, COMPUTER EMPORIUM, Cherry Hill, New Jersey. Congratulations, Joel! (Mind if we send along your friendly roving reporter for an on-the-spot interview?)

Second prize, a Sony Trinitron Color TV goes to HARRIET SHAIR, COMPUTER CORNER, White Plains, NY.

Third prize winners of the Polaroid Pronto Camera Outfits are: BILL BLOCK, INTERACTIVE COMPUTERS, Houston, TX; GRAHAM MEAKINS, BASIC COMPUTER GROUP, Vancouver, B.C.; LARRY STEIN, COMPUTER MART OF NEW JERSEY, Iselin, NJ; and ROBERT VIPOND, COMPUTER MAGIC, Modesto, CA.

Congratulations to all these winners, and a special Thank You to everyone who participated in this event.

Fairchild named new sales manager

"My objective is to enhance the working relationship between dealers and our Sales Department. I intend to clarify certain policies," says Elizabeth Fairchild, our new Sales Manager. "For example, putting our OEM sales program in writing so dealers will be able to work with us in pursuing OEM accounts."

As Processor Technology becomes more involved in the business market, Elizabeth plans to give dealers more sales support similar to the Word-Wizard Marketing Overview and product training seminars.

Elizabeth has a background in elctronics, and before joining Processor Technology in April 1976, she worked in retail sales.

Prior to promotion to Sales Manager, Elizabeth headed up our Advertising Department and was largely responsible for helping to develop our public image.

We know you'll enjoy working with Elizabeth and her staff. They're a hard working team committed to getting results!

Special Holiday Issue

You want it when?!

Item	Shipment Begins
DEBUG	Shipment begun
SolPrinter 2, 2E, and 3	Shipment begun
WordWizard	Shipment begun

Around the country in fifteen days

Twenty thousand miles and fifteen days later we had managed to visit many of you and give your staff a personal introduction to WordWizard and the advertising program developed to promote it. Your response was more than enthusiastic. It was obvious you shared our excitement for WordWizard and its market potential. As Ben Bennett from Roms 'N Rams in Raleigh, North Carolina put it, "The market is ready for a low-cost system which is as easy to learn and as simple to use as WordWizard. We already have customers ready to buy."

Again, Gary Ingram, Mike Mills, Ralph Palsson and I would like to thank all of you for making us feel so welcome. Although we succeeded in totally exhausting ourselves, we all agree that we were glad for the opportunity to visit your stores and get better acquainted with each of you.

We appreciate the time you took from your hectic schedules to share with us your comments and suggestions on WordWizard as well as your thoughts and ideas about Processor Technology and the industry in general. Your input is extremely valuable. The information exchanged will help us in formulating future policies and programs--policies and programs which will, hopefully, enable all of us to work together in the most effective and profitable manner.

From myself and the staff of the Personal Computer Retailer, HAPPY HOLIDAYS to all of you and your families!

Whodunnit

Editor.....Dottie Wanat

Asst. Editor.....Sally Turner

Graphics.....Lois Esser
Ray Oleson

General Input.....Diane Ascher
Elizabeth Fairchild
Connie Hawkinson

Copyright © 1978,
Processor Technology Corporation.
All rights reserved.

What you've wanted to know about Sol Printers

...but didn't know whom to ask

Q. Why can't I put a plastic wheel into a metal wheel machine and vice-versa?

A. The mechanism is completely different in these two machines. Using the wrong print wheel will damage the wheel and void the warranty.

Q. What's the difference between the metal and plastic wheels?

A. Metal print wheels are actually a very hard plastic that has been metal-plated. Metal wheels last 4 times as long as plastic and are available in different fonts.

Q. How can I tell when the wheel is beginning to wear out?

A. Serifs, periods, commas, hyphens and underscores wear out first. You can spot-check these symbols periodically.

Q. When will I receive delivery?

A. Delivery will be limited this quarter. We will get the major part of your scheduled order to you. In some cases, the SolPrinters are beige rather than black. We expect the delivery situation to improve next quarter.

Q. Where do I get supplies such as ribbons and additional print wheels?

A. These are available from Processor Technology as well as many other sources. We encourage dealers to find a local source.

Q. What about maintenance contracts from Diablo?

A. Diablo Warranty Extension contracts are $135 (SolPrinter 2 & 2E) and $110 (SolPrinter 3) and available to the end-user at time of purchase. Five maintenance depots exist around the U.S. We highly recommend this contract to you since without one, Diablo will charge you $80/hr. for the first hour and $40/hr. thereafter for service.

(Continued on page 4)

Software Sentre

We're back again with Software Sentre. Here's some new software available for Helios:

CP/M will soon be available for Helios from: Lifeboat Associates, 164 W. 83rd St., New York, NY 10024. (212) 580-0082.

General ledger, Accounts Receivable, Accounts Payable, Payroll with Job Accounting, Inventory Control and Order Entry...a complete package for Sol Systems III and IV available now with extensive dealer training. Contact Dick Babb, MicroSun, 2989 North Main St., Walnut Creek, CA 94649. (415) 933-6252.

...about Sol Printers

(Continued from page 3)

Model	SolPrinter 2	SolPrinter 2E	SolPrinter 3
Type	Word processing (letter & reproduction quality) Metal daisywheel	Word processing (letter quality) Plastic daisywheel	High-speed dot matrix (drafts & reports)
Nominal speed	33 cps (480 wpm)	37 cps (540 wpm)	200 cps (2400 wpm)
Interface included	Parallel	Parallel	Serial (RS-232) (110-9600 baud)
Model # of Diablo Print Mechanism	1355 WP	1345 A	2300
Forms Tractor	Optional ($250)	Optional ($250)	Included
Print Wheel included	Titan 10 - 96 characters (others will be available in future as option)	Courier 10 (others available as option)	7 x 9 dot matrix (both upper & lower case with descenders) - equivalent to 10-pitch type
Ribbon	Carbon film or cloth	Carbon film or cloth	self-re-inking ribbon
Carbon Copies	Up to 12 plus original	Up to 12 plus original	Up to 5 plus original
Print Wheel life expectancy	Approximately 16 million characters (30-50 million characters possible)	Approximately 4 million characters	N/A
Forms Width	15 inches max.	15 inches max.	15 inches max.
Carriage Return time	300 ms max.	300 ms max.	350 ms max.
Character Set	96 characters	88-96 characters	96 characters
Bidirectional Printing	Yes	Yes	Yes
High Speed Vertical and Horizontal Tabbing	Yes	Yes	Yes
Automatic Tabbing for Multiple Spaces	Yes	Yes	Yes

Return Authorization Policy now in full effect

The Return Authorization lets us know in advance that you are returning a unit. It enables us to schedule technicians, order parts, issue credit, etc., more efficiently. So, to avoid unnecessary delays, remember these procedures:

1. Notify Customer Service by phone or mail what equipment you're sending in for repair. Give the serial number and the symptoms of the problem.
2. Use the Return Authorization number as a packing label on the outside of the box.
3. If more than one item is covered by a single Return Authorization slip, but the items are being shipped separately, please do the following:

A. Indicate on the Return Authorization slip which of the items is being shipped under separate cover;
B. Attach the Return Authorization slip to the outside of either of the boxes;
C. On the outside of the other box, clearly indicate the Return Authorization number covering the return. OR, better yet, attach a copy of the original Return Authorization slip.

Memory Board upgrade program

If you have customers who would like to upgrade their system but don't have the additional memory or the cost is too high, here's their chance to purchase one of our new state-of-the-art memory modules.

For a limited time only (offer good through January 30, 1979) we are offering a special program which allows a customer to trade-in their present memory board(s) for credit toward the purchase of one of our new, more powerful memory modules, 32KRA-1, 48KRA-1, and 64KRA-1.

The benefits of this program are two-fold:

THE DEALER

This provides you with a good opportunity to contact old customers and fill them in on new products which are currently available:

* New energy and space conserving memory modules
* New software application programs such as WordWizard

THE CUSTOMER

Since the trade-in values are high, the customer gets an upgraded system at minimal cost. New applications programs (such as WordWizard) can be easily run on his upgraded system. And the nKRA family of memory modules offers:

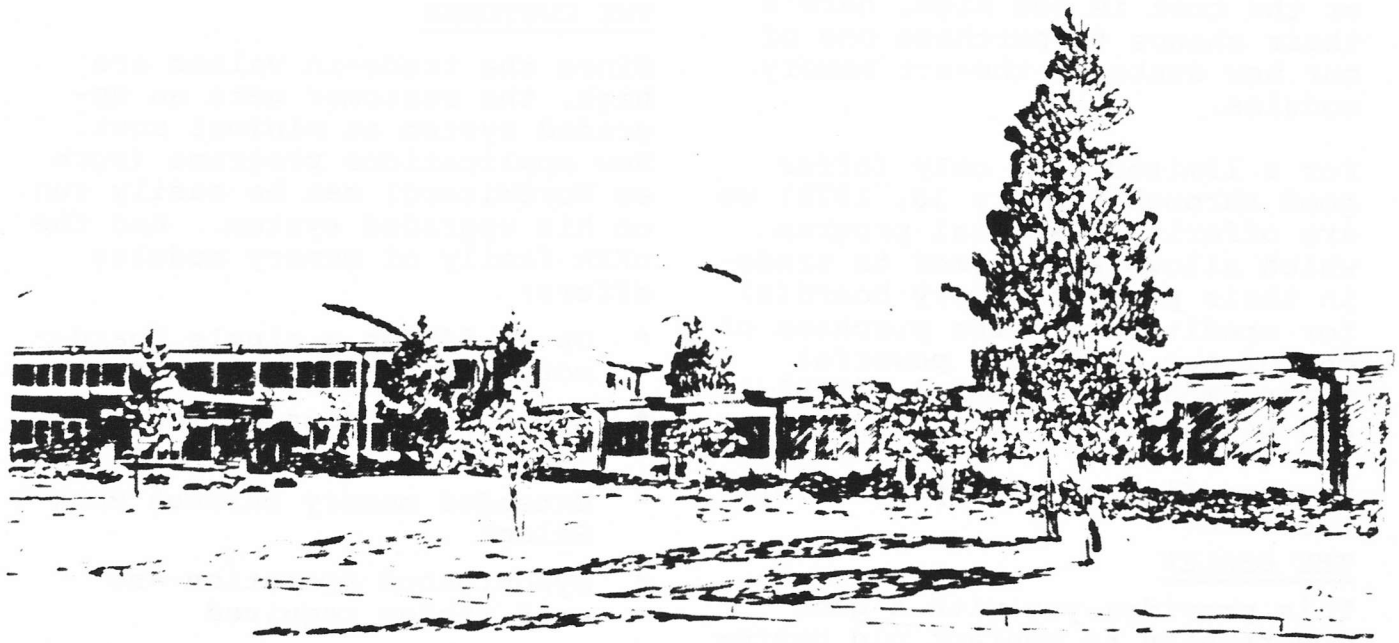
* Up to 64K on a single board--more usuable backplane slots
* Lower power consumption and less heat
* Extended memory through Bank Select
* Synchronous operation--no wait states required

Specific details of the program may have already reached you. If not, you will be receiving them very shortly. So, get ready to sell those boards!

The staff at
Processor Technology
wish all of you

happy holidays

filled with good old-fashioned cheer



THE PERSONAL COMPUTER RETAILER

Vol.3 No.1 Published by Processor Technology Corporation January 26, 1979
7100 Johnson Drive, Pleasanton, CA 94566

The next in our series of technical seminars is scheduled for February 12-16. It is especially designed for the technician who has not yet attended a Processor Technology tech seminar. Since both hardware and PTDOS software will be discussed, you may want to send two technicians. Enrollment is limited to ten people, so get those reservations in now. Phone your dealer representative, Debby Oester or Brett Bullington.

In this issue...

The next in our series of technical seminars..see page 1

An updated PTDOS 1.5 has been released.....page 2

Businessmen will get a chance to see WordWizard through ads placed just for them.....page 2

Enthusiasm for WordWizard continues.....page 3

More sales literature is coming your way.....page 3

DAP has some new business software; see Software Sentre.....page 4

Technical seminar scheduled

The seminar will be conducted as follows:

HARDWARE

Monday: System III trouble-shooting and interfacing

Tuesday: Morning--Helios service and maintenance
Afternoon--SolPrinters

Wednesday: SolPrinters (continued)--differences and board swapping.

SOFTWARE

Thursday: Introduction to PTDOS 1.5

Friday: PTDOS System interfacing and device drivers

Be sure to bring a PTDOS manual and two blank diskettes.

Prerequisites for attendance:

Hardware technicians must be familiar with the Sol and Helios manuals and the nKRA theory of operation (theory will not be covered during the seminar). Technicians should have an AA or the equivalent in digital electronics or computer maintenance plus at least six months experience with microcomputers.

Software attendees should be familiar with the 8080 assembly language and should be able to read and understand the code. Each should have primary responsibility for writing and/or maintaining applications software and/or drivers.

Technicians attending the seminar will be responsible for their own transportation and motel accomodations. We recommend the California 6 Motel in Pleasanton at (415) 462-2626, or Howard Johnson's in Dublin at (415) 828-7750. Both are close to the plant.

WordWizard enthusiasm running high

by Sally Turner

Here we are embarking on another year and perhaps the most exciting quarter we've yet experienced. We're well into the most comprehensive sales campaign we've ever undertaken.

It began with the whirlwind cross-country visits by Gary Ingram, Dottie Wanat, Mike Mills and Ralph Palsson. Various dealers had the chance to preview WordWizard and hear all about the marketing plan to introduce it.

When asked what he thought of the visit, Eri Golembo (Computer Mart of New Jersey) said, "In this day and age it's important to have a close relationship with the vendor in order to do the best job. I was very pleased with the visit. In fact, we sold a WordWizard the day after Dottie and Gary were here. An attorney walked in looking for a word processing system. We were 'hot' from the visit. It was a natural."

Following the cross-country visit, the first of our advertising campaigns began in the WALL STREET JOURNAL. Responses have been good. When a phone call is received, the customer is referred to his local dealer. The caller's name and address is also recorded, and then this information is sent to you along with all the inquiries received from the mail-in coupons. If a customer requests a demonstration, we phone the local dealer immediately. In all cases, we immediately send the prospective customer a WordWizard data sheet.

The Monday Machine co-op ad mats were designed to help you create an awareness of both you and WordWizard in your local community. When used in conjunction with the WALL STREET JOURNAL ad dates, the impact has been good. Be sure to take full advantage of your co-op advertising allowance and run these ads. Remember, we pay 60% of those costs.

The marketing overview which you received with your first WordWizard shipment provides you with a general overview of the system, its position in the market place, and tips (based on our experiences) on how to do an effective

(Continued on page 4)

More sales literature coming

The WordWizard, nKRA, and Extended BASIC data sheets which you recently received are the first in a series of new product literature planned for publication during this year.

These data sheets are designed to be used in conjunction with, or independently from, the colorful new six-page brochure which is currently in production. You should receive these brochures within the next month (weather conditions permitting the photography work to be completed).

The brochure is directed at a very general audience. This allows you to custom-tailor your sales literature package to the needs of a specific customer. All you have to do is insert the appropriate data sheet or sheets into the flap on the last page. Or, if a single data sheet will suffice, you can simply give that to the customer since each sheet is self-explanatory.

Software Sentre

DAP Systems is releasing the Osborne and Associates accounting package on Helios II diskettes. Representing five man-years of development by Osborne personnel, the Helios version is marketed exclusively through qualified dealers and consultants.

The package consists of three modules: payroll with cost accounting, accounts payable/receivable, and general ledger. Each can operate independently or in conjunction with the others, sharing data files where appropriate.

The dealer price for the diskettes is $100 per module. After making the initial purchase, dealers may copy the diskettes for resale for no additional royalty.

The documentation was prepared by Osborne and is available from them under their normal bookseller's terms. It includes file layouts, file maintenance programs, data entry instructions, screen display formats, and report printout formats. In addition, the individual programs are documented with a user's manual, flowcharts, and a narrative description. The package runs under CP/M and CBASIC, available for the Helios II from Lifeboat Associates (164 West 83rd St., New York, NY 10024).

The first module, payroll with cost accounting, is now being delivered. Accounts payable/receivable and general ledger are scheduled for release in February and March respectively. Prepayment is required with orders.

DAP Systems is located at 3901 Victoria Lane, Midlothian, VA 23113. (NOTE: This product has not yet been tested by Processor Technology.)

WordWizard

(Continued from page 3)
demonstration. Your early responses indicate that this too has proven to be a very useful sales tool.

In fact, from all indications, it appears that all of these marketing efforts are very effective. Your enthusiasm is high. I believe Ed Hammontree (Computer Port, Arlington, TX) put it best when he said, "Y'all started right and followed through; finest promotion we've ever seen in the computer business." And, I might add, we intend to keep it that way.

Sol saga

As told to Phil Dorcas (Computer Port, Arlington, TX): Last June, John Cade (a Computer Port customer), his two sons, and a Sol System III took a 3,000 mile, cross-country trip in a Winnebago. Weather conditions varied from 110 degrees in the desert to below freezing in the Colorado Mountains.

Hooked to a 110 volt auxillary power unit, Sol ran all day, everyday... the kids played games, and John did his work long into the night.

While crossing the hot desert during a sandstorm, the air-conditioner went out. The boys opened the windows in the van...sand went flying everywhere, and John...well, he ran off the road and the Helios disk drive fell to the floor.

John and his sons picked up the unit, shook out the sand, and plugged it in. It worked! "And to this day it still works perfectly." THE MORAL OF THIS STORY: Sol Systems are even better than they're cracked up to be!

Monday Machine ad's aiming at business

Again this quarter we have planned an aggressive advertising campaign for the WordWizard aimed at the business end-user. The "Introducing the Monday Machine" ad will appear in the following publications:

THE WALL STREET JOURNAL (East and West Coast editions)
THE OFFICE (a special issue on word processing systems)
THE AMERICAN WAY
SMALL BUSINESS COMPUTERS (a special issue on word processing systems)
BYTE.

As before, you will receive all sales leads generated from these ads either through our Dealer Locator Hotline or from the mail-in coupon request.

When a customer calls or sends in a coupon, he receives a WordWizard data sheet from us and, of course, the name and address of his local dealer. BUT, don't wait for them to contact you! Remember, these leads are HOT so be sure to follow-up on them as fast as you can.

Updated PTDOS 1.5 now being shipped

The new release of our disk operating system, PTDOS 1.5, is being shipped with all Helios orders. All reported bugs in PTDOS 1.4 have been fixed and a number of useful features added. The changes are summarized in the preface of the second edition of the PTDOS User's Manual.

Customers using PTDOS 1.4 will undoubtedly want to switch to 1.5. We encourage you to offer the new release at no charge to customers. Anyone switching, however, should have the second edition of the User's Manual, which we suggest you sell for $30.

Hawaiian holiday begins for winners

With great eagerness, Joel and Sheryl Shusterman embarked on their seven day trip to the beautiful island of Maui in Hawaii. (And ready they were since both their children spent the preceeding weeks with chickenpox.)

They're staying at the Kaanapali Beach Hotel on the strip. Since neither one had ever been to Hawaii, their excitement and anticipation were high. But we'll let them tell you all about it in our next issue. (In case you hadn't guessed, your friendly, roving reporter didn't get to go along despite all her protests!)

Whodunnit

Editor.....Dottie Wanat
Asst. Editor.....Sally Turner
Graphics.....Lois Esser
Ray Oleson
General Input.....Mike Mills
Diane Ascher
Connie Hawkinson

Copyright © 1979,
Processor Technology Corporation.
All rights reserved.

YOUR PROCESSOR TECHNOLOGY 90 DAY LIMITED WARRANTY

1. **If you are not satisfied** with the service, warranty or otherwise, provided by Processor Technology or its factory authorized warranty centers, write directly to our Customer Service Manager, 7100 Johnson Dr., Pleasanton, California 94566. We will make sure your problem receives our immediate and personal attention.

2. **Processor Technology warrants** each of its Products to be free from defects in materials and/or workmanship for a period of ninety (90) days from the date the product was originally purchased. This warranty applies to the original purchaser only and is limited in all cases to repair or replacement of the defective product.

4. **Please carefully note** that this is a two-way agreement. We promise to make repairs or replacements as stated above, but you agree that except for our obligation to make good on this promise we shall not be responsible for any expense or inconvenience which you might incur or experience with respect to the product, and which may relate to any incidential or consequential damages, nor shall we be liable for defects, damage or failures caused by unreasonable use, accident or abuse, including failure to provide reasonable and necessary maintenance after the products have been delivered to you. Any product which has been mechanically, electrically or electronically altered without specific written authorization by Processor Technology is excluded from this warranty.

5. **You are responsible** for the transportation cost to and from the Processor Technology authorized Warranty Repair Center. If you need to ship the product to the Warranty Repair Center you must use the original carton or a suitable replacement to prevent shipping damage. We recommend you store the original cartons for this possibility. To expedite Warranty service you must return the Warranty Registration Card to the Processor Technology Customer Service Manager, at 7100 Johnson Drive, Pleasanton, CA 94566.

6. **No other person or entity** is authorized to give any other warranties or to assume any other liabilities on Processor Technology's behalf unless made or agreed to in writing by Processor Technology, and no other person or entity is authorized to give any warranties or to assume any liabilities on the seller's behalf unless made or agreed to in writing by the seller.

7. **INCIDENTAL OR CONSEQUENTIAL DAMAGES** SUCH AS TELEPHONE CALLS, LOSS OF TIME, INCONVENIENCE, OR COMMERCIAL LOSS, ARE NOT COVERED UNDER THIS WARRANTY.

8. **Any Processor Technology product** purchased outside of the United States is not covered by this warranty.

9. **The foregoing paragraphs** constitute Processor Technology's entire warranty with respect to any product purchased or operated for personal use, as opposed to commercial use.

10. **ANY IMPLIED WARRANTIES,** INCLUDING THOSE OF MERCANTABILITY OF FITNESS, ARE LIMITED TO THE NINETY (90) DAY DURATION OF THIS WRITTEN WARRANTY.

11. **SOME STATES DO NOT ALLOW** THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, OR LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.

THIS WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING WITHOUT LIMITATION WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICLUAR PURPOSE, ALL OTHER REPRESENTATIONS TO THE FIRST USER PURCHASER, AND ALL OTHER OBLIGATIONS OR LIABILITIES, INCLUDING LIABILITY FOR INCIDENTAL AND CONSEQUENTIAL DAMAGES, ON THE PART OF PROCESSOR TECHNOLOGY OR THE SELLER.

Revised 2/6/79

Processor Technology Corporation
7100 Johnson Drive, Pleasanton, California 94566 (415) 829-2600

THE PERSONAL COMPUTER RETAILER

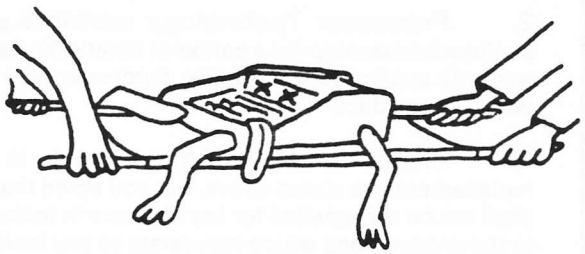
Vol.3 No.3 Published by Processor Technology Corporation April 18, 1979
7100 Johnson Drive, Pleasanton, CA 94546

New hardware and software debut at NCC

An entire line of new Processor Technology hardware and software products will be shown to the public for the first time at the National Computer Conference in New York, June 4-7.

Dealers who attended our April seminar had the opportunity to preview many of these new products.

In this issue...



Our new Quality Assurance Policy is now in effect....see page 2

An update disk for WordWizard and FORTRAN is now available.....page 5

NCC...NCC...and more NCC.....starting on page 1

HARDWARE

In the hardware department we'll be showing the all new SolDisk 1, a double-density, dual mini-floppy packaged in a compact enclosure. It has a storage capacity of 350K bytes. SolDisk 3 is a hard disk packaged in a rack-mountable enclosure. It stores up to 28 Megabytes of data.

The controller hardware for each of these systems is divided into two modules. One module resides in the processor on the S-100 bus backplane and can communicate with up to seven device controllers. The device controller resides in the first disk drive enclosure and allows for the connection of up to three additional drives before requiring another controller device.

SOFTWARE

PTDOS 2 is a modified and enhanced version of PTDOS that will support the new hardware.

In the area of Business Applications software, there are programs to manage mailing lists, do general ledger, accounts payable and accounts receivable. All packages are completely integrated and compatible with each other as well as with WordWizard. They may be purchased individually or as an entire package.

MailMaster, a list management system, provides the end-users with

(Continued on page 3)

NewsWorthy Notes...

Joel Shusterman, Computer Emporium, Cherry Hill, New Jersey, sold 30 Sol Systems to one company as the result of his participation in trade shows. "The sale resulted from the Mini-Micro Show in Philadelphia in May of '78 and the Office Products & Environment/ Sales Expo '79 at the Philadelphia Civic Center in December," reported Joel's proud wife, Cheryl.

Terry Batt, Midwest Microcomputers, Lombard, Illinois, recently exhibited at the Biz Services and Equipment Show in Chicago. He got follow-up requests from 83 companies. To handle all the requests, he's holding Open House in his store on weekends.

Roms 'N' Rams, Raleigh, North Carolina, has a new name-- Bennett-Stiles Computers. They also have a new full-time outside salesman, Jerry Thomas, who coordinated their Business Applications Seminar.

Due to popular demand, the Memory Upgrade Program will continue until June 30.

Bristol of Georgetown Computers in Washington, D.C. recently bought the Computer Systems Store in McLean, Virginia.

Wernor Electronik in Sweden has developed Swedish and Danish character generators for the Sol and Processor Technology software. They will be distributing throughout Sweden and Denmark. Interested dealers should contact their Sales Representative for further information.

Whodunnit

Editor.....Dottie Wanat

Asst. Editor.....Sally Turner

Graphics.....Lois Esser
Ray Oleson

General Input.....Mike Mills
Elizabeth Fairchild
Connie Hawkinson
Edith Reisner

Copyright © 1979,
Processor Technology Corporation.
All rights reserved.

New D.O.A. policy now in effect

Beginning March 26, a new policy went into effect for equipment received "Dead on Arrival."

If you test an out-of-box product and find it faulty within 15 days of our shipment date, call Customer Service to receive a Return Authorization number. Be prepared to provide the following information:

A. Product name
B. Date received
C. Invoice number
D. Serial number
E. Problem with the product.

Repack the product in our shipping container and return it to us freight collect. When packing be sure to:

A. Enclose a brief description of the problem;
B. Mark the outside of the container with the R.A. number in two places.

We will repair or replace the unit within 48 hours of our receipt of the product and ship it back to you (postpaid).

If you have any questions, please contact Linda Bilbrey in Customer Service.

New products debut at NCC

(Continued from page 1)
the ability to expand their data management capabilities while cutting some routine paperwork. Some of its more powerful features include:

1. Operator can use two to four disks, allowing for 1500 to 5000 names.
2. Entries can be located within three seconds, can be updated, added on, or deleted.
3. Groups of entries sharing specific characteristics may be deleted or printed.
4. Prints out list of all changes after update.
5. Prints out a selected group of entries or a statistical sample, either alphabetically or by zipcodes.
6. Prints mailing labels for each entry, a selected group, or a sample.
7. When used in conjunction with WordWizard Electronic Typing System:
   A. The mailing list data can be merged into form letters or tables;
   B. Prints out personalized form letters to all entries or a selected group without re-initiating the processing for each letter.
8. Print speed can be up to 540 words per minute for letter quality, 2400 for draft type quality.

General Ledger is a comprehensive bookkeeping system for financial reporting, offering flexibility for specifying own criteria (numbering of accounts and defining departmentalization). Some features include:

1. Maintaining data on a monthly basis for 24 months, stores budget figures for 12 months.
2. Data need only be input once for automatic transfer from receivables, payables and payroll into general ledger program.
3. Closes income and expense accounts to retained earnings at year end.
4. Fully automatic audit trail shows printed reports of all transactions facilitating the tracing of an item back to a source document.
5. Reports produced automatically include:
   A. Chart of accounts
   B. Work sheet trial balance
   C. Source journals
   D. Audit trail control reports.

Accounts Payable records all incoming invoices and due dates by vendor. Other features include:

1. Outgoing checks are printed

(Continued on 4)

35 Dealers attending Software Seminar

By the time most of you receive this newsletter our Software Seminar will be underway. Scheduled to begin on Sunday, April 22, the seminar is designed to introduce you to Processor Technology's new line of business software.

For those of you who were unable to attend, we plan to hold a condensed version of this seminar in conjunction with the National Computer Convention (NCC) show in New York in June. Also, for those unable to attend, the May issue of Personal Computer Retailer will include a summary of the seminar.

New products debut at NCC

(Continued from page 3)

automatically, making appropriate deductions to payables.
2. Multiple invoices can be paid on one check.
3. Credit memos and adjustments maintained with accurate payables totals.
4. Prepares financial reports valuable for planning cash flow requirements.
5. Outstanding invoices listed according to last possible due date each week.
6. Schedule of payments picture, focusing on discount opportunities with discount date, amount and final payment date.
7. Aged trial balance sheet reviews monies owing.
8. Aged cash requirements report breaks out monies owed according to due date.

Accounts Receivable provides customer master maintenance file recording all information regarding balances and sales history. Other features include:
1. All invoices entered into computer and posted to receivables automatically.
2. Payments and/or adjustments are posted against the invoices as received.
3. Computerized month-end processing initiates balance forward process wherein all invoices with nil balance are removed, leaving only outstanding invoices.
4. Produces upon request:
   A. Customer statements
   B. Aged trial balance reports
   C. Exception reports listing by due date.
   D. Offers advice on customer credit limit.

SolSign Astrology System is a complete natal astrology program which includes the calculation and explanation of the birth chart. Accurate calculations of the planets, sun, moon and house cusps. Other features include:
1. Complete transit (daily) astrology system gives daily interpretations as back-ground, and specific interpretation of the daily sun and moon positions.
2. Easy data input, stores the longitude and latitude of major U.S. cities and remembers any new cities that are input.
3. Gives printed output, or can be restricted to the TV screen.
4. Allows chart storage on the disk for research or for personal daily transit information.

Miss our usual?

Frankly, so do we, but in this case, don't feel as though you need to consult your local physician because you're not seeing green. Due to the prolonged strike at the paper mills, we couldn't get our regular green paper for this issue. We hope to be back "on the green" for the next edition of PCR.

New, updated disk versions of WordWizard and FORTRAN available

A new, improved version of WordWizard has recently been released. It includes:

+ Print control statements to center and offset printed text;
+ A center command to center text typed in a document;
+ A screen print activity that lets you preview the printed format of a document without actually printing it;
+ A type-ahead feature that lets you keep typing during a disk access;
+ Various general improvements.

The package is available in two forms:

1. A regular software distribution kit consisting of a System Disk (Rev. F), a Document Disk (Rev. D), the keyboard and diskette labels, and a User's Manual;
2. An update disk--free to dealers who have already bought WordWizard. It will put the new version on existing WordWizard disks.

The update disk will update not only WordWizard, but also FORTRAN. An existing WordWizard disk is updated to Release 4.0 (mod 1); an existing FORTRAN disk is updated to Release 1.1 (mod 0). Any dealer who has purchased WordWizard and/or FORTRAN at any time in the past will automatically receive this update disk free of charge.

Seminar teaches customers AND dealer

We asked Eri Golembo, Vice President of retail sales at the Computer Mart of New Jersey, if he thought their recent Business Applications Seminar was sucessful. Here's his reply:

"We sure learned a lot from that seminar. We learned just how important a structured sales approach is. If you just sit and chat with someone about, let's say, WordWizard, you may forget to mention some important features. Then the customer does some competitive shopping, and someone mentions some special feature you forgot to point out. The customer thinks, 'Wow, that other machine didn't have all this,' and you've lost a sale.

"Another thing we've learned," says Eri, "was the average awareness level of our customers. We've always assumed that the people who came to us inquirying about computers really knew a lot. Our seminar showed us that we were wrong.

"Most of our potential customers actually know very little about computers--only that they think they need one. Often, they don't even know the questions to ask. Or, amybe they're just too embarrassed to ask. They feel that they're the only ones who don't understand the terminology. So, it's important to use use simple, easy-to-understand, non-technical language when explaining the features. Avoid 'computerese'."

Processor Technology's Elizabeth Fairchild and Debby Oester participated in this seminar.

New York, New York!

We're hoping to see all of you at the National Computer Conference (NCC) in New York in June. Our exhibit is located on the 4th floor, booth numbers 4048, 4050, 4052 and 4054.

In the next few weeks you will be receiving from us five guest passes for the show. If you will need more, please let your sales representative know.

We'll have an entirely new booth with two elevated sections devoted to business applications software plus a special area to demonstrate the new disk systems.

Besides showing the new product line, we're also planning a reception for all of you, as well as a condensed version of the dealer business software seminar.

We'll be publishing a full schedule of events in the May issue.

79 Jun 25 *Intelligent Machines Journal*

Processor Technology Goes Under

by Tom Williams

Processor Technology Corporation, of Pleasanton, California, has closed its doors and had its telephone disconnected. On June 8th, police cars were observed at the doors of the building. A source close to *IMJ* asked one of the officers what was going on, and was told that they were attempting to serve a summons, but had as yet been unable to do so.

IMJ has been able to verify that at least two computer dealers in the San Francisco area have been contacted by the Bank of America, and told to make payment on invoices due to Processor Technology directly to the Bank of America and not to the company. Checking with the Federal Bankruptcy Court has revealed that the company has not filed for bankruptcy or for Chapter 11. It is also known that Processor Technology has laid off approximately 75 employees.

A telephone number, (415) 846-9951, has been located which appears to reach the company's offices. Repeated attempts by *IMJ* to reach Bob Marsh, president of the company, for comment, or to get comments from other officers of the company, have proven fruitless.

There has also been talk, albeit unconfirmed, that the company was planning to liquidate assets, such as office furniture and, perhaps, some equipment, by means of a public auction to be held sometime in the near future.

The question that arises for owners and users of Processor Technology equipment is: what sort of future support can they look forward to, and from what source would it come? *IMJ* has learned that Proteus, the Processor Technology User Society, is attempting to keep things together for users. Plans are to try to preserve as much information as possible by collecting and distributing manuals and technical data.

Processor Technology is not committed to providing manuals or notices of engineering changes. Manufacturers marketing Processor Technology accessories will also try to keep up enough orders for another production run. In addition, Proteus will try to set up a service center, employing factory-trained technicians. More information on Proteus' efforts can be obtained from: Proteus, 1690 Woodside Road, Suite 219, Redwood City, CA 94061.

Product Description

About the Author

Bob Bumpous is a machine controls design engineer with Adolph Coors Company and has been doing discrete and computerized machine control design for the past four years. He previously worked for Bendix Navigation & Control Division on documentation of SKYLAB experiments. He is a 1969 graduate of Texas Technological College with a BS in engineering physics, and has an avocational interest in personal computing.

A User's Reaction to the SOL-10 Computer

Robert Bumpous
212 N Ford
Golden CO 80401

When I first saw the Processor Technology SOL it was love at first sight. After acquiring the SOL system, I found that it has everything (almost) a hobbyist needs in one package.

The SOL-10 system which I purchased consists of a single 10 inch by 16 inch board computer, a keyboard, a power supply and cabinet. Figure 1 shows a memory map of the system which gives an idea of its overall structure. Processor Technology also makes the SOL-20 which has a heavier duty power supply and backplane for five extra S-100 cards.

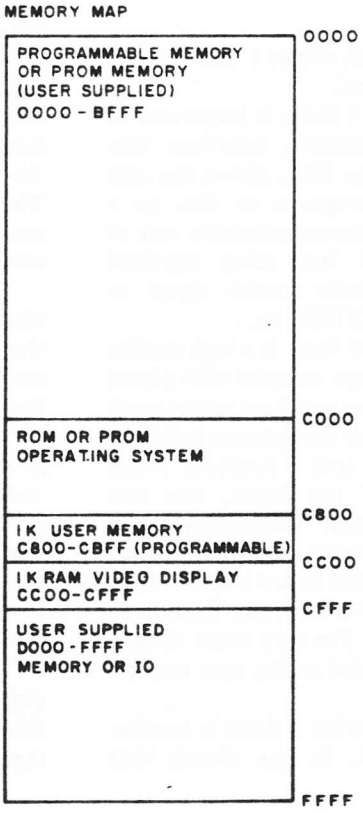


Figure 1: Memory allocations of the SOL design. Within the IO area at D000 to FFFF, the SOL-10 has a UART dedicated to the tape interface, a UART dedicated to the serial communications interface, a parallel keyboard interface, one 8 bit input port and one 8 bit output port.

The SOL single board computer is a complete beginning system. The board contains an 8080A processor with fully buffered data and address buses which are brought to a connector at the center of the board where an S-100 compatible backplane daughter board can be plugged in. It also has a complement of IO ports. The board and operating system are oriented around the parallel keyboard input port and video display output port. 1 K of the 2 K of programmable memory on the board is used to store the 64 character by 16 line video display contents. The output of the display driver is an EIA composite video signal (1.0 to 2.5 V peak to peak) which will drive any standard monitor or even a modified TV set. The full 128 upper and lower case ASCII character set is available for display. With the addition of a video monitor the system becomes a smart video terminal with either full or half duplex communications modes selectable from the keyboard.

The SOL design has an asychronous serial interface port with switch selectable data rates from 70 to 9600 bps. This serial IO port has both an RS-232 and 20 mA current loop output. Also included is an 8 bit

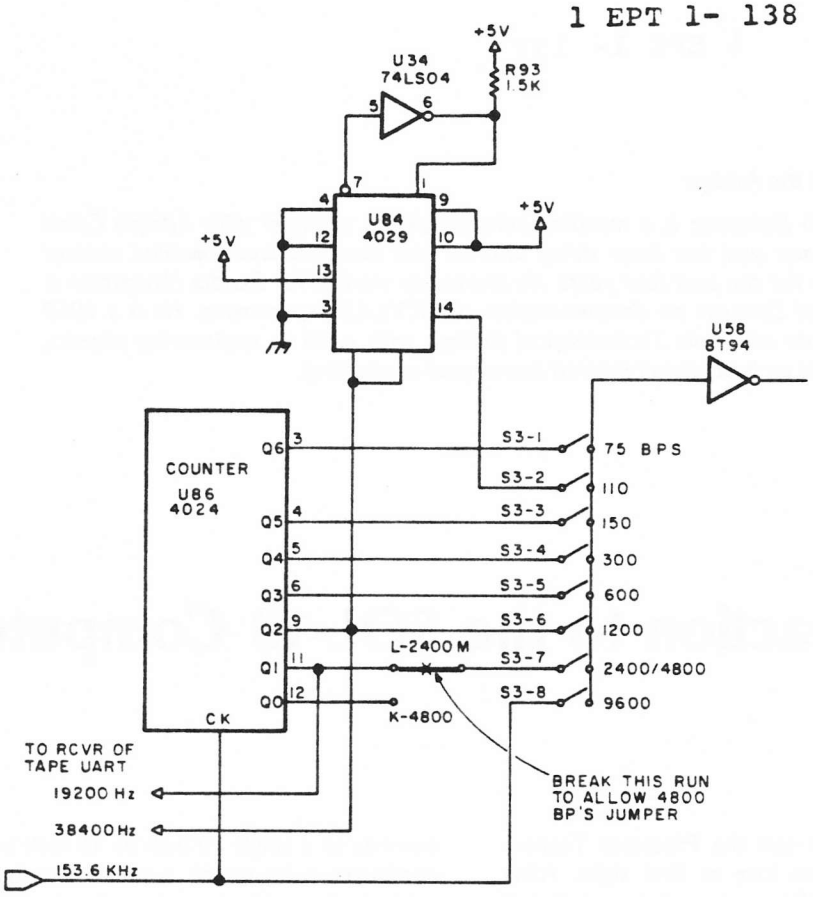


Figure 2: One minor printed circuit bug found by the author in testing out the SOL computer was the run from point L to point M from U86 to the data rate selection switch. If 4800 bps operation is desired as an option, the run in question has to be broken and a jumper wire from point K to point M on the board has to be substituted.

parallel IO port which allows a user to drive any external peripheral.

An option which I think is important to any system is the cassette interface. This option, used with the SOL, allows the user to read or write programs or data to a cassette tape at a software selectable rate of 300 bps or 1200 bps using standard asychronous Manchester coded signal at 1200/2400 Hz or 600/1200 Hz.

The SOL-PC board itself is a high quality double sided G-10 type material with plated thru holes. There is an excellent solder mask on the board with all component locations clearly labeled. The unit I received, which was Revision D of the board, had two defects which Processor Technology asked the assembler to change: two lines crossed on the same side of the board and two lines to the S-100 bus were reversed. Both were easy to correct and I'm sure these defects will have been corrected on the next revision of the board.

I also discovered what I think is another error on the board. In the circuit that selects the serial data rate there is a jumper that enables either 2400 or 4800 bps to be switch selectable. The board had the switch permanently connected to the 2400 b' position. This caused the serial IO to wo improperly when 4800 bps was selected. Because there are lines feeding from this circuit to the cassette interface circuit, the cassette interface would not work properly. The problem is easily corrected by breaking the lines between the L and M jumper contacts (see figure 2).

There is a moderate amount of large scale integration on the board. The 8080A processor, the ASCII 7 by 9 dot matrix encoder, the two UARTs used for the serial and cassette tape interfaces and the 5204 (512 by 8) EROMs in the personality module are the LSI chips used. The programmable memory for the computer and display is made up of 2102 parts. All of the IO select switches and the IO connectors are located at the rear of the board for easy access. The personality module is located along the back edge to allow an easy change of the operating system.

The system timing is derived from a 14.31818 MHz crystal controlled oscillator that feeds the video character generator, UARTs, and 8080 clock divider circuits, MSI and SSI TTL along with an AH 0026 dual MOS clock driver supply the 808' clock signals. There is an RC charge-u circuit that provides an automatic reset signal when power is turned on and a controlled "wait" period immediately after the reset signal.

The keyboard supplied as part of the SOL-10 system is a custom unit from Key Tronics which connects to the SOL-PC board through a ribbon cable connector which plugs into a header on the board. The keyboard uses no LSI encoder and no mechanical contacts and can easily be disassembled for cleaning.

The keyboard is easy to assemble and check out. All of the data bits are latched so that they can be examined at leisure. The strobe bit is only on for a very short time, two to ten μs. There is only one tricky part of the keyboard assembly procedure. This is the installation of the four SIP resistor networks. Extra care must be used in installing these since they are not as clearly marked as most of the other items, and the two different sets of values will not allow the keyboard to work properly if interchanged.

The power supply for the SOL-10 consists of a power transformer, an externa filter capacitor for the 5 V supply and a regulator card to supply the required vol-

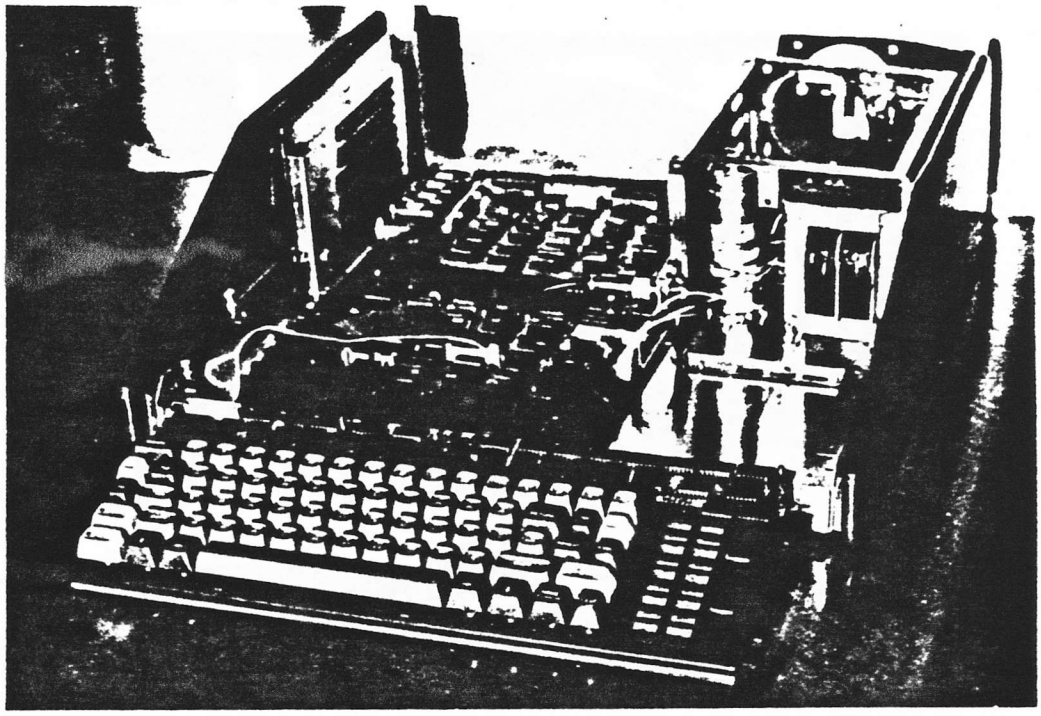


Photo 1: Processor Technology SOL, with cabinet cover removed to reveal its inner secrets.

tages (12 VDC at 150 mA, 5 VDC at 2.5 A, -12 VDC at 200 mA) to the SOL-PC and keyboard. The SOL-20 system power supply, in addition to these voltages, provides 16 VDC at 1 A unregulated, -16 VDC at 1 A unregulated, and 8 VDC at 6 A unregulated for the backplane Altair (S-100) bus daughter board. The regulated 5 V to the SOL-PC board has an over voltage crowbar on its output which, on my original version, had an overly sensitive turn on characteristic. Processor Technology has since sent a revision which corrected this problem.

The SOL system cabinet is sheet metal with solid walnut sides. It is designed to house the SOL-PC, the keyboard, power supply, backplane and five additional S-100 cards in a fan cooled card cage. The cabinet is constructed so that the power supply is isolated from any radio frequency interference (RFI) switching transients in the memory or processor cards. The whole system looks well shielded, which is an important consideration in this age of CBers. The walnut sides come sanded but unfinished; they look nice but take a lot of time to finish properly.

The assembly manual that came with the system describes the assembly procedure very well. It features the kind of complete, step by step format that kit builders have come to expect. It seems very complete, including pictures and drawings of how to assemble the components and subassemblies into a useful computer and terminal. However, there were no flowcharts, block diagrams or explanations of how the different sections of the computer are supposed to work, or how they interact with each other. As of this writing (spring 1977) I have still not received two sections of my manual, operating procedures, and theory of operation. Most of the information that I have on the operation of the machine has come from an article in the February *Digital Design* by Lee Felsenstein of LGC Engineering and Robert Marsh (vice president of Processor Technology), who are the co-developers of the SOL terminal system. This article gives a fairly complete though not detailed description of the operation and interactions of the SOL system components.

The CONSOL operating system, which is the minimal operating system, is stored in 1024 K bytes of EROM starting at C000. This operating system has seven commands that can be executed from the keyboard, sufficient to allow the operator to enter his or her own programs or prerecorded programs and execute them. The following are the commands available through CONSOL:

1. DUmp ADDR1 ADDR2 dumps the contents of memory between address 1 and address 2 onto the CRT terminal in a hexadecimal format.
2. ENter ADDR1 enters hexadecimal format data into memory starting at address 1.

1 EPT 1- 140

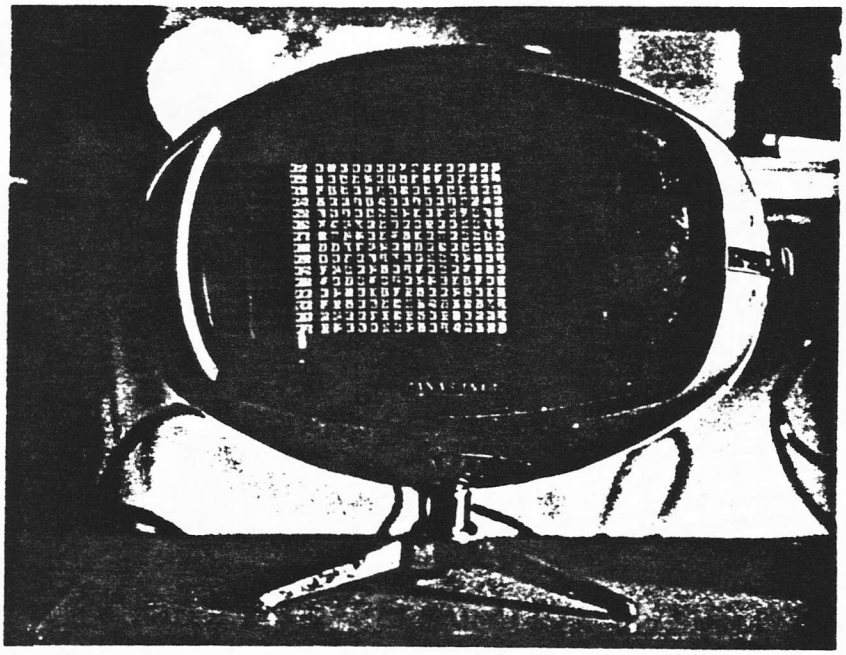


Photo 2: The SOL's display generator produced this video character graphics pattern on the author's television monitor.

3. EXecute ADDR1 executes a program in memory starting at address 1.
4. TErminal is a program designed to allow the SOL system to be used like a standard CRT terminal with serial 20 mA or RS232 interface. This ability could make the SOL system very attractive to a small business user who needs an inexpensive terminal.
5. TL (tape load) loads a prerecorded cassette tape into memory at an address specified by the tape. The information supplied with the software doesn't give a very good definition of how the TL command looks at the tape header data. Because of this, it is very difficult for the user to configure a tape-header for any data he or she would like to store on a cassette.
6. BAsic executes a program located at address 0000, typically BASIC.
7. The final command is the MODE command which calls a program that allows all the other commands to be entered on the keyboard and recognized by the computer system.

The early version of CONSOL which I received does not have a command to dump data from memory into a cassette tape. This is a great disadvantage when trying to develop a program. Whether the program being developed bombs out or is successful it is nice to have a copy so it doesn't all have to be typed in again.

The purchaser of a SOL system is also supposed to receive a 5 K BASIC operating system with the purchase of the on board audio cassette interface. This 5 K BASIC includes two computer video games. I had not yet received this when this article was written in the spring of 1977.

I ordered my SOL system on September 4 1976 and anxiously awaited its arrival. Only a portion, the SOL-PC, arrived on November 18 1976 along with part of the manual. When I opened the box I was confronted with what appeared to be a rather large, random accumulation of parts. However, all of the parts called out in the parts list were there. The assembly would have gone faster if the components had been sorted, placed in small bags, and marked as to the section of the assembly procedure they were for.

The procedure given in the manual calls for construction and checkout to proceed together. To follow the manual you must have a power supply and keyboard to properly assemble the complete SOL system. Because I didn't have either of these items I was only able to proceed with the assembly until I came to the first step, which required the power supply; there I stalled. I spent all of Christmas 1976 lurking around the mail box, awaiting the delivery of the rest of my system. Finally, on January 4, it arrived and off to my basement "laboratory" I scurried, not to emerge "til the beaste was up and running."

After much soldering, bending and clipping, I reached the point of the first operations test. I plugged in my TV monitor, turned on the SOL system power, hit the MODE key, and typed in the DUMP command. Nothing! All I could get was the CURSOR in the upper left hand corner of the screen. This is when the lack of block diagrams and explanations of the system really hit me. After what seemed an extremely long time, I managed to solve the problem. The display uses the first several words of user memory to store the line and character positions for the video display, and somehow, in my haste to complete the assembly, I had bent a pin under on one of the memory chips. Dumb, but almost impossible to locate. Once this problem was solved, the rest of the system went together very smoothly.

The total assembly time, excluding the

time I spent looking for my own error, was about 36 hours. The sanding and coating of the walnut cabinet sides took about five more hours. I used clear satin polyurethane finish of the type found in most hardware stores. This is a very hard, clear, fast drying resin that puts a tough, mar resistant sealer on the side panels.

Overall, I feel that the Processor Technology entry in the microprocessor race is a good buy. I have had what seems to me excessive delays but I believe that now that the production of these units is in full swing, there should be a minimal amount of lead time on the system hardware. Processor Technology says that they have sold almost 1,000 of these systems. The software development, however, seems to be lagging well behind the hardware as might be expected from the history of computer science in general.

If you are planning to develop your own software or plan on buying it somewhere other than Processor Technology, the SOL system will serve you nicely. The CONSOL routine provides a step up from the front panel toggle switch design philosophy of most of the other microcomputer manufacturers and I feel it warrants the purchase of this system. ■

Editors' Note:

At the time this article was edited (August 15 1977) a SOL-20 with 16 K memory and CONSOL was resident in our office at BYTE. 5 K BASIC is now being delivered, according to Terry Holmes of Processor Technology.

The Following Comments Were Received from Bob Bumpous with His Author's Proofs on October 7 1977.

Since completing my SOL-10 I have written a couple of short programs to test the various hardware options of my system.

I have also purchased the SOL upgrade kit which adds the larger power supply fan and backplane and an IMS 8 K memory.

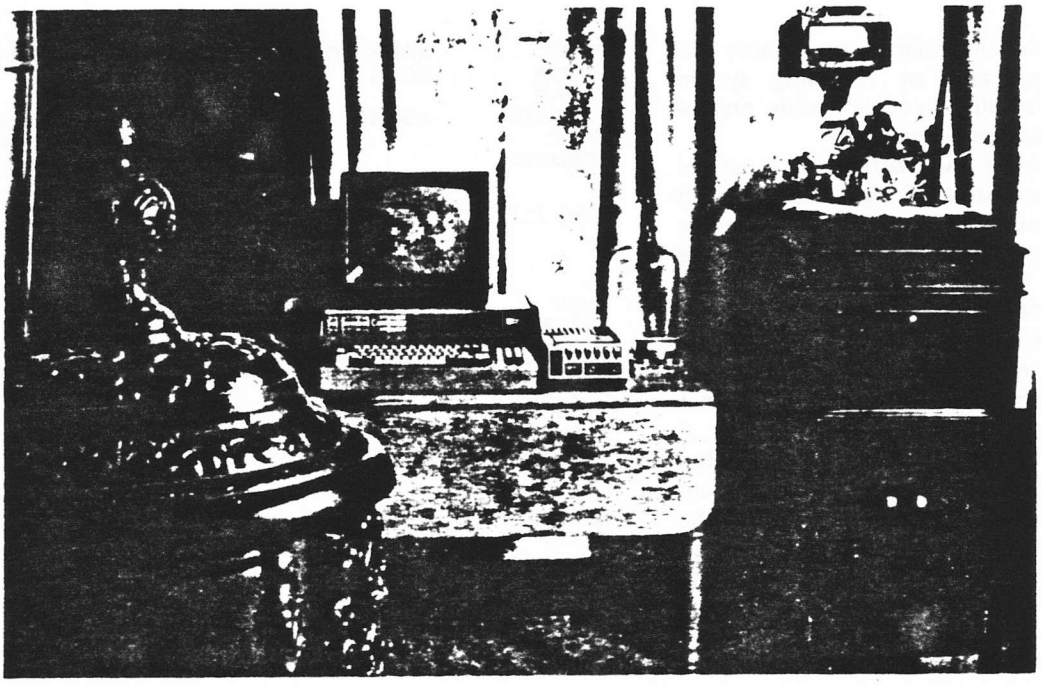
I have received the two missing sections of my manual and my BASIC tape. I have not yet been able to run BASIC due to a problem I'm having getting my cassette tape interface to operate properly.

I have also received my SOLOS operating system; all of the functions except the tape interface seem to work well.

Circle 38 on inquiry card.

This page intentionally left blank.

1 EPT 1- 143



The SOL-20 in a New England parlor context, obviously enjoying life. The entire system fits on one table with a video monitor on top of the computer and an audio tape recorder for mass storage to the right.

User's Report: The SOL-20

Dennis E Barbour
2 Coliseum Av
Nashua NH 03060

John Kemmeny in his book, *Man and the Computer*, reveals his thoughts concerning the future of computers. Kemmeny believes that people will have access to large regional computers in the future. This will allow people to do many jobs in the home and provide entertainment and knowledge. A computer terminal in the home will be tied to a large regional supercomputer giving people the ability to do many things at home not now possible, such as reading the latest news stories, studying subjects for degree credit, storing recipes to allow the input of a week's menu and production of a grocery list of needed items, and the recording and computation of family financial matters. The possible uses of this type of computer are unlimited.

Processor Technology has taken this concept of the home computer terminal attached to a supercomputer one step toward reality by providing the home user with a terminal and microprocessor combination, the SOL, capable of functioning with, or independent from, another computer. The operating systems provided by Processor Technology allow the SOL system to operate as a microprocessor independent of any outside processor with its own peripherals until the terminal command is issued, which causes SOL to operate as a computer terminal. This combination will allow the greater use of your home microcomputer today and in the future. What follows is a brief description of a SOL system based on my personal experiences.

Hardware

My microprocessor, a SOL-20, uses an 8080 processor chip. (The 8080A and the 9080A are also available.) This system has both parallel and serial communication interfaces, a keyboard interface, audio cassette interface, a video display generator, 1024 8 bit words of volatile system memory, and 1024 8 bit words of volatile display refresh memory. My system also has 8 K

bytes of volatile main user memory for programming and an operating system in a "personality module" with preprogrammed read only memories.

A feature of the hardware is a parallel interface consisting of eight bits for each input and output plus control handshaking signals. The output bus uses three state TTL levels for bidirectional interfaces. The serial interface circuit includes both asynchronous RS-232 and 20 mA current loop provisions and operates at from 75 to 9600 bits per second, switch selectable. (When I say that something is switch selectable, I mean that there is a series of small switches on the printed circuit board and an explanation in the hardware manual of what the switch does and what your options are.) The keyboard interface handles a 7 level ASCII encoded TTL keyboard which requires a strobe pulse after the data is stable.

The audio cassette interface is program controllable and self-clocking with a phase lock loop. It operates at the dual rate of 300 or 1200 bits per second. Included is an automatic level control in both the record and playback modes. Recording is the CUTS/BYTE standard compatible, an asynchronous Manchester code at 1200/2400 Hz or 600/1200 Hz.

The video display circuitry generates 16 lines from data stored in its 1 K refresh memory. Alphanumeric and control characters (the character set includes all upper and lower ASCII plus control characters) are displayed in either black on white or white on black (this feature is switch selectable). The solid video cursor with a switch selectable blink is programmable. This circuitry can drive any video monitor or monochrome television converted to handle video input.

The assembly of the SOL appears to be relatively easy. I bought my SOL fully assembled, but I read the assembly instructions, looked at the diagrams, and thoroughly examined the hardware. I believe that I could quite easily put a SOL together. The instructions are simple and straightforward. The assembly notes include parts lists, assembly tips and precautions, pictures and drawings, and step-by-step instructions on assembly and testing. Also included are instructions on handling integrated circuits, soldering techniques, and a list of the tools that will be needed. Processor Technology provides schematic and detailed assembly drawings in a separate section of their manual.

Another important feature of the SOL is the system's completeness and modularity. All of the hardware is in one package and looks like a large electric typewriter. The video monitor and cassette recorders are attached at the back panel with cables. The system does not have exposed boards or parts to collect dust or be damaged. The system can be easily transported.

DUMP	ADDR, ADDR	Dump memory locations to screen
ENTER	ADDR	Enter data to memory
TLOAD		Load to memory
EXECUTE	ADDR	Execute a program at a given address
BASIC		Execute a program located at address zero
TERMINAL		Return to terminal mode

Table 1: CONSOL operating system command summary.

DUMP	ADDR, ADDR	Dump memory locations to screen
ENTER	ADDR	Enter data to memory
TLOAD		Load data from tape to memory
EXECUTE	ADDR	Execute a program at a given address
BASIC		Execute a program located at address zero
TERMINAL		Return to terminal mode
TSAVE (NAME)	ADDR, ADDR	Save certain memory locations on tape
TXEQ (NAME)	ADDR	Load a certain program from tape and execute it
TCAT		List tape header information
SET		Allows the setting of the following things
SET DI		Set display speed
SET IN		Set input driver
SET OUT		Set output driver
SET TAPE		Set tape speed
SET CI		Set custom input driver address
SET CO		Set custom output driver address
SET TY		Set types into tape save header
SET XE		Set execute address into tape save header

Table 2: SOLOS operating system command summary.

Operating Systems

SOL, at present, has three operating systems. They are CONSOL, SOLED and SOLOS. CONSOL allows simple terminal operations and computer functions. SOLED allows more advanced terminal operations than CONSOL with additional screen, file and cassette tape operations. SOLOS allows use of the full range of the microprocessor plus the full functions of an intelligent terminal. I cannot describe SOLED any further, not having the information available to me. However, CONSOL and SOLOS will be described in detail.

CONSOL, the simplest of these operating systems, allows the SOL to function both as a standard CRT terminal and as a microcomputer. This software system is not as sophisticated as SOLOS which has many more commands and greater abilities.

All the operating systems work in the following fashion. The same switch to power up the hardware serves to initialize the operating system. The operating system initializes system memory, clears the screen and enters the terminal mode. Depressing the mode key causes SOL to operate as a microcomputer. The operating system issues a prompt character and awaits further commands.

```
10 REM THIS PROGRAM IS FOR TEACHING YOUNGSTERS MULTIPLICATION.
20 PRINT "DO YOU WANT TO TRY THE MULTIPLICATION PROGRAM?"
30 PRINT "ANSWER 1 FOR YES, AND 2 FOR NO."
40 INPUT X
50 IF X=2 THEN 340
60 PRINT
70 PRINT "INPUT THE FIRST NUMBER."
80 INPUT A
90 PRINT A;" TIMES"
100 PRINT
110 PRINT "INPUT THE SECOND NUMBER."
120 PRINT B
130 PRINT A;" TIMES";B;" EQUALS"
140 PRINT
150 PRINT "INPUT WHAT YOU THINK THE ANSWER SHOULD BE."
160 J=A*B
170 INPUT C
180 PRINT A;" TIMES";B;" EQUALS";C
190 IF C=J THEN 260
200 PRINT
210 PRINT "YOUR ANSWER WAS INCORRECT."
220 PRINT
230 PRINT A;" TIMES";B;" EQUALS";J
240 PRINT
250 GOTO 290
260 PRINT
270 PRINT "CONGRATULATIONS, YOUR ANSWER WAS CORRECT."
280 PRINT
290 PRINT "DO YOU WANT TO CONTINUE?"
300 PRINT "ANSWER 1 FOR YES, AND 2 FOR NO."
310 INPUT X
320 IF X=2 THEN 340
330 GOTO 60
340 PRINT
350 PRINT "HAVE A NICE DAY!"
360 END


*RUN


DO YOU WANT TO TRY THE MULTIPLICATION PROGRAM?
ANSWER 1 FOR YES, AND 2 FOR NO.
?1

INPUT THE FIRST NUMBER.
?2
    2  TIMES

INPUT THE SECOND NUMBER.
?2
    2  TIMES    2  EQUALS

INPUT WHAT YOU THINK THE ANSWER SHOULD BE.
?2
    2  TIMES    2  EQUALS   2

YOUR ANSWER WAS INCORRECT.

    2  TIMES    2  EQUALS   4

DO YOU WANT TO CONTINUE?
ANSWER 1 FOR YES, AND 2 FOR NO.
?1

INPUT THE FIRST NUMBER.
?2
    2  TIMES

INPUT THE SECOND NUMBER.
?2
    2  TIMES    2  EQUALS

INPUT WHAT YOU THINK THE ANSWER SHOULD BE.
?4
    2  TIMES    2  EQUALS   4

CONGRATULATIONS, YOUR ANSWER WAS CORRECT.

DO YOU WANT TO CONTINUE?
ANSWER 1 FOR YES, AND 2 FOR NO.
?2

HAVE A NICE DAY!
```

Listing 1: A tutorial BASIC program of the simplest variety, created one evening to instruct a 6 year old in some of the complexities of arithmetic.

There are five commands for operating SOL as a microcomputer under CONSOL, two for entering data, two for executing programs, and one for displaying the contents of memory on the monitor. (See table 1 for the complete list of CONSOL commands.) A person can enter data either through the keyboard by means of the ENTER command or through a cassette tape recorder by means of the TLOAD command. The ENTER command permits the altering of any byte or series of bytes of storage with either data or a machine language program. Program execution begins with the issuance of either of two commands. EXECUTE allows the execution of a program beginning at any point that you name in memory, whereas the issuance of the BASIC command starts program execution from memory location zero. The fifth command, DUMP, allows the displaying of any byte or series of bytes on the video monitor.

The SOLOS operating system allows for more advanced operation of the SOL. (See table 2 for a complete list of SOLOS commands.) Besides the six basic commands allowed under CONSOL, there are four additional commands, with one of these additional commands, SET, having eight different options. To begin with, the TLOAD command is different under SOLOS than under CONSOL. Under SOLOS, the TLOAD command allows the searching of a tape with multiple files for a certain named file, skipping over all files before it. Under CONSOL, you have to manually move the tape to the beginning of the tape file.

TSAVE allows the saving on tape of the contents of the memory locations indicated. TXEC allows the loading of a named program from tape to memory and the immediate execution of that program. TCAT allows the listing of the tape file header information on the monitor. This information includes the file name, execution address, and the beginning and ending addresses of the program when loaded to memory.

The SET command has eight different options. One can set the display speed, allowing the speeding up and slowing down of characters on the monitor to suit your programming. Using this option of the BASIC language interpreter, I was able to generate a very impressive lightning bolt image for my young son and his friends. The input and output driver speeds can be controlled through the use of the SET command along with the tape speed, and

the input and output driver addresses. The tape save headers can also be altered using SET. The more useful of the two operating systems is obviously SOLOS since it allows greater flexibility.

Software

My system has a BASIC interpreter, the BASIC-5 package. This interpreter has many of the usual BASIC commands and statements. It also allows use of system commands within programs. BASIC-5 allows the usage of single dimension arrays, many BASIC functions, and the ability to read and write tapes. It also has 8 digit precision and the ability to link to machine language subprograms.

I would advise the purchase of the 8 K BASIC for one important reason: the ability to have several commands and features not available in the BASIC-5 interpreter. These would be the RNUM command which allows the renumbering of statements, and also the ability to utilize alphanumerics and matrices (instead of only single dimension arrays) which would be particularly useful.

Uses

I have utilized my SOL for several things. First and foremost, I have introduced my 6 year old son to computers through games and learning programs, such as the multiplication program reproduced and described in listing 1. I have also used it for entertainment especially through the use of the target game provided by Processor Technology, and a few games I have written. Lastly, I have used it to fool around with my first love, history. I have written several statistical analysis programs for the SOL using the BASIC-5 software to continue with my research on colonial New England. Also reproduced here, as listing 2, is a sample of a simple program to do elementary fertility figures.

I have reproduced and wholly written a few programs to assist in the mathematical education of my son. MULT and two other programs have been written as interactive quizzing programs to test and teach mathematics through praise when he is correct and by showing him the correct way when he is wrong. Thus the computer acts as a teacher and could be utilized in this fashion in every home and classroom for elementary school aged children. More sophisticated programs could be implemented on the SOL for older children. These programs, in the form of games, could serve two purposes, entertainment and education.

1 EPT 1- 146

Listing 2: A sample of a statistical BASIC application typical of programs used by the author in his avocation of historical studies.

```
10 PRINT"THIS PROGRAM WILL PRODUCE FERTILITY FIGURES"
11 PRINT"FOR THE DATA INPUT. THE OUTPUT CONSISTS OF"
12 PRINT"THE PROPORTION OF BIRTHS TO MARRIAGES OVER"
13 PRINT"A DECADE WITH FIVE YEAR INCREMENTS BETWEEN DATA."
14 PRINT"THESE FERTILITY FIGURES (THE BIRTH RATE) CONSIST OF"
15 PRINT"THE NUMBER OF BIRTHS FOR A DECADE DIVIDED BY THE"
16 PRINT"NUMBER OF MARRIAGES FOR A DECADE WHICH BEGAN FIVE"
17 PRINT"YEARS EARLIER."
65 D1=0
66 I=0
100 PRINT"INPUT THE STARTING YEAR (BETWEEN 1650/1795) DESIRED    "
105 INPUT Y
106 PRINT"INPUT THE NUMBER OF SETS OF YEARS OF DATA              "
107 INPUT D
110 PRINT"INPUT THE NUMBER OF FIVE YEAR INCREMENTS DESIRED       "
120 INPUT N
125 DIM Y1(40),M1(40),B1(40),F1(40),Y2(40)
129 DIM F2(40),Y3(40)
130 FOR J=1 TO D
140 READ Y1(J),M1(J),B1(J)
160 NEXT J
195 PRINT"              MARRIAGE YEARS              FERTILITY"
200 FOR J=1 TO D
205 IF I=N THEN 340
210 IF Y1(J)<Y THEN 290
220 Y2(J)=Y1(J+2)-1
230 F1(J)=(B1(J+1)+B1(J+2))/(M1(J)+M1(J+1))
250 PRINT Y1(J),Y2(J),F1(J)
260 S=S+F1(J)
270 D1=D1+1
280 I=I+1
285 F2(I)=F1(J)
286 Y3(I)=Y1(J)
290 NEXT J
340 M=S/D1
350 PRINT"MEAN OF FERTILITY FIGURES                             "M
395 PRINT
396 PRINT
398 PRINT " YEAR                        FERTILITY"
399 PRINT"+++++++++++++++++++++++++++++++++++++++++++++++++++++++"
400 FOR K=1 TO I
420 L=INT((F2(K)*10)/2)
430 PRINT Y3(K);"++";TAB(L);"*"
490 NEXT K
500 PRINT"+++++++++++++++++++++++++++++++++++++++++++++++++++++++"
900 DATA 1650,5,28,1655,8,32,1660,14,43,1665,9,44
910 DATA 1670,20,78,1675,18,90,1680,26,125,1685,29,168
920 DATA 1690,24,163,1695,20,156,1700,24,179,1705,34,196
930 DATA 1710,54,239,1715,40,247,1720,64,287,1725,66,289
940 DATA 1730,84,333,1735,88,328,1740,88,371,1745,92,354
950 DATA 1750,87,372,1755,105,365,1760,88,388,1765,80,377
960 DATA 1770,65,348,1775,80,311,1780,108,298,1785,105,344
970 DATA 1790,118,228,1795,99,350
999 END

THIS PROGRAM WILL PRODUCE FERTILITY FIGURES
FOR THE DATA INPUT. THE OUTPUT CONSISTS OF
THE PROPORTION OF BIRTHS TO MARRIAGES OVER
A DECADE WITH FIVE YEAR INCREMENTS BETWEEN DATA.
THESE FERTILITY FIGURES (THE BIRTH RATE) CONSIST OF
THE NUMBER OF BIRTHS FOR A DECADE DIVIDED BY THE
NUMBER OF MARRIAGES FOR A DECADE WHICH BEGAN FIVE
YEARS EARLIER.
INPUT THE STARTING YEAR (BETWEEN 1650/1795) DESIRED:
?1650
INPUT THE NUMBER OF SETS OF YEARS OF DATA:
?30
INPUT THE NUMBER OF FIVE YEAR INCREMENTS DESIRED:
?20
          MARRIAGE YEARS                FERTILITY
        1650            1659            5.769231
        1655            1664            3.954545
        1660            1669            5.304348
        1665            1674            5.793100
        1670            1679            5.657895
        1675            1684            6.659091
        1680            1689            6.018182
        1685            1694            6.018868
        1690            1699            7.613636
        1695            1704            8.522727
        1700            1709                7.5
        1705            1714            5.522727
        1710            1719            5.680851
        1715            1724            5.538462
        1720            1729            4.784615
        1725            1734            4.406667
        1730            1739            4.063953
        1735            1744            4.119318
        1740            1749            4.033333
        1745            1754            4.117318
MEAN OF FERTILITY FIGURES:   5.553943
```

SEMINARS

Micro Diversions
Continuing Microcomputer Education
announces

MICROCOMPUTING HARDWARE AND SOFTWARE

An introduction to the design, construction, and programming of microprocessor-based computers.

These are companion "how-to-do-it" courses on the hardware and software technologies of personal computing taught by Micro Diversions product designers.

SOFTWARE OUTLINE

The Software Concept
(A) The stored program - bits and bytes
(B) Algorithms in problem solving
(C) Assemblers and high level languages

Introduction to Programming - BASIC
(A) Elementary programming concepts
(B) BASIC language statements
(C) Simple data structures and examples

Introduction to Assembly Language
(A) Registers, memory, input/output
(B) Data representation
(C) Data movement and exchange
(D) Arithmetic and logical operations

The Z-80 Microprocessor Instruction Set
(A) Registers and addressing modes
(B) Memory and register data transfers
(C) Arithmetic and logical operations
(D) Bit manipulation
(E) Transfer of control, jumps
(F) I/O instructions

Assembly Language Techniques
(A) General conventions
(B) Data structures and types
(C) Modular programming - subroutines
(D) Using the instruction set

Interrupt and I/O Programming
(A) Interrupt structures
(B) I/O handlers
(C) Data rates and trade-offs

Software Architecture of a Monitor
(A) The monitor as procedure
(B) Functions of a monitor
(C) The jump table - vectored entry
(D) Analysis of a monitor

Loading and Debugging
(A) Getting the code in
(B) Relocation and linking
(C) Breakpoints and snapshots
(D) Patching and polishing

HARDWARE OUTLINE

History and Culture of Microcomputing
(A) When, where, why
(B) Literature, reference books, journals
(C) Personal computing, applications of microcomputers

Digital Electronics
(A) Introduction to and survey of component families
(B) Common components and how to use them
(C) Large-scale integration components
(D) Basics of digital design

Introduction to Microcomputer Architecture
(A) Buss concept
(B) Addressing, input/output
(C) User interface techniques

How a Central Processing Unit Works
(A) Introduction to the CPU concept
(B) Intel 8080 CPU
(C) Zilog Z80 CPU
(D) Interrupt architectures and uses
(E) Real time clocks

How a Microcomputer's Memory Works
(A) Random access memory, its architecture and design
(B) Read-only memory
(C) Mass storage (disks and tape cassettes)
(D) Bubble memories

How TV Interfaces Work
(A) Character display principles and design
(B) Full graphics interfaces for plotting and drawing
(C) Light pens and how to interface them

How Microcomputer Peripherals Work
(A) Keyboard scanners
(B) Tape cassette interfaces
(C) Telephone interfaces (UARTs and MODEMs)
(D) A/D converters

Hobbyist Construction Techniques
(A) Layout and technical considerations
(B) Mounting techniques
(C) Wire wrapping tools and techniques
(D) How to get started (what to buy, where to buy)

Eight Monday and Wednesday Evening Sessions
1 May through 24 May

Each course $110.00 or $190.00 for both.

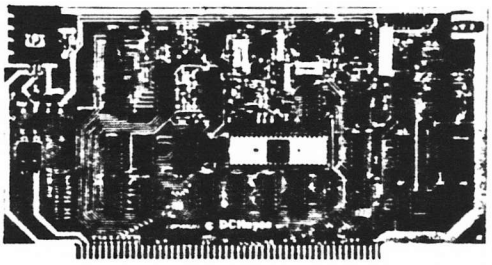
Micro Diversions, Inc.
7900 Westpark Drive, Suite 308
McLean, Virginia 22101
(703) 827-0888

Listing 2, continued:

```
YEAR                    FERTILITY
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
1650 ++
1655 ++
1660 ++
1665 ++
1670 ++
1675 ++
1680 ++
1685 ++
1690 ++
1695 ++
1700 ++
1705 ++
1710 ++
1715 ++
1720 ++
1725 ++
1730 ++
1735 ++
1740 ++
1745 ++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

In closing, I find the SOL system easy to use, fun to program, and important in our home. It provides my family with entertainment in the form of games both from Processor Technology and written by us. It also provides a mathematics tutor at home for my son. It has also given my son an interest in and understanding of computers far beyond his years and education. I definitely recommend the SOL, based on my experiences with it.■

80-103A Serial I/O and FSK modem for professional and hobby communications.



- Completely compatible with your IMSAI, ALTAIR* SOL** or other S-100 microcomputers. Trademarks of *MITS, **Processor Technology
- Designed for use on the dial telephone or TWX networks, or 2-wire dedicated lines, meets all FCC regulations when used with a CBT coupler.
- All digital modulation and demodulation with on board cyrstal clock and precision filter mean that NO ADJUSTMENTS ARE REQUIRED
  - Bell 103 standard frequencies
  - Automated dial (pulsed) and answer
  - Originate and answer mode
  - 110 or 300 BPS speed select
  - Complete self test capability
  - Character length, stop bit, and parity
  - 90 day warranty and full documentation

PRICES Bare Board and Manual 49.95
Assembled (48 hour burn in) 279.95
Master Charge or Visa accepted.

DC Hayes Assoc.
P.O. Box 9884, Atlanta, Ga. 30319, 404/231-0574

DATA COMMUNICATIONS ADAPTER

This page intentionally left blank.

Chapter 2. Processor Technology ACCESS Newsletter
Vol 1, No 1, February 1977..... 2
Vol 1, No 2, April 1977.....14
Vol 1, No 3, June 1977.....30
Vol 1, No 4, November 1977.....50
Vol 2, No 1, March 1978.....86

PROCESSOR TECHNOLOGY

ACCESS.

75¢ per issue

Published by Wible/Rampton Advertising, San Francisco. Volume One, Number One, February 1977

A Letter from the Editor

"The flowers, the gorgeous, mystic multi-colored flowers are not the flowers of life, but people, yes people are the true flowers of life: and it has been a most precious pleasure to have temporarily strolled in your garden."

Lord Buckley

No, this isn't a poetry journal or philosophical tract, but we do feel that Lord Buckley elucidates the concept behind the Processor Technology newsletter: its contents will be the kind of information that allows you to expand the creative applications of your personal computing system.

ACCESS is dedicated to the creation of a personal dialog between us at PTC and you, the people using our products, because we believe that it's important for you to understand your present system well enough to use it effectively, before you become trapped in the "bigger means better" syndrome.

Hence there will be no advertising of any kind in ACCESS, and no thinly disguised pitches aimed at getting you to part with your hard earned money for the unneccessary duplication of existing hardware.

Each issue *will* contain articles, engineering/application bulletins, and software (PROGRAM) listings which we think will help you get more fun out of owning your computer.

All we expect from you in return is what's known in technical and psychological circles as feedback. If you have devised a neat hardware modification or written a nifty program, send it to us and you'll get no financial remuneration. You'll just get the glory of seeing your name in print. Very spiritual, eh what?

Aram Attarian II

Subscription Information

ACCESS will be published every six weeks. This first issue is being sent free to all of our customers. If you like what you see, we hope you'll send us $4.00 for a year's subscription so we can keep the info coming. Write to us at Processor Technology, 6200 Hollis Street, Emeryville, CA 94608.

One to One Communication

Open two-way communication is our goal, and it would be great to have unlimited time to rap with you over the phone. But we can't manage that logistically, so we are instituting procedures that will get your questions answered efficiently, without eliminating the personal touch.

You can get through to our engineers and technical staff any time between 9:30 and noon or 1:30 and 4, Monday through Friday. The number is (415) 652-8080. They'll try to answer your technical questions and provide more detailed product information than you may glean from our literature. It helps a lot if you think out your questions before you call, maybe make a few notes. It helps, too, if you have the manual and/or equipment handy to the phone. The easier it is for you to figure out exactly what we're talking about (and vice versa), the more help we can give in one phone call and the more phone calls we can handle. We'll also be happy to answer your questions if you drop us a letter.

There are two or three situations in which calling us isn't the fastest way to get help. **If you have a problem with a Processor Technology product which you purchased from a local dealer,** try your dealer first. They're all authorized to service the products they sell, precisely because they can give you faster and more personalized service than anybody can by mail. Our dealers all have diagnostic tapes available now, so if you need a PTC product tested it can be handled on the spot.

If you need a defective part replaced, just mail it to us with a note of explanation. There's nothing we can do about it over the phone anyway, so you might as well save yourself the time. (Naturally, it helps if yc explanation is as clear and concise as possible, but we can't all be Hemingways.) If you need service that isn't covered by your warranty, please enclose a check for the $20.00 minimum service charge. If the cost exceeds that (heaven forbid!) you will be notified before we proceed with the service.

If you do send in one of our products for repair, **please send only the defective board.** For obvious reasons, we cannot be responsible for the care and feeding of your monitors, keyboards or other products from other manufacturers.

MATCHMAKING-Software Division

Since most important programs are written for a particular system, modifications are sometimes in order to improve compatibility with your system. We'll be passing on more good ideas as we come across them; here are two dealing with MITS BASIC and the VDM.

MITS 12K Extended Disk and the VDM

If you don't have a 3P+S interface, you can still use the BASIC program in your VDM manual to link MITS 12K Extended Disc to your VDM driver. Simply change these six lines in the listing of VDM-1 to BASIC link appearing on page AV-18 of your VDM manual, and then run the program as explained there. Change the statements to read as follows:

```
0106 FOR Y=L TO L+419
0140 POKE G,195:POKE G+1,0:POKE G+2,P:POKE G+3,0
0160 DATA195,151,1001,195,156,1001,245,229,213,197
0162 DATA205,58,1000,193,209,195,147,1001,245,58
0232 DATA0,0,6,225,241,193,201,254,26,195,142
0234 DATA13,254,127,202,142,13,195,46,1000
```

MITS 8K or 12K BASIC and the VDM

If you have I/O ports that are assigned 20 and 21 instead of 0 and 1, you'll need to make five statement changes in the VDM-1 to BASIC link program in your VDM manual. The listing appears on pp. AV-17 to AV-19 of the manual, and should be modified as follows:

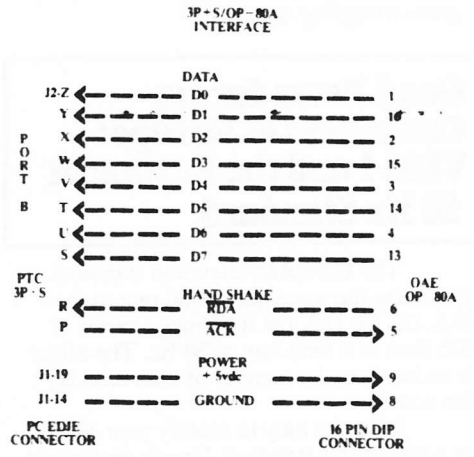
```
0078 IF A=219 AND B=16 GOTO 92
0080 IF A=219 AND B=17 THEN I=K:GOTO 106
0152 DATA219,255,31,210,13,1000,219,16,230,2000
0224 DATA201,219,16,230,6000,201,58,140,1001,254
0228 DATA201,219,17,230,127,254,1,202,46,1000
```

MATCHMAKING- Hardware Division

When you're combining equipment from different manufacturers, optimum performance often depends on specific information about the idiosyncracies of their connections. Owner's manuals can't cover every permutation and combination, so think of this feature as a running supplement that gets down to cases.

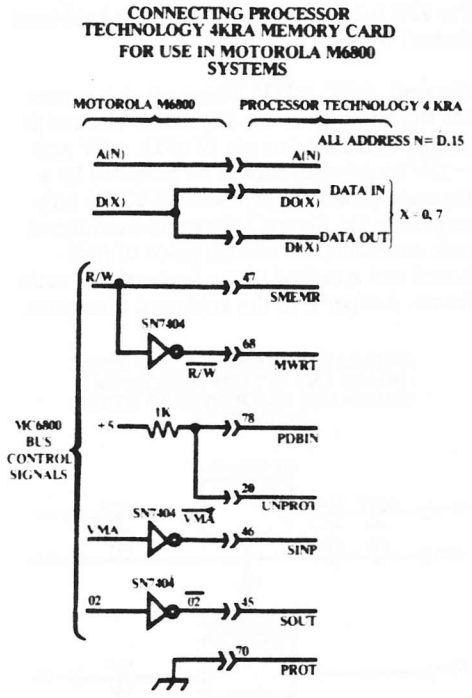
3P+S and the OP-80A

We've had a lot of inquiries about using the OP-80A high speed manual punchpaper tape reader with the 3P+S, so we asked Oliver Audio Engineering for the straight scoop on interconnecting it with the 3P+S interface. Note that the OP-80A acknowledge line must be programmed for a negative true (ack) signal. We've also heard from many of our customers that the use of black tape and rolled tape (instead of fanfold) produces more accurate data transfer, using the OP-80.



PT 4KRA Memory Boards and the Motorola M6800

The following diagram should help you expand your M6800 system with minimum headaches. Note that the M6800 can drive 2 4KRA cards at reduced clock rates without bus drivers; however, full buffering is recommended.



VDM-1 and the 6800 Microprocessor

The VDM-1 can live quite happily with your 6800 if a small amount of signal processing is performed externally.

Signals which may be directly used by the 6800 system are:

ADR0-ADR15 High-active address

DO0-DO7, DI0-DI7 High-active data lines. They may be connected in parallel to form a bi-directional data bus if necessary.

Signals which require conversion or re-interpretation:

PSYNC Connect to high-active VMA (Valid Memory Address)

Φ2 Connect to Φ1 clock

Pin 4 of IC 18 (74LS132) Break connection to Pin 3 and connect to Pin 2

SINP, SOUT Connect to the highest-order address bits which are "1" when registers are addressed. (Bits 14 and 15 are suggested.) Presence of a "1" on either line will cause the address decoder to switch its comparison to to the status port address. The status port will therefor respond to any address whose top six bits are set by jumpers (see your VDM-1 manual), and whose bits 8 and 9 are zero. The low-order 8 bits are not decodec during status port response.

$\overline{PWR}$ Externally generated signal consisting of VMA • Φ2 • R

MWRITE Inverted PWR

PDBIN Externally generated signal consisting of VMA • Φ2 • $\overline{R}$

XRDY, PRDY Not used

Note that +8V, +16V and −16V are still required to operate the board, and that 6800 systems designed for all 5-volt operation may have to be augmented. Simple unregulated supplies will perform well, but care should be taken not to exceed +10V on the +8V line to avoid excessive dissipation in the +5V regulator.

Announcing the SOL USERS GROUP!

The Sol Users Group was recently organized by members of the Homebrew Computer Club in Palo Alto; SUG is not affiliated in any way with Processor Tech-Technology. The purposes of SUG are to exchange software and other applications, and to create standards. If plenty of interest is shown, a Sol Newsletter will be published and sent to members.

If you own or have ordered a Sol, send your name, address, phone number and ideas to:

Bill Burns
4190 Maybell Way
Palo Alto, CA 94306
(Please don't call.)

What's the Best Monitor for your Sol or VDM-1?

That's one of the questions we're asked most frequently. We always recommend a black & white monitor designed for use with closed circuit TV systems or videotape recorders. Check your local Processor Technology dealer for good sources; they're up on that sort of thing.

If you want to use a regular black & white TV, try for a solid state model with an isolation transformer. In either case, remember that you don't have to spend a lot of money to get a decent picture, so let the rest of the family enjoy that big screen super color set.

PROCESSOR TECHNOLOGY
ACCESS.

Editor: Aram Attarian II
Publisher: Wible/Rampton Advertising, 727 15th Avenue, San Francisco, CA 94118
ACCESS is published approximately every six weeks. Subscription rate: $4 per year, from Processor Technology Corp., 6800 Hollis St., Emeryville, CA 94608.
ACCESS Copyright © February 1977 by Processor Technology Corp. All Rights Reserved. Material in this publication may not be reproduced in any form without permission from Processor Technology Corp.

terfacing a Keyboard and VDM with ALS-8, using the 3P+S.

To get your keyboard connection up and running, you'll need to connect 7 data lines, the keyboard strobe, and ground to the J2 connector of your 3P+S. Figure 1 shows the typical keyboard connections.

USING A KEYBOARD AND THE VDM WITH THE ALS-8

KEYBOARD INTERFACE

TYPICAL KEYBOARD CONNECTIONS

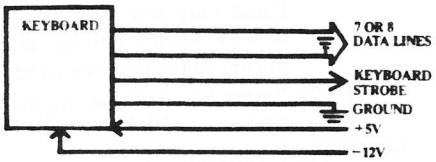


Fig. 1

The keyboard strobe. This circuit tells the processor when a key has been pressed. You'll need to install an additional IC (preferably in a 16 pin DIP socket) in the unused IC pad in the lower right corner of your 3P+S, and connect it to pin 7 of J2. It's used to condition the strobe. Use a 74LS109 or 74109. (Figure 2)

STROBE CONDITIONING CIRCUIT

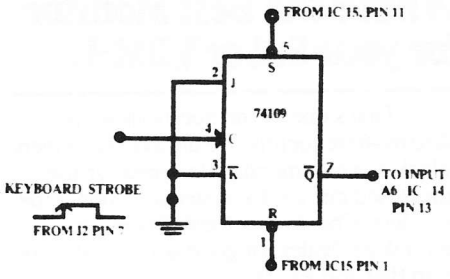


Fig. 2

To wire the 74109:

1) Connect pins 2 and 3 to ground.
2) Pins 16 (VCC) and 8 (GND) are already connected on board.
3) Connect pin 1 to pin 1 of IC 15 to provide a pull-up connection.
4) Connect pin 5 to pin 11 of IC 15 to reset the flip-flop when data has been accepted.
5) Connect pin 7 to a point on 3P+S leading to J2 pin 12. This point will go low when the strobe occurs.
6) Connect pin 4 to a point on 3P+S leading to J2 pin 7, the point where the strobe from the keyboard will be connected.

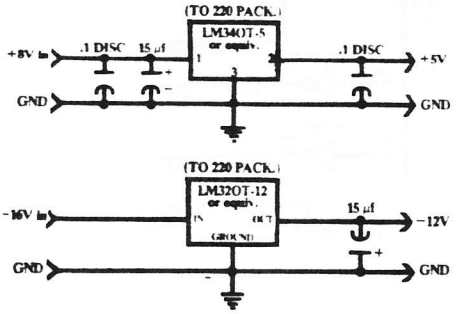
The data lines. Either seven or eight data lines are used to transmit the ASCII code for the key being pressed to the computer. They're connected to the B port inputs of the 3P+S in the following manner:

Keyboard Signal	3P+S J2 Connector	Data Bit
Bit 1	Z	D0
Bit 2	Y	D1
Bit 3	X	D2
Bit 4	W	D3
Bit 5	V	D4
Bit 6	U	D5
Bit 7	T	D6
Bit 8*	S	D7

*Pin J2-S is left unconnected if your keyboard doesn't have an eighth bit.

Ground, +5V, −12V. These provide power for the circuitry of the keyboard. Ground is simply connected to pin 12 of J2. +5V and −12V (regulated) should be provided by a separate power supply. Some KYBDS only require +5V. Figure 3 shows an example of one, assembled on a small piece of perf board and attached to the keyboard or main frame. Jumper it to the keyboard connector.

ASSEMBLE ON SMALL PIECE OF PERF BOARD AND ATTACH TO KYBD OR MAINFRAME AND JUMPER TO KYBD CONNECTOR



Jumper areas. Only two are important for the keyboard interface: Area A must be jumpered for address 0 (i.e., all to ground), and Area B must be jumpered from left to center, to set port A at 0 (status) and port B at 1 (data) to correspond to Processor Technology software.

Testing the interface with ALS-8.

1) Turn on the computer and examine location E000H. When you hit Run, the address lights should look like this:

A15	A14	A13	A12	A11	A10	A9	A8
•	•	o	•	o	o	o	•

• = Light on o = Light out

2) Be sure the keyboard is sending upper case characters. Type EXEC E024, then hit the carriage return key. Address light A13 should blink instantaneously at this point.
3) Type IODR/SYSIO/0 FE77, then hit the return key. Again, address light A13 should blink.
4) Now hit Space, Space, Return; wait a moment and then hit Control Z. The screen should now be blank.
5) Hit Control S. The word SPEED? should now appear in the lower left corner of the screen. If it doesn't, you may have to adjust the horizontal and vertical controls on the VDM to get this message onto the visible portion of the screen.
6) Type 1; you should get an automatic carriage return, then type DUMP 0 FFFF and hit Return. Memory will now dump on the VDM display. You can hit the Escape key to stop the dump.
7) Now you're ready to start programming with the ALS-8! Try some of the examples from the ALS-8 manual to familiarize yourself with its operation.

Hoare's Law of Large Programs
Inside every large program is a small program struggling to get out.

Good News for our Customers in Europe: VDM-1 and the European 50 Hz Standard

The European television standard maintains the same horizontal rate as the U.S. (15,750 Hz), but it defines a raster of 625 lines at a field rate of 50 Hz. The effect is to increase the number of scan lines on the screen.

It's quite easy to modify your VDM-1 to work on this standard. Simply disconnect pin 5 of IC 8 from pin 6, and reconnect it to ground (pin 4). This increases the modulus of the counter to 8 in the VDISP time, resulting in 4 extra character lines (52 scan lines) on the display. The total is now 312 scan lines per field, the equivalent of 624 per frame.

The field rate should be close enough to 50 Hz to reduce any swim effects to less than 0.1 Hz. You may have some trouble centering the display in the frame, because the standoff time to VSYNC from the bottom of the display is still on the 60 Hz standard. If the effect is objectionable, increase Resistor R 34 in series with the VPOS control.

Rule of Accuracy
When working toward the solution of a problem it always helps you to know the answer.
Sattinger's Law
It works better if you plug it in.

Newett Awl's Choo Choo Train
Or, Idle Fantasies on a VDM Screen

Once upon a time, in a curious little place, there was a Sol system and a programmer sitting around with nothing to do. So Newett Awl decided to tell his computer a bedtime story, and VDM-1 helped him out by drawing the pictures.

Try it on your system. Ol' Uncle Sol makes a great babysitter.

```
0000                    0000
0000                    0010
0000                    0020
0000                    0030
0000                    0040
0000                    0050
0000                    0060
0000                    0070
0000 AF                 0071  TRAIN   XRA    A       ONCE UPON A TIME,
0001 D3 C8              0072          OUT    0C8H  IN A CURIOUS LITTLE PLACE
0003 21 84 01           0080          LXI    H,SHED   THERE WAS
0006 01 00 08           0090          XSI    B,2048   ..A TINY TRAIN
0009 36 20              0100  EMPTY   MVI    M,20H   ..AND IT STAYED IN A
000B 23                 0110          INX    H       ..TINY SHED
000C 0B                 0120          DCX    B       ..THAT WAS ALL EMPTY
000D AF                 0130          XRA    A
000E A8                 0140          XRA    B
000F C2 09 00           0150          JNZ    EMPTY
0012 21 F7 02           0160          LXI    H,CLOUD   ..EXCEPT FOR A HUGE
0015 11 00 00           0170          LXI    D,0
0018 06 0F              0180          MVI    B,15 B
001A CD 50 00           0190          CALL   SMO1   I
001D 11 2B 00           0200          LXI    D,43    L
0020 06 0D              0210          MVI    B,13     L
0022 CD 50 00           0220          CALL   SMO1      O
0025 11 30 00           0230          LXI    D,48       W
0028 06 09              0240          MVI    B,9          Y
002A CD 50 00           0250          CALL   SMO1
002D 11 34 00           0260          LXI    D,52           C
0030 06 04              0270          MVI    B,4             L
0032 CD 50 00           0280          CALL   SMO1              O
0035 11 3B 00           0290          LXI    D,59               U
0038 06 02              0300          MVI    B,2                 D
003A CD 50 00           0310          CALL   SMO1
003D 11 3C 00           0320          LXI    D,60                    O
0040 06 01              0330          MVI    B,1                      F
0042 CD 50 00           0340          CALL   SMO1
0045 11 3E 00           0350          LXI    D,62                        S
0048 06 01              0360          MVI    B,1                          M
004A CD 50 00           0370          CALL   SMO1                          O
004D C3 59 00           0380          JMP    ENGIN                          K
0050                    0390  .                                               E
0050 19                 0400  SMO1    DAD    D
0051 36 6F              0410  SMO2    MVI    M,SMOKE   ..COUGH
0053 23                 0420          INX    H
0054 05                 0430          DCR    B                ..COUGH
0055 C2 51 00           0440          JNZ    SMO2
0058 C9                 0450          RET    .       COMING OUT OF THE STACK
                                              OF
0059                    0460  .
0059 21 A6 04           0470  ENGINE  LXI    H,SHED+322H  A TINY LOCOMOTIVE
005C 36 16              0480          MVI    M,16H  WITH A LITTLE SMOKESTAK
005E 21 A9 04           0490          LXI    H,SHED+325H  AND A LITTLE BELL
0061 36 07              0500          MVI    M,07H
0063 21 AB 04           0510          LXI    H,SHED+327H  ..AND A TINY DOME
0066 36 6E              0520          MVI    M,6EH
0068 2A 66 01           0530          LHLD   CAB1    ..AND A CAB
006B 22 AE 04           0540          SHLD   SHED+32AH WITH WINDOWS
006E 2A 68 01           0550          LHLD   CAB2    .. SO THAT YOU COULD
0071 22 B0 04           0560          SHLD   SHED+32CH SEE INTO WHERE THE
0074 2A 6A 01           0570          LHLD   CAB3    .. ENGINEER AND THE
0077 22 B2 04           0580          SHLD   SHED+32EH FIREMAN SAT.
007A 2A 6C 01           0590          LHLD   BOI1    .. IT HAD A BEAUTIFUL
007D 22 E5 04           0600          SHLD   SHED+361H POLISHED
0080 2A 6E 01           0610          LHLD   BOI2    .. BRASS
0083 22 E7 04           0620          SHLD   SHED+363H  BOILER
0086 2A 70 01           0630          LHLD   BOI3    ..  WITH
0089 22 E9 04           0640          SHLD   SHED+365H  THE NUMBER
008C 22 EB 04           0650          SHLD   SHED+367H  "99" ON THE
008F 2A 72 01           0660          LHLD   BOI4    ..  SIDE OF
0092 22 ED 04           0670          SHLD   SHED+369H  THE CAB
0095 2A 74 01           0680          LHLD   BOI5    ..  BUT YOU COULDN'T
0098 22 EF 04           0690          SHLD   SHED+36BH SEE EITHER THE
009B 2A 76 01           0700          LHLD   BOI6    .. ENGINEER OR THE
```

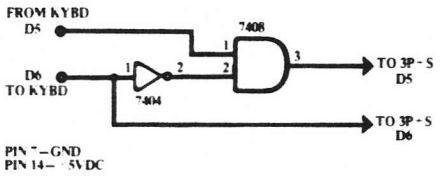
(continued on page 11)

Ups and Downs
or How to Type in Upper Case Only without Shifting

If you have a keyboard with both upper and lower case operation, the frequent shifts are a pain when you're entering alphanumeric data. Here's a simple cure that sets data bit D5 low when a lower case alphabetic character is output from the keyboard, thus presenting it to the computer as upper case.

You'll need two chips, either a 7404 and a 7408 or a 74LS04 and a 74LS08. (Either pair works fine.) Install them on a small piece of perf board as shown in the diagram, and tie all unused inputs to +5V.

The pins you'll be using on the 04 chip are 3, 5, 9, 11, and 13; on the 08 chip use 4, 5, 9, 10, 12, and 13.



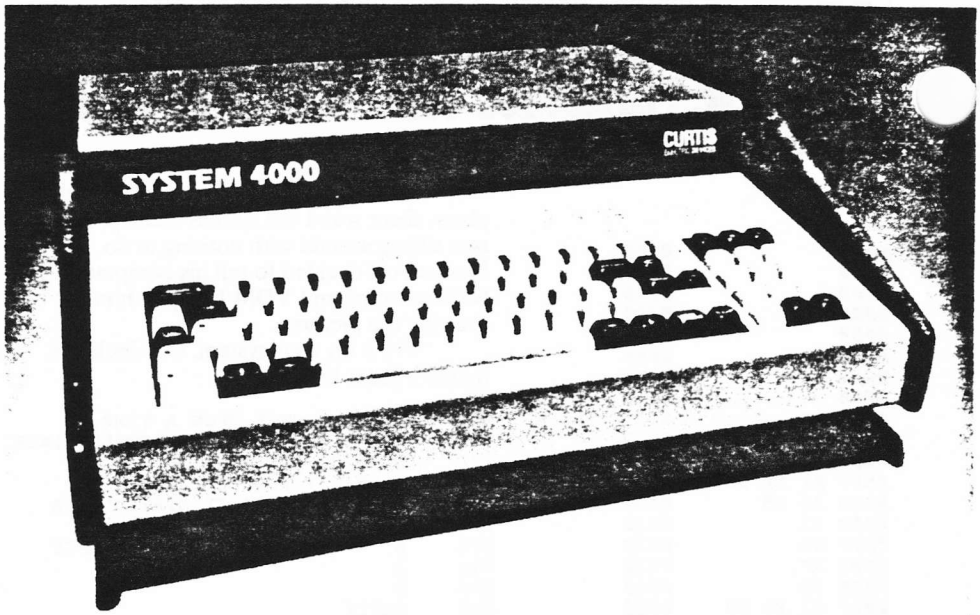
FLASH -- VDM Access Flicker Eliminated

That blasted flicker you get whenever the VDM memory is being frequently accessed occurs because the screen is blanked for a very short time whenever the processor reads or writes to memory. You can eliminate it by modifying the timing system so that VDM memory is accessed only when the beam of the picture tube is off the visible portion of the screen. The VDM has a timing signal that indicates this condition, and you can use it to synchronize access to display memory.

Connect a jumper wire from pin 13 of IC 39 to pin 5 of IC 39. This hooks up an unused section of IC 39 to DI bit 1. Connect a wire from IC 39 pin 14 to IC 15 pin 9. (This allows programs which access the VDM memory to use the timing signal.) Now when C8 is input, bit 1 will be low whenever the display memory can be accessed.

We'll explore the implications further in the next issue, and provide an example program or two.

Murphy's Law of Thermodynamics
Things get worse under pressure.
Lowery's Law
If it jams—force it. If it breaks, it needed replacing anyway.



Applications News
Ham Computer Based on Sol Terminal System

SYSTEM 4000 ham computer was developed by Curtis Electro Devices, Inc., Box 4090, Mountain View, Ca. 94040. The company makes ham keyers and an industrial line of PROM programmers. We asked the president, John G. Curtis, to comment on working with the Sol system; he did all of the hardware and software work himself, developing programs for the Sol Terminal on ALTAIR with ALS-8, VDM-1, 40K of RAM, 3P+1, Tarbell cassette system, Bytesaver, ASR-33 TTY, and Motorola video monitor. Jack's report was glowing:

"I had no previous experience or train-in software development. It was all learned on the fly from books and practical experience. (Try, try and try again!!!) . . . Not too much assistance was required. The ALS-8 is easy to use and the Simulator program is absolutely essential. The Editor is also absolutely necessary. (Everything is necessary!!). . .

"The Sol went together with a minimum of effort and trouble even though it was one of the first units actually put on the line. There were things which didn't work but with the help of PTC (especially Bob Marsh and Aram) things were put in order in a hurry.

"In my opinion, the Sol terminal is the most ideal small computer system available today (certainly at that price) and every needed feature is there. For our purposes it was absolutely ideal from every standpoint. We are now able to get the computer into the hamshack on a commercial basis. This has been predicted for some time, now it has happened."

System 4000 Ham Computer

The SYSTEM 4000 is a full scale standard desk top minicomputer specially equipped with firmware programs and interfaces for the amateur radio operator. Being a stand-alone computer, it can also run bu ness, educational, scientific or games programs.

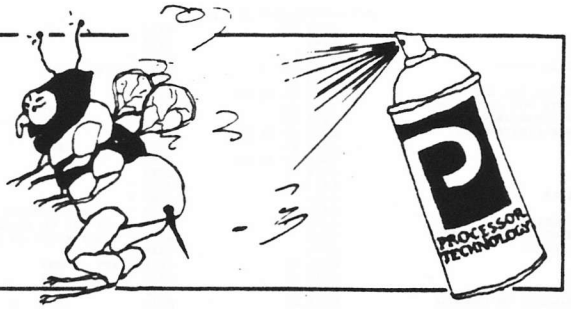
The SYSTEM 4000 is derived from the Sol terminal Computer and can take advantage of Processor Technology's programs and accessories. System 4000 uses the standard S-100 bus for plug-in accessory boards, and the owner can add or exchange PROM integrated circuits to update the system. Features:

Morse reader, capable of receiving code at speeds of 6-250 WPM (or higher). CRT or TTY output, selectable for upper and lower case.
Paddle keyer, with dot and dash memory, iambic and full self completion. CRT or TTY printout.
Keyboard keyer, sends Morse from keyboard. CRT or TTY output.
ASCII terminal, half or full duplex.

Complete details can be obtained by writing to: Curtis Electro Devices, Inc., P. O. Box 4090, Mountain View, Ca. 94040.

Murphy's Third Law
In any field of scientific endeavor, anything that can go wrong will go wrong.
Sevarenid's Law
The chief cause of problems is solutions.
Brooks's Law
Adding manpower to a late software project makes it later.
O'Tooles's Commentary on Murphy's Laws
Murphy was an optimist.

Bug Squad

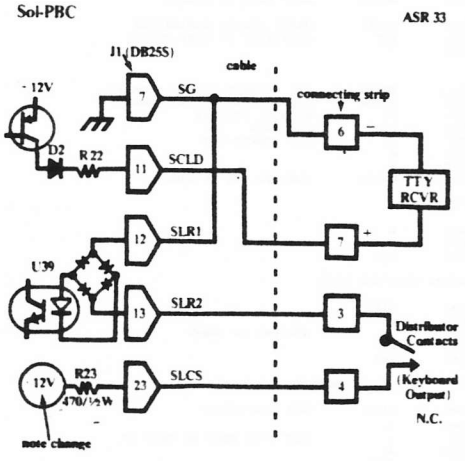


A major purpose of this newsletter is helping you stomp out the bugs that inevitably occur even in the best of systems. Bug Squad will be a regular feature. We'll tackle the problems we've encountered most frequently, and you're also invited to send in any problems that have been bugging you. Send solutions, too, if you've found them. Share the wealth.

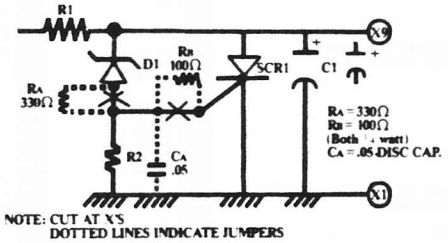
For starters, here's how to fix a few of the bugs that crept into early Sol systems:

Memory Protect/Unprotect Lines
***The bug:* Erratic behavior when early Sol-PC, Sol-10, and Sol-20 units are used with S-100 bus compatible memory modules.**
The squasher: On early Sol circuit boards, the protect (pin 70) and unprotect (pin 20) lines are floating. Simply ground bus line 70 on the Sol PCB itself to disable the memory protect signal.

Current Loop Source
***The bug:* R23, a 470 ohm ½w resistor, is incorrectly tied to +5 on early Sol PC boards, producing less than a full 20mA current to teletype connections.**
The squasher: Reconnect R23 to +12V, which is only ¼" away. Check to make sure that R29 remains connected to +5V. We suggest that the 20mA current loop connections (such as for ASR33) be made as follows:



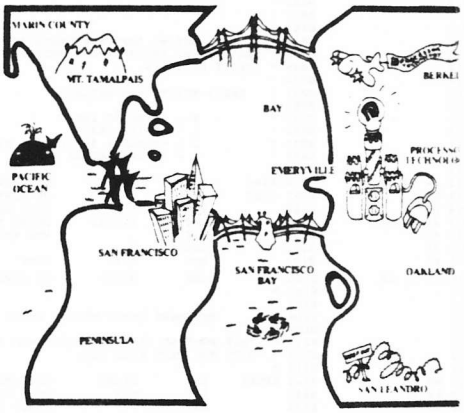
Accidental Avalanche
***The bug:* On some Sol Power Supply REG Boards, the SCR1 (MCR 106-2) in the +5 volt regulated supply will intermittently go into the avalanche mode. This draws sufficient current to drop the +5 volt to about +1V. This modification will prevent the accidental triggering, but will retain the overvoltage protection of the crowbar circuit.**
The squasher: Make the modification here by adding the components shown and cutting appropriate traces (as shown).



Law of Selective Gravity (The Buttered Side Down Law)
An object will fall so as to do the most damage.

Where the %&*# is Emeryville?

Emeryville, Ca., is located at the foot of the Oakland Bay Bridge, squeezed between Oakland and Berkeley. Neither wishes to claim it. Emeryville consists largely of mudflats and factories, the principle landmarks being a highrise apartment complex called Watergate (no relation), a shoreline assemblage of driftwood sculptures, and Processor Technology. Before the rise of Processor Technology, the main industry was legal gambling clubs. We, however, plan to put Emeryville on the map.



Consol Source Listing

For those of you who haven't seen it yet, here's the source list for the minimum Sol operating System, CONSOL. It provides all necessary display routines, along with standardized calling points for input/output operations.

```
--- CONSOL ---
COPYRIGHT 1976

C000                    0001 *
C000                    0002 *
C000                    0003 *
C000                    0004 *
C000                    0005 *
C000                    0006 *
C000                    0007 *
C000                    0008 *
C000                    0009 *
C000                    0010 *
C000                    0011 *
C000                    0012 *
C000                    0013 *                    SYSTEM SOLFTWARE
C000                    0014 *
C000                    0015 *                  <<>>     CONSOL     <<>>
C000                    0016 *
C000                    0017 *      VERSION:  1.0:5
C000                    0018 *      RELEASE: 10.11.76
C000                    0019 *
C000                    0020 *
C000                    0021 *
C000                    0022 *
C000                    0023 *
C000                    0024 *
C000                    0025 *     COMMANDS ARE ALSO AVAILABLE TO ENTER DATA TO MEMORY
C000                    0026 *  AND TO DISPLAY MEMORY DATA ON THE SCREEN.  AN EXECUTE
C000                    0027 *  COMMAND IS PROVIDED TO ALLOW PROGRAM EXCUTION OUTSIDE
C000                    0028 *  OF THE CONSOL PROM AND PROVISION IS MADE FOR CASSETTE
C000                    0029 *  TAPE LOAD OF PROGRAMS OR DATA.
C000                    0030 *
C000                    0031 *
C000                    0032 *  NOTE: CONSOL,SOLOS AND SOLED ARE REGISTERED TRADEMARKS
C000                    0033 *        OF:
C000                    0034 *                PROCESSOR TECHNOLOGY CORP.
C000                    0035 *                   EMERYVILLE,CALIF
C000                    0036 *
C000                    0037 *    THE FOLLOWING CODE IS THE PROPERTY OF PROCESSOR
C000                    0038 *  TECHNOLOGY CORP.  IT IS DISTRIBUTED ON A 'PERSONAL USE
C000                    0039 *  BASIS FOR THE BENIFIT OF SOL SYSTEM OWNERS.  ALL PORMS
C000                    0040 *  OF THE CODE ARE COPYRIGHT 1976 BY PROCESSOR TECHNOLOGY
C000                    0041 *  AND ALL RIGHTS THERIN ARE RESERVED.
C000                    0042 *
C000                    0043 *
C000                    0044 *
C000                    0045 *    AUTO-STARTUP CODE
C000                    0046 *
C000 00                 0047          DB        0           FOUR PHASE WONDER
C001 C3 65 C2           0048 BGIN     JMP       STRTA
C004                    0049 *
C004                    0050 *
C004                    0051 *
```

```
--- CONSOL ---
COPYRIGHT 1976

                        ;
                        ;
                        ;
```

```
C11E C3 13 C1           0337          JMP     PCUR
C121                    0338 *
C121                    0339 *   ROUTINE TO CALCULATE SCREEN ADDRESS
C121                    0340 *
C                       0341 *   ENTRY AT:    RETURNS:
                        0342 *
                        0343 *         VDADD  CURRENT SCREEN ADDRESS
                        0344 *         VDAD2  ADDRESS OF CURRENT LINE, CHAR 'C'
                        0345 *         VDAD   LINE 'A', CHARACTER POSITION 'C'
                        0346 *
                  00 C8 0347 VDADD    LDA     HCHAR       GET CHARACTER POSITION
                        0348          MOV     C,A         'C' KEEPS IT
C125 3A 01 C8           0349 VDAD2    LDA     LINE        LINE POSITION
C128 6F                 0350 VDAD     MOV     L,A         INTO 'L'
C129 3A 02 C8           0351          LDA     BOT         GET TEXT OFFSET
C12C 85                 0352          ADD     L           ADD IT TO THE LINE POSITION
C12D 0F                 0353          RRC     .           TIMES TWO
C12E 0F                 0354          RRC     .           MAKES FOUR
C12F 6F                 0355          MOV     L,A         L HAS IT
C130 E6 03              0356          ANI     3           MOD THREE FOR LATER
C132 57                 0357          MOV     D,A
C133 3E CC              0358          MVI     A,<VDMEM    LOW SCREEN OFFSET
C135 82                 0359          ADD     D
C136 67                 0360          MOV     H,A         NOW H IS DONE
C137 7D                 0361          MOV     A,L         TWIST L'S ARM
C138 E6 C0              0362          ANI     0C0H
C13A 81                 0363          ADD     C
C13B 6F                 0364          MOV     L,A
C13C C9                 0365          RET                 H & L ARE NOW PERVERTED
C13D                    0366 *
C13D                    0367 *   ROUTINE TO REMOVE CURSOR
C13D                    0368 *                                                  *
C13D CD 21 C1           0369 CREM     CALL    VDADD       GET CURRENT SCREEN ADDRESS
C140 7E                 0370          MOV     A,M
C141 E6 7F              0371          ANI     7FH         STRIP OFF THE CURSOR
C143 77                 0372          MOV     M,A
C144 C9                 0373          RET
C145                    0374 *
C145                    0375 *     ROUTINE TO BACKSPACE
C145                    0376 *
C145 CD 0D C1           0377 PBACK    CALL    PLEFT
C148 CD 21 C1           0378          CALL    VDADD       GET SCREEN ADDRESS
C14B 36 20              0379          MVI     M,' '       PUT A BLANK THERE
C14D C9                 0380          RET
C14E                    0381 *
C14E                    0382 *     ROUTINE TO PROCESS A CARRIAGE RETURN
C14E                    0383 *
C14E CD E3 C0           0384 PCR      CALL    CLINE       CLEAR FROM CURRENT CURSOR TO END OF LINE
C151 AF                 0385          XRA     A           REWIND IT
C152 C3 13 C1           0386          JMP     PCUR        AND STORE THE NEW VALUE
C155                    0387 *
C155                    0388 *   ROUTINE TO PROCESS LINEFEED
C155                    0389 *
C155 3A 01 C8           0390 PLF      LDA     LINE        GET LINE COUNT
C158 FE 0F              0391          CPI     15          ARE WE AT THE BOTTOM?
C15A D2 61 C1           0392          JNC     SC
C15D 3C                 0393          INR     A
C15E C3 C1 C0           0394          JMP     CUR         ONE MORE LINE UP
C161                    0395 *
C161 AF                 0396 SC       XRA     A
C162 C3 9C C0           0397          JMP     SROL
C165                    0398 *
C165                    0399 *
C165                    0400 *
C165                    0401 *     THIS TABLE DEFINES THE CHARACTERS FOR SPECIAL
C165                    0402 * PROCESSING.  IF THE CHARACTER IS NOT IN THE TABLE IT
C165                    0403 * GOES TO THE SCREEN.
C165                    0404 *
C1   9B                 0405 TBL      DB      CLEAR       SCREEN
C    S C0               0406          DW      PERSE
C                       0407          DB      UP          CURSOR
     C1                 0408          DW      PUP
     C0                 0409          DB      DOWN        "
                        0410          DW      PDOWN
C1C  0D C1              0411          DB      LEFT        "
C171 93                 0412          DW      PLEFT
C172 17 C1              0413          DB      RIGHT       "
C174 8E                 0414          DW      PRIT
C175 F3 C0              0415          DB      BORE        "
C177 00                 0416          DW      PHOME
C178 4E C1              0417          DB      CR          CARRIAGE RETURN
C17A 0A                 0418          DW      PCR
C17B 55 C1              0419          DB      LF          LINE FEED
C17D 5F                 0420          DW      PLF
C17E 45 C1              0421          DB      BACKS       BACK SPACE
C180 00                 0422          DW      PBACK
C181 B4 C1              0423          DB      MODE        MODE KEY
C183 00                 0424          DW      COMND
                        0425          DB      0           END OF TABLE
C184                    0426 *
C184                    0427 *   OUTPUT DEVICE TABLE
C184                    0428 *
C184 4C C0              0429 OTAB     DW      VDM01       VDM DRIVER
C186 4B C0              0430          DW      SEROT       SERIAL OUTPUT
C188 AD C1              0431          DW      ERROT       ERROR HANDLER (FOR CONSOL)
C18A AD C1              0432          DW      ERROT       ERROR HANDLER
C18C                    0433 *
C18C                    0434 *   INPUT DEVICE TABLE
C18C                    0435 *
C18C 27 C0              0436 ITAB     DW      KREA1       KEYBOARD INPUT
C18E 36 C0              0437          DW      SREA1       SERIAL INPUT
C190 AD C1              0438          DW      ERROT       ERROR HANDLER (FOR CONSOL)
C192 AD C1              0439          DW      ERROT       ERROR HANDLER
C194                    0440 *
C194                    0441 *
C194                    0442 *            COMMAND TABLE
C194                    0443 *
C194                    0444 *  THIS TABLE DESCRIBES THE VALID COMMANDS FOR CONSOL
C194                    0445 *
C194 54 45              0446 COMTAB   ASC     'TE'
C196 72 C2              0447          DW      TERM
C198 44 55              0448          ASC     'DU'
C19A A1 C2              0449          DW      DUMP
C19C 45 4E              0450          ASC     'EN'
C19E 87 C3              0451          DW      ENTER
C1A0 45 58              0452          ASC     'EX'
C1A2 35 C3              0453          DW      EXEC
C1A4 54 4C              0454          ASC     'TL'
C1A6 39 C3              0455          DW      TLOAD
C1A8 42 41              0456          ASC     'BA'        SPECIAL COMMAND TO EXECUTE 0
C1AA 00 00              0457          DW      0
C1AC 00                 0458          DB      0           END OF TABLE MARK
C1AD                    0459 *
C1AD                    0460 *
C1AD                    0461 *            CONSOL PORT ERROR HANDLER
C1AD                    0462 *
C1AD AF                 0463 ERROT    XRA     A
C1AE 32 04 C6           0464          STA     IPORT       DEFAULT TO SCREEN
C1B1 32 03 C8           0465          STA     OPORT       DEFAULT TO SCREEN
C1B4                    0466 *
C1B4                    0467 *
C1                      0468 *
C                       0469 *
C                       0470 *               ---  COMMAND MODE  ---
                        0471 *
C1                      0472 *
                        0473 *   THIS ROUTINE GETS AND PROCESSES COMMANDS
C1B4                    0474 *
C1B4 31 00 CC           0475 COMND    LXI     SP,SYSTP    SET STACK POINTER
C1B7 CD 12 C2           0476          CALL    PROMPT      PUT PROMPT ON SCREEN
C1BA CD C3 C1           0477          CALL    GCLIN       GET COMMAND LINE
C1BD CD E1 C1           0478          CALL    COPRC       PROCESS THE LINE
C1C0 C3 B4 C1           0479          JMP     COMND       OVER AND OVER
C1C3                    0480 *
C1C3                    0481 *
C1C3                    0482 *
C1C3                    0483 *    THIS ROUTINE READS A COMMAND LINE FROM THE SYSTEM
C1C3                    0484 *   KEYBOARD AND PROCESSES IT TO THE SCREEN.
C1C3                    0485 *
C1C3                    0486 *   C/R    TERMINATES THE SEQUENCE ERASING ALL CHARS TO THE
C1C3                    0487 *          RIGHT OF THE CURSOR
C1C3                    0488 *   L/F    TERMINATES THE SEQUENCE
C1C3                    0489 *   MODE   RESTARTS THE COMMAND LINE.
C1C3                    0490 *
C1C3 CD 28 C0           0491 GCLIN    CALL    KREAD       READ KEYBOARD
C1C6 FE 20              0492          CPI     20H
C1C8 47                 0493          MOV     B,A
C1C9 DA D2 C1           0494          JC      PROCS       PROCESS CONTROL CHARACTER
C1CC CD 4B C0           0495 CONT     CALL    VDMOT
C1CF C3 C3 C1           0496          JMP     GCLIN
C1D2                    0497 *
C1D2                    0498 *    PROCESS CONTROL KEYS
C1D2                    0499 *
C1D2 FE 0D              0500 PROCS    CPI     CR
C1D4 CA DD C1           0501          JZ      CRPRC       ERASE THE REMAINING LINE PRIOR TO RETURN
C1D7 FE 0A              0502          CPI     LF
C1D9 C8                 0503          RZ      .           IF NO GO PROCESS
C1DA C3 C3 C1           0504          JMP     GCLIN       NO CONTROL CHARS TO SCREEN
C1DD                    0505 *
C1DD                    0506 *
C1DD CD E3 C0           0507 CRPRC    CALL    CLINE       CLEAR REMAINING LINE
C1E0 C9                 0508          RET     .           NOW PROCESS
C1E1                    0509 *
C1E1                    0510 *
C1E1                    0511 *        FIND AND PROCESS COMMAND
C1E1                    0512 *
C1E1 CD 3D C1           0513 COPRC    CALL    CREM        REMOVE THE CURSOR
C1E4 0E 01              0514          MVI     C,1         SET FOR CHARACTER POSITION
C1E6 CD 25 C1           0515          CALL    VDAD2       GET SCREEN ADDRESS
C1E9 EB                 0516          XCHG
C1EA CD 32 C2           0517          CALL    SCHR        SCAN PAST BLANKS
C1ED CA 8E C3           0518          JZ      ERR1        NO COMMAND?
C1F0 EB                 0519          XCHG    .           HL HAS FIRST CHR
C1F1 11 94 C1           0520          LXI     D,COMTAB    POINT TO COMMAND TABLE
C1F4                    0521 *
C1F4                    0522 *   THIS ROUTINE SEARCHES THROUGH A TABLE, POINTED TO
C1F4                    0523 *  BY 'DE', FOR A DOUBLE CHARACTER MATCH OF THE 'HL
C1F4                    0524 *  MEMORY CONTENT.  IF NO MATCH IS FOUND THE SCAN ENDS
C1F4                    0525 *  BY PLACING A QUESTION MARK WITHIN THE SEARCH STRING.
C1F4                    0526 *
C1F4 1A                 0527 FDCOM    LDAX    D
C1F5 B7                 0528          ORA     A           TEST FOR TABLE END
C1F6 CA 8F C3           0529          JZ      ERR2        NOT FOUND..COMMAND ERROR
C1F9 E5                 0530          PUSH    H           SAVE START OF SCAN ADDRESS
C1FA BE                 0531          CMP     M           TEST FIRST CHR
C1FB 13                 0532          INX     D
C1FC C2 0B C2           0533          JNZ     NCOM
C1FF                    0534 *
C1FF 23                 0535          INX     H
C200 1A                 0536          LDAX    D
C201 BE                 0537          CMP     M           NOW SECOND CHARACTER
C202 C2 0B C2           0538          JNZ     NCOM        GOODNESS
C205                    0539 *
C205 C1                 0540          POP     B           CLEAR THE STACK
C206 EB                 0541          XCHG    .           DE HAS SCAN ADDRESS
C207 23                 0542          INX     H           HL HAS COMMAND ADDRESS
C208 C3 71 C0           0543          JMP     DISPT       DISPATCH TO IT
C20B                    0544 *
C20B                    0545 *
C20B 13                 0546 NCOM     INX     D           GO TO NEXT ENTRY
C20C 13                 0547          INX     D
C20D 13                 0548          INX     D
C20E E1                 0549          POP     H           GET BACK ORIGINAL ADDRESS
C20F C3 F4 C1           0550          JMP     FDCOM       CONTINUE SEARCH
C212                    0551 *
C212                    0552 *
C212                    0553 *        OUTPUT A CRLF FOLLOWED BY A PROMPT
C212                    0554 *    ( WITH CONSOL ALL OPERATIONS ARE ON THE SCREEN)
C212                    0555 *
C212 CD 1A C2           0556 PROMPT   CALL    CRLF
C215 06 3E              0557          MVI     B,'>'       THE PROMPT
C217 C3 4B C0           0558          JMP     VDMOT       PUT IT ON THE SCREEN
C21A                    0559 *
C21A 06 0A              0560 CRLF     MVI     B,LF        LINE FEED
C21C CD 4B C0           0561          CALL    VDMOT
C21F 06 0D              0562          MVI     B,CR        CARRIAGE RETURN
C221 C3 4B C0           0563          JMP     VDMOT       PUT IT OUT AND RETURN
C224                    0564 *
C224                    0565 *
C224                    0566 *   SCAN OVER UP TO 12 CHARACTERS LOOKING FOR A BLANK
C224                    0567 *
C224 0E 0C              0568 SBLK     MVI     C,12        MAXIMUM COMMAND STRING
C226 1A                 0569 SBLK1    LDAX    D
C227 FE 20              0570          CPI     BLANK
C229 CA 32 C2           0571          JZ      SCHR        GOT A BLANK NOW SCAN PAST IT
C22C 13                 0572          INX     D
C22D 0D                 0573          DCR     C           NO MORE THAN TWELVE
C22E C2 26 C2           0574          JNZ     SBLK1
C231 C9                 0575          RET     .           GO BACK WITH ZERO FLAG SET
C232                    0576 *
C232                    0577 *
C232                    0578 *   SCAN PAST UP TO 10 BLANK POSITIONS LOOKING FOR
C232                    0579 * A NON BLANK CHARACTER.
C232                    0580 *
C232 0E 0A              0581 SCHR     MVI     C,10        SCAN TO FIRST NON BLANK CHR WITHIN 10
C234 1A                 0582 SCHR1    LDAX    D           GET NEXT CHARACTER
C235 FE 20              0583          CPI     SPACE
C237 C0                 0584          RNZ     .           WE'RE PAST THEM
C238 13                 0585          INX     D           NEXT SCAN ADDRESS
C239 0D                 0586          DCR     C
C23A C8                 0587          RZ      .           COMMAND ERROR
C23B C3 34 C2           0588          JMP     SCHR1       KEEP LOOPING
C23E                    0589 *
C23E                    0590 *   THIS ROUTINE SCANS OVER CHARACTERS, PAST BLANKS AND
C23E                    0591 * CONVERTS THE FOLLOWING ADDRESS TO HEX.  ERRORS RETURN TO
C23E                    0592 * THE ERROR HANDLER.
C23E                    0593 *
C23E CD 24 C2           0594 SCONV    CALL    SBLK
C241 CA 8E C3           0595          JZ      ERR1
C244                    0596 *
C244                    0597 *   THIS ROUTINE CONVERTS ASCII DIGITS INTO BINARY FOLLOWING
C244                    0598 * A STANDARD HEX CONVERSION.  THE SCAN STOPS WHEN AN ASCII
C244                    0599 * SPACE IS ENCOUNTERED.  PARAMETER ERRORS REPLACE THE ERROR
C244                    0600 * CHARACTER ON THE SCREEN WITH A QUESTION MARK.
C244                    0601 *
C244 21 00 00           0602 SHEX     LXI     H,0         CLEAR H & L
C247 1A                 0603 SHE1     LDAX    D           GET CHARACTER
C248 FE 20              0604          CPI     20H         IS IT A SPACE?
C24A C8                 0605          RZ      .           IF SO
C24B                    0606 *
C24B 29                 0607 NCONV    DAD     H           MAKE ROOM FOR THE NEW ONE
C24C 29                 0608          DAD     H
C24D 29                 0609          DAD     H
C24E 29                 0610          DAD     H
C24F CD 5B C2           0611          CALL    NCOV1       DO THE CONVERSION
C252 D2 8E C3           0612          JNC     ERR1        NOT VALID HEXIDECIMAL VALUE
C255 85                 0613          ADD     L
C256 6F                 0614          MOV     L,A         MOVE IT IN
C257 13                 0615          INX     D           BUMP THE POINTER
C258 C3 47 C2           0616          JMP     SHE1
C25B                    0617 *
C25B D6 30              0618 NCOV1    SUI     48          REMOVE ASCII BIAS
C25D FE 0A              0619          CPI     10
C25F D8                 0620          RC      .           IF LESS THAN 9
C260 D6 07              0621          SUI     7           IT'S A LETTER??
C262 FE 10              0622          CPI     10H
```

```
C264 C9              8623          RET                     WITH TEST IN HAND
C265                 8624 *
C265                 8625 *
C265                 8626 *    SYSTEM START UP, CLEAR PART OF RAM AND SET STACK
C265                 8627 * POINTER, FALLING THROUGH TO TERMINAL MODE.
C265                 8628 *
C265 ,               8629 STRTA   XRA     A
C266 4,              8630         MOV     C,A             WE CLEAR THE FIRST 256 BYTES
C267 21 00 C8        8631         LXI     H,SYSRAM        POINT TO SYSTEM RAM
C26A                 8632 *
C26A 77              8633 CLERA   MOV     M,A
C26B 23              8634         INX     H
C26C 0C              8635         INR     C
C26D C2 6A C2        8636         JNZ     CLERA           CLEAR FIRST 256 BYTES
C270                 8637 *
C270                 8638 *
C270                 8639 *
C270                 8640 *                       TERM COMMAND
C270                 8641 *
C270                 8642 *  THIS ROUTINE GETS CHARACTERS FROM THE SYSTEM KEYBOARD
C270                 8643 *  AND OUTPUTS THEM TO THE SERIAL OUTPUT PORT.  IT IS
C270                 8644 *  INTENDED TO CONFIGURE THE SOL AS A STANDARD VIDEO
C270                 8645 *  TERMINAL.  COMMAND KEYS ARE NOT OUTPUT TO THE OUTPUT
C270                 8646 *  PORT BUT ARE INTERPRETED AS DIRECT SOL COMMANDS.
C270                 8647 *  THE MODE COMMAND, RECEIVED BY THE KEYBOARD, PUTS THE
C270                 8648 *  SOL IN THE COMMAND MODE.
C270                 8649 *
C270                 8650 *
C270 31 00 CC        8651 TERM    LXI     SP,SYSTP        SET STACK POINTER
C273 CD ED C3        8652         CALL    TOPF            SLOW DOWN THE TAPES
C276 CD C5 C0        8653         CALL    PERSE           CLEAR THE SCREEN
C279                 8654 *
C279 CD 22 C0        8655 KIN     CALL    KSTAT           IS THERE ONE WAITING?
C27C C2 90 C2        8656         JNZ     TIN
C27F DB FC           8657         IN      KDATA           GET THE CHARACTER
C281 47              8658         MOV     B,A
C282 E6 80           8659         ANI     80H             COMMAND KEY?
C284 CA 8D C2        8660         JZ      TOUT
C287 CD 4B C0        8661         CALL    VDMOT           PROCESS IT
C28A C3 90 C2        8662         JMP     TIN
C28D                 8663 *
C28D CD 41 C0        8664 TOUT    CALL    SDROT           OUTPUT IT TO THE SERIAL PORT
C290 CD 31 C0        8665 TIN     CALL    SSTAT           GET SERIAL STATUS
C293 CA 79 C2        8666         JZ      KIN             LOOP IF NOT
C296 DB F9           8667         IN      SDATA           GET DATA
C298 E6 7F           8668         ANI     7FH             NO HIGH BITS FROM HERE
C29A 47              8669         MOV     B,A             IT'S OUTPUT FROM 'B'
C29B CD 4B C0        8670         CALL    VDMOT           PUT IT ON THE SCREEN
C29E C3 79 C2        8671         JMP     KIN             LOOP OVER AND OVER
C2A1                 8672 *
C2A1                 8673 *
C2A1                 8674 *
C2A1                 8675 *                  DUMP COMMAND
C2A1                 8676 *
C2A1                 8677 *     THIS ROUTINE DUMPS CHARACTERS FROM MEMORY TO THE
C2A1                 8678 *  CURRENT OUTPUT DEVICE.  (WITH CONSOL ALL OUTPUT GOES TO
C2A1                 8679 *  THE SCREEN).  ALL VALUES ARE DESPLAYED AS ASCII HEX.
C2A1                 8680 *
C2A1                 8681 *  THE COMMAND FORM IS AS FOLLOWS:
C2A1                 8682 *
C2A1                 8683 *          Dump  addr1  addr2
C2A1                 8684 *
C2A1                 8685 *    THE VALUES FROM ADDR1 TO ADDR2 ARE THEN OUTPUT TO THE
C2A1                 8686 *  OUTPUT DEVICE.  IF ONLY ADDR1 IS SPECIFIED THEN THE
C2A1                 8687 *  VALUE AT THAT ADDRESS IS OUTPUT.
C2A1                 8688 *
C2A1 CD 3E C2        8689 DUMP    CALL    SCONV           SCAN TO FIRST ADDRESS AND CONVERT IT
C2A4 E5              8690         PUSH    H               SAVE THE VALUE
C2A5 CD 32 C2        8691         CALL    SCHR            GET THE NEXT
C2A8 '               8692         POP     H
C2A9       C2        8693         JZ      POVER           NO SECOND VALUE
C2AC                 8694         PUSH    H
C2AD C.  . C2        8695         CALL    SHEX            GET SECOND
C2B0 D1              8696         POP     D               THIS IS THE FIRST
C2B1 C3 B6 C2        8697         JMP     NPASS           MIND BENDERS
C2B4                 8698 *
C2B4 54              8699 POVER   MOV     D,H             NO SECOND PARAMETER COPY FIRST TO DE
C2B5 5D              8700         MOV     E,L
C2B6 EB              8701 NPASS   XCHG    .               HL HAS START, DE HAS END
C2B7                 8702 *
C2B7 CD 1A C2        8703 DLOOP   CALL    CRLF
C2BA DB FC           8704         IN      KDATA
C2BC FE 8F           8705         CPI     MODE            'MODE KEY' WILL ESCAPE THE DUMP
C2BE CA B4 C1        8706         JZ      COMND
C2C1 CD DC C2        8707         CALL    ADOUT           OUTPUT ADDRESS
C2C4 CD E5 C2        8708         CALL    BOUT            ANOTHER SPACE TO KEEP IT PRETTY
C2C7 0E 10           8709         MVI     C,16            VALUES PER LINE
C2C9                 8710 *
C2C9 7E              8711 DLP1    MOV     A,M             GET THE CHR
C2CA C5              8712         PUSH    B               SAVE VALUE COUNT
C2CB CD E2 C2        8713         CALL    HBOUT           SEND IT OUT WITH A BLANK
C2CE CD B2 C3        8714         CALL    ACOMP           COMPARE ADDRESSES
C2D1 D2 B4 C1        8715         JNC     COMND           ALL DONE
C2D4 C1              8716         POP     B               VALUES PER LINE
C2D5 23              8717         INX     H
C2D6 0D              8718         DCR     C               BUMP THE LINE COUNT
C2D7 C2 C9 C2        8719         JNZ     DLP1            NOT ZERO IF MORE FOR THIS LINE
C2DA C3 B7 C2        8720         JMP     DLOOP           DO A LFCR BEFORE THE NEXT
C2DD                 8721 *
C2DD                 8722 *    OUTPUT HL AS HEX 16 BIT VALUE
C2DD                 8723 *
C2DD 7C              8724 ADOUT   MOV     A,H             H FIRST
C2DE CD EA C2        8725         CALL    HEOUT
C2E1 7D              8726         MOV     A,L             THEN L FOLLOWED BY A SPACE
C2E2                 8727 *
C2E2 CD EA C2        8728 HBOUT   CALL    HEOUT
C2E5 06 20           8729 BOUT    MVI     B,' '
C2E7 C3 4B C0        8730         JMP     VDMOT           CONSOL PUTS IT ON THE SCREEN
C2EA                 8731 *
C2EA 4F              8732 HEOUT   MOV     C,A             GET THE CHARACTER
C2EB 0F              8733         RRC
C2EC 0F              8734         RRC             .       MOVE THE HIGH FOUR DOWN
C2ED 0F              8735         RRC
C2EE 0F              8736         RRC
C2EF CD F3 C2        8737         CALL    HEOU1           PUT THEM OUT
C2F2 79              8738         MOV     A,C             THIS TIME THE LOW FOUR
C2F3                 8739 *
C2F3 E6 0F           8740 HEOU1   ANI     0FH             FOUR ON THE FLOOR
C2F5 C6 30           8741         ADI     48              WE WORK WITH ASCII HERE
C2F7 FE 3A           8742         CPI     58              0-9?
C2F9 DA FE C2        8743         JC      OUTH            YUP!
C2FC C6 07           8744         ADI     7               MAKE IT A LETTER
C2FE 47              8745 OUTH    MOV     B,A             OUTPUT IT FROM REGISTER 'B'
C2FF C3 4B C0        8746         JMP     VDMOT
C302                 8747 * -
C302                 8748 *    COMPARE DE AND HL
C302                 8749 *
C302 7D              8750 ACOMP   MOV     A,L
C303 93              8751         SUB     E
C304 7C              8752         MOV     A,H
C305 9A              8753         SBB     D
C306 C9              8754         RET     .               FLAGS ARE SET
C307                 8755 *
C307                 8756 *
C307                 8757 *                  ENTER COMMAND
C307                 8758 *
C307                 8759 *    THIS ROUTINE GETS VALUES FROM THE KEYBOARD AND ENTERS
C307                 8760 * THEM INTO MEMORY.  THE INPUT VALUES ARE SCANNED FOLLOWING
C307                 8761 * A STANDARD 'GCLIN' INPUT SO ON SCREEN EDITING MAY TAKE
C307                 8762 * PLACE PRIOR TO THE LINE TERMINATOR.  A BACK SLASH '/'
C307                 8763 * ENDS THE ROUTINE AND RETURNS CONTROL TO THE COMMAND MODE.
C307                 8764 *
C307 CD 3E C2        8765 ENTER   CALL    SCONV           SCAN OVER CHARS AND GET ADDRESS
C30A E5              8766         PUSH    H               SAVE ADDRESS
C30B                 8767 *
C30B CD 1A C2        8768 ENLOP   CALL    CRLF
C30E 06 3A           8769         MVI     B,':'
C310 CD CC C1        8770         CALL    CONT            GET LINE OF INPUT
C313 CD 3D C1        8771         CALL    CREM            REMOVE THE CURSOR
C316 0E 01           8772         MVI     C,1             START SCAN
C318 CD 25 C1        8773         CALL    VDAD2           GET ADDRESS
C31B EB              8775         XCHG    .               ....TO DE
C31C                 8776 *
C31C 0E 03           8777 ENLO1   MVI     C,3             NO MORE THAN THREE SPACES BETWEEN VALUES
C31E CD 34 C2        8000         CALL    SCHR1 SCAN TO NEXT VALUE
C321 CA 0B C3        8778         JZ      ENLOP           LAST ENTRY FOUND START NEW LINE
C324 1A              8779 ENLO2   LDAX    D               GET THE CHR
C325 FE 2F           8780         CPI     '/'             COMMAND TERMINATOR?
C327 CA B4 C1        8781         JZ      COMND           IF SO...
C32A CD 44 C2        8782         CALL    SHEX            CONVERT VALUE
C32D 7D              8783         MOV     A,L             GET LOW PART AS CONVERTED
C32E E1              8784         POP     H               GET MEMORY ADDRESS
C32F 77              8785         MOV     M,A             PUT IN THE VALUE
C330 23              8786         INX     H
C331 E5              8787         PUSH    H               BACK GOES THE ADDRESS
C332 C3 1C C3        8788         JMP     ENLO1           CONTINUE THE SCAN
C335                 8789 *
C335                 8790 *
C335                 8791 *
C335                 8792 *                  EXECUTE COMMAND
C335                 8793 *
C335                 8794 *   THIS ROUTINE GETS THE FOLLOWING PARAMETER AND DOES A
C335                 8795 * PROGRAM JUMP TO THE LOCATION GIVEN BY IT.  IF PROPER
C335                 8796 * STACK OPERATIONS ARE USED WITHIN THE EXTERNAL PROGRAM
C335                 8797 * IT CAN DO A STANDARD 'RET'URN TO THE CONSOL COMMAND MODE.
C335                 8798 *
C335                 8799 *
C335 CD 3E C2        8800 EXEC    CALL    SCONV           SCAN PAST BLANKS AND GET PARAMETER
C338 E9              8801         PCHL    .               GO.....(AND TAKE NOTE)
C339                 8802 *
C339                 8803 *
C339                 8804 *
C339                 8805 *                  TAPE LOAD COMMAND
C339                 8806 *
C339                 8807 *     THIS ROUTINE READS FROM EITHER TAPE UNIT PLACING
C339                 8808 * THE READ DATA INTO MEMORY.  WHILE SPACE WITHIN CONSOL
C339                 8809 * DOES NOT ALLOW FOR 'STANDARD' TAPE ROUTINES THIS
C339                 8810 * COMMAND WILL LOAD SOL- BASIC5 AND OTHER STANDARD SOL
C339                 8811 * SYSTEM SOLFTWARE FOR DIRECT EXECUTION.
C339                 8812 *
C339                 8813 *
C339 CD 24 C2        8814 TLOAD   CALL    SBLK            SCAN TO SPEED PARAMETER
C33C CA 4A C3        8815         JZ      DFLT            DEFAULT TO HIGH SPEED IF NONE
C33F CD 44 C2        8816         CALL    SHEX            CONVERT IT
C342 7D              8817         MOV     A,L             GET VALUE
C343 E6 01           8818         ANI     1               ONLY BIT ZERO COUNTS
C345 3E 20           8819         MVI     A,32            PRETEND ITS SLOW
C347 C2 4B C3        8820         JNZ     SETSP
C34A                 8821 *
C34A AF              8822 DFLT    XRA     A               MAKE IT FAST
C34B F6 C0           8823 SETSP   ORI     TAPE1+TAPE2     CONSOL STARTS BOTH TAPES
C34D D3 FA           8824         OUT     STAPT           START TAPES AND SELECT SPEED
C34F CD F1 C3        8825         CALL    DELAY           WAIT WHILE THE TAPE UNIT WINDS UP
C352 DB FB           8826         IN      TDATA           CLEAR THE UART FLAGS
C354                 8827 *
C354 CD 94 C3        8828 TLOD1   CALL    RHEAD           READ PAST HEADER
C357 C2 54 C3        8829         JNZ     TLOD1           IF ERROR START OVER
C35A                 8830 *
C35A 2A 0C C8        8831         LHLD    BLOCK           GET BLOCK SIZE
C35D EB              8832         XCHG    .               ...TO DE
C35E 2A 0E C8        8833         LHLD    LOADR           GET LOAD ADDRESS
C361                 8834 *
C361 7A              8835 LOLOOP  MOV     A,D             GET COUNT
C362 B3              8836         ORA     E
C363 CA ED C3        8837         JZ      TOPF            COUNT IS ZERO-TURN OFF TAPE AND RETURN
C366 01 00 FF        8838         LXI     B,-256          THIS MANY PRIOR TO CRC TEST
C369 EB              8839         XCHG    .               COUNT TO HL
C36A 09              8840         DAD     B               A LITTLE MATH
C36B D2 E1 C3        8841         JNC     LBLK            NO CARRY, IT'S THE LAST BLOCK
C36E 06 00           8842         MVI     B,0             256 TO READ
C370                 8843 *
C370 0E 00           8844 RDBLK   MVI     C,0             ZERO THE CRC
C372 EB              8845         XCHG    .               ROUND ROBIN
C373                 8000 *
C373 CD C6 C3        8846 RTBYT   CALL    TAPIN           GET CHARACTER
C376 77              8847         MOV     M,A             STORE IT
C377 23              8848         INX     H               BUMP MEMORY LOCATION
C378 A9              8849         XRA     C               UPDATE THE CRC
C379 2F              8850         CMA
C37A 91              8851         SUB     C
C37B 4F              8852         MOV     C,A             STORE THE NEW
C37C 05              8853         DCR     B               COUNT DOWN
C37D C2 73 C3        8854         JNZ     RTBYT           STILL MORE IF NOT ZERO
C380                 8855 *
C380 CD C1 C3        8856         CALL    CRCCK           CHECK CRC AND FALL THROUGH TO ERROR IF NO GOOD
C383 CA 61 C3        8857         JZ      LOLOOP          TEST OK
C386                 8858 *
C386 06 07           8859 TERR    MVI     B,'G'-40H       BELL CHARACTER
C388 CD 4B C0        8860         CALL    VDMOT           PUT IT ON THE SCREEN
C38B C3 B4 C1        8861         JMP     COMND
C38E                 8862 *
C38E                 8863 *
C38E                 8864 *    CONSOL ERROR HANDLER
C38E                 8865 *
C38E EB              8866 ERR1    XCHG    .               GET SCAN ADDRESS
C38F 36 3F           8867 ERR2    MVI     M,'?'           PUT A QUESTION MARK THERE
C391 C3 B4 C1        8868         JMP     COMND           AND GO TO COMMAND MODE
C394                 8869 *
C394                 8870 *   READ THE HEADER
C394                 8871 *
C394 06 0A           8872 RHEAD   MVI     B,10            FIND 10 NULLS
C396 DB FA           8873 RHEA1   IN      STAPT           GET A BYTE
C398 E6 40           8874         ANI     TDR
C39A CA 96 C3        8875         JZ      RHEA1
C39D DB FB           8876         IN      TDATA           IGNORE ERROR CONDITIONS
C39F B7              8877         ORA     A               ZERO?
C3A0 C2 94 C3        8878         JNZ     RHEAD
C3A3 05              8879         DCR     B
C3A4 C2 96 C3        8880         JNZ     RHEA1           LOOP UNTIL 10 IN A ROW
C3A7                 8881 *
C3A7                 8882 *    WAIT FOR THE START CHARACTER
C3A7                 8883 *
C3A7 CD C6 C3        8884 SOHL    CALL    TAPIN
C3AA 3D              8885         DCR     A
C3AB C2 A7 C3        8886         JNZ     SOHL            WAIT FOR A '1'
C3AE                 8887 *
C3AE                 8888 *    NOW GET THE HEADER
C3AE                 8889 *
C3AE 21 05 C8        8890         LXI     H,THEAD         POINT TO BUFFER
C3B1 01 00 10        8891         LXI     B,HLEN*256      LENGTH OF HEADER IN 'B',C<0
C3B4                 8892 *
C3B4 CD C6 C3        8893 RHED1   CALL    TAPIN           GET BYTE
C3B7 77              8894         MOV     M,A             STORE IT
C3B8 23              8895         INX     H               INCREMENT ADDRESS
C3B9 A9              8896         XRA     C               NOW CALCULATE THE CRC
C3BA 2F              8897         CMA     .               INSIDE OUT AND UPSIDE DOWN
C3BB 91              8898         SUB     C               SQUEEZE IT
C3BC 4F              8899         MOV     C,A             AND SAVE AGAIN
C3BD 05              8900         DCR     B               WHOLE HEADER YET?
C3BE C2 B4 C3        8901         JNZ     RHED1           LOOP UNTIL DONE
C3C1                 8902 *
C3C1                 8903 *    THIS ROUTINE GETS THE NEXT BYTE AND COMPARES IT
C3C1                 8904 * TO THE VALUE IN REGISTER C.  THE FLAGS ARE SET ON
C3C1                 8905 * RETURN.
C3C1                 8906 *
C3C1 CD C6 C3        8907 CRCCK   CALL    TAPIN           GET CRC BYTE
```

```
C3C4 B9          0908          CMP     C          COMPARE IT WITH CALCULATED
C3C5 C9          0909          RET
C3C6             0910 *
C3C6             0911 *
C3C6             0912 *    THIS ROUTINE GETS THE NEXT AVAILABLE BYTE FROM THE
C3C6             0913 *  TAPE.  WHILE WAITING FOR THE BYTE THE KEYBOARD IS TESTED
C3C6             0914 *  FOR A 'MODE' COMMAND.  IF RECEIVED THE TAPE LOAD IS
C3C6             0915 *  TERMINATED AND A RETURN TO THE COMMAND MODE IS MADE.
C3C6             0916 *
C3C6 DB FA       0917 TAPIN    IN      STAPT      CHECK STATUS
C3C8 E6 40       0918          ANI     TDR
C3CA C2 D7 C3    0919          JNZ     TREDY      ONE IS AVAILABLE
C3CD DB FC       0920          IN      KDATA      CHECK FOR MODE WHILE WE'RE WAITING
C3CF FE 80       0921          CPI     MODE
C3D1 CA 84 C1    0922          JZ      COMMD      MODE WAS GIVEN..ABORT OPERATION
C3D4 C3 C6 C3    0923          JMP     TAPIN      NOT MODE...STAY IN LOOP
C3D7             0924 *
C3D7 DB FA       0925 TREDY    IN      STAPT
C3D9 E6 18       0926          ANI     TFE+TOE    DATA ERROR?
C3DB C2 86 C3    0927          JNZ     TERR       IF FRAMING OR OVERRUN ERROR
C3DE DB FB       0928          IN      TDATA      GET THE DATA
C3E0 C9          0929          RET
C3E1             0930 *
C3E1             0931 * THIS ROUTINE CALCULATES THE LENGTH OF THE LAST BLOCK
C3E1             0932 *
C3E1 01 FF FF    0933 LBLK     LXI     B,-1
C3E4 09          0934          DAD     B          COMPLEMENT HL
C3E5 23          0935          INX     H          .......TWO'S
C3E6 45          0936          MOV     B,L        LENGTH TO REGISTER B
C3E7 21 00 00    0937          LXI     H,0        TELL DE WE'RE DONE
C3EA C3 78 C3    0938          JMP     RDBLK      ONWARD TO THE END
C3ED             0939 *
C3ED             0940 *    THIS ROUTINE TURNS THE TAPE UNITS OFF
C3ED             0941 *
C3ED AF          0942 TOFF     XRA     A
C3EE D3 FA       0943          OUT     STAPT      GIVE COMMAND
C3F0 C9          0944          RET     .          AND GRIND TO A SLOW STOP
C3F1             0945 *
C3F1             0946 *
C3F1 11 00 00    0947 DELAY    LXI     D,0        START LOOP
C3F4 1B          0948 DLOP1    DCX     D          DOWN COUNT
C3F5 7A          0949          MOV     A,D
C3F6 B3          0950          ORA     E          TEST FOR ZERO
C3F7 C2 F4 C3    0951          JNZ     DLOP1      IF NOT
C3FA C9          0952          RET
C3FB             0953 *
C3FB             0954 *
C3FB             0955 *
C3FB             0956 *
C3FB             0957 *                 <<  SOL SYSTEM EQUATES >>
C3FB             0958 *
C3FB             0959 *
C3FB             0960 *          VDM PARAMETERS
C3FB             0961 *
C3FB             0962 VDMEM    EQU     0CC00H     VDM SCREEN MEMORY
C3FB             0963 *
C3FB             0964 *
C3FB             0965 *          KEYBOARD SPECIAL KEY ASSIGNMENTS
C3FB             0966 *
C3FB             0967 DOWN     EQU     9AH
C3FB             0968 UP       EQU     97H
C3FB             0969 LEFT     EQU     81H
C3FB             0970 RIGHT    EQU     93H
C3FB             0971 LOADK    EQU     8CH        LOAD KEY
C3FB             0972 MODE     EQU     80H
C3FB             0973 CLEAR    EQU     8BH
C3FB             0974 HOME     EQU     80EH
C3FB             0975 BACKS    EQU     5FH        BACKSPACE
C3FB             0976 LF       EQU     10
C3FB             0977 CR       EQU     13
C3FB             0978 BLANK    EQU     ' '
C3FB             0979 SPACE    EQU     BLANK
C3FB             0980 CX       EQU     'X'-40H
C3FB             0981 *
C3FB             0982 *          PORT ASSIGNMENTS
C3FB             0983 *
C3FB             0984 STAPT    EQU     0FAH       STATUS PORT GENERAL
C3FB             0985 SERST    EQU     0F8H       SERIAL STATUS PORT
C3FB             0986 SDATA    EQU     0F9H       SERIAL DATA
C3FB             0987 TDATA    EQU     0FBH       TAPE DATA
C3FB             0988 KDATA    EQU     0FCH       KEYBOARD DATA
C3FB             0989 PDATA    EQU     0FDH       PARALLEL DATA
C3FB             0990 DSTAT    EQU     0FEH       VDM DISPLAY PARAMETER PORT
C3FB             0991 SENSE    EQU     0FFH       SENSE SWITCHES
C3FB             0992 *
C3FB             0993 *
C3FB             0994 *
C3FB             0995 *          BIT ASSIGNMENT MASKS
C3FB             0996 *
C3FB             0997 SCD      EQU     1          SERIAL CARRIER DETECT
C3FB             0998 SDSR     EQU     2          SERIAL DATA SET READY
C3FB             0999 SPE      EQU     4          SERIAL PARITY ERROR
C3FB             1000 SFE      EQU     8          SERIAL FRAMING ERROR
C3FB             1001 SOE      EQU     16         SERIAL OVERRUN ERROR
C3FB             1002 SCTS     EQU     32         SERIAL CLEAR TO SEND
C3FB             1003 SDR      EQU     64         SERIAL DATA READY
C3FB             1004 STBE     EQU     128        SERIAL TRANSMITTER BUFFER EMPTY
C3FB             1005 *
C3FB             1006 KDR      EQU     1          KEYBOARD DATA READY
C3FB             1007 PDR      EQU     2          PARALLEL DATA READY
C3FB             1008 PEDR     EQU     4          PARALLEL DEVICE READY
C3FB             1009 TFE      EQU     8          TAPE FRAMING ERROR
C3FB             1010 TOE      EQU     16         TAPE OVERFLOW ERROR
C3FB             1011 TDR      EQU     64         TAPE DATA READY
C3FB             1012 TTBE     EQU     128        TAPE TRANSMITTER BUFFER EMPTY
C3FB             1013 *
C3FB             1014 SOK      EQU     1          SCROLL OK FLAG
C3FB             1015 *
C3FB             1016 TAPE1    EQU     64         TAPE ONE 'ON' BIT
C3FB             1017 TAPE2    EQU     128        TAPE TWO
C3FB             1018 *
C3FB             1019 *
C3FB             1020 *
C3FB             1021 *
C3FB             1022 *               SOL SYSTEM GLOBAL AREA
C3FB             1023 *
C3FB             1024          ORG     0C800H     START OF 1K RAM AREA
C800             1025 *
C800             1026 SYSRAM   EQU     $          START OF SYSTEM RAM
C800             1027 SYSTP    EQU     $+1024     STACK IS AT THE TOP
C800             1028 *
C800             1029 *
C800             1030 *          CONSOL PARAMETER AREA
C800             1031 *
C800             1032 NCHAR    DS      1          CURRENT CHARACTER POSITION
C801             1033 LINE     DS      1          CURRENT LINE POSITION
C802             1034 BOT      DS      1          BEGINNING OF TEXT DISPLACEMENT
C803             1035 OPORT    DS      1          OUTPUT PORT
C804             1036 IPORT    DS      1          INPUT PORT
C805             1037 *
C805             1038 *
C805             1039 *
C805             1040 THEAD    DS      5          NAME
C80A             1041          DS      1          THIS BYTE MUST BE ZERO
C80B             1042 HTYPE    DS      1          TYPE
C80C             1043 BLOCK    DS      2          BLOCK SIZE
C80E             1044 LOADR    DS      2          LOAD ADDRESS
C810             1045 XEQAD    DS      2          AUTO EXECUTE ADDRESS
C812             1046 HSPR     DS      3          SPARES
C815             1047 *
C815             1048 HLEN     EQU     $-THEAD    LENGTH OF HEADER
C815             1049 *
C815             1050 *

ACORP  C382   ADOUT  C2DD   AINP   C816   AOUT   C807
BACKS  005F   AGIN   C801   BLANK  0020   BLOCK  C80C
BOT    C802   BOUT   C2E5   CHAR   C07B   CLEAR  008B
CLEAR  C26A   CLIN1  C8E9   CLINE  C0E3   COMND  C1B4
CUNTA  C194   CONT   C1CC   COPRC  C1E1   CR     000D
CRCCE  C3C1   CREM   C13D   CRLF   C21A   CRPRC  C1DD
CUR    C0C1   CX     0018   DELAY  C3F1   DFLT   C34A
DISPT  C871   DLOOP  C2B7   DLOP1  C3F4   DLP1   C2C9
DOWN   009A   DSTAT  00FE   DUMP   C2A1   ENLO1  C31C
ENLO2  C324   ENLOP  C30B   ENTER  C3W7   ERAS1  C0CA
ERAS2  C0D6   ERAS3  C0DD   ERR1   C3BE   ERR2   C3BF
ERROT  C1AD   ESCC   C335   FDCOM  C1F4   GCLIN  C1C3
GOBAC  C056   GOHK   C05D   HBOUT  C2E2   HCONV  C24B
HCOV1  C25B   HLOU1  C2F3   HEOUT  C2EA   HLEN   0018
HOME   008E   HSPR   C812   HTYPE  C80B   IPORT  C804
ITAB   C18C   KDATA  00FC   KDR    0001   KIN    C279
KMEA1  C027   KMEAD  C028   KSTAT  C022   LBLK   C3E1
LF1    0081   LF     000A   LINE   C801   LOADK  008C
LOADR  C80E   LOLEO  C361   MODE   0080   MCHAR  C800
MCOM   C208   NEXT   C876   NPASS  C2B6   OCHAR  C001
OK     C8AD   OPORT  C803   OTAB   C104   OUTH   C2FE
PBACE  C145   PCR    C145   PCUR   C113   PDATA  00FD
PDOWN  C8FA   PDR    0002   PEASE  C0C5   PHOME  C0F3
PLEFT  C10D   PLF    C155   POVER  C2B4   PRIT   C117
PMOCS  C1D2   PROMP  C212   PUP    C104   PXDR   0004
RDBLK  C378   RMEA1  C396   RMEAD  C394   RMED1  C3B4
RIGHT  0093   RTBYT  C373   SBLK   C224   SBLK1  C226
SC     C161   SCD    0001   SCHK   C232   SCHK1  C234
SCOMV  C23A   SCROL  C498   SCIS   0020   SDATA  00F9
SDR    0040   SEMOT  C441   SDSR   0002   SENSE  00FF
SERST  C048   SERST  00F8   SETSP  C34B   SFE    0008
SME1   C247   SMEM   C244   SINP   C813   SOE    0010
SOEL   C3A7   SOK    0001   SOUT   C004   SPACE  0020
SPE    0004   SREA1  C036   SREAD  C037   SROL   C09C
SSTAT  C831   STAPT  00FA   STBE   0080   STRTA  C265
SYSRA  C800   SYSTP  CC00   TAPE1  0040   TAPE2  0080
TAPIN  C3C6   TBL    C165   TDATA  00FB   TDR    0040
TERR   C270   TERR   C386   TFE    0008   THEAD  C805
TIME   C29E   TLOAD  C339   TLOD1  C354   TOE    0010
TOFF   C3ED   TOUT   C28D   TREDY  C3D7   TSRCH  C462
TTBE   0080   UP     0097   VDAD   C128   VDAD2  C125
VDADD  C121   VDMEM  CC00   VDMO1  C04C   VDMOT  C048
XEQAD  C81E
```

(continued from page 4)

```
ØØ[illegible] 22 F1 Ø4    Ø71Ø          SHLD   SHED+36DH FIREMAN
Ø     2A 78 Ø1    Ø72Ø          LHLD   FRA1    .. THE LITTLE ENGINE
Ø[illegible]  22 24 Ø5    Ø73Ø          SHLD   SHED+3AØH ALSO
ØØA7  2A 7A Ø1    Ø74Ø          LHLD   FRA2    .. HAD A COWCATCHER
ØØAA  22 26 Ø5    Ø75Ø          SHLD   SHED+3A2H AND LOTS OF FUNNY
ØØAD  22 28 Ø5    Ø76Ø          SHLD   SHED+3A4H WHEELS AND THE
ØØBØ  22 2A Ø5    Ø77Ø          SHLD   SHED+3A6H THINGS THAT CONNECTED
ØØB3  2A 7C Ø1    Ø78Ø          LHLD   FRA3    .. AND TWO VERY TINY
ØØB6  22 2C Ø5    Ø78Ø          SHLD   SHED+3A8H WHEELS AT THE VERY
ØØB9  2A 7E Ø1    Ø8ØØ          LHLD   FRA4    .. VERY BACK
ØØBC  22 2E Ø5    Ø81Ø          SHLD   SHED+3AAH ALTOGETHER IT LOOKED
ØØBF  2A 8Ø Ø1    Ø82Ø          LHLD   FRA5    .. QUITE LONELY AND YET
ØØC2  22 3Ø Ø5    Ø83Ø          SHLD   SHED+3ACH IT APPEARED VERY
ØØC5  Ø1 DF FF    Ø84Ø          LXI    B,-21H  . FUNNY JUST SITTING
ØØC8  21 62 Ø5    Ø85Ø          LXI    H,SHED+3DEH THERE ON THE
ØØCB  36 19       Ø86Ø  RAILS   MVI    M,19H   .. RAILS
ØØCD  Ø3          Ø87Ø          INX    B       .. WITH NOTHING AT ALL
ØØCE  23          Ø88Ø          INX    H       .. EVER
ØØCF  AF          Ø89Ø          XRA    A       .. TO DO
ØØDØ  A8          Ø9ØØ          XRA    B       ..   W E L L !
ØØD1  C2 CB ØØ    Ø91Ø          JNZ    RAILS   .. LET'S RUN IT JUST FOR
                                                 FUN
ØØD4  21 ØØ CC    Ø92Ø  HOSTL   LXI    H,RRY   GET TRAIN OUT OF SHED
ØØD7  EB          Ø93Ø          XCHG   .       MOVE IT FROM SHED
ØØD8  21 84 Ø1    Ø94Ø          LXI    H,SHED
ØØDB  7E          Ø95Ø  FIRE    MOV    A,M     LIGHT FIRE
ØØDC  23          Ø96Ø          INX    H
ØØDD  EB          Ø97Ø          XCHG
ØØDE  77          Ø98Ø          MOV    M,A     ..MOVE IT
ØØDF  23          Ø99Ø          INX    H
ØØEØ  7C          1ØØØ          MOV    A,H
ØØE1  EB          1Ø1Ø          XCHG
ØØE2  FE DØ       1Ø2Ø          CPI    ØDØH    END OF YARD?
ØØE4  C2 DB ØØ    1Ø3Ø          JNZ    FIRE    NO, MAKE MORE STEAM!
ØØE7  CD ED ØØ    1Ø4Ø          CALL   TRAVL   GO TAKE TRIP
ØØEA  C3 D4 ØØ    1Ø5Ø          JMP    HOSTLE  NO ROUND TRIPS, JUST DO
                                                 AGAIN
ØØED              1Ø6Ø  .
ØØED  Ø1 4Ø Ø3    1Ø7Ø  TRAVL   LXI    B,64.13 MAKE TRIP 13 MILES
ØØFØ  C5          1Ø8Ø  CHOO    PUSH   B       CHUFF ONCE
Ø[illegible]  CD FF ØØ    1Ø9Ø          CALL   STROK   MAKE FORWARD MOTION
Ø.    CD ØF Ø1    11ØØ          CALL   TURN    TURN WHEELS
ØØ[illegible] C1          111Ø          POP    B
ØØF8  ØB          112Ø          DCX    B       CLICK ODOMETER
ØØF9  AF          113Ø          XRA    A
ØØFA  A8          114Ø          XRA    B       13 MILES YET?
ØØFB  C2 FØ ØØ    115Ø          JNZ    CHOO    NO
ØØFE  C9          116Ø          RET    .       DO NEXT TRIP
ØØFF              117Ø  .
ØØFF  21 Ø1 CC    118Ø  STROK   LXI    H,ØCCØ1H MOVE DOWN THE TRACK
Ø1Ø2  ØE DØ       119Ø          MVI    C,ØDØH
Ø1Ø4  7E          12ØØ  COAL    MOV    A,M     ADD MORE COAL
Ø1Ø5  2B          121Ø          DCX    H
Ø1Ø6  77          122Ø          MOV    M,A
Ø1Ø7  23          123Ø          INX    H
Ø1Ø8  23          124Ø          INX    H
Ø1Ø9  7C          125Ø          MOV    A,H
Ø1ØA  B9          126Ø          CMP    C       ENOUGH COAL?
Ø1ØB  C2 Ø4 Ø1    127Ø          JNZ    COAL    NO, PUT MORE ON!
Ø1ØE  C9          128Ø          RET
Ø1Ø[illegible]            129Ø  .
Ø1Ø[illegible] 21 ØØ CC    13ØØ  TURN    LXI    H,ØCCØØH TURN WHEELS
Ø112  7E          131Ø  HISS    MOV    A,M
Ø113  23          132Ø          INX    H
Ø114  FE Ø6       133Ø          CPI    6       FIND COWCATCHER
Ø116  C2 12 Ø1    134Ø          JNZ    HISS    LOOK AGAIN
Ø119  23          135Ø  FOUND   INX    H
Ø11A  7E          136Ø  AXLE    MOV    A,M
Ø11B  FE 11       137Ø          CPI    11H     WHAT QUARTER TURN?
Ø11D  C2 31 Ø1    138Ø          JNZ    NXT1
Ø12Ø  11 7E 14    139Ø          LXI    D,147EH NEXT QUARTER TURN
Ø123              14ØØ  .
Ø123  Ø6 Ø7       141Ø  AXL1    MVI    B,7     DO 4 AXLES
Ø125  72          142Ø  AXL2    MOV    M,D     WHEELS MOVED HERE
Ø126  Ø5          143Ø          DCR    B       LAST WHEEL?
Ø127  CA 52 Ø1    144Ø          JZ     WORK    I'VE BEEN WORKIN'...
Ø12A  23          145Ø          INX    H       ..ON THE RAILROAD..
Ø12B  73          146Ø          MOV    M,E     (SIDE RODS MOVED HERE)
Ø12C  23          147Ø          INX    H       ..ALL THE LIVE..
Ø[illegible]  Ø5          148Ø          DCR    B       ..LONG DAY.....
Ø.    C2 25 Ø1    149Ø          JNZ    AXL2
Ø131              15ØØ  .
Ø131  FE 14       151Ø  NXT1    CPI    14H     QUARTER TURN
Ø133  C2 3C Ø1    152Ø          JNZ    NXT2
Ø136  11 2D 13    153Ø          LXI    D,132DH
Ø139  C3 23 Ø1    154Ø          JMP    AXL1
Ø13C  FE 13       155Ø  NXT2    CPI    13H     QUARTER TURN
```

```
0130 C2 47 01     1560          JNZ     NXT3
0141 11 5F 12     1570          LXI     D,125FH
0144 C3 23 01     1580          JMP     AXL1
0147 FE 12        1590  NXT3    CPI     12H     QUARTER TURN
0149 C2 1A 01     1600          JNZ     AXLE
014C 11 2D 11     1610          LXI     D,112DH
014F C3 23 01     1620          JMP     AXL1
                  1630  .
    01 B8 0B      1640  WORK    LXI     B,3000
    0B            1650  OIL     DCX     B,
0156 DB 00        1660          IN      0       DID THE CONDUCTOR
0158 E6 40        1670          ANI     RDA     ..FLAG US DOWN?
015A C2 63 01     1680          JNZ     QUIT    QUITTIN' TIME
015D AF           1690          XRA     A       NO, THEN HIGHBALL
015E A8           1700          XRA     B
015F C2 55 01     1710          JNZ     OIL     NO SQUEEKS, PLEASE!
0162 C9           1720          RET
0163              1730  .
0163              1740  . PUT A JUMP TO WHEREVER YOU NEED TO RETURN
                                                  IN THE NEXT
0163 C3 60 E0     1750  QUIT    JMP     ALS8    ..YOUR RETURN LINK GOES
                                                  HERE
0166              1760  .
0166              1770  RDA     EQU     40H     PUT YOUR DATA READY FLAG
                                                   HERE
0166              1780  RRY     EQU     0CC00H
0166              1790  SMOKE   EQU     6FH     SMOKE CHARACTER
0166 01 10        1800  CAB1    DW      1001H   CAB DESCRIPTION
0168 10 5D        1810  CAB2    DW      5D10H   "   "
016A 7E 20        1820  CAB3    DW      207EH   "   "
016C 28 0A        1830  BOI1    DW      0A28H   BOILER DESCRIPTION
016E 0A 19        1840  BOI2    DW      190AH   "       "
0170 0A 0A        1850  BOI3    DW      0A0AH   "       "
0172 0A 5B        1860  BOI4    DW      5B0AH   "       "
0174 39 39        1870  BOI5    DW      3939H   "       "
0176 5D 20        1880  BOI6    DW      205DH   "       "
0178 06 20        1890  FRA1    DW      2006H   FRAME DESCRIPTION
017A 11 2D        1900  FRA2    DW      2D11H   "       "
017C 11 20        1910  FRA3    DW      2011H   "       "
017E 6F 2D        1920  FRA4    DW      2D6FH   "       "
0180 6F 20        1930  FRA5    DW      206FH   "       "
0182 19 19        1940  TIES    DW      1919H   TIES DESCRIPTION
0184 00           1950  SHED    NOP     .       THIS IS SHED AREA
0185              1960  CLOUD   EQU     $+0172H  BEGINNING OF CLOUD
0185              1970  .
0185
```

```
:1A00[illegible]AFD4C82184010100008362023000BAFA8C2009000021F7021100[illegible]06000007
:1A00[illegible]1A00CD5000112B0006000CD5000113000006009CD5000011340006004CD5006A
:1A0034000113B0006002CD5000113C0000601CD5000113E000601CD5000C39A
:1A004E0059000193660F23005C251000C921A6004361621A9004360721AB004366EE7
:1A0068000A66000122AE002A68000122B0002A6A000122B2002A6C000122E5002A6E009
:1A0082000122E7002A70000122E9002EB002A72000122ED002A74000122EF002A0D
:1A009C000760122F1002A78001222400520A7A001222600522080522200520A7C0195
:1A00B6000222C052A7E001222E052A8000122300501DFFF2162053619003230AF52
:1A00D0000ABC2CB000210000CCEB2184017E23EB77237CEBFED0C2DB000CDED000B1
:1A00EA00C3D4000140003C5CDFF000CD0F01C10BAFA8C2F000C921010CC0ED0049
:1A0104000E2B7723237CB9C2040100C921000CC7E23FE06C2120012370EFE11C2DD
:1A011E000310011176140600772005CA52012373230C225001FE14C23C01112D5C
:1A0138000013C323001FE13C2470011150F12C323001FE12C21A01112D11C323010C
:1A0152000010B800BD0BDB000E6400C263010AFA8C2550100C9C360E0001101005D7E2046
:19016C000028000A0A19000A0A0A5B39395D2000620112D112006F2D6F2001919000CB
:00
```

PROCESSOR TECHNOLOGY
ACCESS.

Processor Technology Corp.
6800 Hollis Street
Emeryville, CA 94608

V.1, 1, No.1

Bulk Rate
U.S. Postage
PAID
Permit No. 3729
Oakland, Calif.

Stanley M Sokolow
1690 Woodside Rd
Redwood City, CA
94061

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

PROCESSOR TECHNOLOGY

ACCESS.

75¢ per issue

Published by Wible/Rampton Advertising, San Francisco. Volume One, Number Two, April 1977

A Letter from the Editor

"Everyone should believe in something. I believe I'll have another drink."—Steele's Plagiarism of Somebody's Philosophy

Welcome to issue #2 of ACCESS. I hope the first one was able to provide you with something useful, after waiting so patiently for delivery. Wasn't entirely our fault this time, honest; we now know from personal experience that mailing services have their gremlins too. Anyway, issue #2 is now here and filled with goodies we hope you can use on your pet project.

One thing issue #2 isn't filled with is feedback from you. No way it could be, since it went to press before most of you ever got your hands on #1. We really do want to facilitate an exchange of ideas, comments, gripes, what have you, so please get those cards and letters into us. Are you more interested in hardware stuff than software stuff? Vice versa? What bugs bug you? What do you do with your Sol anyway? Got any hot tips on interfacing with exotic equipment? Just want to sound off about something? Or maybe even say something nice about us? We definitely want issue #3 to have more of you in it, instead of just us talking to the walls.

Speaking of us, you'll be reading this just about the time of the 1st West Coast Computer Faire. Come give us some feedback in person; we'll be there along with all the competition, and we'll have some good stuff to show you. You might even get to see me if you look closely; I'm the one wearing the red suspenders. For those of you who can't make it to San Francisco, we'll have a report in ACCESS #3.

Got to sign off now and fix that Sol PCB that just came in for warranty repair. (Probably sabotage—whoever heard of a SOL not working?)

Aram Attarian II

Subscription Information

ACCESS is published every six weeks. If you like what you see, we hope you'll send us $4.00 for a year's subscription so we can keep the info coming. Write to us at Processor Technology, 6200 Hollis Street, Emeryville, CA 94608.

Inside Emeryville!

One to One Communication

We're going to make this column a regular feature, under the peerless direction of our Customer Service Manager, the world renowned Ralph I. Palsson (applause). He'll keep you informed on what's happening in the world of availability, delivery dates, and other such precious tidbits, and do his best to keep the lines of communication open. Good luck, Ralph. (A.A.II, Editor)

A Customer Service Department has only one reason for its existence: to fix the responsibility on someone for keeping the communication lines open, so the company doesn't become just another faceless entity, vaguely malevolent to your attempts to pursue the love and lore of computing. The someone's me, Ralph Palsson; I'm here to welcome your telephone and letter inquiries and provide you with immediate, personal service. Whether it be pre-sales information, placing an order, finding the nearest Processor Technology dealer, checking on the availability of existing or forthcoming products—we'll do our best to help.

Are you interested in more information about a product than you could glean from the catalog description? How about the VDM-1 kit you ordered way back when and still haven't received? Even the best of us make occasional errors, and letting us know about your problems is the first step towards their solution. Letters are now being sent regularly to advise you, our customers, of anticipated production and delivery schedules. PTC dealers are being kept informed too, so you can call them for on-going status reports. We're trying continually to expand and improve the Customer Service Department with the aim of giving even more efficient, reliable service. Some of the ways we're doing it:

Teletroubleshooting. Having a problem with a PTC kit? Is your Sol-20 displaying 0's and 9's and nothing else? Need technical advice on interfacing? One of our applications engineers is available daily for phone consultation and technical assistance. Call (415) 652-8080 between 9:30 am and noon or 1:30 and 4:00 pm Pacific Time. Technically oriented software questions

(continued on page 2)

One to One Communication
(continued from page 1)

should go to our new division, Software Technology, at (415) 349-8080. (That's right, 8080!)

When you make a technical call about a malfunctioning kit, it helps if you take time to think out your questions first, maybe make a few notes. It's also a good idea to have your kit manual handy, and if possible, the recalcitrant piece of equipment. All calls are taken on a first come, first serve basis, so if you do have to hold for a few minutes, please be assured that you're not running up your bill for the benefit of someone who called in later. If you do run into a wait, you can also ask us to call you back collect. We will.

Again, let us remind you that if you purchased your PTC product through a dealer, try him/her first. They're all authorized to service the equipment they sell, precisely because they can give you more personalized, faster service than anyone can by mail or phone.

Need a defective part replaced? Again, the best way to handle this is through your dealer. If he's out of stock, or if you purchased directly from us, send us the defective part for replacement. Similarly, if you're missing a part from a kit, call your dealer or write to us, giving as complete a description as you can of what's missing.

Finally. We are always interested in improving our one-to-one communication, and we're open to your comments, criticisms, and suggestions. Please feel free to contact us if we can be of service in any way.

Error, Error, Does Not Compute

Errors do appear from time to time in all high class publications. ACCESS is no exception—our score for the first issue was three unfortunate oversights.

1. The instructions for wiring the 3P+S for a keyboard interface (page 3). At Step 5, we told you to connect pin 7 to a point on 3P+S leading to J2 pin 12. Pin 12 is a ground point, and you don't really want to do that. Pin 7 *should* be connected to a point on the 3P+S leading to IC 14 pin 13.

2. The article on interfacing PT 4KRA memory boards with the Motorola M6800 (page 2). It's all right as far as it goes, but we neglected to give proper credit to the contributor. Our apologies and thanks to J. W. Schook, P.O. Box 185, Rocky Point, N.Y.

3. Last, but not least, those ever-present gremlins managed to alter our mailing address to 6800 Hollis St. Now, that's really adding insult to injury. Our correct address is 6200 Hollis St., Emeryville, CA 94608.

Introducing Software Technology Corporation

We've pulled a protozoan trip and split in two: Software Technology Corp. is a separate new company formed to take over software development from Processor Technology. The idea is to give you more and better software support by not scattering our energies about, randomly.

Software Technology is three people with an impressively vast reservoir of experience in operating systems, time sharing, real time systems, compilers, interpreters, simulators, business systems, and software development. Did we leave anything out? During this transitional period, they'll be maintaining, supporting and producing all the current PTC software, plus developing some of their own. Later (not too much later), they'll be coming out with lots of good stuff designed to get the most out of Sol.

You should have your first chance to see the caliber of Software Technology developments at the Computer Faire in San Francisco. But we're not going to tell you what, because one fundamental policy of *Software Technology* is *not* to announce anything that isn't ready. Nobody likes a tease.

Meanwhile, Software is in business and even has a phone. If you have any problems with Processor Technology software you're using now, you can call their 24-hour phone line. When nobody's there, there's an answering machine to take your message, and they *will* get back to you promptly. The number is (415) 349-8080. You're also welcome to write your questions: P.O. Box 5260, San Mateo, CA 94402

For the time being, orders, questions about delivery and of course hardware questions should still be directed to Processor Technology. (See One to One Communication)

Read on for your first example of Software Technology software.

Announcing the SOL USERS GROUP!

The Sol Users Group was recently organized by members of the Homebrew Computer Club in Palo Alto; SUG is not affiliated in any way with Processor Technology. The purposes of SUG are to exchange software and other applications, and to create standards. If plenty of interest is shown, a Sol Newsletter will be published and sent to members.

If you own or have ordered a Sol, send your name, address, phone number and ideas to:

Bill Burns
4190 Maybell Way
Palo Alto, CA 94306
(Please don't call.)

Attention, CONSOL Users

All Processor Technology software has recently been modified in a very important way that encourages standardization. If you've been looking forward to having some Sol software for your CONSOL Proms, you'll have to let us reprogram for you. There's *no charge*. Just mail your CONSOL Proms, suitably packaged, to Processor Technology; we'll take care of the rest. Turn-around time is about 2 weeks, dependent on the vagaries of the U.S. Postal Service, natch.

If you're planning to upgrade to SOLOS anyway, don't worry about it.

Clarkes third law: Any sufficiently advanced technology is indistinguishable from magic.

CUTE, CUTEST, CUTER

Below is the complete source listing for the control and monitor programs for the CUTS board. This program, CUTER, was made available with the CUTS module in the form of a cassette selling for $11.

The CUTER cassette contains object code along with a relocating loader for loading the program in any 256-byte boundary of available RAM. A new version of BASIC-5 for use with CUTER and SOLOS is also on the cassette, followed by the complete source code of CUTER. Not a bad deal for $11.

So why buy the cassette now that we're giving you the listing? It's not that we're mercenary, but consider: if you want to use PTC's software packages on cassettes, then CUTER is going to be a tremendous asset in loading those tapes, right? Right—otherwise you could spend 3 or 4 hours toggling it in from the front panel of your A#$%&R or I(*@I.

PROCESSOR TECHNOLOGY
ACCESS.

Editor: Aram Attarian II
Publisher: Wible/Rampton Advertising, 727 15th Avenue, San Francisco, CA 94118
ACCESS is published approximately every six weeks. Subscription rate: $4 per year, from Processor Technology Corp., 6200 Hollis St., Emeryville, CA 94608.
ACCESS Copyright © April 1977 by Processor Technology Corp. All Rights Reserved. Material in this publication may not be reproduced in any form without permission from Processor Technology Corp.

Double Your Pleasure, Double Your Fun, Or, How to Use Both Parallel Ports on the 3P+S

You can take maximum advantage of the 3P+S's versatility by making a simple modification that gives you simultaneous access to both parallel ports. Then you'll be able to use one port for a keyboard input, the other for paper tape input/output. The serial port is available for an RS232 or TTY, with status and control available at Port 0 for Processor Technology software compatibility.

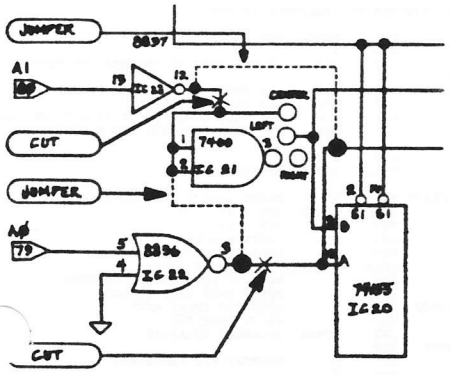
The change procedure is as follows:

1. Cut the existing trace from IC 22 pin 3 to IC 20 pin 13; at IC 22 pin 3
2. Cut the existing trace from IC 23 pin 12 to IC 21 pins 1 and 2; at IC 23 pin 12
3. Run a jumper from IC 23 pin 12 to IC 20 pin 13
4. Run a jumper from IC 22 pin 3 to IC 21 pins 1 and 2
5. Select area "B" option; jumper from left to right

Assuming that area "A" board address options "00" have been selected, the ports will now be set up as follows:

Port 0 = Channel C. Control and status
Port 1 = Channel A.* Parallel data (keyboard)
Port 2 = Channel D. UART (RS232 or TTY)
Port 3 = Channel B.* Parallel data (paper tape reader)
*The strobe latches for the parallel ports remain with their respective channels.

Changing the data available status.
The 3P+S interface is laid out to respond to a negative strobe input pulse with low active status (i.e., FA, FB). For compatibility with Processor software, you'll want to select a high active data available status. Run a jumper from the $\overline{Q}$ output of the respective strobe latch (i.e., AKA, AKB) to the correct status bit input in area "G." In the above configuration, this amounts to jumpering keyboard data available IC 15 pin 7 to area "G" point C6.



Osborns Law: Variables won't. Constants aren't.

FLASH - VDM Access Flicker Eliminated, Part 2

In the last issue, we gave you a hardware modification to take care of the flicker produced whenever the VDM memory is being accessed. We also promised to explore the implications a bit further this time. So here's a subroutine for the VDM driver program that implements the modification via software.

One thing to remember: if reads *from* the screen memory are needed by the driver, they have to be synchronized by a subroutine similar to this. This one only takes care of flickerless access *to* the memory.

Next issue, we'll continue the flicker saga by providing a version of the Processor PATTERN program which contains this routine for flickerless display.

```
1000 *  THIS ROUTINE MOVES A CHAR. IN REGISTER B
1005 *  TO VDM DISPLAY MEMORY
1010 *
1015 ***** ASSUMPTIONS:
1020 *
1025 *  1.  THE CALLER HAS SET UP A STACK
1030 *
1035 *  2.  REGISTER PAIR H&L CONTAIN THE
1040 *      VDM DISPLAY MEMORY ADDRESS
1045 *
1050 *  3.  THE ROUTINE IS RUNNING IN MEMORY
1055 *      WITH NO (0) WAIT STATES.
1060 *
1065 *
1070 *
1075 *
1080 START  PUSH   B
1085        MVI    C,2     SYNC BIT MASK (BIT 1)
1090 *
1095 *  CATCH FALLING EDGE OF SYNC
1100 *
1105 HILP   IN     0C8H    VDM STATUS PORT
1110        ANA    C       ONLY BIT 1 REMAINS
1115        JZ     HILP    UNTIL SYNC IS HI
1120 *
1125 *  SYNC IS HI. NOW WAIT
1130 *  TILL IT IS LOW
1135 *
1140 LOLP   IN     0C8H    VDM STATUS PORT
1145        ANA    C       ONLY BIT 1 REMAINS
1150        JNZ    LOLP
1155 *
1160 *  DISPLAY SWEEP IS NOW AT LEFT MARGIN
1165 *
1170        CALL   WAIT    SO SWEEP CAN MOVE TO
1175 *                      RIGHT MARGIN
1180 *
1185 *  NOW THAT SWEEP IS OFF THE SCREEN:
1190 *
1195        MOV    M,B     ACCESS DISPLAY MEMORY
1200 *
1205 *
1210        POP    B       GIVE IT BACK
1215        RET
1220 *
1225 *  THIS ROUTINE DELAYS FOR JUST ENOUGH TIME
1230 *  TO ALLOW THE SWEEP TO CROSS THE SCREEN
1235 *  ***  ALTER IT WITH CARE  ***
1240 *
1245 WAIT   NOP
1250        PUSH   H
1255        POP    H
1260        PUSH   H
1265        POP    H
1270        RET
```

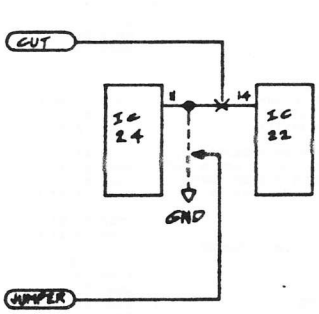
And now, the VDM-½! Or, Modifying Your VDM-1 for 32-character Display

We're pleased to announce that unceasing technical progress has now made possible a reduction in the number of characters per line of the VDM-1 display from 64 to 32! Seriously, the modification is quite handy if you want to work with large type display, or feed RF modulated signals to a TV antenna input.

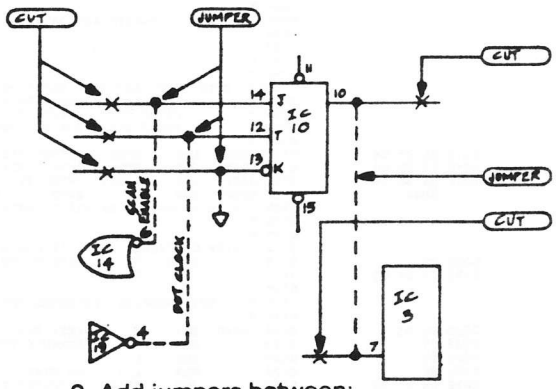
The hardware modifications are shown in the schematic drawing below. One word of caution: since we're tying the low-order address bit to ground, only character locations with bit 0 equal to 0 (i.e., even-numbered addresses), will be displayed. So if you use the modified VDM with unmodified software, you'll end up with only alternate characters on the screen and a rather cryptic message!

In the next issue of ACCESS, we'll have some spiffy software routines to use with the "VDM-½." Meanwhile, you can probably come up with some of your own. (Drop us a line if you run across anything interesting.) Just remember that a left shift of a 64-character address will give you the 32-character address, providing that you shift in a 0 to bit 0. Good luck!

Cann's Axiom: When all else fails, read the instructions.



1. Cut traces at:
A—pin 10 IC 10
B—pin 12 IC 10
C—pin 13 IC 10
D—pin 14 IC 10
E—pin 7 IX 3
F—pin 14 IC 22



2. Add jumpers between:
A—IC 24 pin 11 and Ground
B—IC 10 pin 10 and IC 3 pin 7
C—IC 10 pin 12 and IC 19 pin 4
D—IC 10 pin 13 and Ground
E—IC 10 pin 14 and IC 14 pin 6

```
                        **      PROGRAM DEVELOPMENT SYSTEM      **

                                SOFTWARE TECHNOLOGY CORP.
CUTER (TM)   77-03-27           P.O. BOX 5260
COPYRIGHT (C) 1977              SAN MATEO, CA  94402


                        9999            COPY    CUTER1/1
                        0002 *
                        0003 *
                        0004 *
                        0005 *          CUTER(TM)
                        0006 *
                        0007 *                  COPYRIGHT (C) 1977
                        0008 *                  SOFTWARE TECHNOLOGY CORP.
                        0009 *                  P.O. BOX 5260
                        0010 *                  SAN MATEO, CA 94402
                        0011 *                  (415) 349-8080
                        0012 *
                        0013 *    A L L    R I G H T S    R E S E R V E D ! ! !
                        0014 *
                        0015 *
                        0016 *          VERSION  1.3
                        0017 *                   77-03-27
                        0018 *
                        0019 *
                        0020 *  THIS PROGRAM IS DESIGNED TO BE A STANDALONE CUTS
                        0021 *  OPERATING SYSTEM.  CUTER IS DESIGNED TO BE READ IN FROM
                        0022 *  CASSETTE TAPE OR TO BE RESIDENT IN READ-ONLY-MEMORY.
                        0023 *  CUTER SUPPORTS VARIOUS DEVICES INCLUDING SERIAL,
                        0024 *  PARALLEL, THE PROCESSOR TECHNOLOGY VDM(TM) AND UP TO
                        0025 *  TWO CUTS TAPE DRIVES.
                        0026 *
                        0027 *  CUTER(TM) HAS BEEN WRITTEN SO AS TO BE COMPATIBLE WITH
                        0028 *  SOLOS(TM).  THE FOLLOWING KEYS ARE USED BY CUTER(TM)
                        0029 *  IN PLACE OF THE SPECIAL KEYS ON THE SOL KEYBOARD:
                        0030 *
                        0031 *     CURSOR UP        CTL-W
                        0032 *     CURSOR LEFT      CTL-A
                        0033 *     CURSOR RIGHT     CTL-S
                        0034 *     CURSOR DOWN      CTL-Z
                        0035 *     CURSOR HOME      CTL-N
                        0036 *     CLEAR SCREEN     CTL-K
                        0037 *     MODE             CTL-@
                        0038 *
                        0039 *
                        0040 *
                        0041 *
                        0042 *
                        0043 *    AUTO-STARTUP CODE
                        0044 *
C000 7F                 0045 START  MOV     A,A     SHOW THIS IS CUTER (SOLOS=00)
                        0046 *      THIS BYTE ALLOWS AUTOMATIC POWER ON ENTRY
                        0047 *      WHEN IN ROM SUPPORTING THIS HARDWARE FEATURE.
C001 C3 D7 C1           0048 INIT   JMP     STRTA   SYSTEM RESTART ENTRY POINT
                        0049 *
                        0050 *    THESE JUMP POINTS ARE PROVIDED TO ALLOW COMMON ENTRY
                        0051 * LOCATIONS FOR ALL VERSIONS OF CUTER.  THEY ARE USED
                        0052 * EXTENSIVELY BY CUTS SYSTEM PROGRAMS AND IT IS RECOMMENDED
                        0053 * THAT USER ROUTINES ACCESS CUTER ROUTINES THROUGH THESE
                        0054 * POINTS ONLY!
                        0055 *
C004 C3 18 C2           0056 RETRN  JMP     COMND   RETURN TO CUTER COMMAND PROCESSOR
C007 C3 DC C5           0057 FOPEN  JMP     BOPEN   CASSETTE OPEN FILE ENTRY
C00A C3 FF C5           0058 FCLOS  JMP     PCLOS   CASSETTE CLOSE FILE ENTRY
C00D C3 A2 C6           0059 RDBYT  JMP     RTBYT   CASSETTE READ BYTE ENTRY
C010 C3 7F C6           0060 WRBYT  JMP     WTBYT   CASSETTE WRITE BYTE ENTRY
C013 C3 C7 C6           0061 RDBLK  JMP     RTAPE   CASSETTE READ BLOCK ENTRY
C016 C3 7B C7           0062 WRBLK  JMP     WTAPE   CASSETTE WRITE BLOCK ENTRY
                        0063 *
                        0064 *       SYSTEM I/O ENTRY POINTS
                        0065 *
                        0066 *  THESE FOUR ENTRY POINTS ARE USED TO EITHER INPUT
                        0067 *  OR OUTPUT TO CUTER PSUEDO PORTS.
                        0068 *  THESE PSUEDO PORTS ARE AS FOLLOWS:
                        0069 *
                        0070 *  PORT   INPUT                OUTPUT
                        0071 *  ----   ------------------   ----------------------
                        0072 *    0    KEYBOARD INPUT       BUILT-IN VDM DRIVER
                        0073 *         ACTUAL PORT 3        PORT C8, MEMORY FROM CC00
                        0074 *    1    SERIAL PORT          SERIAL PORT
                        0075 *         ACTUAL PORT 1        ACTUAL PORT 1
                        0076 *    2    PARALLEL PORT        PARALLEL PORT
                        0077 *         ACTUAL PORT 2        ACTUAL PORT 2
                        0078 *    3    USER'S INPUT RTN     USER'S OUTPUT ROUTINE
                        0079 *
                        0080 *  STATUS FOR ACTUAL PORTS 1, 2 AND 3 IS VIA ACTUAL
                        0081 *  PORT 0.  THE BITS OF PORT ZERO ARE DEFINED AS FOLLOWS:
                        0082 *
                        0083 *    :     :     :     :     :     :----- : --- : --- :
                        0084 *    : TBE : RDA :     :     :     :PXDR : PDR : KDR :
                        0085 * BIT  7     6     5     4     3     2     1     0
                        0086 *
                        0087 *  WHERE:
                        0088 *    TBE    1=TRANSMITTER BUFFER EMPTY (SERIAL)
                        0089 *    RDA    1=READER DATA AVAILABLE (SERIAL)
                        0090 *    ----
                        0091 *    PXDR   0=PARALLEL EXTERNAL DEVICE READY
                        0092 *    ---
                        0093 *    PDR    0=PARALLEL DATA READY
                        0094 *    ---
                        0095 *    KDR    0=KEYBOARD DATA READY
                        0096 *
                        0097 *
                        0098 *
                        0099 *
                        0100 *  NOTE: SOUT AND SINP ARE "LDA" INSTRUCTIONS.
                        0101 *        THIS FACT IS USED TO ALLOW ACCESS TO THE
                        0102 *        BYTES "OPORT" AND "IPORT" DYNAMICALLY.
                        0103 *        THESE MUST REMAIN "LDA" INSTRUCTIONS!!!!!
                        0104 *
C019 3A 07 C8           0105 SOUT   LDA     OPORT   OUTPUT VIA STANDARD OUTPUT PSUEDO PORT
C01C C3 2E C0           0106 AOUT   JMP     OUTPR   OUTPUT VIA PSUEDO PORT SPECIFIED IN REG A
C01F 3A 06 C8           0107 SINP   LDA     IPORT   INPUT VIA STANDARD INPUT PSUEDO PORT
     C022               0108 AINP   EQU     $       INPUT VIA PSUEDO PORT SPECIFIED IN REG A
                        0109 * ----------END OF SYSTEM ENTRY POINTS----------
                        0110 *
                        0111 *
                        0112 * AINP CONTINUES HERE (IT COULD HAVE BEEN A "JMP" THOUGH)
C022 E5                 0113        PUSH    H       SAVE HL FM ENTRY
C023 21 09 C3           0114        LXI     H,ITAB
                        0115 *
                        0116 *    THIS ROUTINE PROCESSES THE I/O REQUESTS
                        0117 *
C026 E6 03              0118 IOPRC  ANI     3       KEEP REGISTER "A" TO FOUR VALUES
C028 07                 0119        RLC     .       COMPUTE ENTRY ADDRESS
C029 85                 0120        ADD     L
C02A 6F                 0121        MOV     L,A     WE HAVE ADDRESS
C02B C3 87 C2           0122        JMP     DISPT   DISPATCH TO IT
                        0123 *
                        0124 *
     C02E               0125 OUTPR  EQU     $       PROCESS OUTPUT REQUESTS
C02E E5                 0126        PUSH    H       SAVE REGS
C02F 21 01 C3           0127        LXI     H,OTAB  POINT TO OUTPUT DISPATCH TABLE
C032 C3 26 C0           0128        JMP     IOPRC   DISPATCH FOR PROPER PSUEDO PORT
                        0129 *
                        0130 *
                        0131 *
                        0132 * CUTER SYSTEM I/O ROUTINES
                        0133 *
                        0134 *
                        0135 *    THIS ROUTINE IS A MODEL OF ALL INPUT ROUTINES WITHIN
                        0136 *  CUTER.  THE FIRST ROUTINE "KREA1" PERFORMS THE IN
                        0137 *  FROM THE STANDARD KEYBOARD ON PARALLEL PORT 3.
                        0138 *  ALL STANDARD INPUT DRIVERS RETURN EITHER THE CHA
                        0139 *  WITH A NON-ZERO FLAG, OR JUST A ZERO FLAG INDICA
                        0140 *  THAT NO CHARACTER IS AVAILABLE YET.  IT WILL BE
                        0141 *  RESPONSIBILITY OF THE USER TO LOOP WAITING FOR A
                        0142 *  CHARACTER, OR TO USE THE INPUT AS A STATUS REQUEST.
                        0143 *  WHEN A CHARACTER IS AVAILABLE, IT IS RETURNED IN REG A.
                        0144 *
                        0145 *  THE FOLLOWING KEYBOARD ROUTINE MAY BE USED AS A SAMPLE
                        0146 *  OF HOW TO WRITE A USER INPUT ROUTINE.
                        0147 *
                        0148 *          KEYBOARD INPUT ROUTINE
                        0149 *
     C035               0150 KREA1  EQU     $       KEYBOARD READ ROUTINE
C035 DB 00              0151        IN      STAPT   GET STATUS WORD
C037 2F                 0152        CMA     .       INVERT IT FOR PROPER RETURN
C038 E6 01              0153        ANI     KDR     TEST NOT KEYBOARD DATA READY
C03A C8                 0154        RZ      .       ZERO IF NO CHARACTER RECEIVED
                        0155 *
C03B DB 03              0156        IN      KDATA   GET CHARACTER
C03D C9                 0157        RET     .       GO BACK WITH IT
                        0158 *
                        0159 *
                        0160 *
                        0161 *    SERIAL INPUT ROUTINE
                        0162 *
     C03E               0163 SREA1  EQU     $       SERIAL INPUT ROUTINE
C03E DB 00              0164        IN      STAPT   GET STATUS
C040 E6 A0              0165        ANI     SDR     TEST FOR SERIAL DATA READY
C042 C8                 0166        RZ      .       FLAGS ARE SET
                        0167 *
C043 DB 01              0168        IN      SDATA   GET DATA BYTE
                        0169 *  IT IS UP TO THE CALLER TO STRIP PARITY IF DESIRED
C045 C9                 0170        RET     .       WE HAVE IT
                        0171 *
                        0172 *
                        0173 *    SERIAL DATA OUTPUT
                        0174 *
     C046               0175 SEROT  EQU     $       SERIAL OUTPUT ROUTINE
C046 DB 00              0176        IN      STAPT   GET STATUS
C048 17                 0177        RAL     .       PUT HIGH BIT IN CARRY
C049 D2 46 C0           0178        JNC     SEROT   LOOP UNTIL TRANSMITTER BUFFER IS EMPTY
C04C 78                 0179        MOV     A,B     GET THE CHARACTER BACK
C04D D3 01              0180        OUT     SDATA   SEND IT OUT
C04F C9                 0181        RET     .       AND WE'RE DONE
                        0182 *
                        0183 *
                        0184 *    PARALLEL DATA INPUT
     C050               0185 PARIT  EQU     $       GET CHAR FM PARALLEL PORT
C050 DB 00              0186        IN      STAPT   STATUS
C052 2F                 0187        CMA     .       INVERT FOR PROPER RETURN
C053 E6 02              0188        ANI     PDR     IS DATA READY?
C055 C8                 0189        RZ      .       NO--JUST EXIT
C056 DB 02              0190        IN      PDATA   YES--GET CHAR THEN
C058 C9                 0191        RET     .       THEN EXIT
                        0192 *
                        0193 *
                        0194 *    PARALLEL DATA OUTPUT ROUTINE
     C059               0195 PAROT  EQU     $       OUTPUT CHAR TO PARALLEL PORT
C059 DB 00              0196        IN      STAPT   STATUS
C05B E6 04              0197        ANI     PXDR    IS EXTERNAL DEVICE READY?
C05D C2 59 C0           0198        JNZ     PAROT   NO--WAIT TIL IT IS
C060 78                 0199        MOV     A,B     GET CHAR
C061 D3 02              0200        OUT     PDATA   SEND DATA NOW
C063 C9                 0201        RET     .       DONE
                        0202 *
                        0203 *
                        0204 * USER DEFINED INPUT/OUTPUT ROUTINES
     C064               0205 ERRIT  EQU     $       USER INPUT ROUTINE
C064 E5                 0206        PUSH    H       SAVE ORIG HL
C065 2A 00 C8           0207        LHLD    UIPRT   GET USER'S RTN ADDR
C068 C3 6F C0           0208        JMP     ERRO1   MERGE TO VERIFY THE ADDR
                        0209 *
     C06B               0210 ERROT  EQU     $       USER OUTPUT ROUTINE
C06B E5                 0211        PUSH    H       SAVE ORIG HL
C06C 2A 02 C8           0212        LHLD    UOPRT   GET USER'S RTN ADDR
     C06F               0213 ERRO1  EQU     $       WE MERGE HERE TO VFY ADDR
C06F 7D                 0214        MOV     A,L     ZERO=UNDEFINED
C070 B4                 0215        ORA     H       IS IT?
C071 C2 8B C2           0216        JNZ     DISP1   NO--VALID--OFF TO IT
C074 C3 0F C2           0217        JMP     STRTD   RESET I/O PORTS AND BACK TO COMMAND MODE
                        0218 *
                        0219 *
                        0220 *
                        0221 *                  VIDEO DISPLAY ROUTINES
                        0222 *
                        0223 *
                        0224 *  THESE ROUTINES ALLOW FOR STANDARD VIDEO TERMINAL
                        0225 *  OPERATIONS.  ON ENTRY, THE CHARACTER FOR OUTPUT IS IN
                        0226 *  REGISTER B AND ALL REGISTERS ARE UNALTERED ON RETURN.
                        0227 *
                        0228 *
                        0229 *
     C077               0230 VDMO1  EQU     $       VDM OUTPUT DRIVER
C077 E5                 0231        PUSH    H       SAVE HL
C078 D5                 0232        PUSH    D       SAVE DE
C079 C5                 0233        PUSH    B
                        0234 *
                        0235 *  PROCESS ESC SEQUENCE IF ANY
                        0236 *
C07A 3A 0C C8           0237        LDA     ESCFL   GET ESCAPE FLAG
C07D B7                 0238        ORA     A
C07E C2 87 C1           0239        JNZ     ESCS    IF NON ZERO GO PROCESS THE REST OF THE S
                        0240 *
C081 78                 0241        MOV     A,B     GET CHAR
C082 E6 7F              0242        ANI     7FH     CLR HI BIT IN CASE
C084 47                 0243        MOV     B,A     USE CHAR STRIPPED OF HI BIT FOR COMPATAB
C085 CA 9F C0           0244        JZ      GOBK    MAKE A QUICK EXIT FOR A NULL
                        0245 *
C088 21 E2 C2           0246        LXI     H,TBL
C08B CD A5 C0           0247        CALL    TSRCH   GO PROCESS
                        0248 *
     C08E               0249 GOBACK EQU     $       RESET CURSOR AND DELAY
C08E CD 44 C1           0250        CALL    VDADD   GET SCRN ADDR
C091 7E                 0251        MOV     A,M     GET CHAR
C092 F6 80              0252        ORI     80H     INVERSE VIDEO
C094 77                 0253        MOV     M,A     CURSOR IS NOW THERE
C095 2A 0A C8           0254        LHLD    SPEED-1 GET DELAY SPEED
C098 2C                 0255        INR     L       MAKE IT DEFINITELY NON-ZERO
C099 AF                 0256        XRA     A       DELAY ENDS WHEN H=ZERO
C09A 2B                 0257 TIMER  DCX     H       LOOP FOR DELAY AMNT
C09B BC                 0258        CMP     H       IS IT DONE YET
C09C C2 9A C0           0259        JNZ     TIMER   NO--KEEP DELAYING
C09F C1                 0260 GOBK   POP     B
C0A0 D1                 0261        POP     D       RESTORE ALL REGISTERS
C0A1 E1                 0262        POP     H
C0A2 C9                 0263        RET     .       EXIT FROM VDMOT
                        0264 *
```

```
                        0265 *
      C0A3              0266 NEXT   EQU    $       GO TO NEXT CHR
C0A3 23                 0267        INX    H
C0A4 23                 0268        INX    H
                        0269 *
                        0270 *  THIS ROUTINE SEARCHES FOR A MATCH OF THE CHAR IN "B"
                        0271 *  TO THE CHAR IN THE TBL POINTED TO BY HL.
                        0272 *
 C0A5 7E                0273 TSRCH  MOV    A,M     GET CHR FROM TABLE
C0A6 B7                 0274        ORA    A       SEE IF END OF TBL
C0A7 CA B7 C0           0275        JZ     CHAR    ZERO IS THE LAST
C0AA B8                 0276        CMP    B       TEST THE CHR
C0AB 23                 0277        INX    H       POINT FORWARD
C0AC C2 A3 C0           0278        JNZ    NEXT
C0AF E5                 0279        PUSH   H       FOUND ONE...SAVE ADDRESS
C0B0 CD 5E C1           0280        CALL   CREM    REMOVE CURSOR
C0B3 E3                 0281        XTHL   .       RESTORE ADDR OF CHAR ENTRY IN TBL
C0B4 C3 B7 C2           0282        JMP    DISPT   DISPATCH FOR CURSOR CONTROL
                        0283 *
                        0284 *
      C0B7              0285 CHAR   EQU    $       WE HAVE A CHAR
C0B7 78                 0286        MOV    A,B     GET CHARACTER
C0B8 FE 7F              0287        CPI    7FH     IS IT A DEL?
C0BA C8                 0288        RZ     .       GO BACK IF SO
                        0289 *
                        0290 *
                        0291 *
C0BB CD 4A C1           0292 OCHAR  CALL   VDADD   GET SCREEN ADDRESS
C0BE 70                 0293        MOV    M,B     PUT CHR ON SCREEN
C0BF 3A 08 C8           0294        LDA    NCHAR   GET CHARACTER POSITION
C0C2 FE 3F              0295        CPI    63      END OF LINE?
C0C4 DA E4 C0           0296        JC     OK
C0C7 3A 09 C8           0297        LDA    LINE
C0CA FE 0F              0298        CPI    15      END OF SCREEN?
C0CC C2 E4 C0           0299        JNZ    OK
                        0300 *
                        0301 *   END OF SCREEN...ROLL UP ONE LINE
                        0302 *
C0CF AF                 0303 SCROLL XRA    A
C0D0 32 08 C8           0304        STA    NCHAR   BACK TO FIRST CHAR POSITION
C0D3 4F                 0305 SROL   MOV    C,A
C0D4 CD 4B C1           0306        CALL   VDAD    CALCULATE LINE TO BE BLANKED
C0D7 AF                 0307        XRA    A
C0D8 CD 22 C1           0308        CALL   CLIN1   CLEAR IT
C0DB 3A 0A C8           0309        LDA    BOT
C0DE 3C                 0310        INR    A
C0DF E6 0F              0311        ANI    0FH
C0E1 C3 11 C1           0312        JMP    ERAS3
                        0313 *
                        0314 *   INCREMENT LINE COUNTER IF NECESSARY
                        0315 *
C0E4 3A 08 C8           0316 OK     LDA    NCHAR   GET CHR POSITION
C0E7 3C                 0317        INR    A
C0E8 E6 3F              0318        ANI    3FH     MOD 64
C0EA 32 08 C8           0319        STA    NCHAR   STORE THE NEW
C0ED C0                 0320        RNZ    .       MORE CHARS THIS LINE
      C0EE              0321 PDOWN  EQU    $       MOVE CURSOR DOWN ONE LINE
C0EE 3A 09 C8           0322        LDA    LINE    GET THE LINE COUNT
C0F1 3C                 0323        INR    A
C0F2 E6 0F              0324 CURSC  ANI    0FH     MOD 15 INCREMENT
C0F4 32 09 C8           0325 CUR    STA    LINE    STORE THE NEW
C0F7 C9                 0326        RET
                        0327 *
                        0328 *    ERASE SCREEN
                        0329 *
C0F8 21 00 CC           0330 PERSE  LXI    H,VDMEM  POINT TO SCREEN
C0FB 36 A0              0331        MVI    M,80H+' '  THIS IS THE CURSOR
                        0332 *
C0FD 23                 0333        INX    H       NEXT CHAR
      C0FE              0334 ERAS1  EQU    $       LOOP TO CLR SCRN
C0FE 36 20              0335        MVI    M,' '   BLANK IT OUT
C100 23                 0336        INX    H       NEXT SCRN LOC
C101 7C                 0337        MOV    A,H     SEE IF DONE
C102 FE D0              0338        CPI    0D0H    DID IT GO ABOVE VDM
C104 DA FE C0           0339        JC     ERAS1   NO--MORE
C107 37                 0340        STC    .       SAY WE WANT TO DROP THRU TO ERAS3
                        0341 *
      C108              0342 PHOME  EQU    $       RESET CURSOR TO HOME
C108 3E 00              0343        MVI    A,0     CLEAR, LEAVE CARRY AS IS
C10A 32 09 C8           0344        STA    LINE    ZERO LINE
C10D 32 08 C8           0345        STA    NCHAR   LEFT SIDE OF SCREEN
C110 D0                 0346        RNC    .       THIS IS JUST A HOME OPERATION
                        0347 *
C111 D3 C8              0348 ERAS3  OUT    DSTAT   RESET SCROLL PARAMETERS
C113 32 0A C8           0349        STA    BOT     BEGINNING OF TEXT OFFSET
C116 C9                 0350        RET
                        0351 *
                        0352 *
      C117              0353 CLIN2  EQU    $       HERE TO SEE IF VDM OUTPUT
C117 3A 07 C8           0354        LDA    OPORT   GET CRNT OUTPUT PORT
C11A B7                 0355        ORA    A
C11B C0                 0356        RNZ    .       NOT VDM--DONE THEN
C11C CD 4A C1           0357 CLINE  CALL   VDADD   GET CURRENT SCREEN ADDRESS
C11F 3A 08 C8           0358        LDA    NCHAR   CURRENT CURSOR POSITION
C122 FE 40              0359 CLIN1  CPI    64      NO MORE THAN 63
C124 D0                 0360        RNC    .       ALL DONE
C125 36 20              0361        MVI    M,' '   ALL SPACED OUT
C127 23                 0362        INX    H
C128 3C                 0363        INR    A
C129 C3 22 C1           0364        JMP    CLIN1   LOOP TO END OF LINE
                        0365 *
                        0366 *
                        0367 *  ROUTINE TO MOVE THE CURSOR UP ONE LINE
                        0368 *
C12C 3A 09 C8           0369 PUP    LDA    LINE    GET LINE COUNT
C12F 3D                 0370        DCR    A
C130 C3 F2 C0           0371        JMP    CURSC   MERGE
                        0372 *
                        0373 *  MOVE CURSOR LEFT ONE POSITION
                        0374 *
C133 3A 08 C8           0375 PLEFT  LDA    NCHAR
C136 3D                 0376        DCR    A
      C137              0377 PCUR   EQU    $       TAKE CARE OF CURSOR SAME LINE
C137 E6 3F              0378        ANI    03FH    LET CURSOR WRAP AROUND
C139 32 08 C8           0379        STA    NCHAR   UPDATED CURSOR
C13C C9                 0380        RET
                        0381 *
                        0382 *    CURSOR RIGHT ONE POSITION
                        0383 *
C13D 3A 08 C8           0384 PRIT   LDA    NCHAR
C140 3C                 0385        INR    A
C141 C3 37 C1           0386        JMP    PCUR
                        0387 *
                        0388 *   ROUTINE TO CALCULATE SCREEN ADDRESS
                        0389 *
                        0390 *   ENTRY AT:    RETURNS:
                        0391 *
                        0392 *         VDADD  CURRENT SCREEN ADDRESS
                        0393 *         VDAD2  ADDRESS OF CURRENT LINE, CHAR 'C'
                        0394 *         VDAD   LINE 'A', CHARACTER POSITION 'C'
                        0395 *
C144 3A 08 C8           0396 VDADD  LDA    NCHAR   GET CHARACTER POSITION
C147 4F                 0397        MOV    C,A     'C' KEEPS IT
C148 3A 09 C8           0398 VDAD2  LDA    LINE    LINE POSITION
C14B 6F                 0399 VDAD   MOV    L,A     INTO 'L'
C14C 3A 0A C8           0400        LDA    BOT     GET TEXT OFFSET
C14F 85                 0401        ADD    L       ADD IT TO THE LINE POSITION
C150 0F                 0402        RRC    .       TIMES TWO
C151 0F                 0403        RRC    .       MAKES FOUR
C152 6F                 0404        MOV    L,A     L HAS IT
C153 E6 03              0405        ANI    3       MOD THREE FOR LATER
C155 C6 CC              0406        ADI    <VDMEM  LOW SCREEN OFFSET
C157 67                 0407        MOV    H,A     NOW H IS DONE
C158 7D                 0408        MOV    A,L     TWIST L'S ARM
C159 E6 C0              0409        ANI    0C0H
C15B 81                 0410        ADD    C
C15C 6F                 0411        MOV    L,A
C15D C9                 0412        RET    .       H & L ARE NOW PERVERTED
                        0413 *
                        0414 *    ROUTINE TO REMOVE CURSOR
                        0415 *
C15E CD 4A C1           0416 CREM   CALL   VDADD   GET CURRENT SCREEN ADDRESS
C161 7E                 0417        MOV    A,M
C162 E6 7F              0418        ANI    7FH     STRIP OFF THE CURSOR
C164 77                 0419        MOV    M,A
C165 C9                 0420        RET
                        0421 *
                        0422 *    ROUTINE TO BACKSPACE
                        0423 *
C166 CD 33 C1           0424 PBACK  CALL   PLEFT
C169 CD 4A C1           0425        CALL   VDADD   GET SCREEN ADDRESS
C16C 36 20              0426        MVI    M,' '   PUT A BLANK THERE
C16E C9                 0427        RET
                        0428 *
                        0429 *    ROUTINE TO PROCESS A CARRIAGE RETURN
                        0430 *
C16F CD 1C C1           0431 PCR    CALL   CLINE   CLEAR FROM CURRENT CURSOR TO END OF LINE
                        0432 *  NOTE THAT A COMES BACK=64 WHICH WILL BE CLEARED AT PCUR
C172 C3 37 C1           0433        JMP    PCUR    AND STORE THE NEW VALUE
                        0434 *
                        0435 *   ROUTINE TO PROCESS LINEFEED
                        0436 *
C175 3A 09 C8           0437 PLF    LDA    LINE    GET LINE COUNT
C178 3C                 0438        INR    A       NEXT LINE
C179 E6 0F              0439        ANI    15      SEE IF IT WRAPPED AROUND
C17B C2 F4 C0           0440        JNZ    CUR     IT DID NOT--NO SCROLL
                        0441 *
C17E C3 D3 C0           0442        JMP    SROL    SCROLL ONE LINE--CURSOR SOME POSITION
                        0443 *
                        0444 *    SET ESCAPE PROCESS FLAG
                        0445 *
C181 3E FF              0446 PESC   MVI    A,-1
C183 32 0C C8           0447        STA    ESCFL   SET FLAG
C186 C9                 0448        RET
                        0449 *
                        0450 *     PROCESS ESCAPE SEQUENCE
                        0451 *
C187 CD 5E C1           0452 ESCS   CALL   CREM    REMOVE CURSOR
C18A CD 90 C1           0453        CALL   ESCSP   PROCESS THE CHARACTER
C18D C3 8E C0           0454        JMP    GOBACK
                        0455 *
C190 3A 0C C8           0456 ESCSP  LDA    ESCFL   GET ESCAPE FLAG
C193 FE FF              0457        CPI    -1      TEST FLAG
C195 CA B8 C1           0458        JZ     SECOND
                        0459 *
                        0460 *  PROCESS THIRD CHR OF ESC SEQUENCE
                        0461 *
C198 21 0C C8           0462        LXI    H,ESCFL
C19B 36 00              0463        MVI    M,0
C19D FE 02              0464        CPI    2
C19F DA B0 C1           0465        JC     SETX    SET X
C1A2 CA B4 C1           0466        JZ     SETY    SET Y
C1A5 FE 08              0467        CPI    8       SPECIAL SET SPEED
C1A7 CA 94 C5           0468        JZ     STSPD   YES--SET THE SPEED WITH IT THEN
C1AA FE 09              0469        CPI    9
C1AC DA BB C0           0470        JC     OCHAR   PUT IT ON THE SCREEN
C1AF C0                 0471        RNZ
                        0472 *
                        0473 *  TAB ABSOLUTE TO VALUE IN REG B
                        0474 *
C1B0 78                 0475 SETX   MOV    A,B
C1B1 C3 37 C1           0476        JMP    PCUR
                        0477 *
                        0478 *  SET CURSOR TO LINE "B"
                        0479 *
C1B4 78                 0480 SETY   MOV    A,B
C1B5 C3 F2 C0           0481        JMP    CURSC
                        0482 *
                        0483 *
                        0484 *   PROCESS SECOND CHR OF ESC SEQUENCE
                        0485 *
C1B8 78                 0486 SECOND MOV    A,B
C1B9 FE 03              0487        CPI    3
C1BB CA CE C1           0488        JZ     CURET
C1BE FE 04              0489        CPI    4
C1C0 C2 CA C1           0490        JNZ    ARET2
                        0491 *
C1C3 44                 0492 ARET   MOV    B,H
C1C4 4D                 0493        MOV    C,L     PRESENT SCREEN ADDRESS TO BC FOR RETURN
C1C5 E1                 0494 ARET1  POP    H       RETURN ADDRESS
C1C6 D1                 0495        POP    D       OLD B
C1C7 C5                 0496        PUSH   B
C1C8 E5                 0497        PUSH   H
C1C9 AF                 0498        XRA    A
C1CA 32 0C C8           0499 ARET2  STA    ESCFL
C1CD C9                 0500        RET
                        0501 *
                        0502 *
                        0503 *     RETURN PRESENT SCREEN PARAMETERS IN BC
                        0504 *
C1CE 21 08 C8           0505 CURET  LXI    H,NCHAR
C1D1 46                 0506        MOV    B,M     CHARACTER POSITION
C1D2 23                 0507        INX    H
C1D3 4E                 0508        MOV    C,M     LINE POSITION
C1D4 C3 C5 C1           0509        JMP    ARET1
                        0510 *
                        0511 *
                        0512 *
                        0513 *              START UP SYSTEM
                        0514 *
                        0515 *   CLEAR SCREEN AND THE FIRST 256 BYTES OF GLOBAL RAM
                        0516 *  THEN ENTER THE COMMAND MODE.
                        0517 *
C1D7 AF                 0518 STRTA  XRA    A
C1D8 4F                 0519        MOV    C,A
C1D9 21 04 C8           0520        LXI    H,DFLTS  CLEAR AFTER USER PORT ADDRESSES
                        0521 *
C1DC 77                 0522 CLERA  MOV    M,A
C1DD 23                 0523        INX    H
C1DE 0C                 0524        INR    C
C1DF C2 DC C1           0525        JNZ    CLERA
                        0526 *
                        0527 * DETERMINE THE DEFAULT PORTS
                        0528 *     THIS COULD BECOME "MVI A,XX" FOR YOUR SPECIFIC PORTS
C1E2 DB FF              0529        IN     SENSE   GET SWITCHES
                        0530 *
C1E4 47                 0531        MOV    B,A     SAVE IT
C1E5 E6 03              0532        ANI    3       MAKE IT A VALID PORT
C1E7 32 05 C8           0533        STA    DFLTS+1  SET DEFAULT OUTPUT PORT
```

(continued on page 10)

Bug Squad

This issue's Bug Squad focuses on the Sol PC board Revision D, the one you have. All the fixes described are aimed at worst case conditions, so hopefully you've never encountered the problems. But we do recommend making the changes now to forestall future headaches that might crop up if you use demanding peripherals such as discs. All the modifications described will be incorporated in the forthcoming Revision E board, so in the future we'll be designing on the assumption that all boards in the field have these fixes.

Now, you may wonder where we get the gall to blurt out that our product is not the ultimate in all respects. Quite simply, we have too much experience in product design to let ourselves get away with the attitude that goes, "We're perfect. Something must be wrong with *you*." So things can be better, and here's how.

1. Clock Width Fix

The bug: Currently the width of the phase 1 (Ø1) clock pulse is 70 nanoseconds. If you want to bring it into spec with existing 8080 chips, you should increase it to 140 nanoseconds. (8080A or 9080A are OK at 70 nsec.)

The squasher: On the top (component) side of the board, cut the trace between jumpers D and E of (U90 and U91) of the clock generator. On the bottom (solder) side of the board, connect the jumper from pin E to the feedthrough which leads to pin 5 of U91.

2. Phantom Glitch Fix

The bug: Occasionally a Sol will power up with three "phantom" cycles instead of the necessary four, causing a "crash." These are the cycles which use the "four phase wonder" software in the monitor.

The squasher: Connect a jumper on the solder side of the board as shown in Figure B. It goes from pin 4 of U76 to the feedthrough immediately below pin 1 of U76.

3. Ground Noise Fix

The bug: The paths from the bus drivers to the bus ground are too long, producing occasional ground noise.

The Squasher: Shorten them by connecting jumpers on the solder side of the board as shown in Figure B. They go from pin 8 of IC's U33, U50, U68, and U81 to the ground feedthrough leading to C45.

4. Protect Fix

The bug: The protect line is floating, which allows noise pulses to set a memory board "protected" at the most inconvenient times.

The squasher: Connect a jumper wire on the solder side as shown in Figure B. It goes from the ground terminal of C11 to pin 70 of the 100-pin bus connector J11.

5. DMA/Interrupt Unscramble

This fix has probably been included in your kit or preassembled board, but better check the connections just to make sure.

The bug: PINT (pin 73), PHOLD (pin 74), and PINTE (pin 26) got scrambled at an early stage in development and weren't noticed until too late.

The squasher: On the component side of the board, cut the trace leading to pin 73 on J11; on the solder side, cut the trace leading to pin 1 of U45, and also the second trace to the right of U64. (Refer to Figure B). Now connect three jumpers: From pin 73 of J11 to pin 1 of U45. From pin 28 of J11 to the feedthrough indicated (the one that was isolated by the cut on the component side). From the feedthrough directly below pin 1 of U45 to the feedthrough to the right of pin 3 of U64.

6 MWRITE Fix

The bug: If you want to operate with DMA devices which write into memory, such as discs, you need to be able to generate the MWRITE pulse externally.

The squasher: You accomplish this by connecting the signals which generate MWRITE directly to the bus. On the solder side of the board, cut the trace which leads to pin 7 of U93. Now connect a jumper from the trace which has been isolated, to the feedthrough leading to pin 9 of U94. Now, still on the solder side, locate the feedthrough immediately below pin 1 of U92 and break the trace leading to it; do NOT break the trace leading to pin 1 of U92. Connect a jumper from that feedthrough to pin 13 of U107.

Further Remarks on D and E

The other major change between the D and E revision Sol boards involves reversing the order of the parallel input data lines as connected to the parallel connector J2. The schematic is correct for the D board, and the list of signals in the manual is correct for the E board. The change will simplify future connections to J2 by placing the POD lines adjacent to the PID lines; that way, you can create a bi-directional input/output bus with a simple jumpering scheme at the connector.

We're planning an adaptor connector to convert Rev. D J2 into Rev. E J2. It will have a 25-pin plug, a PC board which reverses the connections from pins 6 to 13, and a 25-pin socket connector. If you're developing a device which plugs into the Sol J2, reverse the order of pins 6 through 13 and use this adapter (PTC pt. 900011) to ensure that your plug-in device will be compatible with E revision Sol's.

7. Current Loop Fix

The bug: R23 and R24 should be connected to +12 volts instead of +5 volts.

The squasher: Break the large trace on the solder side of the board which lead to these two resistors. Still on the solder side, connect a jumper from the isolated end of R23 to the +12 volt feedthrough as shown. Be sure that you do not accidentall connect to the −12 volt feedthrough, which is slightly higher than the +12 volt one.

And a Bug in 5K(pre-Sol)

The bug: Our attention has been called to some problems with the integer function in BASIC-5—the non-Sol version.

The squasher: Page 32 of the 5K BASIC manual (Software #2) should be changed to read as follows:

```
                                BASIC-5 INTEGER FIX
@BAA                    ORG     @BAAH
@BAA BA
@BAB D6 81
@BAD 16 05
@BAF FA BF 08
@BB2 00
@BB3 D6 05
@BB5 D0
@BB6 57
@BB7 0B
@BB8 0B
@BB9 0A
@BBA E6 F0
@BBC 02
@BBD 16
@BBE C9
@BBF AF
@BC0 02
@BC1 14
@BC2 C2 B8 0B
@BC5 C9
```

```
@@00 *
@@01 *
@@02 *                          BASIC-5 INTEGER FIX
@@03 *
@@04 *
@@05 *                  ORG     @BAAH
@@06 AINT       LDAX    B
@@07            SUI     129
@@08            MVI     D,FPSIZ
@@09            JM      AINT2
@@10            NOP                     SPACE
@@11 *
@@12 *  EXP > 0
@@13 *
@@14            SUI     FPNIB-1
@@15            RNC
@@16            MOV     D,A             COUNT
@@17            DCX     B
@@18 *
@@19 AINT2      DCX     B
@@20            LDAX    B
@@21            ANI     36@Q
@@22            STAX    B
@@23            INR     D
@@24            RZ
@@25 *
@@26 AINT3      XRA     A
@@27            STAX    B
@@28            INR     D
@@29            JNZ     AINT2
@@30            RET
@@31 *
@@32 *
```

Ninety-Ninety 'Rule of Project Schedules: The first ninety percent of the task takes ninety percent of the time and the last ten percent takes the other ninety percent.

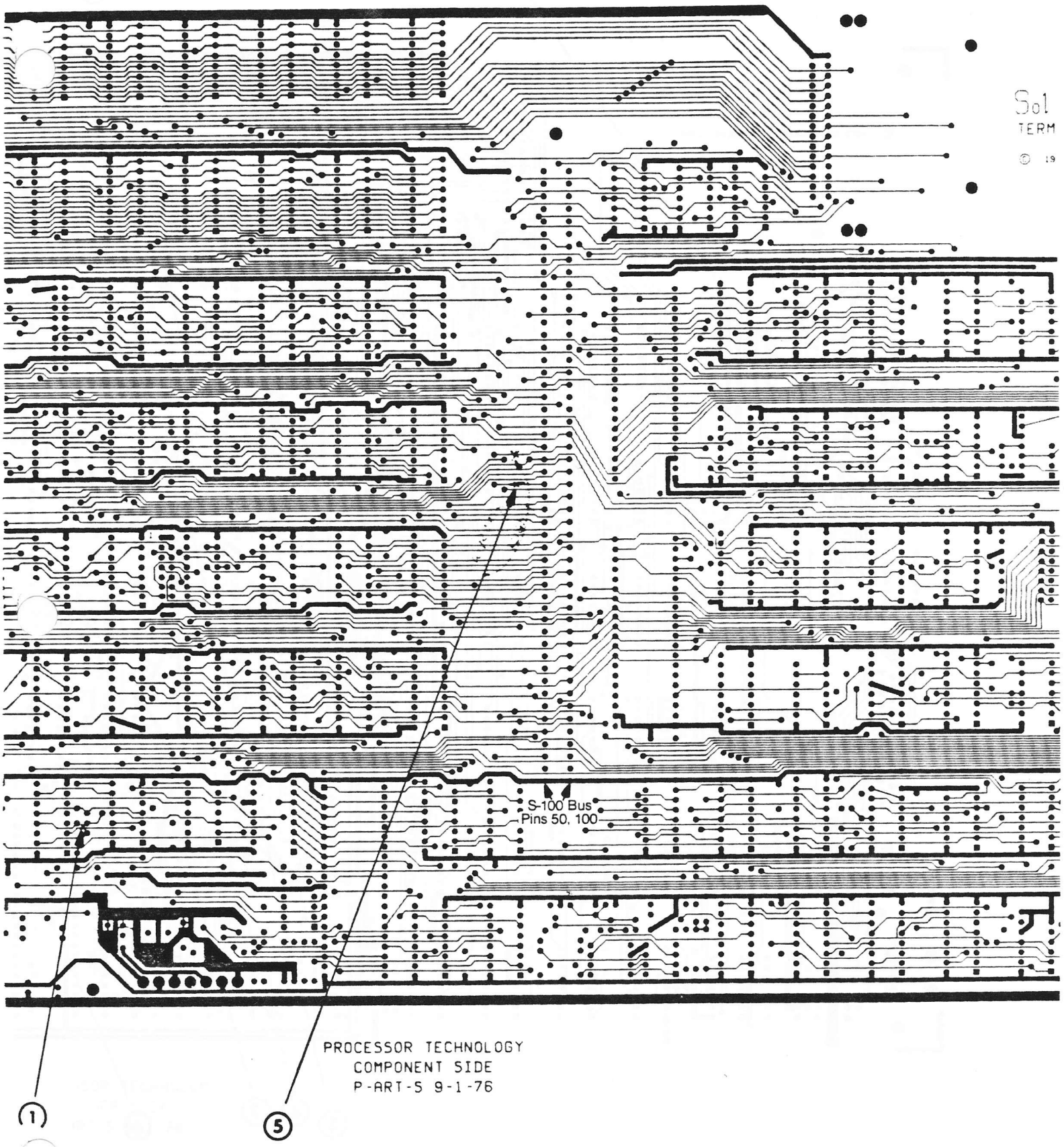
Your ALS-8 Application Notes are on the Way!

After a seemingly interminable delay, the first batch of Application notes are really and truly in the mail to all you ALS-8 Users' Group members. Once the material is in your hot little hands (in a very fancy binder, no less), you'll be better able to appreciate the power and versatility of your ALS-8, and you should have some very happy hours of computing.

If you haven't received your notes by the time you read this, please drop us a note right now, so we can track down what ever clerical or shipping errors crossed you up.

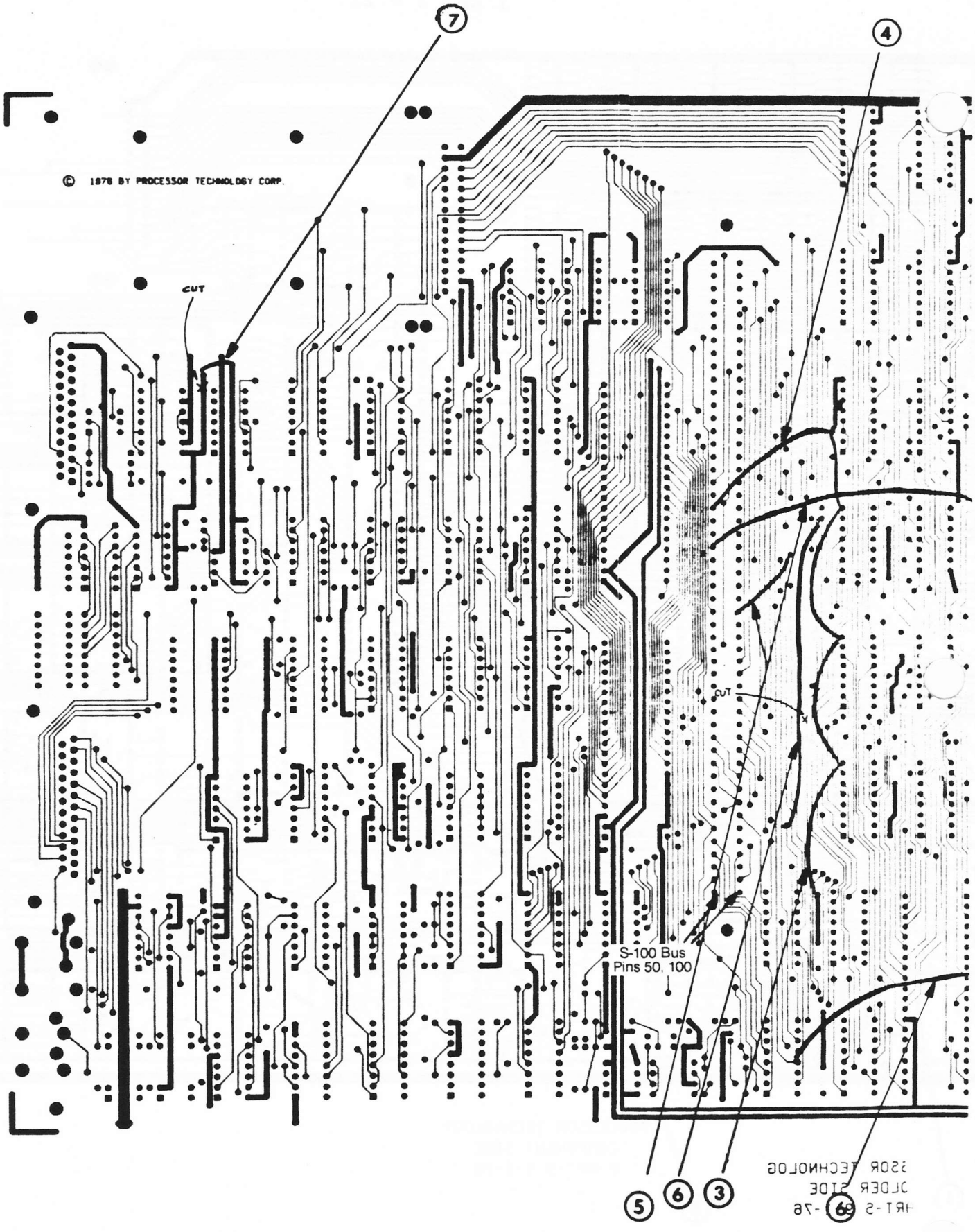
Our sincerest apologies for the delay.

Weinberg's Law: If builders built buildings the way programmers wrote programs then the first woodpecker that came along would destroy civilization.

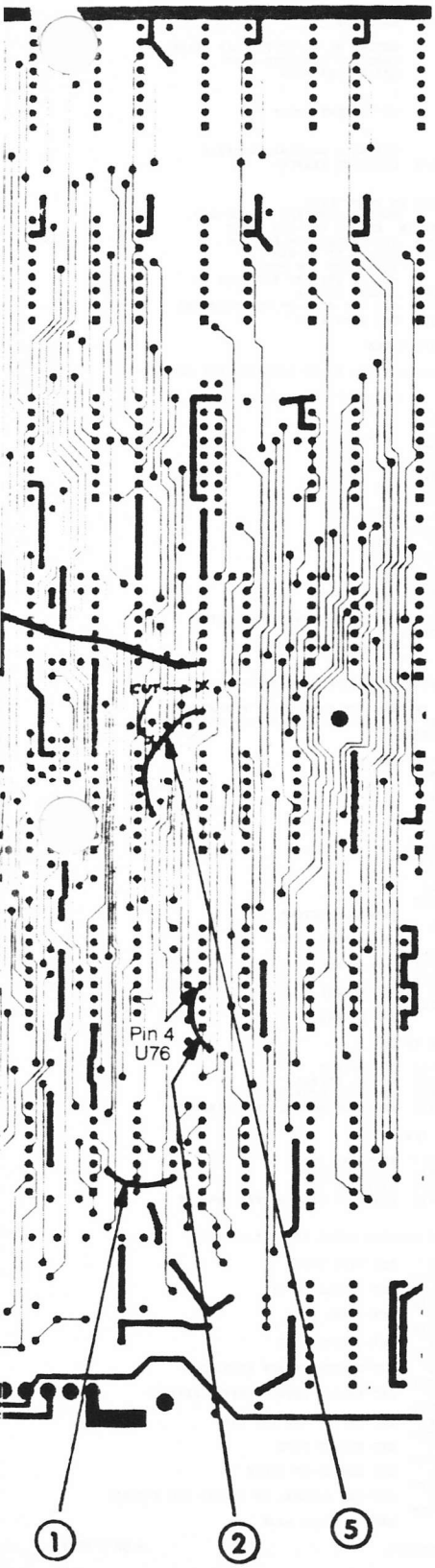


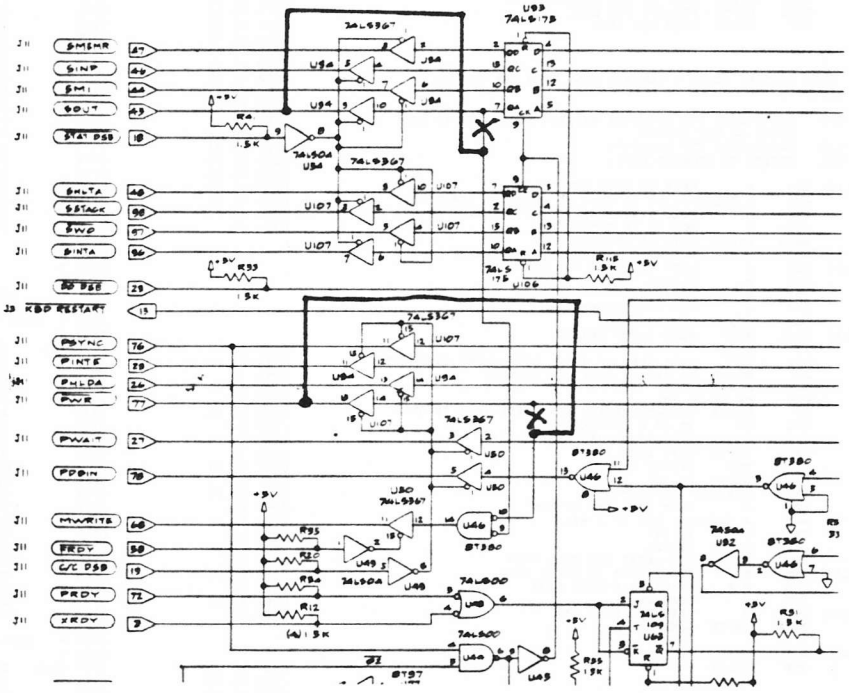
Sol-PC (Rev. D)—Component Side

Figure A



Sol-PC (Rev. D)—Solder Side
Figure B


CUT
Pin 4
U76
1
2
5


SMEMR
SINP
SM1
SOUT
STAT DSB
SHLTA
SSTACK
SWO
SINTA
INTE DSB
RST 7
SYNC
INTE
PHLDA
PWR
PWAIT
PDBIN
MWRITE
RDY
C/C DSB
PRDY
XRDY
SCHEMATIC, CPU & BUS, SOI
ROBERT M MARSH
PROCESSOR TECHNOLOGY
REV D

MWRITE Fix

(continued from page 5)

```
                        ORA     A       SEE IF THIS THE VDM
                        JNZ     STRTB   NO--DO NOT RESET VDM
                        LXI     SP,SYSTP  SET UP THE STACK FOR CALL
                        CALL    PERSE   (REG A ASSUMED TO COME BACK ZERO)
                STRTB   EQU     $       FINISH OFF THIS PORT THEN DO NEXT
                        LXI     H,0     USE FOR CLEARING USER ADDRESSES
                        CPI     3       IS IT A USER PORT
                        JZ      STRTC   YES-- DO NOT CLEAR IT
                        SHLD    UOPRT   NO--CLEAR ADDR
                STRTC   EQU     $       OUTPUT PORT ALL SET
                        MOV     A,B     FM SENSE SWITCHES
                        RAR     .
                        RAR     .       NEXT 2 BITS ARE INPUT PORT
                        ANI     3       VALID PORT
                        STA     DFLTS   THIS IS DEFAULT INPUT PORT
                        CPI     3       IS THIS ONE A USER PORT
                        JZ      STRTD   YES--DO NOT CLEAR IT THEN
                        SHLD    UIPRT   NO--FORCE USER ADDRESS ZERO
                STRTD   EQU     $       1ST TIME INITIALIZATION ALL DONE NOW
                        LHLD    DFLTS   PICK UP DEFAULT PORTS
                        SHLD    IPORT   FORCE PORTS TO DEFAULT
                CONN1   EQU     $       HERE TO TURN OFF TAPES, THEN COMMAND MODE
                        XRA     A
                        OUT     TAPPT   BE SURE TAPES ARE OFF
                *
                *
                *
                *               ---  COMMAND MODE  ---
                *
                *
                *
                *   THIS ROUTINE GETS AND PROCESSES COMMANDS
                *
                COMND   LXI     SP,SYSTP  SET STACK POINTER
                        CALL    PROMPT  PUT PROMPT ON SCREEN
                        CALL    GCLIO   INIT TO GET COMMAND LINE
                        CALL    COPRC   PROCESS THE LINE
                        JMP     COMND   OVER AND OVER
                *
                *
                *
                *   THIS ROUTINE READS A COMMAND LINE FROM THE SYSTEM
                *  KEYBOARD
                *
                *  C/R    TERMINATES THE SEQUENCE ERASING ALL CHARS TO THE
                *         RIGHT OF THE CURSOR
                *  L/F    TERMINATES THE SEQUENCE
                *  ESC    RESETS TO COMMAND MODE.
                *
                GCLIO   EQU     $       HERE TO INIT FOR GCLIN
                        LXI     H,INLIN-1  PT TO CHAR IN FRONT OF INPUT BFR
                        MVI     M,7     MAKE SURE IT IS "BELL" TO KEEP FM DEL'ING TOO FAR
                        INX     H       NOW PT TO INPUT BFR
                        SHLD    INPTR   SAVE AS STARTING PTR
                        MVI     A,80    NUMBER OF CHARS IN LINE (MAX)
                GCLI1   EQU     $       LOOP TO BLANK OUT LINE BFR
                        MVI     M,      BLANKS
                        INX     H       NEXT CHAR
                        DCR     A       FOR THIS COUNT
                        JNZ     GCLI1   ENTIRE LINE
                GCLIN   CALL    SINP    READ INPUT DEVICE
                        JZ      GCLIN
                        ANI     7FH     MAKE SURE NO X'80' BIT DURING CMND MODE
                        JZ      STRTD   IF EITHER MODE (OR CTL-@)
                        MOV     B,A
                        CPI     CR IS IT CR?
                        JZ      CLIN2   YES--TERMINATE LINE HERE (CLR IF VDM)
                        CPI     LF      IS IT A LINEFEED
                        NZ      .       YES--TERMINATE LINE AS IS
                        LHLD    INPTR   CRNT LINE PTR
                        CPI     7FH     DELETE CHR?
                        JNZ     GCLI2   NO--OK
                        MVI     B,BACKS  REPLACE IT
                        DCX     H       BACK LINE PTR UP TOO
                        MVI     A,'G'-40H  SEE IF A BELL
                        CMP     M       IS IT?
                        JNZ     GCLI3   NO--OK
                        MOV     B,A     YES--RING THE BELL THEN
                GCLI2   EQU     $       STORE CHAR IN INPUT AREA
                        MOV     M,B     PLACE CHAR INTO LINE
                        INX     H       NEXT CHAR
                GCLI3   EQU     $       SAVE NEW LINE PTR
                        SHLD    INPTR   SAVE PTR
                *
                CONT    CALL    SOUT
                        JMP     GCLIN
                *
                *
                *
                *
                *       FIND AND PROCESS COMMAND
                *
                COPRC   EQU     $       PROCESS THIS COMMAND LINE
                        CALL    STUP    SETUP TO PROCESS INPUT LINE
                        XCHG    .       DE=ADDR
                        LXI     H,START  PREP SO THAT HL WILL PT TO CUTER LATER
                        PUSH    H       PLACE PTR TO CUTER ON STACK FOR LATER DISPT
                        CALL    SCHR    SCAN PAST BLANKS
                        JZ      ERR1    NO COMMAND?
                        XCHG    .       HL HAS FIRST CHR
                        LXI     D,COMTAB  POINT TO COMMAND TABLE
                        CALL    FDCOM   SEE IF IN PRIMARY TABLE
                        CZ      FDCOU   TRY CUSTOM ONLY IF NOT PRIMARY COMMAND
                DISP0   EQU     $       HERE TO EITHER DISPATCH OR DO ERROR
                        JZ      ERR2    NOT IN EITHER TABLE
                        INX     D       PT DE TO ADDR OF RTN
                        XCHG    .       HL=ADDR OF ADDR OF RTN
                * **** DROP THRU TO DISPT ***
                *
                * THIS ROUTINE DISPTACHES TO THE ADDR AT CONTENTS OF HL.
                * HL ARE RESTORED PRIOR TO GOING TO ROUTINE.
                *
                DISPT   EQU     $       DISPATCH
                        MOV     A,M     LOW BYTE
                        INX     H
                        MOV     H,M     HI BYTE
                        MOV     L,A     AND LO, HL NOW COMPLETE
                DISP1   EQU     $       HERE TO GO OFF TO HL DIRECTLY
                        XTHL    .       HL RESTORED AND ADDR ON STACK
                        MOV     A,L     ALWAYS PASS L IN "A" (PRIMARILY FOR SET'S)
                        RET     .       OFF TO ROUTINE
                *
                *
                *
                *   THIS ROUTINE SEARCHES THROUGH A TABLE, POINTED TO
                *  BY 'DE', FOR A DOUBLE CHARACTER MATCH OF THE 'HL'
                *  MEMORY CONTENT.  IF NO MATCH IS FOUND THE SCAN ENDS
                *  WITH THE ZERO FLAG SET, ELSE NON-ZERO SET.
                *
                FDCOU   EQU     $       HERE TO SCAN CUSTOM TABLE
                        LXI     D,CUTAB  PT TO CUSTOM RTN TBL
                FDCOM   LDAX    D
                        ORA     A       TEST FOR TABLE END
                        RZ      .       NOT FOUND POST THAT AND RETURN
                        PUSH    H       SAVE START OF SCAN ADDRESS
                        CMP     M       TEST FIRST CHR
                        INX     D
                        JNZ     NCOM
                *
                        INX     H
                        LDAX    D
                        CMP     M       NOW SECOND CHARACTER
                        JNZ     NCOM    GOODNESS
                *
                        POP     H       RETURN HL TO PT TO CHAR START
                        ORA     A       FORCE TO NON-ZERO FLAG
                        RET     .       LET CALLER KNOW
                *
                *
                NCOM    INX     D       GO TO NEXT ENTRY
                        INX     D
                        INX     D
                        POP     H       GET BACK ORIGINAL ADDRESS
                        JMP     FDCOM   CONTINUE SEARCH
                *
                *
                * SET UP TO PROCESS AN INPUT LINE
                STUP    EQU     $       PREPARE WHETHER VDM OR NOT
                        LXI     H,INLIN  ASSUME NON-VDM INPUT
                        SHLD    INPTR   ALSO RESET PTR FOR NOW
                        LDA     OPORT   SEE IF IT IS VDM
                        ORA     A       IS IT THE VDM PORT
                        RNZ     .       NO--HL ARE SET PROPERLY
                        CALL    CREM    REMOVE CURSOR
                        MVI     C,1     GET VDM ADDR FM POSITION ONE
                        JMP     VDAD2   GET SCRN ADDR
                *
                *               COMMAND TABLE
                *
                *  THIS TABLE DESCRIBES THE VALID COMMANDS FOR CUTER
                *
                COMTAB  EQU     $       START OF KNOWN COMMANDS
                        ASC     'DU'    DUMP
                        DW      DUMP
                        ASC     'EN'    ENTR
                        DW      ENTER
                        ASC     'EX'    EXEC
                        DW      EXEC
                        ASC     'GE'    GET
                        DW      TLOAD
                        ASC     'SA'    SAVE
                        DW      TSAVE
                        ASC     'XE'    XEQ
                        DW      TXEQ
                        ASC     'CA'    CAT
                        DW      TLIST
                        ASC     'SE'    SET COMMAND
                        DW      SET
                        ASC     'CU'    CUSTOM COMMAND ENTER/CLEAR
                        DW      CUSET
                        DB      0       END OF TABLE MARK
                *
                *
                *               DISPLAY DRIVER COMMAND TABLE
                *
                *    THIS TABLE DEFINES THE CHARACTERS FOR SPECIAL
                *  PROCESSING.  IF THE CHARACTER IS NOT IN THE TABLE IT
                *  GOES TO THE SCREEN.
                *
                TBL     DB      CLEAR   SCREEN
                        DW      PERSE
                        DB      UP      CURSOR
                        DW      PUP
                        DB      DOWN    "
                        DW      PDOWN
                        DB      LEFT    "
                        DW      PLEFT
                        DB      RIGHT   "
                        DW      PRIT
                        DB      HOME    "
                        DW      PHOME
                        DB      CR      CARRIAGE RETURN
                        DW      PCR
                        DB      LF      LINE FEED
                        DW      PLF
                        DB      BACKS   BACK SPACE
                        DW      PBACK
                        DB      ESC     ESCAPE KEY
                        DW      PESC
                        DB      0       END OF TABLE
                *
                *   OUTPUT DEVICE TABLE
                *
                OTAB    DW      VDM01   VDM DRIVER
                        DW      SEROT   SERIAL OUTPUT
                        DW      PAROT   PARALLEL OUTPUT
                        DW      ERROT   ERROR OR USER DRIVER HANDLER
                *
                *   INPUT DEVICE TABLE
                *
                ITAB    DW      KREA1   KEYBOARD INPUT
                        DW      SREA1   SERIAL INPUT
                        DW      PARIT   PARALLEL INPUT
                        DW      ERRIT   ERROR OR USER DRIVER HANDLER
                *
                *
                *       SECONDARY COMMAND TABLE FOR SET COMMAND
                *
                SETAB   ASC     'TA'    SET TAPE SPEED
                        DW      TASPD
                        ASC     'SP'    SET DISPLAY SPEED
                        DW      DISPD
                        ASC     'IP'    SET INPUT PORT
                        DW      SETIN
                        ASC     'OP'    SET OUTPUT PORT
                        DW      SETOT
                        ASC     'CI'    SET CUSTOM DRIVER ADDRESS
                        DW      SETCI
                        ASC     'CO'    SET CUSTOM OUTPUT DRIVER ADDRESS
                        DW      SETCO
                        ASC     'XE'    SET HEADER XEQ ADDRESS
                        DW      SETXQ
                        ASC     'TY'    SET HEADER TYPE
                        DW      SETTY
                        ASC     'NU'    SET NUMBER OF NULLS
                        DW      SETNU
                        ASC     'CR'    SET CRC (NORMAL OR IGNORE CRC ERRORS)
                        DW      SETCR
                        DB      0       END OF TABLE MARK
                * -0-
                9999    COPY    CUTER2/1                                  2 OF 3 ****
                *
                *
                *       OUTPUT A CRLF FOLLOWED BY A PROMPT
                *
                PROMPT  CALL    CRLF
                        MVI     M,'>'   THE PROMPT
                        JMP     SOUT    PUT IT ON THE SCREEN
                *
                CRLF    MVI     B,LF    LINE FEED
                        CALL    SOUT
                        MVI     B,CR    CARRIAGE RETURN
                        CALL    SOUT
                        LDA     NUCNT   GET COUNT OF NULLS TO OUTPUT
                        MOV     C,A     SAVE COUNT IN C
```

```
                        0807 NULOT  DCR   C
                        0808        RM    .       COUNTED DOWN PAST ZERO (MAY COUNT IS X'7F')
                        0809        XRA   A       HERE IS THE NULL
                        0810        CALL  OUTH    OUTPUT IT
                        0811        JMP   NULOT   LOOP FOR NUMBER OF NULLS
                        0812 *
                        0813 *
                        0814 *  SCAN OVER UP TO 12 CHARACTERS LOOKING FOR A BLANK
                        0815 *
                        0816 SBLK   MVI   C,12    MAXIMUM COMMAND STRING
                        0817 SBLK1  LDAX  D
                        0818        CPI   BLANK
                        0819        JZ    SCHR    GOT A BLANK NOW SCAN PAST IT
                        0820        INX   D
                        0821        CPI   '='     A EQUAL WILL ALSO STOP US (AT NEXT CHAR)
                        0822        JZ    SCHR    FOUND, DE PT TO NEXT CHAR
                        0823        DCR   C       NO MORE THAN TWELVE
                        0824        JNZ   SBLK1
                        0825        RET   .       GO BACK WITH ZERO FLAG SET
                        0826 *
                        0827 *
                        0828 *  SCAN PAST UP TO 10 BLANK POSITIONS LOOKING FOR
                        0829 *  A NON BLANK CHARACTER.
                        0830 *
                        0831 SCHR   MVI   C,10    SCAN TO FIRST NON BLANK CHR WITHIN 10
                        0832 SCHR1  LDAX  D       GET NEXT CHARACTER
                        0833        CPI   SPACE
                        0834        RNZ   .       WE'RE PAST THEM
                        0835        INX   D       NEXT SCAN ADDRESS
                        0836        DCR   C
                        0837        RZ    .       COMMAND ERROR
                        0838        JMP   SCHR1   KEEP LOOPING
                        0839 *
                        0840 *  THIS ROUTINE SCANS OVER CHARACTERS, PAST BLANKS AND
                        0841 *  CONVERTS THE FOLLOWING ADDRESS TO HEX.  ERRORS RETURN TO
                        0842 *  THE ERROR HANDLER.
                        0843 *
C418 CD 78 C3           0844 SCONV  CALL  SBLK
C417 E5                 0845        JZ    ERR1
                        0846 *
C418 CD 42 C3           0847 *  THIS ROUTINE CONVERTS ASCII DIGITS INTO BINARY FOLLOWING
C41B 06 3A              0848 *  A STANDARD HEX CONVERSION.  THE SCAN STOPS WHEN AN ASCII
C41D CD 19 C0           0849 *  SPACE IS ENCOUNTERED.  PARAMETER ERRORS REPLACE THE ERROR
C420 CD 27 C2           0850 *  CHARACTER ON THE SCREEN WITH A QUESTION MARK.
C423 CD AA C2           0851 *
C426 EB                 0852 $HEX   LXI   H,0     CLEAR H & L
                        0853 SHE1   LDAX  D       GET CHARACTER
                        0854        CPI   20H     IS IT A SPACE?
C427 0E 03              0855        RZ    .       IF SO
C429 CD 6E C3           0856        CPI   '/'
C42C CA 18 C4           0857        RZ
                        0858        CPI   ':'
C42F FE 2F              0859        RZ
C431 CA 18 C2           0860 *
C434 CD 7E C3           0861 HCONV  DAD   H       MAKE ROOM FOR THE NEW ONE
C437 FE 3A              0862        DAD   H
C439 CA 44 C4           0863        DAD   H
C43C 7D                 0864        DAD   H
C43D E1                 0865        CALL  HCOV1   DO THE CONVERSION
C43E 77                 0866        JNC   ERR1    NOT VALID HEXIDECIMAL VALUE
C43F 23                 0867        ADD   L
C440 E5                 0868        MOV   L,A     MOVE IT IN
C441 C3 27 C4           0869        INX   D       BUMP THE POINTER
                        0870        JMP   SHE1
                        0871 *
C444 E3                 0872 HCOV1  SUI   48      REMOVE ASCII BIAS
C445 13                 0873        CPI   10
C446 C3 27 C4           0874        RC    .       IF LESS THAN 9
                        0875        SUI   7       IT'S A LETTER??
                        0876        CPI   10H
                        0877        RET   .       WITH TEST IN HAND
                        0878 *
                        0879 *
                        0880 *  THIS ROUTINE WILL SEE IF A FIELD (OPERAND) IS PRESENT.
                        0881 *  IF NOT, THEN HL WILL REMAIN AS THEY WERE ON ENTRY.
                        0882 *  IF IT WAS PRESENT, THEN HL=THAT VALUE IN HEX.
                        0883 *
C3A5                    0884 PSCAN  EQU   $       OPTIONAL FIELD SCANNER
3A5 CD 59 C3            0885        CALL  SBLK    SEE IF FIELD IS PRESENT
3A9 C8                  0886        RZ    .       RETURN LEAVING HL AS THEY WERE ON ENTRY
3A9 CD 7E C3            0887        CALL  $HEX    FIELD IS THERE, GO GET IT
3AC C9                  0888        RET   .       HL= EITHER OPTIONAL FIELD (HEX), OR AS IT WAS
                        0889 *
                        0890 *
                        0891 *
                        0892 *
                        0893 *            DUMP COMMAND
                        0894 *
                        0895 *      THIS ROUTINE DUMPS CHARACTERS FROM MEMORY TO THE
                        0896 *  CURRENT OUTPUT DEVICE.
                        0897 *  ALL VALUES ARE DESPLAYED AS ASCII HEX.
                        0898 *
                        0899 *  THE COMMAND FORM IS AS FOLLOWS:                          C451
                        0900 *                                           C451 21 1C C8
                        0901 *         DUMP  ADDR1  ADDR2                C45A CD 59 C3
                        0902 *                                           C457 06 06
                        0903 *    THE VALUES FROM ADDR1 TO ADDR2 ARE THEN OUTPUT TO THE
                        0904 *  OUTPUT DEVICE.  IF ONLY ADDR1 IS SPECIFIED THEN THE      C459 1A
                        0905 *  VALUE AT THAT ADDRESS IS OUTPUT.                         C45A FE 20
                        0906 *                                           C45C CA 80 C4
                        0907 *  IF WHILE DUMPING, THE MODE KEY IS PRESSED, THE DUMP WILL C45F FE 2F
                        0908 *  BE TERMINATED.  IF THE SPACE BAR IS PRESSED, THE DUMP    C461 CA 80 C4
                        0909 *  WILL BE TEMPORARILY SUSPENDED UNTIL ANY KEY IS PRESSED.  C464 77
                        0910 *                                           C465 13
C3AD                    0911 DUMP   EQU   $       SET UP REGS TO DUMP SPECIFIED AREA C466 23
3AD CD 78 C3            0912        CALL  SCONV   GET START ADDR (REQUIRED)          C467 05
3B0 E5                  0913        PUSH  H       SAVE THE START ADDR                C468 C2 59 C4
3B1 CD A5 C3            0914        CALL  PSCAN   GET OPTIONAL END ADDR, HL=THIS OR START ADDR
3B4 D1                  0915        POP   D       DE=START ADDR
3B5 EB                  0916        XCHG  .       DE=END ADDR, HL=START ADDR NOW
                        0917 *                                           C46B EB
3B6 CD 42 C3            0918 DLOOP  CALL  CRLF                           C46C 36 3F
3B9 CD D9 C3            0919        CALL  ADOUT   OUTPUT ADDRESS           C46E 3A D7 C8
3BC CD F7 C3            0920        CALL  BOUT    ANOTHER SPACE TO KEEP IT PRETTY
3BF 0E 10               0921        MVI   C,16    VALUES PER LINE          C471 B7
                        0922 *                                           C472 CA 18 C2
3C1 7E                  0923 DLP1   MOV   A,M     GET THE CHR              C475 CD 42 C3
3C2 C5                  0924        PUSH  B       SAVE VALUE COUNT         C478 06 3F
3C3 CD DE C3            0925        CALL  HBOUT   SEND IT OUT WITH A BLANK C47A CD 19 C0
3C6 7C                  0926        MOV   A,H     CRNT ADDR                C47D C3 18 C2
3C7 BA                  0927        CMP   D       VERSUS ENDING ADDR
3C8 DA D0 C3            0928        JC    DLP1A   NOT DONE YET
3CB 7D                  0929        MOV   A,L     TRY LOW ORDER BYTE
3CC BB                  0930        CMP   E
3CD D2 18 C2            0931        JNC   COMND   ALL DONE WHEN CRNT REACHES ENDING
                        0932 DLP1A  EQU   $       HERE TO KEEP DUMPING     C480 36 00
3D0                     0933        POP   B       VALUES PER LINE          C482 23
3D1                     0934        INX   H                                C483 05
3D2 C                   0935        DCR   C       BUMP THE LINE COUNT      C484 C2 80 C4
3D3 C                   0936        JNZ   DLP1    NOT ZERO IF MORE FOR THIS LINE
3D6 C3                  0937        JMP   DLOOP   DO A LFCR BEFORE THE NEXT C487 FE 2F
                        0938 *                                           C489 3E 01
                        0939 *    OUTPUT HL AS HEX 16 BIT VALUE          C48B C2 94 C4
                        0940 *                                           C48E 13
3D9 7C                  0941 ADOUT  MOV   A,H     H FIRST                C48F CD 6C C3
3DA CD FC C3            0942        CALL  HBOUT                          C492 D6 30
3DD 7D                  0943        MOV   A,L     THEN L FOLLOWED BY A SPACE
                        0944 *                                           C494

                        0945 HBOUT  CALL  HEOUT
                        0946        CALL  SINP    SEE IF WE SHD ESCAPE FM DUMP
                        0947        JZ    BOUT    NO--ADD THE SPACE THEN
                        0948        ANI   7FH     MAKE SURE ITS CLEAR OF PARITY
                        0949        JZ    COMND   EITHER MODE (OR CTL-@)
                        0950        CPI   ' '     IS IT SPACE
                        0951        JNZ   BOUT    NO--IGNORE THE CHAR
                        0952 WTLP1  CALL  SINP    ON SPACE, WAIT FOR ANY OTHER CHAR
                        0953        JZ    WTLP1   JUST LOOP AFTER A SPACE UNTIL ANY KEY PRESSED
                        0954 BOUT   MVI   B,' '
                        0955        JMP   SOUT    PUT IT OUT
                        0956 *
                        0957 HEOUT  MOV   C,A     GET THE CHARACTER
                        0958        RRC
                        0959        RRC           MOVE THE HIGH FOUR DOWN
                        0960        RRC
                        0961        RRC
                        0962        CALL  HEOU1   PUT THEM OUT
                        0963        MOV   A,C     THIS TIME THE LOW FOUR
                        0964 *
                        0965 HEOU1  ANI   0FH     FOUR ON THE FLOOR
                        0966        ADI   48      WE WORK WITH ASCII HERE
                        0967        CPI   58      0-9?
                        0968        JC    OUTH    YUP!
                        0969        ADI   7       MAKE IT A LETTER
                        0970 OUTH   MOV   B,A     OUTPUT IT FROM REGISTER 'B'
                        0971        JMP   SOUT
                        0972 *
                        0973 *
                        0974 *           ENTR COMMAND
                        0975 *
                        0976 *   THIS ROUTINE GETS VALUES FROM THE KEYBOARD AND ENTERS
                        0977 *  THEM INTO MEMORY.  THE INPUT VALUES ARE SCANNED FOLLOWING
                        0978 *  A STANDARD 'GCLIN' INPUT SO ON SCREEN EDITING MAY TAKE
                        0979 *  PLACE PRIOR TO THE LINE TERMINATOR.  A SLASH '/'
                        0980 *  ENDS THE ROUTINE AND RETURNS CONTROL TO THE COMMAND MODE.
                        0981 *
                        0982 ENTER  CALL  SCONV  SCAN OVER CHARS AND GET ADDRESS
                        0983        PUSH  H       SAVE ADDRESS
                        0984 *
                        0985 ENLOP  CALL  CRLF
                        0986        MVI   B,':'
                        0987        CALL  SOUT    DSPLY THE COLON
                        0988        CALL  GCLIO   INIT AND PROCESS A LINE
                        0989        CALL  STUP    SET UP TO PROCESS INPUT LINE
                        0990        XCHG  .       ....TO DE
                        0991 *
                        0992 *
                        0993 ENLO1  MVI   C,3     NO MORE THAN THREE SPACES BETWEEN VALUES
                        0994        CALL  SCHR1   SCAN TO NEXT VALUE
                        0995        JZ    ENLOP   LAST ENTRY FOUND START NEW LINE
                        0996 *
                        0997        CPI   '/'     COMMAND TERMINATOR?
                        0998        JZ    COMND   IF SO...
                        0999        CALL  $HEX    CONVERT VALUE
                        1000        CPI   ':'     ADDRESS TERMINATOR?
                        1001        JZ    ENLO3   GO PROCESS IF SO
                        1002        MOV   A,L     GET LOW PART AS CONVERTED
                        1003        POP   H       GET MEMORY ADDRESS
                        1004        MOV   M,A     PUT IN THE VALUE
                        1005        INX   H
                        1006        PUSH  H       BACK GOES THE ADDRESS
                        1007        JMP   ENLO1   CONTINUE THE SCAN
                        1008 *
                        1009 ENLO3  XTHL  .       PUT NEW ADDRESS ON STACK
                        1010        INX   D       MOVE SCAN PAST TERMINATOR
                        1011        JMP   ENLO1
                        1012 *
                        1013 *
                        1014 *           EXECUTE COMMAND
                        1015 *
                        1016 *   THIS ROUTINE GETS THE FOLLOWING PARAMETER AND DOES A
                        1017 *  PROGRAM JUMP TO THE LOCATION GIVEN BY IT.  IF PROPER
                        1018 *  STACK OPERATIONS ARE USED WITHIN THE EXTERNAL PROGRAM
                        1019 *  IT CAN DO A STANDARD 'RET'URN TO THE CUTER COMMAND MODE.
                        1020 *
                        1021 *
                        1022 EXEC   CALL  SCONV  SCAN PAST BLANKS AND GET PARAMETER
                        1023 EXEC1  EQU   $       HERE TO GO TO HL
                        1024        PUSH  H       SAVE ON STACK
                        1025        LXI   H,START  LET USER KNOW WHERE WE ARE
                        1026        RET   .       AND OFF TO USER
                        1027 *
                        1028 *
                        1029 *
                        1030 *
                        1031 *   THIS ROUTINE GETS A NAME OF UP TO 5 CHARACTERS
                        1032 *  FROM THE INPUT STRING.  IF THE TERMINATOR IS A
                        1033 *  SLASH (/) THEN THE CHARACTER FOLLOWING IS TAKEN
                        1034 *  AS THE CASSETTE UNIT SPECIFICATION.
                        1035 *
                        1036 *
                        1037 NAMEO  EQU   $       ENTER HERE TO SET HL TO THEAD
                        1038        LXI   H,THEAD  PT WHERE TO PUT NAME
                        1039 NAME   CALL  SBLK    SCAN OVER TO FIRST CHRS
                        1040        MVI   B,6
                        1041 *
                        1042 NAME1  LDAX  D       GET CHARACTER
                        1043        CPI   ' '     NO UNIT DELIMITER
                        1044        JZ    WFIL
                        1045        CPI   '/'     UNIT DELIMITER
                        1046        JZ    WFIL
                        1047        MOV   M,A
                        1048        INX   D       BUMP THE SCAN POINTER
                        1049        INX   H
                        1050        DCR   B
                        1051        JNZ   NAME1   NAME IS OK, FALL THRU TO 'ERR1' IF NOT
                        1052 *
                        1053 *     CUTER ERROR HANDLER
                        1054 *
                        1055 ERR1   XCHG  .       GET SCAN ADDRESS
                        1056 ERR2   MVI   M,'?'   FLAG THE ERROR
                        1057        LDA   OPORT   SEE IF VIA VDM DRIVER
                        1058        ORA   A
                        1059        JZ    COMND   YES--VDM SCREEN NOW WAS THE ?
                        1060        CALL  CRLF
                        1061        MVI   B,'?'   SET UP THE ????
                        1062        CALL  SOUT    INDICATE INPUT NOT VALID
                        1063        JMP   COMND   NOW READY FOR NEXT INPUT
                        1064 *
                        1065 *
                        1066 *
                        1067 *  HERE WE HAVE SCANNED OFF THE NAME.  ZERO FILL IN FOR
                        1068 *  NAMES LESS THAN FIVE CHARACTERS.
                        1069 *
                        1070 WFIL   MVI   M,0     PUT IN AT LEAST ONE ZERO
                        1071        INX   H
                        1072        DCR   B
                        1073        JNZ   WFIL    LOOP UNTIL B IS ZERO
                        1074 *
                        1075        CPI   '/'     IS THERE A UNIT SPECIFICATION?
                        1076        MVI   A,1     PRETEND NOT
                        1077        JNZ   DEFLT
                        1078        INX   D       MOVE PAST THE TERMINATOR
                        1079        CALL  SCHR    GO GET IT
                        1080        SUI   '0'     REMOVE ASCII BIAS
                        1081 *
                        1082 DEFLT  EQU   $       CNVRT TO INTERNAL BIT FOR TAPE CONTROL
```

```
C49A E6 01              1083          ANI     1       JUST BIT ZERO
C49C 3E 80              1084          MVI     A,TAPE1  ASSUME TAPE ONE
C49E C2 9C C4           1085          JNZ     STUNT   IF NON ZERO, IT IS ONE
C4A1 3E                 1086          MVI     .       ELSE MAKE IT TAPE TWO
C4  54 C8               1087 STUNT    STA     FNUMF   SET IT IN
C                       1088          RET
                        1089 *
                        1090 *
                        1091 *
                        1092 *    THIS ROUTINE PROCESSES THE XEQ AND GET COMMANDS
                        1093 *
                        1094 *
C4A0 3E                 1095 TXEQ     DB      3EH     THIS BEGINS "MVI" OF THE "XRA" FOLLOWING
C4A1 AF                 1096 TLOAD    XRA     A       A=0 TLOAD, A=AF (@0) THEN XEQ
C4A2 F5                 1097          PUSH    PSW     SAVE FLAG TO SAY WHETHER LOAD OR XEQ
C4A3 21 2C C8           1098          LXI     H,DHEAD  PLACE DUMMY HDR HERE FOR COMPARES
C4A6 CD 54 C4           1099          CALL    NAME    SET IN NAME AND UNIT
C4A9 21 00 00           1100          LXI     H,0     ASSUME LOAD ADDR NOT GIVEN
C4AC CD A5 C3           1101          CALL    PSCAN   HL EITHER =0, OR OVERRIDE LOAD ADDR
                        1102 *
C4AF EB                 1103 TLOA2    XCHG    .       PUT ADDRESS IN DE
C4B0 21 2C C8           1104          LXI     H,DHEAD  PT TO NORMAL HDR
C4B3 7E                 1105          MOV     A,M     GET 1ST CHAR OF NAME
C4B4 B7                 1106          ORA     A       IS THERE A NAME?
C4B5 C2 BB C4           1107          JNZ     TLOA3   YES--LOOK FOR IT
C4B8 21 1C C8           1108          LXI     H,THEAD  PT TO SAME HDR TO LOAD NEXT FILE
C4BB E5                 1109 TLOA3    PUSH    H       SAVE PTR TO WHICH HDR TO USE
C4BC CD A8 C5           1110          CALL    ALOAD   GET UNIT AND SPEED
C4BF E1                 1111          POP     H       RESTORE PTR TO PROPER HDR TO USE
C4C0 CD C7 C6           1112          CALL    RTAPE   READ IN THE TAPE
C4C3 DA 10 C5           1113          JC      TAERR   TAPE ERROR?
                        1114 *
C4C6 CD 4C C5           1115          CALL    WAOUT   PUT OUT THE HEADER PARAMETERS
C4C9 F1                 1116          POP     PSW     RESTORE FLAG SAYING WHETHER IT WAS LOAD OR XEQ
C4CA B7                 1117          ORA     A
C4CB C8                 1118          RZ      .       AUTO XEQ NOT WANTED
C4CC 3A 22 C8           1119          LDA     WTYPE   CHECK TYPE
C4CF B7                 1120          ORA     A       SET FLAGS
C4D0 FA 10 C5           1121          JM      TAERR   TYPE IS NON XEQ
C4D3 3A 21 C8           1122          LDA     THEAD+5
C4D6 B7                 1123          ORA     A
C4D7 C2 10 C5           1124          JNZ     TAERR   THE BYTE MUST BE ZERO FOR AUTO XEQ
C4DA 2A 27 C8           1125          LHLD    XEQAD   GET THE TAPE ADDRESS
C4DD C3 AC C4           1126          JMP     EXEC1   AND GO OFF TO IT
                        1127 *
                        1128 *
                        1129 *
                        1130 *    THIS ROUTINE IS USED TO SAVE PROGRAMS AND DATA ON
                        1131 *    THE CASSETTE UNIT.
                        1132 *
                        1133 *
      C4E0              1134 TSAVE    EQU     $       SAVE MEMORY IMAGE TO TAPE
C4E0 CD 51 C4           1135          CALL    NAMEO   GET NAME AND UNIT
C4E3 CD 78 C3           1136          CALL    SCONV   GET START ADDRESS
C4E6 E5                 1137          PUSH    H       SAVE START ADDR FOR SIZE COMPUTATION LATER
C4E7 CD 78 C3           1138          CALL    SCONV   GET END ADDR (REQUIRED)
C4EA E3                 1139          XTHL    .       HL=START ADDR NOW, STACK=END ADDR
C4EB E5                 1140          PUSH    H       STACK =START FOLLOWED BY END
C4EC CD A5 C3           1141          CALL    PSCAN   SEE IF RETRIEVE FROM ADDR
C4EF 22 25 C8           1142          SHLD    LOADR   EITHER ACTUAL START, OR OVERRIDE INTO HDR
C4F2 E1                 1143          POP     H       HL=START ADDR
C4F3 D1                 1144          POP     D       DE=END ADDR
C4F4 E5                 1145          PUSH    H       PUT START BACK ONTO STACK
C4F5 7B                 1146          MOV     A,E     SIZE=END-START+1
C4F6 95                 1147          SUB     L
C4  C                   1148          MOV     L,A
C4                      1149          MOV     A,D
C4      00              1150          SBI     0       THIS EQUALS A "SBB M"
C4FB 94                 1151          SUB     H       THIS IS NEEDED
C4FC 67                 1152          MOV     H,A
C4FD 23                 1153          INX     H
C4FE 22 23 C8           1154          SHLD    BLOCK   STORE THE SIZE
C501 E5                 1155          PUSH    H       SAVE AS THE BLOCK SIZE
                        1156 *
C502 CD A4 C5           1157          CALL    ALOAD   GET UNIT AND SPEED
C505 21 1C C8           1158          LXI     H,THEAD  PT TO HEADER TO WRITE
C508 CD AB C7           1159          CALL    WHEAD   TURN TAPE ON, THEN WRITE HEADER
C50B D1                 1160          POP     D       GET BACK THE SIZE
C50C E1                 1161          POP     H       AND GET BACK THE ACTUAL START ADDR
C50D C3 8C C7           1162          JMP     WTAP1   WRITE THE BLK (W/EXTRA PUSH)
                        1163 *
                        1164 *    OUTPUT ERROR AND HEADER
                        1165 *
C510 CD A2 C3           1166 TAERR    CALL    CRLF
C513 16 06              1167          MVI     D,6
C515 21 21 C5           1168          LXI     H,ERRM
C518 CD 66 C5           1169          CALL    WLOOP   OUTPUT ERROR
C51B CD 4C C5           1170          CALL    WAOUT   THEN THE HEADER
C51E C3 15 C2           1171          JMP     COMN1
                        1172 *
C521 A5 52 52 AF        1173 ERRM     ASC     !ERROR !
     52 20
                        1174 *
                        1175 *
                        1176 *          CAT COMMAND
                        1177 *
                        1178 *    THIS ROUTINE READS HEADERS FROM THE TAPE AND OUTPUTS
                        1179 *    THEM TO THE OUTPUT DEVICE.  IT CONTINUES UNTIL THE
                        1180 *    MODE KEY IS DEPRESSED.
                        1181 *
      C527              1182 TLIST    EQU     $       PRODUCE A LIST OF FILES ON A TAPE
C527 CD 51 C4           1183          CALL    NAMEO   GET UNIT IF ANY (NAME IS IGNORED)
C52A CD A2 C3           1184          CALL    CRLF    START ON A FRESH LINE
                        1185 *
                        1186 *
C52D CD A4 C5           1187 LLIST    CALL    ALOAD
C530 06 01              1188          MVI     B,1
C532 CD EB C7           1189          CALL    TON     TURN ON THE TAPE
C535 CD 1F C7           1190 LIST1    CALL    RHEAD
C538 DA 15 C2           1191          JC      COMN1   TRUN OFF THE TAPE UNIT
C53B C2 35 C5           1192          JNZ     LIST1
C53E CD AC C5           1193          CALL    WAOUT   OUTPUT THE HEADER
C541 C3 2D C5           1194          JMP     LLIST
                        1195 *
                        1196 *
                        1197 *    THIS ROUTINE GETS THE CASSETTE UNIT NUMBER AND
                        1198 *    SPEED TO REGISTER "A" FOR THE TAPE CALLS
                        1199 *
C544 21 5A C8           1200 ALOAD    LXI     H,FNUMF  POINT TO THE UNIT SPECIFICATION
C547 3A 0D C8           1201          LDA     TSPD    GET THE TAPE SPEED
C54A B6                 1202          ORA     M       PUT THEM TOGETHER
C54B C9                 1203          RET     .       AND GO BACK
                        1204 *
                        1205 *    THIS ROUTINE OUTPUTS THE NAME AND PARAMETERS OF
                        1206 *    THEAD TO THE OUTPUT DEVICE.
                        1207 *
                        1208 *
C54     0B              1209 WAOUT    MVI     D,8
C54E 21 1B C8           1210          LXI     H,THEAD-1  POINT TO THE HEADER
C551 CD 66 C5           1211          CALL    WLOOP   OUTPUT THE HEADER
C554 CD F7 C3           1212          CALL    BOUT    ANOTHER BLANK
C557 2A 25 C8           1213          LHLD    LOADR   NOW THE LOAD ADDRESS
C55A CD D9 C3           1214          CALL    ADOUT   PUT IT OUT
C55D 2A 23 C8           1215          LHLD    BLOCK   AND THE BLOCK SIZE
C560 CD D9 C3           1216          CALL    ADOUT
C563 C3 A2 C3           1217          JMP     CRLF    DO THE CRLF AND RETURN

C566 7E                 1218 *
C567 B7                 1219 *
C568 C2 6D C5           1220 WLOOP    MOV     A,M     GET CHARACTER
C56B 3E 20              1221          ORA     A
      C56D              1222          JNZ     CHRLI   IF IT ISN'T A ZERO
C56D CD 10 C4           1223          MVI     A,' '   SPACE OTHERWISE
C570 23                 1224 CHRLI    EQU     $       CHAR IS OK TO SEND
C571 15                 1225          CALL    OUTH    OUTPUT IT FROM A REG
C572 C2 66 C5           1226          INX     H
C575 C9                 1227          DCR     D
                        1228          JNZ     WLOOP
                        1229          RET
                        1230 *
                        1231 *
                        1232 *
                        1233 *
                        1234 *       "SET" COMMAND
                        1235 *
                        1236 *    THIS ROUTINE GETS THE ASSOCIATED PARAMETER AND
                        1237 *    DISPATCHES TO THE PROPER ROUTINE FOR SETTING
                        1238 *    MEMORY VALUES.
                        1239 *
C576 CD 59 C3           1240 SET      CALL    SBLK    SCAN TO SECONDARY COMMAND
C579 CA 6B C4           1241          JZ      ERR1    MUST HAVE AT LEAST SOMETHING!!
C57C D5                 1242          PUSH    D       SAVE SCAN ADDRESS
C57D CD 78 C3           1243          CALL    SCONV   CONVERT FOLLOWING VALUE
C580 E3                 1244          XTHL    .       HL=SAVED SCAN ADDR AND STACK=VALUE
C581 11 11 C3           1245          LXI     D,SETAB  SECONDARY COMMAND TABLE
C584 CD 91 C2           1246          CALL    FDCOM   TRY TO LOCATE IT
C587 C3 82 C2           1247          JMP     DISPO   OFF TO IT OR ERROR IF NOT IN TBL
                        1248 *
                        1249 *
                        1250 *    THIS ROUTINE SETS THE TAPE SPEED
                        1251 *
      C58A              1252 TASPD    EQU     $       GET CONVERTED VALUE
C58A B7                 1253          ORA     A       IS IT ZERO?
C58B CA 90 C5           1254          JZ      SETSP   YES--THAT IS A PROPER SPEED
C58E 3E 20              1255          MVI     A,32    NO--SET SPEED PROPERLY THEN
C590 32 0D C8           1256 SETSP    STA     TSPD
C593 C9                 1257          RET
                        1258 *
                        1259 *
      C59A              1260 STSPD    EQU     $       VDM ESCAPE SEQUENCE COMES HERE
C59A 78                 1261          MOV     A,B     GET CHAR FOR FOLLOWING DISPD
      C595              1262 DISPD    EQU     $       SET DISPLAY SPEED
C595 32 0B C8           1263          STA     SPEED
C598 C9                 1264          RET
                        1265 *
                        1266 *
      C599              1267 SETIN    EQU     $       SET AN INPUT PSUEDO PORT
C599 32 06 C8           1268          STA     IPORT
C59C C9                 1269          RET
                        1270 *
                        1271 *
      C59D              1272 SETOT    EQU     $       SET AN OUTPUT PSUEDO PORT
C59D 32 07 C8           1273          STA     OPORT
C5A0 C9                 1274          RET
                        1275 *
                        1276 *
      C5A1              1277 SETUI    EQU     $       DEFINE USER INPUT RTN ADDR
C5A1 22 00 C8           1278          SHLD    UIPRT
C5A4 C9                 1279          RET
                        1280 *
                        1281 *
      C5A5              1282 SETCO    EQU     $       DEFINE USER OUTPUT RTN ADDR
C5A5 22 02 C8           1283          SHLD    UOPRT
C5A8 C9                 1284          RET
                        1285 *
                        1286 *
      C5A9              1287 SETTY    EQU     $       SET TAPE HDR TYPE
C5A9 32 22 C8           1288          STA     WTYPE
C5AC C9                 1289          RET
                        1290 *
                        1291 *
      C5AD              1292 SETXQ    EQU     $       SET TAPE EXECUTE ADDDR FOR HDR
C5AD 22 27 C8           1293          SHLD    XEQAD
C5B0 C9                 1294          RET
                        1295 *
                        1296 *
      C5B1              1297 SETNU    EQU     $       HERE TO SET NUMBER OF NULLS
C5B1 32 10 C8           1298          STA     NUCNT   THIS IS IT
C5B4 C9                 1299          RET     .
                        1300 *
                        1301 *
      C5B5              1302 SETCR    EQU     $       SET CRC TO BE NORMAL, OR IGNORE CRC ERRORS
C5B5 32 11 C8           1303          STA     IGNCR   FF=IGNORE CRC ERRORS, ELSE=NORMAL
C5B8 C9                 1304          RET     .
                        1305 *
                        1306 *
      C5B9              1307 CUSET    EQU     $       TRY TO SET/CLEAR CUSTOM ROUTINE ADDR
C5B9 CD 51 C4           1308          CALL    NAMEO   GET A NAME (S/B 2 CHARS OR MORE)
C5BC 21 18 C2           1309          LXI     H,COMND  PT HERE IN CASE ADDR NOT GIVEN
C5BF CD A5 C3           1310          CALL    PSCAN   GET OPTIONAL OPERAND IF ANY
C5C2 E5                 1311          PUSH    H       SAVE THAT VALUE (IF ANY)
C5C3 21 1C C8           1312          LXI     H,THEAD  PT TO NAME
C5C6 CD 8E C2           1313          CALL    FDCOU   SEE IF NAME IS KNOWN IN CUST TABLE
C5C9 CA CF C5           1314          JZ      CUSE2   NO--PROCEED TO KNOW IT
C5CC 1B                 1315          DCX     D       DE PT TO 1ST CHAR OF NAME IN TBL
C5CD 36 00              1316          MVI     M,0  (HL CAME BACK PT'ING TO THEAD)  CLR THIS NAME
      C5CF              1317 CUSE2    EQU     $    ENTER NEW ONE IN TBL
C5CF 7E                 1318          MOV     A,M  GET 1ST CHAR OF NAME
C5D0 12                 1319          STAX    D    PUT NAME INTO TABLE
C5D1 13                 1320          INX     D
C5D2 23                 1321          INX     H
C5D3 7E                 1322          MOV     A,M     GET 2ND CHAR OF NAME
C5D4 12                 1323          STAX    D       NAME IS NOW POSTED
C5D5 13                 1324          INX     D       PT TO 1ST BYTE OF ADDR
C5D6 E1                 1325          POP     H       RESTORE SAVED RTN ADDR
C5D7 EB                 1326          XCHG    .       DE=RTN ADDR, HL=THIS CU ENTRY
C5D8 73                 1327          MOV     M,E     LO BYTE
C5D9 23                 1328          INX     H
C5DA 72                 1329          MOV     M,D     AND HI BYTE
C5DB C9                 1330          RET     .       ALL DONE
                        1331 *
                        1332 *
                        1333 * -0-
9999                    COPY    CUTER3/1                                3 OF 3 000000
                        1334 *
                        1335 *
                        1336 *
                        1337 *
                        1338 *   THE FOLLOWING ROUTINES PROVIDE "BYTE BY BYTE" ACCESS
                        1339 *  TO THE CASSETTE TAPES ON EITHER A READ OR WRITE BASIS.
                        1340 *
                        1341 *  THE TAPE IS READ ONE BLOCK AT A TIME AND INDIVIDUAL
                        1342 *  TRANSFERS OF DATA HANDLED BY MANAGING A BUFFER AREA.
                        1343 *
                        1344 *  THE BUFFER AREA IS CONTROLLED BY A FILE CONTROL BLOCK
                        1345 *  (FCB) WHOSE STRUCTURE IS:
                        1346 *
                        1347 *
                        1348 *     7 BYTES FOR EACH OF THE TWO FILES STRUCTURED AS
                        1349 *   FOLLOWS:
                        1350 *
                        1351 *          1 BYTE -  ACCESS CONTROL   00  IF CLOSED
                        1352 *                                     FF  IF READING
```

```
                        *                              FE  IF WRITING
                        *          1 BYTE -  READ COUNTER
                        *          1 BYTE -  BUFFER POSITION POINTER
                        *          2 BYTE -  CONTROL HEADER ADDRESS
                        *          2 BYTE -  BUFFER LOCATION ADDRESS
                        *
                        *
                        *
                        *        THIS ROUTINE "OPENS" THE CASSETTE UNIT FOR ACCESS
                        *
                        *   ON ENTRY:  A - HAS THE TAPE UNIT NUMBER (1 OR 2)
                        *              HL - HAS USER SUPPLIED HEADER FOR TAPE FILE
                        *
                        *
                        *   NORMAL RETURN:    ALL REGISTERS ARE ALTERED
                        *                     BLOCK IS READY FOR ACCESS
                        *
                        *   ERROR RETURN:     CARRY BIT IS SET
                        *
                        *   ERRORS:  BLOCK ALREADY OPEN
                        *
C5DC E5                 BOPEN  PUSH    H       SAVE HEADER ADDRESS
C5DD CD 2F C6           CALL   LFCB    GET ADDRESS OF FILE CONTROL
C5E0 C2 F6 C5           JNZ    TERE2   FILE WAS ALREADY OPEN
C5E3 36 01              MVI    M,1     NOW IT IS
C5E5 23                 INX    H       POINT TO READ COUNT
C5E6 77                 MOV    M,A     ZERO
C5E7 23                 INX    H       POINT TO BUFFER CURSOR
C5E8 77                 MOV    M,A     PUT IN THE ZERO COUNT
                        *
                        *  ALLOCATE THE BUFFER
                        *
C5E9 11 63 C8           LXI    D,FBUF1  POINT TO BUFFER AREA
C5EC 3A 54 C8           LDA    FNUMF   GET WHICH ONE WE ARE GOING TO USE
C5EF 82                 ADD    D
C5F0 57                 MOV    D,A     256 BIT ADD
                        *
C5F1 C1                 UBUF   POP    B       HEADER ADDRESS
C5F2 B7                 ORA    A       CLEAR CARRY AND RETURN AFTER STORING PARAMS
C5F3 C3 B2 C6           JMP    PSTOR   STORE THE VALUES
                        *
                        *    GENERAL ERROR RETURN POINTS FOR STACK CONTROL
                        *
C5F6 E1                 TERE2  POP    H
C5F7 D1                 TERE1  POP    D
C5F8 AF                 TERE0  XRA    A       CLEAR ALL FLAGS
C5F9 37                        STC    .       SET ERROR
C5FA C9                        RET
                        *
C5FB 3D                 EOFER  DCR    A       SET MINUS FLAGS
C5FC 37                        STC    .       AND CARRY
C5FD D1                        POP    D       CLEAR THE STACK
C5FE C9                        RET    .       THE FLAGS TELL ALL
                        *
                        *
                        *
                        *   THIS ROUTINE CLOSES THE FILE BUFFER TO ALLOW ACCESS
                        *   FOR A DIFFERENT CASSETTE OR PROGRAM.  IF THE FILE
                        *   OPERATIONS WERE "WRITE" THEN THE LAST BLOCK IS WRITTEN
                        *   OUT AND AN "END OF FILE" WRITTEN TO THE TAPE.  IF
                        *   THE OPERATIONS WERE "READS" THEN THE FILE IS JUST
                        *   MADE READY FOR NEW USE.
                        *
                        *   ON ENTRY:  A - HAS WHICH UNIT (1 OR 2)
                        *
                        *   ERROR RETURNS:  FILE WASN'T OPEN
                        *
C5FF C[illegible]  C6   PCLOS  CALL   LFCB    GET CONTROL BLOCK ADDRESS
C602 C8                        RZ     .       WASN'T OPEN, CARRY IS SET FROM LFCB
C603 B7                        ORA    A       CLEAR CARRY
C604 3C                        INR    A       SET CONDITION FLAGS
C605 36 00              EOFW   MVI    M,0     CLOSE THE CONTROL BYTE
C607 C8                        RZ     .       WE WERE READING...NOTHING MORE TO DO
                        *
                        *    THE FILE OPERATIONS WERE "WRITES"
                        *
                        *   PUT THE CURRENT BLOCK ON THE TAPE
                        *   (EVEN IF ONLY ONE BYTE!!)
                        *   THEN WRITE AN END OF FILE TO THE TAPE
                        *
C608 23                        INX    H
C609 23                        INX    H
C60A 7E                        MOV    A,M     GET CURSOR POSITION
C60B 7E                        MOV    A,M
C60C CD BB C6           CALL   PLOAD  BC GET HEADER ADDRESS, DE BUFFER ADDRESS
C60F C5                 PUSH   B       HEADER TO STACK
C610 21 07 00           LXI    H,BLKOF  OFFSET TO BLOCK SIZE
C613 09                 DAD    B
C614 B7                 ORA    A       TEST COUNT
C615 CA 27 C6           JZ     EOFW    NO BYTES...JUST WRITE EOF
                        *
                        *    WRITE LAST BLOCK
                        *
C618 E5                 PUSH   H       SAVE BLOCK SIZE POINTER FOR EOF
C619 77                 MOV    M,A     PUT IN COUNT
C61A 23                 INX    H
C61B 36 00              MVI    M,0     ZERO THE HIGHER BYTE
C61D 23                 INX    H
C61E 73                 MOV    M,E     BUFFER ADDRESS
C61F 23                 INX    H
C620 72                 MOV    M,D
C621 60                 MOV    H,B
C622 69                 MOV    L,C     PUT HEADER ADDRESS IN HL
C623 CD 78 C7           CALL   WFBLK   GO WRITE IT OUT
C626 E1                 POP    H       BLOCK SIZE POINTER
                        *
                        *   NOW WRITE END OF FILE TO CASSETTE
                        *
C627 AF                 EOFW   XRA    A     PUT IN ZEROS FOR SIZE:  EOF MARK IS ZERO BYTES
C628 77                        MOV    M,A
C629 23                        INX    H
C62A 77                        MOV    M,A
C62B E1                        POP    H       HEADER ADDRESS
C62C C3 78 C7           JMP    WFBLK   WRITE IT OUT AND RETURN
                        *
                        *
                        *
                        *   THIS ROUTINE LOCATES THE FILE CONTROL BLOCK POINTED TO
                        *   BY REGISTER "A".  ON RETURN HL POINT TO THE CONTROL BYT
                        *   AND REGISTER "A" HAS THE CONTROL WORD WITH THE FLAGS
                        *   SET FOR IMMEDIATE CONDITION DECISIONS.
                        *
C62F            8       LFCB   LXI    H,FCBAS  POINT TO THE BASE OF IT
C632                           RAR    .       MOVE THE 1 & 2 TO 0 & 1 LIKE COMPUTERS LIKE
C633 E6 01                     ANI    1       SMALL NUMBERS ARE THE RULE
C635 32 54 C8                  STA    FNUMF   CURRENT ACCESS FILE NUMBER
C638 CA 3E C6                  JZ     LFCB1   UNIT ONE (VALUE OF ZERO)
C63B 21 5C C8                  LXI    H,FCBA2  UNIT TWO--PT TO ITS FCB
C63E                    LFCB1  EQU    $       HL PT TO PROPER FCB

C642 CD 2F C6                  CALL   LFCB    LOCATE THE FILE CONTROL BLOCK
C645 C8                        RZ     .       FILE NOT OPEN
C646 3C                        INR    A       TEST IF FF
C647 FA F8 C5                  JM     TERE0   ERROR WAS WRITING
C64A 36 FF                     MVI    M,-1    SET IT AS READ  (IN CASE IT WAS JUST OPENED)
C64C 23                        INX    H
C64D 7E                        MOV    A,M     GET READ COUNT
C64E E5                        PUSH   H       SAVE COUNT ADDRESS
C64F 23                        INX    H
C650 CD BB C6           CALL   PLOAD  GET THE OTHER PARAMETERS
C653 E1                 POP    H
C654 B7                 ORA    A
C655 C2 71 C6           JNZ    GTBYT   IF NOT EMPTY GO GET BYTE
                        *
                        *  CURSOR POSITION WAS ZERO...READ A NEW BLOCK INTO
                        *  THE BUFFER.
                        *
C658 D5                 RDNBLK PUSH   D       BUFFER POINTER
C659 E5                        PUSH   H       TABLE ADDRESS
C65A 23                        INX    H
C65B CD A2 C6           CALL   PHEAD  PREPARE THE HEADER FOR READ
C65E CD C4 C6           CALL   RFBLK  READ IN THE BLOCK
C661 DA F6 C5           JC     TERE2  ERROR POP OFF STACK BEFORE RETURN
C664 E1                 POP    H
C665 7B                 MOV    A,E     LOW BYTE OF COUNT (WILL BE ZERO IF 256)
C666 B2                 ORA    D       SEE IF BOTH ARE ZERO
C667 CA FB C5           JZ     EOFER   BYTE COUNT WAS ZERO....END OF FILE
C66A 73                 MOV    M,E     NEW COUNT ( ZERO IS 256 AT THIS POINT)
C66B 23                 INX    H       BUFFER LOCATION POINTER
C66C 36 00              MVI    M,0
C66E 2B                 DCX    H
C66F 7B                 MOV    A,E     COUNT TO A
C670 D1                 POP    D       GET BACK BUFFER ADDRESS
                        *
                        *
                        *
                        *   THIS ROUTINE GETS ONE BYTE FROM THE BUFFER
                        *  AND RETURNS IT IN REGISTER "A".  IF THE END
                        *  OF THE BUFFER IS REACHED IT MOVES THE POINTER
                        *  TO THE BEGINNING OF THE BUFFER FOR THE NEXT
                        *  LOAD.
                        *
C671 3D                 GTBYT  DCR    A       BUMP THE COUNT
C672 77                        MOV    M,A     RESTORE IT
C673 23                        INX    H
C674 7E                        MOV    A,M     GET BUFFER POSITION
C675 34                        INR    M       BUMP IT
                        *
C676 83                        ADD    E
C677 5F                        MOV    E,A .   DE NOW POINT TO CORRECT BUFFER POSITION
C678 D2 7C C6           JNC    RT1
C67B 14                 INR    D
C67C 1A                 RT1    LDAX   D       GET CHARACTER FROM BUFFER
C67D B7                 ORA    A       CLEAR CARRY
C67E C9                 RET    .       ALL DONE
                        *
                        *
                        *
                        *       THIS ROUTINE IS USED TO WRITE A BYTE TO THE FILE
                        *
                        *       ON ENTRY:   A -  HAS FILE NUMBER
                        *                   B -  HAS DATA BYTE
                        *
C67F CD 2F C6           WTBYT  CALL   LFCB    GET CONTROL BLOCK
C682 C8                        RZ     .       FILE WASN'T OPEN
C683 3C                        INR    A
C684 C8                        RZ     .       FILE WAS READ
C685 36 FE                     MVI    M,OFEH  SET IT TO WRITE
C687 23                        INX    H
C688 23                        INX    H
C689 78                 MOV    A,B     GET CHARACTER
C68A F5                 PUSH   PSW
C68B E5                 PUSH   H       SAVE CONTROL ADDRESS+2
                        *
                        *   NOW DO THE WRITE
                        *
C68C CD BB C6           CALL   PLOAD  BC GETS HEADER ADDR, DE BUFFER ADDRESS
C68F E1                 POP    H
C690 7E                 MOV    A,M     COUNT BYTE
C691 83                 ADD    E
C692 5F                 MOV    E,A
C693 D2 97 C6           JNC    WT1
C696 14                 INR    D
C697 F1                 WT1    POP    PSW     CHARACTER
C698 12                 STAX   D       PUT CHR IN BUFFER
C699 B7                 ORA    A       CLEAR FLAGS
C69A 34                 INR    M       INCREMENT THE COUNT
C69B C0                 RNZ    .       RETURN IF COUNT DIDN'T ROLL OVER
                        *
                        *   THE BUFFER IS FULL.  WRITE IT TO TAPE AND RESET
                        *  CONTROL BLOCK.
                        *
C69C CD A2 C6           CALL   PHEAD  PREPARE THE HEADER
C69F C3 78 C7           JMP    WFBLK  WRITE IT OUT AND RETURN
                        *
                        *
                        *
                        *   THIS ROUTINE PUTS THE BLOCK SIZE (256) AND BUFFER
                        *  ADDRESS IN THE FILE HEADER.
                        *
C6A2 CD BB C6           PHEAD  CALL   PLOAD  GET HEADER AND BUFFER ADDRESSES
C6A5 C5                 PUSH   B       HEADER ADDRESS
C6A6 21 06 00           LXI    H,BLKOF-1  PSTOR DOES AN INCREMENT
C6A9 09                 DAD    B       HL POINT TO BLOCKSIZE ENTRY
C6AA 01 00 01           LXI    B,256
C6AD CD B2 C6           CALL   PSTOR
C6B0 E1                 POP    H       HL RETURN WITH HEADER ADDRESS
C6B1 C9                 RET
                        *
C6B2 23                 PSTOR  INX    H
C6B3 71                 MOV    M,C
C6B4 23                 INX    H
```

```
C6BE 70                 1625            MOV     M,B
C6BA 23                 1626            INX     H
C6BB 73                 1627            MOV     M,E
C6BE 23                 1628            INX     H
C6BF 72                 1629            MOV     M,D
        C9              1630            RET
                        1631 *
                        1632 *
C6BE 23                 1633 PLOAD      INX     H
C6BF 4E                 1634            MOV     C,M
C6C0 23                 1635            INX     H
C6C1 46                 1636            MOV     B,M
C6C2 23                 1637            INX     H
C6C3 5E                 1638            MOV     E,M
C6C4 23                 1639            INX     H
C6C5 56                 1640            MOV     D,M
C6C6 C9                 1641            RET
                        1642 *
                        1643 *
                        1644 *
                        1645 *
                        1646 *
                        1647 *   THIS ROUTINE SETS THE CORRECT UNIT FOR SYSTEM READS
C6C7 CD DA C7           1648 RFBLK  CALL    GTUNT   SET UP A=UNIT WITH SPEED
                        1649 *
                        1650 *
                        1651 *
                        1652 *
                        1653 *               TAPE READ ROUTINES
                        1654 *
                        1655 *     ON ENTRY:       A   HAS UNIT AND SPEED
                        1656 *                     HL  POINT TO HEADER BLOCK
                        1657 *                     DE  HAVE OPTIONAL PUT ADDRESS
                        1658 *
                        1659 *     ON EXIT:        CARRY IS SET IF ERROR OCCURED
                        1660 *                     TAPE UNITS ARE OFF
                        1661 *
                        1662 *
C6CA D5                 1663 RTAPE  PUSH    D       SAVE OPTIONAL ADDRESS
C6CB 06 03              1664        MVI     B,3     SHORT DELAY
C6CD CD EB C7           1665        CALL    TON
C6D0 DB FB              1666        IN      TDATA   CLEAR THE UART FLAGS
                        1667 *
C6D2 E5                 1668 RTAP1  PUSH    H       HEADER ADDRESS
C6D3 CD 1F C7           1669        CALL    RHEAD   GO READ HEADER
C6D6 E1                 1670        POP     H
C6D7 DA 02 C7           1671        JC      TERR    IF AN ERROR OR ESC WAS RECEIVED
C6DA C2 D2 C6           1672        JNZ     RTAP1   IF VALID HEADER NOT FOUND
                        1673 *
                        1674 *  FOUND A VALID HEADER NOW DO COMPARE
                        1675 *
C6DD E5                 1676        PUSH    H       GET BACK AND RESAVE ADDRESS
C6DE 11 1C C8           1677        LXI     D,THEAD
C6E1 CD CE C7           1678        CALL    BHCMP   COMPARE DE-HL HEADERS
C6E4 E1                 1679        POP     H
C6E5 C2 D2 C6           1680        JNZ     RTAP1
                        1681 *
                        1682 *
C6E8 D1                 1683        POP     D       OPTIONAL "PUT" ADDRESS
C6E9 7A                 1684        MOV     A,D
C6EA B3                 1685        ORA     E       SEE IF DE IS ZERO
C6EB 2A 23 C8           1686        LHLD    BLOCK   GET BLOCK SIZE
C6EE EB                 1687        XCHG    .       ...TO DE
                        1688 *  DE WAS NBLOCK....HL WAS USER OPTION
C6EF C2 F2 C6           1689        JNZ     RTAP    IF DE WAS ZERO GET TAPE LOAD ADDRESS
C6F2 2A 25 C8           1690        LHLD    LOADR   GET TAPE LOAD ADDRESS
                        1691 *
                        1692 *
                        1693 *     THIS ROUTINE READS "DE" BYTES FROM THE TAPE
                        1694 *     TO ADDRESS HL.  THE BYTES MUST BE FROM ONE
                        1695 *     CONTIGUOUS PHYSICAL BLOCK ON THE TAPE.
                        1696 *
                        1697 *          HL HAS "PUT" ADDRESS
                        1698 *          DE HAS SIZE OF TAPE BLOCK
                        1699 *
C6F5 D5                 1700 RTAP   PUSH    D       SAVE SIZE FOR RETURN TO CALLING PROGRAM
                        1701 *
        C6F6            1702 RTAP2  EQU     $       HERE TO LOOP RDING BLKS
C6F6 CD 11 C7           1703        CALL    DCRCT   DROP COUNT, B=LEN THIS BLK
C6F9 CA 0C C7           1704        JZ      RTOFF   ZERO=ALL DONE
                        1705 *
C6FC CD A0 C7           1706        CALL    RHED1   READ THAT MANY BYTES
C6FF DA 02 C7           1707        JC      TERR    IF ERROR OR ESC
C702 CA F6 C6           1708        JZ      RTAP2   RD OK--READ SOME MORE
                        1709 *
                        1710 *  ERROR RETURN
                        1711 *
C705 AF                 1712 TERR   XRA     A
C706 37                 1713        STC     .       SET ERROR FLAGS
C707 C3 0D C7           1714        JMP     RTOF1
                        1715 *
                        1716 *
C70A 06 01              1717 TOFF   MVI     B,1
C70C CD E0 C7           1718        CALL    DELAY
C70F AF                 1719 RTOFF  XRA     A
C710 D3 FA              1720 RTOF1  OUT     TAPPT
C712 D1                 1721        POP     D       RETURN BYTE COUNT
C713 C9                 1722        RET
                        1723 *
                        1724 *
        C714            1725 DCRCT  EQU     $       COMMON RTN TO COUNT DOWN BLK LENGTHS
C714 AF                 1726        XRA     A       CLR FOR LATER TESTS
C715 47                 1727        MOV     B,A     SET THIS BLK LEN=256
C716 B2                 1728        ORA     D       IS AMNT LEFT < 256
C717 C2 1D C7           1729        JNZ     DCRC2   NO--REDUCE AMNT BY 256
C71A B3                 1730        ORA     E       IS ENTIRE COUNT ZERO
C71B C8                 1731        RZ      ALL DONE--ZERO=THIS CONDITION
C71C 43                 1732        MOV     B,E     SET THIS BLK LEN TO AMNT REMAINING
C71D 5A                 1733        MOV     E,D     MAKE ENTIRE COUNT ZERO NOW
C71E C9                 1734        RET     .       ALL DONE (NON-ZERO FLAG)
        C71F            1735 DCRC2  EQU     $       REDUCE COUNT BY 256
C71F 15                 1736        DCR     D       DROP BY 256
C720 B7                 1737        ORA     A       FORCE NON-ZERO FLAG
C721 C9                 1738        RET     .       NON-ZERO=NOT DONE YET (BLK LEN=256)
                        1739 *
                        1740 *
                        1741 *   READ THE HEADER
                        1742 *
C722 06 0A              1743 RHEAD  MVI     B,10    FIND 10 NULLS
C724 CD 59 C7           1744 RHEA1  CALL    STAT
C727 D8                 1745        RC      .       IF ESCAPE
C728 DB FB              1746        IN      TDATA   IGNORE ERROR CONDITIONS
C72A B7                 1747        ORA     A       ZERO?
C72B C2 1F C7           1748        JNZ     RHEAD
C72E 05                 1749        DCR     B
C72F C2 24 C7           1750        JNZ     RHEA1   LOOP UNTIL 10 IN A ROW
                        1751 *
                        1752 *    WAIT FOR THE START CHARACTER
                        1753 *
C732 CD 6B C7           1754 SOHL   CALL    TAPIN
C735 D8                 1755        RC      .       ERROR OR ESCAPE
C736 FE 01              1756        CPI     1       ARE WE AT THE 01 YET (START CHAR)
C738 DA 2F C7           1757        JC      SOHL    NO, BUT STIL ZEROES
C73B C2 1F C7           1758        JNZ     RHEAD   NO, LOOK FOR ANOTHER 10 NULLS
                        1759 *
                        1760 *    WE HAVE 10 (OR MORE) NULLS FOLLOWED IMMEDIATELY
                        1761 *    BY AN 01.  NOW READ THE HEADER.
C73B 21 1C C8           1762 *
C73E 06 10              1763        LXI     H,THEAD  POINT TO BUFFER
                        1764        MVI     B,HLEN  LENGTH TO READ
        C740            1765 *
C740 0E 00              1766 RHED1  EQU     $       RD A BLOCK INTO HL FOR B BYTES
        C742            1767        MVI     C,0     INIT THE CRC
C742 CD 6B C7           1768 RHED2  EQU     $       LOOP HERE
C745 D8                 1769        CALL    TAPIN   GET A BYTE
C746 77                 1770        RC
C747 23                 1771        MOV     M,A     STORE IT
C748 CD A4 C7           1772        INX     H       INCREMENT ADDRESS
C74B 05                 1773        CALL    DOCRC   GO COMPUTE THE CRC
C74C C2 42 C7           1774        DCR     B       WHOLE HEADER YET?
                        1775        JNZ     RHED2   DO ALL THE BYTES
                        1776 *
                        1777 *   THIS ROUTINE GETS THE NEXT BYTE AND COMPARES IT
                        1778 *  TO THE VALUE IN REGISTER C.  THE FLAGS ARE SET ON
                        1779 *  RETURN.
                        1780 *
C74F CD 6B C7           1781        CALL    TAPIN   GET CRC BYTE
C752 A9                 1782        XRA     C       CLR CARRY AND SET ZERO IF MATCH, ELSE NON-ZERO
C753 C8                 1783        RZ      .       CRC IS FINE
C754 3A 11 C8           1784        LDA     IGNCR   BAD CRC, SHD WE STILL ACCEPT IT
C757 3C                 1785        INR     A       SEE IF IT WAS FF, IF FF THEN ZERO SAYS IGN ERR
                        1786 *   NOW, CRC ERR DETECTION DEPENDS ON IGNCR.
C758 C9                 1787        RET
                        1788 *
                        1789 *    THIS ROUTINE GETS THE NEXT AVAILABLE BYTE FROM THE
                        1790 *  TAPE.  WHILE WAITING FOR THE BYTE THE KEYBOARD IS TESTED
                        1791 *  FOR AN ESC COMMAND.  IF RECEIVED THE TAPE LOAD IS
                        1792 *  TERMINATED AND A RETURN TO THE COMMAND MODE IS MADE.
                        1793 *
C759 DB FA              1794 STAT   IN      TAPPT   TAPE STATUS PORT
C75B E6 40              1795        ANI     TDR
C75D C0                 1796        RNZ
C75E CD 1F C0           1797        CALL    SINP    CHECK INPUT
C761 CA 59 C7           1798        JZ      STAT    NOTHING THERE YET
C764 E6 7F              1799        ANI     7FH     CLEAR PARITY 1ST
C766 C2 59 C7           1800        JNZ     STAT    EITHER MODE OR CTL-@
C769 37                 1801        STC     .       SET ERROR FLAG
C76A C9                 1802        RET     .       AND RETURN
                        1803 *
                        1804 *
                        1805 *
C76B CD 59 C7           1806 TAPIN  CALL    STAT    WAIT UNTIL A CHARACTER IS AVAILABLE
C76E D8                 1807        RC
                        1808 *
C76F DB FA              1809 TREDY  IN      TAPPT   TAPE STATUS
C771 E6 18              1810        ANI     TPE+TOE  DATA ERROR?
C773 DB FB              1811        IN      TDATA   GET THE DATA
C775 C8                 1812        RZ      .       IF NO ERRORS
C776 37                 1813        STC     .       SET ERROR FLAG
C777 C9                 1814        RET
                        1815 *
                        1816 *
                        1817 *  THIS ROUTINE GETS THE CORRECT UNIT FOR SYSTEM WRITES
C778 CD DA C7           1818 WFBLK  CALL    GTUNT   SET UP A WITH UNIT AND SPEED
                        1819 *
                        1820 *
                        1821 *
                        1822 *       WRITE TAPE BLOCK ROUTINE
                        1823 *
                        1824 *   ON ENTRY:   A   HAS UNIT AND SPEED
                        1825 *               HL  HAS POINTER TO HEADER
                        1826 *
                        1827 *
        C77B            1828 WTAPE  EQU     $       HERE TO WRITE TAPE
C77B E5                 1829        PUSH    H       SAVE HEADER ADDRESS
C77C CD AB C7           1830        CALL    WHEAD   TURN ON, THEN WRITE HDR
C77F E1                 1831        POP     H
C780 11 07 00           1832        LXI     D,BLKOF  OFFSET TO BLOCK SIZE IN HEADER
C783 19                 1833        DAD     D       HL POINT TO BLOCK SIZE
C784 5E                 1834        MOV     E,M
C785 23                 1835        INX     H
C786 56                 1836        MOV     D,M     DE HAVE SIZE
C787 23                 1837        INX     H
C788 7E                 1838        MOV     A,M
C789 23                 1839        INX     H
C78A 66                 1840        MOV     H,M
C78B 6F                 1841        MOV     L,A     HL HAVE STARTING ADDRESS
                        1842 *
                        1843 *    THIS ROUTINE WRITES ONE PHYSICAL BLOCK ON THE
                        1844 *  TAPE "DE" BYTES LONG FROM ADDRESS "HL".
                        1845 *
                        1846 *
        C78C            1847 WTAP1  EQU     $       HERE FOR THE EXTRA PUSH
C78C E5                 1848        PUSH    H       A DUMMY PUSH FOR LATER EXIT
        C78D            1849 WTAP2  EQU     $       LOOP HERE UNTIL ENTIRE AMOUNT READ
C78D CD 11 C7           1850        CALL    DCRCT   DROP COUNT IN DE AND SET UP B W/LEN THIS BLK
C790 CA 07 C7           1851        JZ      TOFF    RETURNS ZERO IF ALL DONE
C793 CD BF C7           1852        CALL    WTBL    WRITE BLOCK FOR BYTES IN B (256)
C796 C3 8D C7           1853        JMP     WTAP2   LOOP UNTIL ALL DONE
                        1854 *
                        1855 *
C799 F5                 1856 WRTAP  PUSH    PSW
C79A DB FA              1857 WRWAT  IN      TAPPT   TAPE STATUS
C79C E6 80              1858        ANI     TTBE    IS TAPE READY FOR A CHAR YET
C79E CA 9A C7           1859        JZ      WRWAT   NO--WAIT
C7A1 F1                 1860        POP     PSW     YES--RESTORE CHAR TO OUTPUT
C7A2 D3 FB              1861        OUT     TDATA   SEND CHAR TO TAPE
                        1862 *
        C7A4            1863 DOCRC  EQU     $       A COMMON CRC COMPUTATION ROUTINE
C7A4 91                 1864        SUB     C
C7A5 4F                 1865        MOV     C,A
C7A6 A9                 1866        XRA     C
C7A7 2F                 1867        CMA
C7A8 91                 1868        SUB     C
C7A9 4F                 1869        MOV     C,A
C7AA C9                 1870        RET     .       ONE  BYTE NOW WRITTEN
                        1871 *
                        1872 *
                        1873 *   THIS ROUTINE WRITES THE HEADER POINTED TO BY
                        1874 *   HL TO THE TAPE.
                        1875 *
        C7AB            1876 WHEAD  EQU     $       HERE TO 1ST TURN ON THE TAPE
C7AB CD E9 C7           1877        CALL    WTON    TURN IT ON, THEN WRITE HEADER
C7AE 16 32              1878        MVI     D,50    WRITE 50 ZEROS
C7B0 AF                 1879 NULOP  XRA     A
C7B1 CD 99 C7           1880        CALL    WRTAP
C7B4 15                 1881        DCR     D
C7B5 C2 B0 C7           1882        JNZ     NULOP
                        1883 *
C7B8 3E 01              1884        MVI     A,1
C7BA CD 99 C7           1885        CALL    WRTAP
C7BD 06 10              1886        MVI     B,HLEN  LENGTH TO WRITE OUT
                        1887 *
C7BF 0E 00              1888 WTBL   MVI     C,0     RESET CRC BYTE
C7C1 7E                 1889 WLOOP  MOV     A,M     GET CHARACTER
C7C2 CD 99 C7           1890        CALL    WRTAP   WRITE IT TO THE TAPE
C7C5 05                 1891        DCR     B
C7C6 23                 1892        INX     H
C7C7 C2 C1 C7           1893        JNZ     WLOOP
C7CA 79                 1894        MOV     A,C     GET CRC
C7CB C3 99 C7           1895        JMP     WRTAP   PUT IT ON THE TAPE AND RETURN
                        1896 *
                        1897 *
                        1898 *   THIS ROUTINE COMPARES THE HEADER IN THEAD TO
```

```
                        *   THE USER SUPPLIED HEADER IN ADDRESS HL.
                        *   ON RETURN IF ZERO IS SET THE TWO NAMES COMPARED
                        *
C7CE                    DHCMP   MVI     B,5
C7D0                    DHLOP   LDAX    D
C7D1                            CMP     M
C7D2                            RNZ
C7D3                            DCR     B
C7D4                            RZ      .       IF ALL FIVE COMPARED
C7D5                            INX     H
C7D6 13                         INX     D
C7D7 C3 D0 C7                   JMP     DHLOP
                        *
     C7DA               GTUNT   EQU     $       SET A=SPEED + UNIT
C7DA 3A 54 C8                   LDA     FNUMF   GET UNIT
C7DD B7                         ORA     A       SEE WHICH UNIT
C7DE 3A 0D C8                   LDA     TSPD    BUT 1ST GET SPEED
C7E1 C2 E6 C7                   JNZ     GTUN2   MAKE IT UNIT TWO
C7E4 C6 A0                      ADI     TAPE2   THIS ONCE=UNIT 2, TWICE=UNIT 1
C7E6 C6 A0              GTUN2   ADI     TAPE2   UNIT AND SPEED NOW SET IN A
C7E8 C9                         RET     .       ALL DONE
                        *
C7E9 06 04              WTON    MVI     B,4     SET LOOP DELAY  (BIT LONGER ON A WRITE)
     C7EB               TON     EQU     $       HERE TO TURN A TAPE ON THEN DELAY
C7EB D3 FA                      OUT     TAPPT   GET TAPE MOVING, THEN DELAY
                        *
C7ED 11 00 00           DELAY   LXI     D,0
C7F0 1B                 DLOP1   DCX     D
C7F1 7A                         MOV     A,D
C7F2 B3                         ORA     E
C7F3 C2 F0 C7                   JNZ     DLOP1
C7F6 05                         DCR     B
C7F7 C2 ED C7                   JNZ     DELAY
C7FA C9                         RET
                        *
                        *
                        ***** -- END OF PROGRAM--
                        *
                        *
                        *
                        *    S Y S T E M    E Q U A T E S
                        *
                        *
                        *           VDM PARAMETERS
                        *
     CC00               VDMEM   EQU     0CC00H  VDM SCREEN MEMORY
                        *
                        *
                        *           KEYBOARD SPECIAL KEY ASSIGNMENTS
                        *
                        *  THESE DEFINITIONS ARE DESIGNED TO ALLOW
                        *  COMPATABILITY WITH SOLOS(TM).  THESE ARE THE
                        *  SAME KEYS WITH BIT 7 (X'80') STRIPPED OFF.
                        *
     001A               DOWN    EQU     1AH     CTL Z
     0017               UP      EQU     17H     CTL W
     0001               LEFT    EQU     01H     CTL A
     0013               RIGHT   EQU     13H     CTL S
     000B               CLEAR   EQU     0BH     CTL K
     000E               HOME    EQU     0EH     CTL N
     0000               MODE    EQU     00H     CTL-@
     005F               BACKS   EQU     5FH     BACKSPACE
     000A               LF      EQU     10
                        CR      EQU     13
                        BLANK   EQU     ' '
                        SPACE   EQU     BLANK
                        CX      EQU     'X'-40H
                        ESC     EQU     1BH
                        *
                        *           PORT ASSIGNMENTS
                        *
     0000               STAPT   EQU     0       STATUS PORT GENERAL
     0001               SDATA   EQU     1       SERIAL DATA
     0002               PDATA   EQU     2       PARALLEL DATA
     0003               KDATA   EQU     3       KEYBOARD DATA
     00C8               DSTAT   EQU     0C8H    VDM CONTROL PORT
     00FA               TAPPT   EQU     0FAH    TAPE STATUS PORT
     00FB               TDATA   EQU     0FBH    TAPE DATA PORT
     00FF               SENSE   EQU     0FFH    SENSE SWITCHES
                        *
                        *
                        *           BIT ASSIGNMENT MASKS
                        *
     0001               SCD     EQU     1       SERIAL CARRIER DETECT
     0002               SDSR    EQU     2       SERIAL DATA SET READY
     0004               SPE     EQU     4       SERIAL PARITY ERROR
     0008               SFE     EQU     8       SERIAL FRAMING ERROR
     0010               SOE     EQU     16      SERIAL OVERRUN ERROR
     0020               SCTS    EQU     32      SERIAL CLEAR TO SEND
     0040               SDR     EQU     64      SERIAL DATA READY
     0080               STBE    EQU     128     SERIAL TRANSMITTER BUFFER EMPTY
                        *
     0001               KDR     EQU     1       KEYBOARD DATA READY
     0002               PDR     EQU     2       PARALLEL DATA READY
     0004               PXDR    EQU     4       PARALLEL DEVICE READY
     0008               TFE     EQU     8       TAPE FRAMING ERROR
     0010               TOE     EQU     16      TAPE OVERFLOW ERROR
     0040               TDR     EQU     64      TAPE DATA READY
     0080               TTBE    EQU     128     TAPE TRANSMITTER BUFFER EMPTY
                        *
     0001               SOK     EQU     1       SCROLL OK FLAG
                        *
     0080               TAPE1   EQU     80H     1=TURN TAPE ONE ON
     0040               TAPE2   EQU     40H     1=TURN TAPE TWO ON
                        *
                        *
                        *
                        *
                        *    S Y S T E M    G L O B A L    A R E A
                        *
C800                            ORG     START+0800H   RAM STARTS JUST AFTER ROM
                        *
     C800               SYSRAM  EQU     $       START OF SYSTEM RAM
     CBFF               SYSTP   EQU     SYSRAM+3FFH  STACK WORKS FM TOP DOWN
                        *
                        *
                        *   PARAMETERS STORED IN RAM
                        *
C800                    UIPRT   DS      2       USER DEFINED INPUT RTN IF NON ZERO
C802                    UOPRT   DS      2       USER DEFINED OUTPUT RTN IF NON ZERO
C804                    DFLTS   DS      2       DEFAULT PSUEDO I/O PORTS
C806                    IPORT   DS      1       CRNT INPUT PSUEDO PORT
C807                    OPORT   DS      1       CRNT OUTPUT PSUEDO PORT
C808                    WCHAR   DS      1       CURRENT CHARACTER POSITION
C809                    LINE    DS      1       CURRENT LINE POSITION
C80A                    BOT     DS      1       BEGINNING OF TEXT DISPLACEMENT
C80B                    SPEED   DS      1       SPEED CONTROL BYTE
C80C                    ESCFL   DS      1       ESCAPE FLAG CONTROL BYTE
C80D                    TSPD    DS      1       CURRENT TAPE SPEED
C80E                    INPTR   DS      2       PTR TO NEXT CHAR POSITION IN INLIN
C810                    NUCNT   DS      1       NUMBER OF NULLS AFTER CRLF
C811                    IGNCR   DS      1       IGN CRC ERR FLAG, FF=IGN CRC ERRS, ELSE=NORMAL
                        *
C812                            DS      10      ROOM FOR FUTURE EXPANSION
                        *
                        *
                        *
                        *
                        *
```

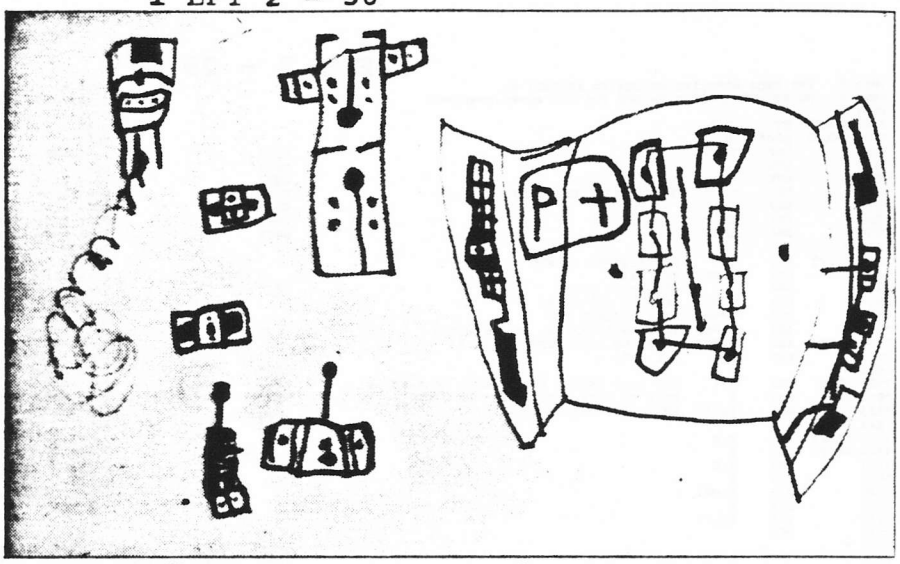
1 EPT 2 - 30

Waiting for BASIC-5

A lot of people have asked why Sol BASIC-5 took so long to be released. The main reason is Processor's policy conning the release of new products: we don't ship 'til we have the finalized version, and that means product PLUS documentation. We feel that our reputation is based on selling products that live up to the advertising claims, and we intend to live up to our reputation. (Note: we've never been forced to recall a product.)

We could have provided you with a version of our original BASIC-5 a long time ago, with a modification of only 12 instructions. But that wasn't what we advertised Sol BASIC-5 to be. And in getting it to be what we wanted, we kept thinking it would be neat to add just a couple more nifty features and then just one more and one more and . . . So as with all fanatics, one thing led to another 'til somebody remembered that this *is* a business and there *are* customers out there who could only stand so much "neat stuff" (especially if they had to wait until the year 2000).

Anyway, by now you will have received said BASIC-5, so load it in and run it for a while. We think you'll agree that it was almost worth the wait.



Computer of the Future

Contributions Welcome!

Drawing by Brian Marsh, Age 6.

PROCESSOR TECHNOLOGY
ACCESS.

Vol. 1, No. 2

Bulk Rate
U.S. Postage
PAID
Permit No. 3729
Oakland, Calif.

Processor Technology Corp.
6200 Hollis Street
Emeryville, CA 94608

PROCESSOR TECHNOLOGY

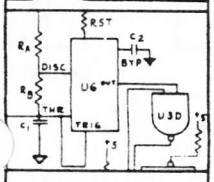
ACCESS.

75¢ per issue

Published by Processor Technology Corp., Emeryville. Volume One, Number Three, June 1977

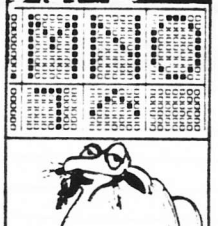
In This Issue

A Photo Essay of the First Annual West Coast Computer Faire, by our own multi-talented Rick Meyer. Lots of people, lots of excitement. Take a look for yourself ..... page 4 & 5.



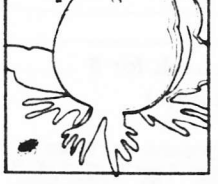
Software Technology debuts! An informative article on the care and choosing of your cassette tape system .....page 11.

How to stop your Sol and see what's happenin'. The first in a series of articles on creating a Run/Stop circuit for your Sol .....page 7

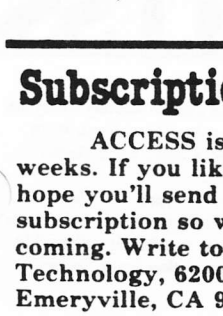


Sol goes international. Easy operation of the Sol at 50 Hz ....page 2.

Letter from the Editor .....page 3.



Testing your audio cassette hardware; a useful matching language routine .....page 8.



Reader Feedback ..... page 6 & 7.

The Newest from Newett Awl: swapping bytes .....page 3.

Bug Squad .....page 9.

Subscription Information

ACCESS is published every six weeks. If you like what you see, we hope you'll send us $4.00 for a year's subscription so we can keep the info coming. Write to us at Processor Technology, 6200 Hollis Street, Emeryville, CA 94608.

Here we are, now in the third issue of ACCESS, and as Aram and I have told you in the past, we openly solicit your thoughts, opinions, input and output on many fronts. Critiques are always welcome — some good ideas have been coming in the mail, and we'll be happy to see others on how we can improve our service.

Notes on availability and delivery. Since the last ACCESS, we've pretty well completed deliveries on BASIC/5 and CUTER software packages. GAMEPAC 1, MATHPACK, and TREK 80 on cassette are next in line; deliveries on them will begin in July, and much of our other software will be forthcoming later this summer.

Deliveries of Helios II, our dual-drive floppy disc system, are targeted to begin in mid-to-late July. We'll ship the earliest orders first, but due to the large number of back orders on file, it will be some time before we're totally caught up on deliveries.

We've already begun deliveries of our 16KRA dynamic memory board — although slowly. If you have one on order, you can be assured of receiving it soon.

Specifications sheets for the new GPM (General Purpose Memory) Module will be available in July. You can get them from our Customer Service Department or your local PTC dealer.

Sol Users' Group. If you currently own a Sol computer, you may be interested in the Sol Users' Group. The Group was created to provide a dialogue among Sol owners, by sharing applications, software, and other information for mutual member benefit. The Sol Users' Group (not affiliated officially with Processor Technology) can be contacted through Bill Buras, 4190 Maybell Way, Palo Alto, CA 94306. (No phone calls, please!)

We need you! In the interest of providing continued informational support, especially in connection with interfacing Processor Technology hardware with other manufacturers' software, we'd like to hear your ideas, see your programs, whatever you've got. We will be happy to publish useful information that you might contribute. Mom will finally see your name in print!

PTC dealers. New ones are continually added to our ever-growing list. They're all authorized to both sell and service Processor Technology products, and to act as distributors for Processor/Software Technology software. To keep you up to date, the current dealer roster is listed in this issue of ACCESS. Of course, your dealer's always interested in hearing from PTC customers, so why not pay him a visit?

ACCESS subscriptions. Let us know if you've subscribed and are having trouble receiving your copy regularly. Is your mailing address correct as printed on the label? Have any suggestions for improvement?

ACCESS Issue No. 1 is now out of print; no back copies are available.

(continued on page 2)

50 Hz Operation of the Sol-PC

The demand for Sols is international, and a great many countries have power systems which operate at 50 Hz (cycles per second). Sol transformers are designed to operate at this frequency without derating, and a 220, 235 or 250 volt power supply is available from the factory.

In most countries where the line frequency is 50 Hz, the television display standards have been adapted so that the vertical sweep frequency is also 50 Hz (this reduces flicker). The resulting display contains 625 scan lines per frame, or 312½ lines per frame.

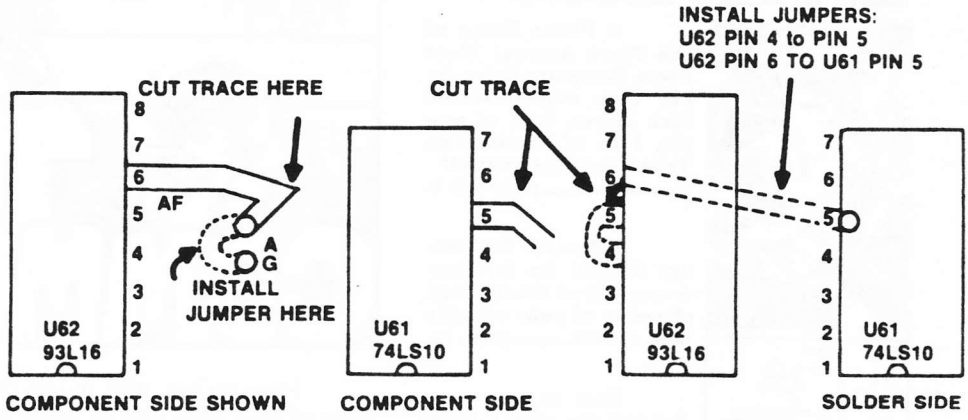
The frequency of the synchronization pulses produced by the Sol can be changed to this new standard quite easily on the E revision boards, and with a little more difficulty on the D revision boards. In both cases, the change involves disconnecting pin 5 of the U62 from pin 6 and reconnecting it to pin 4 of U62. Jumper pads have been provided for this purpose on the E revision, so that the change is a simple one-cut, one-jumper modification performed on the top side of the boards. The angled trace leading from pad AF is cut and a jumper is installed between pad AF and AG.

The D revision boards requires two trace cuts. One is made on the top side of the board between U 62 and U 61 (the trace leading from pin 5 of U 61 is cut). The remaining trace cut is made on the bottom (solder) side of the board. One is between pins 4 and 5 of U 62. Two jumpers must now be installed, both on the solder side of the board. One is between pins 4 and 5 of U 62. The other connects pin 6 of U 62 with pin 5 of U 61.

The VDM-1 may also be modified to allow 625-line operation. On all revisions of the board, cut the trace connecting pins 5 and 6 of IC8. Using a wire jumper, connect pin 5 of IC 8 pin 4 of IC 8. This is all that is required to modify the VDM-1.

These modifications will present a display of 16 visible text lines which appear somewhat more compact vertically than the 60 Hz display. This is because the time from the bottom of the display to the top has been doubled from the equivalent of four text lines to eight text lines.

Lee Felsenstein



(Cont. from Page 1)

Finally, we do appreciate all the support and feedback many ACCESS readers have provided. The comments, positive, negative or otherwise, indicate your interest in PTC and help us to give you what you want. I think those of you who have contacted us on the one-to-one communication basis (by phone or letter) have found responsive follow-through on our end. Further improvement can come from your suggestions, so let us know what you're thinking. Again, the number is (415) 652-8080 for teletroubleshooting (between 9:30 am and noon or 1:30 to 4 pm Pacific Time); for technically oriented software questions, call (415) 349-8080.

See you next issue.

Ralph I. Palsson

Your ALS-8 Applications Notes are on the Way!

After a seemingly interminable delay, the first batch of Application notes are really and truly in the mail to all you ALS-8 Users' Group members. Once the material is in your hot little hands (in a very fancy binder, no less), you'll be better able to appreciate the power and versatility of your ALS-8, and you should have some very happy hours of computing.

If you haven't received your notes by the time you read this, please drop us a note right now, so we can track down whatever clerical or shipping errors crossed you up.

Our sincerest apologies for the delay.

Will Vern ever finish telling us the "real story" behind the VDM-1? Yes, indeed! The continuing saga of the Flickerless VDM-1 has been temporarily interrupted. It'll be back with the next issue. Stay tuned.

Basic Theorem of Programming: The program is absolutely right, therefore the computer must be wrong.

Subscription Blank

Name _____

Address _____

City _____

State _____

Zip _____

Enclosed is a check for $ _____ **fo**

____ subscription(s)

PROCESSOR TECHNOLOGY ACCESS.

Editor: Aram Attarian II

ACCESS is published approximately every six weeks. Subscription rate: $4 per year. from Processor Technology Corp., 6200 Hollis St., Emeryville, CA 94608.

ACCESS Copyright © April 1977 by Proc Technology Corp. All Rights Reserved. Mater in this publication may not be reproduced in any form without permission from Processor Technology Corp.

A Letter From the Editor

"Give me a cigarette, because I'm so hungry that I got nobody to sleep with tonight."

Old Russian Proverb

Well, who would have thought it. ...e I was at 8 o'clock Saturday mo...g, slightly hungover, arriving at th... Civic Center for the First West Coast Computer Faire — and there were already people lined up clear around the corner. (Unless I was still seeing double from the Processor Technology Dealer Banquet, which featured Peking Duck, a few Tsingtsao beers, Sweet and Sour Chicken, a few more Tsingtsao's, Manchurian Beef, a few etc.)

By the time the last plug was pulled Sunday night, more than 12,000 people had paid $6 to $9 a head to get a look at the latest marvels from Processor, IMSAI, Polymorphic, Apple, Byte Shops, and well over 100 other exhibitors. There were lots of neat things: programs for kids, color graphics, music (including the premiere of the Software Technology Music System — but you can read about that in our literature. No advertising here). And seminars on everything from tutorials for the complete novice to bus and interface standards. The building was basically hot, and it was crowded, and everybody was pr... high.

...ll in all, the impression was that ... was a very good event. The people who planned the Faire did a good job, and we'll probably draw even bigger crowds next year, what with people like Radio Shack and Heathkit getting into the act. The price probably put it out of range for a lot of people who might have come this time, but given the rate of growth in the home computer field ...

Aram Attarian II

P.S. Now that we've got ACCESS going, and we're starting to get some feedback, we're re-evaluating our procedures. Issue No. 4 will be a little late, because we're going to make some changes to make it a better quality production, in line with some of the suggestions you've made. So long 'til then.

Attention, CONSOL Users

All Processor Technology software has recently been modified in a very important way that encourages standardization. If you've been looking forward to having some Sol software for your CONSOL Proms, you'll have to let us re-program for you. There's *no charge*. Just mail your CONSOL Proms, suitably packaged, to Processor Technology; we'll take care of the rest. Turn-around time is about 2 weeks, dependent on the vagaries of the U.S. Postal Service, natch.

If you're planning to upgrade to SOLOS anyway, don't worry about it.

Newett Awl's Screen Sort

This little program does what's called a bubble sort: each byte in the sorted area is compared to the next higher byte. If the first byte has a higher value, the two are swapped in the memory. Then the same thing is done for the next two, and so on through the entire sort area. After the first pass, the last byte in the sort area will be the highest value, so when you set up to pass over the sort area again, you can shorten it by one byte at the high end. And if you keep track of what swaps take place during the sort, you can quit as soon as no swapping occurs.

I have located this routine in low memory, so you don't need much to play with it. I used the VDM screen for the sort area, so you can see exactly what's going on — if your eyes can be synched to the screen! Try putting a whole mess of garbage on the screen, then executing the program at location 0000. Your screen may never be the same!

Those of you who are adventuresome and mischievious might try modifying line 0047 to read 0047 SIZE DW, 2047 ARBITRARY LENGTH and line 0048 to read 0048 TOP EQU, 0200H USE TV DAZZLER AREA.
Then if you have Lichen Wang's Kaleidoscope, assemble this program so that it is above 0900 (HEX), get Kaleidoscope running, then stop and run the sort program.

I've purposely coded the program so that it will sort any length field by manipulating the SIZE and TOP. A little practice and imagination should do the rest. I sorted the train program (naturally), but it broke my heart to see the little engine torn apart so badly and so quickly.

A couple of program notes: at line 40, the zero flag is being set or reset by the ADD B in line 37. As we all know but often forget, the DCX H at line 39 has no effect on the accumulator flags. At line 25, I just arbitrarily stuff register B with the last byte looked at. This is an interesting use for the flag. At line 37, all I need to know is whether or not swapping occured. If the test at line 20 tells me that a byte in register A has a higher value than the byte in memory, I know for sure that it isn't zero and can safely use it for a simple flag.

Reprint Permission of Newett Awl

```
0001 ;*************************************************
0002 ;*                                               *
0003 ;*  NEWETT AWL'S SCREEN SORT FOR THE SIGHT       *
0004 ;*              OF YOUR LIFE!                    *
0005 ;*                                               *
0006 ;*                                               *
0007 ;*************************************************
0008
0000 2A 3F 00       0009 SORT   LHLD   SIZE    GET SIZE OF SORT AREA
0003 7D             0010        MOV    A,L     SEE IF SIZE IS
0004 B4             0011        ORA    H       ..ZERO
0005 CA 3A 00       0012        JZ     DONE    IF YES, LEAVE
0008 22 3D 00       0013 CON1   SHLD   COMPCT  SET UP COMPARISON POINT
000B 22 3B 00       0014        SHLD   ITMCNT  RESER ITEM COUNT
000E 01 00 00       0015        LXI    B,0     CLEAR B/C
0011 21 00 CC       0016        LXI    H,TOP   POINT TO TOP OF AREA
0014 11 01 CC       0017        LXI    D,TOP+1 POINT TO TOP+1 OF AREA
0017 1A             0018 CON2   LDAX   D       GET ITEM B FROM AREA
0018 BE             0019        CMP    M       COMPARE TO ITEM A
0019 D2 21 00       0020        JNC    CON3    GO CON3 IF A<B
001C 4E             0021        MOV    C,M     GET A TO REG. C
001D 77             0022        MOV    M,A     PUT B WHERE A WAS
001E 79             0023        MOV    A,C     PUT A WHERE B WAS
001F 12             0024        STAX   D       PUT A WHERE B WAS
0020 47             0025        MOV    B,A     USE BYTE FOR SWAP FLAG
0021 23             0026 CON3   INX    H       INCREMENT A POINTER
0022 13             0027        INX    D       INCREMENT B POINTER
0023 E5             0028        PUSH   H       KEEP MEMORY REFERENCE
0024 2A 3B 00       0029        LHLD   ITMCNT  RESTORE ITEM COUNT
0027 2B             0030        DCX    H       DECREMENT IT
0028 22 3B 00       0031        SHLD   ITMCNT  RESTORE ITEM COUNT
002B 7D             0032        MOV    A,L     TEST IF
002C B4             0033        ORA    H       ..ITEM COUNT IS ZERO
002D E1             0034        POP    H       GET MEMORY REF. BACK
002E C2 17 00       0035        JNZ    CON2    NO, GO DO MORE
0031 AF             0036        XRA    A       CLEAR FLAGS
0032 80             0037        ADD    B       GET SWAP FLAG
0033 2A 3D 00       0038        LHLD   COMPCT  GET NEW ITEM COUNT
0036 2B             0039        DCX    H       REDUCE COMPARE COUNT BY 1
0037 C2 08 00       0040        JNZ    CON1    GO CON1 IF IT DID SWAP
003A C9             0041 DONE   RET
                    0042 ;
                    0043 ; WORKING STORAGE AREA
                    0044 ;
003B 00 00          0045 ITMCNT DW     0       INNER ITEM COUNT
003D 00 00          0046 COMPCT DW     0       NO. OF ITEMS IN THE LIST
003F FF 03          0047 SIZE   DW     1023    ARBITRARY LENGTH
     CC00           0048 TOP    EQU    0CC00H  USE SCREEN FOR AREA OF B
                    0049 ;

                    0004 ; END OF PROGRAM

COMPC  003D    0013 0036
CON1   0008    0040
CON2   0017    0035
CON3   0021    0026
DONE   003A    0012
ITMCN  003B    0014 0029 0031
SIZE   003F    0009
SORT   0000
TOP    CC00    0016 0017
```

Eggleston's Extension Principle: Programming errors which would normally require one day to find will take five days when the programmer is in a hurry.

Top Photo, Processor Technology's booth at the First West Coast Computer Faire, with some spacy stuff up on the big display. Yep, it really was that crowded all weekend.

Left, Kids engrossed in a game of Target. Maybe you're right about CHOO-CHOO, Joe — still, I know some big kids that got off on it. (See Reader Feedback).

Above, Kids of all ages got turned on by Sol.

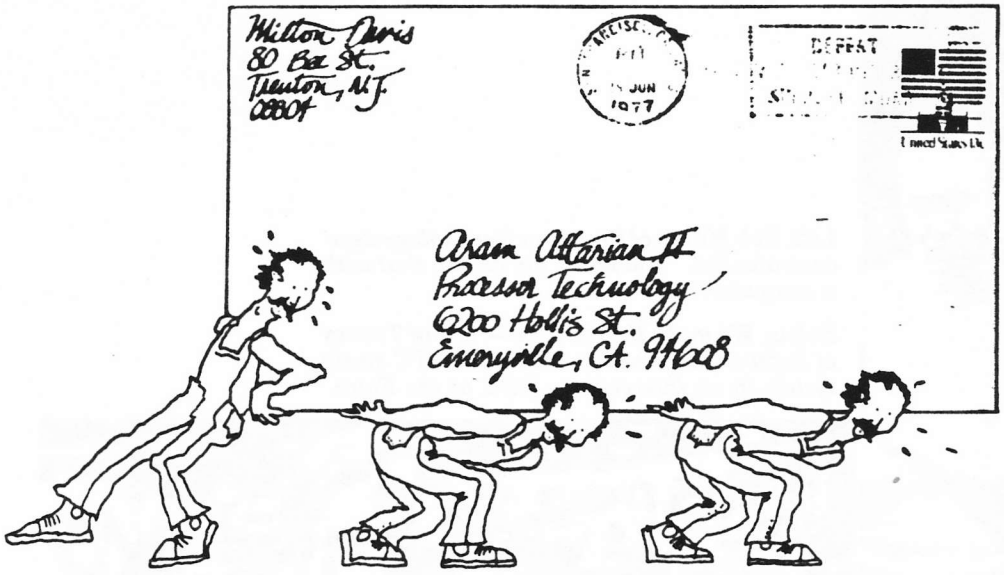
At right, The Processor booth. No, that's not me in the background. Bet he plays a mean game of Target, though.

Left, Bob Marsh of Processor Technology demonstrates Sol. "You mean you can do that with a computer?"

Below, We make the big time — Glenn Tenney of Software Technology details the STC music system in an interview by NBC at the Faire.

Left, And here I am in my red suspenders, as promised.

All photos by Rick Meyer



Reader Feedback

We don't have space to print all the letters we're receiving, but we'll try to get as many in as we can. We reserve the right to edit for space and content so we can include as wide a cross-section of opinion as possible. We'll reply to all of you personally if you include a self-addressed stamped envelope.

Aram Attarian II

Sir:

As evidenced by the many excerpts in ACCESS, you obviously enjoy Conrad Schneiker's **Abridged Collection of Interdisciplinary Laws.** Although the **Laws** are not copyrighted, don't you think it would only be fair to credit the source?

Gregg Townsend
Sunnyvale, CA

Dear Gregg:

My source was **Logical Laws, Accurate Axioms, Profound Principles, Trusty Truisms, Homey Homilies, Colorful Corollaries, Quotable Quotes, and Rambunctious Ruminations.** This mouthful is the title of a pamphlet "compiled by the computer company that's out to make life a little simpler." Happy to credit both sources.

AA II

Dear Aram:

Enclosed check for $4 for a year of ACCESS. Now my observations.

1. You are doing it wrong. Should be notebook-size sheets — how does one file for reference?

2. Last year I filled out a card for which I was supposed to receive something over the year, supposedly edited by Gordon French. What happened?

3. In order for ACCESS to be meaningful, it should be understandable. The biggest fault I find with all publications is that someone just plunks down a program, period. There is a crying need to explain, in detail, at least a few times, the meaning of all the listings as they relate to hardware. **Nobody** to date has done this. Why do editors assume that we hobbyists know all this?

Joseph P. Chalala
Willow Street, PA

Dear Joe:

Thanks for the sub. In answer to your points:

1. Our format will be changing in the near future, and we'll take your suggestion into consideration. Good point.

2. See the reference to the ALS-8 User's Group in this issue (also in last issue).

3. Most listings include comments that are supposed to explain the software, but you're right, more text is needed for a really good understanding.

AA II

Dear Gentlepersons:

As you hopefully know by now, Newett Awl's choo choo train has at least one error in it. (Vol 1, No. 1 of Access). At address OODO the listing (page 11) shows a "A8", while the dump format on page 12 shows a "AB". Now, please, I beg of you, tell me what kind of an assembler you have that generates incorrect code . . . (by the way, an "AB" wipes out most of memory, I trust the "A8" to be correct). But even so, we have entered and checked the code three times, and it certainly does't seem to work on my Sol . . . the front part of a steam engine does appear, and it vibrates back and forth for about 5 seconds, and then as it apparently attempts to do something else, it crashes, writing "E9" 8S at 25, 26, A5, A6, 125, 126. Perhaps it could be a hardware problem, you will have better idea about that than I.

2) Would it be possible to provid me with a copy of SOLOS and SOLED (a listing would suffice), as I would like to find out the correct way to function the tape drives among other things. I d appreciate your relative thoroughness i design and implementation of the Sol, you did a rather professional job.

I am managing the West Lafayette Data Domain Store and I pu together a Sol-PC. In the mean time, w have lost the envelope and registration etc. Would you please either send another set of reg. cards & whatever or simply send what I was supposed to asl you to send (other chapters of the manual, personality module exchange, whatever, I've forgotten even what it was . . .)

J. Gregory Madde
Mgr. Data Domain of W. Lafayett
Pres. JGM Development Lab

Dear Greg:

1. Choo choo was written for VDM-1, and we'll be publishing Sol patches and updates in the next ACCESS.

2. See this issue for SOLOS

3. Customer Service has ta care of your problem by now.

AA

Dear Aram:

In the "Bug Squad," ACCESS 2. Fix No. 5, you show a cut on trace to pin 26 of J11. The descriptive matter says pin 73. Which is correct and are th other fixes as accurate?

Dr. George L. Hall
Naples, Flori

Dear George:

See "Bug Squad" in this issue. The other fixes are OK.

AA

Mr. Editor:

As of this date your world renowned (sic) service mgr. is still promising heaven on earth to us (I will admit) suckers! I suggest you change "protecth" to "mana-na." That's Spain for tomorrow, capito. (sic, sic, and sic).

Anyway, in my opinion you guys are full of crabs, who in his right mind would establish a computer company in a "rundown, skid row, full of yech! Goo forsaken place like Emeryville." Obviously, this company did, even the newspaper it publishes and perso a affected by the locale — it's full Care to publish this letter go ahead.

Your ads are pretty and very convincing, look who it caught with $1000 plus to buy your A=$!!?? Sol-20 board that won't work **ME.** (and a few

more sics) "Whoever heard of a Sol not working." Look for mine stupid!

If I had to do it all over again — will buy a Heathkit color TV set kit t a "goddam" Protech Sol-20. Count out as your ardent supporter of Sol — it's for the birds.

Hal Martino II

Dear Hal:

Have you hugged your kids today? Love,

AA II

Dear Ralph and Aram:

Here are some general and specific comments on ACCESS.

1. The software listings are good and useful, but they take a lot of room. Might it not be better to bind up the long ones separately and sell like No. 1 and No. 2? Actually, I do hope you will provide listings of SOLED, 8K BASIC & FOCAL, ALS-8, TREK 80, etc. I am somewhat of a software listing junky — that's why I'm a PTC customer (i.e., free software).

1A. Choo Choo Train is a bit immature, games like Life and Target are more suitable even for kids. I have a 9-year-old playing Trek.

2. I am a little concerned that you are using ACCESS to distribute errata for Sol. I think you should also compile and send out loose-leaf sheets for the instruction book — or even charge for an updated instruction set. Sol systems are likely to be around for a long time and the book is likely to remain with the system, but a stack of old ACCESS's is likely to be lost and with them the errata.

2a. Should not Errata No. 3 (Ground Noise Fix) concern itself with IC #107 and not IC #33?

2b. Any reason for not revising the 8 leads to J2 on the card (cut and jump) if I want? (i.e., is it more involved?)

3. "Waiting for BASIC-5." I do see both sides of the issue. I do expect a properly designed and debugged product, with good documentation, for my money and I will wait a reasonable time for it. And I have seen how several publications have been very critical of certain minor problems — which can really hurt a manufacturer.

But then again, it's a new product and it is intended for the sophisticated hobbyist who is capable of understanding and making fixes to PC cards and software. Sol and SW#2 are good examples. Still I cannot believe that things take as long as they do, or have in the past. Perhaps the answer is a system where you have two levels of customers — those who want preliminary releases and are willing to pay for fixes or updates, and those who want a final, warranted product.

Joseph Gaffney
Lyndhurst, NJ

Dear Joe:

1. See the letter from Joe Chalala.

2. This will be the case in the future, but you have to consider the amount of bookkeeping necessary for that type of project. We'll be using Helios II to handle it, but they're hard to get right now.

2a. No.

2b. No.

Thanks for a very thoughtful letter. Readers like you will help us make ACCESS a lot more responsive.

AA II

A Run/Stop Circuit for User Control of Sol's X-Ready Line

The ability to stop and examine the various state of a system as the program is executing allows the user to detect errors embedded within the system. To this end, I have written a series of articles dealing with the implementation of "Front Panel" features on an S-100 system. These should assist both hobbyist and professional with standard system maintenance.

This first installment deals with stopping or slowing the instruction execution time from tens of microsecond to a user defined time. As most S-100 bus processors follow a set of protocol, the heart of any front panel is the run/stop circuit. By holding down the ready line, the user can cause the processor to idle as long as that line is down. Thus, the toggling of the ready line will allow the slowing or stopping of the processor from machine cycle to machine cycle.

With the R/S switch, you can control the X-READY line by setting and resetting the R/S (Run/Stop) flipflop U1A. With the R/S switch in the run position, the R/S flipflop is reset on Φ2, forcing X-READY high. In the stop position, the switch NAND's together the stop value (Φ2, PSYNC) and the MI/FETCH cycle (DO5). A new fetch will then set one input high to the X-READY NAND gate.

When the R/S flipflop is set in the stop position, the state of the X-READY line is determined by the state of the single step S/S (single step) flipflop. If it's set, the $\overline{Q}$ output applied to the other input of the X-READY NAND gate U3C will be low. And the X-READY line will be high, thus allowing the processor to run. The S/S flipflop will be reset at the next PSYNC pulse, stopping the processor.

The inverters U2D and U2E are wired as a R/S flipflop to debounce the single step switch. SSS U6 is a multi-vibrator used to set the S/S flipflop at a variable rate. The vibrator will be inhibited by U2C until the multiple step switch is depressed; its rate is determined by the values of RA, RB, and C1:

Charge time — .685 (RA + RB) C1
Discharge time — .685 (RB)C1
Period — .685 (RA + 2R2)C1

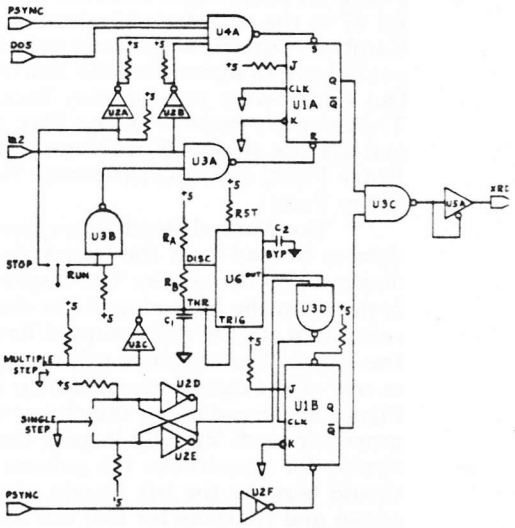
$$\text{Frequency} — \frac{1.46}{(RA + 2RB)C1}$$

Parts list:

U1 74109 For approximately 10 steps per second
U2 7406 RA = 3.3 meg
U3 7400 RB = 3.3 meg
U4 7420 C1 = .01
U5 555 C2 = .001 uf (C2 is a bypass capacitor of approximately .001 uf)
U6 8T97 The pull-up resistor values are 2.2K
Stop/Run switch SPDT
Single Step switch SPPT
Multiple Step switch SPST Normal Open

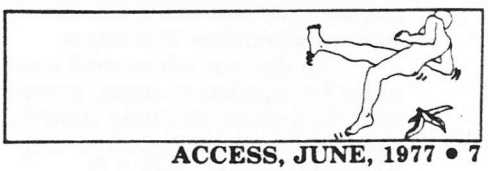
In the next of the series, I will diagram the proper connection of indicator lights to the S-100 bus, showing use of the first two installments.

Steve Wong



OOPS!

Our kit packers made a mistake! A few Sol kits shipped in April or May '77 contain form 74LS153 integrated circuits. Do not use these chips. We will gladly replace them with the proper part, i.e., 74LS253's. Let us know, and we'll take care of it immediately.



Audio Cassette Test Routine

This is a machine language routine written to run on a Sol PC, Sol 10 or Sol 20 as a functional test of the audio cassette hardware. It uses only the ENTER, DUMP, and EXECUTE commands, and requires no additional memory.

Enter the routine through the command mode as shown in the first section of the program. At the end of this section, your Sol will return to the command mode. Then use the second section to dump memory C900 through C94F. Check that the data has been entered correctly; ignore bytes C946 through C94F.

Put good quality blank tape in your recorder and connect its aux input to the Sol Audio Out J6, using audio cable. Set the recorder's volume and tone controls per Sol Manual Section 8. Push Record and Play simultaneously to move past the leader. Pure clocks will be recorded until you give the command EXEC C900.

This command will cause a 256-bit incrementing data pattern to be recorded continuously until you stop it. When you've recorded as much tape as you want, you'll have to do a restart and return to the command mode. Rewind the tape to the beginning, then attach an audio cable from Sol Audio IN J7 to the recorder's Monitor or Earphone output. With some recorders, you'll have to disconnect the Aux/Audio Out cable before you can play back. Then put the recorder in the Play mode and execute either C90E (Normal Read Entry Point) or C91C (Alternate Read Entry Point).

The Normal Read Entry causes data to be read from the tape and displayed on the screen. The display will begin when the beginning of the data is recognized, and it will occupy 6 lines of the screen. The sequence will correspond to one of the data patterns shown in Figure E, depending on the character generator used. Visually inspect the display for correctness: the pattern should begin at the left margin of the screen and continue for two full lines, then repeat with reversed screen polarity for another two lines. The last two lines will repeat the original pattern to fill out the display. Now, data will continue to be read from the tape and written **over** the previous display: only the character polarity should change, causing the display to alternate from white-on-black to black-on-white. Neither the horizontal nor vertical position should change; a shift in position indicates either the loss or addition of random characters.

To discover where read errors might be expected to occur, experiment with the volume and tone controls, audio cables, or phase-locked loop adjustments.

You can realign the received data display at any time by doing a restart to command mode and then executing C90E again. Careful examination of the display will reveal any out-of-sequence characters, which indicate read errors.

If the beginning of the pattern can't be found, use the Alternate Read Entry Point to display whatever is read.

Pat Tu

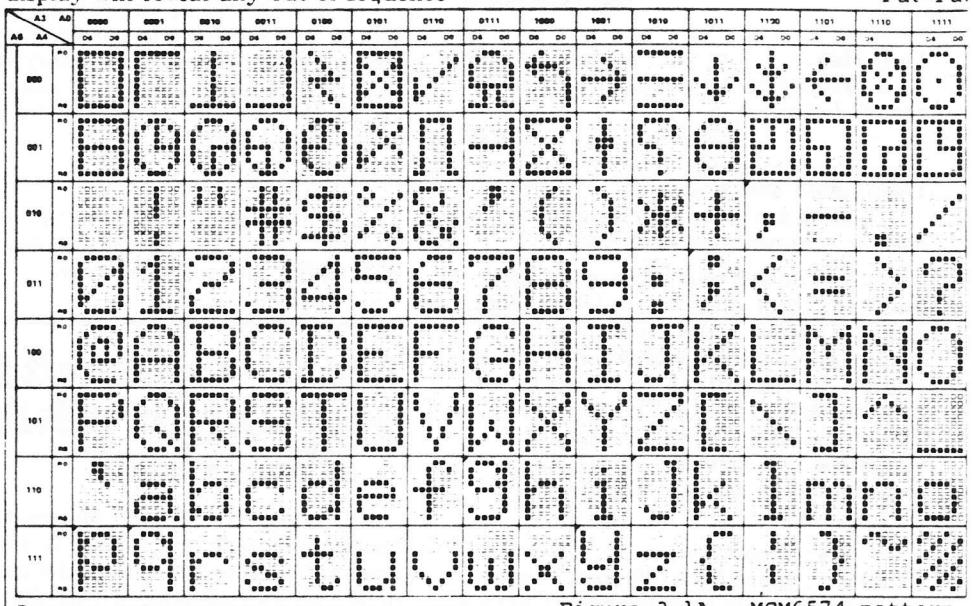


Figure 3-1A. MCM6574 pattern.

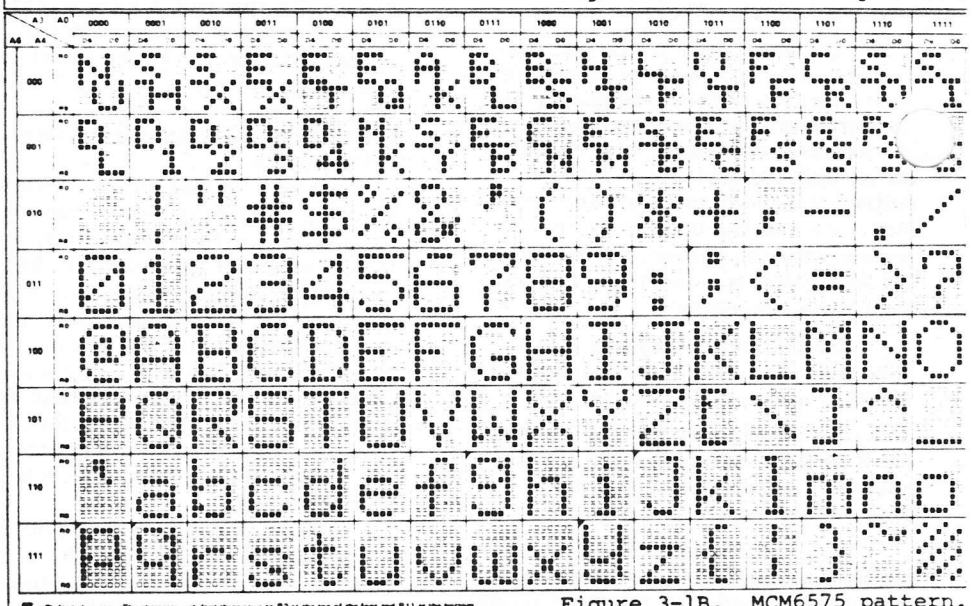


Figure 3-1B. MCM6575 pattern.

```
C900                    8001 START  EQU   0C900H
C900                    8002 RD1    EQU   0C90EH
C900                    8003 RD2    EQU   0C91CH
C900                    8004 RD3    EQU   0C91FH
C900                    8005 RD4    EQU   0C932H
C900                    8006 RDA    EQU   40H
C900                    8007 TBE    EQU   80H
C900                    8008 STS    EQU   0FAH
C900                    8009 DAT    EQU   0FBH
C900 DB FA              8011 START  IN    STS     GET TAPE STATUS
C902 E6 80              8012        ANI   TBE     AND WAIT FOR TBE
C904 CA 00 C9           8013        JZ    START
C907 0C                 8014        INR   C       COMPUTE PATTERN
C908 79                 8015        MOV   A,C
C909 D3 FB              8016        OUT   DAT     WRITE TAPE
C90B C3 00 C9           8017        JMP   START   FOREVER
C90E DB FA              8018 RD1    IN    STS     GET TAPE STATUS
C910 E6 40              8019        ANI   RDA     AND WAIT FOR RDA
C912 CA 0E C9           8020        JZ    RD1
C915 DB FB              8021        IN    DAT     GET TAPE DATA
C917 DE FF              8022        SBI   0FFH    WAIT FOR START
C919 C2 0E C9           8023        JNZ   RD1
C91C 21 00 CE           8024 RD2    LXI   H,0CE00H  POINT TO SCREEN
C91F DB FA              8025 RD3    IN    STS     GET TAPE STATUS
C921 E6 40              8026        ANI   RDA     AND WAIT FOR RDA
C923 CA 1F C9           8027        JZ    RD3
C926 DB FB              8028        IN    DAT     GET TAPE DATA
C928 77                 8029        MOV   M,A     PUT IT ON THE SCREEN
C929 2C                 8030        INR   L       COMPUTE SCREEN POSITION
C92A 7D                 8031        MOV   A,L
C92B DE 00              8032        SBI   00H
C92D C2 1F C9           8033        JNZ   RD3     GO GET MORE DATA
C930 26 CF              8034        MVI   H,0CFH  SET SCREEN OVERFLOW
C932 DB FA              8035 RD4    IN    STS     GET TAPE STATUS
C934 E6 40              8036        ANI   40H     AND WAIT FOR RDA
C936 CA 32 C9           8037        JZ    RD4
C939 DB FB              8038        IN    DAT     GET TAPE DA
C93B 77                 8039        MOV   M,A     PUT IT ON T
C93C 2C                 8040        INR   L       COMPUTE POS
C93D 7D                 8041        MOV   A,L
C93E DE 00              8042        SBI   00H
C940 C2 32 C9           8043        JNZ   RD4     GET MORE DA
C943 C3 1C C9           8044        JMP   RD2     REPEAT READ

DAT   00FB    RD1   C90E    RD2   C91C    RD3   C91F
RDA   0032    RDA   0040    START C900    STS   00FA
TBE   0080

>EXEC C900      *** CAUSES DATA PATTERN TO BE WRITTEN
                    TO AUDIO CASSETTE INTERFACE

>EXEC C90E      *** NORMAL READ ENTRY POINT.  READS FROM T
                    WAITS FOR BEGINNING OF PATTERN, AND DI
                    DATA READ FROM CASSETTE ON THE SCREEN

>EXEC C91C      *** ALTERNATE READ ENTRY POINT.  READS FRO
                    AND DISPLAYS DATA ON THE SCREEN

*** USUALLY INSPECT DATA ON SCREEN.  YOU S      E
    INCREMENTING PATTERN FOR SIX LINES.  THE HOR      L
    OF EACH CHARACTER MUST NOT CHANGE.  YOU WILL SEE EAC
    CHARACTER ALTERNATE FROM WHITE-ON-BLACK TO BLACK-ON-
    AT A 300 OR 1200 BAUD RATE.  CHECK FOR CHARACTERS OU
    SEQUENCE.
```

1 EPT 2 - 39

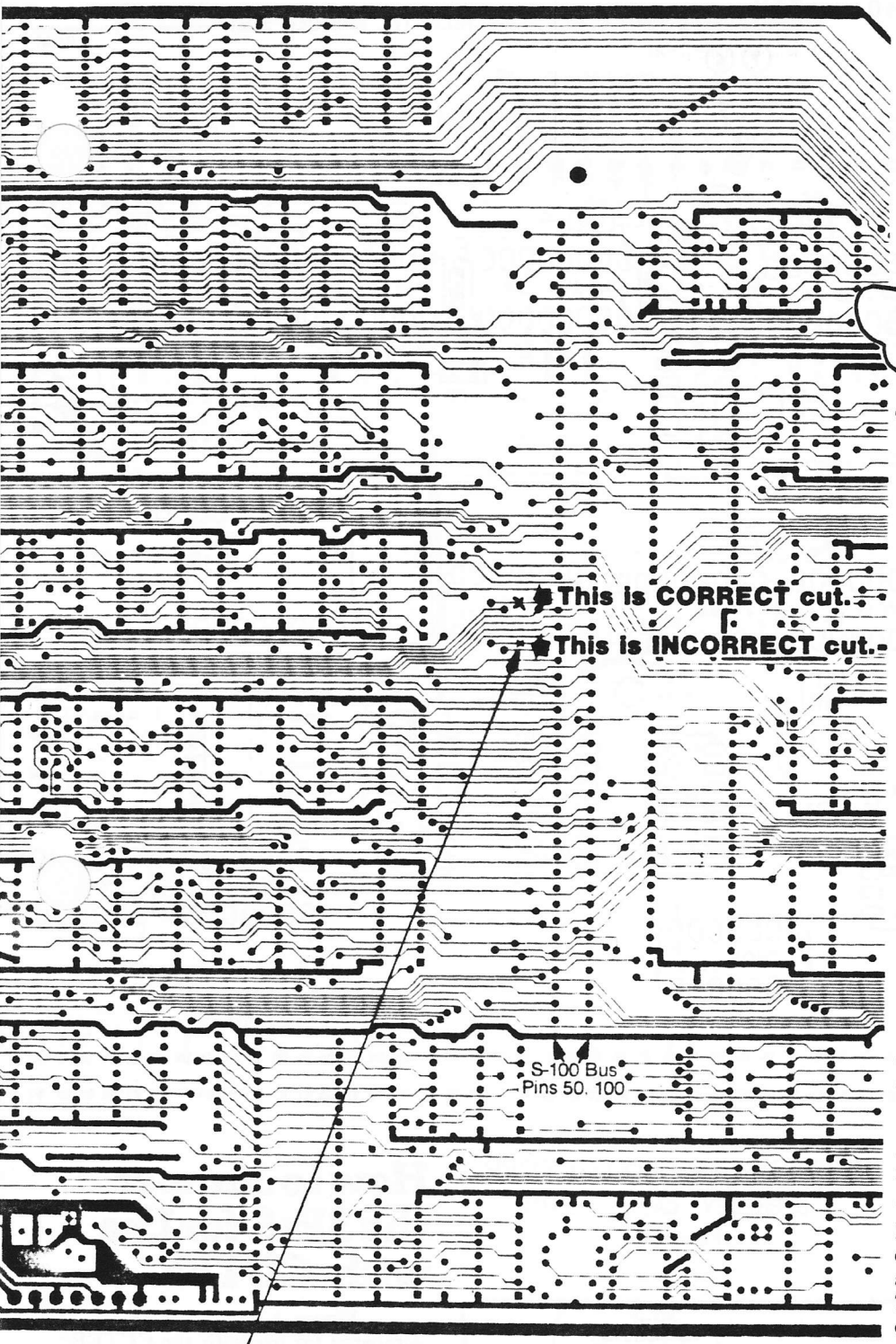


Figure C

Bug Squad

Error in the Sol Input/Output Schematic

The bug: Two pin numbers are transposed on the schematic.

The squasher: The pin for J1 (Carrier Detect-EIA) SCD should be labeled "8"; the pin for J1 (Data Set Ready-EIA) SDSR should be labeled "6". (See figure B)

/done

Error in the Sol Manual

The bug: There are two mistakes on page AVII-7 pertaining to the Serial I/O Control.

The squasher: On S4-1, reverse the "odd" and "even." On S4-2 and S4-3, all "ON's" should be "OFF's" and all "OFF's" should be "ON's.

/7-8-77 done

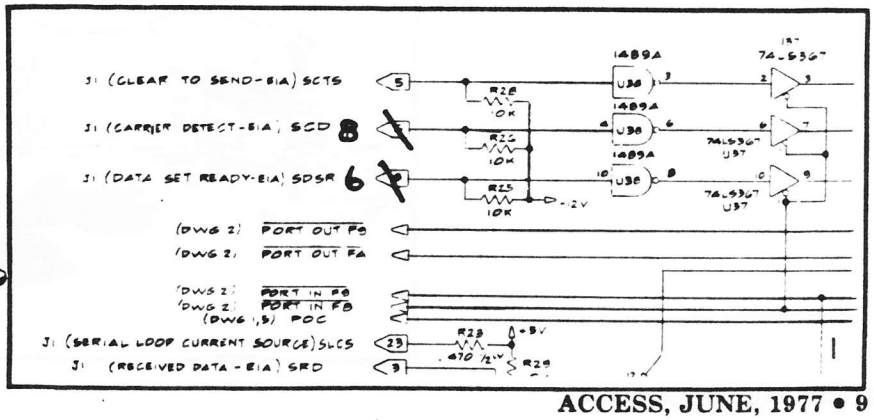
Error in the DMA/Interrupt Unscramble Fix

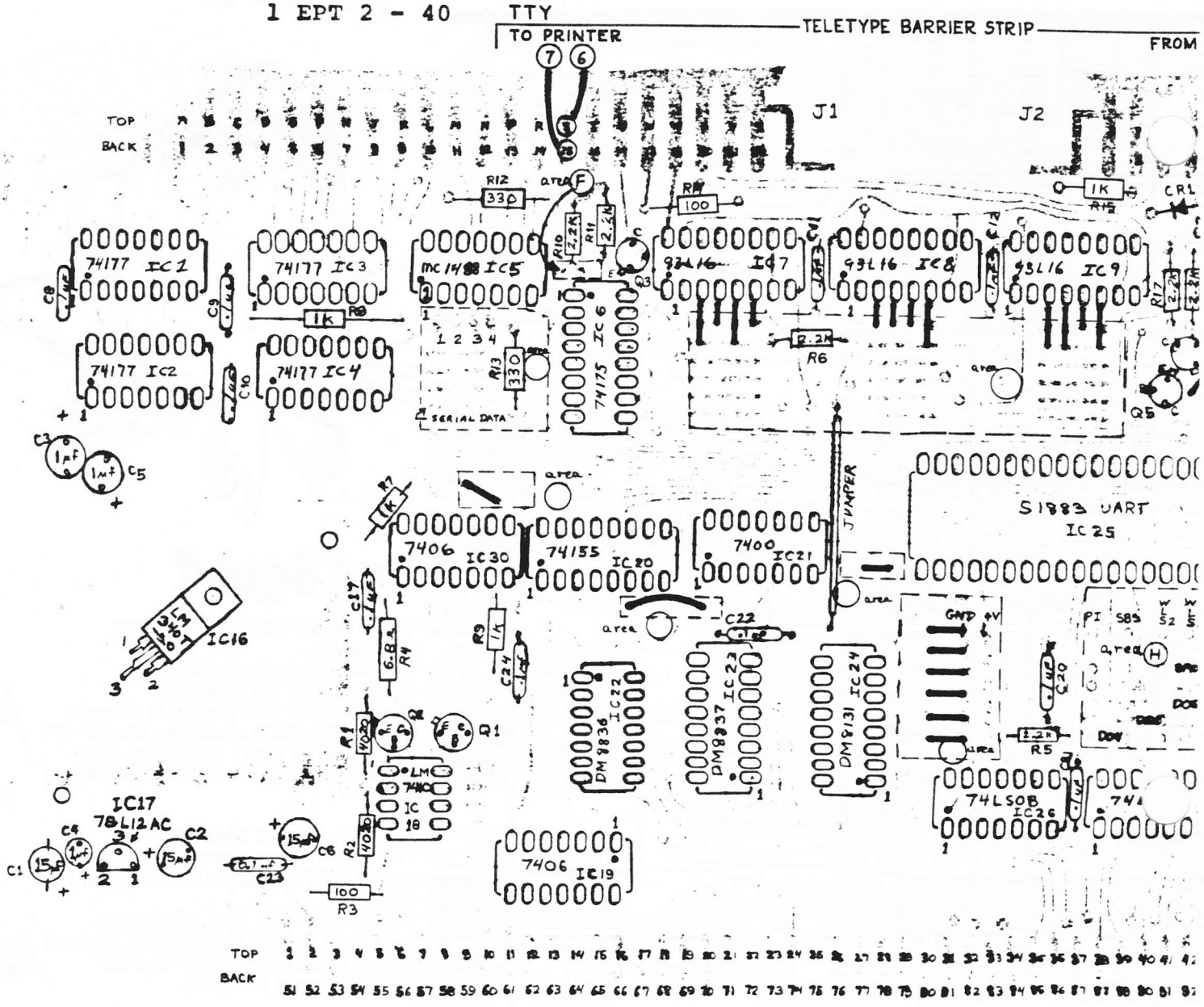
The bug: We gave you this fix in the last issue of ACCESS; the wording was correct, but the drawing that accompanied it wasn't.

The squasher. Figure C shows the correct change.

/OK

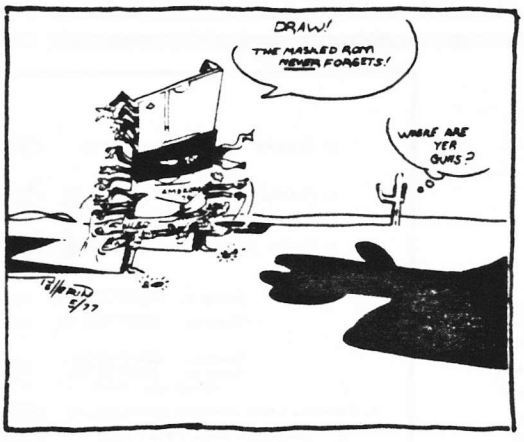
Figure B





© 1975 PROCESSOR TECHNOLOGY CORP.
BERKELEY, CALIF. 94710

Figure A



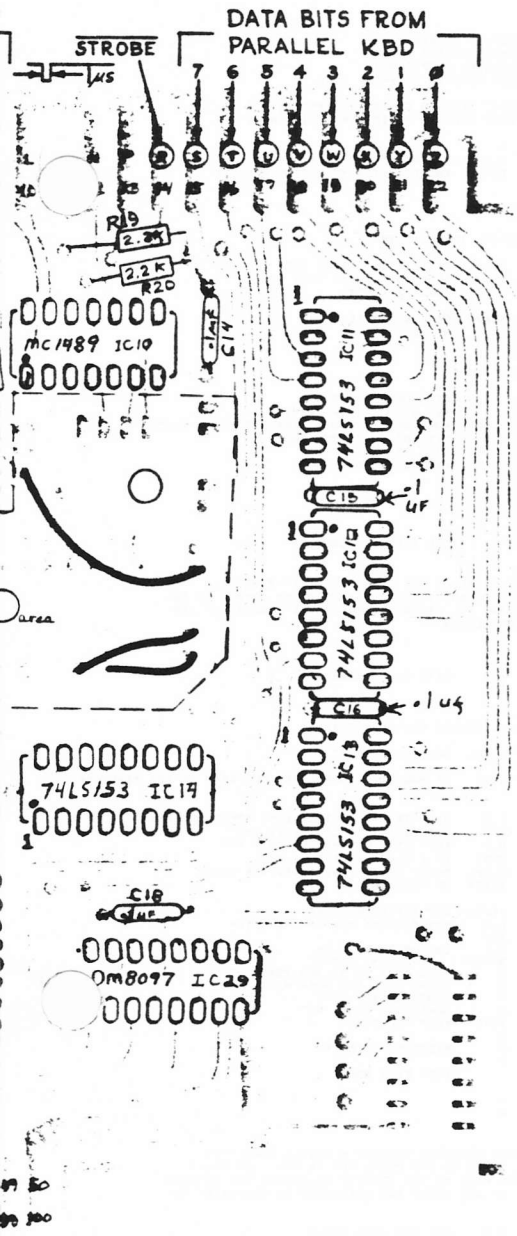
How to Jumper the 3P+S for Use with CUTER

Figure A shows the 3P+S jumpered for use with the CUTER operating system with a parallel keyboard and ASR-33 teletype.

The teletype is accessed as the serial pseudoport, PORT #1.

The keyboard is the normal pseudoport #Ø input device. Note that keyboard strobe signal is high until a key is pressed, then goes low for approximately 1 microsecond. If the pulse width of the strobe is greate tha the time it takes to process the k board information (about 20 microseconds), the system will not wor reliably.

Vern Mu



Software Techniques: The Care and Feeding of Cassette Tape Systems

Dropped bits! Yes, the symptoms of a malnourished cassettte system can be pretty ugly, but they can be avoided by paying proper attention to all three basic elements of your setup — the cassette, the tape machine, and the most important of all, the human element. Follow these tips, and you'll find that the handling and maintenance routine for cassette system is fairly simple — bu ch out if it isn't observed!

he cassette. Blank cassettes range in price from under 50¢ to well over $2. The more expensive ones have better **audio** fidelity, but that may or may not be relevant to your needs for recording digital data. Only trial and error will give you the experience for selecting the most suitable tape, so run some tests using inexpensive cassettes; if cost is no object, you may want to try some of the more expensive ones as well. Record a few long files (program or data) and then check for read errors. (Pat Tuell's program is designed for just this.) If you have friends with experience in recording programs, by all means ask them for brand recommendations.

Length is something to consider too. A C40 cassette provides about 20 minutes of recording time per side, and at 1200 baud (120 characters per second), each minute is about 7K bytes. Think about it. Are you likely to record 140K bytes of data? If not, a C30 cassette (15 minutes a side) is quite adequate and less expensive.

Finally, a word about care. A cassette appears to be durable and impervious to the elements — quite the contrary. They can be crushed or broken, and they seem to act like a magnet for ashes, dirt and liquids. Always keep them in their protective plastic covers when they're not in use, and never leave them lying around loose.

The tape machine. Again, you may not need the high audio fidelity standards of the most expensive decks, in fact hi fi cassette units may not work as well as ordinary portable machines. You should choose a reliable, well-made machine that will keep the tape moving smoothly at a constant speed. We highly recommend the Panasonic RQ-413AS, retail list $89.95. Some features that are especially helpful (though not absolutely necessary) for digital data recording:

- An AUX recording input to facilitate hookups
- A tone control
- A digital counter to help you locate files on the tape
- A review feature for fast forward or rewind while you're listening to the tape (again, to facilitate finding files).

You. It's not just the equipment and the tape that make a good cassette memory system, but what you do and how you do it. Here are some ideas to help you get the most from your system.

After you've recorded a tape, set the digital counter to zero and issue a CAT command (tape catalog — SOLOS/CUTER command). As each file header is displayed, make a note of the reading on the digital counter, the exact file name, load address and file length. Label the cassette with this information. Then later you won't have to wonder where each file begins, where it will normally be loaded, or how much memory is required.

If you're putting more than one file on a tape side, save a special file (we use the name END for it) at the end of the last one. It will let you know when you've gone past the recorded portion in reading back.

When you're recording data, make sure that the volume control is set at the proper level — about 2/3 of full volume on many portable machines, but you should experiment to find the best setting for yours. If the volume is too high, the signal may be distorted; but believe it or not, one of the most common causes of playback problems is simply not having the volume control turned up. If you have a tone control, it should be adjusted for maximum frequency response.

Take good care of your machine. Clean the tape heads every so often, because dirty heads are a good way to introduce random read errors. Make sure all cables are properly connected — a common source of problems.

And finally, make sure that the machine is set to "Record" when you're saving programs or data onto tape. Sounds elementary, but everybody has had the experience of forgetting and kicking himself for it at least once!

SOLOS Unveiled

Printed below is the entire 2048 byte SOLOS program which is the standard operating system for a Sol computer and its peripherals.

In this listing you will find a useful library of pre-written routines for developing your own programs. Included are input/output routines for all the ports, load and save routines for cassette tape, and string processing routines.

Furthermore, this listing is the best way we know to demystify SOLOS and give you the inside scoop on how it really works. The program is well written (if we don't say so ourselves), highly compact, and provides a standardized software and hardware interface between all of the Sol's internal circuitry and many external peripheral devices. We hope you will find the programming techniques it incorporates valuable.

```
                    **      PROGRAM DEVELOPMENT SYSTEM      **

                            SOFTWARE TECHNOLOGY CORP.
SOLOS(TM)    77-03-27       P.O. BOX 5260
COPYRIGHT (C) 1977          SAN MATEO, CA  94402


                    9999        COPY    SOLOS1/1
                    0002 *
                    0003 *
                    0004 *
                    0005 *
                    0006 *                    *****               *
                    0007 *                   *                    *
                    0008 *                   * * *       **       *
                    0009 *                       *       * *      *
                    0010 *                   *****       **       *
                    0011 *
                    0012 *                      SYSTEM SOLFTWARE
                    0013 *
                    0014 *
                    0015 *
                    0016 *     NOTE:    CONSOL, SOLOS AND SOLED ARE REGISTERED
                    0017 *              TRADEMARKS OF:
                    0018 *
                    0019 *                      PROCESSOR TECHNOLOGY CORP.
                    0020 *                          EMERYVILLE, CALIF
                    0021 *
                    0022 *
                    0023 *
                    0024 *                      -----       SOLOS       -----
                    0025 *
                    0026 *
                    0027 *      VERSION  1.3
                    0028 *      RELEASE  77-03-27
                    0029 *
                    0030 *
                    0031 *        THIS 2048 BYTE PROGRAM IS THE STANDARD Sol STAND
                    0032 *    ALONE OPERATING SYSTEM.  IT IS CONFIGURED TO OPTIMIZE
                    0033 *    THE CONVENIENCE AND POWER OF THE Sol-20 AND ONE OR TWO
                    0034 *    CASSETTE RECORDERS IN STAND ALONE COMPUTER APPLICATIONS.
                    0035 *
                    0036 *
                    0037 *
                    0038 *     AUTO-STARTUP CODE
                    0039 *
C000 00             0040 START  DB      0       FOUR PHASE WONDER
C001 C3 AF C1       0041 INIT   JMP     STRTA   SYSTEM RESTART ENTRY POINT
                    0042 *
                    0043 *
                    0044 *                      ENTRY POINTS
                    0045 *
                    0046 *        THESE JUMP POINTS ARE PROVIDED TO ALLOW COMMON ENTRY
                    0047 *    LOCATIONS FOR ALL VERSIONS OF SOLOS.  THEY ARE USED
                    0048 *    EXTENSIVLY BY Sol SYSTEM PROGRAMS AND IT IS RECOMMENDED
                    0049 *    THAT USER ROUTINES ACCESS SOLOS THROUGH THESE POINTS.
                    0050 *
C004 C3 C9 C1       0051 RETRN  JMP     COMND   RETURN TO SYSTEM ENTRY POINT
C007 C3 E0 C5       0052 FOPEN  JMP     ROPEN   FILE OPEN ENTRY
C00A C3 03 C6       0053 FCLOS  JMP     PCLOS   FILE CLOSE ENTRY
C00D C3 46 C6       0054 RDBYT  JMP     RTBYT   CASSETTE READ BYTE ENTRY
C010 C3 83 C6       0055 WRBYT  JMP     WTBYT   CASSETTE WRITE BYTE ENTRY
C013 C3 CB C6       0056 RDBLK  JMP     RTAPE   CASSETTE READ BLOCK ENTRY
C016 C3 7F C7       0057 WRBLK  JMP     WTAPE   CASSETTE WRITE BLOCK ENTRY
                    0058 *
                    0059 *
                    0060 *          SYSTEM I/O ENTRY POINTS
                    0061 *
                    0062 *    THESE ROUTINES PERFORM SYSTEM I/O
                    0063 *    THERE ARE TWO ENTRY TYPES:
                    0064 *     SINP/SOUT       REG "A" WILL BE SET TO THE STANDARD
                    0065 *                     SYSTEM PSEUDO PORT.
                    0066 *     AINP/AOUT       REG "A" MUST BE SET BY THE USER AND WILL
                    0067 *                     SPECIFY THE DESIRED PSEUDO PORT.
                    0068 *
                    0069 *    THE FOLLOWING ARE THE PSEUDO PORTS:
                    0070 *      PORT    DESCRIPTION
                    0071 *      ----    -----------------------------------------
                    0072 *       0      KEYBOARD WHEN INPUT, AND VDM WHEN OUTPUT
                    0073 *       1      SERIAL
                    0074 *       2      PARALLEL
                    0075 *       3      USER DEFINED
                    0076 *
C019 3A 07 C8       0077 SOUT   LDA     OPORT   SOUT ENTRY POINT
C01C C3 3B C0       0078 AOUT   JMP     OUTPR   AOUT ENTRY POINT
C01F 3A 06 C8       0079 SINP   LDA     IPORT   SINP ENTRY POINT
     C022           0080 AINP   EQU     $       AINP ENTRY POINT
                    0081 * * * * * * * * * * END OF SYSTEM ENTRY POINTS * * * *
C022 E5             0082        PUSH    H       THIS IS ACTUALLY AINP
C023 21 9A C2       0083        LXI     H,ITAB
                    0084 *
                    0085 *
                    0086 *     THIS ROUTINE PROCESSES THE I/O REQUESTS BY DISPATCHING
                    0087 *    TO THE DRIVER REQUESTED IN REGISTER "A".  ON ENTRY HL
                    0088 *    HAVE THE PROPER DISPATCH TABLE.
                    0089 *
C026 E6 03          0090 IOPRC  ANI     3       KEEP REGISTER "A" TO FOUR VALUES
C028 07             0091        RLC     .       COMPUTE ENTRY ADDRESS
C029 85             0092        ADD     L
C02A 6F             0093        MOV     L,A     WE HAVE ADDRESS
C02B C3 27 C2       0094        JMP     DISPT   DISPATCH TO IT
                    0095 *
                    0096 *
                    0097 *
                    0098 *
                    0099 *
                    0100 *                -----  Sol SYSTEM I/O ROUTINES  -----
                    0101 *
                    0102 *
                    0103 *     THIS ROUTINE IS A MODEL OF ALL INPUT ROUTINES WITHIN
                    0104 *    SOLOS.  EACH ROUTINE FIRST TESTS THE STATUS INPUT FOR
                    0105 *    DATA AVAILABLE.  IF NO CHARACTER HAS BEEN RECEIVED THE
                    0106 *    ROUTINE RETURNS WITH THE ZERO FLAG SET.  OTHERWISE THE
                    0107 *    CHARACTER IS INPUT AND A RETURN MADE WITH THE CHARACTER
                    0108 *    IN THE ACCUMULATOR AND THE ZERO FLAG RESET.
                    0109 *
                    0110 *
                    0111 *           KEYBOARD INPUT DRIVER
                    0112 *
C02E DB FA          0113 KSTAT  IN      STAPT   GET STATUS WORD
C030 2F             0114        CMA     .       INVERT IT FOR PROPER RETURN
C031 E6 01          0115        ANI     KDR     TEST KEYBOARD BIT
C033 C8             0116        RZ      .       ZERO IF NO CHARACTER RECEIVED
                    0117 *
C034 DB FC          0118        IN      KDATA   GET CHARACTER
C036 C9             0119        RET     .       GO BACK WITH IT
                    0120 *
                    0121 *
                    0122 *     THIS JUMP IS PART OF THE AUTO START UP CODE
                    0123 *
C                   0124        DB      0       VERIFY ADDR=C037 SO NEXT INSTRUCTION IS AT C038
C     3 01 C0       0125        JMP     INIT
                    0126 *
                    0127 *
                    0128 *              JUMP TABLE OUTPUT ROUTINES
                    0129 *
                    0130 *     THIS ROUTINE SETS UP THE DISPATCH TABLE FOR OUTPUT
                    0131 * ROUTINES.  THE CHARACTER FOR OUTPUT IS IN REGISTER "B".
                    0132 * OUTPUT IS MADE TO THE DRIVER POINTED TO BY THE REGISTER
                    0133 * "A".  THE DEVICE DRIVERS ARE DEFINED AS FOLLOWS:
                    0134 *
                    0135 *
                    0136 *       0 - DISPLAY SCREEN
                    0137 *       1 - SERIAL OUTPUT PORT
                    0138 *       2 - PARALLEL OUTPUT PORT
                    0139 *       3 - USER DEFINED OR ERROR FLAG
                    0140 *
                    0141 * ENTRY AT:  SOUT SELECTS CURRENT OUTPUT DEVICE
                    0142 *             AOUT SELECTS DEVICE IN REGISTER "A"
                    0143 *
C03B E5             0144 OUTPR  PUSH    H
C03C 21 92 C2       0145        LXI     H,OTAB  POINT TO OUTPUT TABLE
C03F C3 26 C0       0146        JMP     IOPRC   AND DISPATCH TO OUTPUT ROUTINE
                    0147 *
                    0148 *
                    0149 *
                    0150 *
                    0151 *              SERIAL INPUT DRIVER
                    0152 *
C042 DB F8          0153 SSTAT  IN      SERST   GET SERIAL STATUS WORD
C044 E6 40          0154        ANI     SDR     TEST FOR SERIAL DATA READY
C046 C8             0155        RZ      .       FLAGS ARE SET .
                    0156 *
C047 DB F9          0157        IN      SDATA   GET DATA BYTE
C049 C9             0158        RET     .       WE HAVE IT
                    0159 *
                    0160 *
                    0161 *     SERIAL DATA OUTPUT
                    0162 *
C04A DB F8          0163 SDROT  IN      SERST   GET PORT STATUS
C04C 17             0164        RAL     .       PUT HIGH BIT IN CARRY
C04D D2 4A C0       0165        JNC     SDROT   LOOP UNTIL TRANSMITTER BUFFER IS EMPTY
C050 78             0166        MOV     A,B     GET THE CHARACTER BACK
C051 D3 F9          0167        OUT     SDATA   SEND IT OUT
C053 C9             0168        RET     .       AND WE'RE DONE
                    0169 *
                    0170 *
                    0171 *
                    0172 *
                    0173 *
                    0174 *                  VIDEO DISPLAY ROUTINES
                    0175 *
                    0176 *
                    0177 *    THESE ROUTINES ALLOW FOR STANDARD VIDEO TERMINAL
                    0178 *    OPERATIONS.  ON ENTRY, THE CHARACTER FOR OUTPUT IS IN
                    0179 *    REGISTER B AND ALL REGISTERS EXCEPT "A" AND FLAGS ARE
                    0180 *    UNALTERED ON RETURN.
                    0181 *
                    0182 *
                    0183 *
C054 E5             0184 VDMOT  PUSH    H       SAVE MOST REGISTERS
C055 D5             0185        PUSH    D
C056 C5             0186        PUSH    B
                    0187 *
                    0188 *    TEST IF ESC SEQUENCE HAS BEEN STARTED
                    0189 *
C057 3A 0C C8       0190        LDA     ESCFL   GET ESCAPE FLAG
C05A B7             0191        ORA     A
C05B C2 5F C1       0192        JNZ     ESCS    IF NON ZERO GO PROCESS THE REST OF THE SEQUENCE
                    0193 *
                    0194 *
C05E 78             0195 CHPCK  MOV     A,B     SAVE IN B...STRIP PARITY BEFORE SCREEN!
C05F E6 7F          0196        ANI     7FH     CLR PARITY TO LOCATE IN TBL
C061 47             0197        MOV     B,A     KEEP IT W/OUT PARITY IN B TOO
C062 CA 7C C0       0198        JZ      GOBK    DO A QUICK EXIT IF A NULL
C065 21 73 C2       0199        LXI     H,TBL   POINT TO SPECIAL CHARACTER TABLE
C068 CD B2 C0       0200        CALL    TSRCH   GO PROCESS
                    0201 *
C06B CD 1C C1       0202 GOBACK CALL    VDADD   GET SCREEN ADDRESS
C06E 7E             0203        MOV     A,M     GET PRESENT CURSOR CHARACTER
C06F F6 80          0204        ORI     80H
C071 77             0205        MOV     M,A     CURSOR IS BACK ON
C072 2A 0A C8       0206        LHLD    SPEED-1  GET DELAY SPEED
C075 2C             0207        INR     L       MAKE SURE IT IS NON-ZERO
C076 AF             0208        XRA     A       DELAY WILL END WHEN H=0
C077 2B             0209 TIMER  DCX     H       TIMER DELAYS HERE
C078 BC             0210        CMP     H       DONE WITH DELAY YET
C079 C2 77 C0       0211        JNZ     TIMER   KEEP DELAYING
C07C C1             0212 GOBK   POP     B
C07D D1             0213        POP     D       RESTORE REGISTERS
C07E E1             0214        POP     H
C07F C9             0215        RET     .       EXIT FROM VDMOT
                    0216 *
C080 23             0217 NEXT   INX     H
C081 23             0218        INX     H
                    0219 *
                    0220 *
                    0221 *     THIS ROUTINE SEARCHES THROUGH A SINGLE CHARACTER
                    0222 *    TABLE FOR A MATCH TO THE CHARACTER IN "B".  IF FOUND
                    0223 *    A DISPATCH IS MADE TO THE ADDRESS FOLLOWING THE MATCHED
                    0224 *    CHARACTER.  IF NOT FOUND THE CHARACTER IS DISPLAYED ON
                    0225 *    THE MONITOR.
                    0226 *
C082 7E             0227 TSRCH  MOV     A,M     GET CHR FROM TABLE
C083 B7             0228        ORA     A
C084 CA 9A C0       0229        JZ      CHAR    ZERO IS THE LAST
C087 B8             0230        CMP     B       TEST THE CHR
C088 23             0231        INX     H       POINT FORWARD
C089 C2 80 C0       0232        JNZ     NEXT
C08C E5             0233        PUSH    H       FOUND ONE...SAVE ADDRESS
C08D CD 36 C1       0234        CALL    CREM    REMOVE CURSOR
C090 E3             0235        XTHL    .       GET DISPATCH ADDRESS TO HL
C091 C3 27 C2       0236        JMP     DISPT   DISPATCH NOW
                    0237 *
                    0238 *      PUT CHARACTER TO SCREEN
                    0239 *
C094 78             0240 CHAR   MOV     A,B     GET CHARACTER
C095 FE 7F          0241        CPI     7FH     IS IT A DEL?
C097 C8             0242        RZ      .       GO BACK IF SO
                    0243 *
                    0244 *
                    0245 *
                    0246 OCHAR  EQU     $       ACTUALLY PUT CHAR TO SCREEN NOW
C098 CD 1C C1       0247        CALL    VDADD   GET SCREEN ADDRESS
C09B 70             0248        MOV     M,B     PUT CHR ON SCREEN
                    0249 *
C09C 3A 08 C8       0250        LDA     NCHAR   GET CHARACTER POSITION
C09F FE 3F          0251        CPI     63      END OF LINE?
C0A1 DA C1 C0       0252        JC      OK
C0A4 3A 09 C8       0253        LDA     LINE
C0A7 FE 0F          0254        CPI     15      END OF SCREEN?
C0A9 C2 C1 C0       0255        JNZ     OK
                    0256 *
                    0257 *     END OF SCREEN...ROLL UP ONE LINE
                    0258 *
C0AC AF             0259 SCROLL XRA     A
C0AD 32 08 C8       0260        STA     NCHAR   BACK TO FIRST CHAR POSITION
C0B0 4F             0261 SROL   MOV     C,A
C0B1 CD 23 C1       0262        CALL    VDAD    CALCULATE LINE TO BE BLANKED
C0B4 AF             0263        XRA     A
C0B5 CD FA C0       0264        CALL    CLIN1   CLEAR IT
C0B8 3A 0A C8       0265        LDA     BOT
C0BB 3C             0266        INR     A
C0BC E6 0F          0267        ANI     0FH
C0BE C3 EE C0       0268        JMP     ERAS3
                    0269 *
                    0270 *     INCREMENT LINE COUNTER IF NECESSARY
                    0271 *
C0C1 3A 08 C8       0272 OK     LDA     NCHAR   GET CHR POSITION
C0C4 3C             0273        INR     A
C0C5 E6 3F          0274        ANI     3FH     MOD 64 AND WRAP
C0C7 32 08 C8       0275        STA     NCHAR
C0CA C0             0276        RNZ     .       DIDN'T HIT END OF LINE, OK
     C0CB           0277 PDOWN  EQU     $       CURSOR DOWN ONE LINE HERE
C0CB 3A 09 C8       0278        LDA     LINE    GET THE LINE COUNT
```

```
CDCE 3C          0279          INR     A
CDCF E6 0F       0280 CURSC    ANI     0FH     MOD 15 INCREMENT
CDD1 32 09 C8    0281 CUR      STA     LINE    STORE THE NEW
CDD4 C9          0282          RET
                 0283 *
                 0284 *    ERASE SCREEN
                 0285 *
C0:      00 CC   0286 PERSE    LXI     H,VDMEM  POINT TO SCREEN
C01      \0      0287          MVI     M,80H+' '  THIS IS THE CURSOR
                 0288 *
C0D              0289          INX     H       BUMP 1ST
                 0290 ERAS1    EQU     $       LOOPS HERE TO ERASE SCREEN
C0DB             0291          MVI     M,' '   BLANK IT OUT
CDDD 23          0292          INX     H       NEXT
CDDE 7C          0293          MOV     A,H     SEE IF END OF SCREEN YET
CDDF FE D0       0294          CPI     0D0H    ?
CDE1 DA DB C0    0295          JC      ERAS1   NO--KEEP BLANKING
CDE4 37          0296          STC     .       CARRY WILL SAY COMPLETE ERASE
                 0297 *
CDE5 3E 00       0298 PHOME    MVI     A,0     RESET CURSOR--CARRY=ERASE, ELSE HOME
CDE7 32 09 C8    0299          STA     LINE    ZERO LINE
CDEA 32 08 C8    0300          STA     WCHAR   LEFT SIDE OF SCREEN
CDED D0          0301          RNC     .       IF NO CARRY, WE ARE DONE WITH HOME
                 0302 *
CDEE D3 FE       0303 ERAS3    OUT     DSTAT   RESET SCROLL PARAMETERS
CDF0 32 0A C8    0304          STA     BOT     BEGINNING OF TEXT OFFSET
CDF3 C9          0305          RET
                 0306 *
                 0307 *
CDF4 CD 1C C1    0308 CLINE    CALL    VDADD   GET CURRENT SCREEN ADDRESS
CDF7 3A 08 C8    0309          LDA     WCHAR   CURRENT CURSOR POSITION
CDFA FE 40       0310 CLIN1    CPI     64      NO MORE THAN 63
CDFC D0          0311          RNC     .       ALL DONE
CDFD 36 20       0312          MVI     M,' '   ALL SPACED OUT
CDFF 23          0313          INX     H
CE00 3C          0314          INR     A
CE01 C3 FA CD    0315          JMP     CLIN1   LOOP TO END OF LINE
                 0316 *
                 0317 *
                 0318 *  ROUTINE TO MOVE THE CURSOR UP ONE LINE
                 0319 *
CE04 3A 09 C8    0320 PUP      LDA     LINE    GET LINE COUNT
CE07 3D          0321          DCR     A
CE08 C3 CF C0    0322          JMP     CURSC   MERGE TO HANDLE CURSOR
                 0323 *
                 0324 *  MOVE CURSOR LEFT ONE POSITION
                 0325 *
CE0B 3A 08 C8    0326 PLEFT    LDA     WCHAR
CE0E 3D          0327          DCR     A
     C10F        0328 PCUR     EQU     $       CURSOR ON SAME LINE
C10F E6 3F       0329          ANI     3FH     LET CURSOR WRAP
C111 32 08 C8    0330          STA     WCHAR   UPDATED CURSOR
C114 C9          0331          RET
                 0332 *
                 0333 *     CURSOR RIGHT ONE POSITION
                 0334 *
C115 3A 08 C8    0335 PRIT     LDA     WCHAR
C118 3C          0336          INR     A
C119 C3 0F C1    0337          JMP     PCUR
                 0338 *
                 0339 *   ROUTINE TO CALCULATE SCREEN ADDRESS
                 0340 *
                 0341 *   ENTRY AT:    RETURNS:
                 0342 *
                 0343 *         VDADD   CURRENT SCREEN ADDRESS
                 0344 *         VDAD2   ADDRESS OF CURRENT LINE, CHAR 'C'
                 0345 *         VDAD    LINE 'A', CHARACTER POSITION 'C'
                 0346 *
C11C 3A 08 C8    0347 VDADD    LDA     WCHAR   GET CHARACTER POSITION
C11              0348          MOV     C,A     'C' KEEPS IT
C1               0349 VDAD2    LDA     LINE    LINE POSITION
C1               0350 VDAD     MOV     L,A     INTO 'L'
C12      C8      0351          LDA     BOT     GET TEXT OFFSET
C12              0352          ADD     L       ADD IT TO THE LINE POSITION
C12              0353          RRC     .       TIMES TWO
C129             0354          RRC     .       MAKES FOUR
C12A 6F          0355          MOV     L,A     L HAS IT
C12B E6 03       0356          ANI     3       MOD THREE FOR LATER
C12D C6 CC       0357          ADI     <VDMEM  LOW SCREEN OFFSET
C12F 67          0358          MOV     H,A     NOW H IS DONE
C130 7D          0359          MOV     A,L     TWIST L'S ARM
C131 E6 C0       0360          ANI     0C0H
C133 81          0361          ADD     C
C134 6F          0362          MOV     L,A
C135 C9          0363          RET     .       H & L ARE NOW PERVERTED
                 0364 *
                 0365 *    ROUTINE TO REMOVE CURSOR
                 0366 *
C136 CD 1C C1    0367 CREM     CALL    VDADD   GET CURRENT SCREEN ADDRESS
C139 7E          0368          MOV     A,M
C13A E6 7F       0369          ANI     7FH     STRIP OFF THE CURSOR
C13C 77          0370          MOV     M,A
C13D C9          0371          RET
                 0372 *
                 0373 *     ROUTINE TO BACKSPACE
                 0374 *
C13E CD 0B C1    0375 PBACK    CALL    PLEFT
C141 CD 1C C1    0376          CALL    VDADD   GET SCREEN ADDRESS
C144 36 20       0377          MVI     M,' '   PUT A BLANK THERE
C146 C9          0378          RET
                 0379 *
                 0380 *     ROUTINE TO PROCESS A CARRIAGE RETURN
                 0381 *
C147 CD FA C0    0382 PCR      CALL    CLINE   CLEAR FROM CURRENT CURSOR TO END OF LINE
                 0383 *   REG A COMES BACK W/64 AND WILL BE CLEARED BY PCUR
C14A C3 0F C1    0384          JMP     PCUR    AND STORE THE NEW VALUE
                 0385 *
                 0386 *   ROUTINE TO PROCESS LINEFEED
                 0387 *
C14D 3A 09 C8    0388 PLF      LDA     LINE    GET LINE COUNT
C150 3C          0389          INR     A
C151 E6 0F       0390          ANI     15      SEE IF IT WRAPPED AROUND
C153 C2 D1 C0    0391          JNZ     CUR     NO--NO NEED TO SCROLL
C156 C3 B0 C0    0392          JMP     SROL    YES--THEN SCROLL
                 0393 *
                 0394 *     SET ESCAPE PROCESS FLAG
                 0395 *
C159 3E FF       0396 PESC     MVI     A,-1
C15B 32 0C C8    0397          STA     ESCFL   SET FLAG
C15E C9          0398          RET
                 0399 *
                 0400 *      PROCESS ESCAPE SEQUENCE
                 0401 *
C15F CD 36 C1    0402 ESCS     CALL    CREM    REMOVE CURSOR
C162 CD 68 C1    0403          CALL    ESCSP   PROCESS THE NEXT PART OF SEQUENCE
C165 C3 6B C0    0404          JMP     GOBACK
                 0405 *
C168 3A 0C C8    0406 ESCSP    LDA     ESCFL   GET ESCAPE FLAG
C16B FE FF       0407          CPI     -1      TEST FLAG
C16D CA 90 C1    0408          JZ      SECOND
                 0409 *
                 0410 *  PROCESS THIRD CHR OF ESC SEQUENCE
                 0411 *
C17      C8      0412          LXI     H,ESCFL
C173             0413          MVI     M,0     NO MORE PARTS TO THE SEQUENCE
C175             0414          CPI     2
C177     C1      0415          JC      SETX    SET X  IF IS ONE
C17A CA 8C C1    0416          JZ      SETY    SET Y  IF IS TWO
C17D FE 08       0417          CPI     8
C17F CA 98 C5    0418          JZ      STSPD   SET NEW DISPLAY SPEED IF "8"
C182 FE 09       0419          CPI     9
C184 DA 98 C0    0420          JC      OCHAR   PUT IT ON THE SCREEN
C187 C0          0421          RNZ
                 0422 *
                 0423 *  TAB ABSOLUTE TO VALUE IN REG B
                 0424 *
C188 78          0425 SETX     MOV     A,B     GET CHARACTER
C189 C3 0F C1    0426          JMP     PCUR
                 0427 *
                 0428 *  SET CURSOR TO LINE "B"
                 0429 *
C18C 78          0430 SETY     MOV     A,B
C18D C3 CF C0    0431          JMP     CURSC
                 0432 *
                 0433 *
                 0434 *   PROCESS SECOND CHR OF ESC SEQUENCE
                 0435 *
C190 78          0436 SECOND   MOV     A,B     GET WHICH
C191 FE 03       0437          CPI     3
C193 CA A6 C1    0438          JZ      CURET   RETURN CURSOR PARAMETERS
C196 FE 0A       0439          CPI     A
C198 C2 A2 C1    0440          JNZ     ARET2
                 0441 *
                 0442 *   ESC <A>  RETURN ABSOLUTE SCREEN ADDRESS
                 0443 *
C19A AA          0444 ARET     MOV     B,H
C19C 4D          0445          MOV     C,L     PRESENT SCREEN ADDRESS TO BC FOR RETURN
                 0446 *
C19D E1          0447 ARET1    POP     H       RETURN ADDRESS
C19E D1          0448          POP     D       OLD B
C19F C5          0449          PUSH    B
C1A0 E5          0450          PUSH    H
C1A1 AF          0451          XRA     A
C1A2 32 0C C8    0452 ARET2    STA     ESCFL
C1A5 C9          0453          RET
                 0454 *
                 0455 *
                 0456 *     RETURN PRESENT SCREEN PARAMETERS IN BC
                 0457 *
C1A6 21 08 C8    0458 CURET    LXI     H,WCHAR
C1A9 46          0459          MOV     B,M     CHARACTER POSITION
C1AA 23          0460          INX     H
C1AB 4E          0461          MOV     C,M     LINE POSITION
C1AC C3 9D C1    0462          JMP     ARET1
                 0463 *
                 0464 *
                 0465 *                START UP SYSTEM
                 0466 *
                 0467 *   CLEAR SCREEN AND THE FIRST 256 BYTES OF GLOBAL RAM
                 0468 *  THEN ENTER THE COMMAND MODE.
                 0469 *
C1AF AF          0470 STRTA    XRA     A
C1B0 4F          0471          MOV     C,A
C1B1 21 00 C8    0472          LXI     H,SYSRAM  CLEAR THE FIRST PAGE
                 0473 *
C1B4 77          0474 CLERA    MOV     M,A
C1B5 23          0475          INX     H
C1B6 0C          0476          INR     C
C1B7 C2 B4 C1    0477          JNZ     CLERA
                 0478 *
C1BA 31 FF CB    0479          LXI     SP,SYSTP  SET UP THE STACK FOR CALL
C1BD CD D5 C0    0480          CALL    PERSE
C1C0 AF          0481 COMN1    XRA     A
C1C1 D3 FA       0482          OUT     STAPT  BE SURE TAPES ARE OFF
C1C3 32 07 C8    0483          STA     OPORT
C1C6 32 06 C8    0484          STA     IPORT
                 0485 *
                 0486 *
                 0487 *
                 0488 *              ---  COMMAND MODE  ---
                 0489 *
                 0490 *
                 0491 *   THIS ROUTINE GETS AND PROCESSES COMMANDS
                 0492 *
C1C9 31 FF CB    0493 COMND    LXI     SP,SYSTP  SET STACK POINTER
C1CC 3A 07 C8    0494          LDA     OPORT  GET PORT
C1CF F5          0495          PUSH    PSW
C1D0 AF          0496          XRA     A
C1D1 32 07 C8    0497          STA     OPORT  FORCE SCREEN OPERATIONS
C1D4 CD F1 C2    0498          CALL    PROMPT  PUT PROMPT ON SCREEN
C1D7 CD E4 C1    0499          CALL    GCLIN  GET COMMAND LINE
C1DA F1          0500          POP     PSW
C1DB 32 07 C8    0501          STA     OPORT  RESTORE DEFAULT PORT
C1DE CD 05 C2    0502          CALL    COPRC  PROCESS THE LINE
C1E1 C3 C9 C1    0503          JMP     COMND  OVER AND OVER
                 0504 *
                 0505 *
                 0506 *
                 0507 *  THIS ROUTINE READS A COMMAND LINE FROM THE SYSTEM
                 0508 *  KEYBOARD
                 0509 *
                 0510 *  C/R   TERMINATES THE SEQUENCE ERASING ALL CHARS TO THE
                 0511 *        RIGHT OF THE CURSOR
                 0512 *  L/F   TERMINATES THE SEQUENCE
                 0513 *  MODE  RESTARTS THE COMMAND LINE.
                 0514 *
C1E4 CD 1F C0    0515 GCLIN    CALL    SINP    READ INPUT DEVICE
C1E7 CA E4 C1    0516          JZ      GCLIN
C1EA E6 7F       0517          ANI     7FH     CLEAR PARITY BIT
C1EC CA C0 C1    0518          JZ      COMN1  THIS WAS A MODE (OR EVEN CTL-@)
C1EF A7          0519          MOV     B,A
C1F0 FE 0D       0520          CPI     CR      CARRIAGE RETURN
C1F2 CA F4 C0    0521          JZ      CLINE  YES--DONE WITH LINE
C1F5 FE 0A       0522          CPI     LF      LINE FEED
C1F7 C8          0523          RZ      .       YES--DONE WITH LINE, LEAVE AS IS
C1F8 FE 7F       0524          CPI     7FH     DELETE CHR?
C1FA C2 FF C1    0525          JNZ     CONT
C1FD 06 5F       0526          MVI     B,BACKS  REPLACE IT
                 0527 *
C1FF CD 19 C0    0528 CONT     CALL    SOUT
C202 C3 E4 C1    0529          JMP     GCLIN
                 0530 *
                 0531 *
                 0532 *      FIND AND PROCESS COMMAND
                 0533 *
C205 CD 36 C1    0534 COPRC    CALL    CREM    REMOVE THE CURSOR
C208 0E 01       0535          MVI     C,1     SET FOR CHARACTER POSITION
C20A CD 20 C1    0536          CALL    VDAD2  GET SCREEN ADDRESS
C20D EB          0537          XCHG
C20E 21 00 C0    0538          LXI     H,START  MAKE SURE HL PT TO SOLOS START
C211 E5          0539          PUSH    H       SAVE IT FOR LATER DISPT
C212 CD 2E C3    0540          CALL    SCHR    SCAN PAST BLANKS
C215 CA 80 C4    0541          JZ      ERR1    NO COMMAND?
C218 EB          0542          XCHG    .       HL HAS FIRST CHR
                 0543 *
C219 11 4A C2    0544          LXI     D,COMTAB  POINT TO COMMAND TABLE
C21C CD 31 C2    0545          CALL    FDCOM  SEE IF IN PRIMARY COMMAND TABLE
C21F CC 2E C2    0546          CZ      FDCOU  IF NOT, TRY CUSTOM TABLE NEXT
     C222        0547 DISPD    EQU     $       HERE TO SEE IF ERROR OR DISP
C222 CA 81 C4    0548          JZ      ERR2    NOT VALID, ERROR
C225 13          0549          INX     D       BUMP TO PTR OF RTN
C226 EB          0550          XCHG    .       HL PT TO RTN ADDR
                 0551 *
                 0552 *
                 0553 *  THIS IS THE DISPATCH ROUTINE.
                 0554 *  HL PT TO RTN ADDRESS, HL WILL BE RESTORED FM STACK
                 0555 *  SO THAT HL ARE RESTORED BEFORE DISPATCH.
                 0556 *
     C227        0557 DISPT    EQU     $       OFF TO A ROUTINE
C227 7E          0558          MOV     A,M     LO ADDR
C228 23          0559          INX     H
C229 66          0560          MOV     H,M     HI ADDR
C22A 6F          0561          MOV     L,A     HL NOW COMPLETE
     C22B        0562 DISP1    EQU     $       HERE TO GO OFF TO HL
C22B E3          0563          XTHL    .       XCHG HL W/HL ON STACK
C22C 7D          0564          MOV     A,L     ALSO COPY HERE FOR SETS
C22D C9          0565          RET     .       AND GO OFF TO THE RTN
```

```
                        0566 *
                        0567 *
                        0568 *   THIS ROUTINE SEARCHES THROUGH A TABLE, POINTED TO
                        0569 *  BY 'DE', FOR A DOUBLE CHARACTER MATCH OF THE 'HL'
                        0570 *  MEMORY CONTENT.  IF NO MATCH IS FOUND THE SCAN ENDS
                        0571 *  WITH HL POINTING TO ORIGINAL VALUE AND ZERO FLAG SET.
                        0572 *
C22E 11 3C C8           0573 FDCOU  LXI    D,CUTAB  HERE TO SCAN CUSTOM TBL ONLY
                        0574 *
C231 1A                 0575 FDCOM  LDAX   D
C232 B7                 0576        ORA    A       TEST FOR TABLE END
C233 C8                 0577        RZ     .       NOT FOUND..COMMAND ERROR
C234 E5                 0578        PUSH   H       SAVE START OF SCAN ADDRESS
C235 BE                 0579        CMP    M       TEST FIRST CHR
C236 13                 0580        INX    D
C237 C2 43 C2           0581        JNZ    NCOM
                        0582 *
C23A 23                 0583        INX    H
C23B 1A                 0584        LDAX   D
C23C BE                 0585        CMP    M       NOW SECOND CHARACTER
C23D C2 43 C2           0586        JNZ    NCOM    GOODNESS
                        0587 *
C240 E1                 0588        POP    H       RESTORE ORIGINAL SCAN ADDR
C241 B7                 0589        ORA    A       SET NON-ZERO FLAG SAYING FOUND
C242 C9                 0590        RET    .       WITH NON-ZERO SET
                        0591 *
                        0592 *
C243 13                 0593 NCOM   INX    D       GO TO NEXT ENTRY
C244 13                 0594        INX    D
C245 13                 0595        INX    D
C246 E1                 0596        POP    H       GET BACK ORIGINAL ADDRESS
C247 C3 31 C2           0597        JMP    FDCOM   CONTINUE SEARCH
                        0598 *
                        0599 *
                        0600 *                  COMMAND TABLE
                        0601 *
                        0602 *  THIS TABLE DESCRIBES THE VALID COMMANDS FOR SOLOS
                        0603 *
C24A 54 45              0604 COMTAB ASC    'TE'    TERMINAL MODE
C24C 67 C3              0605        DW     TERM
C24E 44 55              0606        ASC    'DU'    DUMP
C250 BF C3              0607        DW     DUMP
C252 45 4E              0608        ASC    'EN'    ENTR
C254 23 C4              0609        DW     ENTER
C256 45 58              0610        ASC    'EX'    EXEC
C258 5E C4              0611        DW     EXEC
C25A 47 45              0612        ASC    'GE'    GET A FILE
C25C A7 C4              0613        DW     TLOAD
C25E 53 41              0614        ASC    'SA'    SAVE A FILE
C260 E6 C4              0615        DW     TSAVE
C262 58 45              0616        ASC    'XE'    XEQ (EXECUTE) A FILE
C264 A6 C4              0617        DW     TXEQ
C266 43 41              0618        ASC    'CA'    CATALOG OF FILES
C268 2B C5              0619        DW     TLIST
C26A 53 45              0620        ASC    'SE'    SET COMMAND
C26C 7A C5              0621        DW     SET
C26E 43 55              0622        ASC    'CU'    CUSTOM COMMAND
C270 BD C5              0623        DW     CUSET
C272 00                 0624        DB     0       END OF TABLE MARK
                        0625 *
                        0626 *
                        0627 *                DISPLAY DRIVER COMMAND TABLE
                        0628 *
                        0629 *     THIS TABLE DEFINES THE CHARACTERS FOR SPECIAL
                        0630 *  PROCESSING.  IF THE CHARACTER IS NOT IN THE TABLE IT
                        0631 *  GOES TO THE SCREEN.
                        0632 *
C273 0B                 0633 TBL    DB     CLEAR-80H  SCREEN
C274 D5 C0              0634        DW     PERSE
C276 17                 0635        DB     UP-80H  CURSOR
C277 04 C1              0636        DW     PUP
C279 1A                 0637        DB     DOWN-80H
C27A CB C0              0638        DW     PDOWN
C27C 01                 0639        DB     LEFT-80H
C27D 0B C1              0640        DW     PLEFT
C27F 13                 0641        DB     RIGHT-80H
C280 15 C1              0642        DW     PRIT
C282 0E                 0643        DB     HOME-80H
C283 E5 C0              0644        DW     PHOME
C285 0D                 0645        DB     CR      CARRIAGE RETURN
C286 A7 C1              0646        DW     PCR
C288 0A                 0647        DB     LF      LINE FEED
C289 AD C1              0648        DW     PLF     .
C28B 5F                 0649        DB     BACKS   BACK SPACE
C28C 3E C1              0650        DW     PBACK
C28E 1B                 0651        DB     ESC     ESCAPE KEY
C28F 59 C1              0652        DW     PESC
C291 00                 0653        DB     0       END OF TABLE
                        0654 *
                        0655 *
                        0656 *            OUTPUT DEVICE TABLE
                        0657 *
C292 54 C0              0658 OTAB   DW     VDMOT   VDM DRIVER
C294 4A C0              0659        DW     SDROT   SERIAL OUTPUT
C296 E6 C2              0660        DW     PROUT   PARALLEL OUTPUT
C298 D2 C2              0661        DW     ERROT   ERROR OR USER DRIVER HANDLER
                        0662 *
                        0663 *
                        0664 *           INPUT DEVICE TABLE
                        0665 *
C29A 2E C0              0666 ITAB   DW     KSTAT   KEYBOARD INPUT
C29C 42 C0              0667        DW     SSTAT   SERIAL INPUT
C29E DD C2              0668        DW     PASTAT  PARALLEL INPUT
C2A0 CB C2              0669        DW     ERRIT   ERROR OR USER DRIVER HANDLER
                        0670 *
                        0671 *
                        0672 *        SECONDARY COMMAND TABLE FOR SET COMMAND
                        0673 *
C2A2 54 41              0674 SETAB  ASC    'TA'    SET TAPE SPEED
C2A4 8E C5              0675        DW     TASPD
C2A6 53 3D              0676        ASC    'S='    SET DISPLAY SPEED
C2A8 99 C5              0677        DW     DISPD
C2AA 49 3D              0678        ASC    'I='    SET INPUT PORT
C2AC 9D C5              0679        DW     SETIN
C2AE 4F 3D              0680        ASC    'O='    SET OUTPUT PORT
C2B0 A1 C5              0681        DW     SETOT
C2B2 4E 3D              0682        ASC    'N='    NULLS
C2B4 B5 C5              0683        DW     SETNU
C2B6 43 49              0684        ASC    'CI'    SET CUSTOM DRIVER ADDRESS
C2B8 A5 C5              0685        DW     SETCI
C2BA 43 4F              0686        ASC    'CO'    SET CUSTOM OUTPUT DRIVER ADDRESS
C2BC A9 C5              0687        DW     SETCO
C2BE 58 45              0688        ASC    'XE'    SET HEADER XEQ ADDRESS
C2C0 B1 C5              0689        DW     SETXQ
C2C2 54 59              0690        ASC    'TY'    SET HEADER TYPE
C2C4 AD C5              0691        DW     SETTY
C2C6 43 52              0692        ASC    'CR'    SET CRC TO ALLOW IGNORING OF CRC ERRORS
C2C8 B9 C5              0693        DW     SETCR
C2CA 00                 0694        DB     0       END OF TABLE MARK
                        0695 *
                        0696 *
                        0697 *  SOLOS PORT ERROR HANDLER
                        0698 *
C2CB E5                 0699 ERRIT  PUSH   H       SAVE HL ONCE AGAIN
C2CC 2A 00 C8           0700        LHLD   UIPRT   GET USER INPUT PORT ADDRESS
C2CF C3 D6 C2           0701        JMP    ERRO1   AND GO PROCESS
                        0702 *
C2D2 E5                 0703 ERROT  PUSH   H
C2D3 2A 02 C8           0704        LHLD   UOPRT   GET USER OUTPUT PORT ADDRESS
C2D6 7D                 0705 ERRO1  MOV    A,L     TEST HL FOR ZERO
C2D7 B4                 0706        ORA    H
C2D8 CA C0 C1           0707        JZ     COMM1   IF ZERO RETURN TO COMMAND MODE
C2DB E3                 0708        XTHL   .       ADDRESS TO STACK...OLD HL TO HL
C2DC C9                 0709        RET    .       GO TO THE DRIVER
                        0710 * -0-
                        9999        COPY   SOLOS2/1
                        0711 *
                        0712 *    THIS ROUTINE IS THE PARALLEL DEVICE HANDLER
                        0713 *  NO PROVISION IS MADE FOR CONTROLLING THE PORT
                        0714 *  CONTROL BIT.
                        0715 *
                        0716 *
                        0717 *      PARALLEL INPUT DRIVER
                        0718 *
C2DD DB FA              0719 PASTAT IN     STAPT
C2DF 2F                 0720        CMA    .       INVERT STATUS FLAGS
C2E0 E6 02              0721        ANI    PDR     TEST BIT
C2E2 C8                 0722        RZ     .       WITH FLAG SET
C2E3 DB FD              0723        IN     PDATA   GET DATA
C2E5 C9                 0724        RET
                        0725 *
                        0726 *     PARALLEL OUTPUT HANDLER
                        0727 *
C2E6 DB FA              0728 PROUT  IN     STAPT   GET STATUS
C2E8 E6 04              0729        ANI    PXDR    TEST IF DEVICE IS READY
C2EA C2 E6 C2           0730        JNZ    PROUT   LOOP UNTIL SO
C2ED 78                 0731        MOV    A,B
C2EE D3 FD              0732        OUT    PDATA
C2F0 C9                 0733        RET
                        0734 *
                        0735 *
                        0736 *      OUTPUT A CRLF FOLLOWED BY A PROMPT
                        0737 *
C2F1 CD F9 C2           0738 PROMPT CALL   CRLF
C2F4 06 3E              0739        MVI    B,'>'   THE PROMPT
C2F6 C3 19 C0           0740        JMP    SOUT    PUT IT ON THE SCREEN
                        0741 *
                        0742 *
C2F9 06 0A              0743 CRLF   MVI    B,LF    LINE FEED
C2FB CD 19 C0           0744        CALL   SOUT
C2FE 06 0D              0745        MVI    B,CR    CARRIAGE RETURN
C300 CD 19 C0           0746        CALL   SOUT
                        0747 *  NOW OUTPUT THE NULLS
C303 3A 10 C8           0748        LDA    NUCNT   GET DESIRED COUNT
C306 4F                 0749        MOV    C,A     STORE IN C
C307 0D                 0750 NULOT  DCR    C
C308 F8                 0751        RM     .       RETURN WHEN PAST ZERO
C309 AF                 0752        XRA    A       GET A NULL
C30A CD 1F C4           0753        CALL   OUTH
C30D C3 07 C3           0754        JMP    NULOT
                        0755 *
                        0756 *
                        0757 *   SCAN OFF OPTIONAL PARAMETER.  IF PRESENT RETURN WITH
                        0758 * VALUE IN HL AND COPY OF "L" IN "A".  IF NOT PRESENT
                        0759 * RETURN WITH A "0" IN "A" AND HL UNTOUCHED.
                        0760 *
C310 CD 1B C3           0761 PSCAN  CALL   SBLK
C313 3E 01              0762        MVI    A,1     DEFAULT VALUE
C315 C8                 0763        RZ     .       IF NONE
C316 CD A0 C3           0764        CALL   XHEX    CONVERT VALUE
C319 7D                 0765        MOV    A,L     GET LOWER HALF
C31A C9                 0766        RET
                        0767 *
                        0768 *
                        0769 *  SCAN OVER UP TO 12 CHARACTERS LOOKING FOR A BLANK
                        0770 *
C31B 0E 0C              0771 SBLK   MVI    C,12    MAXIMUM COMMAND STRING
C31D 1A                 0772 SBLK1  LDAX   D
C31E FE 20              0773        CPI    BLANK
C320 CA 2E C3           0774        JZ     SCHR    GOT A BLANK NOW SCAN PAST IT
C323 13                 0775        INX    D
C324 FE 3D              0776        CPI    '='     ALSO ALLOW AN EQUAL TO STOP US
C326 CA 2E C3           0777        JZ     SCHR    IF SO, PTR AT CHAR FOLLOWING
C329 0D                 0778        DCR    C       NO MORE THAN TWELVE
C32A C2 1D C3           0779        JNZ    SBLK1
C32D C9                 0780        RET    .       GO BACK WITH ZERO FLAG SET
                        0781 *
                        0782 *
                        0783 *    SCAN PAST UP TO 10 BLANK POSITIONS LOOKING FOR
                        0784 *  A NON BLANK CHARACTER.
                        0785 *
C32E 0E 0A              0786 SCHR   MVI    C,10    SCAN TO FIRST NON BLANK CHR WITHIN 10
C330 1A                 0787 SCHR1  LDAX   D       GET NEXT CHARACTER
C331 FE 20              0788        CPI    SPACE
C333 C0                 0789        RNZ    .       WE'RE PAST THEM
C334 13                 0790        INX    D       NEXT SCAN ADDRESS
C335 0D                 0791        DCR    C
C336 C8                 0792        RZ             COMMAND ERROR
C337 C3 30 C3           0793        JMP    SCHR1   KEEP LOOPING
                        0794 *
                        0795 *
                        0796 *    THIS ROUTINE SCANS OVER CHARACTERS, PAST BLANKS AND
                        0797 * CONVERTS THE FOLLOWING VALUE TO HEX.  ERRORS RETURN TO
                        0798 * THE ERROR HANDLER.
                        0799 *
C33A CD 1B C3           0800 SCONV  CALL   SBLK    FIND IF VALUE IS PRESENT
C33D CA 80 C4           0801        JZ     ERR1    ABORT TO ERROR IF NONE
                        0802 *
                        0803 *
                        0804 *  THIS ROUTINE CONVERTS ASCII DIGITS INTO BINARY FOLLOWIN
                        0805 * A STANDARD HEX CONVERSION.  THE SCAN STOPS WHEN AN ASCII
                        0806 * SPACE IS ENCOUNTERED.  PARAMETER ERRORS REPLACE THE ERR
                        0807 * CHARACTER ON THE SCREEN WITH A QUESTION MARK.
                        0808 *
C340 21 00 00           0809 SHEX   LXI    H,0     CLEAR H & L
C343 1A                 0810 SHE1   LDAX   D       GET CHARACTER
C344 FE 20              0811        CPI    20H     IS IT A SPACE?
C346 C8                 0812        RZ     .       IF SO
C347 FE 2F              0813        CPI    '/'     SLASH IS ALSO LEGAL
C349 C8                 0814        RZ
C34A FE 3A              0815        CPI    ':'     EVEN THE COLON IS ALLOWED
C34C C8                 0816        RZ
                        0817 *
C34D 29                 0818 HCONV  DAD    H       MAKE ROOM FOR THE NEW ONE
C34E 29                 0819        DAD    H
C34F 29                 0820        DAD    H
C350 29                 0821        DAD    H
C351 CD 5D C3           0822        CALL   HCOV1   DO THE CONVERSION
C354 D2 80 C4           0823        JNC    ERR1    NOT VALID HEXIDECIMAL VALUE
C357 85                 0824        ADD    L
C358 6F                 0825        MOV    L,A     MOVE IT IN
C359 13                 0826        INX    D       BUMP THE POINTER
C35A C3 43 C3           0827        JMP    SHE1
                        0828 *
C35D D6 30              0829 HCOV1  SUI    48      REMOVE ASCII BIAS
C35F FE 0A              0830        CPI    10
C361 D8                 0831        RC     .       IF LESS THAN 9
C362 D6 07              0832        SUI    7       IT'S A LETTER
C364 FE 10              0833        CPI    10H
C366 C9                 0834        RET    .       WITH TEST IN HAND
                        0835 *
                        0836 *
                        0837 *                  TERM COMMAND
                        0838 *
                        0839 *   THIS ROUTINE GETS CHARACTERS FROM THE SYSTEM [illegible]ARD
                        0840 *  AND OUTPUTS THEM TO THE SELECTED OUTPUT PORT.
                        0841 *  INTENDED TO CONFIGURE THE Sol AS A STANDARD V
                        0842 *  TERMINAL.  COMMAND KEYS ARE NOT OUTPUT TO THE
                        0843 *  PORT BUT ARE INTERPRETED AS DIRECT Sol COMMANDS.
                        0844 *  THE MODE COMMAND, RECEIVED BY THE KEYBOARD, PUTS THE
                        0845 *  Sol IN THE COMMAND MODE.
                        0846 *
                        0847 *
                        0848 *
```

```
                0849 TERM   CALL   PSCAN   FIND IF INPUT PARAMETER IS PRESENT
                0850        STA    IPORT   SINP WILL USE THIS DRIVER (DEFAULT IS 1)
                0851        CALL   PSCAN   NOW FOR THE OUTPUT DRIVER
                0852        STA    OPORT
                0853 *
C373 CD 2E C0   0854 TERM1  CALL   KSTAT   IS THERE ONE WAITING?
C376 CA 8B C3   0855        JZ     TIN     IF NOT
C379            0856        MOV    B,A     SAVE IT IN B
C37A      80    0857        CPI    MODE    IS IT MODE
C37C      70 C1 0858        JZ     COMN1   YES--RESET AND QUIT TERM
C37F       8 C3 0859        JC     TOUT    NON-CURSOR KEY---SEND TO TERM PORT
C382       4 C0 0860        CALL   VDMOT   PROCESS IT
C385         C3 0861        JMP    TIN
                0862 *
C388         C0 0863 TOUT   CALL   SOUT    OUTPUT IT TO THE SERIAL PORT
C38B CD 1F C0   0864 TIN    CALL   SINP    GET INPUT STATUS
C38E CA 73 C3   0865        JZ     TERM1   LOOP IF NOT
C391 E6 7F      0866        ANI    7FH     NO HIGH BITS FROM HERE
C393 CA 73 C3   0867        JZ     TERM1   A NULL IS IGNORED
C396 47         0868        MOV    B,A     IT'S OUTPUT FROM 'B'
C397 FE 1B      0869        CPI    1BH     IS IT A CONTROL CHAR TO BE IGNORED
C399 D2 B9 C3   0870        JNC    TERM2   NO--TO VDM AS IS THEN
C39C FE 0D      0871        CPI    CR      CR OR LF ARE SPECIAL CASES THOUGH
C39E CA B9 C3   0872        JZ     TERM2   AND MUST BE PASSED STD MODE TO VDM DRIVER
C3A1 FE 0A      0873        CPI    LF
C3A3 CA B9 C3   0874        JZ     TERM2
C3A6 3A 0C C8   0875        LDA    ESCFL   A CTL CHAR---ARE WE W/IN AN ESC SEQUENCE?
C3A9 B7         0876        ORA    A       IF YES, THEN OUTPUT CTL CHAR DIRECTLY TO VDM
C3AA C2 B9 C3   0877        JNZ    TERM2   WE SURE ARE, LET VDM DRIVER HANDLE IT
C3AD C5         0878        PUSH   B       SAVE THE CHAR
C3AE 06 1B      0879        MVI    B,ESC   CTL CHAR TO VDM VIA ESC SEQUENCE
C3B0 CD 54 C0   0880        CALL   VDMOT
C3B3 06 07      0881        MVI    B,7     SAY TO PUT OUT NEXT CHAR AS IS
C3B5 CD 54 C0   0882        CALL   VDMOT   ALMOST READY
C3B8 C1         0883        POP    B       RESTORE CHAR
      C3B9      0884 TERM2  EQU    $       ALL READY TO OUTPUT THE CHAR
C3B9 CD 54 C0   0885        CALL   VDMOT   PUT IT ON THE SCREEN
C3BC C3 73 C3   0886        JMP    TERM1   LOOP OVER AND OVER
                0887 *
                0888 *
                0889 *
                0890 *                      DUMP COMMAND
                0891 *
                0892 *       THIS ROUTINE DUMPS CHARACTERS FROM MEMORY TO THE
                0893 *  CURRENT OUTPUT DEVICE.  ALL VALUES ARE DISPLAYED AS
                0894 *  ASCII HEX.
                0895 *
                0896 *  THE COMMAND FORM IS AS FOLLOWS:
                0897 *
                0898 *                  DUmp  addr1  addr2
                0899 *
                0900 *    THE VALUES FROM ADDR1 TO ADDR2 ARE THEN OUTPUT TO THE
                0901 *  OUTPUT DEVICE.  IF ONLY ADDR1 IS SPECIFIED THEN THE
                0902 *  VALUE AT THAT ADDRESS IS OUTPUT.
                0903 *
C3BF CD 3A C3   0904 DUMP   CALL   SCONV  SCAN TO FIRST ADDRESS AND CONVERT IT
C3C2 E5         0905        PUSH   H      SAVE THE VALUE
C3C3 CD 10 C3   0906        CALL   PSCAN  SEE IF SECOND WAS GIVEN
C3C6 D1         0907        POP    D      GET BACK START
C3C7 EB         0908        XCHG   .      HL HAS START, DE HAS END
                0909 *
C3C8 CD F9 C2   0910 DLOOP  CALL   CRLF
C3CB CD EE C2   0911        CALL   ADOUT  OUTPUT ADDRESS
C3CE CD 06 C4   0912        CALL   BOUT   ANOTHER SPACE TO KEEP IT PRETTY
C3D1 0E 10      0913        MVI    C,16   VALUES PER LINE
                0914 *
C3D3 7E         0915 DLP1   MOV    A,M    GET THE CHR
C3D4 C5         0916        PUSH   B      SAVE VALUE COUNT
C3D5 CD ED C3   0917        CALL   HBOUT  SEND IT OUT WITH A BLANK
C3D8 7D         0918        MOV    A,L    COMPARE DE & HL
C3D9            0919        SUB    E
C3DA            0920        MOV    A,H
C3DB            0921        SBB    D
C3DC      9 C1  0922        JNC    COMND  ALL DONE
C3DF            0923        POP    B      VALUES PER LINE
C3E0            0924        INX    H
C3E1            0925        DCR    C      BUMP THE LINE COUNT
C3E2      3 C3  0926        JNZ    DLP1   NOT ZERO IF MORE FOR THIS LINE
C3E5 C3 C8 C3   0927        JMP    DLOOP  DO A LFCR BEFORE THE NEXT
                0928 *
                0929 *
                0930 *    OUTPUT HL AS HEX 16 BIT VALUE
                0931 *
C3E8 7C         0932 ADOUT  MOV    A,H    H FIRST
C3E9 CD 0B C4   0933        CALL   HEOUT
C3EC 7D         0934        MOV    A,L    THEN "L" FOLLOWED BY A SPACE
                0935 *
C3ED CD 0B C4   0936 HBOUT  CALL   HEOUT
C3F0 CD 1F C0   0937        CALL   SINP   SEE IF A CHR WAITING
C3F3 CA 06 C4   0938        JZ     BOUT   NO
C3F6 E6 7F      0939        ANI    7FH    CLR PARITY 1ST TWO
C3F8 CA C9 C1   0940        JZ     COMND  EITHER MODE OR CTL-@
C3FB FE 20      0941        CPI    ' '    IS IT A SPACE
C3FD C2 06 C4   0942        JNZ    BOUT   NO--IGN THE CHAR
C400 CD 1F C0   0943 WTLP1  CALL   SINP   IF SPACE, WAIT UNTIL ANY OTHER KEY HIT
C403 CA 00 C4   0944        JZ     WTLP1  THIS ALLOWS LOOKING AT THE DISPLAY
C406 06 20      0945 BOUT   MVI    B,' '
C408 C3 19 C0   0946        JMP    SOUT   PUT IT OUT
                0947 *
C40B 4F         0948 HEOUT  MOV    C,A    GET THE CHARACTER
C40C 0F         0949        RRC
C40D 0F         0950        RRC    .      MOVE THE HIGH FOUR DOWN
C40E 0F         0951        RRC
C40F 0F         0952        RRC
C410 CD 18 C4   0953        CALL   HEOU1  PUT THEM OUT
C413 79         0954        MOV    A,C    THIS TIME THE LOW FOUR
                0955 *
C414 E6 0F      0956 HEOU1  ANI    0FH    FOUR ON THE FLOOR
C416 C6 30      0957        ADI    48     WE WORK WITH ASCII HERE
C418 FE 3A      0958        CPI    58     0-9?
C41A DA 1F C4   0959        JC     OUTH   YUP!
C41D C6 07      0960        ADI    7      MAKE IT A LETTER
C41F 47         0961 OUTH   MOV    B,A    OUTPUT IT FROM REGISTER 'B'
C420 C3 19 C0   0962        JMP    SOUT
                0963 *
                0964 *
                0965 *                      ENTR COMMAND
                0966 *
                0967 *    THIS ROUTINE GETS VALUES FROM THE KEYBOARD AND ENTERS
                0968 * THEM INTO MEMORY.  THE INPUT VALUES ARE SCANNED FOLLOWING
                0969 * A STANDARD 'GCLIN' INPUT SO ON SCREEN EDITING MAY TAKE
                0970 * PLACE PRIOR TO THE LINE TERMINATOR.  A BACK SLASH '/'
                0971 * ENDS THE ROUTINE AND RETURNS CONTROL TO THE COMMAND MODE.
                0972 * A COLON ':' SETS THE PREVIOUS VALUE AS A NEW ADDRESS FOR
                0973 * ENTER.
                0974 *
C423 CD 3A C3   0975 ENTER  CALL   SCONV  SCAN OVER CHARS AND GET ADDRESS
C426 E5         0976        PUSH   H      SAVE ADDRESS
C427 AF         0977        XRA    A
C428 32 07 C8   0978        STA    OPORT  ENTER VALUES TO SCREEN BUFFER
                0979 *
C42B CD F9 C2   0980 ENLOP  CALL   CRLF
C42E      3A    0981        MVI    B,':'
C430      FF C1 0982        CALL   CONT   GET LINE OF INPUT
C433      16 C1 0983        CALL   CREM   REMOVE THE CURSOR
C436       1    0984        MVI    C,1    START SCAN
C437         C1 0985        CALL   VDAD2  GET ADDRESS
C43A            0986        XCHG   .      ....TO DE
                0987 *
                0988 *
C43C 0E 03      0989 ENLO1  MVI    C,3    NO MORE THAN THREE SPACES BETWEEN VALUES
C43E CD 30 C3   0990        CALL   SCHR1  SCAN TO NEXT VALUE
C441 CA 2B C4   0991        JZ     ENLOP  LAST ENTRY FOUND START NEW LINE
                0992 *
C444 FE 2F      0993        CPI    '/'    COMMAND TERMINATOR?
C446 CA C0 C1   0994        JZ     COMN1  IF SO...RETURN TO STANDARD INPUT
C449 CD 40 C3   0995        CALL   SHEX   CONVERT VALUE
C44C FE 3A      0996        CPI    ':'    ADDRESS TERMINATOR?
C44E CA 59 C4   0997        JZ     ENLO3  GO PROCESS IF SO
C451 7D         0998        MOV    A,L    GET LOW PART AS CONVERTED
C452 E1         0999        POP    H      GET MEMORY ADDRESS
C453 77         1000        MOV    M,A    PUT IN THE VALUE
C454 23         1001        INX    H
C455 E5         1002        PUSH   H      BACK GOES THE ADDRESS
C456 C3 3C C4   1003        JMP    ENLO1  CONTINUE THE SCAN
                1004 *
C459 E3         1005 ENLO3  XTHL   .      PUT NEW ADDRESS ON STACK
C45A 13         1006        INX    D      MOVE SCAN PAST TERMINATOR
C45B C3 3C C4   1007        JMP    ENLO1
                1008 *
                1009 *
                1010 *                    EXECUTE COMMAND
                1011 *
                1012 *       THIS ROUTINE GETS THE FOLLOWING PARAMETER AND DOES A
                1013 *  PROGRAM JUMP TO THE LOCATION GIVEN BY IT.  IF PROPER
                1014 *  STACK OPERATIONS ARE USED WITHIN THE EXTERNAL PROGRAM
                1015 *  IT CAN DO A STANDARD 'RETURN TO THE SOLOS COMMAND MODE.
                1016 *  THE STARTING ADDRESS OF SOLOS IS PASSED TO THE PROGRAM
                1017 *  IN REGISTER PAIR HL SO IT CAN ADJUST INTERNAL PARAMETERS
                1018 *  FOR SOLOS OPERATION.
                1019 *
                1020 *
C45E CD 3A C3   1021 EXEC   CALL   SCONV  SCAN PAST BLANKS AND GET PARAMETER
C461 E5         1022 EXEC1  PUSH   H      PUT GO ADDRESS ON STACK
C462 21 00 C0   1023        LXI    H,START  TELL THE PROGRAM WHERE WE CAME FROM
C465 C9         1024        RET    .      AND DISPATCH TO IT
                1025 *
                1026 *
                1027 *    THIS ROUTINE GETS A NAME OF UP TO 5 CHARACTERS
                1028 *  FROM THE INPUT STRING.  IF THE TERMINATOR IS A
                1029 *  SLASH (/) THEN THE CHARACTER FOLLOWING IS TAKEN
                1030 *  AS THE CASSETTE UNIT SPECIFICATION.
                1031 *
                1032 *
C466 21 1C C8   1033 NAMES  LXI    H,THEAD  POINT TO INTERNAL HEADER
C469 CD 1B C3   1034 NAME   CALL   SBLK   SCAN OVER TO FIRST CHRS
C46C 06 06      1035        MVI    B,6    UP TO SIX ARE ACCEPTED
                1036 *
C46E 1A         1037 NAME1  LDAX   D      GET CHARACTER
C46F FE 20      1038        CPI    ' '    NO UNIT DELIMITER
C471 CA 86 C4   1039        JZ     NFIL
C474 FE 2F      1040        CPI    '/'    UNIT DELIMITER
C476 CA 86 C4   1041        JZ     NFIL
C479 77         1042        MOV    M,A
C47A 13         1043        INX    D      BUMP THE SCAN POINTER
C47B 23         1044        INX    H
C47C 05         1045        DCR    B
C47D C2 6E C4   1046        JNZ    NAME1  FALL THROUGH TO ERR1 IF TOO MANY CHRS IN NAME
                1047 *
                1048 *
                1049 *    SOLOS ERROR HANDLER
                1050 *
C480 EB         1051 ERR1   XCHG   .      GET SCAN ADDRESS TO HL
C481 36 3F      1052 ERR2   MVI    M,'?'  PUT QUESTION MARK ON SCREEN
C483 C3 C0 C1   1053        JMP    COMN1  AND RETURN TO COMMAND MODE
                1054 *
                1055 *
                1056 *  HERE WE HAVE SCANNED OFF THE NAME.  ZERO FILL FOR
                1057 *  NAMES LESS THAN FIVE CHARACTERS.
                1058 *
C486 36 00      1059 NFIL   MVI    M,0    PUT IN AT LEAST ONE ZERO
C488 23         1060        INX    H
C489 05         1061        DCR    B
C48A C2 86 C4   1062        JNZ    NFIL   LOOP UNTIL B IS ZERO
                1063 *
C48D FE 2F      1064        CPI    '/'    IS THERE A UNIT SPECIFICATION?
C48F 3E 01      1065        MVI    A,1    PRETEND NOT
C491 C2 9A C4   1066        JNZ    DEFLT
C494 13         1067        INX    D      MOVE PAST THE TERMINATOR
C495 CD 2E C3   1068        CALL   SCHR   GO GET UNIT SPEC
C498 D6 30      1069        SUI    '0'    REMOVE ASCII BIAS
                1070 *
      C49A      1071 DEFLT  EQU    $      MOVE OVER TO INTERNAL REPRESENTATION
C49A E6 01      1072        ANI    1      JUST BIT ZERO
C49C 3E 80      1073        MVI    A,TAPE1  ASSUME TAPE ONE
C49E C2 A2 C4   1074        JNZ    STUNT  IF NON-ZERO, ITS ONE
C4A1 1F         1075        RAR
C4A2 32 5A C8   1076 STUNT  STA    FNUMF  SET IT IN
C4A5 C9         1077        RET
                1078 *
                1079 *
                1080 *
                1081 *    THIS ROUTINE PROCESSES THE XEQ AND GET COMMANDS
                1082 *
                1083 *
C4A6 3E         1084 TXEQ   DB     3EH    THIS BECOMES "MVI A,0AFH"
C4A7 AF         1085 TLOAD  XRA    A      A=0 MEANS TLOAD, ELSE TXEQ
C4A8 F5         1086        PUSH   PSW    SAVE FLAG FOR LATER
C4A9 21 2C C8   1087        LXI    H,DHEAD  PLACE DUMMY HEADER HERE
C4AC CD 69 C4   1088        CALL   NAME   SET IN NAME AND UNIT
C4AF 21 00 00   1089        LXI    H,0    PRETEND NO SECOND VALUE
C4B2 CD 10 C3   1090        CALL   PSCAN  GO GET THE ADDRESS (IF PRESENT)
                1091 *
C4B5 EB         1092 TLOA2  XCHG   .      PUT ADDRESS IN DE
C4B6 21 2C C8   1093        LXI    H,DHEAD  PT TO DUMMY HEADER W/ NAME TO LOAD
C4B9 7E         1094        MOV    A,M    SEE IF A NAME WAS ENTERED
C4BA B7         1095        ORA    A      IS THERE A NAME?
C4BB C2 C1 C4   1096        JNZ    TLOA3  YES--SEARCH FOR IT
C4BE 21 1C C8   1097        LXI    H,THEAD  NO NAME, LOAD 1ST FILE
C4C1 E5         1098 TLOA3  PUSH   H      SAVE PTR TO NAME TO LOAD
C4C2 CD 48 C5   1099        CALL   ALOAD  GET UNIT AND SPEED
C4C5 E1         1100        POP    H      RESTORE PTR TO HDR TO LOAD
C4C6 CD CB C6   1101        CALL   RTAPE  READ IN THE TAPE
C4C9 DA 14 C5   1102        JC     TAERR  TAPE ERROR?
                1103 *
C4CC CD 50 C5   1104        CALL   WAOUT  PUT OUT THE HEADER PARAMETERS
C4CF F1         1105        POP    PSW    RESTORE FLAG FROM ORIGINAL ENTRY
C4D0 B7         1106        ORA    A
C4D1 C8         1107        RZ     .      AUTO XEQ NOT WANTED
C4D2 3A 22 C8   1108        LDA    WTYPE  CHECK TYPE
C4D5 B7         1109        ORA    A      SET FLAGS
C4D6 FA 14 C5   1110        JM     TAERR  TYPE IS NON XEQ
C4D9 3A 21 C8   1111        LDA    THEAD+5  GET CHARACTER PAST NAME
C4DC B7         1112        ORA    A
C4DD C2 14 C5   1113        JNZ    TAERR  THE BYTE MUST BE ZERO FOR AUTO XEQ
C4E0 2A 27 C8   1114        LHLD   XEQAD  GET THE TAPE ADDRESS
C4E3 C3 61 C4   1115        JMP    EXEC1  AND GO TO IT
                1116 *
                1117 *
                1118 *                      -- GET --
                1119 *
                1120 *    THIS ROUTINE IS USED TO SAVE PROGRAMS AND DATA ON
                1121 *    THE CASSETTE UNIT.
                1122 *
                1123 *
C4E6 CD 66 C4   1124 TSAVE  CALL   NAMES  GET NAME AND UNIT
C4E9 CD 3A C3   1125        CALL   SCONV  GET START ADDRESS
C4EC E5         1126        PUSH   H      USE THE STACK AS A REGISTER
C4ED CD 3A C3   1127        CALL   SCONV  GET END ADDRESS
C4F0 E3         1128        XTHL   .      PUT END ON STACK, GET BACK START
C4F1 E5         1129        PUSH   H      SAVE START ON TOP OF STACK
C4F2 CD 10 C3   1130        CALL   PSCAN  SEE IF OPTIONAL HEADER ADDRESS WAS GIVEN
C4F5 22 25 C8   1131        SHLD   LOADR  PUT HEADER ADDRESS IN PLACE
                1132 *
```

```
C4F8 E1            1133        POP    H      "FROM" ADDRESS TO HL
C4F9 D1            1134        POP    D      GET BACK "END" ADDRESS
C4FA E5            1135        PUSH   H      SAVE FROM AGAIN FOR LATER
C4FB 7B            1136        MOV    A,E    NOW CALCULATE SIZE
C4FC 95            1137        SUB    L      SIZE=END-START+1
C4FD 6F            1138        MOV    L,A
C4FE 7A            1139        MOV    A,D
C4FF 9C            1140        SBB    H
C500 67            1141        MOV    H,A
C501 23            1142        INX    H
C502 22 23 C8      1143        SHLD   BLOCK  STORE THE SIZE
C505 E5            1144        PUSH   H      SAVE IT FOR THE READ ALSO
                   1145 *
C506 CD A8 C5      1146        CALL   ALOAD  GET UNIT AND SPEED
C509 21 1C C8      1147        LXI    H,THEAD  POINT TO HEADER
C50C CD AF C7      1148        CALL   WHEAD  AND WRITE IT OUT
                   1149 *   NOW WRITE OUT THE DATA
C50F D1            1150        POP    D      GET SIZE TO DE
C510 E1            1151        POP    H      GET BACK "FROM" ADDRESS
C511 C3 90 C7      1152        JMP    WRLO1  WRITE OUT THE DATA AND RETURN
                   1153 *
                   1154 *
                   1155 *   OUTPUT ERROR AND HEADER
                   1156 *
C514 CD F9 C2      1157 TAERR  CALL   CRLF
C517 16 06         1158        MVI    D,6
C519 21 25 C5      1159        LXI    H,ERRM  POINT TO ERROR MESSAGE
C51C CD 6A C5      1160        CALL   NLOOP  OUTPUT ERROR
C51F CD 50 C5      1161        CALL   NAOUT  THEN THE HEADER
C522 C3 C0 C1      1162        JMP    CONN1  AND BE SURE THE TAPE UNITS ARE OFF
                   1163 *
C525 A5 52 52 4F   1164 ERRM   ASC    !ERROR !
     52 20
                   1165 *
                   1166 *
                   1167 *   THIS ROUTINE READS HEADERS FROM THE TAPE AND OUTPUTS
                   1168 *   THEM TO THE OUTPUT DEVICE.  IT CONTINUES UNTIL THE
                   1169 *   MODE KEY IS DEPRESSED.
                   1170 *
C52B CD 66 C4      1171 TLIST  CALL   NAMES  SET UP UNIT IF GIVEN
C52E CD F9 C2      1172        CALL   CRLF
                   1173 *
                   1174 *
C531 CD A8 C5      1175 LLIST  CALL   ALOAD
C534 06 01         1176        MVI    B,1
C536 CD EF C7      1177        CALL   TON    TURN ON THE TAPE
                   1178 *
C539 CD 23 C7      1179 LIST1  CALL   RHEAD
C53C DA C0 C1      1180        JC     CONN1  TURN OFF THE TAPE UNIT
C53F C2 39 C5      1181        JNZ    LIST1
C542 CD 50 C5      1182        CALL   NAOUT  OUTPUT THE HEADER
C545 C3 39 C5      1183        JMP    LIST1  LOOP UNTIL MODE IS DEPRESSED
                   1184 *
                   1185 *
                   1186 *   THIS ROUTINE GETS THE CASSETTE UNIT NUMBER AND
                   1187 *   SPEED TO REGISTER "A" FOR THE TAPE CALLS
                   1188 *
C548 21 5A C8      1189 ALOAD  LXI    H,FNUMF  POINT TO THE UNIT SPECIFICATION
C54B 3A 0D C8      1190        LDA    TSPD   GET THE TAPE SPEED
C54E B6            1191        ORA    M      PUT THEM TOGETHER
C54F C9            1192        RET    .      AND GO BACK
                   1193 *
                   1194 *
                   1195 *   THIS ROUTINE OUTPUTS THE NAME AND PARAMETERS OF
                   1196 *   THEAD TO THE OUTPUT DEVICE.
                   1197 *
                   1198 *
C550 16 08         1199 NAOUT  MVI    D,8
C552 21 1B C8      1200        LXI    H,THEAD-1  POINT TO THE HEADER
C555 CD 6A C5      1201        CALL   NLOOP  OUTPUT THE HEADER
C558 CD 06 C4      1202        CALL   BOUT   ANOTHER BLANK
C55B 2A 25 C8      1203        LHLD   LOADR  NOW THE LOAD ADDRESS
C55E CD E8 C3      1204        CALL   ADOUT  PUT IT OUT
C561 2A 23 C8      1205        LHLD   BLOCK  AND THE BLOCK SIZE
C564 CD E8 C3      1206        CALL   ADOUT
C567 C3 F9 C2      1207        JMP    CRLF   DO THE CRLF AND RETURN
                   1208 *
                   1209 *
C56A 7E            1210 NLOOP  MOV    A,M    GET CHARACTER
C56B B7            1211        ORA    A
C56C C2 71 C5      1212        JNZ    CHRLI  IF IT ISN'T A ZERO
C56F 3E 20         1213        MVI    A,' '
C571 CD 1F C4      1214 CHRLI  CALL   OUTH   OUTPUT CHAR NOW
C574 23            1215        INX    H
C575 15            1216        DCR    D
C576 C2 6A C5      1217        JNZ    NLOOP
C579 C9            1218        RET
                   1219 *
                   1220 *
                   1221 *
                   1222 *
                   1223 *       "SET" COMMAND
                   1224 *
                   1225 *   THIS ROUTINE GETS THE ASSOCIATED PARAMETER AND
                   1226 *   DISPATCHES TO THE PROPER ROUTINE FOR SETTING
                   1227 *   GLOBAL VALUES.
                   1228 *
      C57A         1229 SET    EQU    $      THIS IS THE SET COMMAND
C57A CD 1B C3      1230        CALL   SBLK   LOOK FOR SET NAME
C57D CA 80 C4      1231        JZ     ERR1   MUST HAVE AT LEAST SOMETHING!!
C580 D5            1232        PUSH   D      SAVE SCAN ADDRESS
C581 CD 3A C3      1233        CALL   SCONV  CONVERT FOLLOWING VALUE
C584 E3            1234        XTHL   .      GET SCAN ADDRESS BACK..SAVE VALUE ON STACK
C585 11 A2 C2      1235        LXI    D,SETAB  SECONDARY COMMAND TABLE
C588 CD 31 C2      1236        CALL   FDCOM  SEE IF IN TABLE
C58B C3 22 C2      1237        JMP    DISPO  AND EITHER ERR OR OFF TO IT
                   1238 *
                   1239 *
                   1240 *   THIS ROUTINE SETS THE TAPE SPEED
                   1241 *
C58E B7            1242 TASPD  ORA    A      IS IT ZERO?
C58F CA 94 C5      1243        JZ     SETSP  YES--THAT'S A VALID SPEED
C592 3E 20         1244        MVI    A,32   SET TO SLOW IF NON-ZERO
C594 32 0D C8      1245 SETSP  STA    TSPD   SPEED IS STORED HERE
C597 C9            1246        RET
                   1247 *
                   1248 *
C598 78            1249 STSPD  MOV    A,B    ESCAPE COMES HERE TO SET SPEED
C599 32 0B C8      1250 DISPD  STA    SPEED  SET DISPLAY SPEED
C59C C9            1251        RET    .
                   1252 *
                   1253 *   SET INPUT DRIVER
                   1254 *
      C59D         1255 SETIN  EQU    $
C59D 32 06 C8      1256        STA    IPORT
C5A0 C9            1257        RET
                   1258 *
                   1259 *   SET OUTPUT DRIVER
                   1260 *
      C5A1         1261 SETOT  EQU    $
C5A1 32 07 C8      1262        STA    OPORT
C5A4 C9            1263        RET
                   1264 *
                   1265 *   SET USERS CUSTOM INPUT DRIVER ADDRESS
                   1266 *
C5A5 22 00 C8      1267 SETCI  SHLD   UIPRT
C5A8 C9            1268        RET
                   1269 *
                   1270 *   SET USERS CUSTOM OUTPUT DRIVER ADDRESS
                   1271 *
C5A9 22 02 C8      1272 SETCO  SHLD   UOPRT
C5AC C9            1273        RET
                   1274 *
                   1275 *   SET TYPE BYTE INTO HEADER
                   1276 *
C5AD 32 22 C8      1277 SETTY  STA    HTYPE
C5B0 C9            1278        RET
                   1279 *
                   1280 *   SET EXECUTE ADDRESS INTO HEADER
                   1281 *
C5B1 22 27 C8      1282 SETXQ  SHLD   XEQAD
C5B4 C9            1283        RET
                   1284 *
                   1285 *
C5B5 32 10 C8      1286 SETNU  STA    NUCNT  SET THE NULL COUNT
C5B8 C9            1287        RET    .      THAT'S DONE
                   1288 *
                   1289 *
      C5B9         1290 SETCR  EQU    $      SET TO IGNORE CRC ERRORS
C5B9 32 11 C8      1291        STA    IGNCR  FF=IGNORE ERRORS, ELSE=NORMAL
C5BC C9            1292        RET    .
                   1293 *
                   1294 *
                   1295 *
                   1296 *   CUSTOM COMMAND NAME AND ADDRESS INTO CUSTOM COMMAND
                   1297 *
C5BD CD 66 C4      1298 CUSET  CALL   NAMES  CUSTOM COMMAND ENTRY/REMOVAL
C5C0 21 C9 C1      1299        LXI    H,COMND  DEFAULT ADDR IF NONE GIVEN
C5C3 CD 10 C3      1300        CALL   PSCAN  GET RTN ADDR
C5C6 E5            1301        PUSH   H      SAVE RTN ADDR
C5C7 21 1C C8      1302        LXI    H,THEAD  PT AT NAME TO SEARCH
C5CA CD 2E C2      1303        CALL   FDCOU  SEARCH IT IN CUSTOM TABLE
C5CD CA D3 C5      1304        JZ     CUSE2  NOT IN TABLE--ENTER IT
C5D0 1B            1305        DCX    D      IN TABLE, REMOVE IT
C5D1 36 00         1306        MVI    M,0    CHANGE NEW NAME TO BE ZERO
C5D3 7E            1307 CUSE2  MOV    A,M    GET 1ST CHAR OF NAME
C5D4 12            1308        STAX   D      ENTER IT INTO TABLE
C5D5 13            1309        INX    D      AND THE 2ND NAME
C5D6 23            1310        INX    H
C5D7 7E            1311        MOV    A,M
C5D8 12            1312        STAX   D      NAME NOW ENTERED
C5D9 13            1313        INX    D      GET SET TO ENTER ADDRESS
C5DA E1            1314        POP    H      RESTORE RTN ADDR
C5DB EB            1315        XCHG   .
C5DC 73            1316        MOV    M,E    SET ADDR IN NOW
C5DD 23            1317        INX    H      AND HI BYTE OF ADDR
C5DE 72            1318        MOV    M,D
C5DF C9            1319        RET    .      NAME IS NOW ENTERED OR CLEARED
                   1320 *
                   1321 * -0-
                   9999        COPY   SOLOS3/1
                   1322 *
                   1323 *
                   1324 *
                   1325 *
                   1326 *   THE FOLLOWING ROUTINES PROVIDE "BYTE BY BYTE" ACCESS
                   1327 *  TO THE CASSETTE TAPES ON EITHER A READ OR WRITE BASIS.
                   1328 *
                   1329 *  THE TAPE IS READ ONE BLOCK AT A TIME AND INDIVIDUAL
                   1330 *  TRANSFERS OF DATA HANDLED BY MANAGING A BUFFER AREA.
                   1331 *
                   1332 *  THE BUFFER AREA IS CONTROLLED BY A FILE CONTROL BLOCK
                   1333 *  (FCB) WHOSE STRUCTURE IS:
                   1334 *
                   1335 *
                   1336 *     7 BYTES FOR EACH OF THE TWO FILES STRUCTURED AS
                   1337 *   FOLLOWS:
                   1338 *
                   1339 *          1 BYTE -  ACCESS CONTROL   00  IF CLOSED
                   1340 *                                     FF  IF READING
                   1341 *                                     FE  IF WRITING
                   1342 *          1 BYTE -  READ COUNTER
                   1343 *          1 BYTE -  BUFFER POSITION POINTER
                   1344 *          2 BYTE -  CONTROL HEADER ADDRESS
                   1345 *          2 BYTE -  BUFFER LOCATION ADDRESS
                   1346 *
                   1347 *
                   1348 *
                   1349 *        THIS ROUTINE "OPENS" THE CASSETTE UNIT FOR ACCESS
                   1350 *
                   1351 *   ON ENTRY:  A - HAS THE TAPE UNIT NUMBER (1 OR 2)
                   1352 *              HL - HAS USER SUPPLIED HEADER FOR TAPE FILE
                   1353 *
                   1354 *
                   1355 *   NORMAL RETURN:    ALL REGISTERS ARE ALTERED
                   1356 *                     BLOCK IS READY FOR ACCESS
                   1357 *
                   1358 *   ERROR RETURN:     CARRY BIT IS SET
                   1359 *
                   1360 *   ERRORS:  BLOCK ALREADY OPEN
                   1361 *
                   1362 *
C5E0 E5            1363 BOPEN  PUSH   H      SAVE HEADER ADDRESS
C5E1 CD 33 C6      1364        CALL   LFCB   GET ADDRESS OF FILE CONTROL
C5E4 C2 FA C5      1365        JNZ    TERE2  FILE WAS ALREADY OPEN
C5E7 36 01         1366        MVI    M,1    NOW IT IS
C5E9 23            1367        INX    H      POINT TO READ COUNT
C5EA 77            1368        MOV    M,A    ZERO
C5EB 23            1369        INX    H      POINT TO BUFFER CURSOR
C5EC 77            1370        MOV    M,A    PUT IN THE ZERO COUNT
                   1371 *
                   1372 *  ALLOCATE THE BUFFER
                   1373 *
C5ED 11 63 C8      1374        LXI    D,FBUF1  POINT TO BUFFER AREA
C5F0 3A 54 C8      1375        LDA    FNUMF  GET WHICH ONE WE ARE GOING TO USE
C5F3 82            1376        ADD    D
C5F4 57            1377        MOV    D,A    256 BIT ADD
                   1378 *
C5F5 C1            1379 UBUF   POP    B      HEADER ADDRESS
C5F6 B7            1380        ORA    A      CLEAR CARRY AND RETURN AFTER STORING PARAMS
C5F7 C3 B6 C6      1381        JMP    PSTOR  STORE THE VALUES
                   1382 *
                   1383 *    GENERAL ERROR RETURN POINTS FOR STACK CONTROL
                   1384 *
C5FA E1            1385 TERE2  POP    H
C5FB D1            1386 TERE1  POP    D
C5FC AF            1387 TERE0  XRA    A      CLEAR ALL FLAGS
C5FD 37            1388        STC    .      SET ERROR
C5FE C9            1389        RET
                   1390 *
                   1391 *
C5FF 3D            1392 EOFER  DCR    A      SET MINUS FLAGS
C600 37            1393        STC    .      AND CARRY
C601 D1            1394        POP    D      CLEAR THE STACK
C602 C9            1395        RET    .      THE FLAGS TELL ALL
                   1396 *
                   1397 *
                   1398 *
                   1399 *
                   1400 *   THIS ROUTINE CLOSES THE FILE BUFFER TO ALLOW ACCESS
                   1401 *   FOR A DIFFERENT CASSETTE OR PROGRAM.  IF THE FILE
                   1402 *   OPERATIONS WERE "WRITE" THEN THE LAST BLOCK IS WRITTEN
                   1403 *   OUT AND AN "END OF FILE" WRITTEN TO THE TAPE.  IF
                   1404 *   THE OPERATIONS WERE "READS" THEN THE FILE IS JUST
                   1405 *   MADE READY FOR NEW USE.
                   1406 *
                   1407 *   ON ENTRY:  A - HAS WHICH UNIT (1 OR 2)
                   1408 *
                   1409 *   ERROR RETURNS:  FILE WASN'T OPEN
                   1410 *
                   1411 *
```

```
C603 CD 33 C6        1812 PCLOS  CALL   LFCB   GET CONTROL BLOCK ADDRESS
C606 C8              1813        RZ     .      WASN'T OPEN, CARRY IS SET FROM LFCB
C607 B7              1814        ORA    A      CLEAR CARRY
C608 -               1815        INR    A      SET CONDITION FLAGS
C609                 1816        MVI    M,0    CLOSE THE CONTROL BYTE
C60B                 1817        RZ     .      WE WERE READING...NOTHING MORE TO DO
                     1818 *
                     1819 *     THE FILE OPERATIONS WERE "WRITES"
                     1820 *
                     1821 *  PUT THE CURRENT BLOCK ON THE TAPE                          C6B3 CD 33 C6
                     1822 *  (EVEN IF ONLY ONE BYTE!!)                                  C6B6 C8
                     1823 *  THEN WRITE AN END OF FILE TO THE TAPE                      C6B7 3C
                     1824 *                                                             C6B8 C8
                     1825 *                                                             C6B9 36 FE
C60C 23              1826        INX    H                                              C6BB 23
C60D 23              1827        INX    H                                              C6BC 23
C60E 7E              1828        MOV    A,M    GET CURSOR POSITION                     C6BD 78
C60F 7E              1829                                                              C6BE F5
C610 CD BF C6        1830        CALL   PLOAD  BC GET HEADER ADDRESS, DE BUFFER ADDRESS C6BF E5
C613 C5              1831        PUSH   B      HEADER TO STACK
C614 21 07 00        1832        LXI    H,BLKOF  OFFSET TO BLOCK SIZE
C617 09              1833        DAD    B
C618 B7              1834        ORA    A      TEST COUNT
C619 CA 2B C6        1835        JZ     EOFW   NO BYTES...JUST WRITE EOF               C690 CD BF C6
                     1836 *                                                            C693 E1
                     1837 *    WRITE LAST BLOCK                                        C694 7E
                     1838 *                                                            C695 83
                                                                                       C696 5F
C61C E5              1839        PUSH   H      SAVE BLOCK SIZE POINTER FOR EOF         C697 D2 9B C6
C61D 77              1840        MOV    M,A    PUT IN COUNT                            C69A 14
C61E 23              1841        INX    H                                              C69B F1
C61F 36 00           1842        MVI    M,0    ZERO THE HIGHER BYTE                    C69C 12
C621 23              1843        INX    H                                              C69D B7
C622 73              1844        MOV    M,E    BUFFER ADDRESS                          C69E 34
C623 23              1845        INX    H                                              C69F C0
C624 72              1846        MOV    M,D
C625 60              1847        MOV    H,B
C626 69              1848        MOV    L,C    PUT HEADER ADDRESS IN HL
C627 CD 7C C7        1849        CALL   WFBLK  GO WRITE IT OUT                         C6A0 CD A6 C6
C62A E1              1850        POP    H      BLOCK SIZE POINTER                      C6A3 C3 7C C7
                     1851 *
                     1852 *   NOW WRITE END OF FILE TO CASSETTE
                     1853 *
C62B AF              1854 EOFW   XRA    A      PUT IN ZEROS FOR SIZE:  EOF MARK IS ZERO BYTES!
C62C 77              1855        MOV    M,A
C62D 23              1856        INX    H
C62E 77              1857        MOV    M,A
C62F E1              1858        POP    H      HEADER ADDRESS
C630 C3 7C C7        1859        JMP    WFBLK  WRITE IT OUT AND RETURN                 C6A6 CD BF C6
                     1860 *                                                            C6A9 C5
                     1861 *                                                            C6AA 21 06 00
                     1862 *                                                            C6AD 09
                     1863 *                                                            C6AE 01 00 01
                     1864 *   THIS ROUTINE LOCATES THE FILE CONTROL BLOCK POINTED TO    C6B1 CD B6 C6
                     1865 *   BY REGISTER "A".  ON RETURN HL POINT TO THE CONTROL BYT   C6B4 E1
                     1866 *   AND REGISTER "A" HAS THE CONTROL WORD WITH THE FLAGS      C6B5 C9
                     1867 *   SET FOR IMMEDIATE CONDITION DECISIONS.
                     1868 *
                     1869 *
C633 21 55 C8        1870 LFCB   LXI    H,FCBAS  POINT TO THE BASE OF IT               C6B6 23
C636 1F              1871        RAR    .      MOVE THE 1 & 2 TO 0 A 1 LIKE COMPUTERS LIKE C6B7 71
C637 E6 01           1872        ANI    1      SMALL NUMBERS ARE THE RULE              C6B8 23
C639 32 54 C8        1873        STA    FNUMF  CURRENT ACCESS FILE NUMBER              C6B9 70
C63C CA A2 C6        1874        JZ     LFCB1  UNIT ONE (VALUE OF ZERO)                C6BA 23
C63F 21 5C C8        1875        LXI    H,FCBA2  UNIT TWO--PT TO ITS FCB               C6BB 73
     C642            1876 LFCB1  EQU    $      HL PT TO PROPER FCB                    C6BC 23
C642 7E              1877        MOV    A,M    PICK UP FLAGS FM FCB                    C6BD 72
C643 B7              1878        ORA    A      SET FLAGS BASED ON CONTROL WORD         C6BE C9
C644                 1879        STC    .      SET CARRY IN CASE OF IMMEDIATE ERROR RETURN
C645                 1880        RET
                     1881 *
                     1882 *
                     1883 *
                     1884 *
                     1885 *    READ TAPE BYTE ROUTINE
                     1886 *
                     1887 *    ENTRY:       -  A -  HAS FILE NUMBER
                     1888 *    EXIT: NORMAL -  A -  HAS BYTE
                     1889 *          ERROR
                     1890 *            CARRY SET     - IF FILE NOT OPEN OR
                     1891 *                            PREVIOUS OPERATIONS WERE WRITE
                     1892 *            CARRY & MINUS - END OF FILE ENCOUNTERED
                     1893 *
                     1894 *
                     1895 *
                     1896 *                                                            C6C8 CD DE C7
C646 CD 33 C6        1897 RTBYT  CALL   LFCB   LOCATE THE FILE CONTROL BLOCK
C649 C8              1898        RZ     .      FILE NOT OPEN
C64A 3C              1899        INR    A      TEST IF FF
C64B FA FC C5        1900        JM     TERE0  ERROR WAS WRITING
C64E 36 FF           1901        MVI    M,-1   SET IT AS READ  (IN CASE IT WAS JUST OPENED)
C650 23              1902        INX    H
C651 7E              1903        MOV    A,M    GET READ COUNT
C652 E5              1904        PUSH   H      SAVE COUNT ADDRESS
C653 23              1905        INX    H
C654 CD BF C6        1906        CALL   PLOAD  GET THE OTHER PARAMETERS
C657 E1              1907        POP    H
C658 B7              1908        ORA    A
C659 C2 75 C6        1909        JNZ    GTBYT  IF NOT EMPTY GO GET BYTE
                     1910 *
                     1911 *  CURSOR POSITION WAS ZERO...READ A NEW BLOCK INTO           C6CB D5
                     1912 *  THE BUFFER.                                               C6CC 06 03
                     1913 *                                                            C6CE CD EF C7
C65C D5              1914 RDNBLK PUSH   D      BUFFER POINTER                          C6D1 DB FB
C65D E5              1915        PUSH   H      TABLE ADDRESS                           C6D3 E5
C65E 23              1916        INX    H                                              C6D4 CD 23 C7
C65F CD A6 C6        1917        CALL   PHEAD  PREPARE THE HEADER FOR READ              C6D7 E1
C662 CD C8 C6        1918        CALL   RFBLK  READ IN THE BLOCK                       C6D8 DA 06 C7
C665 DA FA C5        1919        JC     TERE2  ERROR POP OFF STACK BEFORE RETURN       C6DB C2 D3 C6
C668 E1              1920        POP    H
C669 7B              1921        MOV    A,E    LOW BYTE OF COUNT (WILL BE ZERO IF 256)
C66A B2              1922        ORA    D      SEE IF BOTH ARE ZERO
C66B CA FF C5        1923        JZ     EOFER  BYTE COUNT WAS ZERO....END OF FILE      C6DE E5
C66E 73              1924        MOV    M,E    NEW COUNT ( ZERO IS 256 AT THIS POINT)  C6DF 11 1C C8
C66F 23              1925        INX    H      BUFFER LOCATION POINTER                 C6E2 CD D2 C7
C670 36 00           1926        MVI    M,0    .                                       C6E5 E1
C672 2B              1927        DCX    H                                              C6E6 C2 D3 C6
C673 7B              1928        MOV    A,E    COUNT TO A
C674 D1              1929        POP    D      GET BACK BUFFER ADDRESS
                     1930 *                                                            C6E9 D1
                     1931 *                                                            C6EA 7A
                     1932 *                                                            C6EB B3
                     1933 *    THIS ROUTINE GETS ONE BYTE FROM THE BUFFER              C6EC 2A 23 C8
                     1934 *  AND RETURNS IT IN REGISTER "A".  IF THE END               C6EF EB
                     1935 *  OF THE BUFFER IS REACHED IT MOVES THE POINTER
                     1936 *  TO THE BEGINNING OF THE BUFFER FOR THE NEXT               C6F0 C2 F6 C6
                     1937 *  LOAD.                                                     C6F3 2A 25 C8
                     1938 *
C675 3D              1939 GTBYT  DCR    A      BUMP THE COUNT
C676 -               1940        MOV    M,A    RESTORE IT
C677                 1941        INX    H
C678                 1942        MOV    A,M    GET BUFFER POSITION
C679                 1943        INR    M      BUMP IT
                     1944 *
C67A 6               1945        ADD    E
C67B 5F              1946        MOV    E,A    DE NOW POINT TO CORRECT BUFFER POSITION
C67C D2 80 C6        1947        JNC    RT1                                            C6F6 D5
C67F 14              1948        INR    D
C680 1A              1949 RT1    LDAX   D      GET CHARACTER FROM BUFFER                    C6F7
C681 B7              1950        ORA    A      CLEAR CARRY                             C6F7 CD 15 C7
C682 C9              1951        RET    .      ALL DONE                                C6FA CA 10 C7
                     1952 *
                     1953 *
                     1954 *
                     1955 *
                     1956 *      THIS ROUTINE IS USED TO WRITE A BYTE TO THE FILE
                     1957 *
                     1958 *      ON ENTRY:   A -  HAS FILE NUMBER
                     1959 *                  B -  HAS DATA BYTE
                     1960 *
                     1961 *
                     1962 WTBYT  CALL   LFCB   GET CONTROL BLOCK
                     1963        RZ     .      FILE WASN'T OPEN
                     1964        INR    A
                     1965        RZ     .      FILE WAS READ
                     1966        MVI    M,0FEH  SET IT TO WRITE
                     1967        INX    H
                     1968        INX    H
                     1969        MOV    A,B    GET CHARACTER
                     1970        PUSH   PSW
                     1971        PUSH   H      SAVE CONTROL ADDRESS+2
                     1972 *
                     1973 *   NOW DO THE WRITE
                     1974 *
                     1975        CALL   PLOAD  BC GETS HEADER ADDR, DE BUFFER ADDRESS
                     1976        POP    H
                     1977        MOV    A,M    COUNT BYTE
                     1978        ADD    E
                     1979        MOV    E,A
                     1980        JNC    WT1
                     1981        INR    D
                     1982 WT1    POP    PSW    CHARACTER
                     1983        STAX   D      PUT CHR IN BUFFER
                     1984        ORA    A      CLEAR FLAGS
                     1985        INR    M      INCREMENT THE COUNT
                     1986        RNZ    .      RETURN IF COUNT DIDN'T ROLL OVER
                     1987 *
                     1988 *   THE BUFFER IS FULL.  WRITE IT TO TAPE AND RESET
                     1989 *  CONTROL BLOCK.
                     1990 *
                     1991        CALL   PHEAD  PREPARE THE HEADER
                     1992        JMP    WFBLK  WRITE IT OUT AND RETURN
                     1993 *
                     1994 *
                     1995 *
                     1996 *
                     1997 *  THIS ROUTINE PUTS THE BLOCK SIZE (256) AND BUFFER
                     1998 *  ADDRESS IN THE FILE HEADER.
                     1999 *
                     2000 PHEAD  CALL   PLOAD  GET HEADER AND BUFFER ADDRESSES
                     2001        PUSH   B      HEADER ADDRESS
                     2002        LXI    H,BLKOF-1  PSTOR DOES AN INCREMENT
                     2003        DAD    B      HL POINT TO BLOCKSIZE ENTRY
                     2004        LXI    B,256
                     2005        CALL   PSTOR
                     2006        POP    H      HL RETURN WITH HEADER ADDRESS
                     2007        RET
                     2008 *
                     2009 *
                     2010 PSTOR  INX    H
                     2011        MOV    M,C
                     2012        INX    H
                     2013        MOV    M,B
                     2014        INX    H
                     2015        MOV    M,E
                     2016        INX    H
                     2017        MOV    M,D
                     2018        RET
                     2019 *
                     2020 *
                     2021 PLOAD  INX    H
                     2022        MOV    C,M
                     2023        INX    H
                     2024        MOV    B,M
                     2025        INX    H
                     2026        MOV    E,M
                     2027        INX    H
                     2028        MOV    D,M
                     2029        RET
                     2030 *
                     2031 *
                     2032 *
                     2033 *
                     2034 *
                     2035 *   THIS ROUTINE SETS THE CORRECT UNIT FOR SYSTEM READS
                     2036 RFBLK  CALL   GTUNT  SET UP A=UNIT WITH SPEED
                     2037 *
                     2038 *
                     2039 *
                     2040 *
                     2041 *              TAPE READ ROUTINES
                     2042 *
                     2043 *     ON ENTRY:       A   HAS UNIT AND SPEED
                     2044 *                     HL  POINT TO HEADER BLOCK
                     2045 *                     DE  HAVE OPTIONAL PUT ADDRESS
                     2046 *
                     2047 *     ON EXIT:        CARRY IS SET IF ERROR OCCURED
                     2048 *                     TAPE UNITS ARE OFF
                     2049 *
                     2050 *
                     2051 RTAPE  PUSH   D      SAVE OPTIONAL ADDRESS
                     2052        MVI    B,3    SHORT DELAY
                     2053        CALL   TON
                     2054        IN     TDATA  CLEAR THE UART FLAGS
                     2055 *
                     2056 RTAP1  PUSH   H      HEADER ADDRESS
                     2057        CALL   RHEAD  GO READ HEADER
                     2058        POP    H
                     2059        JC     TERR   IF AN ERROR OR ESC WAS RECEIVED
                     2060        JNZ    RTAP1  IF VALID HEADER NOT FOUND
                     2061 *
                     2062 *  FOUND A VALID HEADER NOW DO COMPARE
                     2063 *
                     2064        PUSH   H      GET BACK AND RESAVE ADDRESS
                     2065        LXI    D,THEAD
                     2066        CALL   DHCMP  COMPARE DE-HL HEADERS
                     2067        POP    H
                     2068        JNZ    RTAP1
                     2069 *
                     2070 *
                     2071        POP    D      OPTIONAL "PUT" ADDRESS
                     2072        MOV    A,D
                     2073        ORA    E      SEE IF DE IS ZERO.
                     2074        LHLD   BLOCK  GET BLOCK SIZE
                     2075        XCHG   .      ...TO DE
                     2076 *  DE HAS HBLOCK....HL HAS USER OPTION
                     2077        JNZ    RTAP   IF DE WAS ZERO GET TAPE LOAD ADDRESS
                     2078        LHLD   LOADR  GET TAPE LOAD ADDRESS
                     2079 *
                     2080 *
                     2081 *     THIS ROUTINE READS "DE" BYTES FROM THE TAPE
                     2082 *     TO ADDRESS HL.  THE BYTES MUST BE FROM ONE
                     2083 *     CONTIGUOUS PHYSICAL BLOCK ON THE TAPE.
                     2084 *
                     2085 *          HL HAS "PUT" ADDRESS
                     2086 *          DE HAS SIZE OF TAPE BLOCK
                     2087 *
                     2088 RTAP   PUSH   D      SAVE SIZE FOR RETURN TO CALLING PROGRAM
                     2089 *
                     2090 RTAP2  EQU    $      HERE TO LOOP RDING BLKS
                     2091        CALL   DCRCT  DROP COUNT, R=LEN THIS BLK
                     2092        JZ     RTOFF  ZERO=ALL DONE
```

```
                        1693 *
C6FD CD 4A C7           1694         CALL    RHED1   READ THAT MANY BYTES
C700 DA 06 C7           1695         JC      TERR    IF ERROR OR ESC
C703 CA F7 C6           1696         JZ      RTAP2   RD OK--READ SOME MORE
                        1697 *
                        1698 *  ERROR RETURN
                        1699 *
C706 AF                 1700 TERR    XRA     A
C707 37                 1701         STC     .       SET ERROR FLAGS
C708 C3 11 C7           1702         JMP     RTOF1
                        1703 *
                        1704 *
C70B 06 01              1705 TOFF    MVI     B,1
C70D CD F1 C7           1706         CALL    DELAY
C710 AF                 1707 RTOFF   XRA     A
C711 D3 FA              1708 RTOF1   OUT     TAPPT
C713 D1                 1709         POP     D       RETURN BYTE COUNT
C714 C9                 1710         RET
                        1711 *
                        1712 *
     C715               1713 DCRCT   EQU     $       COMMON RTN TO COUNT DOWN BLK LENGTHS
C715 AF                 1714         XRA     A       CLR FOR LATER TESTS
C716 47                 1715         MOV     B,A     SET THIS BLK LEN=256
C717 B2                 1716         ORA     D       IS AMNT LEFT < 256
C718 C2 20 C7           1717         JNZ     DCRC2   NO--REDUCE AMNT BY 256
C71B B3                 1718         ORA     E       IS ENTIRE COUNT ZERO
C71C C8                 1719         RZ      ALL DONE--ZERO=THIS CONDITION
C71D 43                 1720         MOV     B,E     SET THIS BLK LEN TO AMNT REMAINING
C71E 5A                 1721         MOV     E,D     MAKE ENTIRE COUNT ZERO NOW
C71F C9                 1722         RET     .       ALL DONE (NON-ZERO FLAG)
     C720               1723 DCRC2   EQU     $       REDUCE COUNT BY 256
C720 15                 1724         DCR     D       DROP BY 256
C721 B7                 1725         ORA     A       FORCE NON-ZERO FLAG
C722 C9                 1726         RET     .       NON-ZERO=NOT DONE YET (BLK LEN=256)
                        1727 *
                        1728 *
                        1729 *    READ THE HEADER
                        1730 *
C723 06 0A              1731 RHEAD   MVI     B,10    FIND 10 NULLS
C725 CD 5D C7           1732 RHEA1   CALL    STAT
C728 D8                 1733         RC              IF ESCAPE
C729 DB FB              1734         IN      TDATA   IGNORE ERROR CONDITIONS
C72B B7                 1735         ORA     A       ZERO?
C72C C2 23 C7           1736         JNZ     RHEAD
C72F 05                 1737         DCR     B
C730 C2 25 C7           1738         JNZ     RHEA1   LOOP UNTIL 10 IN A ROW
                        1739 *
                        1740 *    WAIT FOR THE START CHARACTER
                        1741 *
C733 CD 6F C7           1742 SOHL    CALL    TAPIN
C736 D8                 1743         RC      .       ERROR OR ESCAPE
C737 FE 01              1744         CPI     1       AT LEAST 10 NULLS IMMEDIATELY FOLLOWED BY AN 01
C739 DA 33 C7           1745         JC      SOHL    STILL A NULL, KEEP WAITING
C73C C2 23 C7           1746         JNZ     RHEAD   NON-ZERO, START SEQUENCE OVER AGAIN
                        1747 *
                        1748 *    NOW GET THE HEADER
                        1749 *
C73F 21 1C C8           1750         LXI     H,THEAD  POINT TO BUFFER
C742 06 10              1751         MVI     B,HLEN  LENGTH TO READ
                        1752 *
     C744               1753 RHED1   EQU     $       RD A BLOCK INTO HL FOR B BYTES
C744 0E 00              1754         MVI     C,0     INIT THE CRC
     C746               1755 RHED2   EQU     $       LOOP HERE
C746 CD 6F C7           1756         CALL    TAPIN   GET A BYTE
C749 D8                 1757         RC
C74A 77                 1758         MOV     M,A     STORE IT
C74B 23                 1759         INX     H       INCREMENT ADDRESS
C74C CD A8 C7           1760         CALL    DOCRC   GO COMPUTE THE CRC
C74F 05                 1761         DCR     B       WHOLE HEADER YET?
C750 C2 46 C7           1762         JNZ     RHED2   DO ALL THE BYTES
                        1763 *
                        1764 *   THIS ROUTINE GETS THE NEXT BYTE AND COMPARES IT
                        1765 * TO THE VALUE IN REGISTER C.  THE FLAGS ARE SET ON
                        1766 * RETURN.
                        1767 *
C753 CD 6F C7           1768         CALL    TAPIN GET CRC BYTE
C756 A9                 1769         XRA     C       CLR CARRY AND SET ZERO IF MATCH, ELSE NON-ZERO
C757 C8                 1770         RZ      .       CRC WAS FINE
C758 3A 11 C8           1771         LDA     IGNCR   GET POSSIBLE OVERRIDE CRC ERROR FLAG
C75B 3C                 1772         INR     A       FF=IGNORE CRC ERRORS, ELSE PROCESS CRC ERROR
C75C C9                 1773         RET
                        1774 *
                        1775 *    THIS ROUTINE GETS THE NEXT AVAILABLE BYTE FROM THE
                        1776 *  TAPE.  WHILE WAITING FOR THE BYTE THE KEYBOARD IS TESTED
                        1777 *  FOR AN ESC COMMAND.  IF RECEIVED THE TAPE LOAD IS
                        1778 *  TERMINATED AND A RETURN TO THE COMMAND MODE IS MADE.
                        1779 *
C75D DB FA              1780 STAT    IN      TAPPT  TAPE STATUS PORT
C75F E6 40              1781         ANI     TDR
C761 C0                 1782         RNZ
C762 CD 1F C0           1783         CALL    SINP    CHECK INPUT
C765 CA 5D C7           1784         JZ      STAT    NOTHING THERE YET
C768 E6 7F              1785         ANI     7FH     CLR PARITY 1ST
C76A C2 5D C7           1786         JNZ     STAT    NOT A MODE (OR EVEN CTL-@)
C76D 37                 1787         STC     .       SET ERROR FLAG
C76E C9                 1788         RET     .       AND RETURN
                        1789 *
                        1790 *
                        1791 *
C76F CD 5D C7           1792 TAPIN   CALL    STAT    WAIT UNTIL A CHARACTER IS AVAILABLE
C772 D8                 1793         RC
                        1794 *
C773 DB FA              1795 TREDY   IN      TAPPT  TAPE STATUS
C775 E6 18              1796         ANI     TFE+TOE  DATA ERROR?
C777 DB FB              1797         IN      TDATA  GET THE DATA
C779 C8                 1798         RZ      .       IF NO ERRORS
C77A 37                 1799         STC     .       SET ERROR FLAG
C77B C9                 1800         RET
                        1801 *
                        1802 *
                        1803 *  THIS ROUTINE GETS THE CORRECT UNIT FOR SYSTEM WRITES
C77C CD DE C7           1804 WFBLK   CALL    GTUNT  SET UP A WITH UNIT AND SPEED
                        1805 *
                        1806 *
                        1807 *
                        1808 *       WRITE TAPE BLOCK ROUTINE
                        1809 *
                        1810 *   ON ENTRY:   A   HAS UNIT AND SPEED
                        1811 *               HL  HAS POINTER TO HEADER
                        1812 *
                        1813 *
     C77F               1814 WTAPE   EQU     $       HERE TO WRITE TAPE
C77F E5                 1815         PUSH    H       SAVE HEADER ADDRESS
C780 CD AF C7           1816         CALL    WHEAD  TURN ON, THEN WRITE HDR
C783 E1                 1817         POP     H
C784 11 07 00           1818         LXI     D,BLKOF  OFFSET TO BLOCK SIZE IN HEADER
C787 19                 1819         DAD     D       HL POINT TO BLOCK SIZE
C788 5E                 1820         MOV     E,M
C789 23                 1821         INX     H
C78A 56                 1822         MOV     D,M     DE HAVE SIZE
C78B 23                 1823         INX     H
C78C 7E                 1824         MOV     A,M
C78D 23                 1825         INX     H
C78E 66                 1826         MOV     H,M
C78F 6F                 1827         MOV     L,A     HL HAVE STARTING ADDRESS
                        1828 *
                        1829 *    THIS ROUTINE WRITES ONE PHYSICAL BLOCK ON THE
                        1830 *  TAPE "DE" BYTES LONG FROM ADDRESS "HL".
                        1831 *
                        1832 *
     C790               1833 WRLO1   EQU     $       HERE FOR THE EXTRA PUSH
C790 E5                 1834         PUSH    H       A DUMMY PUSH FOR LATER EXIT
```

```
     C791               1835 WTAP2   EQU     $       LOOP HERE UNTIL ENTIRE AMOUNT READ
C791 CD 15 C7           1836         CALL    DCRCT   DROP COUNT IN DE AND SET UP B W/LEN THIS
C794 CA 0B C7           1837         JZ      TOFF    RETURNS ZERO IF ALL DONE
C797 CD C3 C7           1838         CALL    WTBL    WRITE BLOCK FOR BYTES IN B (256)
C79A C3 91 C7           1839         JMP     WTAP2   LOOP UNTIL ALL DONE
                        1840 *
                        1841 *
C79D F5                 1842 WRTAP   PUSH    PSW
C79E DB FA              1843 WRWAT   IN      TAPPT  TAPE STATUS
C7A0 E6 80              1844         ANI     TTBE    IS TAPE READY FOR A CHAR YET
C7A2 CA 9E C7           1845         JZ      WRWAT   NO--WAIT
C7A5 F1                 1846         POP     PSW     YES--RESTORE CHAR TO OUTPUT
C7A6 D3 FB              1847         OUT     TDATA   SEND CHAR TO TAPE
     C7A8               1848 *
C7A8 91                 1849 DOCRC   EQU     $       A COMMON CRC COMPUTATION ROUTINE
C7A9 AF                 1850         SUB     C
C7AA A9                 1851         MOV     C,A
C7AB 2F                 1852         XRA     C
C7AC 91                 1853         CMA
C7AD AF                 1854         SUB     C
C7AE C9                 1855         MOV     C,A
                        1856         RET     .       ONE  BYTE NOW WRITTEN
                        1857 *
                        1858 *
                        1859 *   THIS ROUTINE WRITES THE HEADER POINTED TO BY
                        1860 *   HL TO THE TAPE.
                        1861 *
     C7AF               1862 WHEAD   EQU     $       HERE TO 1ST TURN ON THE TAPE
C7AF CD ED C7           1863         CALL    WTON    TURN IT ON, THEN WRITE HEADER
C7B2 16 32              1864         MVI     D,50    WRITE 50 ZEROS
C7B4 AF                 1865 NULOP   XRA     A
C7B5 CD 9D C7           1866         CALL    WRTAP
C7B8 15                 1867         DCR     D
C7B9 C2 B4 C7           1868         JNZ     NULOP
                        1869 *
C7BC 3E 01              1870         MVI     A,1
C7BE CD 9D C7           1871         CALL    WRTAP
C7C1 06 10              1872         MVI     B,HLEN  LENGTH TO WRITE OUT
                        1873 *
C7C3 0E 00              1874 WTBL    MVI     C,0     RESET CRC BYTE
C7C5 7E                 1875 WLOOP   MOV     A,M     GET CHARACTER
C7C6 CD 9D C7           1876         CALL    WRTAP   WRITE IT TO THE TAPE
C7C9 05                 1877         DCR     B
C7CA 23                 1878         INX     H
C7CB C2 C5 C7           1879         JNZ     WLOOP
C7CE 79                 1880         MOV     A,C     GET CRC
C7CF C3 9D C7           1881         JMP     WRTAP   PUT IT ON THE TAPE AND RETURN
                        1882 *
                        1883 *
                        1884 *   THIS ROUTINE COMPARES THE HEADER IN THEAD TO
                        1885 *   THE USER SUPPLIED HEADER IN ADDRESS HL.
                        1886 *   ON RETURN IF ZERO IS SET THE TWO NAMES COMPARED
                        1887 *
C7D2 06 05              1888 DHCMP   MVI     B,5
C7D4 1A                 1889 DHLOP   LDAX    D
C7D5 BE                 1890         CMP     M
C7D6 C0                 1891         RNZ
C7D7 05                 1892         DCR     B
C7D8 C8                 1893         RZ      .       IF ALL FIVE COMPARED
C7D9 23                 1894         INX     H
C7DA 13                 1895         INX     D
C7DB C3 D4 C7           1896         JMP     DHLOP
                        1897 *
     C7DE               1898 GTUNT   EQU     $       SET A=SPEED + UNIT
C7DE 3A 5A C8           1899         LDA     FNUMF   GET UNIT
C7E1 B7                 1900         ORA     A       SEE WHICH UNIT
C7E2 3A 0D C8           1901         LDA     TSPD    BUT 1ST GET SPEED
C7E5 C2 EA C7           1902         JNZ     GTUN2   MAKE IT UNIT TWO
C7E8 C6 A0              1903         ADI     TAPE2   THIS ONCE=UNIT 2, TWICE=UNIT 1
C7EA C6 A0              1904 GTUN2   ADI     TAPE2   UNIT AND SPEED NOW SET IN A
C7EC C9                 1905         RET     .       ALL DONE
                        1906 *
C7ED 06 0A              1907 WTON    MVI     B,A     SET LOOP DELAY  (BIT LONGER ON A W
     C7EF               1908 TON     EQU     $       HERE TO TURN A TAPE ON THEN DELAY
C7EF D3 FA              1909         OUT     TAPPT  GET TAPE MOVING, THEN DELAY
                        1910 *
C7F1 11 00 00           1911 DELAY   LXI     D,0
C7F4 1B                 1912 DLOP1   DCX     D
C7F5 7A                 1913         MOV     A,D
C7F6 B3                 1914         ORA     E
C7F7 C2 F4 C7           1915         JNZ     DLOP1
C7FA 05                 1916         DCR     B
C7FB C2 F1 C7           1917         JNZ     DELAY
C7FE C9                 1918         RET
                        1919 *
                        1920 *
                        1921 ***** -- END OF PROGRAM--
                        1922 *
                        1923 *
                        1924 *
                        1925 *
                        1926 *    S Y S T E M    E Q U A T E S
                        1927 *
                        1928 *
                        1929 *          VDM PARAMETERS
                        1930 *
     CC00               1931 VDMEM   EQU     0CC00H  VDM SCREEN MEMORY
                        1932 *
                        1933 *
                        1934 *          KEYBOARD SPECIAL KEY ASSIGNMENTS
                        1935 *
                        1936 *  THESE DEFINITIONS ARE DESIGNED TO ALLOW
                        1937 *  COMPATABILITY WITH CUTER(TM).  THESE ARE THE
                        1938 *  SAME KEYS WITH BIT 7 (X'80') STRIPPED OFF.
                        1939 *
     009A               1940 DOWN    EQU     9AH     CTL-Z
     0097               1941 UP      EQU     97H     CTL-W
     0081               1942 LEFT    EQU     81H     CTL-A
     0093               1943 RIGHT   EQU     93H     CTL-S
     008B               1944 CLEAR   EQU     8BH     CTL-K
     008E               1945 HOME    EQU     8EH     CTL-N
     0080               1946 MODE    EQU     80H     CTL-@
     005F               1947 BACKS   EQU     5FH     BACKSPACE
     000A               1948 LF      EQU     10
     000D               1949 CR      EQU     13.
     0020               1950 BLANK   EQU
     0020               1951 SPACE   EQU     BLANK
     0018               1952 CX      EQU     'X'-40H
     001B               1953 ESC     EQU     1BH
                        1954 *
                        1955 *          PORT ASSIGNMENTS
                        1956 *
     00FA               1957 STAPT   EQU     0FAH    STATUS PORT GENERAL
     00F8               1958 SERST   EQU     0F8H    SERIAL STATUS PORT
     00F9               1959 SDATA   EQU     0F9H    SERIAL DATA
     00FD               1960 PDATA   EQU     0FDH    PARALLEL DATA
     00FC               1961 KDATA   EQU     0FCH    KEYBOARD DATA
     00FE               1962 DSTAT   EQU     0FEH    VDM CONTROL PORT
     00FA               1963 TAPPT   EQU     0FAH    TAPE STATUS PORT
     00FB               1964 TDATA   EQU     0FBH    TAPE DATA PORT
     00FF               1965 SENSE   EQU     0FFH    SENSE SWITCHES
                        1966 *
                        1967 *
                        1968 *
                        1969 *          BIT ASSIGNMENT MASKS
                        1970 *
     0001               1971 SCD     EQU     1       SERIAL CARRIER DETECT
     0002               1972 SDSR    EQU     2       SERIAL DATA SET READY
     0004               1973 SPE     EQU     4       SERIAL PARITY ERROR
     0008               1974 SFE     EQU     8       SERIAL FRAMING ERROR
     0010               1975 SOE     EQU     16      SERIAL OVERRUN ERROR
     0020               1976 SCTS    EQU     32      SERIAL CLEAR TO SEND
```

```
00R0            1977 SDR    EQU   6A     SERIAL DATA READY
0080            1978 STBE   EQU   128    SERIAL TRANSMITTER BUFFER EMPTY
                1979 *
0001            1980 KDR    EQU   1      KEYBOARD DATA READY
0002            1981 PDR    EQU   2      PARALLEL DATA READY
0004            1982 PXDR   EQU   4      PARALLEL DEVICE READY
0008            1983 TFE    EQU   8      TAPE FRAMING ERROR
0010            1984 TOE    EQU   16     TAPE OVERFLOW ERROR
0040            1985 TDR    EQU   6A     TAPE DATA READY
0080            1986 TTBE   EQU   128    TAPE TRANSMITTER BUFFER EMPTY
                1987 *
0001            1988 SOK    EQU   1      SCROLL OK FLAG
                1989 *
0080            1990 TAPE1  EQU   80H    1=TURN TAPE ONE ON
0040            1991 TAPE2  EQU   40H    1=TURN TAPE TWO ON
                1992 *
                1993 *
                1994 *
                1995 *
                1996 *      S Y S T E M   G L O B A L   A R E A
                1997 *
C800            1998        ORG   START+0800H   RAM STARTS JUST AFTER ROM
                1999 *
C800            2000 SYSRAM EQU   $      START OF SYSTEM RAM
CBFF            2001 SYSTP  EQU   SYSRAM+3FFH  STACK WORKS FM TOP DOWN
                2002 *
                2003 *
                2004 *   PARAMETERS STORED IN RAM
                2005 *
C800            2006 UIPRT  DS    2      USER DEFINED INPUT RTN IF NON ZERO
C802            2007 UOPRT  DS    2      USER DEFINED OUTPUT RTN IF NON ZERO
C804            2008 DFLTS  DS    2      DEFAULT PSUEDO I/O PORTS (ALWAYS ZERO IN SOLOS)
C806            2009 IPORT  DS    1      CRNT INPUT PSUEDO PORT
C807            2010 OPORT  DS    1      CRNT OUTPUT PSUEDO PORT
C808            2011 NCHAR  DS    1      CURRENT CHARACTER POSITION
C809            2012 LINE   DS    1      CURRENT LINE POSITION
C80A            2013 BOT    DS    1      BEGINNING OF TEXT DISPLACEMENT
C80B            2014 SPEED  DS    1      SPEED CONTROL BYTE
C80C            2015 ESCFL  DS    1      ESCAPE FLAG CONTROL BYTE
C80D            2016 TSPD   DS    1      CURRENT TAPE SPEED
C80E            2017 INPTR  DS    2      FOR COMPATABILITY W/ CUTER
C810            2018 NUCNT  DS    1      NUMBER OF NULLS AFTER CRLF
C811            2019 IGNCR  DS    1      FF=IGNORE CRC ERRORS, ELSE NORMAL
                2020 *
C812            2021        DS    10     ROOM FOR FUTURE EXPANSION
                2022 *
                2023 * * * * * * * * * * * * * * * * * * * * * * * * * * *
                2024 *   T H I S   I S   T H E   H E A D E R   L A Y O U T   *
                2025 * * * * * * * * * * * * * * * * * * * * * * * * * * *
                2026 *
C81C            2027 THEAD  DS    5      NAME
C821            2028        DS    1      THIS BYTE MUST BE ZERO
C822            2029 HTYPE  DS    1      TYPE
C823            2030 BLOCK  DS    2      BLOCK SIZE
C825            2031 LOADR  DS    2      LOAD ADDRESS
C827            2032 XEQAD  DS    2      AUTO EXECUTE ADDRESS
C829            2033 HSPR   DS    3      SPARES
                2034 *
0010            2035 HLEN   EQU   $-THEAD  LENGTH OF HEADER
0007            2036 BLKOF  EQU   BLOCK-THEAD  OFFSET TO BLOCK SIZE
C82C            2037 DHEAD  DS    HLEN   A DUMMY HDR FOR COMPARES WHILE RD'ING
                2038 *
                2039 *
C83C            2040 CUTAB  DS    6*A    ROOM FOR UP TO 6 CUSTOM USER COMMANDS
                2041 *
                2042 *
                2043 FNUMF  DS    1      FOR CURRENT FILE OPERATIONS
                2044 FCBAS  DS    7      1ST FILE CONTROL BLOCK
                2045 FCBA2  DS    7      2ND FILE CONTROL BLOCK
                2046 FBUF1  DS    2*256  SYSTEM FILE BUFFER BASE
                2047        DS    81     THIS IS AN AREA USED BY CUTER
CAB4            2048 USARE  EQU   $      START OF USER AREA
                2049 *   REMEMBER THAT THE STACK WORKS ITS WAY DOWN FROM
                2050 *   THE END OF THIS 1K RAM AREA.
                2051 *
                2052 * -*-
```

```
ADOUT   C3E8    AINP    C022    ALOAD   C548    AOUT    C01C
ARET    C19B    ARET1   C19D    ARET2   C1A2    BACKS   005F
BLANK   0020    BLKOF   0007    BLOCK   C823    BOPEN   C5F0
BOT     C80A    BOUT    C406    CHAR    C094    CHPCK   C05E
CHRLI   C571    CLEAR   008B    CLERA   C1B4    CLIN1   C0FA
CLINE   C0F4    COMN1   C1C0    COMND   C1C9    COMTA   C29A
CONT    C1FF    COPRC   C205    CR      000D    CREM    C136
CRLF    C2F9    CUR     C0D1    CURET   C1A6    CURSC   C00F
CUSE2   C5D3    CUSET   C5BD    CUTAB   C83C    CX      0018
DCRC2   C720    DCRCT   C715    DEFLT   C49A    DELAY   C7F1
DFLTS   C804    DHCMP   C7D2    DHEAD   C82C    DHLOP   C7DA
DISPO   C222    DISP1   C22B    DISPD   C599    DISPT   C227
DLOOP   C3C8    DLOP1   C7FA    DLP1    C3D3    DOCRC   C7A8
DOWN    009A    DSTAT   00FE    DUMP    C3BF    ENLO1   C43C
ENLO3   C459    ENLOP   C42B    ENTER   C423    EOFER   C5FF
EOFW    C62B    ERAS1   C0DB    ERAS3   C0EE    ERR1    C480
ERR2    C481    ERRIT   C2CB    ERRM    C525    ERRO1   C2D6
ERROT   C2D2    ESC     001B    ESCFL   C80C    ESCS    C15F
ESCSP   C168    EXEC    C45E    EXEC1   C461    FBUF1   C863
FCBA2   C85C    FCBAS   C855    FCLOS   C00A    FDCOM   C231
FDCOU   C22E    FNUMF   C854    FOPEN   C007    GCLIN   C1EA
GOBAC   C06B    GOBK    C07C    GTBYT   C675    GTUN2   C7EA
GTUNT   C7DE    HBOUT   C3ED    HCONV   C3AD    HCOV1   C35D
HEOU1   C41A    HEOUT   C40B    HLEN    0010    HOME    008E
HSPR    C829    HTYPE   C822    IGNCR   C811    INIT    C001
INPTR   C80E    IOPRC   C0C6    IPORT   C806    ITAB    C29A
KDATA   00FC    KDR     0001    KSTAT   C02E    LEFT    0081
LF      000A    LFCB    C633    LFCB1   C642    LINE    C809
LIST1   C539    LLIST   C531    LOADR   C825    MODE    0080
NAME    C469    NAME1   C46E    NAMES   C466    NAOUT   C550
NCHAR   C808    NCOM    C243    NEXT    C080    NFIL    C486
NLOOP   C56A    NUCNT   C810    NULOP   C7BA    NULOT   C307
OCHAR   C098    OK      C0C1    OPORT   C807    OTAB    C292
OUTH    C41F    OUTPR   C03B    PASTA   C2DD    PBACK   C13E
PCLOS   C603    PCR     C147    PCUR    C10F    PDATA   00FD
PDOWN   C0CB    PDR     0002    PERSE   C0D5    PESC    C159
PHEAD   C6A6    PHOME   C0E5    PLEFT   C10B    PLF     C14D
PLOAD   C6BF    PRIT    C115    PROMP   C2F1    PROUT   C2E6
PSCAN   C310    PSTOR   C6B6    PTAP1   C6D3    PUP     C10A
PXDR    000A    RDBLK   C013    RDBYT   C00D    RDNBL   C65C
RETRN   C00A    RFBLK   C6CB    RHEA1   C725    RHEAD   C723
RHED1   C744    RHED2   C746    RIGHT   0093    RT1     C680
RTAP    C6F6    RTAP2   C6F7    RTAPE   C6CB    RTBYT   C6A6
RTOF1   C711    RTOFF   C710    SBLK    C31B    SBLK1   C31D
SCD     0001    SCHR    C32E    SCHR1   C330    SCONV   C33A
SCROL   C0AC    SCTS    0020    SDATA   00F9    SDR     0040
SDROT   C04A    SDSR    0002    SECON   C190    SENSE   00FF
SERST   00F8    SET     C57A    SETAB   C2A2    SETCI   C5A5
SETCO   C5A9    SETCR   C5B9    SETIN   C59D    SETNU   C5B5
SETOT   C5A1    SETSP   C59A    SETTY   C5AD    SETX    C188
SETXQ   C5B1    SETY    C18C    SFE     0008    SHE1    C343
SUPY    C340    SINP    C01F    SOE     0010    SOHL    C733
        0001    SOUT    C019    SPACE   0020    SPE     0004
        C80B    SROL    C0B0    SSTAT   C042    STAPT   00FA
        C000    STAT    C75D    STBE    0080    STRTA   C1AF
        C598    STUNT   C4A2    SYSRA   C800    SYSTP   CBFF
        C514    TAPE1   0080    TAPE2   0040    TAPIN   C76F
        00FA    TASPD   C5BE    TBL     C273    TDATA   00FB
        0040    TERE0   C5FC    TERE1   C5FB    TERE2   C5FA
TERM    C367    TERM1   C373    TERM2   C3B9    TERR    C706
TFE     0008    THEAD   C81C    TIMER   C077    TIN     C3BB
TLIST   C52B    TLOA2   C4B5    TLOA3   C4C1    TLOAD   C4A7
```

```
TOE     0010    TOFF    C70B    TON     C7EF    TOUT    C3AB
TREDY   C773    TSAVE   C4E6    TSPD    C80D    TSRCH   C082
TTBE    0080    TXEQ    C4A6    UBUF    C5F5    UIPRT   C800
UOPRT   C802    UP      0097    USARE   CAB4    VDAD    C123
VDAD2   C120    VDADD   C11C    VDMEM   CC00    VDMOT   C054
WFBLK   C77C    WHEAD   C7AF    WLOOP   C7C5    WRBLK   C015
WRBYT   C010    WRL01   C790    WRTAP   C79D    WRWAT   C79F
WT1     C69B    WTAP2   C791    WTAPE   C77F    WTRL    C7C3
WTBYT   C683    WTLP1   C400    WTON    C7ED    XEQAD   C827
```


Processor Technology Dealers

ALABAMA

ICP, Computerland
1550 Montgomery Hwy
Birmingham, AL 35226
(205) 979-0707

ARIZONA

Byte Shop Tempe
813 N. Scottsdale Rd.
Tempe, AZ 85281
(602) 894-1129

Byte Shop Phoenix
12654 N. 28th Dr.
Phoenix, AZ 85029
(602) 942-7300

Byte Shop Tucson
2612 E. Broadway
Tucson, AZ 85716
(602) 327-4579

CALIFORNIA

The Byte Shop
1514 University Ave.
Berkeley, CA 94703
(415) 845-6366

Byte Shop Computer Store
6041 Greenback Lane
Citrus Heights, CA 95610
(916) 961-2983

Computer Center
1913 Harbor Blvd.
Costa Mesa, CA 92627
(714) 646-0221

Data Consultants, Inc.
2350 W. Shaw, Suite 114
Fresno, CA 93711
(209) 431-6461

Bits 'N Bytes
679 S. State College Blvd.
Fullerton, CA 92631
(714) 879-8386

The Byte Shop
16508 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-2421

The Byte Shop
1063 El Camino Real
Mountain View, CA 94040
(415) 969-5464

Digital Deli
80 W. El Camino Real
Mountain View, CA 94040
(415) 961-2828

The Computer Mart
624 West Katella #10
Orange, CA 92667
(714) 633-1222

The Byte Shop
2227 El Camino Real
Palo Alto, CA 94306
(415) 327-8080

Byte Shop
496 South Lake Ave.
Pasadena, CA 91101
(213) 684-3311

The Computer Store
of San Francisco
1093 Mission Street
San Francisco, CA 94103
(415) 431-0640

Byte Shop
321 Pacific Ave.
San Francisco, CA 94111
(415) 421-8686

The Byte Shop
2626 Union Avenue
San Jose, CA 95124
(408) 377-4685

The Computer Room
124H Blossom Hill Rd.
San Jose, CA 95123
(408) 226-8383

The Byte Shop
509 Francisco Blvd.
San Rafael, CA 94901
(415) 457-9311

The Byte Shop
3400 El Camino Real
Santa Clara, CA 95051
(408) 249-4221

Recreational Computer
Centers
1324 South Mary Ave.
Sunnyvale, CA 94087
(408) 735-7480

Byte Shop of Tarzana
18424 Ventura Blvd.
Tarzana, CA 91356
(213) 343-3919

The Byte Shop
2989 North Main St.
Walnut Creek, CA 94596
(415) 933-6252

Byte Shop
14300 Beach Blvd.
Westminster, CA 92683
(714) 894-9131

COLORADO

Byte Shop
2040 30th St.
Boulder, CO 80301
(303) 449-6233

FLORIDA

Sunny Computer Stores
University Shopping
Center
1238A S. Dixie Hwy.
Coral Gables, FL 33146
(305) 661-6042

Delta Electronics
2000 U.S. Hwy. 441 East
Leesburg, FL 32748
(904) 357-4244

Byte Shop of Miami
7825 Bird Road
Miami, FL 33155
(303) 264-2983

Microcomputer
Systems Inc.
144 So. Dale Mabry Hwy.
Tampa, FL 33609
(813) 879-4301

GEORGIA

Atlanta Computer Mart
5091-B Buford Hwy.
Atlanta, GA 30340
(404) 455-0647

ILLINOIS

The Numbers Racket
623½ South Wright St.
Champaign, IL 61820
(217) 352-5435

itty bitty machine co.
1316 Chicago Ave.
Evanston, IL 60201
(312) 328-6800

Reeves Communications
1550 W. Court St.
Kankakee, IL 60901
(815) 937-4516

itty bitty machine co.
42 West Roosevelt
Lombard, IL 60148
(312) 620-5808

INDIANA

The Data Domain
406 So. College Ave.
Bloomington, IN 47401
(812) 334-3607

The Byte Shop
5947 East 82nd St.
Indianapolis, IN 46250
(317) 842-2983

The Data Domain
7027 N. Michigan Rd.
Indianapolis, IN 46268
(317) 251-3139

The Data Domain
219 West Columbia
West Lafayette, IN 47905
(317) 743-3951

KENTUCKY

The Data Domain
3028 Hunsinger Lane
Louisville, KY 40220
(502) 456-5242

MICHIGAN

The Computer Store
of Ann Arbor
310 East Washington
Ann Arbor, MI 48104
(313) 995-7616

Computer Mart
of Royal Oak
1800 W. 14 Mile Rd.
Royal Oak, MI 48073
(313) 576-0900

Genral Computer Store
2011 Livernois
Troy, MI 48084
(313) 362-0022

NEW JERSEY

Hoboken Computer Works
No. 20 Hudson Place
Hoboken, NJ 07030
(201) 420-1644

The Computer Mart
of New Jersey
501 Route 27
Iselin, NJ 08830
(201) 283-0600

NEW YORK

The Computer Mart
of Long Island
2072 Front Street
East Meadow, L.I.,
NY 11554
(516) 794-0510

Synchro Sound
Enterprises
193-25 Jamaica Ave.
Hollis, NY 11423
(212) 359-1489

The Computer Shoppe
444 Middle Country Rd.
Middle Island, NY 11953

Audio Design Electronics
487 Broadway, Ste. 512
New York, NY 10013
(212) 226-2038

The Computer Mart
of New York
118 Madison Ave.
New York, NY 10001
(212) 686-7923

The Computer Corner
200 Hamilton Ave.
White Plains, NY 10601
(914) 949-3282

OHIO

Cybershop
1451 S. Hamilton Rd.
Columbus, OH 43227
(614) 239-8081

OKLAHOMA

High Technology
1020 West Wilshire Blvd.
Oklahoma City, OK 73116
(405) 842-2021

OREGON

Byte Shop Computer Store
3482 S.W.
Cedar Hills Blvd.
Beaverton, OR 97005
(503) 644-2686

The Real Oregon
Computer Co.
205 West 10th Ave.
Eugene, OR 97401
(503) 484-1040

Byte Shop Computer Store
2033 S.W. 4th Ave.
Portland, OR 97201
(503) 223-3496

RHODE ISLAND

Computer Power, Inc.
M24 Airport Mall
1800 Post Rd.
Warwick, RI 02886
(401) 738-4477

SOUTH CAROLINA

Byte Shop
2018 Green Street
Columbia, SC 29205
(803) 771-7824

TENNESSEE

Microproducts & Systems
2307 E. Center St.
Kingsport, TN 37664
(615) 245-8081

TEXAS

Byte Shop
3211 Fondren
Houston, TX 77063
(713) 977-0664

Computertex
2300 Richmond Ave.
Houston, TX 77098
(713) 526-3456

Interactive Computers
7646½ Dashwood Rd.
Houston, TX 77036
(713) 772-5257

The Micro Store
634 So. Central
Expressway
Richardson, TX 75080
(214) 231-1096

VIRGINIA

The Computer Systems
Store
1984 Chain Bridge Rd.
McLean, VA 22101

Media Reactions Inc.
11303 South Shore Dr.
Reston, VA 22090
(703) 471-9330

WASHINGTON

Byte Shop Computer Store
14701 N.E. 20th Ave.
Bellevue, WA 98007
(206) 746-0651

The Retail Computer Store
410 N.E. 72nd
Seattle, WA 98115
(206) 524-4101

WISCONSIN

The Milwaukee
Computer Store
6916 W. North Ave.
Milwaukee, WI 53213
(414) 259-9140

CANADA

Trintronics
160 Elgin St.
Place Bell Canada
Ottawa, Ontario K2P 2C4
(613) 236-7767

First Canadian Computer
Store, Ltd.
44 Eglinton Ave. West
Toronto, Ontario M4R 1A1
(416) 482-8080

The Computer Place
186 Queen St. West
Toronto, Ontario M5V 1Z1
(416) 598-0262

Pacific Computer Store
4509-11 Rupert St.
Vancouver, B.C. V5R 2J4
(604) 438-3282

Dear sir,
I am sending this picture to ycu.

Name: Brad Burnett
first last
age: 8=9
grade: 3rd

My father's name:

Edward S. Miller

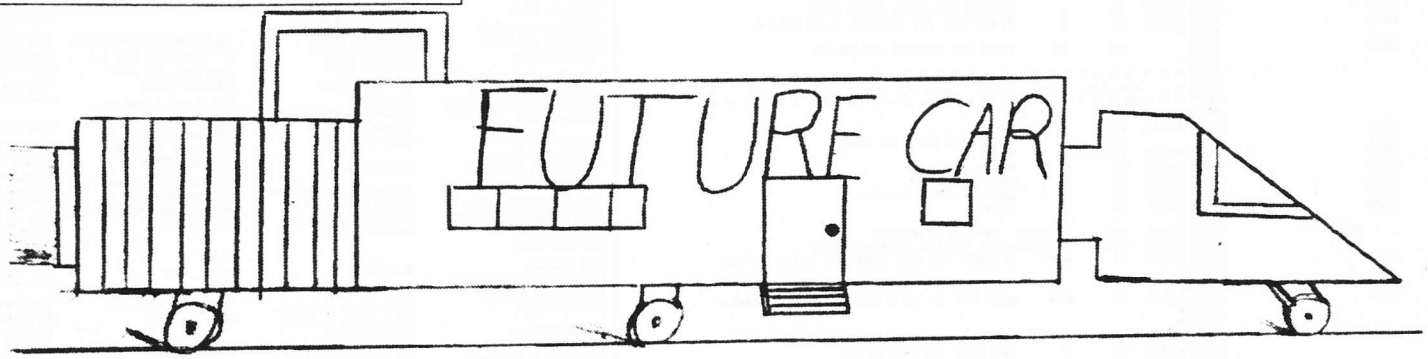
(I have a new father)

1 EPT 2 - 50

Computer Art Museum

Brad Burnett
235 Dodd Ave.
Mt. View. Cal. 94043

Processor Technology Corp
6200 Hollis Stree
Emeryville, Cal. 9460[illegible].



PROCESSOR TECHNOLOGY
ACCESS.

Vol. 1, No. 3

Bulk Rate
U.S. Postage
PAID
Permit No. 3729
Oakland, Calif.

Processor Technology Corp.
6200 Hollis Street
Emeryville, CA 94608

Stanley M. Sokolow, DDS
1577 Canada Lane
Woodside, CA 94062

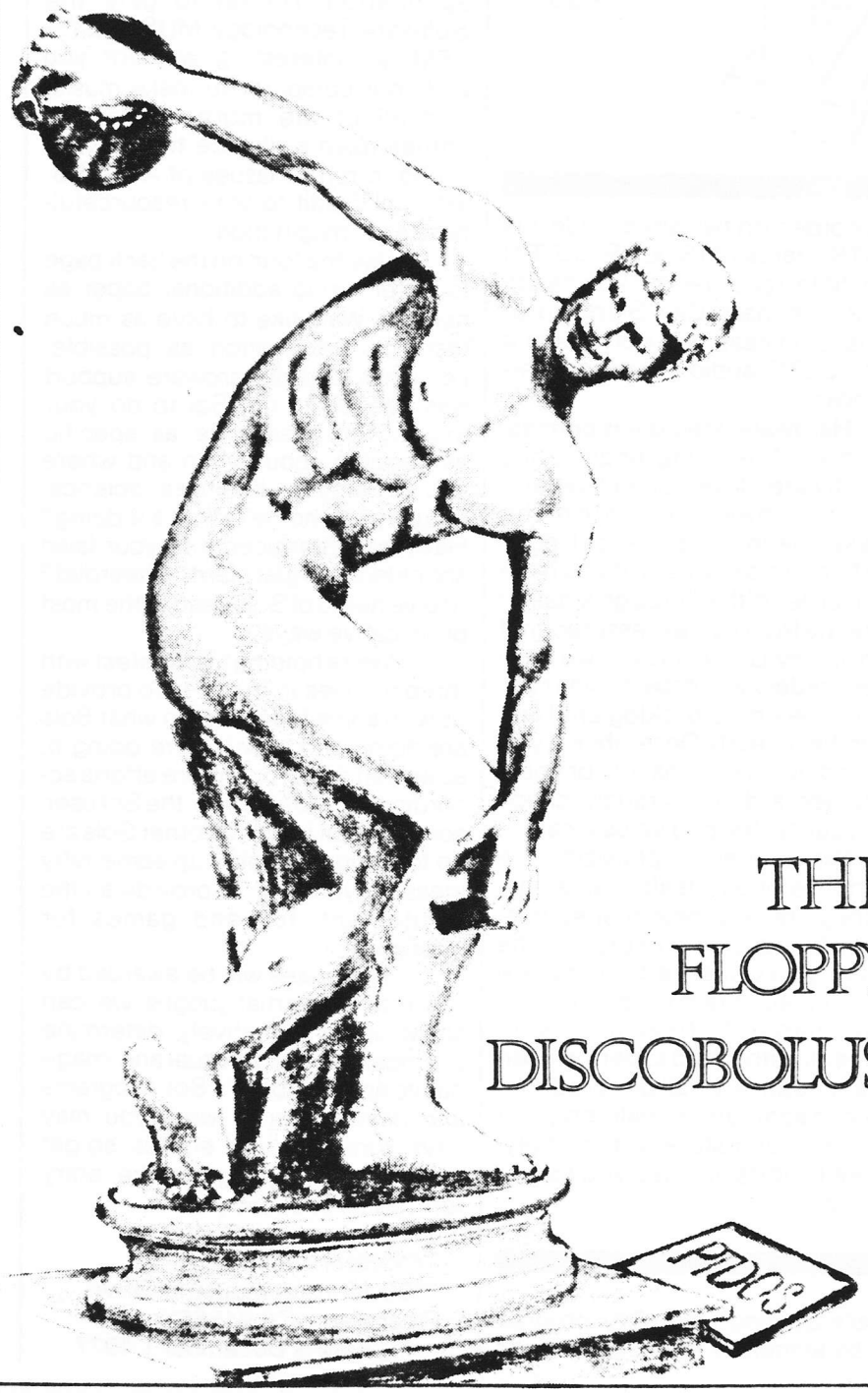
1 EPT 2 - 51

PROCESSOR TECHNOLOGY

ACCESS.

75¢ per issue

Published by Processor Technology Corp., Pleasanton. Volume One, Number Four, November 1977



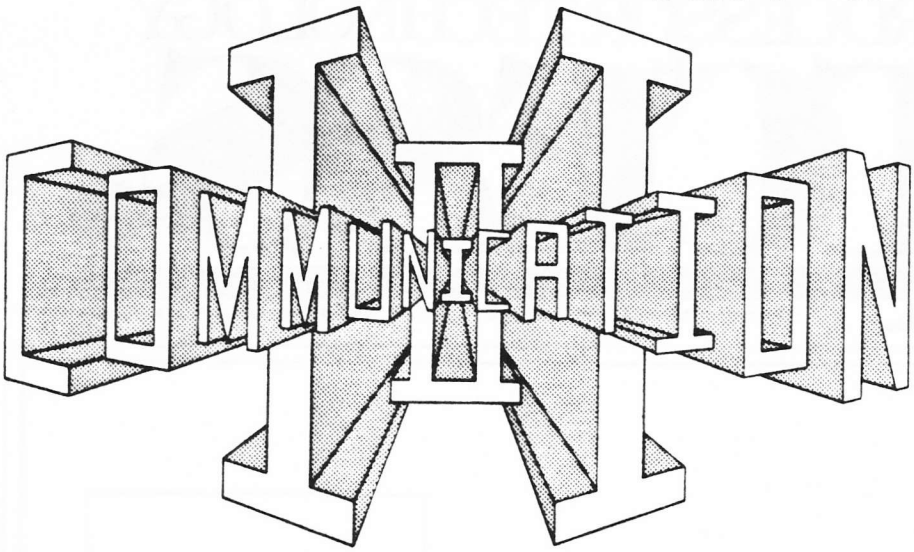
THE FLOPPY DISCOBOLUS

Helios II Reviewed

WINZI: Hic, Opps & Chase

Modified Bytesaver for Sol

Newett Awl's Choo Choo Update for Sol



The feedback is starting to come in at a most gratifying rate, so we'll get on to the news of note after a brief commercial for our technical troubleshooting hotline—call 415-829-2600 weekdays between 9:30 and noon or 1:30 to 4.

Software availability—good news time. Are you ready? A good portion of Processor Technology Software is. Specifically, TREK-80, ALS-8, GAMEPAC, and BASIC/5 (all on the most-wanted list) are now available and in stock at your local dealer. No paper tape versions, though—we're no longer producing any Processor Technology software on paper tape.

As for other software, you can currently look for SW#1 (Assembler) and MATHPAK in mid to late October; New 8080 FOCAL and 8K, and extended DISK BASIC on November 15. Incidentally, the two BASICS will have some capabilities we haven't advertised previously, such as complete matrix operations including matrix addition, subtraction, multiplication and inversion. Sure wish I'd had a Sol with 8K BASIC when I took linear algebra!

And look for some new software surprises turning up soon at your local dealer. We do appreciate your patience in waiting for software production and availability, and you won't be disappointed with the final product.

Software cassettes. Just a reminder that all our audio cassettes are recorded on two sides. Side 1 is in CUTS Standard (SOLOS/CUTER compatible, recorded at 1200 baud); Side 2 is Kansas City Standard at 300 baud, in case you don't have a Sol or CUTS audio cassette interface board.

Hardware: Helios II is coming. After many frustrating engineering and software development delays, deliveries have begun on the Processor Technology Helios II dual-drive floppy disk system. If you have one on order, either through a dealer or directly from us, rest assured that it's on its way. Unfortunately, we have so many orders we probably won't get through the entire backlog until late November or early December. If you want additional information, or you'd like to see a demonstration, check your local dealer, or give us a call.

We've been emphasizing the importance of our dealers all along, and they are your best first source. They're far more able to provide the service you deserve than we are from our distant factory, and they're always happy to hear from you. They're all authorized to perform service and repairs on all our products, and we're adding new dealers rapidly. Retail computer stores are multiplying like rabbits, in case you hadn't noticed.

Lloyd's Second Law: Any program can be written in fewer commands.

ENTER THE GREAT SHOW-OFF-YOUR-SOL CONTEST

and maybe win a GPM/Sol with ALS-8 SIM-1/TXT-2 ROM set

All you have to do is tell us how you are using your Sol. The grand prize will be awarded for the most interesting and/or unusual application; runner-up gets the Software Technology MUSIC SYSTEM, an interesting way for you and your computer to make music. And all of the more interesting entries have a chance to get published in future issues of ACCESS, with full credit to your resourcefulness and imagination.

Use the form on the back page to enter, using additional paper as needed. We'd like to have as much technical information as possible; i.e., what kind of hardware support was needed to get Sol to do your thing? And please be as specific as possible about when and where you're using it—business, science, industry? At home? What's it doing? Have you interfaced it to your lawn sprinklers, burglar alarm, Chevrolet? (We've heard of Sols used in the most provocative ways!)

We're holding this contest with three motives in mind. 1) To provide us with some feedback on what Sols are doing and how they're doing it, so we can direct our future efforts accordingly. 2) To give you, the Sol user, some insight into what other Sols are up to so you can pick up some nifty ideas for yours. 3) To provide a little excitement, fun and games for everyone.

The prizes will be awarded by the most impartial judges we can scout up to objectively determine the most original, unusual and imaginative application of a Sol. Programs and developmental work you may have done are valid entries, so get those pencils going on the entry form. Mail your entry to:

Sol Applications Contest
Processor Technology
7100 Johnson Industrial Drive
Pleasanton, CA 94566

Closing date is December 1, 1977.

Table of Contents

Reader Feedback . . . . . page 4

Sol User's Society gets into action with lots of plans and they're ready for input . . . . . page 6

Colgate Spinx opinionates re the software flexibility of Helios II . . . . . page 7

WINZI?!! A collection of short programs, featuring the likes of Hic, Opps, and Chase. . . . . page 8

Save your Sol's bytes with just a few little changes . . . . . page 13

Some uppercase key software to save wear and tear on your patience whenever you restart or power up Sol. . . . . page 14

How to hear your Sol keyboard, and shape up your touchkey typing . . . . page 15

Run/Stop, Part II, tells you more about monitoring the operation of your Sol. . . . . page 15

PLUS: A program for Home Accounting (p. 17), an 8080 Relocatable Assembler (p. 20), an ALS-8 to Sol Patch (p. 22), an Intel Tape Loader for Sol (p. 23), Newett Awl's Choo Choo Updated for Sol (p. 25), and Galaxy (p. 29).

A Letter From the Editor

"Any minor of order *r* in the adjugate determinant of a square matrix *A* is, on expansion, a certain polynomial in the elements of *A*, fixed in form whether *A* is singular or note."

Determinants and Matrices

"There are no Jewish midgets."
Lenny Bruce

Well, how do you like our new format? Now you can keep our deathless prose (and occasionally valuable information) neatly in a 3-ring binder instead of stacked up on the floor someplace to hide the cigarette burns. (Oh, you **liked** having something to hide the cigarette burns?) It should prove a more useful reference format, and we'll keep working on making ACCESS responsive to your needs. May even get classy soon and go to two-color.

Notes from the show circuit: We had one display at NCC (Dallas) and another at WESCON (San Francisco). Seems like computer shows are cropping up everywhere you look, and unfortunately we just don't have time to keep up with them all. It's great to see so much interest developing though—remember back to the dark ages when you were the only computer freak in town, all of a couple of years ago?

Meanwhile, instead of going to shows we're keeping our noses to the grindstone. The first Helios shipment is out and more are on the way. ALS-8 on CUTS cassette is now shipping, and 8K BASIC is set for mid-November. Ralph has more to say about new products, so take a look at his 1-to-1 column.

By the way, if you send us software contributions, try to give them to us on CUTS cassette. That way we can run out a listing, send it off to the printer, and it turns up in the next ACCESS in a neat standard format just the way you wrote it. If we have to have the listing typeset, there's just that much more chance of an error creeping in and turning your beautiful program into a debugger's nightmare. We are getting some nifty programs from you people—take a look at Guy Campbell's home accounting system and Melvin Schehlein's modification to avoid erasing input lines, both in this issue.

We're always interested in all forms of communication with the outside world. If you are a member of a club, or publish a newsletter yourself, I'd really like to hear about it. We might be able to exchange membership or subscription lists. That way we'd all be getting more information, and know more about each other.

Well, bye now—y'all come see us at our new home in Pleasanton, y'heah?

Aram Attarian II

Subscription Information

Access is published every six weeks. If you like what you see, we hope you'll send us $4.00 for a year's subscription so we can keep the info coming. Write to us at Processor Technology, 7100 Johnson Industrial Way, Pleasanton, CA 94566.

Have YOU Moved?

Please notify us of your change of address. Here's a handy form:

NAME_____

NEW ADDRESS_____

1 EPT 2 - 54

READER FEEDBACK

Gentlemen:

Having picked up a copy of #2 ACCESS at the N.C.C., I've been delighted at your fresh style of presentation.

The Texas A & M Microcomputer Club is composed of a number (about 50) of micro-computer buffs who are mainly software-development oriented. If you visited the Personal Computing Faire at N.C.C. you may have seen the APL-Core booth which is a club project. Robert Arnstein and Ian Kettleborough are former members of the club.

I was wondering if it would be possible for you to send us a few copies of ACCESS whenever it is published for distribution to interested club members. MITS does this with Computer Notes, which, of course is putting the advertising into a spot where it may potentially do the most good. However Computer Notes is rather dry on occasion.

The SOL system has been demonstrated a number of times for the club and it always draws quite a bit of interest. Several members of the faculty in various computer-related disciplines are considering a purchase.

Congratulations on a very readable publication.

Sincerely,
Robert R. Weir
Summer Caretaker,
TAMUCC

Dear Bob:

Thanks for the strokes. We'll put you on our mailing list right away, and we'll be glad to do the same for any other clubs who care to drop us a line.

Aram

Gentlemen:

As the satisfied user of two Sol systems I have encountered a problem when using the device as a terminal. The problem occurs because of the PCR routine in the VDM section of SOLOS. The attached assembly listing should be self explanatory as a successful solution. (See p. 20, this issue.)

Secondly, how come your company wasn't directly represented at Atlantic City this year? I know by talking to friends in the computer hobby field that you are having great difficulty delivering some of your new product line. Your equipment is of such high quality across the board most of us are willing to wait. Do you have something to hide or did you opt for Boston?

Yours truly,
Melvin E. Schehlein,
Ass't. Director of CMS

Dear Mel:

Thanks for a good idea: being able to check the last input certainly helps. Your program appears on page 20 of this issue. As for the shows, the cost of hitting all the ones cropping up on the East Coast especially is just prohibitive. We wouldn't have any money left to produce Sol's with. But we will make the New York City show in October. Right about now in fact.

Aram

Dear Aram:

I recently purchased, assembled and am running a SOL-20 under SOLOS. As a first trial exercise (having only 4K), I entered Newett Awl's Choo Choo Train. After much gnashing of teeth, wringing of hands and tracking of unstructured programming, I got it running. The trick was in the instruction in location 0156H; instead of an "IN 0" (DB00), SOLOS requires an "In FC" (DB FC). This is in addition to the necessary "C3 C9C1" in 0163H to return to SOLOS.

My purpose in writing this letter is not so much to relay the above software changes as to point out where I think ACCESS could play a large part in information dissemination. I received very little information with my SOL-20, regarding how to use the various devices with SOL (tape recorders, etc.). After scanning through the SOL manual and SOLOS listing, I found the skeleton references to the memory map and port addresses.

I could not locate any information of the complete hex-to-VDM character set, including all the obscure characters like ♃ and ≡. How about some examples of programs using the subroutine calls to SOLOS, particularly for animation on the VDM? In closing, I would like to compliment you on the publication of ACCESS. It is reassuring to have some further "feed-forward" from the vendor once the hardware has been sent. Keep up the good work!

Yours truly,
Warren L. Harkness
(In SOLOS UNUM)

Warren:

Thanks for the feedback, and we will keep supplementing the manual with helpful tips via ACCESS. Also, this issue has the Choo Choo listing modified to run on Sol.

Aram

To: Processor Technology Co.
ACCESS Editor

Thanks for a fine publication and outstanding products! Thought you might include this modification to the Sol system in ACCESS.

I am presently stationed in Japan with the U.S. Air Force. The

domestic power here is on the European standard, that is 50hz at 100 volts. Now the SOL-20 power supply doesn't seem to mind that at all, bus voltages are right up there, but the "swim" effect on the display could give you a splitting headache in short order.

The answer to my problem was actually quite simple. I needed four more character rows during the blanked period of the display. To obtain the extra rows I changed the preset count of U62 during the high state of VDISP, the display blanked. This change was implemented by disconnecting pin 5 of U62 from the VDISP line and tying it to ground. This gives me a total of 16 displayed and 8 blanked character rows for a total of 24 rows with 312 scan lines, a close match for the 50hz operation.

I accomplished the MOD without cutting any P.C. foil or removing the main board from the Sol chassis. Parts needed are: one Molex Pin and a short length of flexible wire.

Remove IC 62 from its socket. Carefully scrape a small bare spot on the large ground bus running near the left of the IC socket; just remove the solder mask, don't cut the bus. Solder the Molex Pin to a one-inch length of wire and then the wire to the ground bus. Bend pin 5 of the 93L16 IC outward 45 degrees and insert the IC in the socket leaving pin 5 projecting outwards. Now slip the Molex Socket over the protruding IC pin. . . . That's it. And no permanent disfigurement of my precious Sol when I returned to the land of 60hz.

I hope that others will profit from this modification.

Ray D. Congdon
1956 COMM GP/OLC
APO San Francisco CA 96343

Ray:

Thanks much.

Aram

Dear Mr. Attarian

Enclosed you will find a check for $4.00 for my 1-year subscription to ACCESS. Also, I have a few questions and suggestions.

1. I am a firm supporter of your company. If there is any way I can contribute to its continued success, let me know. I am presently unemployed, so most of my time is spent exploring my Sol 20 system. Unfortunately, I paid for my system via a loan. In another month or so, I won't be able to pay off this loan. If I don't find some sort of income soon, I may have to sell my Sol. I don't want to have to do this!! My being located in the center of the eastern megalopolis should open some possibilities. If there is any way possible, HELP!!

So far, I have written some original programs. One demonstrates the use of control characters and the escape sequences used by BASIC5. I have also written machine subroutines which can be used with BASIC5 to produce some special effects.

I also do some hardware design. I have completed design of a 16K static memory board using MOS TEC 4104's. I am working on a few other things as well.

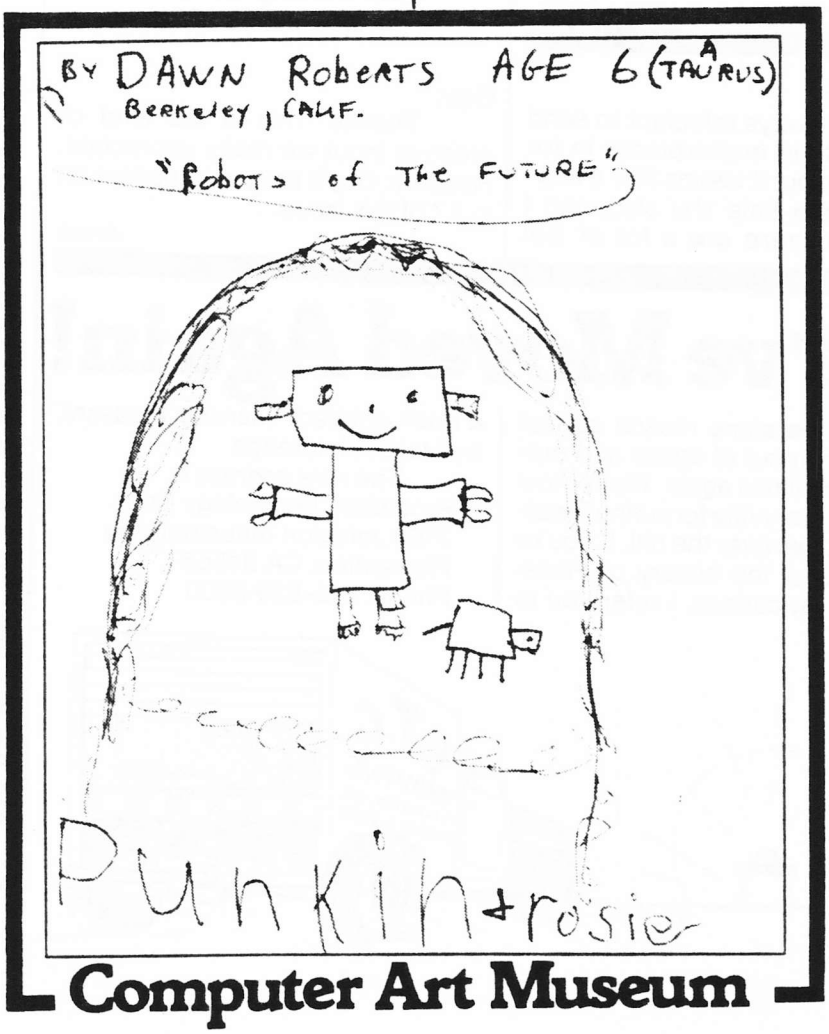
If there is any way I can help, even with nothing in return, please let me know. As far as some sort of income, if you can't help, maybe one of the readers can.

Note: I will relocate.

2. It is said that escape sequences can be used to generate characters in inverse video. I have not been able to figure out how to do this. Could you please shed some light on this subject.

3. In regard to a letter from Joseph P. Chalala, Willow Street, Pa., in Vol. 1, #3, I agree completely with his suggestion for a notebook type publication for ease of filing.

Well, that's all for now. I'm sure there will be more in the future. Remember Murphy's Law of Thermodynamics which states "Things get worse under pressure." So, take the time required to do, whatever,



Computer Art Museum

right (within reason, of course). If not, it will cost you more later.

A dedicated Sol user,
David F. Wrobel

P.S. The Bayshore Amateur Computer Group, of which I am president, consists of dedicated microcomputer hobbyists. Of which, only a few are Sol owners. We would like more. We are located in central N.J. Our address is: BACG P.O. Box 132, Holmdel, N.J. 07733.

David:

There's a good reason you haven't figured out how to generate characters in inverse video—it's **not** true that it can be done. Sorry. Hope the new format meets your filing needs.

And can anybody out there help David find work and save his Sol?

Aram

Dear Editor

I am always reluctant to send one of my little masterpieces in for publication, but it seems like everyone else is a little shy also, and I know that there are a lot of Sol owners, like me, anxious to try out some of its unique features.

For whatever it's worth, I'm sending a short program that makes use of the File commands in Basic 5. It's not meant to be a finished product, but it does demonstrate one way to use commands. In fact, I hope someone will pick up the ball and make a better program out of it.

The purpose of the program is to allow the user to slip last month's data tape into one file, pay the bills, update the household accounts and store the updated accounts on the other file.

I have included a sample program for setting up the original data tape, (similar to the one in the Basic 5 manual), the actual program I use for working the accounts, and a sample run.

Sincerely yours

Guy W. Campbell
5815 Buckley Drive
Jacksonville, FL 32210

Guy:

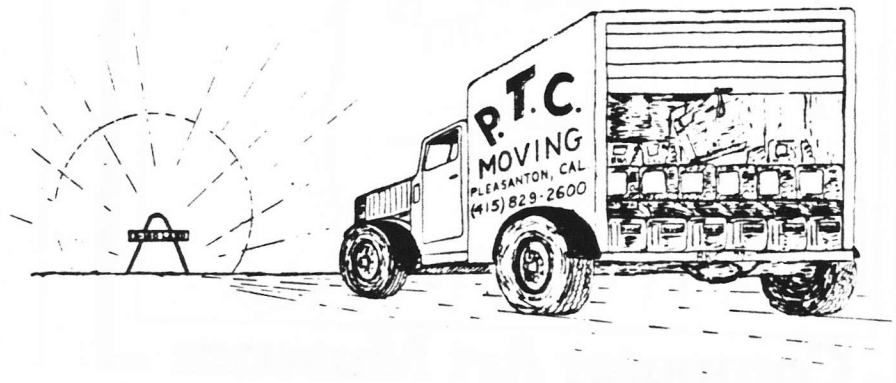
Thanks. This is the kind of creative input we really appreciate. Readers: Guy's program appears on p. 17 of this issue.

Aram

We've Moved Again!

For the same reason as last time—we ran out of space and outgrew the facilities again. We've now forsaken Emeryville for sunny Pleasanton, CA, just over the hill. If you're curious about the history of Pleasanton, or just curious, I refer you to a book entitled "Mammy Pleasant," by Helen Holdreage.

The new address is
Processor Technology Corp.
7100 Johnson Industrial Way
Pleasanton, CA 94566
Phone: 415-829-2600



Join the Sol Users' Society

The Sol Users' Society got under way Sunday, July 31, when about 30-40 Sol users met for the first organizational meeting. The Society is open to everyone who has a Sol or a Sol-type compatible system, so they're hoping to see even more of you at the next meetings.

This first time out a steering committee was elected, and goals were set for the Society. These goals are:

1. To facilitate communication between Sol owners.
2. To provide feedback from Sol owners to PTC.
3. To provide a mechanism for exchanging Sol software.
4. To encourage development of Sol-compatible products by other manufacturers.

Some time was spent just getting to know one another and talking about various projects the club can tackle. Seems that most of the stuff users have to offer is software, but they're also interested in reviewing any hardware submitted to the Society, be it prototype or production. They can't supply certification, though.

One project definitely under way is a Sol Users' Society newsletter. Contributions and comments herewith solicited.

A tidbit that emerged from the first meeting: TDK Auda C-60 cassette tape performs best in a bit-chopping test.

Schedule of meetings. The group is set to meet on Sundays Oct. 16, Nov. 20, and Dec. 18, at Varian Physics Lab, 2nd Floor, Stanford CA. Come meet the new steering committee: Bill Burns, Dave Fylstra, Ron Findlay, Ben Milander, Bill Holding, Stan Sokolow, David Fox.

For more information, please write to:

Bill Burns
4190 Maybell Way
Palo Alto, CA 94306
(no phone calls, please . . .)

Review:

1 EPT 2 - 57

Software Capabilities of the Helios II Disk System

The Helios II system has several capabilities I haven't seen in other disk systems on the market, and one particularly notable advantage for use with the Processor Technology Disk Operating System: you can write I/O routines for the Helios which permit the use of any I/O controller in conjunction with PTDOS, including the Cromemco D to A board and just about any homebrew board you've already built. (Maybe not some that perform DMA or make use of the I/O ports as control ports by the disk controller.)

The reason for this flexibility is that Helios treats all files as data files, including the device files used for I/O routines. These differ from regular files in that data read from or to them will come or go directly to the devices controlled by the I/O routine. With Helios, the only thing you have to worry about is to make sure you follow the guidelines in the PTDOS user's manual when you write your I/O routine.

Software support is another big plus for Helios. It offers a disk assembler, two editors (one ALS-8 type, one Nova-type), library functions, a debugger, language systems, procedures (PROCS), and full interface to PTDOS on command or assembly level.

The disk assembler allows you to generate object and listing files from a source file. You have the options of specifying if the input file is ALS-8 type, if it has line numbers, if it has form control, and a few other things.

The ALS-8 type editor is especially useful on systems with a lot of memory because you can work with text, as in the ALS-8. You have the options of scrolling forward and backward through the text, deleting characters, searching character strings, moving blocks of code, replacing string patterns with others as found. The limitation to this editor is that it requires the VDM-1.

The Nova-type editor can be run on almost any terminal, since all I/O is run through the system console routines. It will yank pages into the edit buffer, change data in the page, and write it out. If offers many of the same functions as the ALS-8 type.

Library functions allow you to assemble several source files which make up one logical program. It's done through the use of a copy verb included in the PTDOS assembler. You could expand the use of this verb to build up a library of source files which perform common functions, then concatenate these files into an object file through the assembler.

The debugger serves a function similar to that of the simulator in the ALS-8: you can run object code in a controlled environment. However, the debugger runs real time instead of the simulator's interpretive mode. You can use it to set numerous break points; examine memory in hex, character, or instruction format, alter memory or output drivers; and do several other useful debugging tasks.

The broad spectrum of language systems available includes a DISK BASIC and DISK FOCAL; FORTRAN is rumored to be among those upcoming. Language support is definitely one of the big advantages of the Helios system.

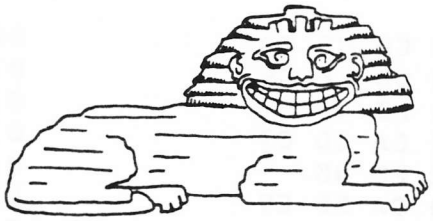
The command interpreter gives access from the console to many of PTDOS's numerous entry points. A partial list of the commands available: SPACE, OPEN, CLOSE, KILL, RANDOM, SEEK, RENAME, REATR, RETYPE, CREATE, READ, WRITE.

With PTDOS, you can also enter procedures in ALS-8 type file formats as a series of commands which may include optional statements.

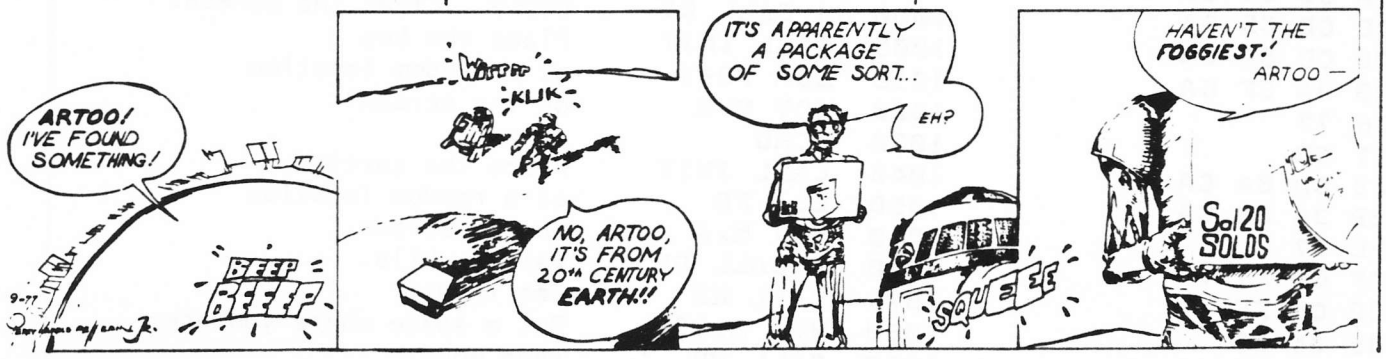
The PROCS itself is simply a list of commands which you can enter and allow to execute consecutively. Very useful for setting up, say, a 3-hour listing to print out while you get some sleep.

All in all, the Helios II disk memory system has proved well worth the time and price from this user's point of view.

Colgate Spinx



Cynic: One who is enough to make anyone a pessimist.



WINZI

WINZI is a collection of programs that were originally written last summer for VDM1 and published in D.D.J. They have been rewritten for a Sol with a SOLOS personality module and addressed to use the 1K of RAM available on the Sol P.C. board.

The programs are:

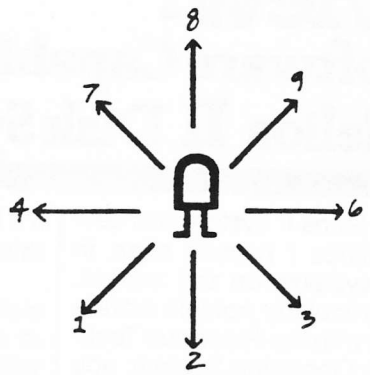
H ic—a random walk
O pps—draw a picture without returning
C hase—make the turtle catch the bouncing bug
L ife—the 'game' of LIFE

The speed of Hic, Opps, Chase and Life are under control of the SET command (see SOLOS manual). Before EXecuting C900 SET S=80. If you fail to set the speed first, the speed is so fast that you won't see it happen.

Type 'H' for Hic, 'O' for Opps, or 'C' for Chase

In Hic the beastie should be moving around leaving asterisks. If it is not moving, the random number generator might not be working. It is important **not** to zero memory before loading this program. In particular the Data Storage area SH should be nonzero. When you get tired of watching it, hit CR and return to the executive.

In Opps, you control the direction the beastie moves. The directions are as shown, upper right.



Type the number corresponding to the desired direction. The beastie will proceed in that direction until another direction is given. '0' will stop the beastie. '5' will cause a wipe out. '↯'. If the beastie ever returns to a location where it's been a '↯' will appear and the program will return to the executive, hence the name: Opps. If you wish to return to the executive at any other time type CR.

Life requires that there is an initial population of asterisks on the screen. Place them there by either Hic or Opps. Then enter Life by typing L from the executive. The CR will return you to the executive. 1K of RAM 0-3FF Hex is needed by Life as a scratchpad.

```
C900 CD C5 CA
C903 CD FE C9
C906 FE 43
C908 CA 1D C9
C90B FE 48
C90D CA 5D C9
C910 FE 4F
C912 CA 76 C9
C915 FE 4C
C917 CA 09 CA
C91A C3 03 C9
C91D CD C5 CA
C920 CD 84 CA
C923 3A CF CA
C926 77
C927 EB
C928 CD 84 CA
C92B 3A CE CA
C92E 77
C92F CD F1 C9
C932 CD FE C9
C935 36 20
C937 CD A3 C9
```

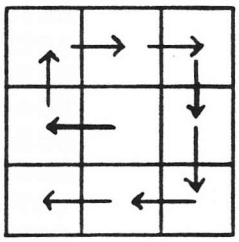
```
0090  CALL CS        Clear the screen
0100 RET CALL KB     The executive routine
0120  CPI 'C'        branches to chase on C,
0130  JZ CH
0140  CPI 'H'        Hic on H,
0150  JZ HI
0160  CPI 'O'        Opps on O, and
0170  JZ OP
0180  CPI 'L'        Life on L.
0190  JZ LI
0220  JMP RET
1000 CH CALL CS      Chase...Clear the screen.
1005  CALL INIT      Place the bug
1010  LDA F0+1       at a random location
1020  MOV M,A        on the screen.
1030  XCHG
1040  CALL INIT      Place the turtle
1050  LDA F0         at a random location
1060  MOV M,A        on the screen.
1070 C1 CALL DL      Wait a while.
1075  CALL KB        Get input.
1080  MVI M,20H      Put a space where you are,
1090  CALL MV        then move.
```

```
C93A 3A CF CA    1100  LDA F0+1       If the bug is there,
 93D BE          1110  CMP M          you've got him.
 93E CA 9E C9    1120  JZ HLT
 941 3A CE CA    1130  LDA F0         Place turtle at new screen
 944 77          1140  MOV M,A        location.
C945 EB          1150  XCHG
C946 CD A0 CA    1180  CALL RND       The bug moves at random
C949 36 20       1185  MVI M,20H      Put a space where it was,
C94B CD A3 C9    1190  CALL MV        then move.
C94E 3A CE CA    1200  LDA F0         If turtle is there,
C951 BE          1210  CMP M          the bugs been had.
C952 CA 9E C9    1220  JZ HLT
C955 3A CF CA    1230  LDA F0+1       Place bug at new screen
C958 77          1240  MOV M,A        location.
C959 EB          1250  XCHG
C95A C3 2F C9    1260  JMP C1         Do it all again
C95D CD C5 CA    1300 HI CALL CS      HIC...Clear the screen
C960 21 20 CE    1305  LXI H,0CE20H   Place the turtle at the
C963 36 07       1310 H1 MVI M,7      center of the screen.
C965 CD FE C9    1315  CALL KB        Should I return to the executive?
C968 CD F1 C9    1320  CALL DL        Wait awhile
C96B CD A0 CA    1330  CALL RND       Move the turtle at random
C96E 36 2A       1340  MVI M,2AH
C970 CD A3 C9    1350  CALL MV        leaving asterisk behind
C973 C3 63 C9    1370  JMP H1         Do it again.
C976 CD C5 CA    1600 OP CALL CS      Opps...Clear the screen
C979 06 00       1602  MVI B,0        Place turtle at the center
C97B 21 20 CE    1605  LXI H,0CE20H   of the screen.
C97E 7E          1610 01 MOV A,M      If you have been
 97F FE 2A       1620  CPI 2AH        here before,
C981 CA 9E C9    1630  JZ HLT         halt.
C984 36 07       1640  MVI M,7        Put the turtle on the screen
C986 CD F1 C9    1650  CALL DL        wait awhile.
C989 CD FE C9    1655 02 CALL KB      Get input.
C98C CA 90 C9    1658  JZ 04          If none, continue
C98F 47          1660  MOV B,A
C990 78          1662 04 MOV A,B
C991 E6 0F       1664  ANI 0FH        If it is a '0', don't move.
C993 CA 89 C9    1666  JZ 02
C996 36 2A       1670  MVI M,2AH      Place asterisk in old location
C998 CD A3 C9    1680  CALL MV        then move.
C99B C3 7E C9    1690  JMP 01         Do it again
C99E 36 04       1700 HLT MVI M,4     Halt--Place '↯' on the screen.
C9A0 C3 03 C9    1710  JMP RET       Return to the executive
C9A3 E6 0F       1800 MV ANI 0FH      Move...
C9A5 FE 06       1850  CPI 6
C9A7 C2 AF C9    1860  JNZ M1         This routine uses the
C9AA 3E 01       1870  MVI A,1        curser move routines
C9AC C3 C0 C9    1880  JMP M3         in SOLOS (PUP, PLEFT, PDOWN,
C9AF FE 01       1890 M1 CPI 1        and PRIT) to move the
C9B1 C2 B9 C9    1900  JNZ M2         contents of the screen
C9B4 3E 06       1910  MVI A,6        location pointed to
C9B6 C3 C0 C9    1920  JMP M3         by the address contained
C9B9 FE 07       1930 M2 CPI 7        in the H & L registers.
```

```
C9BB C2 C0 C9          1940  JNZ M3
C9BE 3E 0C             1950  MVI A,0CH      The right 4 bits of the
C9C0 4F                2000 M3 MOV C,A      accumulator are used
C9C1 CD E0 C9          2010  CALL SVDA       to determine the direction,
C9C4 3E 08             2020  MVI A,8         or combination of
C9C6 A1                2030  ANA C           directions, of the move.
C9C7 C4 04 C1          2040  CNZ 0C104H
C9CA 3E 01             2050  MVI A,1
C9CC A1                2055  ANA C
C9CD C4 15 C1          2060  CNZ 0C115H
C9D0 3E 02             2070  MVI A,2         The ASCII code for
C9D2 A1                2080  ANA C           all of the digits except
C9D3 C4 CB C0          2090  CNZ 0C0CBH      6, 1, and 7 allows a
C9D6 3E 04             2100  MVI A,4         compass rose that
C9D8 A1                2110  ANA C           agrees with the 10 key
C9D9 C4 0B C1          2120  CNZ 0C10BH      numeric pad.  6, 1 and
C9DC CD 1C C1          2130  CALL 0C11CH     7 are changed to agree.
C9DF C9                2140  RET
C9E0 7D                2200 SVDA MOV A,L    This routine assumes that
C9E1 E6 3F             2210  ANI 3FH         the contents of the H,L register
C9E3 32 08 C8          2220  STA 0C808H      pair is a screen address.
C9E6 29                2230  DAD H           It then stores the line
C9E7 29                2240  DAD H           number in the data storage
C9E8 7C                2250  MOV A,H         location used by SOLOS
C9E9 E6 0F             2260  ANI 0FH         called LINE.  Simiarly the
C9EB E6 0F             2270  ANI 0FH         character position is stored
C9ED 32 09 C8          2280  STA 0C809H      in NCHAR for use by
C9F0 C9                2290  RET             PUP, PLEFT, PDOWN, and PRIT
C9F1 E5                3000 DL PUSH H       Delay...Save address.
C9F2 2A 0A C8          3005  LHLD 0C80AH     Get data byte from SET
C9F5 2C                3010  INR L           command: SPEED
C9F6 AF                3020  XRA A
C9F7 2B                3030 D1 DCX H        Use it for a counter.
C9F8 BC                3040  CMP H
C9F9 C2 F7 C9          3050  JNZ D1
C9FC E1                3055  POP H           Restore address
C9FD C9                3060  RET
C9FE CD 2E C0          3100 KB CALL 0C02EH  Get input from keyboard.
CA01 C8                3110  RZ              If none return.
CA02 FE 0D             3120  CPI 0DH         If it is a CR restore the
CA04 C0                3130  RNZ             stack and return to the
CA05 C1                3140  POP B           executive
CA06 C3 03 C9          3150  JMP RET
CA09 11 00 00          4000 LI LXI D,0      Life
CA0C 6B                4010 L1 MOV L,E
CA0D 7A                4020  MOV A,D         D, E points to 1K of RAM
CA0E E6 03             4030  ANI 3           used as a scratch pad.
CA10 F6 CC             4040  ORI 0CCH        H, L points to the screen
CA12 67                4050  MOV H,A         location.
CA13 D5                4060  PUSH D          Save the pointer
CA14 0E 00             4070  MVI C,0         initialize the counter
CA16 2B                4080  DCX H
CA17 CD 7E CA          4090  CALL CT
CA1A 11 C0 FF          4100  LXI D,0FFC0H    Count the neighbors
CA1D 19                4110  DAD D           of this location of
```

up	lf	dn	rt

```
CA1E CD 7E CA          4120  CALL CT        the screen in the
CA21 23                4130  INX H          following order:
 122 CD 7E CA          4140  CALL CT
 125 23                4150  INX H
  26 CD 7E CA          4160  CALL CT
 A29 11 40 00          4170  LXI D,40H
CA2C 19                4180  DAD D
CA2D CD 7E CA          4190  CALL CT
CA30 19                4200  DAD D
CA31 CD 7E CA          4210  CALL CT
CA34 2B                4220  DCX H
CA35 CD 7E CA          4230  CALL CT
CA38 2B                4240  DCX H
CA39 CD 7E CA          4250  CALL CT
CA3C 23                4251  INX H
CA3D 11 C0 FF          4252  LXI D,0FFC0H
CA40 19                4253  DAD D
CA41 D1                4260  POP D          Restore RAM pointer.
CA42 CD 65 CA          4270  CALL RG        Determine next generation
CA45 13                4280  INX D          for this location and put
CA46 7A                4290  MOV A,D        it in the RAM.
CA47 E6 04             4300  ANI 4          Do this to every screen
CA49 CA 0C CA          4310  JZ L1          location.
CA4C 21 00 CC          5000 CP LXI H,0CC00H  Copy the next generation
CA4F 11 00 00          5010  LXI D,0          as stored in the IK of
CA52 1A                5020 C0 LDAX D         scratch pad RAM
CA53 77                5030  MOV M,A          to the screen.
CA54 23                5040  INX H
 A55 13                5050  INX D
 A56 7C                5060  MOV A,H
 A57 FE D0             5070  CPI 0D0H
CA59 C2 52 CA          5080  JNZ C0
CA5C CD F1 C9          5090  CALL DL          Wait awhile.
CA5F CD FE C9          5100  CALL KB          Should I return to the exec.?
CA62 C3 09 CA          5150  JMP LI           If not do another generation.
CA65 79                7000 RG MOV A,C       Follow the rules of LIFE
CA66 FE 02             7010  CPI 2            to determine the next
CA68 CA 7B CA          7020  JZ R3            generation for this
CA6B D2 72 CA          7030  JNC R1           location.
CA6E 3E 20             7040 R2 MVI A,20H
CA70 12                7045  STAX D
CA71 C9                7050  RET
CA72 FE 03             7060 R1 CPI 3
CA74 C2 6E CA          7070  JNZ R2
CA77 3E 2A             7080  MVI A,'*'
CA79 12                7090  STAX D
CA7A C9                7100  RET
CA7B 7E                7110 R3 MOV A,M
CA7C 12                7120  STAX D
CA7D C9                7130  RET
CA7E 7E                8000 CT MOV A,M       Counter
CA7F FE 2A             8010  CPI '*'
CA81 C0                8020  RNZ             If the neighbor is
CA82 0C                8030  INR C
 A83 C9                8040  RET             an asterisk count it.
```



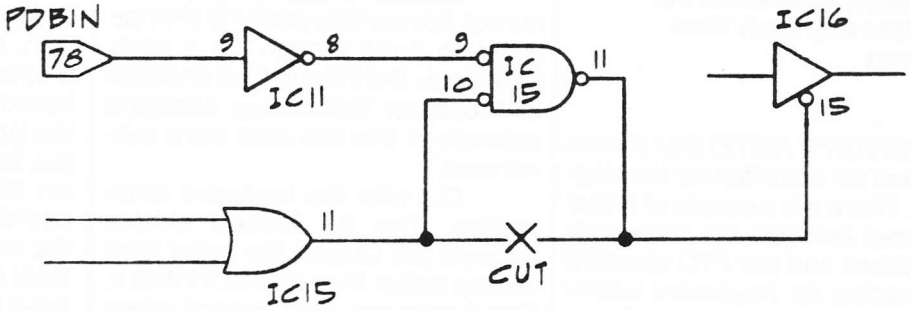
```
CA84 CD A0 CA          8500 INIT CALL RND    Initialize.
CA87 E6 3F             8510  ANI 3FH
CA89 F6 CC             8515  ORI 0CCH        determine a random
CA8B 67                8520  MOV H,A         screen location,
CA8C CD A0 CA          8530  CALL RND        place the line
CA8F 6F                8540  MOV L,A         number in the data
CA90 C9                8550  RET             storage of SOLOS
CA91 32 08 C8          8560  STA 0C808H      called LINE and
CA94 CD A0 CA          8570  CALL RND        the position of the
CA97 E6 0F             8580  ANI 0FH         character in NCHAR.
CA99 32 09 C8          8590  STA 0C809H
CA9C CD E0 C9          8594  CALL SVDA
CA9F C9                8595  RET
CAA0 E5                8600 RND PUSH H       Random number generator
CAA1 21 D3 CA          8610  LXI H,SH+3      from Peoples Computer
CAA4 06 08             8620  MVI B,8         Company,
CAA6 7E                8630  MOV A,M         For it to work propertly
CAA7 07                8640 RTOP RLC         the 4 data storage locations
CAA8 07                8641  RLC             should not be zero.
CAA9 07                8642  RLC
CAAA AE                8650  XRA M
CAAB 17                8660  RAL
CAAC 17                8661  RAL
CAAD 2D                8670  DCR L
CAAE 2D                8671  DCR L
CAAF 2D                8672  DCR L
CAB0 7E                8680  MOV A,M
CAB1 17                8690  RAL
CAB2 77                8695  MOV M,A
CAB3 2C                8700  INR L
CAB4 7E                8710  MOV A,M
CAB5 17                8720  RAL
CAB6 77                8730  MOV M,A
CAB7 2C                8740  INR L
CAB8 7E                8750  MOV A,M
CAB9 17                8760  RAL
CABA 77                8770  MOV M,A
CABB 2C                8780  INR L
CABC 7E                8790  MOV A,M
CABD 17                8800  RAL
CABE 77                8810  MOV M,A
CABF 05                8820  DCR B
CAC0 C2 A7 CA          8830  JNZ RTOP
CAC3 E1                8840  POP H
CAC4 C9                8850  RET
CAC5 CD D5 C0          8900 CS CALL 0C0D5H In SOLOS, the routine PERSE
CAC8 3E 20             8910  MVI A,20H       erases the screen but leaves
CACA 32 00 CC          8920  STA 0CC00H      a cursor in the corner.
CACD C9                8930  RET             Erase that too and return.
CACE 07                9000 F0 DB 7          The turtle font
CACF 0E                9001  DB 0EH          The bug font.
CAD0                   9010 SH DS 4          Storage for RND.
```

Bytesaver Modification for Sol

If you want to use a Chromemco Bytesaver in the Sol, you'll need to make the following modification of the Bytesaver. Data will then be gated onto the Bus only when PDBIN is high or active, necessary in the Sol because the Data IN and Data OUT busses are connected together.

First cut the trace connecting pin 11 of IC 15 (7432) to pin 15 of IC 16 (74367). Now make these connections with small gauge insulated wire:

1. Connect pins 11 and 10 of IC 15.
2. Connect pin 8 of IC 15 to pin 15 of IC 16.
3. Connect pin 8 of IC 11 to pin 9 of IC 15.
4. Connect S-100 Bus pin 78 (PDBIN) to IC 11 pin 9. Pin 78 is the 23rd from the left on the **solder** side of the board.



. . . AND A BYTESAVER ROGRAMMING ROUTINE

This short routine will program e contents of any 1K block of memory into a 2708 EPROM installed in socket 1 of Bytesaver. The Bytesaver should be addressed at 6000H.

A15-L, A14-H, A13-H

The routine is used as a custom command with the Solos/Cuter operating system. Enter the program at C900H, or reassemble it elsewhere if you wish. Then create a custom command by typing:

CU BURN C900 (CR)

NOTE: CR means "strike the return key;" do **not** type the letters as part of the command.

If the program has been reassembled at an arbitrary address of NNNN, type:

CU BURN NNNN (CR)

Now to use the BURN custom command, type:

BURN AAAA (CR)

AAAA being the starting address of the 1K block you wish to program into the 2708. The programming operation takes about 5 minutes, which is in accordance with the published programming instructions for the 2708. When the programming is complete, the routine will return control to Solos/Cutter and a prompt will reappear on the screen.

```
                    C33A

C900 CD 3A C3
C903 7D
C904 B7
C905 C2 04 C0

C908 22 28 C9
C90B 01 00 00

C90E 11 00 64
C911 2A 28 C9

C914 7E
C915 12
C916 23
C917 13
C918 7A
C919 FE 68
C91B C2 14 C9
```

```
0000 **
0000 *** BYTESAVER ROUTINE ***
0000 **
0000 SCONV  EQU   0C33AH  GETS PARAMETERS
0000 *
0000 BURN   CALL  SCONV   SOURCE ADRS· TO HL
0000        MOV   A,L     GET LO ADRS BYTE
0000        ORA   A
0000        JNZ   0C004H  IT MUST BE 0
0000 *
0000        SHLD  SAD     KEEP SOURCE ADRS·
0000        LXI   B,0     PASS COUNT=0
0000 *
0000 BLOOP  LXI   D,6400H  PROM ADRS
0000        LHLD  SAD     SOURCE ADRS·
0000 *
0000 PLOOP  MOV   A,M     GET SOURCE DATA
0000        STAX  D       ZAP THE PROM
0000        INX   H       BUMP SOURCE
0000        INX   D       & PROM ADRS·
0000        MOV   A,D     CHECK HI ADRS·
0000        CPI   68H     PASS COMPLETE ?
0000        JNZ   PLOOP   NOT YET
```

BYTESAVER ROUTINE, cont.

1 EPT 2 - 64

```
                        0000 *
C91E Ø3                 0000        INX     B       BUMP PASS COUNT
C91F 78                 0000        MOV     A,B
C92Ø FE Ø4              0000        CPI     4       IK PASSES ?
C922 C2 ØE C9           0000        JNZ     BLOOP   NOT YET
                        0000 *
C925 C3 Ø4 CØ           0000        JMP     ØCØØ4H ALL DONE
                        0000 *
                        0000 **  RAM AREA **
                        0000 *
C928                    0000 SAD    DS      2       SOURCE ADRS·
                        0000 *
```

A Sol Keyboard Fix:

So You Won't Have to Hit the Upper Case Key Each Time You Restart

EDITOR'S NOTE: Our thanks to Jay Bell for contributing this suggestion. There are a couple of minor differences between the procedure he describes and our PTC standard modification for keyboard upper case initialization, so we're printing our version along with his.

So after days of constructing your Sol, you're finally ready to input the first command, hit the carriage return, and check the screen. WHAAT?!! All you get is some question mark nonsense. So you check the software manual again. Sure enough, it wants upper case. So, you put the keyboard into alpha-shift by pressing the upper case key.

Later you notice your program isn't doing what you expected, so naturally you restart the old four-phase wonder by simultaneously pressing the upper case and repeat keys. More question marks—the restart left you in lower case mode.

By now you've realized this is going to happen every time. There are three solutions: 1) Change the software to accept both upper and lower case commands. 2) Change the keyboard to come up in upper-case mode. 3) Hit the upper case key every time you restart.

Number 3 had already worn me out. I personally prefer to change software, even though I'm a hardware freak. But I figured the chances of Processor Technology changing software at this late date were subminimal.

Out with the keyboard schematics. The fix looked simple enough: just CLEAR the upper case flipflop rather than PRESETTING it. The keyboard gets preset when power is first applied through an RC circuit that is initially low and slowly comes up to +5 volts. Since the signal coming off the keyboard to restart the 8080 is driven by an open collector inverter, it could also be connected to the power-up RC circuit. Then whenever you reset the processor, you also reset the keyboard to its initial power-up state. To make that power-up state turn the upper case flipflop on, you cut the land leading to pin 4 of U15, and the land leading to pin 1 of U15. Then connect the trace that used to lead to pin 4 to pin 1 instead. Similarly, connect the pull-up resistor that was tied to pin 1 to pin 4 instead. Then connect pin 8 of U24 to pin 1 of U15. This last connection ties the restart signal to the clear input of U15 (as well as to the rest of the chips that are initialized at power-up).

There is only one remaining problem for the purists. The flipflop that sets the machine in the local mode will come up in an undetermined state, since its preset pin is tied to pin 4 of U15. Now that you've cut the land to pin 4 and pulled it high, the local flipflop is not being properly reset. Unfortunately, you have to remove U15 in order to cut the land to the local flipflop's preset pin, because the land runs under it on the component side of the board. If you want to be sure that the machine will come up with the local mode off, cut the land between pin 4 of U15 and pin 10 of U15, then connect pin 1 of U15 to pin 10 of U15.

Now you should be able to interact with your Sol the instant you power up or restart, without the bother of hitting the upper case key first.

THE PTC MODIFICATION:

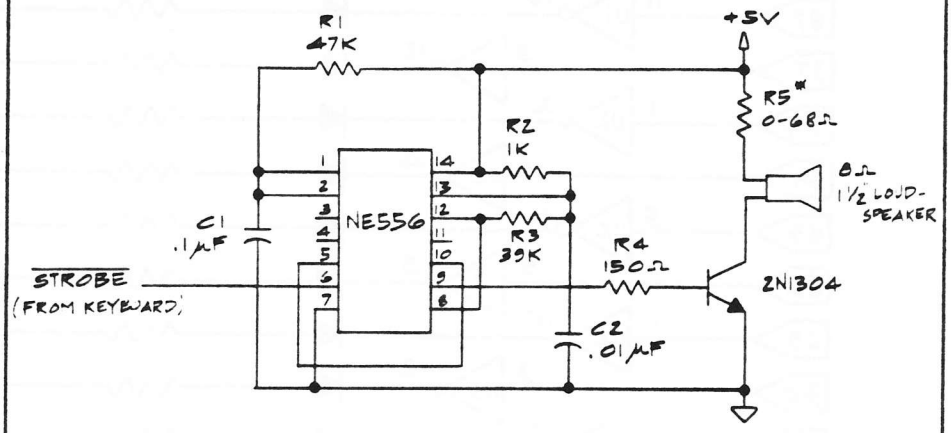
1. Cut trace located between U15 pin 4 and plate through ⅛ inches below pin on the component side.
2. Remove R31, 1.5K ¼ watt Carbon Film, and save for later use.
3. On the Solder Side of the board:
   a. Insert one end of R31 in plate through adjacent to U15 pin 14 and solder.
   b. Bend the other lead of R31 to pin 4 of U15 and solder.
   c. Add a ⅝ inch jumper, stripped ⅛ inch from each end, to the plate through located just below U15 pin 4.
   d. Insert the other end of the jumper through plate through located just above U24 pin 12.

A Keyclick (Audible) Circuit for Sol

Silence may be golden, but there is an advantage to making your Sol keyboard sound like a typewriter. If you're a good fast touch typist entering data from a printed source, it's easier to listen for missed keys than to glance up at the screen all the time. Thanks to Jack Kinney for this audible circuit design; he says that the sound can be altered to suit individual tastes by varying RI burst length and R3 for burst frequency. The circuit operates as follows:

The first section of the dual timer is connected in the monostable mode, and the keyboard strobe triggers a positive-going pulse approximately four milliseconds long. This pulse is connected to the reset of the second section of the timer, which is operating in the astable mode, and is set for an output frequency of approximately 1.5 Khz, ating it "on" for a four-millisecond st. The output transistor inverts ne signal to prevent current draw in the "off" condition. The collector resistor is set for the desired loudness.

Kinney is also checking out a more complex circuit (three more IC's) which will decode the "BEL" code and produce a beep. One of the computers on the network signals for attention by transmitting the "BEL," and this will provide an audible monitor. We'll print this circuit in the next issue of ACCESS.



KEY "CLICK" CIRCUIT (Sol 10/20)

JACK KINNEY
UCLA COMPUTER SCIENCE DEPT.
3413 BOELTER HALL
LOS ANGELES, CA 90024

Katchum's Correction Corollaries: (a) In debugging any type of program, no corrections can be made correctly after 1600 hours Friday. (b) The corrections will be self-evident at 900 hours Monday, (c) When in doubt divide by (2.0).

oren's Law of Graphing: First draw curves, then plot the data.

Run/Stop Circuits: Part II

In ACCESS #3, I described a Run/Stop circuit for user control of Sol's X-Ready line. Now here's the circuit that will let you monitor the operation of your S-100 system by connecting LED's to the buss lines. Each Light Emitting Diode is driven by 1/6 of a hex inverter package (74LS04), current limited by a 470-ohm ¼-watt resistor for each.

To monitor the operation of your system, compare the addresses and data displayed on the LED's as you single step, with the program listing. Most malfunctions can then be seen and corrected with very little effort. A couple of examples:

Quick test for data and address lines. You can discover major failures in these by toggling the reset line while the Run/Stop circuit is enabled in the Stop position. All the LED's should light when the reset line is enabled. All except M1, PDBIN, and PWAIT should darken when the line is disabled. Any LED's that don't respond as indicated reveal a malfunction in the corresponding lines and should be checked with a meter or scope.

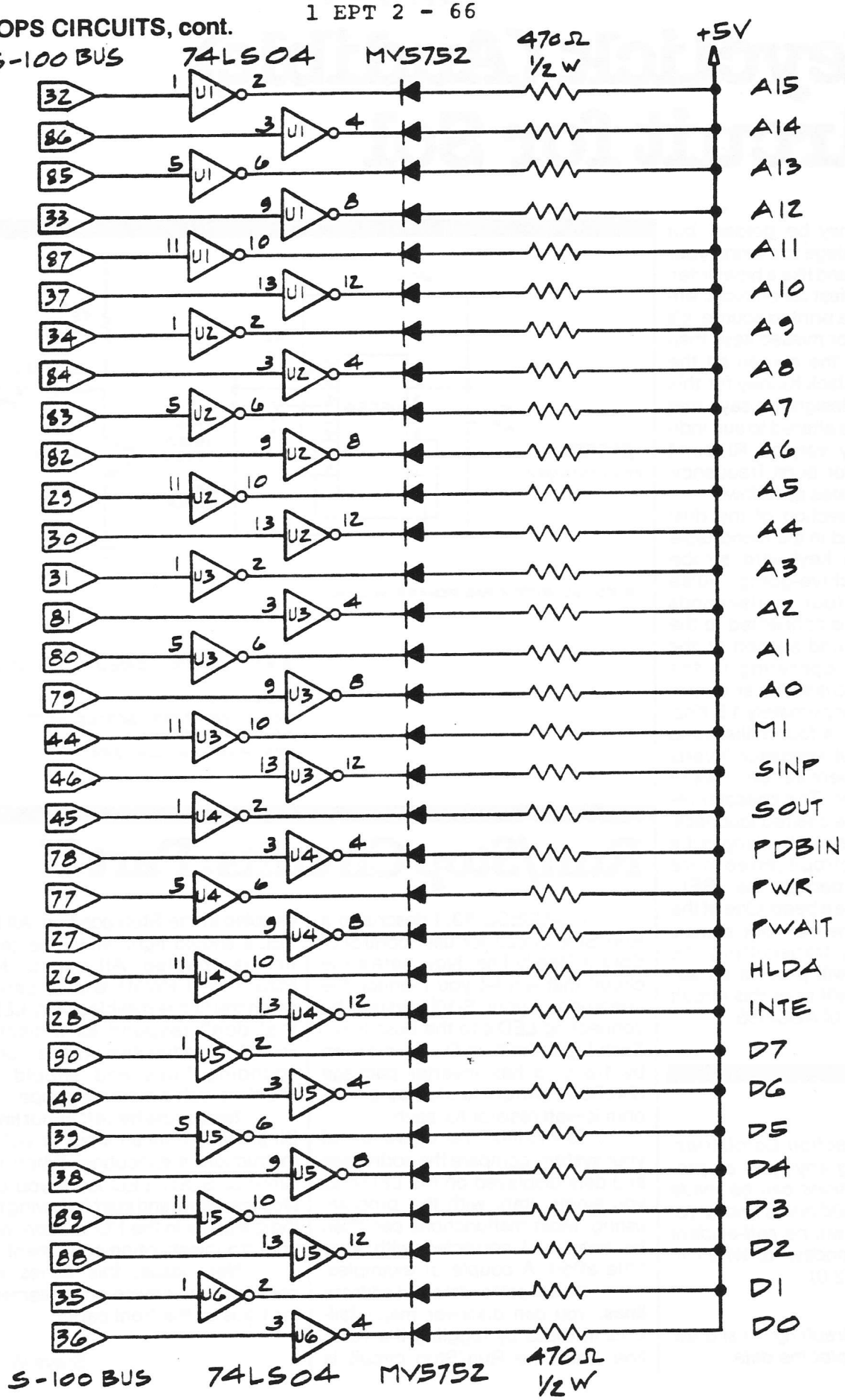
Testing the Input/Output lines. Single step until an input or output instruction is executing. When the SINP or SOUT LED is lit, you can stop stepping, and start following the logic signals in the I/O section, with your troubleshooting equipment.

Next issue, this series will continue with advice on implementing traps on the front panel.

Steve Wong

RUN/STOPS CIRCUITS, cont.

ACCESS • November, 197 jpo 16



Bug Squad

Changes in Assembly Instructions for Sol:

The bug: You'll get an incorrect test result at Step 38 in the instructions; instead of the display shown in Figure 3-9, a display of random characters comes up.
The squasher: *Install U93-74LS175 and U107-74LS367 at Step 35.*

The bug: Your Sol doesn't work at Step 59.
The squasher: *After you do Step 28 of the assembly, perform step 73 before proceeding to Step 29.*
The bug: The waveforms are incorrectly shown in Figure 3-2 on page 3-15 of your Sol manual.
The squasher: *Turn your manual upside down—the waveform shown for Pin 5-U104 is inverted. Same for Pin 7-U104.*

A Program for a Home Accounting System

Contributed by Guy Campbell
See the Letters to the Editor for Guy's comments on his program.

We're delighted to get this kind of input from our readers and pass it on for all Sol users.

```
PROGRAM FOR HOME ACCOUNTING SYSTEM .

THIS PROGRAM WILL RECORD PAYMENTS, UPDATE
BALANCE AND PROVIDE ACCUMULATED INTEREST
FOR TAX PURPOSES.

NEW ACCOUNTS CAN BE ADDED ONLY BY CHANGING
THE PROGRAM.

PUT OLD DATA TAPE ON FILE #2 AND SET FOR PLAY.

PUT NEW TAPE ON FILE #1 AND SET FOR RECORD.

PRESS ANY NUMBER & RETURN TO CONTINUE.  0
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

YOUR FRIENDLY FINANCE CO.
3958 TUFFLUK STREET
CHICAGO, ILL  60683

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

ACCOUNT NUMBER - 12345A

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

BALANCE=$ 51.60                    ANNUAL INTEREST RATE= 21%
TOTAL PRINC. PAID=$ 108.00         TOTAL INT. PAID=$ 6.39

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

YOUR LAST PAYMENT WAS MADE ON 82877 FOR $ 36.00

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
ENTER CHANGES TO ADJUST BALANCE. $ 25.00
ENTER CHANGES TO ADJUST BALANCE. $ 0
YOUR PRESENT BALANCE IS $ 76.60

ENTER PAYMENT TO THIS ACCT. $36.00

ENTER TODAYS DATE 90277
PRINCIPLE PAID=$34.66          INTEREST PAID=$ 1.34

YOUR NEW BALANCE IS $41.94

TOTAL PRINCIPLE PAID TO DATE=$ 144.00
TOTAL INTEREST PAID TO DATE=$ 7.73

ACCOUNT COMPLETE
PRESS ANY NUMBER TO GET NEXT ACCOUNT. 0
**********************************************************************
NEXT ACCOUNT COMES UP - WILL CONTINUE IN THIS FORMAT.
LIST

5     SET S=05
10    REM THE HOME ACCOUNTING PROGRAM
20    REM CREATED BY G. W. CAMPBELL - 1977
30    REM ORIGINAL DATA BANK (TAPE) PREPARED WITH
40    REM SEPARATE PROGRAM.
50    PRINT "PROGRAM FOR HOME ACCOUNTING SYSTEM."
60    PRINT
70    PRINT "THIS PROGRAM WILL RECORD PAYMENTS, UPDATE"
80    PRINT "BALANCE AND PROVIDE ACCUMULATED INTEREST"
90    PRINT "FOR TAX PURPOSES."
100   PRINT
110   PRINT "NEW ACCOUNTS CAN BE ADDED ONLY BY CHANGING"
120   PRINT "THE PROGRAM."
130   PRINT
140   FOR I=1 TO 1200:NEXT
150   PRINT "PUT OLD DATA TAPE ON FILE #2 AND SET FOR PLAY."
160   PRINT
180   PRINT "PUT NEW TAPE ON FILE #1 AND SET FOR RECORD."
190   PRINT
200   INPUT "PRESS ANY NUMBER & RETURN TO CONTINUE. "Z
210   PRINT
220   GOSUB 770
225   FILE #1
230   FILE #2
240   READ #2,A,B,C,D,E,F,G: PRINT "END OF FILE";: GOTO 700
250   IF A=1 THEN GOSUB 1000
260   IF A=2 THEN GOSUB 1070
270   IF A=3 THEN GOSUB 1140

*****CONTINUE THIS SECTION FOR THE NUMBER OF ACCOUNTS NEEDED*****

400   GOSUB 770
410   PRINT "BALANCE=$";%Z2%;C,
420   PRINT TAB (30);"ANNUAL INTEREST RATE=";%%;B;"%";%Z2%
```

```
430    PRINT "TOTAL PRINC. PAID=$";F,
440    PRINT TAB(30);"TOTAL INT. PAID=$";G
450    GOSUB 770
451    PRINT "YOUR LAST PAYMENT WAS MADE ON ";%%;D,
452    PRINT "FOR $";%Z2%;E
453    GOSUB 770
460    INPUT "ENTER CHANGES TO ADJUST BALANCE. $"C1
470    IF C1=0 THEN 500
480    LET C=C+C1
490    GOTO 460
500    PRINT "YOUR PRESENT BALANCE IS $";C
510    PRINT
520    INPUT "ENTER PAYMENT TO THIS ACCT. $"F1
521    LET T1=T1+F1
530    IF F1=0 THEN 583
531    PRINT
532    INPUT "ENTER TODAYS DATE "D1
533    LET D=D1
550    LET G1=( (B/100)*C)/12
560    LET G=G+G1
570    LET C=C-(F1-G1)
580    LET E=F1
581    PRINT "PRINCIPLE PAID=$";(E-G1),
582    PRINT TAB(30);"INTEREST PAID=$";G1
583    PRINT
590    PRINT "YOUR NEW BALANCE IS $";C
595    LET F=F+(E-G1)
600    PRINT
610    PRINT "TOTAL PRINCIPLE PAID TO DATE=$";F
620    PRINT "TOTAL INTEREST PAID TO DATE=$";G
630    PRINT
640    PRINT "ACCOUNT COMPLETE"
650    INPUT "PRESS ANY NUMBER TO GET NEXT ACCOUNT. "Y
670    PRINT #1,A,B,C,D,E,F,G
680    PRINT
681    PRINT "**************************************************
**********"

690    TOTO 240
700    CLOSE #2
710    CLOSE #1
720    PRINT
730    PRINT "TRANSACTIONS COMPLETE"
740    PRINT
741    PRINT "YOUR PAYMENTS TOTALED $";T1;" THIS MONTH."
750    PRINT
751    PRINT "GOODBYE, SEE YOU NEXT MONTH."
752    SET S=0
760    END
770    PRINT "-------------------------------------------------
----------"
780    RETURN

*****THIS IS WHERE YOU PUT THE ACCOUNTS*****
```

```
1000  PRINT "YOUR FRIENDLY FINANCE CO."
1010  PRINT "3958 TUFFLUK STREET"
1020  PRINT "CHICAGO, ILL  60683"
1030  GOSUB 770
1040  PRINT "ACCOUNT NUMBER - 12345A"
1050  GOSUB 770
1060  RETURN
1070*****CONTINUE TO PUT IN ACCOUNTS IN THE SAME FORMAT.
THIS IS A SAMPLE PROGRAM THAT CAN BE USED TO ESTABLISH THE
ORIGINAL DATA BASE TAPE.

LIST

10    FILE #2
20    INPUT "ACCOUNT IDENTIFICATION NO. ?"A
30    IF A=0 THEN 110
35    PRINT
40    INPUT "ANNUAL INTEREST RATE (WHOLE NUMBERS) ?"B
45    PRINT
50    INPUT "BALANCE ?"C
55    PRINT
60    INPUT "DATE AND PAYMENT (LAST PMT MADE) ?"D,E
65    PRINT
70    INPUT "TOTAL PRINCIPLE PAID ?"F
75    PRINT
80    INPUT "TOTAL INTEREST PAID TO DATE ?"G
85    PRINT
90    PRINT #2,A,B,C,D,E,F,G
100   GOTO 20
110   CLOSE #2
120   END
```

8080 Relocatable Assembler

```
+0000              ; (11,19,29)
+0000              ; MODIFIED SOLOS ROUTINE
+0000              ;     REVISED BY:
+0000              ;                     MELVIN SCHEHLEIN
+0000              ;                      COMMUNICATIONS & MEDIA SERVICES
+0000              ;                       TOWSON  STATE UNIVERSITY
+0000              ;                        TOWSON, MARYLAND 21204
+0000              ;
+0000              ;  DATE WRITTEN: AUGUST 3, 1977
+0000              ;
+0000              ; THE PURPOSE OF THIS PROGRAM IS TO CONFIGURE THE SOL TERMINAL
+0000              ;  COMPUTER AS A STANDARD VIDEO TERMINAL TO ACCEPT THE HALF-DUPLEX
+0000              ;  CR, LF RESPONSE FROM COMMUNICATIONS AFTER HAVING SENT A CR
+0000              ;  WITHOUT ERASING THAE LAST INPUT LINE.  THIS HAPPENS BECAUSE
+0000              ;  THE SOLOS MONITOR CLEARS THE LINE FROM ITS PRESENT CHARACTER
+0000              ;  POSITION TO THE END OF THAT LINE.   DURING COMMUNICATIONS
+0000              ;  THE KEYBOARD SENDS OUT A CR, WHICH IS FEED BACK INTO THE SERIAL
+0000              ;  INPUT PORT, AND THEN SENT TO THE VDM DRIVER TO BE PROCESSED.
+0000              ;  THEN THE COMPUTER SENDS OUT A CR LF WHICH GOES TO THE VDM DRIVER
+0000              ;  CLEARING THE LAST INPUT LINE (BECAUSE THE RESPONSE CR WAS IN COLUMN
+0000              ;  1 ) THUS DENYING THE USER THE ABILITY TO CHECK THE ACCURACY OF
+0000              ;  HIS OR HER LAST INPUT.
+0000              ;
```

```
C900                            ORG     0C900H
C900                    ;
C900                    ;  THIS PROGRAM MODIFICATION WILL SIMPLY TEST THE CHARACTER POSITION
C900                    ;   WHENEVER A CR IS RECEIVED,  IF WE ARE IN COLUMN 1 THE CR IS
C900                    ;   IGNORED, IF WE ARE NOT IN COLUMN 1 IT IS PROCESSED NORMALLY.
C900                    ;   THE ONLY CODE THAT IS DIFFERENT FROM THE STANDARD SOLOS TERM
C900                    ;   ROUTINE CAN BE FOUND IN LINES 63 TO 70 TO TEST THE CHARACTER
C900                    ;   POSITION.
C900                    ;   THE PROPGRAM HAS BEEN ASSEMBLED BEGINING AT ADDRESS C900
C900                    ;   SO IT CAN EASILY BE STORED IN SOL SYSTEM RAM.
C900                    ;
C900                    ;       +++++  TERMINAL UPDATE ROUTINE  +++++
C900                    ;
C900 CD 10 C3   TERMU:  CALL    PSCAN   ;FIND IF INPUT PARAMETER IS PRESET
C903 32 06 C8           STA     IPORT   ;SINP WILL USE THIS DRIVER (DEFAULT IS 1)
C906 CD 10 C3           CALL    PSCAN   ;NOW FOR THE OUTPUT DRIVER
C909 32 07 C8           STA     OPORT
C90C                                    ;
C90C CD 2E C0   TERM1U: CALL    KSTAT   ;IS THERE A CHAR WAITING AT THE KEYBOARD
C90F CA 24 C9           JZ      TINU    ;NO, THEN WHAT ABOUT THE SERIAL INPUT PORT
C912 47                 MOV     B,A     ;THE KEYBOARD HAD ONE, PUT IT IN B
C913                                    ;
C913 FE 80              CPI     MODE    ;IS IT A COMMAND MODE
C915 CA C0 C1           JZ      COMN1   ;YES--GET OUT OF TERMU>>AND GO TO SYSTEM MONT
C918 DA 21 C9           JC      TOUTU   ;NON-CURSOR KEY--SEND TO TERM PORT
C91B CD 54 C0           CALL    VDMOT   ;TO THE VDM IT IS A CURSOR CONTROL
C91E C3 24 C9           JMP     TINU    ;IS THERE A CHAR AT THE INPUT PORT
C921                                    ;
C921 CD 19 C0   TOUTU:  CALL    SOUT    ;OUTPUT IT TO THE SERIAL PORT
C924 CD 1F C0   TINU:   CALL    SINP    ;IS A CHAR WAITING AT THE INPUT PORT
C927 CA 0C C9           JZ      TERM1U  ;HOW ABOUT THE KEYBOARD?
C92A E6 7F              ANI     07FH    ;GET RID OF THAT HIGH ORDER PARITY BIT.
C92C CA 0C C9           JZ      TERM1U  ;A NULL IS PROCESSED BY DOING NOTHING
C92F 47                 MOV     B,A     ;IT IS OUTPUT FROM 'B'
C930 FE 1B              CPI     01BH    ;IS IT A CONTROL CHAR TO BE IGNORED
C932 D2 5E C9           JNC     TERM2U  ;NO IT IS >1B SEND TO THE VDM
C935                                    ;
C935                                    ;
C935 FE 0D              CPI     CR      ;CR OR LF ARE SPECIAL CHARACTERS
C937 C2 46 C9           JNZ     NOCR    ;IT WASN'T A CR, BYPASS THE COLUMN TEST
C93A                                    ;
C93A 3A 08 C8           LDA     NCHAR   ;HAS A CR ALREADY PUT US IN COLUMN 1
C93D FE 00              CPI     000H    ;THIS WILL TELL US
C93F CA 0C C9           JZ      TERM1U  ;IT IS COLUMN 1, DON'T PROCESS IT
C942 78                 MOV     A,B     ;LETS GET IT BACK
C943 C3 5E C9           JMP     TERM2U  ;IT ISN'T COLUMN 1, PROCESS IT
C946                                    ;
C946 FE 0A      NOCR:   CPI     LF      ;WAS IT A LINEFEED
C948 CA 5E C9           JZ      TERM2U  ;PROCESS IT
C94B                                    ;
C94B 3A 0C C8           LDA     ESCFL   ;A CTL CHAR---ARE WE IN AN ESC SEQUENCE
C94E B7                 ORA     A       ;IF YES THEN OUTPUT CONTROL CHAR DIRECTLY TO
C94F C2 5E C9           JNZ     TERM2U  ;WE SURE ARE LET THE VDM DRIVER HANDLE IT
C952 C5                 PUSH    B       ;SAVE THE CHARACTER
C953 06 1B              MVI     B,ESC   ;CTL CHAR TO VDM VIA ESC SEQUENCE
C955 CD 54 C0           CALL    VDMOT   ;PUT IT ON THE SCREEN
C958 06 07              MVI     B,7     ;SAY TO PUT OUT NEXT CHAR AS IS
C95A CD 54 C0           CALL    VDMOT   ;ALMOST READY
C95D C1                 POP     B       ;GET IT BACK
C95E            TERM2U: EQU     $       ;LETS PUT OUT THE CHARACTER
C95E CD 54 C0           CALL    VDMOT   ;PUT IT ON THE SCREEN
C961 C3 0C C9           JMP     TERM1U  ;LETS KEEP DOING IT
C964                    ;
C964                    ;
C964                    ;
C964                    ;
```

```
C964                ;   S Y S T E M   E Q U A T E S
C964                ;   - - - - - -   - - - - - - -
C964                ;
000A                LF          EQU     00AH        ;THE LINE FEED ASCII CHAR
000D                CR          EQU     00DH        ;THE CARRIAGE RETURN ASCII CHAR
001B                ESC         EQU     01BH        ;THE ESCAPE ASCII CHAR
0080                MODE        EQU     080H        ;THE P.T. MODE CONTROL CHAR
C964                                                ;
C019                SOUT:       EQU     0C019H      ;SYSTEM OUTPUT ENTRY POINT
C01F                SINP:       EQU     0C01FH      ;SYSTEM INPUT ENTRY POINT
C02E                KSTAT:      EQU     0C02EH      ;KEYBOARD INPUT ROUTINE
C054                VDMOT:      EQU     0C054H      ;THIS ONE PUTS IT ON THE SCREEN
C1C0                COMN1:      EQU     0C1C0H      ;RE-ENTRY INTO SYSTEM MONITOR
C310                PSCAN:      EQU     0C310H      ;INPUT PARAMETER ROUTINE
C806                IPORT:      EQU     0C806H      ;CURRENT INPUT PSEUDO PORT
C807                OPORT:      EQU     0C807H      ;CURRENT OUTPUT PSEUDO PORT
C808                NCHAR:      EQU     0C808H      ;PRESENT VDM CHARACTER POSITION
C80C                ESCFL:      EQU     0C80CH      ;ESCAPE FLAG CONTROL BYTE
C964                                                ;
C964                            END
0 ERRORS DETECTED
```

PA

BE 01 SYMBOL TABLE

COMN1	C1C0	CR	000D	ESC	001B	ESCFL	C80C
IPORT	C806	KSTAT	C02E	LF	000A	MODE	0080
NCHAR	C808	NOCR	C946	OPORT	C807	PSCAN	C310
SINP	C01F	SOUT	C019	TERM1U	C90C	TERM2U	C95E
TERMU	C900	TINU	C924	TOUTU	C921	VDMOT	C054

ALS-8 to Sol Patch

```
                    0000 *
                    0001 * This program alters the ALS-8 ram to
                    0002 * allow an ALS-8 that has not been updated
                    0003 * to interface with a Sol-20 using Solos.
                    0004 *
                    0005 *
                    0006 * Change standard input port to Sol
                    0007 * keyboard.
                    0008 *
D09F                0009            ORG     0D09FH
D09F FC             0010            DB      KDATA
                    0011 *
                    0012 * Alter STAT routine in ALS-8 to match Sol
                    0013 * keyboard status port and compliment data
                    0014 * availible flag.
                    0015 *
D0A4                0016            ORG     0D0A4H
D0A4 DB FA          0017 STAT       IN      KSTAT
D0A6 2F             0018            CMA
D0A7 E6 01          0019            ANI     KDR
D0A9 C9             0020            RET
                    0021 *
                    0022 * This output driver saves the accumulator
                    0023 * and register B before calling SOUT in
                    0024 * Solos. It also checks the keyboard to see
                    0025 * if an escape has been hit. If so, a jump
```

ALS-8 TO SOL PATCH, cont.


```
                    0026 * will be made to EORMS in ALS-8. IF not,
                    0027 * registers A and B will be popped off the
                    0028 * stack after the character has been printed.
                    0029 *
                    0030 OUTP8  PUSH   PSW
D0AA F5             0031        PUSH   B
D0AB C5             0032        CALL   STAT
D0AC CD A4 D0       0033        JZ     NOCHR
D0AF CA BB D0       0034        IN     KDATA
D0B2 DB FC          0035        ANI    7FH
D0B4 E6 7F          0036        CPI    ESC
D0B6 FE 1B          0037        JZ     EORMS
D0B8 CA 60 E0       0038 NOCHR  CALL   SOUT
D0BB CD 19 C0       0039        POP    B
D0BE C1             0040        POP    PSW
D0BF F1             0041        RET
D0C0 C9             0042 *
                    0043 * Now make SYSIO and OUT8 point to this
                    0044 * version of OUTP8 as the standard output
                    0045 * driver.
                    0046 *
                    0047        ORG    0D096H  SYSIO output driver
D096                0048        DW     OUTP8
D096 AA D0          0049 *
                    0050        ORG    0D0D0H  second byte of OUT8
D0D0                0051        DW     OUTP8
D0D0 AA D0          0052 *
                    0053 ESC    EQU    1BH     ASCII escape
     001B           0054 KDR    EQU    01H     data available flag
     0001           0055 SOUT   EQU    0C019H  Solos output entry
     C019           0056 KDATA  EQU    0FCH    Sol keyboard data port
     00FC           0057 KSTAT  EQU    0FAH    keyboard status port
     00FA           0058 EORMS  EQU    0E060H  entry point to ALS-8
     E060           0059 *
                    0060 * One more thing needs to be mentioned.
                    0061 * A non-updated version of the TXT-2
                    0062 * editor does not reset the hardware
                    0063 * scrolling port on a Sol. If this is
                    0064 * not done before entering the editor
                    0065 * the first line of the file may be
                    0066 * on a line other than the first line
                    0067 * of the screen. One remedy for this
                    0068 * is to always hit the CLEAR key before
                    0069 * executing the TXT-2 editor.
                    0070 *
```

Intel Paper Tape Loader for Sol

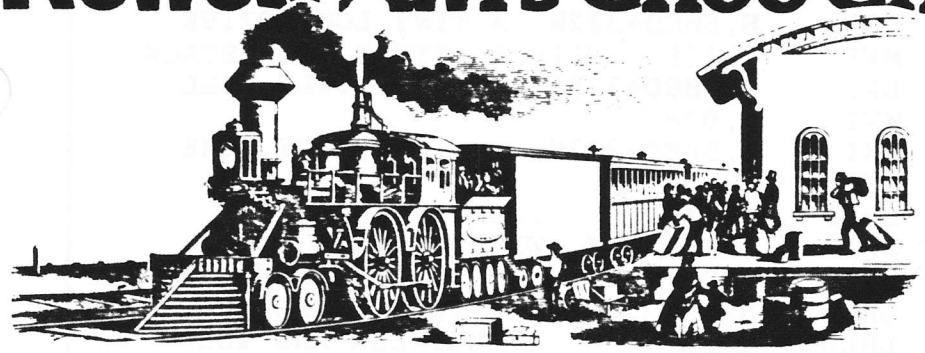
```
                    0000        LST
     1EFF           0000 STAK   EQU    $-1
                    0000 *
1F00 31 FF 1E       0000        LXI    SP,STAK
1F03 CD 06 1F       0000        CALL   READ
```

INTEL PAPER TAPE LOADER FOR SOL, cont.


```
1F06 CD 45 1F          0000 READ   CALL   TTYIN
1F09 FE 3B             0000        CPI    ';'
1F0B C2 06 1F          0000        JNZ    READ
1F0E CD 2A 1F          0000        CALL   CHAR
1F11 57                0000        MOV    D,A
1F12 C8                0000        RZ
1F13 CD 2A 1F          0000        CALL   CHAR
1F16 67                0000        MOV    H,A
1F17 CD 2A 1F          0000        CALL   CHAR
1F1A 6F                0000        MOV    L,A
1F1B CD 2A 1F          0000        CALL   CHAR
                       0000 *
1F1E CD 2A 1F          0000 LOOP   CALL   CHAR
1F21 77                0000        MOV    M,A
1F22 23                0000        INX    H
1F23 15                0000        DCR    D
1F24 C2 1E 1F          0000        JNZ    LOOP
1F27 C3 06 1F          0000        JMP    READ
                       0000 *
1F2A CD 45 1F          0000 CHAR   CALL   TTYIN
1F2D CD 3D 1F          0000        CALL   HEX
1F30 07                0000        RLC
1F31 17                0000        RAL
1F32 17                0000        RAL
1F33 17                0000        RAL
1F34 5F                0000        MOV    E,A
1F35 CD 45 1F          0000        CALL   TTYIN
1F38 CD 3D 1F          0000        CALL   HEX
1F3B 83                0000        ADD    E
1F3C C9                0000        RET
                       0000 *
1F3D D6 30             0000 HEX    SUI    48
1F3F FE 0A             0000        CPI    10
1F41 D8                0000        RC
1F42 D6 07             0000        SUI    7
1F44 C9                0000        RET
                       0000 *
1F45 DB F8             0000 TTYIN  IN     0F8H
1F47 E6 40             0000        ANI    64
1F49 CA 45 1F          0000        JZ     TTYIN
1F4C DB F9             0000        IN     0F9H
1F4E E6 7F             0000        ANI    7FH
1F50 C9                0000        RET

CHAR    1F2A    HEX     1F3D    LOOP    1F1E    READ    1F06
```

Newett Awl's Choo Choo



A lot of you had fun with this program when we listed it in the first issue of ACCESS. We're repeating it now with modifications to run on Sol, and we think you'll like it even better this time around.

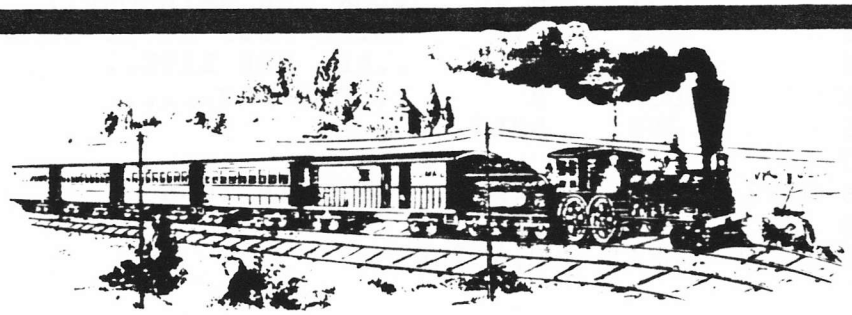
```
                         0000 *
0100 AF                  0001 TRAIN  XRA     A       ONCE UPON A TIME,
0101 D3 FE               0002        OUT     0FEH    IN A CURIOUS LITTLE
0103 21 86 02            0003        LXI     H,SHED   PLACE THERE WAS -
0106 01 00 08            0004        LXI     B,2048   ..A TINY TRAIN
0109 36 20               0005 EMPTY  MVI     M,20H   ..AND IT STAYED IN A
010B 23                  0006        INX     H       ..TINY SHED
010C 0B                  0007        DCX     B       ..THAT WAS ALL EMPTY
010D AF                  0008        XRA     A
010E A8                  0009        XRA     B
010F C2 09 01            0010        JNZ     EMPTY
0112 21 FA 03            0011        LXI     H,CLOUD  ..EXCEPT FOR A HUGE
0115 11 01 00            0012        LXI     D,1
0118 06 0E               0013        MVI     B,14    B
011A CD 50 01            0014        CALL    SMO1     I
011D 11 2B 00            0015        LXI     D,43      L
0120 06 0C               0016        MVI     B,12       L
0122 CD 50 01            0017        CALL    SMO1        O
0125 11 30 00            0018        LXI     D,48         W
0128 06 09               0019        MVI     B,9           Y
012A CD 50 01            0020        CALL    SMO1
012D 11 34 00            0021        LXI     D,52            C
0130 06 04               0022        MVI     B,4              L
0132 CD 50 01            0023        CALL    SMO1              O
0135 11 3A 00            0024        LXI     D,58                U
0138 06 02               0025        MVI     B,2                  D
013A CD 50 01            0026        CALL    SMO1
013D 11 3D 00            0027        LXI     D,61                    O
0140 06 01               0028        MVI     B,1                      F
0142 CD 50 01            0029        CALL    SMO1
0145 11 3E 00            0030        LXI     D,62                        S
0148 06 01               0031        MVI     B,1                          M
014A CD 50 01            0032        CALL    SMO1                          O
014D C3 59 01            0033        JMP     ENGIN                          K
                         0034 *                                              E
0150 19                  0035 SMO1   DAD     D
0151 36 6F               0036 SMO2   MVI     M,SMOKE  ..COUGH
0153 23                  0037        INX     H
0154 05                  0038        DCR     B               ..COUGH
0155 C2 51 01            0039        JNZ     SMO2
0158 C9                  0040        RET     .       COMING OUT OF THE STACK
```

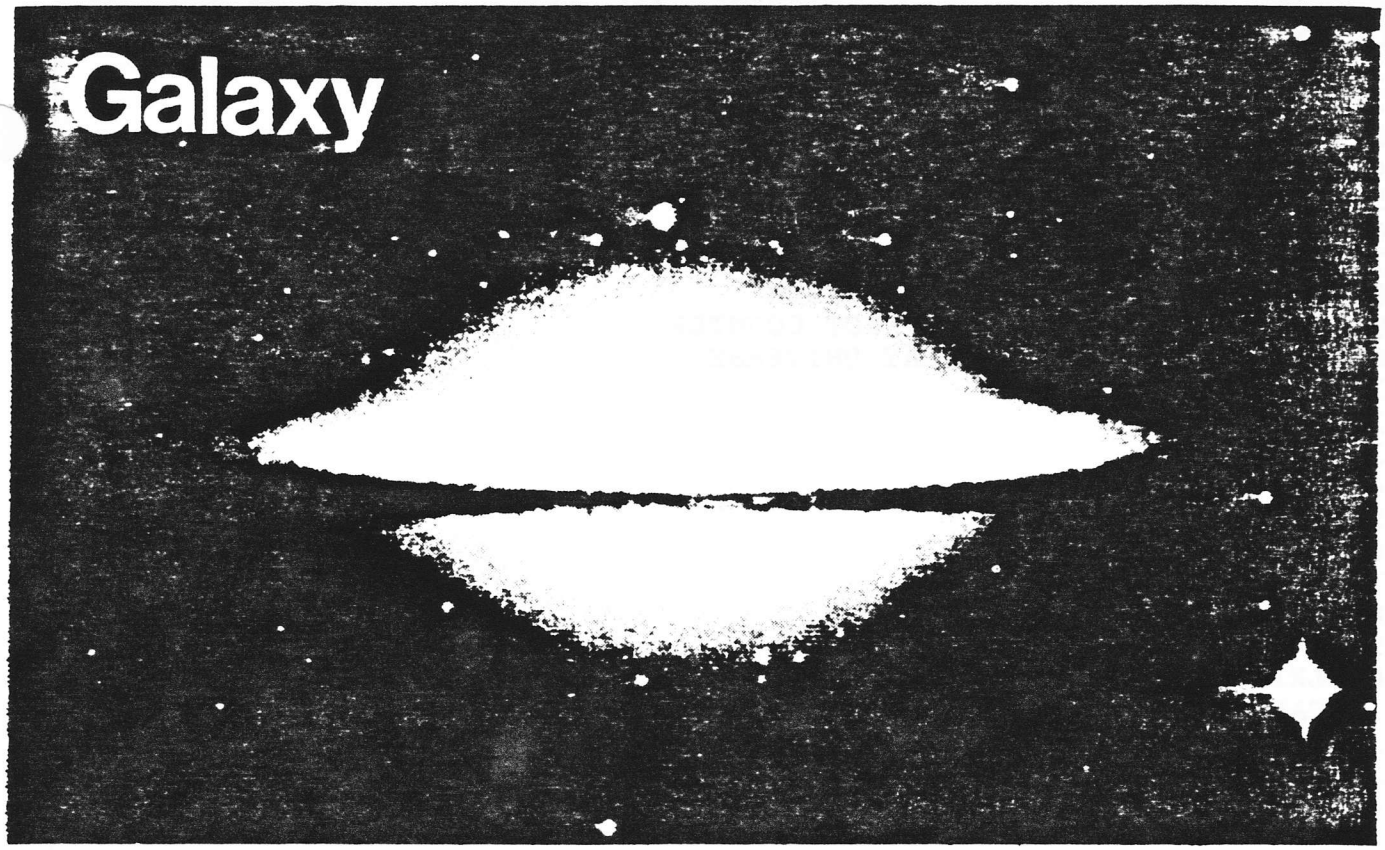
```
                        0041 *                          OF
0159 21 A8 05           0042 ENGIN  LXI     H,SHED+322H  A TINY LOCOMOTIVE
015C 36 16              0043        MVI     M,16H   WITH A LITTLE SMOKESTACK
015E 21 AB 05           0044        LXI     H,SHED+325H  AND A LITTLE BELL
0161 36 07              0045        MVI     M,07H
0163 21 AD 05           0046        LXI     H,SHED+327H  ..AND A TINY DOME
0166 36 6E              0047        MVI     M,6EH
0168 2A 68 02           0048        LHLD    CAB1         ..AND A CAB
016B 22 B0 05           0049        SHLD    SHED+32AH    WITH WINDOWS
016E 2A 6A 02           0050        LHLD    CAB2         ..SO THAT YOU COULD
0171 22 B2 05           0051        SHLD    SHED+32CH    SEE INTO WHERE THE
0174 2A 6C 02           0052        LHLD    CAB3         ..ENGINEER AND THE
0177 22 B4 05           0053        SHLD    SHED+32EH    FIREMAN SAT.
017A 2A 6E 02           0054        LHLD    BOI1         ..IT HAD A BEAUTIFUL
017D 22 E7 05           0055        SHLD    SHED+361H    POLISHED
0180 2A 70 02           0056        LHLD    BOI2         ..BRASS
0183 22 E9 05           0057        SHLD    SHED+363H    BOILER
0186 2A 72 02           0058        LHLD    BOI3         ..WITH
0189 22 EB 05           0059        SHLD    SHED+365H    THE NUMBER
018C 22 ED 05           0060        SHLD    SHED+367H    "99" ON THE
018F 2A 74 02           0061        LHLD    BOI4         ..SIDE OF
0192 22 EF 05           0062        SHLD    SHED+369H    THE CAB
0195 2A 76 02           0063        LHLD    BOI5         ..BUT YOU COULDN'T
0198 22 F1 05           0064        SHLD    SHED+36BH    SEE EITHER THE
019B 2A 78 02           0065        LHLD    BOI6         ..ENGINEER OR THE
019E 22 F3 05           0066        SHLD    SHED+36DH    FIREMAN
01A1 2A 7A 02           0067        LHLD    FRA1         ..THE LITTLE ENGINE
01A4 22 26 06           0068        SHLD    SHED+3A0H    ALSO
01A7 2A 7C 02           0069        LHLD    FRA2         ..HAD A COWCATCHER
01AA 22 28 06           0070        SHLD    SHED+3A2H    AND LOTS OF FUNNY
01AD 22 2A 06           0071        SHLD    SHED+3A4H    WHEELS AND THE
01B0 22 2C 06           0072        SHLD    SHED+3A6H    THINGS THAT CONNECTED
01B3 2A 7E 02           0073        LHLD    FRA3         ..AND TWO VERY TINY
01B6 22 2E 06           0074        SHLD    SHED+3A8H    WHEELS AT THE VERY
01B9 2A 80 02           0075        LHLD    FRA4         ..VERY BACK
01BC 22 30 06           0076        SHLD    SHED+3AAH    ALTOGETHER IT LOOKED
01BF 2A 82 02           0077        LHLD    FRA5         ..QUITE LONELY AND YET
01C2 22 32 06           0078        SHLD    SHED+3ACH    IT APPEARED VERY
01C5 01 DF FF           0079        LXI     B,-21H       ..FUNNY JUST SITTING
01C8 21 64 06           0080        LXI     H,SHED+3DEH  THERE ON THE
01CB 36 19              0081 RAILS  MVI     M,19H     ..RAILS
01CD 03                 0082        INX     B         ..WITH NOTHING AT ALL
01CE 23                 0083        INX     H         ..EVER
01CF AF                 0084        XRA     A         ..TO DO
01D0 A8                 0085        XRA     B         ..  W E L L !
01D1 C2 CB 01           0086        JNZ     RAILS     ..LET'S RUN IT JUST FOR
                        0087 *                              FUN
01D4 21 00 CC           0088 HOSTL  LXI     H,RRY   GET TRAIN OUT OF SHED
01D7 EB                 0089        XCHG    .       MOVE IT FROM SHED
01D8 21 86 02           0090        LXI     H,SHED
01DB 7E                 0091 FIRE   MOV     A,M     LIGHT FIRE
01DC 23                 0092        INX     H
01DD EB                 0093        XCHG
01DE 77                 0094        MOV     M,A     ..MOVE IT
01DF 23                 0095        INX     H
```

```
01E0 7C          0096          MOV     A,H
01E1 EB          0097          XCHG
01E2 FE D0       0098          CPI     0D0H    END OF YARD?
01E4 C2 DB 01    0099          JNZ     FIRE    NO, MAKE MORE STEAM!
01E7 CD ED 01    0100          CALL    TRAVL   GO TAKE TRIP
01EA C3 D4 01    0101          JMP     HOSTL   NO ROUND TRIPS, JUST DO
                 0102 *                        AGAIN
                 0103 *
01ED 01 40 03    0104 TRAVL    LXI     B,64*13  MAKE TRIP 13 MILES
01F0 C5          0105 CHOO     PUSH    B        CHUFF ONCE
01F1 CD FF 01    0106          CALL    STROK    MAKE FORWARD MOTION
01F4 CD 0F 02    0107          CALL    TURN     TURN WHEELS
01F7 C1          0108          POP     B
01F8 0B          0109          DCX     B        CLICK ODOMETER
01F9 AF          0110          XRA     A
01FA A8          0111          XRA     B        13 MILES YET?
01FB C2 F0 01    0112          JNZ     CHOO     NO
01FE C9          0113          RET     .        DO NEXT TRIP
                 0114 *
01FF 21 01 CC    0115 STROK    LXI     H,0CC01H  MOV DOWN THE TRACK
0202 0E D0       0116          MVI     C,0D0H
0204 7E          0117 COAL     MOV     A,M      ADD MORE COAL
0205 2B          0118          DCX     H
0206 77          0119          MOV     M,A
0207 23          0120          INX     H
0208 23          0121          INX     H
0209 7C          0122          MOV     A,H
020A B9          0123          CMP     C        ENOUGH COAL?
020B C2 04 02    0124          JNZ     COAL     NO, PUT MORE ON!
020E C9          0125          RET
                 0126 *
020F 21 00 CC    0127 TURN     LXI     H,0CC00H  TURN WHEELS
0212 7E          0128 HISS     MOV     A,M
0213 23          0129          INX     H
0214 FE 06       0130          CPI     6        FIND COWCATCHER
0216 C2 12 02    0131          JNZ     HISS     LOOK AGAIN
0219 23          0132 FOUND    INX     H
021A 7E          0133 AXLE     MOV     A,M
021B FE 11       0134          CPI     11H      WHAT QUARTER TURN
021D C2 31 02    0135          JNZ     NXT1
0220 11 7E 14    0136          LXI     D,147EH  NEXT QUARTER TURN
                 0137 *
0223 06 07       0138 AXL1     MVI     B,7      DO 4 AXLES
0225 72          0139 AXL2     MOV     M,D      WHEELS MOVED HERE
0226 05          0140          DCR     B        LAST WHEEL?
0227 CA 52 02    0141          JZ      WORK     I'VE BEEN WORKIN'...
022A 23          0142          INX     H       ..ON THE RAILROAD..
022B 73          0143          MOV     M,E     (SIDE RODS MOVED HERE)
022C 23          0144          INX     H       ..ALL THE LIVE..
022D 05          0145          DCR     B       ..LONG DAY.....
022E C2 25 02    0146          JNZ     AXL2
                 0147 *
0231 FE 14       0148 NXT1     CPI     14H      QUARTER TURN
0233 C2 3C 02    0149          JNZ     NXT2
0236 11 2D 13    0150          LXI     D,132DH
```

CHOO CHOO, cont.

```
0239 C3 23 02          0151          JMP     AXL1
023C FE 13             0152 NXT2     CPI     13H       QUARTER TURN
023E C2 47 02          0153          JNZ     NXT3
0241 11 5F 12          0154          LXI     D,125FH
0244 C3 23 02          0155          JMP     AXL1
0247 FE 12             0156 NXT3     CPI     12H       QUARTER TURN
0249 C2 1A 02          0157          JNZ     AXLE
024C 11 2D 11          0158          LXI     D,112DH
024F C3 23 02          0159          JMP     AXL1
                       0160 *
0252 01 B4 14          0161 WORK     LXI     B,5300
0255 0B                0162 OIL      DCX     B
0256 DB FA             0163          IN      0FAH      DID THE CONDUCTOR
0258 E6 01             0164          ANI     RDA       ..FLAG US DOWN
025A DB FC             0165          IN      0FCH      CLEAR INPUT PORT
025C CA 65 02          0166          JZ      QUIT      QUITTIN' TIME
025F AF                0167          XRA     A         NO,    THEN HIGHBALL
0260 A8                0168          XRA     B
0261 C2 55 02          0169          JNZ     OIL       NO SQUEEKS, PLEASE!
0264 C9                0170          RET
                       0171 *
                       0172 *    PUT A JUMP TO WHERE EVER YOU NEED TO
                       0173 *    TO RETURN IN THE NEXT STATION.
0265 C3 04 C0          0174 QUIT     JMP     0C004H YOUR RETURN TICKET HERE
                       0175 *
     0001              0176 RDA      EQU     01H     PUT YOUR DAV FLAG HERE
     CC00              0177 RRY      EQU     0CC00H    VDM RAM
     006F              0178 SMOKE    EQU     6FH       SMOKE CHARACTOR
0268 01 10             0179 CAB1     DW      1001H     CAB DESCRIPTION
026A 10 5D             0180 CAB2     DW      5D10H      "       "
026C 7E 20             0181 CAB3     DW      207EH      "       "
026E 28 0A             0182 BOI1     DW      0A28H     BOILER DESCRIPTION
0270 0A 19             0183 BOI2     DW      190AH       "          "
0272 0A 0A             0184 BOI3     DW      0A0AH       "          "
0274 0A 5B             0185 BOI4     DW      5B0AH       "          "
0276 39 39             0186 BOI5     DW      3939H       "          "
0278 5D 20             0187 BOI6     DW      205DH       "          "
027A 06 20             0188 FRA1     DW      2006H     FRAME DESCRIPTION
027C 11 2D             0189 FRA2     DW      2D11H       "          "
027E 11 20             0190 FRA3     DW      2011H       "          "
0280 6F 2D             0191 FRA4     DW      2D6FH       "          "
0282 6F 20             0192 FRA5     DW      206FH       "          "
0284 19 19             0193 TIES     DW      1919H     TIES DESCRIPTION
0286 00                0194 SHED     NOP     .         THIS IS SHED AREA
     03FA              0195 CLOUD    EQU     $+0173H   BEGINNING OF CLOUD
```






```
0001 *
0002 *
0003 *   SHOOTING STARS  --  A BRAIN TEASER GAME
0004 *   AN INTERACTIVE GAME FOR 8080 COMPUTER
0005 *   THIS VERSION IMPLEMENTED FOR Sol
0006 *   DEVELOPED AND DEBUGGED WITH ALS-8
0007 *   PROGRAM DEVELOPMENT SYSTEM
0008 *
0009 *   PROGRAM ORIGINALLY SUBMITTED TO THE
0010 *   HEWLETT-PACKARD SOFTWARE LIBRARY IN BASIC
0011 *   SUBSEQUENTLY PUBLISHED BY PEOPLE'S COMPUTER
0012 *   COMPANY IN SEPT 74 AS "TEASER"
0013 *   WRITTEN IN 8008 CODE BY WILLARD I. NICO
0014 *   IN BYTE MAY 76 AS "SHOOTING STARS"
0015 *   ADAPTED TO 8080 CODE FOR USE WITH VDM-1
0016 *   BY JERRY BACON
0017 *   S.I.L. PROGRAMMING LABORATORY
0018 *   BRASILIA, D.F. BRASIL
0019 *
0020 *   DATE:  14 JAN 77
0021 *
0022 *
0023  ORG 0
0024 SHSDR LXI H,HEADR  POINT TO HEADER MESSAGE
0025  CALL SCRN     DISPLAY IT
0026  CALL KBD
0027  CPI 'N'       CHECK IF RULES WANTED
0028  JZ START  IF NOT, START GAME
0029  LXI H,PAGE1  OTHERWISE POINT TO 1ST PAGE
```

```
0030  CALL SCRN
0031  CALL KBD
0032  LXI H,PAGE2    POINT TO 2ND PAGE
0033  CALL SCRN
0034  CALL KBD
0035 START LXI D,VDM1
0036  CALL CLER    CLEAR THE SCREEN
0037  LXI B,0001H    INITIALIZE UNIVERSE
0038  MOV D,B     CLEAR SHOT COUNTER
0039 CNTST INR D  INCREMENT SHOT COUNTER
0040 DISP LXI H,STR1  DISPLAY UNIVERSE
0041  CALL TYPE
0042  LXI H,STR2
0043  CALL TYPE
0044  LXI H,STR3
0045  CALL TYPE
0046  LXI H,STR4
0047  CALL TYPE
0048  LXI H,STR6
0049  CALL TYPE
0050  LXI H,STR7
0051  CALL TYPE
0052  LXI H,STR8
0053  CALL TYPE
0054  LXI H,STR9
0055  CALL TYPE
0056  XRA A
0057  MOV A,C
0058  RRC
0059  LXI H,STR5
0060  CALL TYPE1
0061 WNTST MOV A,B      GET UNIVERSE PATTERN
0062  CPI 0FFH     CHECK FOR FRINGE STARS
0063  JNZ LSTST    IF NOT ALL PRESENT, CHECK FOR LOSS
0064  MOV A,C
0065  ORA A        GET CENTER STAR
0066  JNZ GTSTR    CONTINUE IF PRESENT
0067  LXI H,MESS4 IF NOT, GAME IS WON. POINT TO WIN MESS
0068  CALL SCRNB
0069 *
0070 *
0071 *
0072 *
0073 *
0074 *
0075  MVI E,'0'  INITIALIZE BINARY TO DECIMAL CONV.
0076  MOV B,E
0077  MOV C,E
0078  DCR D       GET RID OF LAST SHOT
0079  MVI A,'9'+1  SET OVERFLOW CHECK
0080 MRDEC INR E       INCREMENT 1'S
0081  CMP E       CHECK FOR OVERFLOW
0082  JNZ TALLY  CONTINUE IF NOT
0083  MVI E,'0'  OTHERWISE, RESET 1'S
0084  INR C       INCREMENT 10'S
```

```
0085  CMP C
0086  JNZ TALLY
0087  MVI C,'0'
0088  INR B       INCREMENT 100'S
0089 TALLY DCR D       DECREMENT SHOT COUNTER
0090  JNZ MRDEC
0091  MVI A,'0'
0092  CMP B       CHECK FOR LEADING 0
0093  JNZ THREE    IF NOT, DISPLAY 3 DIGITS
0094  CMP C
0095  JNZ TWO
0096  JMP ONE
0097 THREE MOV M,B      DISPLAY SCORE
0098  INX H
0099 TWO MOV M,C
0100  INX H
0101 ONE MOV M,E
0102  INX H
0103  XCHG
0104  LXI H,MESS5  POINT TO REST OF WIN MESS.
0105  CALL SCRN2
0106 PRNT1 CALL KBD
0107  CPI 'Y'    CHECK FOR RESTART
0108  JZ START  IF YES, START AGAIN
0109  RET .       IF NOT RETURN TO ALS-8
0110 LSTST ORA A       CHECK FOR NO FRINGE STARS
0111  JNZ GTSTR  IF ANY ARE PRESENT CONTINUE GAME
0112  MOV A,C
0113  ORA A       IF NOT, CHECK FOR CENTER STAR
0114  JNZ GTSTR  IF PRESENT CONTINUE
0115  LXI H,MESS3  OTHERWISE POINT TO LOST MESS.
0116  CALL SCRNB
0117  JMP PRNT1
0118 GTSTR LXI H,MESS7  ASK FOR SHOT
0119  CALL SCRNB
0120 NXTST CALL KBD
0121  INX H
0122  MOV M,A      ECHO SHOT
0123  CALL DELAY
0124  MVI E,9      SET MASK COUNTER
0125  LXI H,MASK    POINT TO MASKS
0126 NXGRP CMP M        CHECK FOR SHOT
0127  JZ FOUND
0128  DCR E
0129  JZ INVAL    INVALID SHOT IF NOT FOUND
0130  INX H        POINT TO NEXT ENTRY
0131  INX H
0132  INX H
0133  INX H
0134  JMP NXGRP
0135 FOUND  INX H
0136  MOV A,M
0137  ORA A        CHECK STAR POSITION
0138  JNZ UNIV2    JMP IF FRINGE STAR
0139  MOV A,C
```

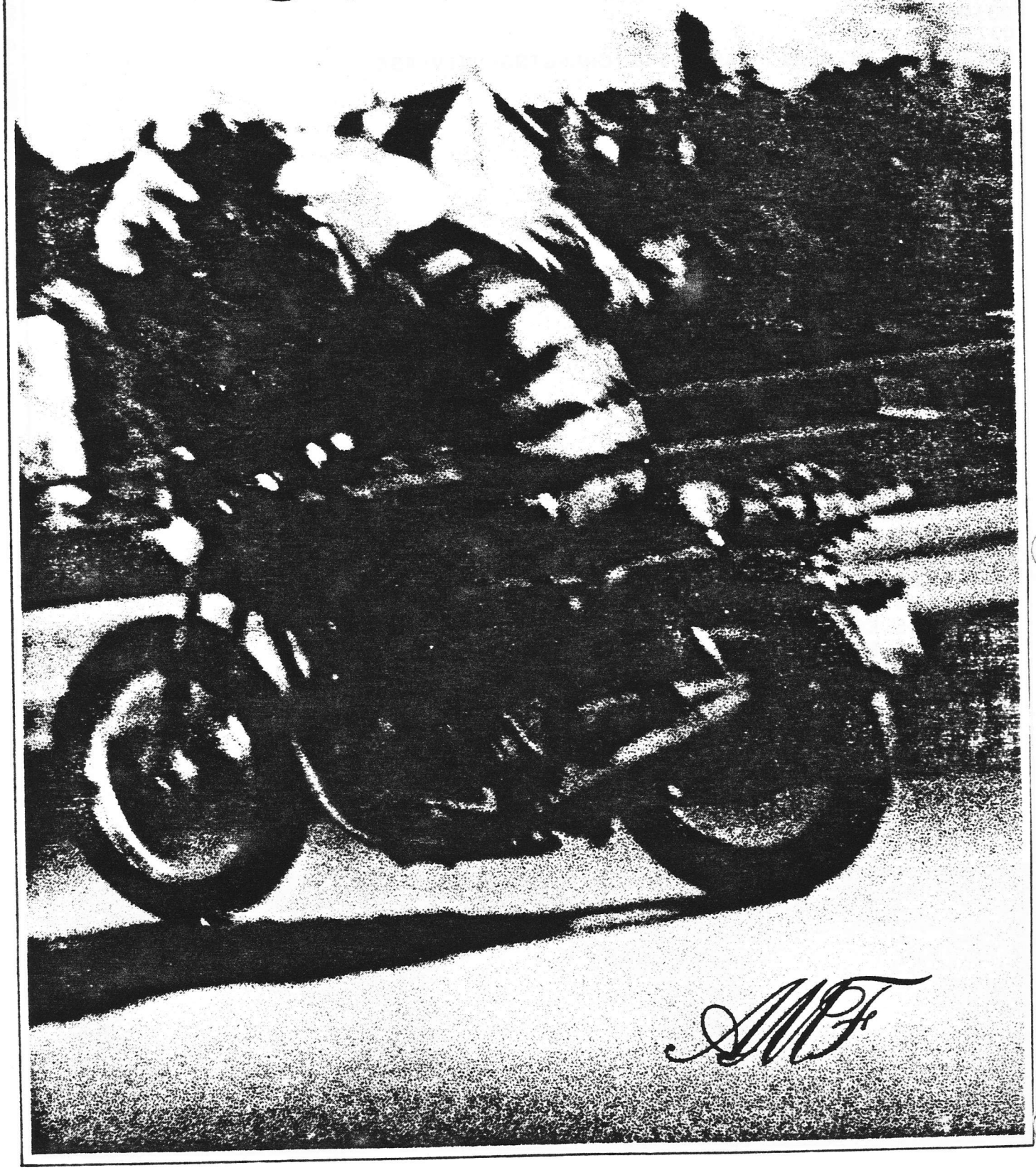
```
0140  CPI 1        CHECK FOR CENTER STAR
0141  JNZ BDFEL    IF NOT PRESENT, BAD SHOT
0142  JMP NXBYT
0143 UNIV2 MOV A,B
0144  ANA M        ISOLATE STAR SHOT
0145  JZ BDFEL    IF NOT PRESENT, BAD SHOT
0146 *
0147 *
0148 *
0149 *
0150 *
0151 NXBYT INX H
0152  MOV A,B
0153  XRA M        ALTER GALAXY
0154  MOV B,A      SAVE NEW PATTERN
0155  INX H
0156  MOV A,C
0157  XRA M        CHANGE CENTER STAR, IF NECESSARY
0158  MOV C,A
0159  JMP CNTST    COUNT SHOT AND DISPLAY NEW UNIVERSE
0160 INVAL CPI ESC      CHECK IF INVALID SHOT WAS AN ESCAPE
0161  JNZ NTVAL
0162  LXI H,MESS6  IF SO POINT TO SURRENDER MESSAGE
0163  CALL SCRNB
0164  JMP PRNT1
0165 NTVAL LXI H,MESS2  POINT TO INVALID STAR MESSAGE
0166  CALL SCRNB
0167  JMP NXTST  GO TO NEXT SHOT
0168 SCRN MOV A,D
0169  STA TEMP    SAVE D (SHOT COUNTER)
0170  LXI D,VDM1  SET SCREEN ADDRESS
0171 SCRN1 CALL CLER  CLEAR & INITIALIZE SCREEN
0172 SCRN2 MOV A,M
0173  CPI EM     CHECK FOR END OF MESSAGE
0174  JZ END
0175  STAX D      DISPLAY CHARACTER
0176  INX H
0177  INX D
0178  JMP SCRN2
0179 END LDA   TEMP  GET SAVED SHOT COUNTER
0180  XCHG
0181  MOV D,A    PUT IT BACK IN D
0182  RET
0183 SCRNB MOV A,D
0184  STA TEMP
0185  LXI D,SCRBT  SET LOWER DISPLAY ADDRESS
0186  JMP SCRN1
0187 CLER PUSH  D      SAVE STARTING ADDRESS
0188  XRA A
0189  OUT VDM    INITIALIZE VDM-1
0190 CLER1 MVI A,' ' GET A SPACE
0191  STAX D
0192  INX D
0193  MOV A,D
0194  CPI BOTT  CHECK FOR END OF SCREEN
```

```
0195  JNZ CLER1
0196  POP D     RESTORE STARTING ADDRESS
0197  RET
0198 TYPE XRA A      CLEAR A & CARRY
0199  MOV A,B    GET UNIVERSE
0200  RRC
0201  MOV B,A
0202 TYPE1 JC STAR
0203  MVI M,'0'  DISPLAY HOLE
0204  RET
0205 STAR MVI M,'*'  DISPLAY STAR
0206  RET
0207 KBD CALL DAV    KEYBOARD INPUT ROUTINE
0208  JZ KBD
0209  IN DATA
0210  ANI 127
0211  RET
0212 DAV IN STAT
0213  CMA .  CAN BE CHANGED TO A NOP
0214  ANI DAVM
0215  RET
0216 BDFEL LXI H,MESS1  POINT TO ERROR MESSAGE
0217  CALL SCRNB
0218  JMP NXTST
0219 DELAY PUSH D        2 SECOND DELAY ROUTINE
0220  PUSH PSW
0221  MVI  D,2
0222 DLY1 MVI E,100
0223 DLY2 XRA A
0224 DLY3 DCR A
0225  JNZ DLY3
0226  DCR E
0227  JNZ DLY2
0228  DCR D
0229  JNZ DLY1
0230  POP PSW
0231  POP D
0232  RET
0233 TEMP DS 1
0234 *
0235 *
0236 *
0237 *
0238 *
0239 *
0240 *
0241 MESS1 ASC "HEY! YOU CAN ONLY SHOOT STARS, NOT BLACK HOLES."
0242  ASC " TRY AGAIN."
0243  DB EM
0244 MESS2 ASC "THAT WASN'T A VALID STAR NUMBER. TRY AGAIN."
0245  DB EM
0246 MESS3 ASC "YOU LOST THE GAME! WANT TO SHOOT SOME MORE"
0247  ASC "STARS?"
0248  DB EM
0249 MESS4 ASC "YOU WIN!!  GOOD SHOOTING!  YOU FIRED "
```

```
0250   DB EM
0251 MESS5 ASC " SHOTS. BEST POSSIBLE    SCORE IS 11 SHOTS."
0252   ASC "WANT TO SHOOT AGAIN, DEADEYE?"
0253   DB EM
0254 MESS6 ASC "YOU GIVE UP TOO EASILY! WANT TO SHOOT SOME MORE"
0255   ASC " STARS?"
0256   DB EM
0257 MESS7 ASC "YOUR SHOT?"
0258   DB EM
0259 HEADR ASC " * * * * * * * * * *  S H O O T I N G  S T A R"
0260   ASC " S * * * * * * * * * "
0261   ASC ". . . . . .                    . . . . . ."
0262   ASC " . . . . . ."
0263   ASC " A    B R A I N   T E A S E R   G A M E ! ! !"
0264   ASC ". . . . . .                    . . . . . ."
0265   ASC " . . . . . . . . . . . . . . . . . . . . "
0266   ASC "DO YOU WANT THE RULES?  (TYPE N FOR NO)
0267   DB EM
0268 PAGE1 ASC "THERE ARE STARS: *  AND THERE ARE HOLES: 0  IN"
0269   ASC " THE UNIVERSE.     "
0270   ASC "YOU SHOOT A STAR, (NOT A BLACK HOLE) BY TYPING "
0271   ASC "ITS NUMBER.          "
0272   ASC " 1  2  3                    "
0273   ASC " *  *  *                    "
0274   ASC " 0  0  0    "
0275   ASC " 4  5  6                    "
0276   ASC " *  0  *                    "
0277   ASC " 0  0  0    "
0278   ASC " 7  8  9                    "
0279   ASC " *  *  *                    "
0280   ASC " 0  0  0   "
0281   ASC "YOU WIN IF YOU GET THE PATTERN IN THE MIDDLE. "
0282   ASC "YOU LOSE IF YOU   GET THE PATTERN ON THE RIGHT."
0283   DB EM
0284 PAGE2 ASC "EACH STAR IS IN A GALAXY. WHEN YOU SHOOT A STAR"
0285   ASC " EVERYTHING IN   ITS GALAXY CHANGES. ALL STARS BECO"
0286   ASC "ME BLACK  HOLES AND ALL BLACK HOLES BECOME STARS."
0287   ASC " . . . . . .GALAXIES:    "
0288   ASC ". . .!. . . .          "
0289   ASC "  1  *  0    *  2  *    0  *  3    *  0  0    "
0290   ASC "0  *  0           "
0291   ASC "  *  *  0    0  0  0    0  *  *    4  0  0    "
0292   ASC "*  5  *           "
0293   ASC "  0  0  0    0  0  0    0  0  0    *  0  0    "
0294   ASC "0  *  0           "
0295   ASC " . . . . . .                    . . . . . "
0296   ASC " . . . . . .          "
0297   ASC " 0  0  *    0  0  0    0  0  0    "
0298   ASC "0  0  0                    "
0299   ASC " 0  0  6    *  *  0    0  0  0    "
0300   ASC "0  *  *                    "
0301   ASC " 0  0  *    7  *  0    *  8  *    "
0302   ASC "0  *  9                    "
0303   ASC " READY TO PLAY. "
0304   ASC "TYPE ANY KEY TO START.  GOOD LUCK!"
```

```
0305   DB EM
0306  *
0307  *
0308  *
0309  *
0310  *
0311  *
0312  MASK DW 0131  MASKS FOR CHANGING UNIVERSE
0313   DW 010BH
0314   DW 0232H
0315   DW 0007H
0316   DW 0433H
0317   DW 0116H
0318   DW 0834H
0319   DW 0029H
0320   DW 0035H
0321   DW 015AH
0322   DW 1036H
0323   DW 0094H
0324   DW 2037H
0325   DW 0168H
0326   DW 4038H
0327   DW 00E0H
0328   DW 8039H
0329   DW 01D0H
0330  VDM1 EQU 0CC00H
0331  SCRBT EQU 0CF00H
0332  VDM EQU 0FEH  CHANGE TO 0C8H FOR USE WITH VDM-1
0333  STR1 EQU 0CD1BH  STAR LOCATIONS
0334  STR2 EQU 0CD20H
0335  STR3 EQU 0CD25H
0336  STR4 EQU 0CD9BH
0337  STR5 EQU 0CDA0H
0338  STR6 EQU 0CDA5H
0339  STR7 EQU 0CE1BH
0340  STR8 EQU 0CE20H
0341  STR9 EQU 0CE25H
0342  STAT EQU 0FAH  KEYBOARD STATUS PORT
0343  DATA EQU 0FCH  KEYBOARD DATA PORT
0344  DAVM EQU 01H  DATA AVAILABLE MASK
0345  BOTT EQU 0D0H
0346  EM EQU 01H
0347  ESC EQU 1BH
```

Guess Who??



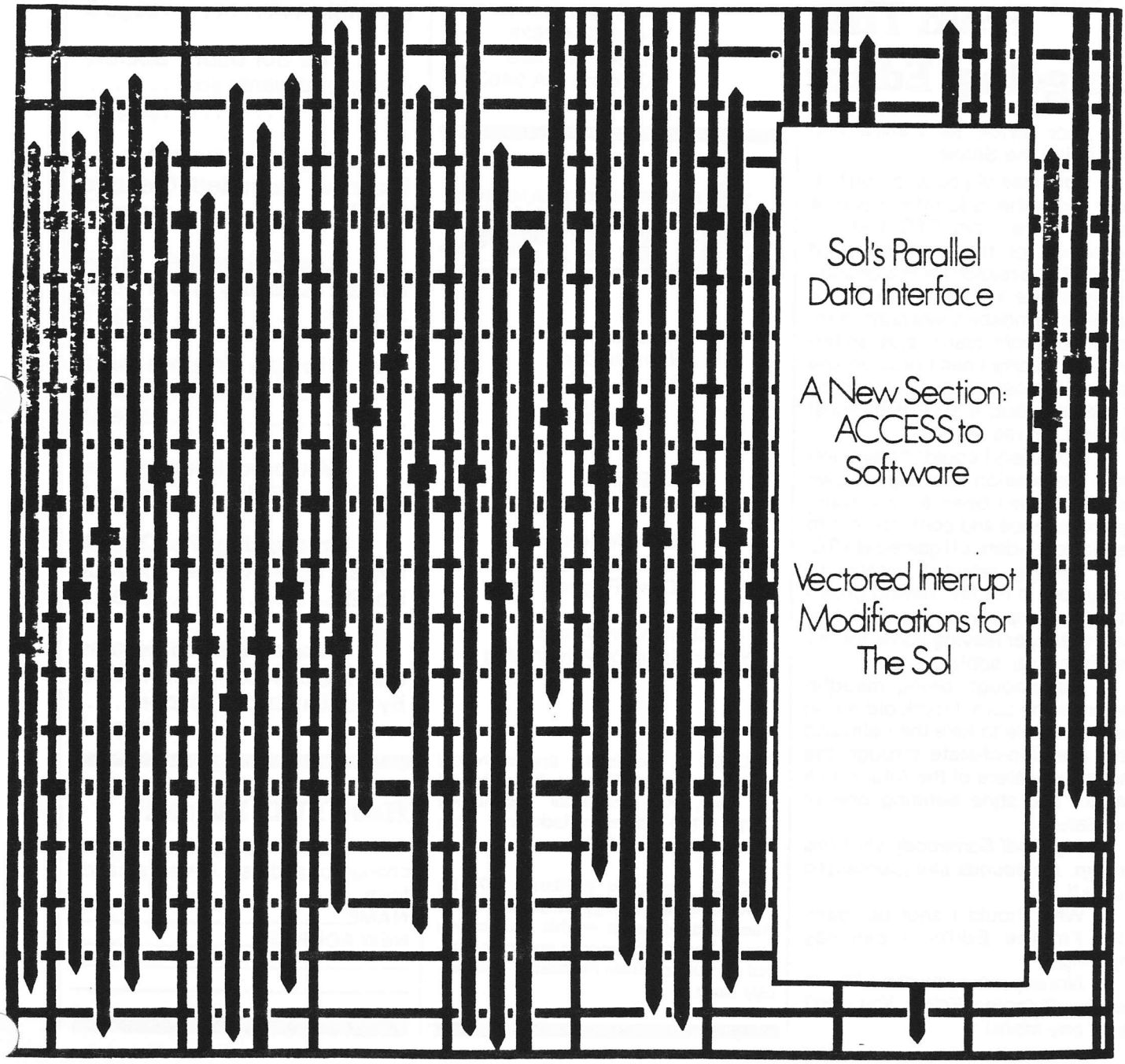
1 EPT 2 - 87

PROCESSOR TECHNOLOGY

ACCESS.

75¢ per issue

Published by Processor Technology Corp., Pleasanton. Volume Two, Number One. March 1978



Sol's Parallel Data Interface

A New Section: ACCESS to Software

Vectored Interrupt Modifications for The Sol

A Letter From The [Outgoing] Editor

"Alas poor Yorick, for I knew him well." Willie the Shake

For those of you who don't already know, this is to inform you of my departure from PTC and my abdication of the editorship of ACCESS. The reason for this change is fairly simple: I'm going to work for myself as an independent consultant (free-lance technician in plebian terminology). Sorry I can't provide any scandalous gossip about my reasons for leaving. But, if you really must ask, well, let me tell you...

Obviously I couldn't have gotten to the position of being my own boss if it hadn't been for the training, experience and contacts (not to mention confidence) I gained at PTC.

Leaving definitely creates an empty spot in my life, having grown with the company (2 years). I feel like a teenager leaving home for the first time (sob, sob!).

But enough being maudlin and on to the corn. I think old Ralph should be able to take the helm and steer the ship-of-state through the uncharted waters of the future, in a manner and style befitting one of this stature.

My God! Somebody shut this fool up. He sounds like journalistic Muzak!!

Why should I shut up, dammitt. I'm the Editor. I can say anything.

Not any more you can't. You're just the Outgoing Editor. You don't count any more!

Well, as you can see, there are good reasons for me being locked up in this place. And, what the hell, the food's OK.

I guess it's time to go, so I'll close this chapter of history with a sentimental goodbye to y'all. And expecially to Les Solomon who could be a guru. But whoever heard of a Jewish guru?

Love and kisses to all,

Aram Attarian

Aram Attarian II
(415) 534-6628
1924 - 10th Ave.
Oakland, CA 94606

PROCESSOR TECHNOLOGY
ACCESS.

Volume Two,
Number One
March 1978

Publisher
Processor Technology Corp.

Editor
Ralph I. Palsson

Staff Writers
Lee Felsenstein
Drew Rogge

Design/Production
Wible/Rampton Advertising

ACCESS is published approximately every six weeks. Subscription rate: $4 per year, from Processor Technology Corp., 7100 Johnson Industrial Way, Pleasanton, CA 94566.

ACCESS Copyright © March 1978 by Processor Technology Corp. All Rights Reserved. Material in this publication may not be reproduced in any form without permission from Processor Technology Corp.

Table of Contents

Reader Feedback . . . . . page 3

A review of our Extended Cassette BASIC, now available . . . . . page 6

The Sol Users' Society grows and wants you! . . . . . page 6

Adjusting BASIC/5 for the new Extended Cassette BASIC . . . . . page 7

Lee Felsenstein elucidates interfacing to Sol's parallel port . . . . . page 9

Having trouble with your 16KRA? Check this . . . . . page 11

Vectored interrupt capability for Sol . . . . . page 11

Interfacing the OP80A Paper Tape Reader to your Sol-20 . . . . . page 13

A selection of programs and applications contributed by our illustrious readers . . . . . page 14

Have YOU Moved?

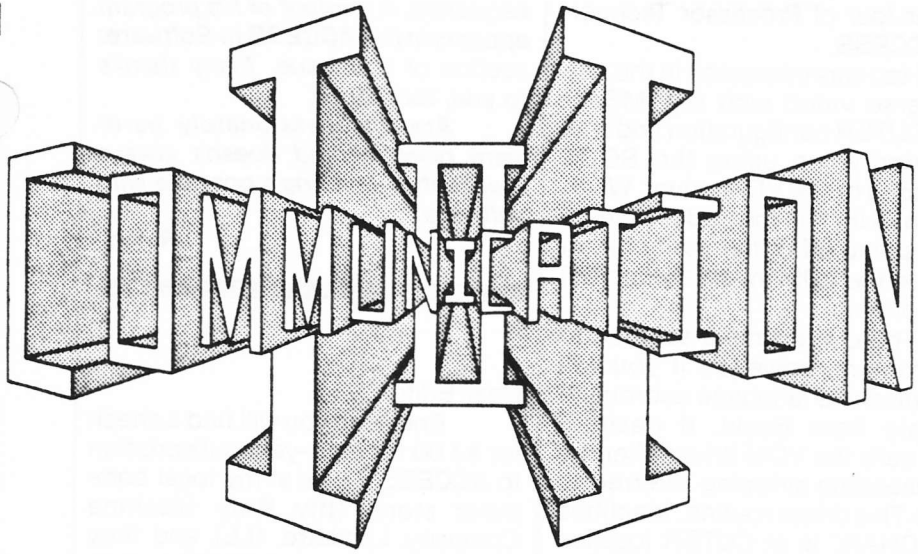
Please notify us of your change of address. Here's a handy form:

NAME_____

NEW ADDRESS_____

_____

_____



A Letter From The [New] Editor

That's me, Ralph Palsson. Remember me from One-to-One Communication? I'm changing hats because Aram Attarian II, former ACCESS editor, lead Warranty Repair Applications Engineer, customer interfacer, and all-round good guy, has succumbed to the siren song of "making it" freelance. He's setting up an independent systems consultant and repair technician, and we're going to miss his red suspenders around here. He devoted a lot of creative energy and talent to making ACCESS what it is today, so I'll do my best to keep up the good work. For Aram's last words of wisdom to us, see his outgoing editorial on p. 2.

Now that I have this column to sound off in, we'll probably discontinue One-to-One Communication as separate feature. The original idea behind ACCESS was to make it a vehicle for one-to-one communication in its entirety, and that's beginning to happen, as you'll see from the many excellent reader contributions in this issue. I'll be laying it pretty heavy on you readers to keep up the good work—send us your program listings, letters, hardware modifications, applications, anything you think is worthwhile and would like to share.

The main change I initially hope to make in ACCESS is getting it on a somewhat more regular publication schedule, aiming at every 6 (±2) weeks. And I promise that everyone who had paid $4.00 for a year's subscription will actually receive 8 issues, regardless of how many leaves get torn off the calendar. As for the new format—well, you people seem to be as enthusiastic about it as we are, and we'll keep adding whatever improvements we think of. Again, your suggestions are valuable, like the punched-hole idea several readers have mentioned.

I like to think of ACCESS as one proof of the pride Processor Technology takes in providing customer support. So far, we have a pretty good track record, but we aim to make it even better. We have a very user-oriented Customer Service Dept., plus the more technically-inclined applications and product support engineers regularly available to help you with debugging and troubleshooting your kits. They're also good with advice on how to interface a Sol or 3P+S to a peripheral device. And we even have a customer service software consultant available for phone discussions. But there's one thing none of these people can do, and that's help you with support for XYZ company's floppy disk or ABC's 16K board. No matter how expert they are, our engineers can't know the personal design quirks of a product they may never have seen, and really your guess is as good as theirs many times. We'll do as much "hand-holding" as we can by phone or letter, but please understand the distinction between help with interfacing and help with someone else's product.

Our Customer Service technicians are currently available daily from 1:30 to 4:00 p.m. PST for phone consultation. The number again is (415) 829-2600. We'll increase the hours as soon as we can add more trained personnel who can give you straight answers you deserve.

I'll end with my usual plug for Processor Technology dealers. These guys can provide you with a wealth of information and assistance without running up your phone bill. Our intentions are good, and we back them up with action as much as possible, but it's a fact of life that we're way out here in Pleasanton, CA. And if you're in Manhattan, Bloomington, or even Lompoc, you will probably get more personal service from the dealer who's helped you plan your system in the first place than you can get from a voice on the phone. The idea is to give you the best support possible, wherever it comes from.

Well, hope you enjoy ACCESS #5, and in the words of AA II, y'all have a good time, y'heah?!

Ralph I. Palsson

Dear Editor:

HISTOGRAMS IN BASIC
(Or how to display multiple cursors)

Attempting to use Basic 5 for plotting the results of stock analysis, I wanted to plot a graph of the derived results. An approach has been

developed,using the TAB function to start a plot, and a calculated value of a variable to end the plot. I also wanted to use reverse video to highlight data or to plot a bar made up of (aO hex) "cursors." Initially this was difficult, since many 7F masks were used in the VDM driver within CUTER to maintain a clean display. Therefore the VDM driver cannot be entered with a CALL to its entry address, since bit 7 is cleanly stripped off. The solution to this problem lies in addressing that portion of the VDM driver starting at location CØBB after having put the AØ into register B. That can be done as an immediate move in machine code necessitating a call from Basic to your machine code which in turn calls the driver, or by using the ARG command in basic before the call to the driver. This latter approach has an advantage in that the value of the ARG can also be calculated thus giving you control over the character chosen for the plot. (Incidentally, the 7F, delete character makes a nice chart due to the spacing between bars.) These ARG and CALL functions are used within a FOR-Next loop which is set up as described in the beginning of this description. That looping puts out the desired number of repeated characters and leaves VDM screen management control within the Basic/Cuter/VDM driver software management system.

In closing, let me send my praise and thanks for your product quality and utility. The PTCO additions have really enhanced my IMSAI.

Donald F. Petrie
831 Ponderosa Dr.
So. Daytona, Fla. 32019

Donald also enclosed a carbon of a letter he wrote in response to Dave Wrobel's query in the last ACCESS:

January 10, 1978

Dear Dave:

Don't despair! Inverse video is attainable. I noticed your letter and Mr. Attarian's response in issue number four of Processor Technology ACCESS.

I too was interested in the use of inverse video with my IMSAI/VDM/CUTER configuration and was frustrated when using the SOUT entry point of the VDM driver. When I use an ANI 7F in the code, a clean-up, stripping off of the parity bit (in our case the VDM inverse video bit) is performed.

There is a simple solution to the problem, however, if you will use a machine language subroutine (callable from Basic, if desired) which calls the VDM driver after the housekeeping stripping referred to above. This driver routine, identified as "OCHAR," is at CUTER location C0BBH and in SOLOS at location C098H. Calling the driver at OCHAR with bit 7 high will result in the desired inverse video display.

You have a number of options in approaching this problem. In machine code, put the character into Register B, with bit 7 high and then call OCHAR. In BASIC 5, establish the character as the value of an ARG function and use the Basic ARG/CALL routines to call OCHAR. And if you want keyboard on-off toggling control of this feature, a simple custom output driver invoked by the SOLOS/CUTER pseudo-port 3 routine can be written to sense your command choice; then either add or strip-off bit 7 as appropriate and put the character in register B before calling OCHAR and returning to your system.

I hope one of these suggestions will be of use to you if you haven't already found a solution to your desire for inverse video display.

D. F. Petrie

cc: Mr. Aram Attarian
Editor, Processor
Technology Access

Dear Donald,

Yes, Virginia, Inverse Video is attainable in Sol BASIC/5, thanks to your information. Reader Tom Digate has also sent us a contribution on this subject, a cassette written in BASIC/5 demonstrating the uses of SOLO's VDM driver escape sequence. A printout of his program appears in the ACCESS to Software: section of this issue. Many thanks to you, too, Tom.

Aram is passionately hardware oriented and doesn't always realize that anything's possible with software!

Ralph

Dear Editor:

Enclosed you will find a check for $4.00 for my 1-year subscription to ACCESS. I was at my local computer store, (Itty Bitty Machine Company, Lombard, ILL), and they gave me a copy of ACCESS to look at. I thoroughly enjoyed reading the articles in the November issue. Congratulations on your fine publication. I have a question that you or maybe your readers could help me on. I have a IMSAI 8080 Micro Computer with the following boards: a 16K Processor Tech. Memory, VDM board by PTC, and my problem, 1 Tarbell Cassette interface board. I am going to purchase your (Sol) Basic Computer and I would like to know if there is a program to convert the CUTS format to Tarbell format for my cassette interface. I would like my IMSAI to run Sol software. If anyone can help, please do.

Thank you.

Robert Bennett
2046 Louis Street
Melrose Park, ILL 60164

Dear Bob,

Thanks for your letter; an ACCESS subscription has been entered for you and hopefully you're now reading this in your first copy.

Note that Sol BASIC/5 is an interpreter, not a compiler. If you're feeling any confusion about it, I'm sure your dealer can help.

About the CUTS/Tarbell matter. PTC doesn't support the Tarbell cassette interface with hardware or software, because we feel that CUTS is a superior system for several reasons. The CUTS (Computer Users Tape Systems) recording method was derived from and is

compatible with the Byte (Kansas City) standard recording method, a system that PTC helped develop at the 1975 Kansas City meeting. We believe strongly in promoting software standardization and have put a lot of effort into the cause. CUTS was especially designed to permit widespread dissemination and exchange of low cost software. To that end, CUTS tapes are less expensive and easier to produce than Tarbell; PTC has shipped over 30,000 of them to date with less 1% failure rate—not bad, if we do say so ourselves! And although CUTS is slightly slower than Tarbell, it requires less adjustment to work with a greater variety of tapes and cassette recorders—again facilitating standardization and exchange.

So while we can't be too encouraging about your conversion project, we hope you will continue your interest in CUTS and Sol software. We think you'll end up very happy with it.

Ralph

Gentlemen,

I am enclosing a listing of a game written in BASIC called Hammurabi. I found it in one of the hobbyist magazines, Personal Computing, I think. I modified it to run in Basic 5 and I and my two boys have enjoyed playing it. I thought perhaps other Sol owners might also enjoy it. It will run in 8K and I run it on the CRT. I have enclosed a printout of a typical "year" of the game ending with a deliberate error which ends the game with a loss of kingdom. The entire successful game would have been too long a printout and I think the short example will give anyone looking at it the idea of how it's played.

I enjoyed the first issue of the new format of *Access*. Keep up the good work.

I visited your booth at the New York Coliseum and was very taken with the Sol set up to record a mailing list on tape which was open to public entry. If a copy of this program is available, I would very much like to obtain one.

Best wishes for your continuing success.

John Dowd
Lynn's Hair Fashions
New Road, Central Ave.
Linwood, N.J. 08221

Dear John,

Thanks for your comments and contributions. The mailing list program you saw in New York was just a demo version and not available in its present form. However, our software development is now shifting to an emphasis on applications, and you'll find many exciting programs forthcoming in the near future.

To our readers,

Besides the Hammurabi program, John also sent us two others, PAROL and COMM. All three appear in the ACCESS to Software section—try them out!

Ralph

RE: Sol20

Gentlemen:

I purchased my Sol20 mid December 1977. After several hours of building (72) the Sol, to explicit and well defined hardware assembly instructions, I had a letter perfect "turn on." My compliments to all connected with this portion of the manual and hardware parts staff.

I have encountered one problem which I have since corrected. This was a very heavy smearing and unstable effect on my video monitor. Distortion severe enough to force shut down manually and of course interrupting work in process. I could still "SAVE" etc. But it was very difficult to read monitor and inputs, when instability occurred. Being more familiar with R.F., video and communications equipment than logic, I found a quick and normal solution to the problem, one which is a standard practice in the R.F. field.

Solution:

The clock generator at "14.31818 MHz" is fairly open to external field excitation, by nearby C.B. and other transmitter frequencies. This unprotected portion of your circuit tends to wander if excited by an external signal and does not settle down. The reason is the XTAL can itself, floating above ground. A short jumper between the can and ground corrected the problem immediately.

I hope this assists other Sol users.

Now the bad news. I realize most people purchasing Sol's are probably more software oriented than I am. Therefore, your company did a painstaking task on the hardware. But your software write-up for a beginner (28 years communications) in computer talk and functions is very poor.

I know it is difficult and a lot of self study must be done, which I am now doing, but!

Might I suggest you acquire some unsuspecting person with no software knowledge, sit him/her in front of the machine, hand him "Console Manual" and "Solos Manual" and watch the confusion. If it is me, I apologize. But after showing your book to 3 top software experts, they agreed, it was written by a software expert.

Example:

After 10 hours of reading, re-reading, 1 trip to vendor (80 miles), no luck, re-reading, etc., trying to work out the CRC computation. Nowhere could I see it written.

"When you desire to save a file. Input SAVE (title) 5 words and press "RETURN."

N.B. "Making sure recorder is in record mode."

But most important, "File header is then automatic."

I guess I'm saying try to use layman's language, if possible. If you re-read "SOLOS" I am sure you will agree.

I am very satisfied with the unit and want to congratulate those who worked hard on design and appearance. Well done.

B. W. Poste
17 Lonsdale Place,
Barrie, Ontario

Dear B.W.,

Thanks for the compliments, and for the hardware fix. Many of our readers will find your contribution from the RF/analog world extremely useful.

Your critical points on documentation and software are very worthwhile. PTC's Technical Documentation Dept. is always looking for input that will help future publications be more clearly presented, on the appropriate level of expertise. Meanwhile, we'll try to provide some concrete followthrough in the next issue of ACCESS.

Ralph

Gentlemen:

Loved the new ACCESS format, but tell me, will my one-year subscription be extended to cover the first 8 issues (since you nominally publish every 6 weeks)? You haven't quite been able to keep up with the announced schedule for either ACCESS or ALS-8 users notes, (though the latter are not officially a product of P.T.). Also, (this may sound like a dumb question), is there any chance of re-publishing the 1st 3 issues in the new format? Even if it costs $2-$3 per copy? The new format is *so* much better, especially for my only good source of a CUTTER (or SOLOS) listing.* Thanks.

Barry Watzman
2330 Millenium Lane
Reston, VA 22091

*Besides, then I could keep a *complete* collection of ACCESS in a 3-ring binder!

Dear Barry,

Yes. We'll make sure you (and other ACCESS subscribers) receive 8 issues before you come up for second year billing. Also, we aim to put the journal back onto a more reasonable schedule.

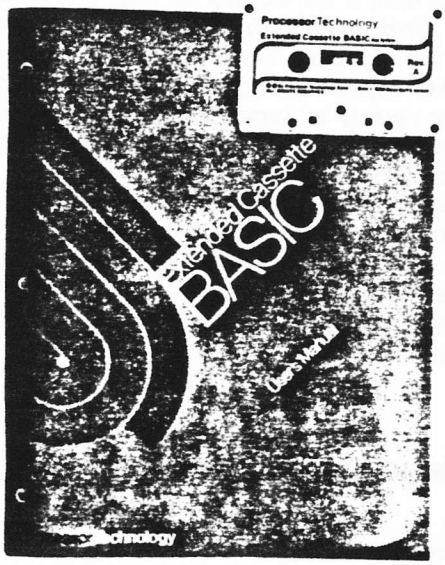
As you might guess, costs of republishing out-of-print ACCESS issues are prohibitive. However, I'm thinking of possibly doing a collection of the "Best of ACCESS" as a separate issue or reprinting the most requested articles from past issues. How do you other readers feel? (NOTE: We still have a lot of our 2nd issue on hand; numbers 1 and 3 are out of print.)

Ralph

Extended Cassette BASIC Is Here!

Our long-promised Extended Cassette BASIC has been released and is now at your local PTC dealer. It's a powerful language, and we'll be publishing assorted program listings in future ACCESS issues to stimulate your imagination on its applications. Just for starters, note that Extended Cassette BASIC features complete string functions and matrix operations. And the documentation is excellent.

It's rumored there may be some bootleg copies floating around. Assure yourself of getting a quality tape and complete package by buying only if the cassette and manual are identical to those shown in the photograph here.



There's one slight bug we've discovered in using manual FOR/NEXT loops. The bug occurs when there is a FOR/NEXT loop within a FOR/NEXT loop which will not allow the execution of the inner loop. The squasher allows the execution of the internal loop rather than searching for a loop elsewhere in the program. It's easily fixed by following this 5-step procedure:

1) GET BASIC from the tape. Note the load address and length.

2) Type EN B3E in SOLOS. (CR)

3) Type the following numbers, separated by a space:
C1 CA 2E 0B/(CR)

4) SAVE BASIC (refer to Section 3.4 in the BASIC User's Manual). Let the tape run in the RECORD mode for 10-15 seconds before re-recording. You can use a new tape if you wish, but the original one will work just as well. Use the load address and length information you obtained in Step 1 of this procedure, and make the auto start address 0 and ending address 3F84.

If you're using the original cassette, affix scotch tape over the two holes on the back of the cartridge. Remove tape when finished; this will afford you "write protection."

The Sol Users' Society Is Up And Running

Last issue we reported on the formation and first meeting of SOLUS, the Sol Users' Society. It's an organization for owners of Sol computers or of Sol-type computers

compatible with other PTC products. Specifically, if you have an 8080 or Z-80 microcomputer you're using with SOLOS, CUTER, SOL/CUTS cassette interface, or any functionally equivalent operating system, you might like to join. The membership is world-wide.

Currently, SOLUS has several good services going, beginning with their own bi-monthly (approximately!) newsletter to keep members informed on hardware, software, new products, bugs, local chapter meetings, and other items of interest. They maintain a Software Library which collects and distributes public domain programs for a nominal charge. Local chapters hold meetings to exchange software and ideas, and the club headquarters, in convenient proximity to PTC, keeps a close communication link going with us. Finally, qualified SOLUS volunteers are testing products for Sol compatibility and reporting their experiences in the newsletter.

To refresh you on the club's goal and organization: The stated goals are to facilitate communication among SOLUS members, to provide a mechanism of exchange for Sol-compatible software, to give PTC feedback from SOLUS members, and to encourage the development and testing of Sol-compatible hardware and software produced by independent sources. Their relationship with PTC and other manufacturers is co-operative but independent. SOLUS is primarily a personal computer users' group, but special interest groups can be formed within the club on subjects like commercial use, health care, education, scientific applications, etc. The society is supported by dues and volunteer efforts.

To join SOLUS: If you live in the U.S., Canada or Mexico, send $10.00 in check or money order to SOLUS, P.O. Box 23471, San Jose, CA 95153. If you live anywhere else, send $15.00. Dealers and manufacturers of Sol-compatible equipment or software should contact SOLUS at the above address for details on special memberships. □

What To Do With Your BASIC/5 Programs Now That You Have Extended Cassette BASIC

Unfortunately, programs written in Sol BASIC/5 won't run with the new Extended Cassette (formerly called 8K) BASIC interpreter. But you can save yourself several thousand tedious keystrokes by using the following assembly language program to save your BASIC/5 programs in text form on CUTS cassette. Then it will be easy to get and run a text file on the new interpreter whenever you wish.

The program uses the byte access tape routines in the SOLOS/CUTER monitor. CLIST is written to be used as custom output diver in conjunction with pseudo port 3.

When you use the program, load the machine code at the addresses indicated in the assembly listing below. Set the custom output port address to the beginning address of this program. Get and execute BASIC/5. Get the program you want to write to the tape. From BASIC/5, set the output port to 3 and type LIST.

After your program has been saved, this routine will set the current output port back to 0. □

```
                        *
     000D               CR      EQU     0DH     ASCII carriage return
     000A               LF      EQU     0AH     ASCII line feed
                        *
                        *  Solos equates
                        *
     C01C               AOUT    EQU     0C01CH
     C007               FOPEN   EQU     0C007H
     C00A               FCLOS   EQU     0C00AH
     C010               WRBYT   EQU     0C010H
     C807               OPORT   EQU     0C807H
                        *
                        *  The program starts here.
                        *  This is also the entry point.
                        *
                        *
CB00                            ORG     0CB00H
                        *
                        *
     CB00               CLIST   EQU     $
                        *
CB00 C5                         PUSH    B       Save registers
CB01 D5                         PUSH    D
CB02 E5                         PUSH    H
                        *
```

```
CB03 78                         MOV     A,B     get character
                        *
CB04 FE 0D                      CPI     CR      Carriage return?
CB06 C2 26 CB                   JNZ     CL2
                        *
CB09 21 73 CB                   LXI     H,LCHAR
CB0C 7E                         MOV     A,M     Get last character
CB0D FE 0D                      CPI     CR      If not two carriage returns in
CB0F C2 26 CB                   JNZ     CL2     ...in a row then nothing special
CB12 23                         INX     H       Else check if first set
CB13 7E                         MOV     A,M     ...of carriage returns
CB14 B7                         ORA     A       If not then close file and
CB15 C2 4B CB                   JNZ     CLOOP   ...reset program
CB18 3D                         DCR     A       Else say we already
CB19 77                         MOV     M,A     ...had first time through
CB1A 3A 62 CB                   LDA     UNIT    Get unit to write to
CB1D 21 63 CB                   LXI     H,HEADR  Point to tape header
CB20 CD 5E CB                   CALL    OPNOP   Open the file
CB23 C3 47 CB                   JMP     GBACK
CB26 78                 CL2     MOV     A,B
CB27 FE 0B                      CPI     LF+1
CB29 DA 2F CB                   JC      CL3
CB2C 32 73 CB                   STA     LCHAR
                        *
CB2F 3E 00              CL3     MVI     A,0     Reflect character to screen
CB31 CD 1C C0                   CALL    AOUT
                        *
CB34 3A 74 CB                   LDA     FIRST   Check if writing yet
CB37 B7                         ORA     A
CB38 CA 47 CB                   JZ      GBACK   Return if not
                        *
CB3B 78                         MOV     A,B
CB3C FE 0B                      CPI     LF+1
CB3E DA 47 CB                   JC      GBACK
                        *
CB41 3A 62 CB                   LDA     UNIT    Get unit to write to
CB44 CD 10 C0                   CALL    WRBYT   Write the byte
                        *
CB47 E1                 GBACK   POP     H       Restore registers
CB48 D1                         POP     D
CB49 C1                         POP     B
                        *
CB4A C9                         RET     .       ...to Basic/5
                        *
     CB4B               CLOOP   EQU     $
                        *
CB4B 3A 62 CB                   LDA     UNIT
CB4E CD 0A C0                   CALL    FCLOS   Close file
CB51 AF                         XRA     A
CB52 32 73 CB                   STA     LCHAR   Reset program for next
CB55 32 74 CB                   STA     FIRST   ...time through
CB58 32 07 C8                   STA     OPORT   Change output port to 0
CB5B C3 47 CB                   JMP     GBACK
                        *
                        *
     CB5E               OPNOP   EQU     $
                        *
CB5E CD 07 C0                   CALL    FOPEN
CB61 C9                         RET
                        *
CB62 01                 UNIT    DB      1       Change to 2 to use tape unit 2
                        *
CB63 43 4C 49 53        HEADR   ASC     'CLIST'
```

```
     54
CB68 00                        DB      0
CB69 54                        DB      'T'     'T' for TEXT
CB6A 00 01                     DW      256     Length of block
CB6C 00 00                     DW      0
CB6E 00 00                     DW      0
CB70 00 00                     DW      0       Three spares
CB72 00                        DB      0
                       *
CB73 00                LCHAR   DB      0
CB74 00                FIRST   DB      0
CB75 00                INFLG   DB      0
                       *
```

Interfacing To The Sol's Parallel Port

by Lee Felsenstein

The parallel input/output port on the Sol, J2, is intended to allow highspeed transfer of data between the CPU and outside devices. The circuitry was designed to be symmetrical, so that two devices having identical interfaces can communicate with each other. In contrast, the 3P+S circuit puts out a positive-going data strobe pulse but requires either a negative-going data strobe pulse or a steady level in the return direction. As a result, two 3P+S boards cannot communicate without additional circuitry, while two Sols can communicate through their parallel ports with nothing but wires between them.

Data is transferred on eight-bit-wide paths into and out of the Sol, through the pins of J2 labelled PID0-7 (data into the CPU) and POD0-7 (data out of the processor). Polarity of the data on these pins is positive—that is, a "high" logical level is equivalent to a binary "1." Data is latched at the output side of the interface and is not latched at the input side. This means that after an output instruction to the parallel port (port FD) is issued, the new data remains present at the output pins of the connector indefinitely, until the next output instruction is issued. Similarly, the external device which provides data to the Sol through J2 must keep its data present on the PID pins long enough for it to be "picked up" by the Sol CPU.

FIG 1.
HANDSHAKING SEQUENCE

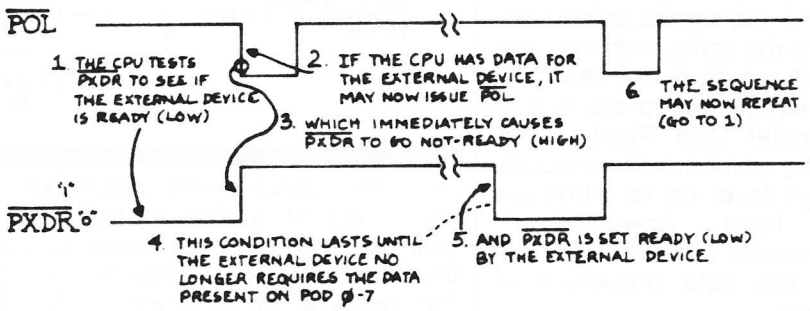
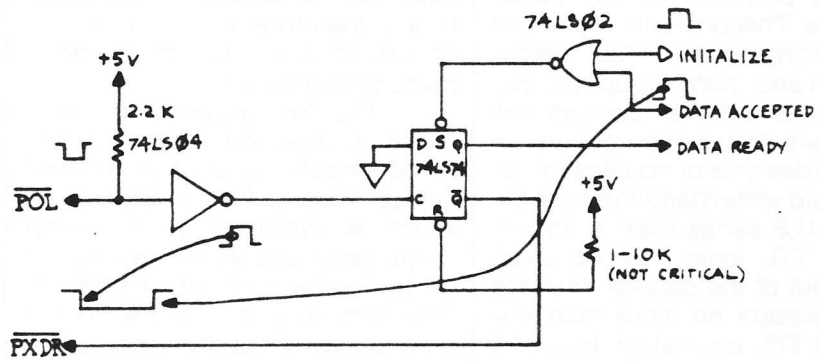


FIG.2.
CIRCUITRY FOR HANDSHAKING



Supervisory signals, known as "handshaking" signals, are provided in both directions to allow the devices on each side ot the interface to know when data has been accepted and when each may present the next eight bits of data. Each side issues a negative-going "strobe pulse" when its data is ready and latched. In the direction going out from the Sol, this signal is called POL, for Parallel Output Load. The line over the signal name signifies that the signal is "active low." All of

the handshaking signals are active low, so that a disconnected plug automatically results in a "not ready" condition. Resistors connected from the in-coming handshaking signals to +5 volts will "pull up" these pins to a high level automatically when this happens. Due to the pecularities of TTL logic, it is almost impossible to "pull down" a disconnected input.

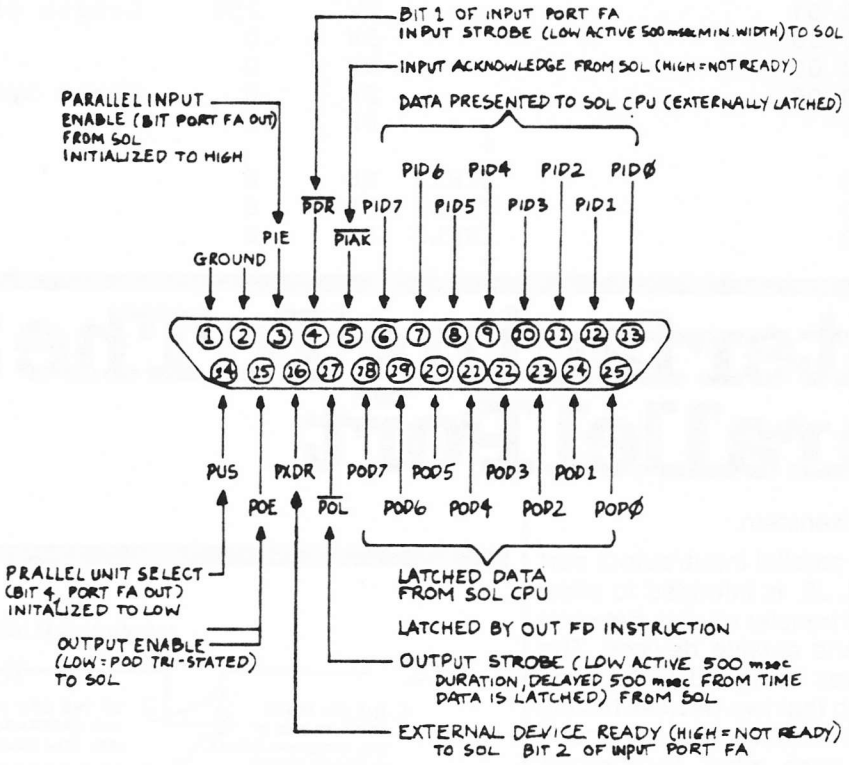
The external device sees the POL signal and immediately acknowledges it with a high level on the PXDR pin, (Parallel External Device Ready). The Sol CPU will test the level of this signal before issuing a new POL. The external device sets the PXDR low again as soon as it has accepted the data and no longer requires its presence on the POD pins. Fig. 1 shows the sequence relationship between these signals.

In the incoming direction to the Sol CPU, the signals and the sequence are the same, with only the names and pin numbers changed. The data strobe fed to the Sol is PDR, (Parallel Data Ready). The Sol circuitry immediately responds with a high level on its PIAK pin (Parallel Input Acknowledge), which continues until the CPU has accepted the data presented at the PID pins.

Fig. 2 shows a suggested circuit for providing the handshaking signals from an external device. Note the provision for an "Initialize" pulse. This prevents the circuit from starting up in a "not ready" condition and "hanging up" the Sol CPU. Some external devices will not require this signal, but as an interface designer or troubleshooter, you should understand the possible pitfall. 74LS series logic is shown, but any TTL logic may be used. Each input of the parallel interface should present no more than one standard TTL equivalent load (1.6 maximum current source).

Three signals remain unexplained so far. These are PIE (Parallel Input Enable), POE (Parallel Output Enable), and PUS (Parallel Unit Select). These are direction-control signals to allow sharing of data lines among several devices. The latches in the Sol have

FIG 3
J2-VIEWED FROM REAR OF SOL



"tri-state" output drivers, which can be "turned off" so that another device can take over the same data lines with its own tri-state drivers. The signal which turns off the Sol drivers is POE (Parallel Output Enable). This signal is high-active, so that a low level on this pin will cause the output drivers to release. It will therefore keep the output drivers on when no connection is made to that pin of J2.

The Sol provides a signal which is intended to control the output-enable signal of an external device. This is PIE, a latched bit which is initialized to a logical "high" level and which can be set be setting bit 3 of output port FA. (The remaining bits of FA control other things, so care must be taken to keep them in the proper pattern when such an output is performed.) PIE may be used as a "direction control" bit if you want to share the data lines in both directions: for example, if you need to save wire or to use the interleaving wires of a ribbon cable for grounds to provide shielding on long runs. PIE would then be connected to POE of the Sol and to a similar signal on the external device which recognized a "low" level as its output-enable signal. The PID and the POD lines may now use the same eight wires.

Note that the layout of the pins on E revision Sols and above (F, etc.) is such that the PID 0 and POD 0 lines are physically adjacent. This allows easy cross-connection of these lines in parallel when using a ribbon cable. If a 25-wire ribbon cable is used, the interleaving wires are free for use as grounds, helping to reduce the problems of "glitches" which often plague high-speed parallel interfaces.

The PUS (Parallel Unit Select) pin is another latched bit, bit 4 of output port FA, which is initialized to a logical "low" level. It may be used to select two different external devices to receive or transmit data. These devices may share the same PID lines if their outputs are tri-state. Together with the PIE bit, the PUS bit allows the CPU to select four external devices, providing that appropriate decoding and response

circuitry is included in the external device.

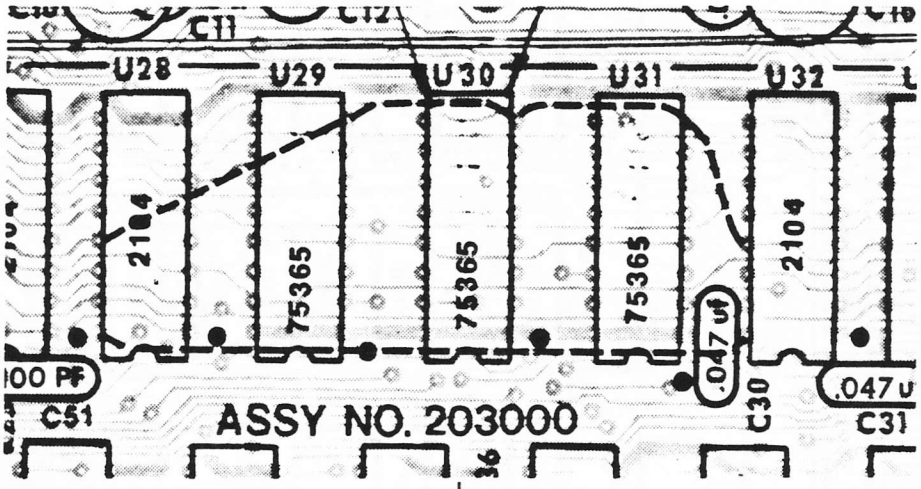
Like all logic-level signals, the parallel interface signals should be kept away from high voltages such as AC and static electricity. Cables longer than 10 feet should not be used; or if they are, precautions should be taken to accommodate the "ringing" effects which will result during signal transitions. The most typical such precaution is "terminating" the line with a resistor, usually about 120 ohms connected to +5 volts. The output circuits of the Sol parallel port cannot drive a load of less than 330 ohms, so additional drivers may be necessary for such terminated lines. It is also a poor practice to connect edge-sensitive inputs such as the clock or trigger input of a flip-flop to the cable directly. Use an inverter or a Schmitt trigger circuit such as the 74LS14, 8837, or 8836 (8T380). Edge effects can be impossible to see on an oscilloscope, so precautions are usually advisable when attempting to "debug" a cranky interface.

The Sol parallel interface provides many opportunities for imaginatvie and economical interfacing. we would be interested in hearing your results in applications which you consider noteworthy. Good luck!

Important Notice To 16KRA Owners

We're currently up to Revision Level J on the 16KRA memory board in our tireless quest to improve the reliability of the circuit. Most of the earlier boards work fine, and if yours isn't exhibiting any intermittent failures, you can stop reading now. We don't recommend making the modification unless you are having trouble.

However, here's the modification procedure for those of you who need it. The assembly in question is Number 203000, shown on page VI-1 of the 16KRA manual. The portion of the drawing reproduced here, Detail A, shows the modification to increase reliability. Five jumper wires have been added to the trace side of the board, all at ground potential, in order to improve ground return paths. U30 is shown as a 75365—the alternate part 3207 should *not* be used at this location. To make the changes:



1) Connect pin 5 of U28 to pin 9 of U29.
2) Connect pin 9 of U29 to pin 9 of U30.
3) Connect pin 9 of U30 to pin 9 of U31.
4) Connect pin 9 of U31 to pin 5 of U32.
5) Connect pin 8 of U28 to pin 1 of U32.

These changes don't require any change in the schematic shown on page VI-2, or in the Theory of Operation, Section V.

□

Vectored Interrupt Capability For Sol

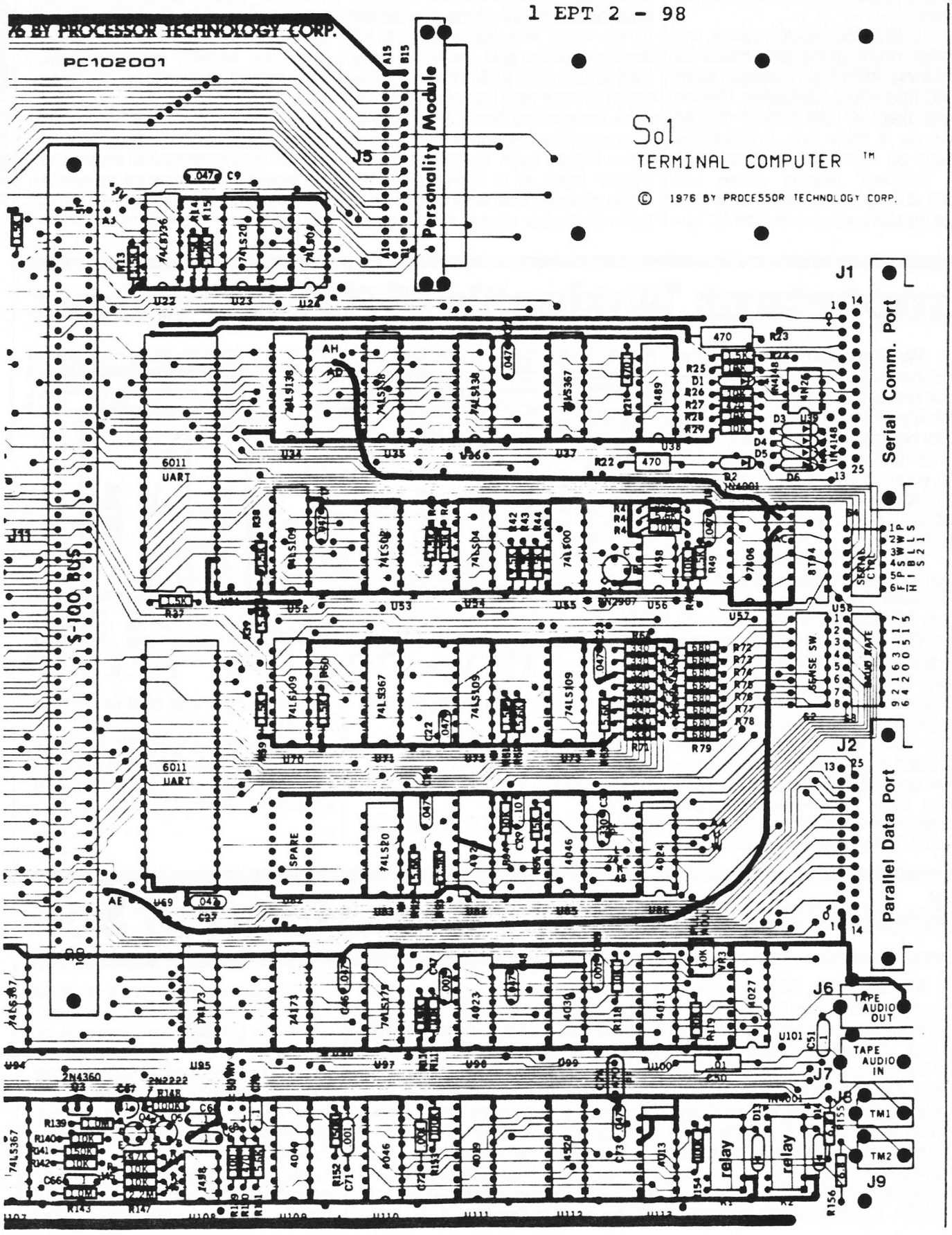
Although the 8080A microprocessor used in the Sol provides vectored interrupt capability, neither Sol nor Processor Technology's other S100 modules require use of this capability. However, Sol does provide a means for implementing vectored interrupt if the interrupt signal is made available by a circuit board inserted in the S100 bus. This board is connected to S100 bus pin 96, SINTA. If you want this board to generate interrupts, you'll need to add two jumpers, shown below, to the Sol P.C. These jumpers enable the SINTA signal to reach the memory decoder circuit; they can be added after assembling the Sol P.C., or even after the entire Sol has been assembled and tested.

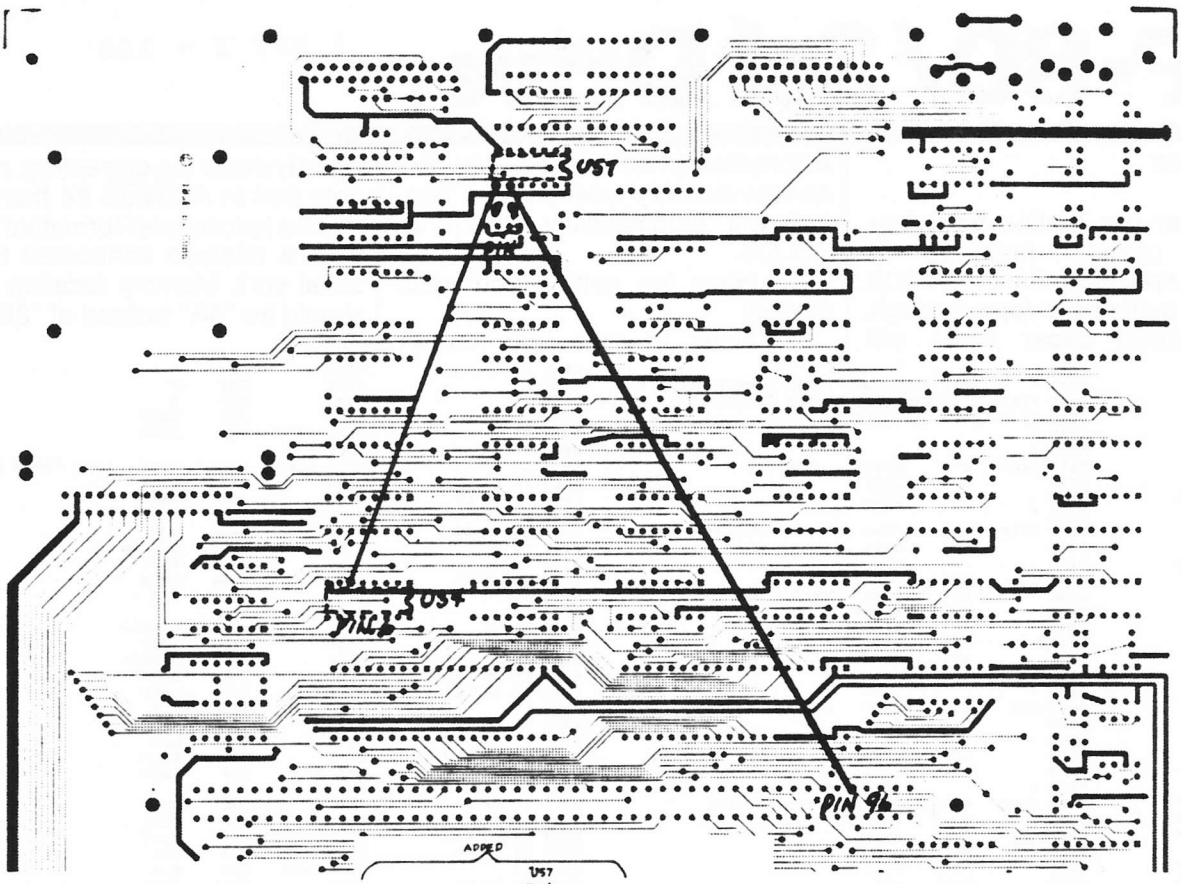
The jumpers should be made from #24 solid insulated wire (not provided). Their electrical effects can be seen in Drawing X-16. They can be left in place even if no S100 board generates interrupts, and S100 bus pin 96 may float with no interference.

The necessary modifications for both Revision D and Revision E level circuit boards are as follows:

Modification to Revision E board, Sol-PCB:

1) Strip .1" of insulation from one end of two 8" lengths of wire. Insert the stripped ends into pads AC and


1 EPT 2 - 98
76 BY PROCESSOR TECHNOLOGY CORP.
PC102001
Personality Module
J5
Sol
TERMINAL COMPUTER ™
© 1976 BY PROCESSOR TECHNOLOGY CORP.
J1
Serial Comm. Port
J2
Parallel Data Port
S-100 BUS
6011 UART
6011 UART
SPARE
J6
TAPE AUDIO OUT
TAPE AUDIO IN
J7
J8
J9
TM1
TM2
relay
relay
470
470
2N4360
2N2222
U22
U23
U24
U34
U35
U36
U37
U38
U52
U53
U54
U55
U56
U57
U58
U69
U70
U71
U73
U85
U94
U95
U97
U98
U99
U100
U101
U107
U108
U109
U110
U111
U112
U113



AB (near U58), from the component side of the board. Solder and check for solder bridges.

2) Dress the wires as shown in the accompanying diagram, trim to the correct length, and strip .1" from the loose ends. Insert the wire from pad AB into pad AD, and the wire from pad AC into pad AE. Again, solder and inspect for solder bridges.

3) Attach the long runs of wire to the board with silicone compound or tape.

Modification to Revision D Board, Sol-PCB:

1) Strip .1" insulation from each end of a 6.25" length of wire. Connect one end to pin 96 of J10 (S100 connector); the other end goes to pin 9 of U57.

2) Strip .1" of insulation from each end of a 4" wire. Connect pin 8 of U57 to pin 6 of U34.

3) Attach the long runs of wire to the board with silicone compound or tape.□

The Sol-20 And The OP-80A Paper Tape Reader

by Drew Rogge

Oliver Audio Engineering's excellent and inexpensive OP-80A Paper Tape reader is one of the most popular peripherals in use with the Sol-20, and Sol User's Manual does contain some information about interfacing it. However, there's an error in the manual, and hardware interfacing can be a tricky chore in any case. Especially when it comes to using the Sol's 8-bit parallel output port. (See Lee Felsenstein's article on the parallel port elsewhere in this issue.) So here's what to do about the manual error, and also some other useful tidbits.

Section VII (Operating Procedures) of the Sol Users' Manual discusses the OP-80A. The error is on page 31: pin 5 of J2 (Sol PDI connector) should be labeled IAK, *not* IAK. Therefore, the "input Acknowledge" jumper on the paper tape reader should be connected to ACK, not ACK.

The intensity of the light source is also an important consideration in bringing the OP-80A up. You can allow just enough brightness to operate the tape reader, yet avoid saturating the tape, if you start by placing the light source far enough above the reader so that the SP LED indicator remains off. Then lower the light source just far enough to activate the LED, and keep it there.

Pat Tuell, of the San Jose Byte Shop, has contributed a driver routine for using the OP-80A, which we've included along with this article.□

OP-80/Sol Driver

1 EPT 2 - 100

by Pat Tuell

Enter the routine from keyboard or cassette; hook up the reader as specified. Execute C900, and start pulling the tape through. Intel-formatted paper tapes will automatically be loaded into the correct memory locations and control will automatically return to SOLOS.

Have fun getting your data loaded!

Note: In case you haven't already made the correction, please note that in ACCESS #4 there is a routine to load Intel-formatted tapes from a teletype connected to the serial port. Memory location 1F0A should be "3A" instead of "3B." □

```
C900                    0010 *
C900                    0020 *     THIS WILL LOAD INTEL FORMAT PAPER TAPE
C900                    0030 *INTO SOL THRU OP80 READER CONNECTED TO SOL
C900                    0040 *PARALLEL DATA INTERFACE
C900                    0050 *
C900                    0060 *
C900 3E 10              0070 START  MVI   A,010H   STARTUP
C902 D3 FA              0080        OUT   0FAH
C904 CD 93 C9           0090        CALL  DAZL
C907                    0100 *
C907 DB FA              0110 PTIN   IN    0FAH    INPUT FROM OP80 ASC.
C909 E6 02              0120        ANI   002H
C90B C2 07 C9           0130        JNZ   PTIN
C90E 3E 00              0140        MVI   A,000H
C910 D3 FA              0150        OUT   0FAH    TOGGLE ACK
C912 3E 10              0160        MVI   A,010H
C914 D3 FA              0170        OUT   0FAH
C916 DB FD              0180        IN    0FDH
C918 E6 7F              0190        ANI   07FH
C91A C9                 0200        RET
C91B                    0210 *
C91B D6 30              0220 HEX    SUI   030H    CONVERT ASC TO HEX
C91D FE 0A              0230        CPI   00AH
C91F D8                 0240        RC
C920 D6 07              0250        SUI   007H
C922 C9                 0260        RET
C923                    0270 *
C923 CD 07 C9           0280 CHAR   CALL  PTIN    ASSEMBLE BYTE
C926 CD 1B C9           0290        CALL  HEX
C929 CD 35 C9           0300        CALL  ROLL
C92C 5F                 0310        MOV   E,A
C92D CD 07 C9           0320        CALL  PTIN
C930 CD 1B C9           0330        CALL  HEX
C933 83                 0340        ADD   E
C934 C9                 0350        RET
C935                    0360 *
C935 07                 0370 ROLL   RLC
C936 17                 0380        RAL
C937 17                 0390        RAL
C938 17                 0400        RAL
C939 C9                 0410        RET
C93A                    0420 *
C93A CD 07 C9           0430 CARD   CALL  PTIN    ASSM BYTE, DISPLAY
C93D F5                 0440        PUSH  006H
C93E 47                 0450        MOV   B,A
C93F CD 19 C0           0460        CALL  SOUT
C942 F1                 0470        POP   006H
C943 CD 1B C9           0480        CALL  HEX
C946 CD 35 C9           0490        CALL  ROLL
C949 5F                 0500        MOV   E,A
C94A CD 07 C9           0510        CALL  PTIN
C94D F5                 0520        PUSH  006H
C94E 47                 0530        MOV   B,A
C94F CD 19 C0           0540        CALL  SOUT
C952 F1                 0550        POP   006H
C953 CD 1B C9           0560        CALL  HEX
C956 83                 0570        ADD   E
C957 C9                 0580        RET
C958                    0590 *
C958 CD 23 C9           0600 LOOP   CALL  CHAR    INPUT DATA
C958 77                 
C95C 23                 
C95D 15                 
C95E C2 58 C9           
C961 C3 64 C9           
C964                    
C964 CD 07 C9           
C967 FE 3A              
C969 C2 64 C9           
C96C 06 0A              
C96E CD 19 C0           
C971 06 0D              
C973 CD 19 C0           
C976 CD 3A C9           
C979 57                 
C97A C8                 
C97B 06 0A              
C97D CD 19 C0           
C980 06 0D              
C982 CD 19 C0           
C985 CD 3A C9           
C988 67                 
C989 CD 3A C9           
C98C 6F                 
C98D CD 23 C9           
C990 C3 58 C9           
C993                    
C993 CD 07 C9           
C996 FE 3A              
C998 C2 9E C9           
C99B C3 6C C9           
C99E 47                 
C99F CD 19 C0           
C9A2 C3 93 C9           
C9A5                    
C9A5                    
C9A5                    
```

We're thinking that most users will only have to read the programs in once, and hoping that you'll become resigned to squinting your eyes as a one-time (per program), character-building experience, worth it for the rewards of acquiring the program.

Also check the reader feedback section of this issue for additional comments by the authors of these programs.

Again, you're encouraged to submit any and all programs you have. For purposes of accuracy and the suppression of typos, we encourage you to send either a cassette or photo-reproducible printout of the listing. If you send a cassette, we'll replace it with a blank one so you don't lose on the deal. Also try to provide as much support information as you can, like your uses for the program, where it originated, etc. It's a nice way to get your name in print!

Contributions to this issue:

Hammurabi (game, modified for BASIC/5) . . John Dowd
PAROL (payroll records) . . . . . John Dowd
COMM (weekly sales/payroll computations) . John Dowd
Inverse Video in BASIC/5 (just that!) . . . . . Tom Digate
BASIC VDM driver fix (to work with MITS 4.0 BASIC) . . . . Warren Saunders

Thanks, fellows!

BASIC VDM Driver Fix to Work with MITS 4.0 BASIC

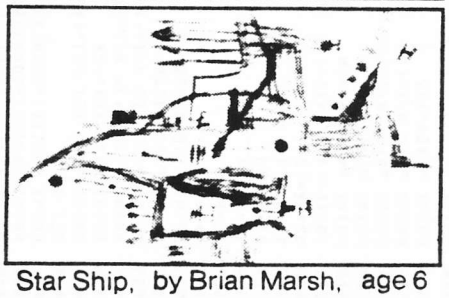
This useful emendation was contributed by ACCESS reader Warren Saunders of Oakton, Virginia. Add the following statements to BASIC and you'll be in the running!

```
0106 FOR Y=L TO L+416
0154 DATA 195,4000,5000,241,245,
  230,127,254,32,210,154
0156 DATA 1001,254,7,202,6,1000,
  254,13,202,47,1000
0158 DATA 254,1,202,47,1000,254,
  26,202,47,1000,241
0160 DATA 201,0,0,245,229,213,197
0233 DATA 254,127,194,47,1000,
  241,201
```

Note: BASIC VDM DRIVER with these changes remains compatible with previous versions of MITS BASIC.

PAROL and COMM

John Dowd told us he adapted this program to work with Sol BASIC/5 from a version found in *Some Common BASIC Programs*, a book by Lon Poole and Mary Borcher. No data is stored on cassette —only the programs. John uses them with a printer to keep records of beauty shop income (Lynne's Carrousel of Hair Fashions in Linwood, NJ), sales analysis, employee commission and payroll. Previous balances and sales totals are entered by keyboard from cash register tapes and payroll records. He says that using Sol has cut the time required for these jobs from 4 hours to 45 minutes!



Star Ship, by Brian Marsh, age 6

PAROL

```
LIST
 10  PRINT "PAYROLL PROGRAM"
 30  DATA 16,33,13,76,22,143,24,192,28,220,32,297,36,355
 40  DATA 15,61,18,135,22,223,25,278,28,355,32,432,36,500
 50  F1=.0625
 60  W2=19.2
 70  W1=14.4
 90  DIM F1(29)
100  FOR I=1 TO 28
110    READ F1(I)
120  NEXT I
130  PRINT CHR$(12)
140  PRINT "MARITAL STATUS (1=SINGLE, 2=MARRIED)"
150  INPUT S
160  PRINT "WITHHOLDING TAX EXEMPTIONS"
170  INPUT W
180  INPUT "REGULAR PAY      ";R
190  INPUT "COMMISSION       ";C
195  P=R+C
200  PRINT "YTD GROSS PAY"
210  INPUT Y
220  PRINT "YTD FICA"
230  INPUT Y2
240  PRINT "YTD FED. W/T"
250  INPUT Y1
260  PRINT "YTD N.J. W/T"
270  INPUT Y3
280  PRINT "YTD SUI"
290  INPUT Y4
300  PRINT "YTD NET"
310  INPUT Y5
311  O=Y-Y1-Y2-Y3-Y4
312  IF O=Y5 THEN 319
313  GOTO 130
319  P1=P1+P
```

```
1 EPT 2 - 101
```

```
320  PRINT "OPERATOR NUMBER"
330  INPUT O
340  S=P-W1+W
350  T1=0
360  FOR I=2 TO 7
370     X=2+I+14+(S-1)-1
380     IF G<=F1(X-1) THEN 460
390     IF G>F1(X+1) THEN 420
400     T1=T1+(G-F1(X-1))+F1(X-2)/100
410     GOTO 460
420     T1=T1+(F1(X+1)-F1(X-1))+F1(X-2)/100
430  NEXT I
440  T1=T1+(G-F1(X+1))+F1(X)/100
450  T1=T1+(G-F1(X+1))+F1(X)/100
460  T2=0
470  T2=INT((P+F1)+100+.5)/100
480  T3=0
481  T3=INT((P-0+W2)+2+2+.5)/100
482  IF T3>=0 THEN 489
483  T3=0
489  T4=0
490  T4=INT(P+.5)/100
495  T1=INT((T1+100)+.5)/100
500  N=P-T1-T2-T3-T4
510  PRINT "OPERATOR NUMBER";O
520  PRINT "GROSS PAY";P
530  PRINT "FICA";T2
540  PRINT "FED W/T";T1
550  PRINT "N.J. W/T";T3
560  PRINT "SUI";T4
570  PRINT "NET PAY";N
571  T6=T8+T1: T5=T5+T2: T7=T7+T3: T8=T8+T4: N1=N1+N
590  Y2=Y2+T2
590  Y1=Y1+T1
600  Y3=Y3+T3
610  Y4=Y4+T4
620  Y5=Y5+N
630  Y=Y+P
631  PRINT "YTD GROSS";Y
632  PRINT "YTD FICA";Y2
633  PRINT "YTD FED W/T";Y1
634  PRINT "YTD N.J. W/T";Y3
635  PRINT "YTD SUI";Y4
636  PRINT "YTD NET PAY";Y5
640  SET O=2
642  PRINT TAB(5);"REGULAR PAY";TAB(25);R
644  PRINT TAB(5);"COMMISSION";TAB(25);C
646  PRINT TAB(5);"GROSS PAY";TAB(25);P
648  PRINT TAB(5);"FICA";TAB(25);T2
650  PRINT TAB(5);"FEDERAL W/T";TAB(25);T1
652  PRINT TAB(5);"NEW JERSEY W/T";TAB(25);T3
654  PRINT TAB(5);"SUI";TAB(25);T4
656  PRINT : PRINT
658  PRINT TAB(5);"NET PAY";TAB(25);N
660  PRINT : PRINT : SET O=0
670  INPUT "AGAIN? (1=YES 0=NO)";A
680  IF A=1 THEN 130
690  PRINT "TOTAL GROSS PAY";P1
700  PRINT "TOTAL FICA";T5
710  PRINT "TOTAL FED W/T";T6
720  PRINT "TOTAL N.J. W/T";T7
730  PRINT "TOTAL SUI";T8
740  PRINT "TOTAL NET PAY";N1
750  R=INT((P1+2+F1)+(P1+.05)+1+T8+T7)
760  PRINT "RESERVE FOR WEEK";R
```

This section prints out a slip for each operator showing reg pay, commission, gross pay, deductions, net pay.

This section gives payroll totals for week and computes amount to be reserved for payment of payroll taxes, state & federal.

COMM

```
LIST
50   INPUT "YTD SALES";G
75   INPUT "YTD SALARIES";P5
100  INPUT "MONDAY TOTALS      ";W1;W2;W3;W4;W5;W6 — Cash register totals for day.
110  PRINT "ENTRY ERROR?  ( 1=YES  0=NO )"
120  INPUT E
130  IF E=1 THEN 100
150  W=W1+W2+W3+W4+W5+W6
160  PRINT "ZZ"
200  PRINT W1;W2;W3;W4;W5;W6;W
300  INPUT "TUESDAY TOTALS     ";T1;T2;T3;T4;T5;T6
310  PRINT "ENTRY ERROR?  (1=YES  0=NO )"
320  INPUT E
330  IF E=1 THEN 300
400  T=T1+T2+T3+T4+T5+T6
500  PRINT T1;T2;T3;T4;T5;T6;T
600  INPUT "WEDNESDAY TOTALS     ";W1;W2;W3;W4;W5;W6
610  PRINT "ENTRY ERROR?  (1=YES  0=NO )"
620  INPUT E
630  IF E=1 THEN 600
700  W=W1+W2+W3+W4+W5+W6
900  PRINT W1;W2;W3;W4;W5;W6;W
940  INPUT "THURSDAY TOTALS     ";H1;H2;H3;H4;H5;H6
910  PRINT "ENTRY ERROR?  (1=YES  0=NO )"
920  INPUT E
930  IF E=1 THEN 900
1000 H=H1+H2+H3+H4+H5+H6
1100 PRINT H1;H2;H3;H4;H5;H6;H
1200 INPUT "FRIDAY TOTALS     ";F1;F2;F3;F4;F5;F6
1210 PRINT "ENTRY ERROR?  (1=YES  0=NO )"
1220 INPUT E
1230 IF E=1 THEN 1200
1300 F=F1+F2+F3+F4+F5+F6
1400 PRINT F1;F2;F3;F4;F5;F6;F
1500 INPUT "SATURDAY TOTALS     ";S1;S2;S3;S4;S5;S6
1510 PRINT "ENTRY ERROR?  ( 1=YES  0=NO )"
1520 INPUT E
1530 IF E=1 THEN 1500
1600 S=S1+S2+S3+S4+S5+S6
1700 PRINT S1;S2;S3;S4;S5;S6;S
1800 INPUT "DAYS WORKED     ";D1;D2;D3;D4;D5;D6
1810 PRINT "ENTRY ERROR?  ( 1=YES  0=NO )"
1820 INPUT E
1830 IF E=1 THEN 1800
1900 PRINT "DAYS WORKED     ";D1;D2;D3;D4;D5;D6
2000 A1=M1+T1+W1+H1+F1+S1
2100 A2=M2+T2+W2+H2+F2+S2
2200 A3=M3+T3+W3+H3+F3+S3
2300 A4=M4+T4+W4+H4+F4+S4
2400 A5=M5+T5+W5+H5+F5+S5
2500 A6=M6+T6+W6+H6+F6+S6
2600 A=A1+A2+A3+A4+A5+A6 — Overall weekly total
2700 G=G+A — New YTD total
2800 C1=INT(((A1-(D1*44))*100)/4)/100   C1—C6—Operator's Commission
2810 IF C1>=0 THEN 2900
2820 C1=0
2900 C2=INT(((A2-(D2*44))*100)/4)/100
2910 IF C2>=0 THEN 3000
2920 C2=0
3000 C3=INT(((A3-(D3*44))*100)/4)/100
```

Each operator's weekly sales total computed

1 EPT 2 - 102

```
3010 IF C3>=0 THEN 3100
3020 C3=0
3100 C4=INT(((90-(D4*44))*100)/4)/100
3110 IF C4>=0 THEN 3200
3120 C4=0
3200 C5=INT(((35-(D5*44))*100)/4)/100
3210 IF C5>=0 THEN 3300
3220 C5=0
3300 C6=INT(((36-(D6*44))*100)/4)/100
3310 IF C6>=0 THEN 3400
3320 C6=0
3400 C=C1+C2+C3+C4+C5+C6
3500 P1=C1+(D1*22)
3600 P2=C2+(D2*20)
3700 P3=C3+(D3*22)
3800 P4=C4+(D4*22)
3900 P5=C5+(D5*22)
4000 P6=C6+(D6*22)
4100 P0=P2+P3+P4+P5+P6
4200 P=P0+(.12*P0)
4300 R=INT((P/3)*100)
4400 R1=INT((31/3)*100)
4500 R2=INT((32/3)*100)
4600 R3=INT((33/3)*100)
4700 R4=INT((34/3)*100)
4800 R5=INT((35/3)*100)
4900 R6=INT((36/3)*100)
5100 PRINT "ENTER  CUSTOMER COUNT (1-V1)"
5200 INPUT L1,L2,L3,L4,L5,L6
5300 PRINT L1;L2;L3;L4;L5;L6
5400 K1=K1+L1: K2=K2+L2: K3=K3+L3: K4=K4+L4: K5=K5+L5: K6=K6+L6
5500 K=K1+K2+K3+K4+K5+K6
5700 L1=INT((K1/K)*100): L2=INT((K2/K)*100): L3=INT((K3/K)*100)
5800 L4=INT((K4/K)*100): L5=INT((K5/K)*100): L6=INT((K6/K)*100)
5900 A=INT((S/K)*100)/100
5940 IF K1>0 THEN 6000
5950 A1=0: GOTO 6035
6000 A1=INT((31/K1)*100)/100
6005 IF K2>0 THEN 6010
6006 A2=0: GOTO 6015
6010 A2=INT((32/K2)*100)/100
6015 IF K3>0 THEN 6020
6016 A3=0: GOTO 6025
6020 A3=INT((33/K3)*100)/100
6025 IF K4>0 THEN 6030
6026 A4=0: GOTO 6035
6030 A4=INT((34/K4)*100)/100
6035 IF K5>0 THEN 6040
6036 A5=0: GOTO 6045
6040 A5=INT((35/K5)*100)/100
6045 IF K6>0 THEN 6050
6046 A6=0: GOTO 6100
6050 A6=INT((36/K6)*100)/100
6100 SET 0=2
6200 PRINT "   I        II        III        IV        V        VI          TOT"
6300 PRINT TAB(1);M1;TAB(8);M2;TAB(17);M3;TAB(28);M4;TAB(38);M5;TAB(48);M6;
6400 PRINT TAB(55);M
6500 PRINT TAB(1);T1;TAB(8);T2;TAB(17);T3;TAB(28);T4;TAB(38);T5;TAB(46);T6;
6600 PRINT TAB(55);T
6700 PRINT TAB(1);W1;TAB(8);W2;TAB(17);W3;TAB(28);W4;TAB(38);W5;TAB(45);W6;
6800 PRINT TAB(55);W
6900 PRINT TAB(1);H1;TAB(8);H2;TAB(17);H3;TAB(28);H4;TAB(38);H5;TAB(46);H5;
7000 PRINT TAB(55);H
7100 PRINT TAB(1);F1;TAB(8);F2;TAB(17);F3;TAB(28);F4;TAB(38);F5;TAB(46);F6;
7200 PRINT TAB(55);F
7300 PRINT TAB(1);S1;TAB(8);S2;TAB(17);S3;TAB(28);S4;TAB(39);S5;TAB(46);S6;
7400 PRINT TAB(55);S
7450 PRINT
7500 PRINT TAB(0);R1;TAB(8);R2;TAB(17);R3;TAB(28);R4;TAB(38);R5;TAB(46);R6;
7600 PRINT TAB(55);R
7700 PRINT "COMMISSIONS"
7800 PRINT TAB(0);C1;TAB(8);C2;TAB(17);C3;TAB(28);C4;TAB(38);C5;TAB(46);C6
7820 PRINT " INDIVIDUAL SALARIES INCLUDING COMMISSION."
7830 PRINT TAB(0);P1;TAB(8);P2;TAB(17);P3;TAB(28);P4;TAB(38);P5;TAB(46);P6
7840 PRINT "                    TOTAL PAY INCLUDING TAXES    PERCENT OF INCOME"
7900 PRINT TAB(38);P;: PRINT TAB(55);R;"%"
7950 PRINT "PERCENTAGE OF TOTAL BUSINESS DONE BY EACH OPERATOR."
8000 PRINT TAB(0);R1;TAB(8);R2;TAB(17);R3;TAB(28);R4;TAB(38);R5;TAB(46);R6
8100 PRINT "   I        II        III        IV        V        VI "
8200 PRINT "PERCENTAGE OF WEEK'S CUSTOMERS SERVED BY EACH OPERATOR."
8300 PRINT TAB(0);L1;TAB(8);L2;TAB(17);L3;TAB(28);L4;TAB(39);L5;TAB(46);L6
8400 PRINT "OPERATOR'S AVERAGE SALE PER CUSTOMER"
8500 PRINT TAB(0);A1;TAB(8);A2;TAB(17);A3;TAB(28);A4;TAB(38);A5;TAB(46);A6
8620 PRINT "SHOP'S  AVERAGE SALE PER CUSTOMER"
8700 PRINT TAB(8);A
8710 PRINT "OPERATOR'S % OF SALARY TO SALES"
8715 IF A1>0 THEN 8720
8716 V1=0: GOTO 8721
8720 V1=INT(((P1+(.12*P1))/31)*100):
8721 IF A2>0 THEN 8725
8722 V2=0: GOTO 8730
8725 V2=INT(((P2+(.12*P2))/A2)*100)
8726 IF A3>0 THEN 8730
8727 V3=0: GOTO 8731
8730 V3=INT(((P3+(.12*P3))/33)*100):
8731 IF A4>0 THEN 8735
8732 V4=0: GOTO 8736
8735 V4=INT(((P4+(.12*P4))/34)*100)
8736 IF A5>0 THEN 8740
8737 V5=0: GOTO 8741
8740 V5=INT(((P5+(.12*P5))/35)*100):
8741 IF A6>0 THEN 8745
8742 V6=0: GOTO 8750
8745 V6=INT(((P6+(.12*P6))/36)*100)
8750 PRINT TAB(0);V1;TAB(8);V2;TAB(17);V3;TAB(28);V4;TAB(38);V5;TAB(46);V6
8800 PRINT : PRINT : PRINT
9100 P9=P3+P
9200 R0=INT(((P9/G)*100)+.5)
9300 PRINT "YTD SALES";G
9400 PRINT "YTD SALARIES";P9
9500 PRINT "YTD PERCENT OF SALARIES/SALES";R0;"%"
9550 SET 0=0
9600 END
```

Each operator's pay for week

Total pay for week excluding operator #1—my wife

Total pay & employer taxes (approximate)

% of total sales done by each operator

Enter # of customers served by each operator

Total customers

% of customers served by each operator.

Average $ val. of sale per customer

Each operator's average $ val. of sale per customer

Output to expandor printer by parallel port

Compute each operator's % of salary paid to sales for week.

% salaries/sales YTD

1 EPT 2 - 103

Hammurabi

1 EPT 2 - 104

```
LIST
 80   PRINT "TRY YOUR HAND AT GOVERNING ANCIENT SUMERIA"
 85  "PRINT "SUCCESSFULLY FOR A 10 YEAR TERM OF OFFICE.": PRINT
 90   LET D1=0: LET P1=0
100  LET Z=0: LET P=95: LET S=2800: LET H=3000: LET E=H-S
110  LET Y=3: LET A=H/Y: LET I=5: LET D=1
210  LET D=0
215  PRINT : PRINT : PRINT "HAMURABI:  I BEG TO REPORT TO YOU,": LET Z=Z+1
217  PRINT "IN YEAR";Z;",";D;" PEOPLE STARVED,";I;" CAME TO THE CITY."
219  LET P=P+I
227  IF 0>0 THEN 230
228  LET P=INT(P/2)
229  PRINT "A HORRIBLE PLAGUE STRUCK!  HALF THE PEOPLE DIED."
!00  PRINT "POPULATION IS NOW";P
232  PRINT "THE CITY NOW OWNS";A;" ACRES."
235  PRINT "YOU HARVESTED";Y;" BUSHELS PER ACRE."
250  PRINT "RATS ATE";E;" BUSHELS."
260  PRINT "YOU NOW HAVE";S;" BUSHELS IN STORE.": PRINT
270  IF Z=11 THEN 860
310  LET C=INT(10*RND(0)): LET Y=C+17
312  PRINT "LAND IS TRADING AT";Y;" BUSHELS PER ACRE."
320  PRINT "HOW MANY ACRES DO YOU WISH TO BUY?";
321  INPUT Q: IF Q<0 THEN 850
322  IF Y*Q<=S THEN 330
323  GOSUB 710
324  GOTO 320
330  IF Q=0 THEN 340
331  LET A=A+Q: LET S=S-Y*Q: LET C=0
334  GOTO 400
340  PRINT "HOW MANY ACRES DO YOU WISH TO SELL?";
341  INPUT Q: IF Q<0 THEN 850
342  IF Q<A THEN 350
343  GOSUB 720
344  GOTO 340
350  LET A=A-Q: LET S=S+Y*Q: LET C=0
400  PRINT
410  PRINT "HOW MANY BUSHELS DO YOU WISH TO FEED YOUR PEOPLE?";
411  INPUT Q
412  IF Q<0 THEN 850
420  IF Q<=S THEN 430
421  GOSUB 710
422  GOTO 410
430  LET S=S-Q: LET C=1: PRINT
440  PRINT "HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED?";
441  INPUT Q: IF Q=0 THEN 511
442  IF Q<0 THEN 850
445  IF Q<=A THEN 450
446  GOSUB 720
447  GOTO 440
450  IF INT(Q/2)<S THEN 455
452  GOSUB 710
453  GOTO 440
455  IF Q<10*P THEN 510
460  PRINT "BUT YOU HAVE ONLY";P;" PEOPLE TO TEND THE FIELDS.  NOW THEN,"
470  GOTO 440
510  LET S=S-INT(Q/2)
511  GOSUB 880
515  LET Y=C: LET H=Q*Y: LET E=
521  GOSUB 880

522  IF INT(C/2)<>C/2 THEN 530
525  LET E=INT(S/C)
530  LET S=S-E+H
531  GOSUB 880
533  LET I=INT(C*(Z+A+S)/P/100+1)
540  LET C=INT(C/20)
542  LET D=INT(10*(2*RND(0)-.3))
550  IF P<0 THEN 210
552  LET D=P-Q: IF D>.45*P THEN 560
553  LET P1=((Z-1)*P1+D)*100/P/Z
555  LET P=Q: LET D1=D1+D: GOTO 216
560  PRINT : PRINT "YOU STARVED";D;" PEOPLE IN ONE YEAR!!!"
565  PRINT "DUE TO YOUR EXTREME MISMANAGEMENT YOU HAVE NOT ONLY"
566  PRINT "BEEN IMPEACHED AND THROWN OUT OF OFFICE, BUT YOU HAVE"
567  PRINT "ALSO BEEN DECLARED 'NATIONAL FINK'!!": GOTO 990
710  PRINT "HAMURABI, THINK AGAIN.  YOU HAVE ONLY"
711  PRINT S;" BUSHELS OF GRAIN.  NOW THEN,"
712  RETURN
720  PRINT "HAMURABI:  THINK AGAIN.  YOU OWN ONLY";A;" ACRES.  NOW THEN,"
730  RETURN
860  LET C=INT(RND(0)*5)+1
861  RETURN
850  PRINT : PRINT "HAMURABI:  I CANNOT DO WHAT YOU WISH."
855  PRINT "GET YOURSELF ANOTHER FLUNKY!!!!"
857  GOTO 990
860  PRINT "IN YOUR 10 YEAR TERM OF OFFICE,";P1;" PERCENT OF THE"
862  PRINT "POPULATION STARVED PER YEAR ON AVERAGE, I.E., A TOTAL OF"
865  PRINT D1;" PEOPLE DIED!!!": LET L=A/P
870  PRINT "YOU STARTED WITH 10 ACRES PER PERSON AND ENDED WITH"
875  PRINT L;" ACRES PER PERSON.": PRINT
880  IF P1>33 THEN 565
885  IF L<7 THEN 565
890  IF P1>10 THEN 940
892  IF L<9 THEN 940
895  IF P1>3 THEN 960
896  IF L<10 THEN 950
900  PRINT "A FANTASTIC PERFORMANCE!!! CHARLEMAGNE, DISRAELI, AND"
905  PRINT "JEFFERSON COMBINED COULD NOT HAVE DONE BETTER!": GOTO 930
910  PRINT "YOUR HEAVY-HANDED PERFORMANCE SMACKS OF NERO AND IVAN IV."
945  PRINT "THE PEOPLE (REMAINING) FIND YOU AN UNPLEASANT RULER, AND,"
950  PRINT "FRANKLY, HATE YOUR GUTS!": GOTO 990
960  PRINT "YOUR PERFORMANCE COULD HAVE BEEN SOMEWHAT BETTER, BUT"
965  PRINT "REALLY WASN'T BAD AT ALL.";INT(P*.8*RND(0));" PEOPLE WOULD"
970  PRINT "DEARLY LIKE TO SEE YOU ASSASSINATED BUT WE ALL HAVE OUR"
975  PRINT "TRIVIAL PROBLEMS."
980  GOTO 995
985  PRINT "SO LONG FOR NOW.": PRINT
999  END
1000 PRINT ZFZ

RUN


TRY YOUR HAND AT GOVERNING ANCIENT SUMERIA
SUCCESSFULLY FOR A 10 YEAR TERM OF OFFICE.


HAMURABI:  I BEG TO REPORT TO YOU,
IN YEAR 1, 0 PEOPLE STARVED, 5 CAME TO THE CITY.
POPULATION IS NOW 100      THE CITY NOW OWNS 1000 ACRES.
YOU HARVESTED 3 BUSHELS PER ACRE.
RATS ATE 200 BUSHELS.
YOU NOW HAVE 2800 BUSHELS IN STORE.

LAND IS TRADING AT 21 BUSHELS PER ACRE.
                    |

HOW MANY ACRES DO YOU WISH TO BUY?
?10

HOW MANY BUSHELS DO YOU WISH TO FEED YOUR PEOPLE?
?2000

HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED?
?  999


HAMURABI:  I BEG TO REPORT TO YOU,
IN YEAR 2, 0 PEOPLE STARVED, 3 CAME TO THE CITY.
A HORRIBLE PLAGUE STRUCK!  HALF THE PEOPLE DIED.
POPULATION IS NOW 51       THE CITY NOW OWNS 1010 ACRES.
YOU HARVESTED 1 BUSHELS PER ACRE.
RATS ATE 45 BUSHELS.
YOU NOW HAVE 1045 BUSHELS IN STORE.

LAND IS TRADING AT 23 BUSHELS PER ACRE.
HOW MANY ACRES DO YOU WISH TO BUY?
?0
HOW MANY ACRES DO YOU WISH TO SELL?
?10

HOW MANY BUSHELS DO YOU WISH TO FEED YOUR PEOPLE?
?500

HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED?
?1010
HAMURABI:  THINK AGAIN.  YOU OWN ONLY 1000 ACRES.  NOW THEN,
HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED?
?1000
BUT YOU HAVE ONLY 51 PEOPLE TO TEND THE FIELDS.  NOW THEN,
HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED?
?509

YOU STARVED 26 PEOPLE IN ONE YEAR!!!
DUE TO YOUR EXTREME MISMANAGEMENT YOU HAVE NOT ONLY
BEEN IMPEACHED AND THROWN OUT OF OFFICE, BUT YOU HAVE
ALSO BEEN DECLARED 'NATIONAL FINK'!!!
SO LONG FOR NOW.


SOL  BASICS

READY
```

Inverse Video in BASIC 5

```
10    REM * SOL VDM DEMONSTRATION PROGRAM
20    REM * THIS PROGRAM WILL DEMONSTRATE
30    REM * HOW TO USE THE ESCAPE SEQUENCES IN
40    REM * THE SOLOS VDM DRIVER.
50    REM * WRITTEN BY TOM DIGATE
55    REM * DECEMBER, 1977
60    REM *
70    REM * INITIALIZE THE MESSAGE DATA TABLE
80    DATA 201,238,246,229,242,244,229,228
90    DATA 160,214,233,228,229,239
91    REM * INITIALIZE VDMOUT DRIVER ADDRESS AND
92    REM * SHIFT VARIABLE FOR ESCAPE CODES.
93    V0=49177: V1=256: V2=27: V3=5
95    DIM J(30)
100   REM * CLEAR SCREEN
110   PRINT "&K": SET S=0
120   PRINT "This is a demonstration of the capabilities of"
130   PRINT "Sol Basic5 output display formatting."
135   PRINT : PRINT
140   FOR I=1 TO 14: READ J(I): NEXT
150   REM * PRINT INVERTED MESSAGE
160   PRINT "The Sol can print in upper and lower case and"
170   PRINT "can even print ";
180   FOR I=1 TO 14
190      C=J(I): GOSUB 65010
200   NEXT I
205   PRINT '."
210   REM * GET CURSOR BACK AT COLUMN 1
220   PRINT "&[&A&A"
230   PRINT : PRINT : PRINT
240   REM * DEMONSTRATE SPEED CONTROL
250   PRINT "The Sol can even change display";
260   SET S=10: PRINT " speeds";: SET S=75: PRINT " dynamically!"
270   SET S=0
280   PRINT : PRINT "Pretty neat, huh!"
285   SET S=15
290   REM * FINISH UP DEMONSTRATION
300   REM * BY SHOWING SPECIAL CHARACTER SET
301   PRINT : PRINT : PRINT "The Sol also has a special set of limited"
302   PRINT "graphics characters. They are shown below in both"
303   PRINT "regular and inverted video. This gives a Basic programmer"
304   PRINT "very good flexibility when designing display formats."
305   PRINT
306   REM * PRINT SPECIAL CHARACTER SET FOR VDM
307   SET S=0
310   FOR I=0 TO 31
320      C=I
330      GOSUB 65010
335      PRINT " ";
340   NEXT I
350   PRINT : PRINT
355   REM * PRINT SPECIAL CHARACTER SET FOR VDM IN INVERTED VIDEO
360   FOR I=0 TO 31
370      C=I+128: GOSUB 65010: C=160: GOSUB 65010
400   NEXT
410   PRINT
420   SET S=18
430   PRINT "This concludes my demonstration of special display"
440   PRINT "features available in BASIC5. Happy computing!!!"
450   PRINT : PRINT : PRINT : SET S=0
460   END


64998 REM ************************************************
64999 REM *
65000 REM * OUTPUT INVERSE VIDEO SUBROUTINE USING SOLOS VDM
65001 REM * PASS DECIMAL VALUE OF CHARACTER
65003 REM * IN VARIABLE 'C'. NOTE- 'B' IS DESTROYED.
65004 REM *
65005 REM ************************************************
65006 REM *
65010 B=ARG(V2*V1): B=CALL(V0): B=ARG(V3*V1): B=CALL(V0)
65020 B=ARG(C*V1): B=CALL(V0): RETURN
65025 REM *
65030 REM * THE ROUTINE ABOVE USES THE FOLLOWING VARIABLES
65040 REM * V0   =ADDRESS OF VDM DRIVER IN SOLOS
65050 REM * V1   =256 TO SHIFT CHARACTER INTO B REGISTER
65060 REM * V2   =27 FOR ESCAPE CHARACTER
65070 REM * V3   =05 FOR ESCAPE CODE TO SEND CHAR. TO SCREEN
65080 REM *                WITHOUT CHANGING HIGH ORDER BIT.
```

PROCESSOR TECHNOLOGY
ACCESS.
7100 Johnson Industrial Drive
Pleasanton, CA 94566

Bulk Rate
U. S. Postage
PAID
Permit No. 54
Pleasanton, CA 94566

Chapter 3. Sol Users Manual.
Section 1. Exploring the Sol System..... 8
Section 2. At the Controls.....18
Section 3. Introduction to Software.....22
Section 4. Loading Programs from Cassette.....28
Section 5. At the Keyboard (Key Functions).....36
Section 6. Unpacking and Installation.....45
Section 7. TLC for Your Sol (Light Maintenance).....54
Appendix 1. Products Available from Processor Tech.....59
Appendix 2. Cassette Recorder Supplement.....65
Selecting a Cassette Recorder.....65
Interconnect Requirements for Two Recorders.....66
Appendix 3. Internal Controls (Switch Settings).....68
Appendix 4. Keyboard Tables.....72
Appendix 5. Sol Specifications.....76

Sol® Terminal Computer User's Manual

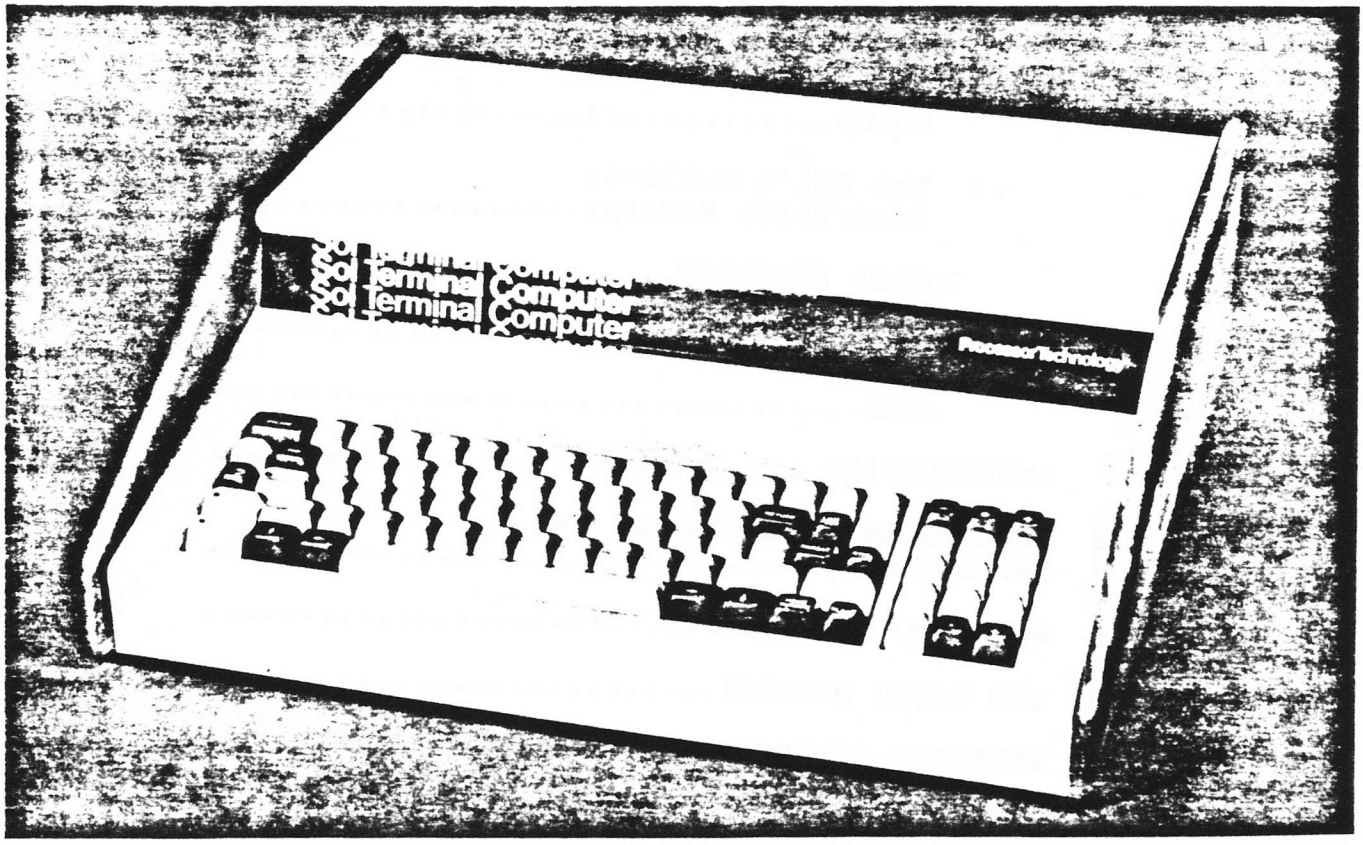
Processor Technology Corporation

7100 Johnson Drive
Pleasanton, CA 94566
Telephone (415) 829-2600

Copyright © 1978, by Processor Technology Corporation
First Edition, First Printing, December 1978
Manual Part No. 730021
All rights reserved.

IMPORTANT NOTICE

Sol and SolPrinter are registered trademarks, and WordWizard and Helios are trademarks of Processor Technology Corporation.



The Sol Terminal Computer

TABLE OF CONTENTS

SECTION		PAGE
1	EXPLORING THE Sol SYSTEM.....	1-1
1.1	THE SCOPE OF THIS MANUAL.....	1-1
1.2	WHAT'S A COMPUTER?.....	1-2
1.2.1	Clockwork 8080.....	1-2
1.3	A COMPUTER WITH A BUILT-IN TERMINAL.....	1-3
1.3.1	"Play a Little Sol Music".....	1-4
1.3.2	"Sol-a-vision".....	1-5
1.4	"INTER-Sol-ar" SYSTEM COMMUNICATION.....	1-5
1.4.1	The Coming of the Sol S-100 Backplane.....	1-7
5	MEMORIES ARE MADE OF THIS.....	1-7
1.5.1	Psychoanalyzing the Sol Personality Module.....	1-8
	SOLOS.....	1-8
.2	The Sol's Built-in Read/Write Memory.....	1-9
	CORDED SOFTWARE.....	1-9
	ANDING "Sol-ar" SYSTEM.....	1-10
	KOLS.....	2-1
	INTRODUCTION.....	2-1
2.2	REQUIREMENTS NEEDED FOR THE EXPEDITION.....	2-1
2.3	POWER TO THE Sol.....	2-2
2.4	THE VIDEO DISPLAY.....	2-3
2.5	KEYBOARD CONTROLS AND KEYBOARD RESTART.....	2-3

SECTION PAGE

3 INTRODUCTION TO SOFTWARE..... 3-1

3.1 WHAT IS SOFTWARE?..... 3-1

3.2 COMPUTER LANGUAGE AND LANGUAGE TRANSLATORS.. 3-2

3.3 SOLOS..... 3-3

3.3.1 Command Mode..... 3-4
3.3.2 "Try It"..... 3-4

4 LOADING PROGRAMS FROM CASSETTE..... 4-1

4.1 INTRODUCTION..... 4-1

4.2 AUDIO CASSETTE RECORDERS (Preliminary Information)..... 4-1

4.2.1 Operating A Cassette Recorder..... 4-2
4.2.2 Cassette Recorder Operating Tips..... 4-2

4.2.3 Notes on Writing to Cassette..... 4-2
4.2.4 Notes on Reading from Cassette..... 4-3

4.3 GETTING INTO BASIC (Procedure for Using A Cassette Recorder)... 4-3

4.4 WRITING PROGRAMS AND SAVING THEM TO CASSETTE..... 4-5

4.4.1 Writing a Program..... 4-5
4.4.2 Saving a Program to Cassette..... 4-7
4.4.3 Recalling a Program from Tape..... 4-7

4.5 THE Sol ALSO RISES..... 4-8

5 AT THE KEYBOARD..... 5-1

5.1 THE KEYBOARD, GENERAL INFORMATION..... 5-1

5.1.1 Key Functions are Programmable..... 5-1
5.1.2 Operating Features and Keyboard Indicators..... 5-2
5.1.3 Using Table A4-1..... 5-2

5.2 CHARACTER KEYS..... 5-3

5.3 FUNCTION KEYS..... 5-3

5.3.1 Cursor Control Keys..... 5-4
5.3.2 Individual Descriptions of Function Keys..... 5-5

TABLE OF CONTENTS (Continued)

SECTION			PAGE
	5.4	THE CONTROL KEYS.....	5-8
		5.4.1 General Information.....	5-8
		5.4.2 Transmission and Display of Control Keys.....	5-8
		5.4.3 Familiarization with the Control Symbols.....	5-9
	5.5	ARITHMETIC PAD KEYS.....	5-8
6		UNPACKING AND INSTALLATION.....	6-1
	6.1	RECEIVING INSPECTION.....	6-1
	6.2	INSTALLATION OF S-100 MODULES.....	6-2
	6.3	BUTTONING UP THE Sol AND INSTALLING THE FUSE.....	6-4
	6.4	CONNECTING THE VIDEO.....	6-6
	6.5	THE CASSETTE CONNECTION.....	6-6
7		TLC FOR YOUR Sol.....	7-1
	7.1	SERVICE AND ORDERING PARTS.....	7-1
	7.2	CARE AND EXTERIOR CLEANING OF YOUR Sol.....	7-1
	7.3	TROUBLE-SHOOTING the Sol INSTALLATION.....	7-2
		7.3.1 Checking and Installing the Fuse.....	7-2
		7.3.2 Removing The Covers Prior to Checkout of the Interior.....	7-3
		7.3.3 Checkout of Cables, Connectors, Sol-PC and Personality Module.....	7-3
		7.3.4 Buttoning the Sol Back Up.....	7-4
	7.4	TROUBLE-SHOOTING THE VIDEO.....	7-4
	7.5	TROUBLE-SHOOTING CASSETTE OPERATIONS.....	7-4
		7.5.1 Problems Encountered When Loading....	7-4
		7.5.2 Possible Faulty Tape?.....	7-5

SECTION PAGE

APPENDICES

1 PRODUCTS AVAILABLE from Processor Technology

2 CASSETTE RECORDER SUPPLEMENT

3 INTERNAL CONTROLS

4 KEYBOARD TABLES

5 Sol SPECIFICATIONS

TABLES

6-1 Sol System Component Lists..... 6-8

A3-1 Sol Internal Controls in Numerical Order..... A3-4

A4-1 Sol Keyboard Character and Control Keys..... A4-1

A4-2 Sol Keyboard Function Keys..... A4-3

A4-3 Names of Graphic Characters on Keytops..... A4-4

FIGURES

1-1 Sol-PC Simplified Block Diagram..... 1-6

2-1 Sol Rear Panel..... 2-2

5-1 Sol Keyboard..... 5-4

6-1 Installing An S-100 Module..... 6-3

6-2 Sol System Interconnections..... 6-5

A1-1 Sol System IV-A..... A1-2

SECTION 1

EXPLORING THE Sol SYSTEM

1.1 THE SCOPE OF THIS MANUAL

If you have obtained your Sol still in the shipping carton(s) please read Section 6, "Unpacking and Installation" before unpacking any further and before you start using your Sol. If you find yourself so anxious to unpack your Sol that you intend to skip the introduction, you may go directly to Section 6, but please read Section 1.1, "The Scope of this Manual".

This manual is a light operating guide and reference handbook for all Sol users. Just as the Sol is a simple yet sophisticated small computer that you can use for elementary applications such as playing video games, learning programmed lessons, as a desk-top computer system to write programs, as the host computer in a somewhat more complex business system, or as a terminal in a network of computer systems, so also, this manual can serve as a novice's learning book or as the more experienced user's introduction to the full capability of the Sol Terminal Computer.

The purpose of the first four sections is to get you quickly "up and running" with your Sol Terminal Computer. The sections and the information in them are presented in a sequence of steps especially designed for the purpose. Please follow the instructions up to the end of Section 4; then we turn you loose.

The rest of this section 1 presents a thumbnail sketch of the typical small computer system, with the spotlight on the Sol Terminal Computer System.

Section 2 is a briefing on the Sol user controls and other preliminary information on the "hardware" in your Sol system. Section 3 is an introduction to software. The object of having a computer, after all, is to run programs; therefore this manual will introduce you to the software manuals you have received with this manual. They are: SOLOS/CUTER User's Manual, Extended Cassette BASIC User's Manual (or Extended Disk BASIC User's Manual if you have a Helios).

Once you have progressed to the end of Section 4, you will have learned to load Extended Cassette BASIC from cassette and performed a few "basic" operations, including some simple programming.

After that, you can refer to the remainder of this manual for more detailed information that you may need from time to time as you gain more experience using your Sol. For example, Section 5 is all about the Sol keyboard. Section 7 contains maintenance and trouble-shooting tips for the qualified experts who are responsible for your system.

The appendices contain brief descriptions of hardware and software products to enhance your Sol. The appendices also contain additional technical information about the Sol.

If you have need for more technical information than this manual provides, for example, theory of operation and schematic drawings, you may order a more technical manual through your dealer, who uses such a manual to help keep your Sol in top condition.

You are cordially invited to participate in the future improvement of this book by sending us your comments and suggestions. A postpaid form is provided at the back of the manual.

1.2 WHAT'S A COMPUTER?

A computer is a manmade tool that extends the powers of the human mind. It does its work by manipulating vast numbers of symbols at tremendous speeds. In a digital computer like the Sol, the smallest, most basic symbol with which the computer works is called a "bit." A bit is a unit of information that answers the question, "Yes or No." In other words, a bit is either on or off. These two states are represented in mathematical notation as "1" for on, "Ø" for off. The two states of a bit are represented in the computer by two different electrical voltage levels. The computer has a myriad of minuscule electronic circuits that can alternate from one of these voltage levels to the other. By arranging these circuits into particular formations, information, also represented in bits, can be made to flow through the computer in a controlled fashion to answer questions and make statements of a mathematical and logical nature.

There are seemingly countless "logic circuits" in even a small computer, but they are always arranged in three basic functional groups, much as the cells of our brain are organized into parts of the brain. The first group functions as the control center, comparable to our thinking center. It is called the Central Processing Unit (CPU). The CPU directs the operations of the other two functional units, the memory an the input/output (I/O), which can be compared to our own memory and to the input/output functions of our senses and organs of communication. This similarity of the computer to the human nervous system is the basis for the study of cybernetics.

1.2.1 Clockwork 8080

It is in the CPU that the user's instructions to the computer are processed. The Sol's CPU is a single large scale integrated circuit named the 8080, which has become a standard in small computer systems and microprocessor-based systems. (An integrated circuit is single "chip" of semiconductor material upon which thousands of interconnected electronic circuits are fabricated by building up alternating layers of variously conducting materials in microscopic patterns.)

The 8080 microprocessor is ensconced on a large rectangular circuit board at the base of the Sol. This circuit board is called the Sol-PC. Making up the other two functional blocks of a computer,

Memory and the Input/Output (I/O), are about a hundred other integrated circuits on the Sol-PC. Memory and I/O support the control functions of the CPU.

The Sol-PC belongs to a category of computers called "Single Board Computers." The Sol-PC is available as a separate product from Processor Technology.

A crystal-controlled clock synchronizes the 8080's operations based on a rate of 2 MHz (two million cycles per second). The Sol attains an overall speed of between 100,000 and 500,000 instructions per second, depending on the type of instruction. The instruction referred to here is a machine language instruction. The "machine" is the microprocessor, whose prefix "micro" distinguishes the species "micro-computer." The 8080 microprocessor speaks a machine language made up of 8-bit words. An 8-bit word is called a "byte."

The second functional grouping of circuits within a computer is the memory. The CPU stores data in the memory prior to and after the data has been processed. Also stored in the memory are programs" which consist of lists of machine language instructions that are executed by the CPU section. In the Sol, the bulk of the memory function is performed by S-100 plug-in modules such as the 32KRA-1. The Sol's memory will be explained further in a succeeding paragraph.

The third and final functional grouping within a computer is the input/output circuits. I/O circuits coordinate the movement of data in and out of the computer. They create a circuit path between the CPU and external devices which perform other data processing or communicating. The circuits in the Sol which form a path between the CPU and a cassette recorder are part of the I/O function. (The Sol I/O interfaces will be discussed further in 1.4). With a CPU to direct the movement of data, a memory in which to store the data, and an input/output (I/O) interface to transfer data in and out of the computer, your Sol is ready to communicate with the outside world of other devices, other computers and with you, the user.

1.3 A COMPUTER WITH A BUILT-IN TERMINAL

Perhaps the most distinctive feature of a Sol is that it is a small computer with built-in "terminal." That is why the Sol is called a "Terminal Computer." A terminal is a device that enables the user to communicate with the computer. It usually consists of a keyboard and video display. The keyboard is used to enter data and commands; the video display is used to monitor the computer operations and display the data and commands being entered and received.

Technically speaking, the terminal is not part of a computer as such; rather it an input/output device and more specifically, a control console for human interaction with the computer. Other input/output devices, such as cassette recorders, collectively are called peripherals. Peripherals are equipment which aid the computer by loading, storing or displaying data, usually under the control of a computer.

An "intelligent" terminal is one that can perform some processing of the data in addition to entering and receiving it. Terminals and computers are often connected by telephone or other telecommunications lines to "time-share" or to form a "network." The Sol can be programmed to act as an intelligent terminal or as a free-standing computer. Most computers need a separate terminal for the user to communicate with it. A terminal consisting of the keyboard, video display and a standard communications interface are already built into every Sol. The Sol is its own terminal.

Despite its sophistication, the Sol is one of the easiest computers to operate. At the beginning you will be using the Sol as a "stand alone" computer, and you will use the keyboard to communicate with your Sol system as you enter and control programs and data.

1.3.1 "Play a Little Sol Music"

Like the keyboard on a piano, the keyboard is an important part of a terminal because it is the interface through which the user communicates with the computer. The Sol's more than ample keyboard compares with keyboards supplied with larger minicomputer systems. It has a 70-key main keyboard with color-coded keys that generate all 128 ASCII code characters. (ASCII is a standard code used by most computer manufacturers.) The ASCII character set includes the standard typewriter upper and lower case characters, plus a set of control characters that are activated when the CTRL (control) key is pressed at the same time as a character key. In addition to the ASCII keys, which are programmable, there are several "hard-wired" (permanent wired) special function keys. These function keys are handy for console operation of the computer, as well as for programming and word processing.

Conveniently located to the right of the main keyboard is a separate 15-key calculator pad provided so that the user does not have to stretch for the numerical keys at the top of the keyboard, when

making lengthy numeric entries. Both keyboards are designed for ease of operation and reliability. There is a built-in memory and scanning circuit that prevents simultaneous key activation and allows you to press any number of keys as fast a you can without losing a character. There are two keyboard status indicator lights to tell you whether the keyboard is set for upper case characters or shifted upper case. A third light indicates whether a Sol being used as a terminal is set for local operation or for transmission of the keyboard entries.

1.3.2 "Sol-a-vision"

The Sol Terminal Computer is self-contained insofar as all it needs to run is to be plugged in and turned on, but we humans need to see what we are entering on its keyboard what it is communicating to us. We need to monitor its operations and its communications with other devices. It needs a "display" for the video output of the Sol. The display is provided for in a separate I/O device, a "video monitor," which is a black and white TV adapted for this use or a similar but specially designed monitor.

The Sol video display generates 16 lines of text, each line 64 characters long. The display can be reversed by the user to show white characters out of black, or black on white.

1.4 "INTER-Sol-ar" SYSTEM COMMUNICATION

(Refer to Fig. 1-1, Sol-PC Simplified Block Diagram.)

"Software" is a broad term that refers to programs that are run on a computer. These programs are lists of instructions that tell the computer how the "data" are to be manipulated. Data is the information upon which the CPU operates. Data and instructions can be intermixed throughout a program; the program keeps track of whether a given bit, byte, line or block of information is to be treated as data or executed as an instruction. In a broader sense, a program is also "data" that the computer understands as a set of instructions. Programs can be stored in various ways. They can be stored outside the computer on magnetic or other media. A record of a program on a sheet of paper is called a "listing."

Programs can be executed only when they are in the computer's memory. Since memory is limited in storage capacity, programs are often stored outside memory until the time comes to load one or more of these programs into the Sol; there must be an I/O "interface" to transfer the program from its external storage medium to the memory in the Sol. Since programs and data generated in the Sol may also have to be moved out of the Sol to more permanent media, most interfaces are bidirectional. An interface, in this instance then, is circuitry that adapts one device having a specific function to another device having a different function (or at least a different design) when both the dissimilar devices have a common boundary where they must interconnect to perform an overall system function.

Thus an interface is used when the CPU is instructed to transfer its internally stored programs and data to an external device and/or

media. The Sol is splendidly equipped with three of these input-output interfaces (in addition to the built-in terminal interfaces for the keyboard and video display). These interfaces are:

The Audio Cassette Interface (ACI)
The Serial Communications Interface (Serial Port)
The Parallel Data Interface (Parallel Port)

If the data (including programs) is to be transferred in a string of bits one at a time, the data goes in or out the Serial Communication Interface. An example of serial data transfer is the Sol communicating via a modem over telephone wires or using its serial port to send data to a SolPrinter 3. The data can also be transferred eight bits at a time, when the Sol uses the Parallel Data Interface. The Sol can use this parallel I/O port to send data to a Solprinter 2. (The serial and parallel interfaces, together with the internal keyboard interface, are called "ports," as in a "seaport" where items of trade go in and out.) The serial and parallel port connectors J1 and J2 can be readily seen on the rear panel of the Sol.

The third I/O interface is the Audio Cassette Interface, which is a specialized serial interface that can load from and store to cassette tape. Two recorders can be used at the same time, one being read from, the other being written to; or the CPU can read or write alternately to two recorders. But there is still another type of interface employed by the Sol, the "bus interface"!

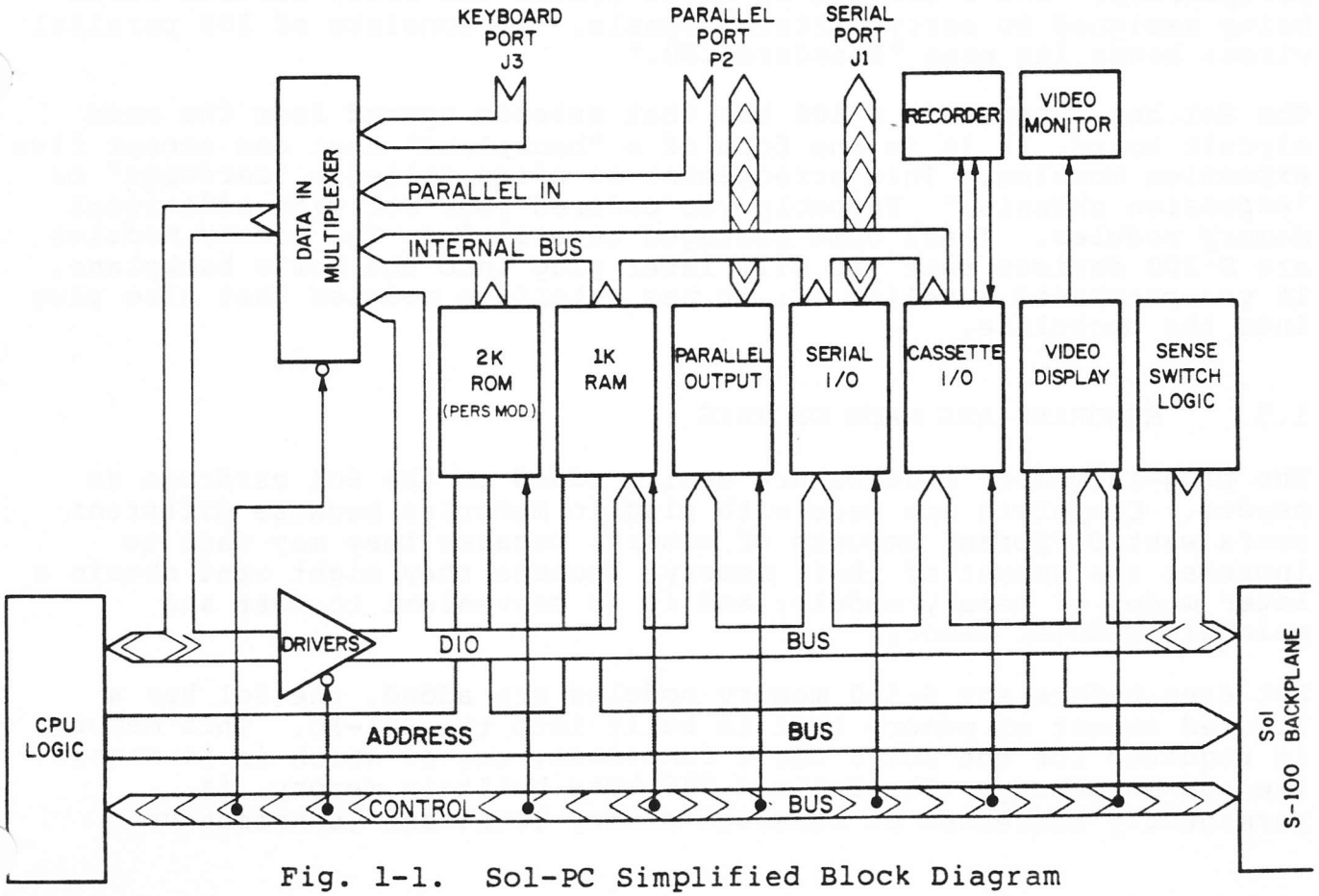


Fig. 1-1. Sol-PC Simplified Block Diagram

1.4.1 The Coming of the Sol S-100 Backplane

One of the most significant and effective characteristics of the Sol computer is a "plug-to-plug compatible" method of interfacing. Plug-to-plug compatible refers to devices having identical connectors and performance specifications so that they are direct replacements for each other. Similarly, a bus interface provides for additional memory modules, special purpose I/O modules, peripheral devices and even other small computers, all having the same connector specifications, to be plugged into a "bus." A bus is a number of parallel wires carrying control, address and data signals in common to all devices connected to the bus. The different bus devices, each plugged into a bus connector, share access to the bus, being synchronized by common control signals, much as the bus used in human transportation enables many people to arrive at their individual destinations via the same vehicle. A big advantage of a bus is that the various devices in a bus system do not need dedicated wiring to each of the other devices with which they must communicate, just as a real bus vehicle obviates the need for many individual cars to get the riders to where they are going. Bus design coupled with the large scale integration of the CPU accounts for much of the small computer's low cost.

Sometimes buses become more or less industry standards and different manufacturers can thereby produce a host of compatible devices. The most widely used bus standard in the small computer industry is the "S-100 bus." This is the bus used by the Sol and its associated peripherals. The S-100 bus is built around the 8080, certain wires being assigned to carry certain signals. It consists of 100 parallel wires; hence its name "Standard-100."

The Sol has a built-in S-100 bus that extends upward from the main circuit board. It is in the form of a "backplane" that can accept five expansion modules. This arrangement is often called a "cardcage" or "expansion chassis." Probably you ordered your Sol with additional memory modules. These come packaged separately. The memory modules are S-100 devices that you will later plug into the Sol's backplane. If you purchased a Helios II, it has interface modules that also plug into the backplane.

1.5 MEMORIES ARE MADE OF THIS

The plug-in memory modules are easily added to the Sol cardcage as needed. Computers are made with plug-in memories because different users want different amounts of memory, because they may want to increase the amount of their memory, because they might want obtain a later model of memory module; and it is convenient to test and maintain modular memory.

But even before any S-100 memory modules are added, the Sol has a limited amount of memory that is built into the Sol-PC. This memory is required for the Sol's basic functions, one of which is to manage the add-on memory. The Sol's 4,096 byte built-in memory, is permanently addressed at reserved memory locations (consecutive

addresses). This memory is divided into 2,048 bytes of Read Only Memory (ROM) and 2,048 bytes of Random Access Memory (RAM). First we shall explore the Read Only Memory.

1.5.1 Psychoanalyzing the Sol Personality Module

The Sol's 2K Read Only Memory (a "K" is 1,024 bytes) is in the form of a plug-in memory module that rides piggy-back on the main circuit board (the Sol-PC). This little module, which should already be installed in your Sol, is the Personality Module. The size of this memory module belies its importance, for without it there would be utter chaos when data begins to stream into the computer from outside devices and is "blitzed" from one circuit to another on the Sol circuit board. For, just as our own personalities program our behavior by selecting, interpreting, and organizing the vast amounts of clamoring data impinging upon our senses from the outside world, so does the Sol Personality Module control the interaction between the user and the Sol system resources: the CPU, the memory and the I/O interfaces.

When humans wake up in the morning, somehow they find themselves with the same personality, more or less, that they had when they went to bed. They don't usually "forget" their personalities. So too, Read Only Memory has a permanent program already manufactured into it. Unlike a magnetic memory and the RAM in S-100 memory modules, ROM can only be read and not over-written or erased by the computer. There is no need to change or modify the program stored in this ROM because of the nature of the program as we shall see below.

A even more important correlary quality of ROM is the fact that it is a "non-volatile" storage medium. Non-volatility means that the program stored in the ROM is not lost when the power supply is turned off. This feature is a convenience to the user who does not have to reload the program stored in the ROM each time the Sol is turned on.

SOLOS

Just what is the program that the user would want to load each time he or she turns on the Sol? What is the Sol's personality? The program that is embodied in the Personality Module is a 2,048 byte program named SOLOS. As manager of the system resources on behalf of the user, and being instantly available when the Sol is turned on, SOLOS is like an executive who is always on the job. In fact, it is called an "executive" or "monitor" program. It is the "operating system" supplied with a Sol. The job of an operating system is to make the system resources readily accessible to the user through a set of "console commands," that you can type in from the keyboard, to engineer the transfer of data among the system resources and external devices, and to act as a "master of ceremonies" during the transition of control from one program to another.

The Sol is able to change it personality modules without becoming schizoid! One module is used by a Sol in a cassette system; another by a Sol in a diskette system, and another can be fitted with a ROM programmed for the user's application. SOLOS and the three available personality modules are described in Section 3, "Introduction to Software." The SOLOS/CUTER User's Manual which comes with the Sol will familiarize you with the SOLOS commands.

1.5.2 The Sol's Built-in Read/Write Memory

Besides its ROM memory, the other section of Sol's built-in memory is read/write memory, called Random Access Memory (RAM). (This is the type of memory used in the additional S-100 memory modules.) Unlike ROM memory, RAM memory can be altered by the CPU. RAM is more versatile than ROM because many different programs can be loaded successively into the same RAM. ROM on the other hand cannot be altered by the CPU and can contain only the information placed in it when it is manufactured.

As a converse of its alterability, however, RAM is volatile. When the power supply is turned off, poof! This property of RAM is usually no disadvantage because the Sol's RAM is really a "workspace" for the CPU to manipulate the data and to store it temporarily until it is transferred to more permanent media if necessary. Most of a computer's memory is of the read/write type because the computer is always changing the contents of its memory, either by moving different blocks of data in or out or by operating on the data stored in memory.

Sol has 2K (2,048 bytes) of built-in RAM, located on the Sol-PC. 1K is used by SOLOS as a scratch pad to keep track of the ever-changing notes and records necessary for SOLOS' executive duties. While the CPU is loading data from cassette, the built-in RAM is sometimes used as a storage buffer. The second 1K stores the current video image ("frame" or "page") being generated by the video display circuitry.

1.6 PRERECORDED SOFTWARE

Instructions and even whole programs can be entered by hand into the Sol through the keyboard, but it is extremely convenient and efficient to have stored programs prerecorded on audio cassettes or floppy diskettes to be fed quickly into the Sol. In conjunction with a cassette recorder or disk memory system, the Sol can load and store any compatible programs and programs or data that you yourself may write or enter on the Sol. As an example, we will be loading Extended Cassette BASIC which comes with the Sol. Section 3 discusses software in general and the Sol software in particular. Some of the games and other programs that are available from Processor Technology are described in Appendix 1.

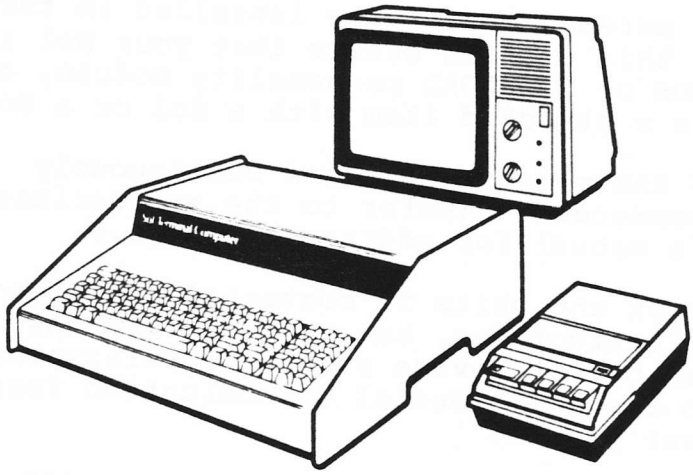
1.7 THE EXPANDING "Sol-ar" SYSTEM

The Sol together with a 16K memory module, a video monitor and a cassette recorder constitute the most fundamental Sol System (Sol System I-A). This system can be expanded by adding plug-in modules to the Sol backplane and/or replacing the memory module(s) with one of larger capacity, available from Processor Technology. The Sol has "slots" in its backplane for five plug-in S-100 modules which can be memories, I/O modules or interface modules such as the Processor Technology Helios II floppy disk controller.

You can also expand your Sol system by adding peripherals such as a second cassette recorder, a Helios II Disk Memory System or a Solprinter. Other useful S-100 devices are available from some manufacturers: if you require additional S-100 backplane connectors, there are add-on cardcages; if you want to add a little color to your "Sol-ar" spectrum, there are S-100 color graphics modules; a modem module can be used to communicate with other Sols and other computers anywhere in the world over telephone lines and satellites.

Refer to Appendix 1 for descriptions of additional equipment for your Sol System.

For further study of small computer systems and software, refer to the bibliography in the appendices of Extended Cassette BASIC User's Manual.



SECTION 2

AT THE CONTROLS

2.1 INTRODUCTION

Information in this section will help you to become familiar with the operation of your Sol Terminal Computer. You will be given brief explanations of the requirements and the operating controls. This section and the following one, Section 3, "Introduction to Software," will acquaint you with the Sol, so that you will feel at ease when, in the Section 4 section, you proceed to load Extended Cassette BASIC and experiment with a simple program.

Certain control switches are inside the Sol; however, these controls are already set for ordinary operation of the Sol. If it becomes necessaary to reset these switches, qualified service persons may refer to Appendix 3, "Internal Controls."

2.2 REQUIREMENTS NEEDED FOR THE EXPEDITION

Available Sol and Sol System configurations are listed in Table 6-1, "Sol System Component Lists." Assemble the items in the following list. You will need these items to load and run Extended Cassette BASIC as presented in Section 4:

) A Sol Terminal Computer connected according to Section 6, "Unpacking and Installation."

2) A SOLOS or BOOTLOAD personality module installed in the Sol. The instructions in this section assume that your Sol is equipped with a SOLOS or BOOTLOAD personality module, one of which is supplied as a standard item with a Sol or a Sol System.

3) A minimum of 16K of RAM memory addressed continuously from Ø. 32K is recommended. (Refer to the appropriate memory module user's manual for address settings.)

4) Video monitor or black and white TV converted for video input. (For TV conversion instructions, see your dealer. As a substitute, a serial output device such as a teletypewriter may be connected to the Sol's Serial Communication Interface connector on the rear panel.)

5) A cassette recorder with motor and audio cables. (If you have not yet selected a recorder, see the appendix, "Cassette Recorder Supplement.")

   OR: A Helios II Disk Memory System, in which case you will be loading Extended Disk BASIC.

6) A cassette prerecorded with Extended Cassette BASIC.

   OR: If you have a Helios II, the PTDOS system diskette.
   (The PTDOS system diskette contains Extended Disk BASIC.)

7) Manuals:

   a) Sol Terminal Computer User's Manual (this manual).

   b) The appropriate memory module user's manual.

   c) SOLOS/CUTER User's Manual.

   d) Extended Cassette BASIC User's Manual.

   e) (and, if you have a Helios:")

      Helios II User's Manual
      PTDOS User's Manual
      Extended Disk BASIC User's Manual

2.3 POWER TO THE Sol (Refer to Fig. 2-1, Sol Rear Panel.)

On the rear panel of the Sol is a square red button that has two positions. This switch is pushed IN to turn on the Sol. (Do not turn on the Sol yet.) In the ON position, a detent holds the switch in. To turn OFF the Sol, push the button again. The detent releases the switch and it pops out to the OFF position.

When turned on, later models of the Sol will have UPPER CASE set as indicated by the red light-emitting diode (LED) in the key.

It is recommended that you turn off your Sol if you are not using it for an extended period of time. This should help extend the life of the memory modules. The Sol needs no warmup time. When the Sol is turned off, the data contained in RAM will be lost.

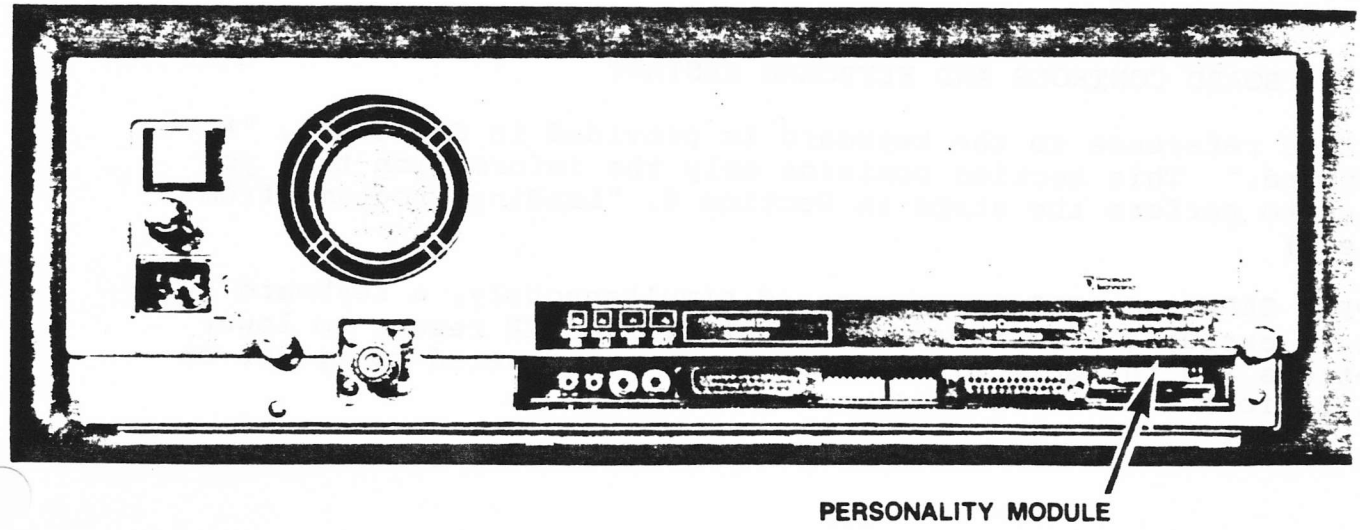


Fig. 2-1. Sol Rear Panel

2.4 THE VIDEO DISPLAY

1) Turn ON the Sol.

2) Turn on POWER to your video monitor by touching its power switch. (The monitor has its own manual if you need to reference it.)

3) The SOLOS prompt character followed by the cursor (>■) should appear at the upper left corner of the screen. The greater-than sign is the SOLOS prompt character; it means "now you can type a SOLOS command." The cursor is the solid vertical rectangle which occupies the character position in the display where the next character is to be printed or the point at which the next operation is to start. Of itself, the cursor does not affect any characters in the display. A prompt character is a distinctive symbol that reminds you which program the computer is running.

4) Adjust the CONTRAST and BRIGHT controls on your video monitor to optimize the display of the cursor.

The SOLOS monitor program automatically takes control in its Command Mode when the Sol is turned on. This sequence is called the power-on initialization. In the Command Mode, which is the primary mode, the Sol is waiting for a command. The two operating modes of SOLOS will be discussed in Section 3, "Introduction to Software."

If the prompt does not appear, refer to Section 7, "TLC For Your Sol."

Your video display, which is controlled by the Sol, is usually set by the factory to display black characters on a white background. If you prefer, a qualified person can reverse the display by using the polarity switch inside the Sol. For details refer to Appendix 3, "Internal Controls."

The lines of program instruction or text are rolled up off the top of the display to make room for a new line at the bottom. This is action is called "scrolling."

2.5 KEYBOARD CONTROLS AND KEYBOARD RESTART

(A detailed reference to the keyboard is provided in Section 5, "At the Keyboard." This section contains only the information that you will need to perform the steps in Section 4, "Loading Programs from Cassette.")

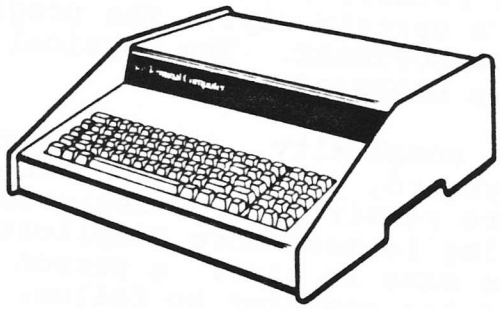
When UPPER CASE and REPEAT are pressed simultaneously, a keyboard restart is executed with the result that UPPER CASE resets to lower case letters and the Sol is reset as if it were powered-up (power-on initialization).

Use the keyboard restart to return to SOLOS from:

1) A program that does not have an exit command or does not recognize its Exit command.

2) A program that is caught in an endless loop as indicated by its inability to respond as it is supposed to. For example, when the program "locks up" the keyboard so that there is no response to pressing any of the keys (except UPPER CASE and REPEAT).

3) A program that has been otherwise rendered defective by an error.



SECTION 3

INTRODUCTION TO SOFTWARE

3.1 WHAT IS SOFTWARE?

A computer serves little purpose by itself. However complex its physical components, however ingenious the design of its circuitry, a computer is useless unless there is a way for a person to tell it what to do.

Control of a computer is achieved by means of "programs" which are lists of instructions written by people in languages that both people and computers can understand. (Actually, the computer does not "understand" anything; rather, it is designed to follow instructions that are expressed in a certain way.) The programs associated with a computer constitute its SOFTWARE. The physical components of the computer constitute its HARDWARE.

Software varies in its complexity. When you type your first BASIC program on the Sol keyboard, you will have created an example of software. The software required to orchestrate and monitor a successful lunar landing is much more complicated than your BASIC program, but it is the same in kind: a person or a group of people write instructions for the computer to follow. Some programs make it possible for other programs to run. For example, your BASIC program will run because a larger program called BASIC is interpreting it for the 8080. BASIC will run because SOLOS (or PTDOS) is regulating the operation of the Sol and its peripherals. (BASIC "makes calls" to SOLOS to perform certain functions so that the functions need not be repeated in BASIC.) When you communicate with the computer, whether by typing on its keyboard or by loading information from a cassette or disk, the computer responds according to instructions that people have given it. The apparent intelligence of the computer is determined by the care that you and other people have taken in instructing it.

A set of instructions that you type on the keyboard is not very permanent: it exists only in the volatile read-write memory of the Sol. If the power fails in your building, or if you turn the computer off, the program disappears. If a program is very short, you might not mind the time it takes to type it on the keyboard every time you want to use it. Most programs, however, are much too long for such a practice to be convenient. Entering a program manually is time-consuming and likely to introduce errors. You might spend hours typing the instructions and then execute them, only to find that you have omitted or mistyped something and accidentally destroyed a part of your program. (Remember that a computer will act only on what you have actually told it, not on what you might have meant to tell it.) Recording software on a medium like cassette tape, disk, or read-only memory is a way of giving permanence to a correct version of the software; it also reduces the amount of time required to load the program into memory. A program that would take hours to type into the computer correctly can be "read" from a cassette in minutes, from

a diskette in a second. A program recorded in read-only memory need not be "read" into memory at all, because it is already there.

Software may be classified according to its function:

The most important program associated with a computer is its OPERATING SYSTEM. The function of an operating system is to provide an interface between all other programs and the computer hardware. Processor Technology Corporation offers two operating systems for the Sol: SOLOS for cassette systems and PTDOS for diskette systems.

A program designed to solve a specific kind of problem -- for example, to keep records of inventory, perform statistical analyses, or process text -- is called an APPLICATION PROGRAM. Most of the programs that you will write, at least at first, will fall into this category.

Programs that help a programmer to develop other programs are called PROGRAM DEVELOPMENT SOFTWARE. Programs in this category include editors, assemblers, interpreters, compilers, and debuggers.

Appendix 1 contains a partial list of prerecorded programs that you can buy for your Sol.

3.2 COMPUTER LANGUAGE AND LANGUAGE TRANSLATORS

Computers are good at arithmetic. This fact has led many people to believe that only mathematicians can "talk" to computers. Anyone who has ever "spoken" to either a computer or a mathematician will realize that, although all computers and most mathematicians are good at arithmetic, language remains the primary vehicle for communication.

Computer languages are simpler than "natural" languages for several important reasons. When people communicate with one another, the meaning of a sentence is conveyed not only by the individual words and their order, but also by the context in which those words occur. (In spoken language, intonation and various physical factors also contribute to context.) A person can guess at the meaning of an unfamiliar word by looking at surrounding words, sentences, and paragraphs. The meaning of an entire sentence can be dependent on context; we have all heard people accused of "taking words out of context" to confuse or mislead others. A computer is not as well equipped to handle context as a human being is. To eliminate ambiguity, a computer language must consist of a limited number of statement types and a carefully defined vocabulary. A computer language, by contrast to a "natural" language, is "context-free," i.e., a statement must have the same meaning, regardless of its context.

In order for statements in a computer language to be understood by a computer, they must be reduced to a pattern of binary codes directly intelligible to the computer hardware. ("Binary" means "in base 2.") This binary "language" is called MACHINE LANGUAGE. Programs that translate statements from other languages into MACHINE LANGUAGE are called LANGUAGE TRANSLATORS.

The language that requires the least translation for a given processor is called its ASSEMBLY LANGUAGE. An assembly language is the symbolic form of the corresponding machine language; it uses symbols to present operations and memory addresses. Because the Sol is based on an 8080 microprocessor, its assembly language is called 8080 Assembly Language. A language translator that translates programs from assembly language to machine language is called an ASSEMBLER.

Writing programs in assembly language is more difficult than writing programs in other computer languages, because the programmer must understand how the central processor is structured and how memory is addressed. It takes longer to write a program in assembly language than in a "higher level" language, but the resulting machine language program can be shorter and more efficient.

The easiest computer languages for people to learn and use are the HIGHER-LEVEL LANGUAGES. These languages, e.g., FORTRAN, BASIC, and PILOT, are closer to natural languages than is assembly language. For example, the way to say

```
PRINT "HELLO"
```

in BASIC is

```
PRINT "HELLO"
```

The structure of a higher-level language does not reflect (or reflects only dimly) the structure of a particular computer. To write a program in FORTRAN, BASIC, or PILOT, you do not have to know anything out the 8080; you DO have to know exactly what you want to accomplish and how you are going to explain it. A language translator that translates higher-level language programs to machine language is called either an INTERPRETER or a COMPILER. For an entertaining and informative discussion of interpreters and compilers, read the article entitled "Your Personal Genie" in the May/June 1977 issue of Personal Computing Magazine.

3.3 SOLOS

SOLOS was introduced in Section 1.5 as the "personality" of the Sol. This program provides the Sol user with a convenient means of accessing and managing the system resources and controlling execution of all other programs that run on the Sol.

SOLOS is available on either of two Personality Modules. (Recall that a Personality Module is a plug-in circuit board containing a read-only memory chip.) If you have a Sol System I-A or II-A, your system has the SOLOS Personality Module; if you have a Sol System III-A, III-B, IV-A, or IV-B, your system has the BOOTLOAD Personality Module.

The two versions of SOLOS are almost identical. Each has 19 commands that allow you to examine and change the contents of memory, control one or two cassette recorders, determine the source of input and destination of output, and execute other programs. Some of the commands perform "housekeeping" functions such as setting the rate at which data are read from tape, or the rate at which characters are

displayed on the video monitor. The respects in which the versions differ are few but important. The version of SOLOS on the SOLOS Personality Module has two possible operating modes: Command Mode, in which the Sol operates as a stand-alone computer, and Terminal Mode, in which the Sol functions as a video terminal for connection to another computer. On the BOOTLOAD Personality Module SOLOS has only one operating mode, the Command Mode; the command to enter Terminal Mode is replaced by a command that loads PTDOS, the Processor Technology Disk Operating System. (For details on PTDOS, refer to PTDOS User's Manual.)

3.3.1 Command Mode

In the SOLOS Command Mode, Sol operates as a stand-alone computer under control of the program contained in the personality module. SOLOS can "hand over" control to other programs that have been loaded into the Sol, usually from cassette tape or diskette. For detailed information about the SOLOS operating system, including a complete discussion of the commands, refer to the SOLOS/CUTER User's Manual, Second Edition.

With the SOLOS Personality Module installed, the computer is in the Command Mode when power is applied to the Sol. The > that you see on the screen is a "prompt" character; it tells you that SOLOS is waiting for you to type a command.

All of the instructions in this section refer to operations in the SOLOS Command Mode. Terminal Mode is described in the Sol technical manual.

3.3.2 "Try It"

In this section you will try out a few SOLOS commands. Follow all instructions exactly. If you make a typing error while entering a command, use the DELete key to backspace and erase characters.

1) Turn on the Sol and the video monitor. Look at the keyboard; if the UPPER CASE key is not illuminated, press it once. (When you type a SOLOS command, you must always use upper case letters; otherwise the command will not be "understood.")

2) Type the following command:

   DUMP CØØØ C0E0

   The DUMP command displays the contents of memory on the video monitor or other output device. The "CØØØ" and the "CØEØ" in the command are hexadecimal numbers: that is, they are numbers in base 16. In this command, the numbers identify addresses in memory. CØØØ is the beginning address of SOLOS.

3) Press RETURN to show that you have finished typing the command. Lines of hexadecimal data will scroll (move) rapidly up the screen. The display will stop scrolling after the contents of all locations from CØØØ to CØEØ, inclusive, have been displayed. The numbers that you see are part of the SOLOS program, in machine language form.

4) Type the following command:

ENTR C9ØØ

The ENTR command is used to enter hexadecimal data from the keyboard into the read-write memory of the Sol. The "C9ØØ" in the command identifies the address at which the first entry will be stored.

5) Press RETURN to show that you have finished typing the command. A colon (:) prompt character will appear at the start of the next line.

6) Type the following data, including blanks. If you make a typing error, use the DELete key to backspace. (The data represent a little program that prints the word "Hello" on the screen. You will be typing the instructions in machine language form; the original assembly language program is printed at the end of this section.)

21 11 C9 7E FE ØØ CA Ø4 CØ 47 CD 19 CØ 23 C3 Ø3 and RETURN

When the colon reappears, type

C9 2Ø 2Ø 48 45 4C 4C 4F ØØ/ and RETURN

(The slash (/) shows that you have finished entering the data.)

The program you have just typed should now occupy addresses C9ØØ to C918, inclusive, in the read-write memory of the Sol.

7) To verify that the program is in memory, give the DUMP command:

DUMP C9ØØ C918

Then press RETURN key. The output should look just like what you typed in step 6, except that the entries will be arranged 16 per line, and each line will begin with the address of its first entry.

8) Type the following command:

EXEC C9ØØ

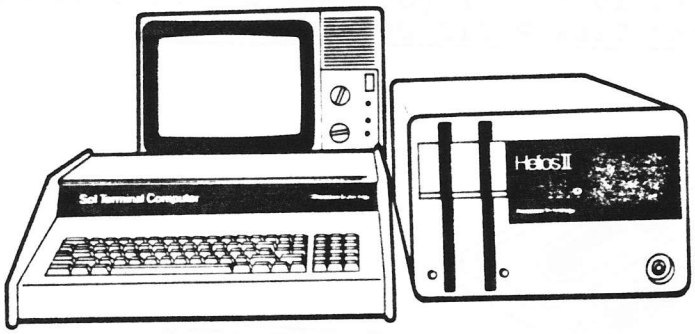
The EXEC command executes (or "runs") the program that starts at address C9ØØ.

9) Press RETURN to show that you have finished typing the command. The word "HELLO" should appear on the screen. If it does not appear, or if something else happens, you made a typing error in step 6. Press UPPER CASE and REPEAT together, and try again.

10) Using your SOLOS/CUTER User's Manual, experiment with the other console commands until you feel at home with your Sol.

* PROGRAM TO PRINT "HELLO" FOR Sol USER'S MANUAL

```
  ORG ØC9ØØH
  LXI H,HELO
SHOW MOV A,M
  CPI Ø
  JZ ØCØØ4H
  MOV B,A
  CALL SOUT
  INX H
  JMP SHOW
HELO ASC ' HELLO'
  DB Ø
SOUT EQU ØCØ19H
```



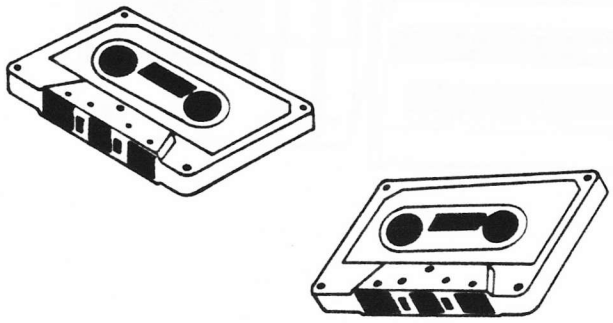
SECTION 4

LOADING PROGRAMS FROM CASSETTE

4.1 INTRODUCTION

In this section you will load from cassette the program Extended Cassette BASIC that is supplied with the Sol. Then, using Extended Cassette BASIC, you will enter a miniature demonstration program, as if you had just programmed it yourself, and write it back out to cassette. Finally, you will load the program back in from cassette and run it again to complete the cycle that applies to all the programs you will compose from here on.

The instructions given in this section are intended to get you started in using your Extended Cassette BASIC User's Manual, which is a handbook for the programming language BASIC when used with a Sol and a cassette recorder. (The edition referred to in this section is the first printing, January, 1978. If you have a Helios II Disk Memory System, use the Extended Disk BASIC User's Manual.



4.2 AUDIO CASSETTE RECORDERS (Preliminary Information)

Refer to your SOLOS/CUTER User's Manual, Appendix 1, "About Cassette Recorders and Cassette Files," for important information related to this section.

A complete procedure for reading and writing on cassette is given in Section 4.3, "Getting into BASIC." The following paragraphs in this section are preliminary notes.

4.2.1 Operating A Cassette Recorder

This section assumes that you have the standard cassette recorder supplied with the Sol System I-A, the RQ-413A. If you do not have a recorder or have a different recorder, refer to the Appendix, "Cassette Recorder Supplement."

If you plan to operate two recorders, also refer to the Appendix, "Cassette Recorder Supplement."

If you have problems loading from or storing to cassette, refer to Section 7.5, "Trouble-shooting Cassette Operations."

4.2.2 Cassette Recorder Operating Tips

For best results when using audio cassette recorders with the Sol, observe these tips:

Set the volume control to number 6 (about 2/3 full volume). Set the tone control at number 10 (maximum). The Sol has automatic gain control that compensates for a wide range of levels; but the above settings give the most reliable results. If you have a recorder different from the one supplied in a Sol System, experiment to find the optimum setting for the volume and tone controls.

CAUTION

> The MICROPHONE input can be live when you are recording through the AUXILIARY input on some recorders other than the one supplied with a Sol system. De-activate the MICROPHONE input according to the manufacturer's instructions. (In some cases, you can de-activate the MICROPHONE input by plugging a termination plug into the MICROPHONE jack.)

4.2.3 Notes on Writing to Cassette

In the case of two recorders, Unit 1 and 2 must be specified in the SAVE command in order to select the desired recorder. A default selects Unit 1. Refer to your SOLOS/CUTER User's Manual for instructions on how to use tape commands.

To make file retrieval much easier, keep a record of where individual files are recorded. After recording the last file on a side, rewind the tape, set the tape counter to zero, and issue a CATalog command (see SOLOS/CUTER User's Manual). As each file header is displayed, make a note of the following and mark the cassette with this information:

1) Tape counter reading.

2) Exact file name.

3) Load address.

4) File length.

4.2.4 Notes on Reading from Cassette

In order to read a specific file from tape, you must start the tape at least two seconds ahead of that file. This delay allows the Sol audio cassette interface circuitry and the recorder playback electronics to stabilize after power is turned on. Because all file searches are in the forward direction, the simplest approach is to rewind the cassette(s) completely before a read operation, unless you know that the file of interest is recorded at least two seconds into the tape from where the head is currently positioned.

4.3 GETTING INTO BASIC (Procedure for Using A Cassette Recorder)

1) Be sure you have connected your system according to Section 6, "Unpacking And Installation," and have familiarized yourself with the information in Section 2, "At the Controls" and Section 3, "Introduction to Software."

2) Turn on your Sol.

3) Select from your system components the cassette labeled "Extended Cassette BASIC."

4) Insert the cassette, label up, into the recorder.

5) Initially, adjust the following controls on the recorder:

   (If you have a recorder other than the one supplied with the Sol System I-A, try different volume settings until you find a reliable setting.)

   a) VOLUME: Set to 6 (about 2/3 full volume).

   b) TONE: Set to 10 (maximum for sensitivity to the higher frequencies).

   c) COUNTER: Reset to Zero.

6) Turn on POWER to your video minitor.

7) The SOLOS prompt character followed by the cursor (>▮) should appear on the left of the screen.

8) When the prompt appears, set UPPER CASE by pressing the UPPER CASE key so that its indicator light comes on. Type on the Sol keyboard the command: CA and press RETURN.

   This command allows the recorder motor to operate under local control. (Alternately, you can disconnect the REMOTE plug from the recorder.)

9) Press REVIEW (REWIND) on the recorder, to rewind the cassette to the starting position. All the tape should be on the left reel of the cassette. Press STOP on the recorder.

10) Play the tape for about 15 seconds to advance it past the leader. (This can also be done using your index finger to turn the takeup reel before inserting the tape.)

11) If you have disconnected the REMOTE plug, reconnect it.

12) On the Sol, press MODE SELECT.

    This will re-initialize SOLOS and remove local control of the recorder.

13) Press PLAY on the recorder.

    The tape should not move. (If it does, continue with the next step, but afterwards refer to Section 7.5, "Trouble-shooting Cassette Operations.")

14) Type in the command: XEQ BASIC

    and press RETURN.

    (XEQ stands for EXECUTE; "BASIC" is the name of the file on the cassette.)

    The cursor should disappear and the tape should move for about 3 minutes. The display should not otherwise change. When the program has been loaded into the Sol, a copyright notice appears followed by an announcement that BASIC is sizing up the amount of memory possessed by the computer in which it has suddenly found itself.

15) Press STOP on the recorder.

    Shortly BASIC will display its findings in hexadecimal and ask that you enter the beginning address of the memory you may wish to reserve from the operations of BASIC.

    For now, press RETURN. This tells BASIC that it is free to use all of memory for its work space.

16) BASIC presents two more questions whose purpose is to reduce the size of the program to leave the maximum amount of memory for working space. It asks if you wish to delete "matrix operations" and "extended functions." You will be able to learn about these in the Extended Cassette BASIC User's Manual, but for now, answer by typing "Y" to both questions.

17) BASIC should now present the message: READY.

    The cursor is already at the beginning of the next line. You are now "in" BASIC.

4.4 WRITING PROGRAMS AND SAVING THEM TO CASSETTE

4.4.1 Writing a Program

Let us write a little program whose purpose is to calculate the miles per gallon you get with your motor vehicle or airplane and the cost per mile. This will be an extremely simple example so, if you already know the programming language BASIC, you can skip along to Section 5, "At the Keyboard."

1) Let us call the program MPG. You could do these calculations on a calculator very easily, so you will be able to follow the steps in writing this program.

2) In BASIC, a "statement" is an instruction and is preceded by a line number. We will number the first statement "10" and the second "20". We skip nine line numbers in between in case we want to go back and insert other instructions as an afterthought.

3) Because we want the program to ask the user to enter the mileage data to be used in the calculations, we will use the PRINT statement to display a message on the screen. Enter the first statement through the keyboard as follows:

   (If you make a typing error, use the DEL key to backup, and retype the entry.)

```
10 PRINT "ENTER MILES";
```

   This statement will cause the program to ask the user to enter the miles traveled on a tank of fuel. The semicolon tells the computer to print the user's response to statement 20 on the same line.

4) In the next statement we want the user of the program to enter the number of miles, which we shall represent by the variable "M." Variables are described in Section 2.3.2, "Variables," of the Extended Cassette BASIC User's Manual. Enter the second statement:

```
20 INPUT M
```

   This causes BASIC to accept a value entered from the keyboard and to associate that value with the name "M."

5) We program the entry for the gallons of fuel consumed in the same manner as the miles:

```
30  P. "ENTER GALLONS";
40  IN. G
```

   Note that we can abbreviate to save time. "P." stands for "PRINT" and "IN." stands for "INPUT." Abbreviations are explained in your Extended Cassette BASIC User's Manual, 2.2, "Definitions of Commands and Statements."

6) The statements for the entry of the cost of the fuel take the same form as those for miles and gallons:

```
50 P.  "ENTER COST IN PENNIES";
60 IN.  C
```

7) Now we program the answer and calculation for the miles per gallon. Enter the statement:

```
70 P."MILES PER GALLON=", M/G
```

As described in your Extended Cassette BASIC User's Manual, 2.3.3, "Expressions," "M/G" means "the value entered for M divided by the value entered for G." The comma that follows the PRINT statement signifies that the answer to the expression is to be printed out on the same line.

8) The final statement takes the same form as statement 70. Enter:

```
80 P."COST PER MILE IN CENTS", C/M
90 END
```

As described in the Extended Cassette BASIC User's Manual, 4.3, "Stopping or Delaying Execution," we terminate the program with the END statement.

9) To see the program you have entered and check it over before running, enter the command: LIST and press RETURN.

(Note that commands do not need line numbers since they are not part of the program.) BASIC will list your program on the screen as follows:

```
10 PRINT "ENTER MILES";
20 INPUT M
30 PRINT "ENTER GALLONS";
40 INPUT G
50 PRINT "ENTER COST IN PENNIES";
60 INPUT C
70 PRINT "MILES PER GALLON=", M/G
80 PRINT "COST PER MILE IN CENTS=", C/M
90 END
```

10) If you wish to correct an error or change a statement in your program, you may do so by simply retyping it including the line number. To insert an additional statement, number it with one of the intervening line numbers.

11) To see that your program runs as it should, enter the command RUN and press RETURN. (In the process of debugging a program, you can run the program starting at any instructions by including the line number of the instruction. For example, RUN 40.)

Enter data as called for by the program to make it continue along.

4.4.2 Saving a Program To Cassette

When you are satisfied with the program, try saving it to cassette as follows:

1) Enter the BASIC command: SAVE MPG,T

and press RETURN.

"MPG" is the name of the program (to become a file on cassette).

The "T" is "Text Mode" in which the program is to be saved.

The SAVE command is described in detail in your Extended Cassette BASIC User's Manual, 3.4.3, "Read or Writing on Tape."

BASIC displays the message:

Prepare Tape Unit 1 for writing to: MPG.

2) Load the recorder with a blank cassette. Do not use your cassette copy of Extended Cassette BASIC.

3) REWIND the cassette if necessary.

4) Play the tape for about 5 seconds to stabilize the recorder.

6) Depress the RECORD and PLAY buttons on the recorder.

7) Press any key to tell BASIC to start recording.

When the program has been recorded, BASIC returns the cursor with the message: READY.

8) Press STOP on the recorder.

4.4.3 Recalling a Program from Tape

(These instructions assume you are still in BASIC, although programs can be recalled from tape while in SOLOS.)

1) Type in the command: GET MPG,T

and press RETURN.

The GET command is described in your Extended Cassette Basic User's Manual in Section 3.4.3. (GET in BASIC is not the same command as GET in SOLOS.)

BASIC displays the message:

Prepare Tape Unit 1 for Reading: MPG.

2) REWIND the cassette if necessary.

3) Press PLAY on the recorder.

4) Press any key to tell BASIC to start reading.

5) Check the tape to see that it is moving. When the program is loaded, BASIC displays the name of the program and its beginning and ending addresses.

6) Press STOP on the recorder.

7) Run the program by entering the command: RUN and press RETURN.

To exit from BASIC, type the command B. and press RETURN. "B." is short for BYE. This command returns control of the computer to SOLOS.

BASIC is still in memory starting at address Ø. To re-enter, type the command: EXEC Ø

4.5 THE Sol ALSO RISES

Having read the previous four sections you are now considered a Sol fledgling and are, therefore, presented with the following choices of how to proceed with this manual. If you are anxious to start writing your own programs, you can start reading your SOLOS/CUTER and Extended Cassette BASIC manuals, using this manual as a reference as the need may arise. If you are interested in obtaining pre-recorded software, or additional equipment, you might read Appendix 1 of this manual. Or if you want more in depth background on the Sol, just read on to Section 5, which is all about the Sol keyboard.

SECTION 5

AT THE KEYBOARD

5.1 THE KEYBOARD, GENERAL INFORMATION

This section is a detailed reference for using the keyboard when the SOLOS monitor program is resident in the Sol, for the display of characters and symbols associated with SOLOS, and for programming the Sol's response to the keyboard output.

The keyboard is an input device that produces ASCII encoded data. (ASCII = American Standard Code for Information Interchange.) The keyboard output is hardwired to the Sol keyboard port. Under the direction of SOLOS or another program currently in control, the ASCII code generated by the keyboard is decoded by a 6574 character generator ROM which generates a dot pattern for the video display.

The Sol Terminal Computer has an ASCII 96-character keyboard. Its key arrangement conforms to the standard typewriter format. The ASCII character set, which includes alphanumerics, punctuation marks and control codes is shown in Appendix 4, Table A4-1, "Sol Keyboard Characters Keys." There are 22 function keys (including ASCII functions and five cursor controls); these are given in Table A4-2, "Sol Keyboard Function Keys." A separate 15-key arithmetic pad is also provided.

5.1.1 Key Functions are Programmable

The exact function of most keys on the Sol keyboard is determined by the software used (for example, the program contained in the Personality Module). Except for the keys that are "hardwired," any other program can also control the response of the system to the codes produced by the keyboard. Any key that generates a code can be redefined by a program to perform a specific function. (The codes are given in the tables in Appendix 4.) However, the high order bit of the binary code (the eighth bit) of some of the function keys is set by the Sol keyboard as determined by the Sol rather than by the ASCII standard. For example, the function key RETURN generates the code 8D, whereas the control character CTRL/M generates ØD. (Control keys and the transmission of keyboard codes are explained in 5.4.)

In this section, each key function is described in terms of its role in the SOLOS Terminal Mode unless otherwise specified. SOLOS in the LOCAL Terminal Mode allows most of the symbols generated by the character generator to be seen when their corresponding keys are activated, whereas SOLOS in the Command Mode is programmed to recognize the keyboard codes according to its own purposes. Many keys, especially control keys, do not serve the same functions in Terminal Mode that they serve in SOLOS Command Mode, or in BASIC, PTDOS, ALS-8, etc. Control keys are discussed in Section 5.4.

For purposes of exposition in this section, the Sol keyboard keys are ivided into three categories:

1) Character keys which are similar to typewriter keys which print characters.

2) Function keys which primarily perform functions rather than print characters.

3) Control keys which are the character keys generating a different code when pressed with the key CTRL.

A separate subsection is devoted to each. Regardless of their type, nearly all the keys generate an ASCII code. Keys that are hard-wired (internally connected) do not generate ASCII codes. As will be seen in this section, some of the keys in one category generate the same ASCII code as some keys in another category. This overlap is especially the case with control keys and function keys.

5.1.2 Operating Features and Keyboard Indicators

The Sol keyboard features N-key rollover. That is, several keys can be pressed at the same time without loss of characters or commands; key entries are made in the order of actual key closures. (A scanning circuit prevents simultaneous key operation.)

Three keys (SHIFT LOCK, UPPER CASE and LOCAL) have indicator lights to indicate keyboard/terminal status. When any of these keys is pressed to turn an indicator light on, the light remains on after the key is released to show that the status persists. Pressing UPPER CASE and LOCAL again turns the light out to indicate the alternate status. SHIFT LOCK is turned off by pressing SHIFT. The indicators are further described under the individual key description.

5.1.3 Using Table A4-1

In Table A4-1, there are three major columns, one for the key when UNSHIFTED (lower case), one for the key UPPER CASE or SHIFTED (shifted dual character) and one labeled "CONTROL" for when CTRL is pressed simultaneously with a character key. Each of these three columns is broken down into two sub-headings. The column headed by "KEY/SYMBOL" gives the labels marked on top of the character keys, in the sequence found on the keyboard. (In the case of control keys, only the symbol displayed is given.) "HEX CODE" gives the hexadecimal form of the code generated by the keyboard when the key is pressed while being shifted, unshifted or "control." In response to the code, the Sol character generator can provide for the display circuitry, a symbol given in the columns "KEY/SYMBOL" and "SYMBOL." Except for control characters, the symbol displayed is generally the same as that on the keytop. (Some keys are programmed by SOLOS to move the cursor without displaying a symbol.)

Looking at the "W" entry in Table A4-1, and reading across the table, we see that:

1) Pressing "w" unshifted generates the code 77 producing a lower case "w."

2) Pressing "W" shifted generates the code 57 producing an upper case "W."

3) Pressing CTRL (control) and "W" together, whether shifted or unshifted, generates the code 17 producing the control character graphic symbol for the ASCII "end of transmission block:" (┤)

5.2 CHARACTER KEYS (Refer to Table A4-1.)

The character keys include upper and lower case alphanumerics, punctuation marks, and special graphic symbols. They are arranged for the most part as on a standard typewriter. Pressing one of these keys causes the associated character to be entered into the Sol. They can be activated in either the unshifted, shifted, or control modes. If there are two characters labeled on a key, the upper character is typed when the key is pressed simultaneously with the SHIFT key. When unshifted, the keyboard generates the lower character. (See also Section 5.3.2, "Upper Case.")

If you wish to see the character set displayed at this time, use the procedure at Section 5.4.3, "Familiarization with the Control Symbols," but press each of the alphanumeric, punctuation and symbol keys without pressing the control key. As each is pressed, the corresponding character in Table A4-1 should appear on the screen.

5.3 FUNCTION KEYS

(Refer to Table A4-2, "Sol Keyboard Function Keys.")

The primary purpose of the function keys is to perform special functions rather than print characters; however, many functions keys do generate standard codes which can be transmitted in Terminal Mode and can print or display their associated graphic symbol, if they are so directed by the program that controls them. (Refer to Table A4-2 to see whether a code and symbol are generated.) An example of a function key that is similar to the carriage return on a typewriter is the RETURN key. Except for DEL, the function keys are unaffected by the SHIFT key.

The function keys generally are distinguished from the character keys by their size and color. They are either black or white, whereas the character keys are grey. The function keys are also mostly larger than the character keys. (There are a few exceptions.)

The ASCII codes of some function keys overlap the codes of some control keys. Control keys are discussed in Section 5.4.

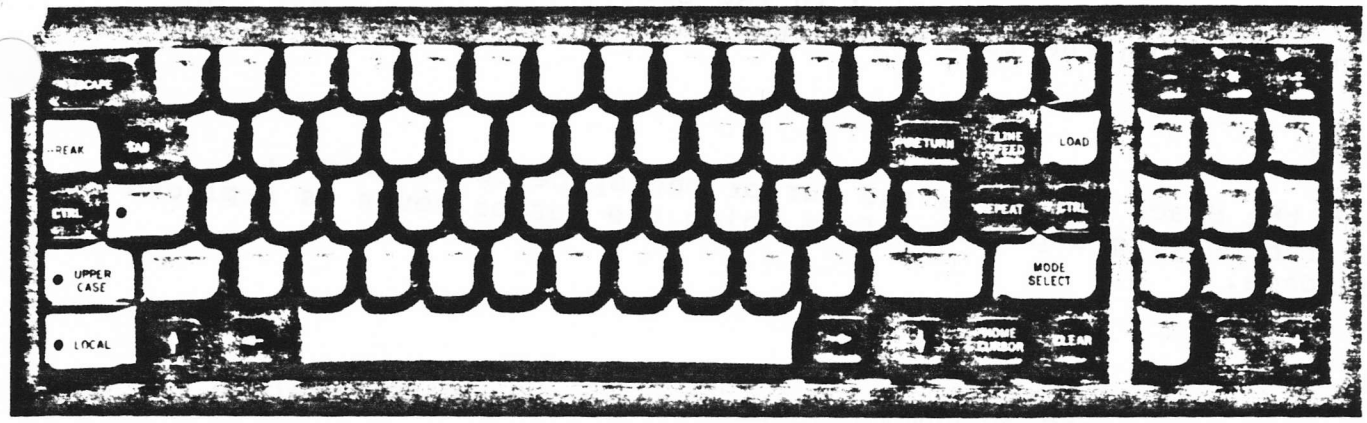


Fig. 5-1. Sol Keyboard

5.3.1 Cursor Control Keys

The cursor is the solid vertical rectangle that occupies the character position in the display where the next character is to be printed or the point at which the next operation is to start. In itself, the cursor does not affect any characters in the display. (The display of the cursor is controlled by internal switches described in the appendix "Internal Controls.")

Five keys control the movement of the cursor (excluding the space bar). They are: HOME CURSOR and the four small black keys marked with directional arrows. They are located in the lowest row of the keyboard, on either side of the space bar.

To move the cursor up, down, left or right, press the applicable cursor control key. Each time you press a key, the cursor moves one unit in the direction you indicate--one space horizontally or one line vertically. These keys may be used with REPEAT to continuously move the cursor. In the Terminal Mode, when the cursor comes to the end of a 64-character line, it moves to the extreme edge of the adjoining line, above or below depending on its direction. In the Command Mode, the cursor returns to the beginning of the same line.

Pressing HOME CURSOR moves the cursor to its home position--the first character position in the upper left corner of the screen; it does not otherwise affect the display.

None of the cursor control keys are affected by SHIFT status. None are displayed or transmitted. (Refer to Section 5.4.2, "Transmission and Display of Control Keys.")

5.3.2 Individual Descriptions of Function Keys

Keys are described here generally in the order found on the keyboard, first in a group on the left side, then in a group on the right side.

SPACE BAR

Pressing the Space Bar moves the cursor one space to the right; a character occupying a position from which the cursor moves is replaced with a space.

ESCAPE

The display of the ESCAPE symbol is masked off (not displayed) in SOLOS. (Refer to SOLOS/CUTER User's Manual.)

BREAK

Pressing BREAK forces the Serial Communication Interface (SCI) output line to a space level for as long as the key is depressed. (Some communications systems use this feature.)

TAB (See Table A4-2.)

CTRL (Control) See the special section on Control Keys, 5.4.

SHIFT LOCK and SHIFT LOCK Key/Indicator

This large grey key on the left of the keyboard works like the LOCK key on a standard typewriter. The SHIFT key is a direct internal operation (hardwired). When pressed it shifts lower case letters to upper case letters and lower dual characters to upper dual characters. The keyboard remains in upper case as long as SHIFT is held down. SHIFT is active independently of the status of UPPER CASE.

Pressing SHIFT LOCK so that the indicator light goes on, locks the SHIFT key electronically in the upper dual character position. Pressing SHIFT again returns the keyboard to lower dual character position and causes the SHIFT LOCK indicator to go out.

UPPER CASE Key/Indicator

This is a large white key on the lower left side of the keyboard. Pressing this key so that the indicator light goes on activates the upper case keyboard function so that all alphabetic characters entered from the keyboard, regardless of SHIFT key status, are produced as upper case characters. Dual character keys are not shifted. (Dual character keys, however, do respond to the SHIFT key.) With the indicator light on, the Sol keyboard essentially simulates a Teletype (TTY ASR 33) keyboard. ("Teletype" is a registered trademark of TeleType Corp. ASR= Automatic Send/Receive.)

Pressing UPPER CASE to turn the indicator light off return the keyboard to normal SHIFT key operation.

UPPER CASE and REPEAT (See Section 2.5, "Keyboard Restart".)

LOCAL Key/Indicator

Pressing LOCAL, so that the indicator light goes on, sets the Sol for local operation in which keyboard entries are not transmitted, but they are "looped back" to the Serial Communication Interface (SCI) input for display. When set, the LOCAL key, through internal circuitry, connects the serial output to the serial input and disables serial transmission external to the Sol. That is, the Sol is not "on line." Pressing LOCAL again, so that the light turns off, ends local operation. (This corresponds to the local/line switch on a TTY.)

RETURN

RETURN is a large black key in the upper right corner of the keyboard. The function of the RETURN key is similar to that of the carriage return on a typewriter. (This is the same action as a TTY carriage return.)

In the Command Mode, RETURN is used to enter a command after it is typed; the command is thereupon executed by SOLOS. All characters on the line to the left of the cursor are interpreted as the command. After the command is executed, the cursor is returned to the left margin on the next line. If more than one command line is on the screen, one can execute any one of them as follows: position the cursor to the right of the desired command and press RETURN. (This procedure will work even if the command has already been executed, as long as the command has not scrolled off the screen.)

In Terminal Mode, when RETURN is pressed, an ASCII CR character is sent to the remote computer and the cursor is moved to the beginning of the line on which it resides.

In either mode RETURN also erases all data in the line to the right of the original cursor position.

LINE FEED

Pressing LINE FEED moves the cursor vertically downward one line. (This is the same action as a TTY line feed.) In the Command Mode, LINE FEED function exactly like RETURN, except that it does not erase any data in the line to the right of the original cursor position.

LOAD

The LOAD key character is displayed but causes no other action in Command Mode.

DEL (Delete)

The delete key is a small grey key labeled "DEL," located on the right side of the keyboard. The delete key is active when unshifted. When the cursor is positioned over a character and the delete key is pressed, the character is replaced with a space and the cursor moves one space to the left. Used in conjunction with the REPEAT key, DEL can be useful in deleting a string of characters to the left of the cursor.

The DEL key is also a dual character key; when shifted, it generates an underline. It is also a control key. Because the DEL key has aspects of both a character key and a function key it is included in both Table A4-1 and A4-2.

REPEAT

This is a medium-sized black key on the right of the keyboard. When pressed at the same time as another key, it repeats the function of the other key until either key is released. For example, when REPEAT is pressed together with the space bar, spaces will be cleared of characters as the cursor moves to the right; when used with the cursor directional keys, REPEAT moves the cursor continuously until released. With a character key, it repeats the character. The rate of repetition is approximately 15 times per second.

REPEAT is a hardwired function that does not generate a code.

When UPPER CASE and REPEAT are pressed simultaneously, a keyboard restart is executed (SOLOS is re-initialized) As a secondary result, UPPER CASE resets to lower case letters. (See 2.5, "Keyboard Restart.")

CTRL (See special section on control keys, 5.4.)

MODE SELECT

This is the large white key in the lower right corner of the main keyboard. If the MODE SELECT key is pressed, while SOLOS is in either mode, the Sol will be re-initialized to the SOLOS Command Mode and display the prompt character followed by the cursor.

HOME CURSOR (See 5.3.1, "Cursor Control Keys.")

CLEAR

In both modes, pressing CLEAR erases the entire screen and moves the cursor to its "home" position (upper left corner of the screen).

5.4 THE CONTROL KEYS

5.4.1 General Information

CTRL is used with character keys to initiate functions or generate control characters defined in Table A4-1. A control character is an ASCII standard character whose code specifies an operation to be performed, rather than a symbol to be displayed. Usually the operation is to be performed by a peripheral such as a printer. For example, a Line Feed (ØA), when transmitted to a printer, causes the printer, which recognizes the ASCII code, to move the paper platen up one line. If the resident program calls for it, the control character can cause the Sol to produce a graphic symbol representing the control character.

A control character is generated when CTRL is held down while pressing a character key (regardless of the status of UPPER CASE AND SHIFT LOCK). For example, CTRL plus J produces ASCII ØA, Line Feed.

Some control characters overlap some of the function keys. For example, the LINE FEED function key is the same as CTRL/J and CTRL/*. Certain ASCII codes are output by two different control keys; for example: Ø8, Backspace is produced by CTRL/H and CTRL/(.

5.4.2 Transmission and Display of Control Keys

control procedure causes the keyboard to generate a 7-bit code. In e Terminal Mode of SOLOS, with the LOCAL key off, this code is sent to the Serial Communications Interface (SCI) for transmission. Certain function keys which have an eighth bit set by the Sol keyboard are sent directly to the VDM for display and consequently are not transmitted.

When the Sol is in the Terminal Mode and the LOCAL key is activated, the outward bound serial data is connected back to the Serial Interface input and displayed on the video monitor as the corresponding control symbol. In the Command Mode, control characters are not displayed.

The display of control characters can be surpressed by an internal switch setting. (Refer to Appendix 3, "Internal Controls.")

In Table A4-1, the three columns under the heading "CONTROL" give the symbol that can be generated by the corresponding control sequence, the code generated by the keyboard, and the ASCII function.

If you wish to see the control characters displayed at this time, follow Table A4-1 to generate the indicated control characters. Use the following exercise, if helpful.

5.4.3 Familiarization with the Control Symbols

This optional exercise requires that a SOLOS Personality Module be installed in the Sol, and that the control character display option be enabled (a switch setting described in Appendix 3, "Internal Controls").

1) Turn on the Sol and monitor.

2) Enter Terminal Mode as follows:

   a) Set UPPER CASE.

   b) Type TERM and press RETURN.

      "TERM" will appear on the screen as you type, and the cursor will disappear when you press the RETURN key. The Sol is in Terminal Mode.

3) Set for local operation by pressing the LOCAL key so that the indicator light goes on. Local operation enables the keyboard entries to be seen on the display screen. UPPER CASE may be in either state.

4) Test the symbol generation of the Terminal Mode by pressing each character key simultaneously with CTRL. As each is pressed, the control character symbol shown in Table A4-1 should appear on the screen.

5.5 ARITHMETIC PAD KEYS

The keys on the numeric pad and the hex codes generated by them are the same as the arithmetic keys on the main keyboard. They are repeated in the standard calculator pad arrangement for convenience in entering large amounts of numerical data.

Except for the division symbol key (÷), pressing these keys causes the Sol to produce the symbol labeled on the key. The division symbol key enters a forward slash (/) character. UPPER CASE, SHIFT, AND CTRL do not affect these keys.

SECTION 6

UNPACKING AND INSTALLATION

Refer to this section when you need to install or move and reconnect a Sol system. This section contains instructions and procedures.

If your dealer has installed and checked out your system or if you are going to use an already installed system, you can skip this section.

6.1 RECEIVING INSPECTION

1) Examine the shipping container(s) for signs of possible damage to the contents during transit.

2) Carefully open the container and take out the components. Save the shipping materials for use in returning your Sol unit to your dealer in case he needs to ship it to the factory.

3) Inspect the contents for damage. If anything is damaged, please contact the carrier and your dealer immediately. Describe to them the condition of both the container and its contents so that they can take appropriate action.

4) Check the contents against Table 6-1, "Sol Systems Component Lists," to make sure you have received everything. Select the list for the system you ordered. If you special-ordered your Sol, you may be able to add or subtract items from the most similar list; otherwise, obtain a list from your dealer. If anything is missing, please contact your dealer at once so that he can take appropriate action. Refer to items by part number and name.

5) Fill out the warranty cards and mail them. Be sure to fill out each and every warranty card completely. There should be one card for each product you have received (other than software). When registered with Processor Technology, the warranty cards establish you as the owner of the product, and allow Processor Technology to send you important information.

6) When you have unpacked and checked your Sol system, continue reading this section.

6.2 INSTALLATION OF S-100 MODULES

(Refer to Fig. 6-1, "Installing An S-100 Module.")

WARNING

Before reading further, please the warning at Section 7.2.

BEFORE PERFORMING ANY SERVICE, DISCONNECT THE AC LINECORD OF THE Sol FROM THE REAR PANEL.

Before installing your S-100 modules, refer to the appropriate user's manual for instructions and precautions. Set the address switches on the memory modules according to the memory manual module before installing the modules.

S-100 modules(s) are installed in the expansion chassis (the cardcage for the backplane located in left rear three quarters of Sol as viewed from the front).

S-100 modules are installed in the Sol as follows:

(You may install any module in any of the five card slots, except for the Helios Controller modules; see the Helios II User's Manual.)

1) Be sure that AC power to the Sol is turned off. The square red button on the rear panel should be out. Disconnect the AC linecord from the rear panel.

2) On the Sol rear panel, unscrew about two turns the two knurled thumbscrews that hold the top cover in place.

3) Remove the top cover of the Sol by lifting it carefully from the rear and unhooking it from the back edge of the keyboard cover. Set it aside.

4) Being careful not to mar the walnut sides, swing the Sol keyboard cover up, unhook it from the front edge of the main chassis, and set it aside.

5) Touch the Sol chassis to discharge any static electricity from your body.

6) If you are going to use the Sol as a terminal, or connect the serial port to a SolPrinter, refer to the Sol Technical Manual and the SolPrinter User's Manual or consult your dealer.

7) If you want to change the polarity of the display or make the cursor blink, refer to the appendix "Internal Controls."

8) With the component side up, insert the edge connector side of module in the card guides. (Refer to Fig. 6-1, "Installing An S-100 Module.")

9) Carefully slide the module in until its edge connector is fully seated in the backplane connector. (The backplane is the vertical circuit board on the front side of the expansion chassis.)

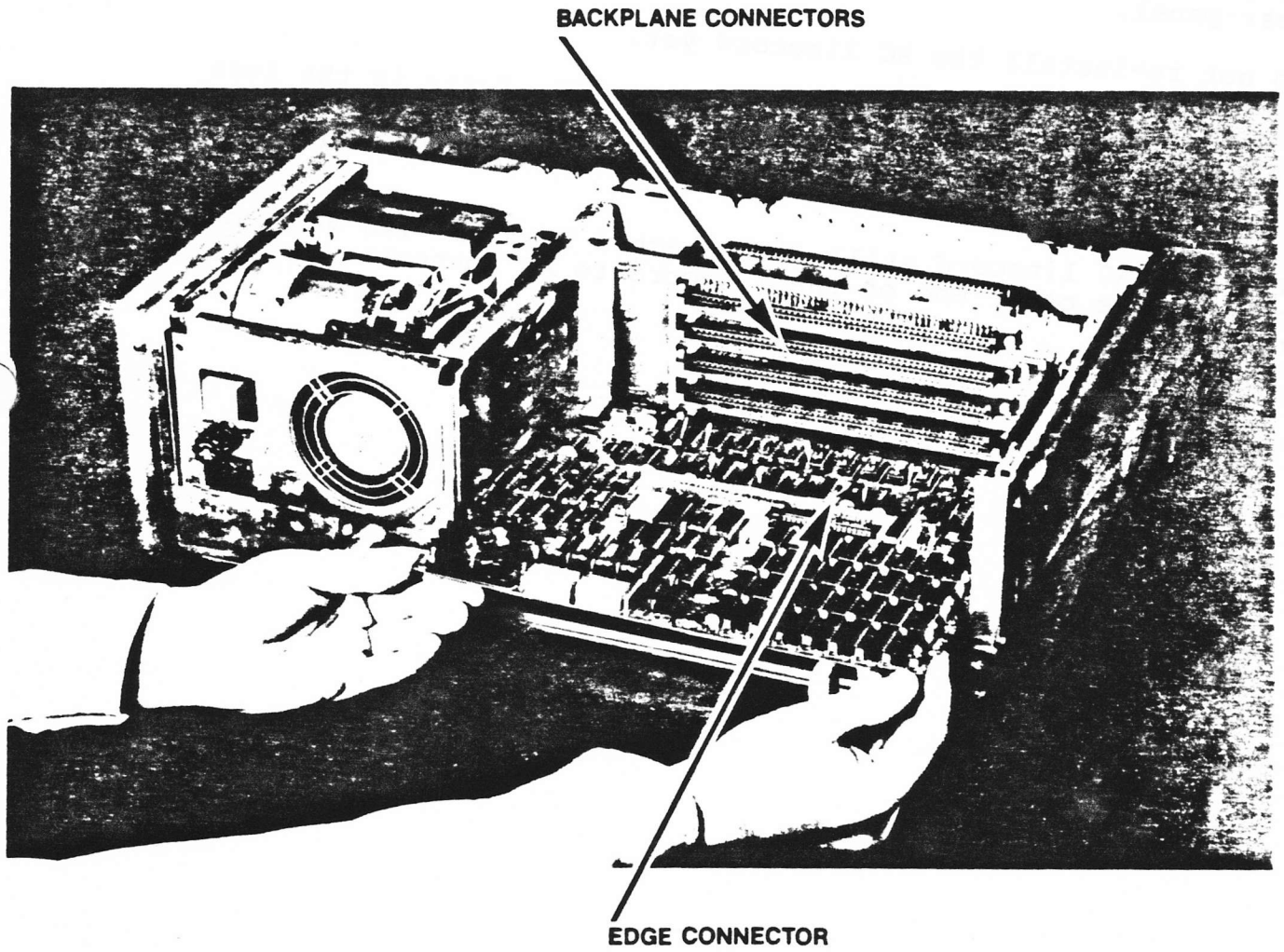


Fig. 6-1. Installing An S-100 Module

6.3 BUTTONING UP THE Sol AND INSTALLING THE FUSE

(Refer to Fig. 6-2., "Sol System Interconnections.")

1) Replace the covers on the Sol:

   Hook the keyboard cover under the front edge of the main chassis and lower it over the keyboard.

   Hook the top cover over the back edge of the keyboard cover and lower the top cover down into place over the rear of the chassis.

   Re-install the two knurled thumbscrews fastening the rear panel.

   Do not re-install the AC linecord yet.

2) Insert one of the two supplied 3.2A Slo-Blo fuses in the fuse cap, push the assembled cap-and-fuse into the fuse holder in the rear panel of the Sol, and turn the cap one-quarter turn clockwise.

3) With the AC linecord still disconnected from the 110 VAC outlet, connect the other end of the linecord to AC connector on Sol rear panel.

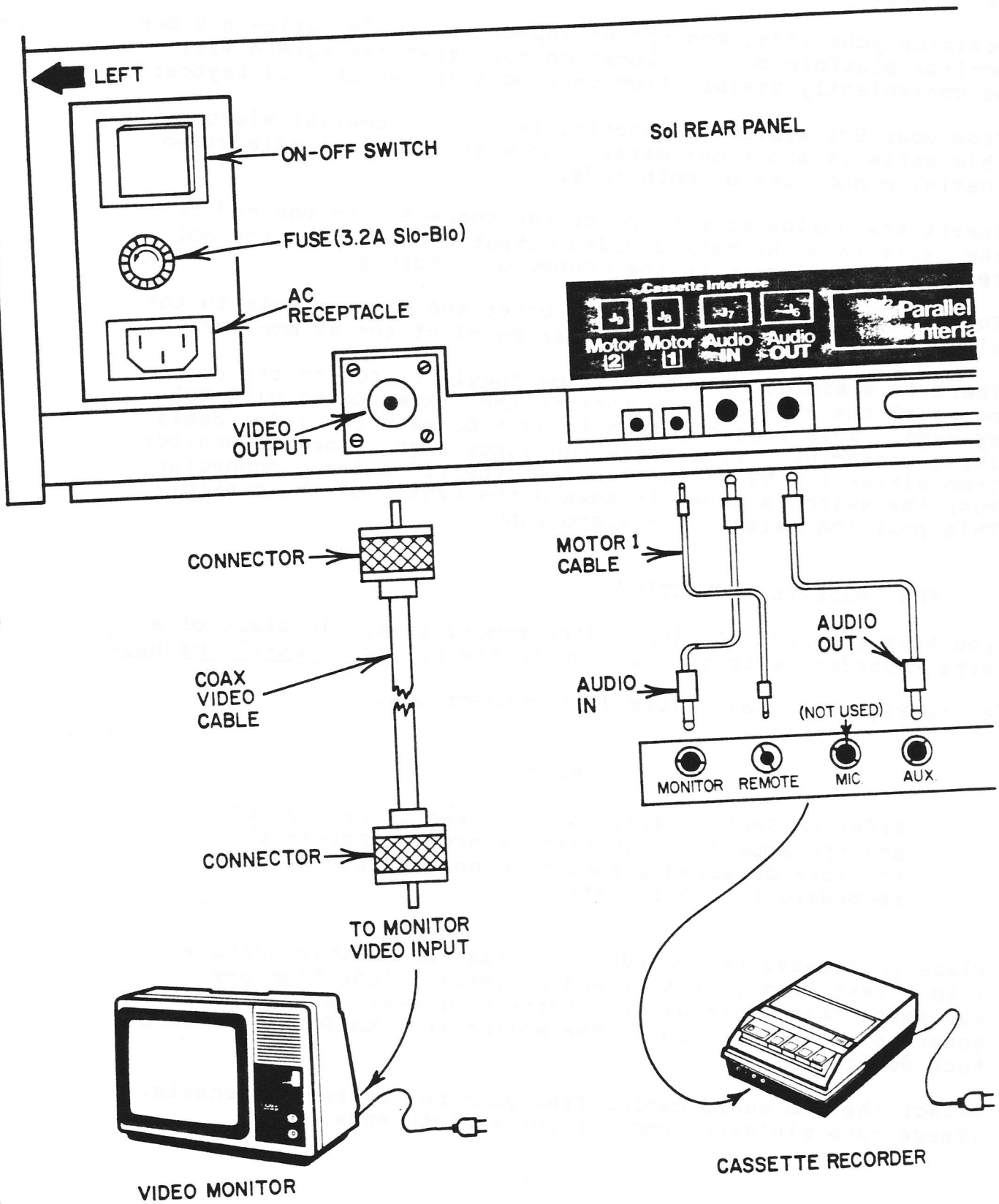


Fig. 6-2. Sol System Interconnections

6.4 CONNECTING THE VIDEO

Refer to Fig. 6-2, "Sol System Interconnections.")

1) Position your video monitor on top of the Sol's built-in video monitor platform or in a location such that the screen will be conveniently visible from your position at the Sol keyboard.

2) From your Sol system components, select the coaxial video cable. This cable is about one meter long with identical male video coaxial connectors on both ends.

3) Insert the hollow male prong of the connector on one end of the cable into the mating video output connector on the Sol rear panel and screw on the connector securely.

4) In the same manner, connect the other end of the cable to the video input connector on the rear panel of the TV monitor.

5) There is a miniature two-position toggle switch on the rear panel of the PT-872 video monitor that is supplied with Sol systems. This toggle switch is just beneath the video cable input connector. It selects the input signal for the monitor from either the TV antenna or from the video input connector. Push the switch's lever in toward the center of the monitor. This position selects the video input.

6.5 THE CASSETTE CONNECTION

(If you have a Helios II floppy disk memory system in place of a cassette recorder, skip this and go to the Helios II User's Manual.)

(Refer to Fig. 6-2, Sol System Interconnections.)

NOTE

Refer to Section 4.2, "Audio Cassette Recorders" and the appendix "Cassette Recorder Supplement" for more detailed information on cassette recorders in a Sol system.

1) Place your cassette recorder on a flat accessible surface 1 to 3 feet from your Sol, and at least a foot from any equipment that contains transformers or devices that may generate magnetic fields. The Helios and the Sol do contain such devices.

2) Select the two audio cables from your Sol system components. (These have miniature phone plugs at both ends.)

3) Let the first audio cable be the "Audio IN" cable. ("IN" is with reference to the computer.) Plug one end of the Audio IN cable into Audio IN jack (J7) on Sol rear panel. Plug the other end into MONITOR or EARPHONE jack on the recorder connector panel.

4) Let the second audio cable be the "Audio OUT" cable. Plug one end of the Audio OUT Cable into the Audio OUT jack (J6) on the Sol's rear panel. Plug the other end into the AUX IN (AUXILIARY IN) jack on the recorder.

NOTE

The use of the MICROPHONE input is not recommended.

5) Select from your Sol system components the motor control cable. (This is thinner than the audio cables, fitted with subminiature phone plugs at both ends.) Let this cable be the "Motor 1" cable. (A second "Motor 2" cable is needed if your system uses two recorders. If you are installing two recorders, refer to the appendix, "Cassette Recorder Supplement.")

6) Plug one end of Motor 1 cable into the Motor 1 jack (J8) on Sol rear panel.

7) Plug the other end into the REMOTE jack on recorder.

5.6 CONNECTING THE AC POWER

1) Be sure the Sol covers are in place.

2) Be sure that the AC power switches for the Sol, the video monitor and the recorder are OFF.

3) Connect their AC linecords to their rear panel receptacles.

4) Plug the linecords into convenient AC outlets.

Your system is now installed and ready. Return to Section 1 of this manual, or if you already read Section 1, please go to Section 2, "At the Controls."

Table 6-1. Sol System Component Lists

ORDER NO.	DESCRIPTION	QUANTITY	(✓)
400410	Sol-20 Terminal Computer		
107000	SOLOS Personality Module	1	( )
727019	Extended Cassette BASIC (cassette)	1	( )
730024	Sol Terminal Computer User's Manual	1	( )
727034	SOLOS/CUTER User's Manual	1	( )
727018	Extended Cassette BASIC User's Manual	1	( )
723018	Fuse, 3.2A Slo-Blo	2	( )
part of 724007	Cap, Fuse Holder	1	( )
718001	Cord, AC Power, 3-Wire	1	( )
400500	Sol-20/16		
400410	Sol-20 Terminal Computer (See list above)	1	( )
214010	16KRA-1 Memory Module	1	( )
730026	32KRA-1 User's Manual	1	( )
730034	"16KRA-1 Product Description"	1	( )
400600	Sol-20/32		
400410	Sol-20 Terminal Computer (See list above)	1	( )
214020	32KRA-1 Memory Module	1	( )
730026	32KRA-1 User's Manual	1	( )
400700	Sol System I-A		
400500	Sol-20/16 Terminal Computer (See list above)	1	( )
722016	PT-872 video monitor (with manual)	1	( )
718005 or 101034	Video Cable Assembly	1	( )
722019	RQ-413A Cassette Recorder	1	( )
718006 or 101041	Audio Cable Assembly	2	( )
718007 or 101042	Motor Control Cable Assembly	1	( )

Table 6-1. Sol System Packing Lists (continued)

(ER #	DESCRIPTION	QUANTITY	(✓)
0800	Sol System II-A		
	(Same as I-A, except that the Sol model is:)		
400500	Sol-20/32 Terminal Computer (See list above)	1	( )
400310	Sol System III-A		
400410	Sol-20 Terminal Computer (See list above)		
107015	BOOTLOAD Personality Module	1	( )
-	Memory module(s) containing 65,536 bytes of memory		
300000	Helios II, Model 2 (See list in Helios II User's Manual)		
722016	PT-872 video monitor (with manual)	1	( )
718005 or	Video Cable	1	( )
101034	(alternate video cable)		
-	appropriate memory module manual(s)		
730031	Helios II User's Manual	1	( )
731029	PTDOS User's Manual	1	( )
7037	Extended Disk BASIC User's Manual (instead of Cassette BASIC)	1	( )
27010	Trek-80 User's Manual	1	( )
400311	Sol System III-B		
	(Same as III-A without video video monitor and cable)		
400350	Sol System IV-A		
	(Same as III-A except that Helios II is a double dual drive model:)		
304000	Helios II, Model 4	1	( )
400351	Sol System IV-B		
	(Same as IV-A without video monitor and cable)		

SECTION 7

TLC FOR YOUR Sol

7.1 SERVICE AND ORDERING PARTS

A convenient and economical way to arrange for preventative maintenance and to be prepared for eventual adjustments as well as unexpected problems, is to purchase a maintenance contract if your dealer has it available to you.

Do not replace devices thought to be defective in a Sol that is still under warranty. To do so may void the warranty. Refer the problem to your factory-authorized dealer, even if the warranty has expired.

Order replacement parts or additional equipment and manuals from your factory-authorized dealer. Be sure to include the Processor Technology order number or part number, the quantity you want to order and a complete description of the item. For example: one (1) Sol Technical Manual, PT-730038. Your dealer has a list of part numbers. He may have a stock of some replacement parts on hand.

7.2 CARE AND EXTERIOR CLEANING OF YOUR Sol

It is recommended that you turn off your Sol if you are not using it for an extended period of time. This practice should help extend the life of certain components. The Sol needs no warmup time.

When required, clean the keyboard and exterior covers of your Sol with a cloth dampened in a mild detergent solution. Be sure to turn off the power first and disconnect the AC linecord from the rear panel.

Once a year, a thin coating of linseed oil applied with a clean cloth will preserve and beautify those famous walnut side panels.

WARNING

Do not remove the cover(s) from the Sol. For your protection the Sol is designed so that the AC linecord must be disconnected before the top cover can be removed. Never reconnect the linecord when the top cover is off.

If you encounter a problem, first reread those parts of this manual which pertain to the operation you were attempting when the problem occurred. If you are still unable to solve the problem or if you have subsequent hardware or software failures, ask help from qualified technical personnel.

WARNING (Continued)

If you yourself are the qualified person finally responsible for the hardware in the system, you may use the following sections to trouble-shoot the problem. If you are still unable to solve the problem, ask your dealer for help. Service on all Processor Technology products, in or out of warranty, is the reponsiblity of the factory-authorized dealer.

The following sections are to be used only by qualified personnel as basic aid in determining whether a problem warrants calling upon the factory-authorized dealer for service.

BEFORE PERFORMING ANY SERVICE, DISCONNECT THE AC LINECORD OF THE Sol FROM THE REAR PANEL.

7.3 TROUBLE-SHOOTING the Sol INSTALLATION

7.3.1 Checking and Installing the Fuse

(See Figure 6-2, "Sol System Interconnections.")

1) Is the fan running? Put your hand at the fan output if you cannot hear the fan running. If the fan is not running, is the AC linecord plugged into a power receptacle? If it is plugged in, the fuse should be checked. The Sol is protected by a 3.2 amp Slo-Blo fuse housed on the rear panel. Check the fuse as follows:

   a) Turn Sol's AC power switch OFF.

   b) Disconnect the AC linecord from the rear panel.

   c) Turn the fuse holder cap one quarter turn counterclockwise; pull straight out and remove the fuse from the cap.

   d) Inspect the finer fuse element that should be connected to the end of the thicker spiral element; it looks like a small "bobby pin", and it should be in one piece connected all the way to the other end of the fuse. If it is broken in the middle, it is blown. A spare fuse is shipped with each Sol.

   e) To install a fuse, insert the fuse into the cap, push the cap containing the fuse into the fuse holder and turn one quarter turn clockwise.

7.3.2 Removing The Covers Prior to Checkout of the Interior

1) Be sure that AC power to the Sol is turned off. The square red button on the rear panel should be out. The fan should be stopped.

   Disconnect the AC linecord from the Sol rear panel.

2) On the Sol rear panel, unscrew the two knurled thumbscrews that hold the top cover in place (about two turns).

3) Remove the top cover of the Sol by lifting it carefully from the rear and unhooking it from the back edge of the keyboard cover. Set it aside.

4) Being careful not to mar the walnut sides, swing the Sol keyboard cover up, unhook it from the front edge of the main chassis, and set it aside.

5) Proceed to the next section.

7.3.3 Checkout of Cables, Connectors, Sol-PC and Personality Module

1) Touch the Sol chassis to discharge any static electricity from your body.

2) Connectors and Cables. The mechanical contacts of connectors must be in proper position. In order to provide versatility and serviceability, a number of connectors are used in the system. Be sure that they are all inserted properly. Secure any loose cable connectors. On the Sol-PC there is a coax connector at the video display circuitry and a ribbon cable connecting the keyboard to the Sol-PC.

   Helios Cables. If you are using a Helios Disk System, be sure that when the cover is placed on the Sol, the cables are not pulled from their sockets.

3) S-100 Cards. Pull the cards back from the backplane and reseat them.

4) Personality Module. Grasping its handle, remove and reseat the personality module located on the rear of the main printed circuit board. (Refer to Fig. 2-1, "Sol Rear Panel.")

7.3.4 Buttoning the Sol Back Up

o replace the covers on the Sol:

1) Hook the keyboard cover under the front edge of the main chassis and lower it over the keyboard.

2) Hook the top cover over the back edge of the keyboard cover and lower the top cover down into place over the rear of the chassis.

3) Re-install the two knurled thumbscrews fastening the cover to the chassis.

4) Reconnect the AC linecord to the rear panel.

7.4 TROUBLE-SHOOTING THE VIDEO

1) Turn ON the AC power switches of the Sol and video monitor.

2) If the monitor display raster is out of sync (a black horizontal bar moves slowly down screen, numerous black lines cut across the raster, or both), adjust monitor vertical and horizontal hold controls for a stable raster.

3) You should see the SOLOS Command mode prompt character followed by thu cursor (>■) in the upper left corner of the screen. If you don't, recheck the video cable connection as in Section 6, "Unpacking And Installation." If still no cursor, go to 7.3.2, "Removing The Covers" and 7.3.3, the checkout of cables, etc.

If control characters do not appear in the SOLOS Terminal mode, check the setting of the internal switch that controls the display of control characters. (See Appendix 3, "Internal Controls".)

7.5 TROUBLE-SHOOTING CASSETTE OPERATIONS

NOTE

The tape head must be clean in order to read or write a tape reliably.

7.5.1 Problems Encountered When Loading

1) If the tape moves while the cassette is under SOLOS control, there is a malfunction in the remote control circuitry or cabling. The recorder is under SOLOS control when MODE SELECT has been pressed to reset any tape commands. With the Sol power OFF, there should be no continuity between the MOTOR 1 jack on the Sol and the REMOTE plug on the cassette. Check to see that the Motor cable for each recorder is pushed in until you feel the stop in the jack (the detent position for the jack; this is not necessarily all of the way in).

2) With certain cassette recorders or cassettes there may be a misreading of the tape when the splice joining the leader to the tape passes the tape head. In this case an ERROR message will appear and the tape will stop. To resume tape loading, position the tape past the leader and retype the command used to load the program.

3) If you continue to have difficulty in loading, check the recorder controls for proper settings and make sure you have followed all appropriate instructions and operating tips in Section 4.2, "Audio Cassette Recorders" and 4.3, "Getting Into BASIC." Try different cassette recorder volume settings until a reliable setting is found.

4) Usually Processor Technology cassettes have the same program recorded more than once on the same cassette to provide against accidental damage or erasure. If you have difficulty loading a program, try the same procedure with the redundant recording.

5) Check all cassette interconnect cables for intermittent connections and shorts. Try substituting cables to detect defective cables.

7.5.2 Possible Faulty Tape?

1) Note the exact tape counter reading at the time of the read error.

2) Rewind the tape and try to read the same part of the tape in which the error occurred. If there is no read error at the same point, the error was not recorded on the tape. If there is, the error was recorded on the tape.

3) Rewind the tape and record a file on the same part of the tape in which the read error occurred. Then read this file. If there is no read error, the original error was generated during the initial recording process. If a read error occurs at the same point, the cassette is faulty.

APPENDIX 1

PRODUCTS AVAILABLE from Processor Technology

A1.1 EQUIPMENT

MEMORY MODULES

Dynamic Read/Write Random Access Memory modules are available from Processor Technology in the following models: (The first two digits give the memory capacity in kilobytes.)

16KRA-1
32KRA-1
48KRA-1
64KRA-1

GENERAL PURPOSE MEMORY MODULE (GPM)

The GPM module is used to store any frequently-accessed programs as firmware in its 10,240 bytes of ROM. The programs thus stored can be ready to accept commands as soon as power is applied to the Sol. Included with this subsystem is 1,024 bytes of RAM for use as scratch-pad memory.

DISKETTE MEMORY SYSTEMS

The Helios II dual diskette drive systems include one or two dual drives, complete with power supply and indicator panel in a single attractive cabinet. An S-1ØØ bus compatible controller is provided in the system. Software includes PTDOS operating system and Extended Disk BASIC. PTDOS operates up to four dual drives. PTDOS contains Extended Disk BASIC, ASSM, DEBUG, EDIT, and TREK-8Ø, plus many other powerful software tools.

The single dual drive model can be upgraded to a double dual drive model. Both models are available in 22Ø/24Ø VAC 5Ø Hz versions.

PRINTERS

SolPrinter 2 is a word processing printer by a Diablo (a Xerox company.) It has a changeable metal print wheel. The interface to the Sol parallel port and power supply is included.

SolPrinter 2-E is an impact printer like the SolPrinter 2 but with a plastic print wheel.

Forms tractors are available for the SolPrinter 2 and 2-E.

SolPrinter 3 is a 200 characters per second dot matrix printer with an RS-232 serial interface. This SolPrinter includes a power supply and ɔrms tractor.

The "engineering geniuses" at Processor Technology are continually developing new products for the Sol system. Ask your dealer for the latest releases.

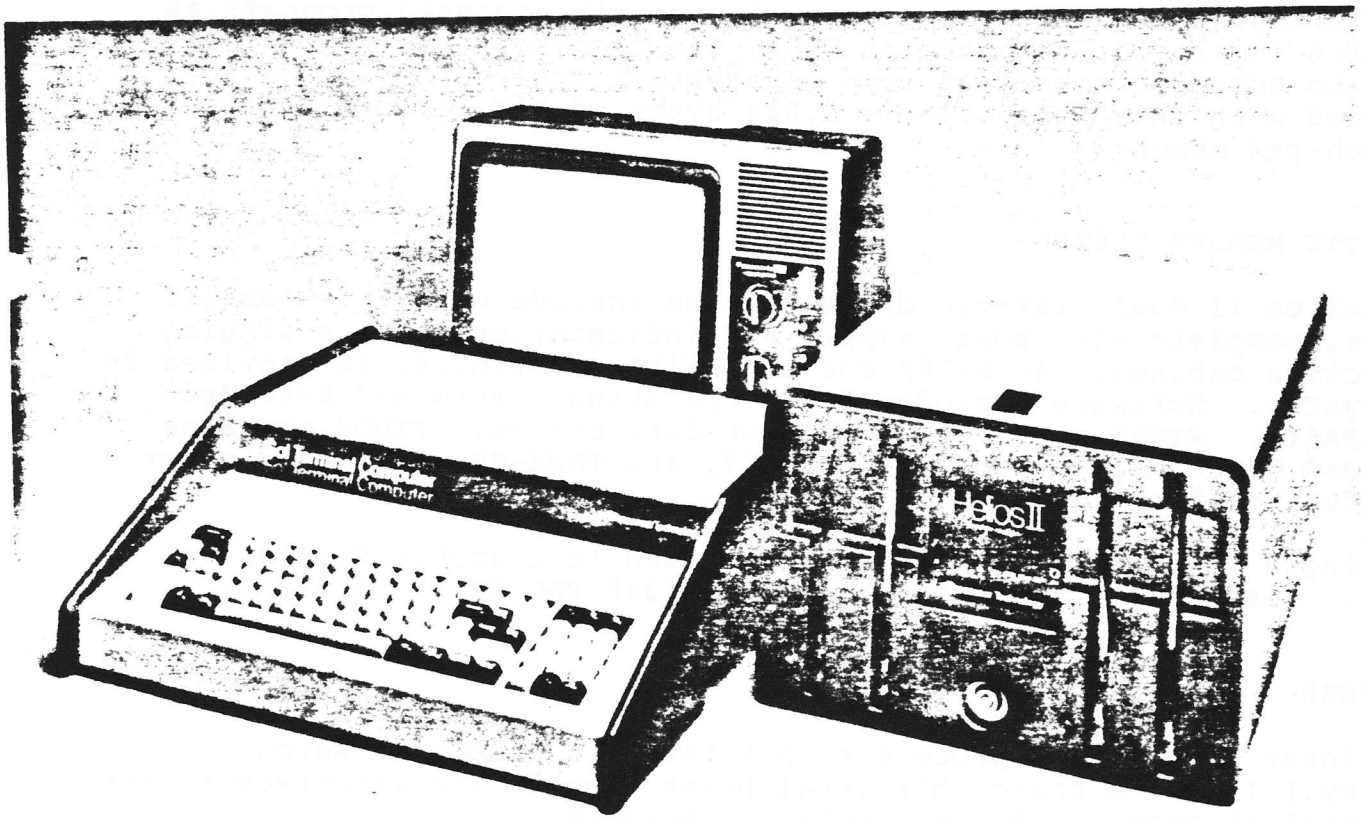


Fig. A1-1. Sol System IV-A

A1.2 SOFTWARE AVAILABLE FROM Processor Technology

PROGRAM DEVELOPMENT SOFTWARE ON CASSETTE

Processor Technology cassette software is designed to run either on a Sol Terminal Computer, with SOLOS, or on another 8080 computer using CUTER software and CUTS format cassette tape.

EDIT

EDIT is a line-oriented text editor that allows the creation or modification of ASCII files such as source files coded in BASIC or assembly language. EDIT allows editing on character, string, line, and page levels; at any of these levels, additions, insertions, substitutions, and deletions of text may be made. Additionally, EDIT offers the option to retain a command string as a macro and execute it repeatedly.

EDIT requires approximately 4K bytes of memory.

ASSM

ASSM is an assembler that translates a symbolic 8080 assembly language program ("source code") into the binary instructions required by the computer to execute the program.

The assembler itself occupies almost 8K bytes of memory; an additional 2K of memory is required for the symbol table. Two cassette recorders are also required.

Also included on the cassette are two other programs, PACK and UNPAC. These programs convert a cassette file from either of the two SOLOS/CUTER file formats (single-block and multiple block) to the other.

DEBUG

DEBUG is an aid for debugging a machine language program. DEBUG permits you to set as many as fifteen "breakpoints" in a program. When that program is executed under control of DEBUG, execution will be interrupted at each breakpoint address so that CPU registers, flags, and specified memory locations may be examined and modified.

DEBUG requires 8K bytes of memory.

ALS-8

ne ALS-8 assembly language development system is an operating system that facilitates the writing, editing, assembling and debugging of your own programs. It includes a resident assembler, simulator, and text editor.

The ALS-8 program requires 8K bytes of memory plus 4K for the symbol tables and system global area. ALS-8 is also available in a ROM on a GPM module (see A1.1). In that form, the system requires a minimum of 2K of RAM.

Software#1

Software #1 is a program development system, more limited than ALS-8, but requiring less memory. Included in the package are an executive to handle memory files, an assembler, and a line-oriented editor.

Software #1 requires at least 6K bytes of memory.

Extended Cassette BASIC

Extended Cassette BASIC is an adaptation of BASIC, a straightforward language combining interactive features and mathematical capabilities. BASIC includes an interactive editor and an interpreter.

xtended Cassette BASIC requires a minimum of 16K bytes of memory. The recommended amount of memory is 24K.

PILOT

PILOT is an interpretive language designed for writing interactive programs. It is suitable for writing educational applications programs, other interactive programs and games.

PILOT requires 16K bytes of memory.

8080 FOCAL

FOCAL is an adaptation of the original FOCAL, a math language written for the PDP-8 mini-computer. "FOCAL" and "PDP-8" are registered trademarks of Digital Equipment Corporation. Many thousands of FOCAL programs exist and can now run in the Sol. A disk version of FOCAL is included on the PTDOS system disk that comes with a Helios II system.

FOCAL requires 10K bytes of memory.

VIDEO GAMES ON CASSETTE

GAMEPAC 1 and GAMEPAC 2 are two anthologies of games including:

Target - an arcade-type "shooting" game.
Hangman - the traditional word game.
QUBIC - a three-dimensional version of TIC-TAC-TOE.
("QUBIC" is a registered trademark of Parker Bros.)

Each game requires 4K bytes of memory.

TREK-80

TREK-80 is a space war game based on the NBC television series Star Trek. This program simulates a real time war with the Klingons. (You can warp through hyperspace and fire phasers, photon torpedos, or experimental rays. If you "just can't make it," you can self-destruct.)

TREK-80 requires 8K bytes of memory.

8080 CHESS

8080 CHESS is the ancient game of pure skill and fathomless variety, adapted to run on the Sol. 8080 CHESS turns your computer into a worthy opponent. A variable difficulty control matches your skill level.

8080 CHESS requires 16K bytes of memory.

FLOPPY DISK SOFTWARE

Extended Disk BASIC

Extended Disk BASIC has all the features of Extended Cassette BASIC and also the advantages of the Helios II floppy disk and PTDOS.

Many dealers offer Optional Precision BASIC, allowing the user to request a version of Extended Disk BASIC with 6, 8, 10, 12, 14, or 16 digits of precision. (Standard BASIC has 8-digit precision.)

Extended Disk BASIC requires 32K bytes of memory, and is supplied on the PTDOS system disk.

Extended Disk FORTRAN

Extended Disk FORTRAN is both a subset and a superset of ANSI standard FORTRAN, a widely used algebraic language. Extended Disk FORTRAN is written to take advantage of the Helios II Disk Memory System and PTDOS.

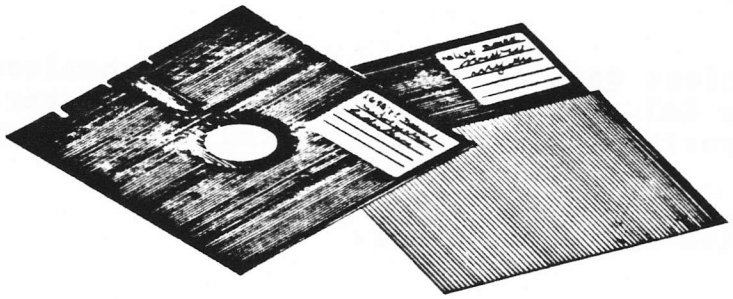
Extended Disk FORTRAN requires 32K bytes of memory.

WordWizard

WordWizard is an electronic typing program that turns your Sol, Helios, and SolPrinter into a powerful office or home tool for composing, correcting, editing and updating letters and other literature. WordWizard can print a file while you are editing a second file.

WordWizard requires 48K bytes of memory.

New software for the Sol is constantly being developed at Processor Technology. Ask your dealer about the latest releases.



APPENDIX 2

CASSETTE RECORDER SUPPLEMENT

A2.1 SELECTING A CASSETTE RECORDER

Not all audio cassette recorders are suitable for data storage use with the Sol. Two models tested and approved by Processor Technology for such use are the Panasonic RQ-413AS and Realistic CTR-21. (Some users report unsuccessful results with the Panasonic RQ-309 and the J. C. Penney recorder, Catalog #851-0018.) Should you wish to use a different recorder than those approved by Processor Technology, it should have the following features:

Auxiliary Input

Though the Sol can be configured for use with the microphone input, such configuration is no longer recommended.

Digital (Tape) Counter

The counter is needed to locate files on the tape.

Tone Control

The existence of a tone control is one indication of high quality electronics.

Monitor Output in Record Mode

Recorders on which the "monitor" jack is labeled MONITOR usually provide a monitor output in the Record Mode. If the jack is labeled EAR or EAR PHONE, the recorder usually does not provide a monitor output in the record mode.

To determine whether the recorder provides a monitor output in the Record mode:

1) Install a blank tape.

2) Plug a headset or earphone into the MONITOR jack.

3) Plug a microphone into the MICROPHONE jack.

4) Set the recorder controls to record.

5) Speak into microphone while listening with the earphone. If you hear yourself through the earphone, the recorder does provide a monitor output in the record mode.

Even if a recorder has the preceding features, there is no guarantee it will work properly with the Sol. Recorders vary greatly in the quality of their electronics. When selecting a "non-approved" recorder, it is recommended that you test it before purchase, if possible, with a long file. Test it in both the record mode (Write) using the SAVE command and in the playback mode (Read) using the GET or XEQ commands. If the recorder is unsatisfactory, you will either:

1) Get an error message in the read mode.

2) Find the differences, upon playback, in what you recorded in the write mode.

3) Both of the above.

A2.2 INTERCONNECT REQUIREMENTS FOR TWO RECORDERS

Your Sol is capable of controlling one or two recorders. (The interconnect requirements for one recorder were previously covered in Section 4.3, "Getting into BASIC.") Since the Sol has only one audio input and one audio output jack, the interconnect requirements for two recorders are somewhat different than for one.

n the case of two recorders, Unit 1 and 2 must be specified in the SAVE command in order to select the desired recorder. A default selects Unit 1.

Reading from One Recorder and Writing to the Second Recorder

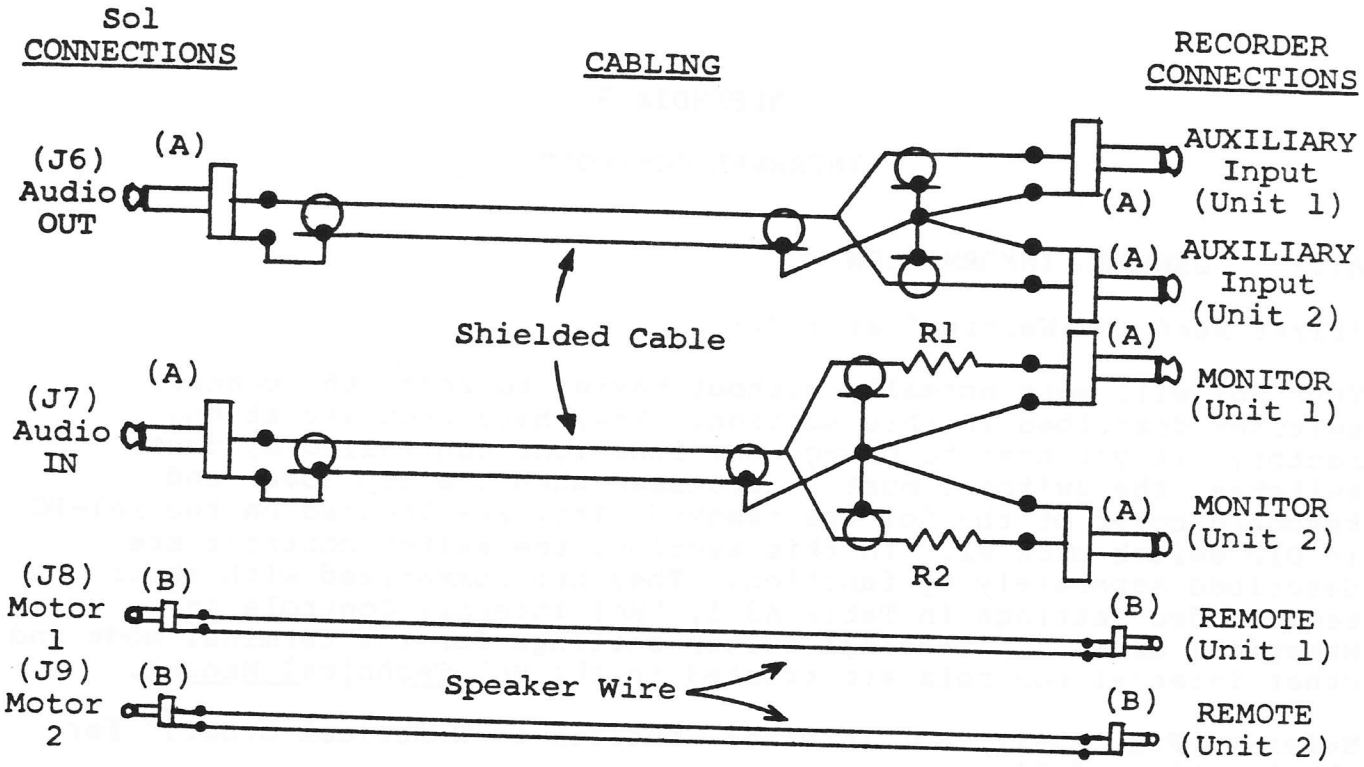
1) Let recorder 1 be the unit read from; connect the Sol's Audio IN cable to the recorder's MONITOR output. Connect MOTOR 1 cable to the recorder's REMOTE jack.

2) Let recorder 2 be the unit written to; connect the Sol's Audio OUT cable to the recorder's AUX input. Connect MOTOR 2 cable to the recorder's REMOTE JACK.

Reading and Writing to Both Recorders

For this setup, you will need two "Y" adapters, one to feed the single Sol audio output to the AUXILIARY input of two recorders and the other to feed the MONITOR output of two recorders to the single Sol audio input. (If you intend to use the Audio In and Out cables described in Section 6.5, "The Cassette Connection," miniature phone jack-to-two miniature phone plug adapters are required.)

Because the recorder outputs should not be connected together directly, it is recommended that you wire a 1000 ohm resistor in series between the MONITOR adapter and the MONITOR output of each recorder.



(A) Miniature Phone Plug
(B) Subminiature Phone Plug

R1 = R2 = 1000 ohms, ¼ watt

Figure Connecting Sol to two cassette recorders.

APPENDIX 3

INTERNAL CONTROLS

A3.1 GENERAL INFORMATION

(First Read the Warning" at 7.2.)

Your Sol will work normally without having to reset the control switches described in this section. They have been set at the factory. If you need to change the functions controlled by these switches, the switches must be accessed when the top cover and keyboard cover of the Sol are removed. They are located on the Sol-PC in DIP switch pack S1. In this section, the switch controls are described separately by function. They are summarized with their recommended settings in Table A3-1, "Sol Internal Controls in Numerical Order." Internal switch settings for the terminal mode and other internal controls are treated in the Sol Technical Manual.

Refer to Fig. A3-1, "Sol Internal Controls in Numerical Order," for the location of S1.

WARNING

BEFORE PERFORMING ANY SERVICE, DISCONNECT THE AC LINECORD OF THE Sol FROM THE REAR PANEL.

A3.2 REMOVING THE Sol COVERS

To access S1, remove the Sol top cover and keyboard cover as instructed in Section 7.3, "Trouble-shooting the Sol Installation."

A3.3 VIDEO DISPLAY SWITCHES

CAUTION

Since each switch lever is extremely small, it may not be practical to use one's finger to set them. As a convenience, you might use the pointed end of a nonconductive tool shaped like a pencil. Do not use a metal object.

Video Display (POLARITY) Switch, S1-4

If you want a normal video display (white characters on a black background), set S1-4 switch to ON. If the switch is in the OFF position, black characters will be displayed on a white background (reverse video display).

Cursor Display (S1-5 and S1-6)

The cursor that appears in SOLOS as the prompt character is usually a solid, non-blinking cursor.

CAUTION

Never put S1-5 and S1-6 ON at the same time. To do so may damage your Sol.

To make it blink on and off at the rate of approximately twice per second:

1) Set the SOLID Switch (S1-6) to OFF.

2) Set the BLINK Switch (S1-5) to ON.

3) To reset the cursor to a solid display, reverse the above settings.

With both S1-5 and S1-6 in their OFF positions, there will be no cursor display.

Blanking Out the Display of Control Characters (S1-3)

Blank out the display of control characters by setting the BLANK Switch (S1-3) to ON. Any control characters generated should not appear on the screen. To display control characters, set S1-3 to OFF. (Control characters are explained in Section 5, "At the Keyboard.")

A3.4 RESTART (RST) SWITCH, S1-1

This switch permits you to restart your Sol without turning the power off. You should normally leave it in its OFF, or run, position. Set it to ON and then OFF to reset the 8080 and restart the Personality Module program. (A manual restart with this switch performs the same function as turning the power on or pressing a keyboard generated restart: UPPER CASE key with REPEAT key. See "Keyboard Restart" in Section 2.5.)

A3.5 REPLACING THE Sol COVERS

(Refer to 7.3.4, "Buttoning the Sol Back Up.")

A3.6 TERMINAL MODE SWITCH SETTINGS

Instructions for setting the Terminal mode switches are in the Sol Technical Manual. Consult your dealer if you intend to use the Terminal mode.

A3.7 PARALLEL DATA INTERFACE (PDI)

Instruction for connecting devices to the Sol's PDI are in the Sol Technical Manual. Consult your dealer for help.

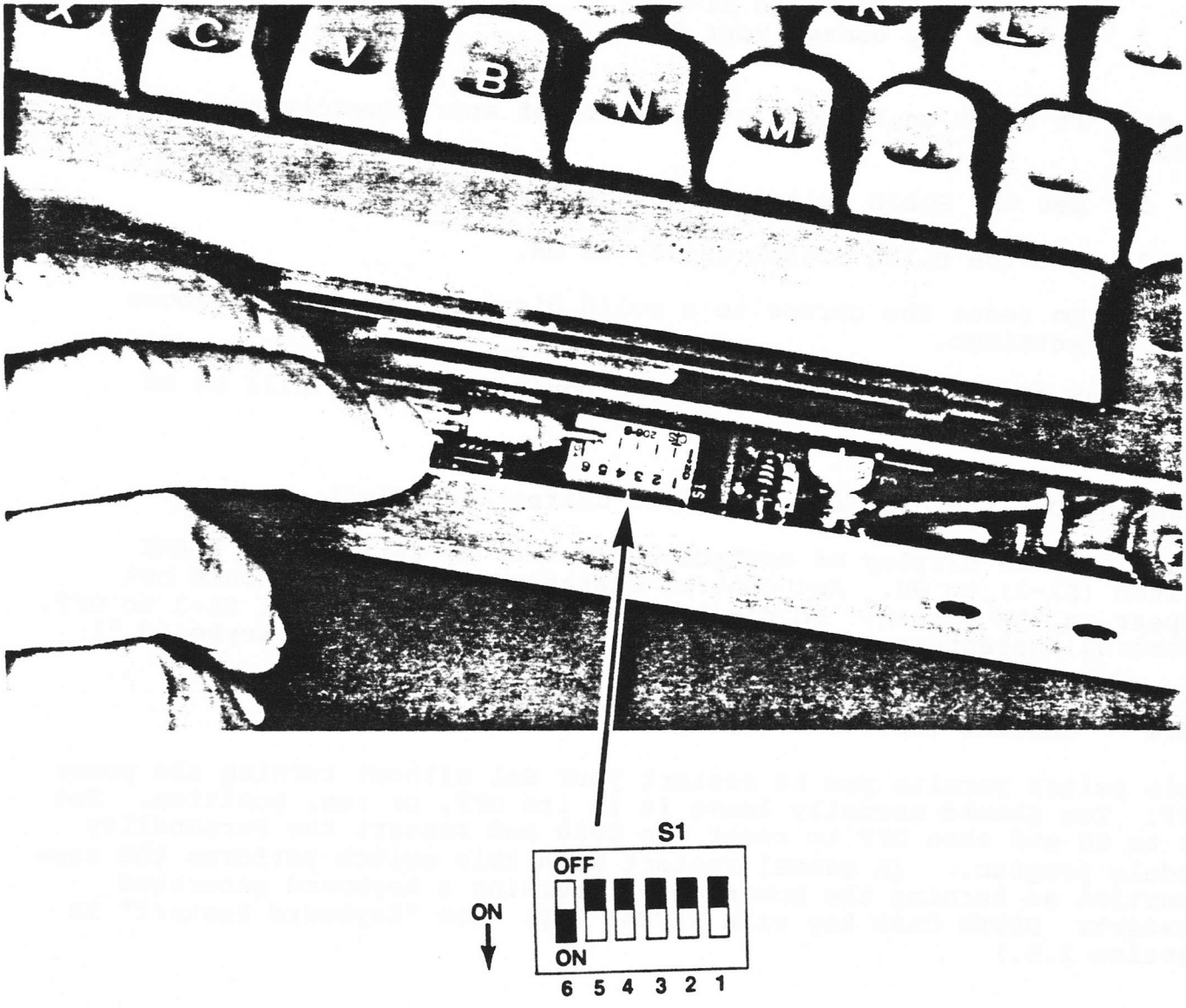


Fig. A3-1. Sol-PC Internal Control Switches (S1)

Table A3-1. Sol Internal Controls in Numerical Order

CONTROL	FUNCTION	FACTORY PRESET
RST (Restart) S1-1	Permits manual restart of Sol without turning power off. (Useful for test purposes.)	OFF
S1-2	Spare	OFF
BLANK CTRL S1-3	Determines whether control characters are displayed or not.	OFF
POLARITY S1-4	Selects normal (white characters on black background) or reverse video display.	OFF
CURSOR BLINK S1-5	Selects blinking cursor.	OFF
SOLID CURSOR S1-6	Selects solid cursor.	ON

APPENDIX 4

KEYBOARD TABLES

Table A4-1. Sol Keyboard Character and Control Keys

(The keys in this table are arranged as found on the keyboard, left to right, top to bottom.)

UNSHIFTED		SHIFTED		CONTROL		
KEY/ SYMBOL	HEX. CODE	KEY/ SYMBOL	HEX. CODE	SYMBOL	HEX. CODE	ASCII CONTROL FUNCTION
1	31	!	21	┌[1]	Ø1	Start of Heading (SOH)
2	32	"	22	┴	Ø2	Start of Text (STX)
3	33	#	23	┘	Ø3	End of Text (ETX)
4	34	$	24	┤	Ø4	End of Transmission (EOT)
5	35	%	25	▒	Ø5	Enquiry (ENQ)
6	36	&	26	✓	Ø6	Acknowledge (ACK)
7	37	'	27	▯	Ø7	Bell (BEL)
8	38	(	28	←	Ø8	Backspace (BS)
9	39	)	29	→	Ø9	Horizontal Tab (HT)[2]
Ø	30	Ø[3]	20[2]	□[4]	ØØ	Null (NUL)
-	2D	=	3D	←[4]	ØD	Return (CR)[2]
^	5E	~	7E	▗	1E	Record Separator (RS)
[	5B	{	7B	ESC[4]	1B	Escape (ESC)[2]
\	5C	|	7C	▖	1C	File Separator (FS)
]	5D	}	7D[5]	▝	1D	Group Separator (GS)
q	71	Q	51	◔	11	Device Control 1 (DC1)(X-ON)
w	77	W	57	┼	17	End of Transmission Block (ETB)
e	65	E	45	▒	Ø5	Enquiry (ENQ)
r	72	R	52	◑	12	Device Control 2 (DC2)(TAPE)
t	74	T	54	◔	14	Device Control 4 (DC4)
y	79	Y	59	▮	19	End of Medium (EM)
u	75	U	55	✗	15	Negative Acknowledge (NAK)

[1] SOLOS cursor left. (Symbols shown are displayed by the 6574 character generator.)

[2] Same as function keys (see Table A4-2).

[3] Space function (SP); Ø is not displayed.

[4] Not displayed in SOLOS.

[5] Alternate mode.

Table A4-1. Sol Keyboard Character and Control Keys (Continued)

UNSHIFTED KEY/ SYMBOL	HEX. CODE	SHIFTED KEY/ SYMBOL	HEX. CODE	CONTROL SYMBOL	HEX. CODE	ASCII CONTROL FUNCTION
i	69	I	49	→	Ø9	Horizontal Tab (HT) [1]
o	6F	O	4F	⊙	ØF	Shift In (SI)
p	70	P	50	⊟	1Ø	Data Link Escape (DLE)
@	40	`	60	□ [2]	ØØ	Null (NUL)
a	61	A	41	← [3]	Ø1	Start of Heading (SOH)
s	73	S	53	ⓢ	13	Device Control 3 (DC3) (X-OFF)
d	64	D	44	↘	Ø4	End of Transmission (EOT)
f	66	F	46	✓	Ø6	Acknowledge (ACK)
g	67	G	47	Ω	Ø7	Bell (BEL)
h	68	H	48	↰	Ø8	Backspace (BS)
j	6A	J	4A	≡ [2]	ØA	Line Feed (LF) [1]
k	6B	K	4B	↓	ØB	Vertical Tab (VT)
l	6C	L	4C	⏚	ØC	Form Feed (FF)
;	3B	+	2B	↓	ØB	Vertical Tab (VT)
:	3A	*	2A	≡ [2]	ØA	Line Feed (LF) [1]
DEL [4]	7F	_ [2]	5F	⊟	1F	Unit Separator (US)
z	7A	Z	5A	⸮	1A	Substitute (SUB)
x	78	X	58	⧖	18	Cancel (CAN)
c	63	C	43	⌟	Ø3	End of Text (ETX)
v	76	V	56	⎍	16	Synchronous Idle (SYN)
b	62	B	42	⊥	Ø2	Start of Text (STX)
n	6E	N	4E	⊗	ØE	Shift Out (SO)
m	6D	M	4D	← [2]	ØD	Return (CR) [1]
'	2C	<	3C	⏚	ØC	Form Feed (FF)
.	2E	>	3E	⊗	ØE	Shift Out (SO)
/	2F	?	3F	⊙	ØF	Shift In (SI)

[1] Same as function keys (see Table A4-2).

[2] Not displayed (masked off) in SOLOS.

[3] SOLOS cursor left.

[4] DEL, 7F, Delete symbol (⌫) is not displayed in SOLOS. Delete functions in Terminal Mode when shifted.

Table A4-2. Sol Keyboard Function Keys

(The keys are ordered in this table as found on the keyboard in two groups, left and right.)

KEY	HEX. CODE	SHIFTED, UNSHIFTED OR CONTROL SYMBOL DISPLAYED
ESCAPE (ESC)	1B	⊖ PTDOS Only.
BREAK	-	
TAB (HT)	Ø9	⟩ Both SOLOS and PTDOS.
CTRL	-	
SHIFT LOCK	-	
UPPER CASE	-	
SHIFT	-	
LOCAL	-	
SPACE BAR (SP)	20	
RETURN (CR)	ØD	← Masked Off in SOLOS.
LINE FEED (LF)	ØA	≡ Masked Off in SOLOS.
DELETE (DEL)	7F	⊟ CTRL Only (1F); 7F displays no symbol.[1]
LOAD	8C	⇩ Both SOLOS and PTDOS.
REPEAT	-	
CTRL	-	
SHIFT	-	
MODE SELECT	80	⟩ SOLOS Terminal Mode Prompt is a programmed display of SOLOS.
CURSOR CONTROL ↑	97	⊣ Not transmitted in SOLOS. PTDOS Only
CURSOR CONTROL ←	81	⌜
CURSOR CONTROL →	93	⊖
CURSOR CONTROL ↓	9A	⌞
HOME CURSOR	8E	⊗
CLEAR	8B	⊥

[1] See DEL in Table A4-1.

Table A4-3. Names of Graphic Characters on Keytops
(The names given here are industry-standard terms.)

SP	Space	<	Less Than
!	Exclamation Point	=	Equals
|	Logical OR (not used in the Sol)	>	Greater Than
"	Quotation Marks	?	Question Mark
#	Number Sign	@	Commercial At
$	Dollar Sign	[	Opening Bracket
%	Percent	\	Reverse Slant
&	Ampersand	]	Closing Bracket
'	Apostrophe	^	Circumflex
(	Opening Parenthesis	¬	Logical NOT
)	Closing Parenthesis	_	Underline
*	Asterisk	`	Grave Accent
+	Plus	{	Opening Brace
,	Comma	¦	Vertical Line (This graphic is stylized to distinguish it from Logical OR)
-	Hyphen (Minus)		
.	Period (Decimal Point)	}	Closing Brace
/	Slant	~	Tilde
:	Colon		
;	Semicolon		

APPENDIX 5

Sol SPECIFICATIONS

Keyboard:

85 key upper/lower case with separate numeric keypad. Upper-case shift, shift-lock, cursor control and repeat keys provided. System reset performed by simultaneous depression of control keys. Indicator lights (LED) for local, upper case and shift.

Character set:

96 printable ASCII upper and lower case characters plus 32 optionally displayable control characters.

Cursor:

Switch-selectable blinking. Block video inversion. Program controlled positioning standard. Cursors may occupy at any or all character locations.

CPU:

8080A - Uses same machine language as other 8080 systems. 2 MHz clock cycle time. 78 instructions.

Cassette Interface:

1200 Baud CUTS format or 300 Baud Kansas City format, selected by software. Recorder remote start-stop connector. AGC for level insensitivity. Phase-locked data recovery tracks with speed variations. Software performs CRC data integrity check each 256 characters.

Serial Interface:

RS-232 and 20 mA current loop, 75 to 9600 baud, asynchronous. 25 pin female "D-type" connector on card.

Parallel Interface:

Eight data bits for input and output; output bus is tristate for bidirectional interfaces; levels are standard TTI. 25 pin male "D-type" connector on card.

External memory:

Expandable to 65,536 bytes total ROM, PROM and RAM. (More than 64K with the extended addressing capability of Processor Technology Memory Modules.) Uses S-100 standard modules.

Video Signal Output:

1.0 to 2.5 volts peak-to-peak. Nominal bandwidth is 7 MHz.

Power Requirements:

117 volts AC, 50/60 Hertz, 250 Watts. 220/240 Volt, 50/60 Hertz option available.

READER COMMENTS

Use this postpaid mailer to send your comments on this manual. We will carefully consider your suggestions for incorporation in future editions. To ask questions or comment about the product, please attach a separate sheet.

NAME OF MANUAL _____

PUBLICATION DATE _____ MANUAL PART # _____

Are there specific points that need clarification or correction? Give details, with page and paragraph references.

_____

_____

_____

_____

_____

Did you find this manual easy to use and understand? Do you think certain aspects should be organized differently? Was any necessary material omitted or was any material unnecessary?

_____

_____

_____

_____

_____

Is there sufficient information on associated products required to use the product described in this manual? If not, what material is missing and where should it be placed?

_____

_____

_____

_____

Please check the box next to your occupation:

- ☐ Clerk, Bookkeeper
- ☐ Office Worker
- ☐ Secretary
- ☐ Typist, Stenographer
- ☐ Electronic Typist
- ☐ Small Business Person
- ☐ Manager
- ☐ Doctor, Dentist
- ☐ Nurse
- ☐ Lawyer
- ☐ Government Worker, Official
- ☐ Programmer
- ☐ Technician
- ☐ Engineer
- ☐ Scientist
- ☐ Home Personal Computer User
- ☐ Student
- ☐ Teacher
- ☐ Writer
- ☐ Other Professional _____
- ☐ Other _____

NAME _____ DATE _____

ORGANIZATION _____

STREET _____

CITY, STATE, ZIP _____

If you want a reply, check here ☐

FOLD FOLD

FIRST CLASS
PERMIT NO. 319
PLEASANTON, CALIF.

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

Processor Technology Corporation

7100 Johnson Drive
Pleasanton, CA 94566

attn: Technical Publications Dept.

FOLD FOLD