SPERRY

UTS 30 CP/M Plus™

Programmer's Technical Reference

CP/M Plus™ is a trademark of Digital Research, Incorporated

This document contains the latest information available at the time of preparation. Therefore, it may contain descriptions of functions not implemented at manual distribution time. To ensure that you have the latest information regarding levels of implementation and functional availability, please consult the appropriate release documentation or contact your local Sperry representative.

Sperry reserves the right to modify or revise the content of this document. No contractual obligation by Sperry regarding level, scope, or timing of functional implementation is either expressed or implied in this document. It is further understood that in consideration of the receipt or purchase of this document, the recipient or purchaser agrees not to reproduce or copy it by any means whatsoever, nor to permit such action by others, for any purpose without prior written permission from Sperry.

PRINTED IN U.S.A.

Preface

This manual describes the unique characteristics of the SPERRY Universal Terminal System 30 (UTS 30) when operating in CP/M Plus™ mode. The manual is divided into the following sections:

- Section 1 Introduction
- Section 2 Hardware Characteristics
- Section 3 I/O Port Definitions
- Section 4 Interrupt Interface
- Section 5 Keyboard Functions
- Section 6 CP/M Plus Screen Functions
- Appendix A ASCII Code Chart

This document is not intended to describe basic UTS 30 terminal operation, nor is it intended as a substitute for the Digital Research manuals describing CP/M Plus. Only features unique to SPERRY UTS 30 CP/M Plus are covered in this manual.

Related Sperry Publications

For information regarding basic UTS 30 terminal operation, refer to the Sperry UTS 30 Single Station Operator's Reference, UP–9798. Additional publications which provide related information include the current versions of the following documents:

UTS 30 CP/M Plus™ Release Description, UP–9838.2

UTS 30 CP/M Plus™ Operator's Guide, UP–9839

UTS 4000 CP/M Plus™ User's Guide, UP–9847

UTS 4000 CP/M Plus™ Programmer's Utilities Guide, UP–9848

UTS 4000 CP/M Plus™ Programmer's Guide, UP–9849

UTS 4000 CP/M Plus™ Symbolic Instruction Debugger (SID™) Reference Manual, UP–9844

UTS 4000 CP/M Plus™ Symbolic Instruction Debugger (SID™) Command Summary, UP–9867

CP/M Plus™ and SID™ are trademarks of Digital Research, Incorporated

Contents

6. SCREEN FUNCTIONS FOR CP/M PLUS MODE

APPENDIX A. ASCII CHART

USER COMMENT SHEET

FIGURES

TABLES

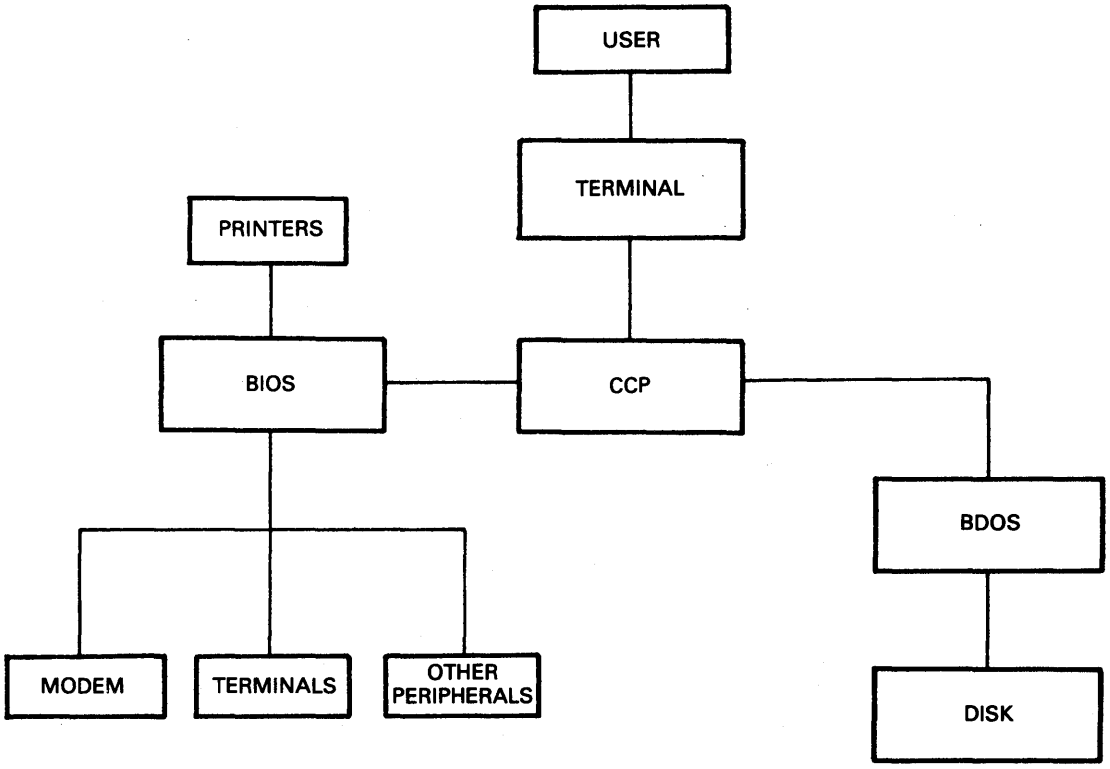
1. Introduction

1.1. GENERAL DESCRIPTION

CP/M® is a disk-operating system designed to operate as a stand-alone microcomputer system supporting peripheral and file management.

The CP/M operating system consists of three major components: the Console Command Processor (CCP), the Basic Disk Operating System (BDOS), and the Basic Input and Output System (BIOS). See figure 1–1.

Figure 1–1. Relationship Between CCP, BIOS, and BDOS



CP/M® is a registered trademark of Digital Research, Incorporated

The CCP interprets commands entered at the keyboard, channeling appropriate tasks to the various resources available through BIOS and BDOS. The BIOS routines manage the flow of data to and from the various devices connected to the system. The BDOS routines manage the disk files.

1.2. SOFTWARE

With SPERRY UTS 30 CP/M Plus, you can take advantage of the wide variety of CP/M-based software packages available. You can also use numerous computer program development languages as well as utilities for the construction, storage, and editing of application programs.

CP/M Plus utilities provide file transfer between peripherals, unattended batch processing, a string-oriented text editor, an Intel-compatible 8080 assembler, hexadecimal display of file contents, conversion of hexadecimal files to CP/M Plus executable files, time and date stamping, and password protection.

SPERRY UTS 30 CP/M Plus also provides several special SPERRY CP/M utilities that take advantage of advanced capabilities of the UTS 30 hardware. These utilities prepare and copy diskettes, configure UTS 30 peripheral devices for use with CP/M, generate and load limited graphics and custom character sets, and reassign default keyboard functions.

1.3. HARDWARE

The SPERRY UTS 30 CP/M Plus system requires a programmable UTS 30 terminal and an 8439 double-sided diskette subsystem with at least one drive. In addition, the optional loadable character set hardware must be installed if custom character sets are to be generated and loaded.

The system also supports one of the following SPERRY printers:

- 0797 serial dot matrix printer
- 0798 serial dot matrix printer
- Model 31 correspondence quality printer
- Model 35 graphics printer
- Model 25 serial dot matrix printer

Refer to the UTS 30 CP/M Plus operator's guide, UP-9839, for instructions on configuration of peripheral devices for use with CP/M Plus.

2. Hardware Characteristics

2.1. HARDWARE FUNCTIONS AND SPECIFICATIONS

Many aspects of the CP/M Plus implementation are dependent on specific hardware characteristics of the UTS 30 terminal. Table 2-1 lists the functions and abbreviated specifications for the UTS 30.

Table 2-1. UTS 30 Functions and Abbreviated Specifications (Part 1 of 2)

Function	Specification
Microprocessor	Z80 CPU (2892668), 4.000 MHz clock No-wait states for local memory Vectored interrupts for peripherals Reset or parity generated NMI
Peripheral LSI	78530 (2893610), serial communication controller MK3801 (2893942), serial/timer/ interrupts HD6845 (2893939), CRT controller
Peripherals	Serial interface keyboard Serial communications channel — full duplex — asynchronous or synchronous — dual baud rate generator — terminal multiplexer 5000 feet max — direct connect 200 feet max — baud rates to 19.2K baud Auxiliary RS-232 (DCE end), up to 900 baud async, full duplex RS-232 standard DCE pinout subset
Memory	128K RAM accessed via address-dependent pages 8K ROM overlay 256 nibbles non-volatile RAM Loadable 512 character generator with all 10 x 16 dots addressable

Table 2-1. UTS 30 Functions and Abbreviated Specifications (Part 2 of 2)

Function	Specification
Display Memory	32 addressable 80-character rows that can be located anywhere in memory 8-bit read/write register for character attribute manipulation
Display Attributes	7 visual character attributes for each character of the display
Display Format	Three Z80 bus interface connections – one for full PC board (graphics) – two for small features 80 columns by 25 lines standard selectable lines from 1 to 32 Standard characters, 7 by 9 in 10 by 15 character matrix
Indicators	Display separator (dotted horizontal line) Audio alarm with programmable frequency No straps Programmable options stored in non-volatile RAM
Keylock	3-position keylock input for terminal and parameter locking

2.2. COMMUNICATIONS SYSTEM

The serial communication controller (SCC) and serial/timer/interrupt controller (STI) combine to support communications. The critical lines, receive data, transmit and receive clock, and clear-to-send B and A are serviced by differential line receivers to reduce noise and improve sensitivity for terminal multiplexer operations. The rest of the signals are lower grade RS-232 receivers. The differential receivers are not biased so that a float results in an indeterminate output. The RS-232 receivers bias the signals to the off state, even though some noise susceptibility exists. The keyboard data is differential and is handled by a differential receiver. The drivers are RS-232 except the transmit data driver, which is assisted to improve capacitive drive capability for long distances on a multiplexer connection. Direct connect is achieved by switching drivers onto the transmit and receive clock lines via a mechanical relay. These drivers do not have extensive drive capability, but should handle 1000 feet of shielded cable. The interface is specified at 200 feet. Longer distances may be permitted in some cases, but you must accept the possibility that it will not operate with an adequate error rate unless the baud rate is reduced.

2.2.1. Serial Communications

The serial communications controller (SCC) integrated circuit (IC) uses channel A for host communications. The SCC is a flexible device, similar to the Z80™ SIO used on the UTS 20 and UTS 40, but with added capabilities. (Details of these capabilities are covered in separate specifications for the IC and in Zilog Corporation vendor literature.) Specific capabilities of the SCC and the applications related to the UTS 30 are discussed in detail in this section.

The other serial communications IC, the serial/timer/interrupt (STI) controller also supports some host communication functions. Channel B of the SCC handles the serial keyboard data. The communications section of the STI handles the auxiliary RS-232 interface for devices. Because these two ICs form an intimate peripheral control group, they are combined in the following paragraphs.

Where reasonable, signals that interface to the ICs are titled by the IC pin name; while actual details of the manipulation of the pins and signals are defined in the specification for that particular IC.

2.2.2. Modem Communications

Most of the modem communication functions are handled by the SCC channel A. The SCC operates in synchronous or asynchronous mode. When asynchronous mode is used, the internal baud rate generator is used. The basic clock frequency is 4.000MHz. The internal receiver divider should be set to the highest value consistent with the baud rate generator. A division by 16 can result in higher error rates than 32 or 64. The maximum asynchronous baud rate is 19.2K, achieved by setting the baud rate generator to divide by 13 and the receiver to divide by 16. The small frequency error is of little consequence.

The signals are supported via the communication cables, features F8209, F8201, and F8211, and are discussed in the following paragraphs.

Transmit to Host

The transmit data signal, CCITT-103, is routed to the SCC TxDA pin 15. Depending upon whether the mode is synchronous or asynchronous, the clock can be programmed to come from the host modem or internally.

Receive from Host

The receive data signal, CCITT-104, is routed to the SCC RxDA pin 13.

Request-to-send

The request-to-send (RTS) signal, CCITT-105, is routed to the SCC RTSA pin 17.

Clear-to-send

The clear-to-send (CTS) signal, CCITT-106, is routed to the communication status register port 41, bit 0. The register must be strobed by an I/O read or write of port 42. The status is then read via I/O port 41. CTS is also routed to STI interrupt 5.

Carrier Detect

The carrier detect (CD) signal, CCITT-109, is routed to the communications status register port 42, bit 1. This line (CD) is logically ORed with CTS and routed through a 30 microsecond delay. This delay

™Z80 is a registered trademark of Zilog Corporation.

generates a high true input at the STI interrupt 7 (pin 15) input. Thus, 30 microseconds after the assertion of CTS or CD, an interrupt occurs. The delay also strobes the register at port 42, latching all the bits. This interrupt function is used primarily for terminal multiplexer handshaking, but can also serve modem operation. A positive interrupt at STI interrupt 7 occurs when CD or CTS asserts (after 30 microseconds); a negative interrupt at STI interrupt 7 occurs when CD and CTS negate (with negligible delay, the 30 microsecond delay does not operate on negation).

Data-Set-Ready

The data-set-ready signal, CCITT–107, is routed to input port 42, bit 7.

Transmit Clock

The transmit clock signal, CCITT–114, is routed to the SCC TRxCA pin 14. This is a bidirectional line. When in the input mode, it receives the timing for transmit data. It is used in the output mode for direct connect mode, and generates both transmit and receive clocks. This line is XORed with the output of I/O port 49, bit 3, for use with the RS-232/V.24 test adapter 2826629. The clocks generated for direct connect mode are out of phase and are used for loopback with the test adapter. The clock must be inverted via I/O port 49, bit 3 in order to generate the proper clock polarity at the SCC.

Receive Clock

The receive clock signal, CCITT–115, is routed to the SCC RTxCA pin 12. When in direct connect mode, this line is driven indirectly by the TRxCA pin.

Data Terminal Ready

The data terminal ready (DTR) signal, CCITT–108.2, is routed to I/O port 49, bit 4. This port bit also turns on an amber light, visible from outside the terminal, that indicates the terminal is operational. Turning off the line will extinguish the light and make the operator aware of a fault condition. This bit and all bits in the port are reset when any reset occurs; therefore, DTR is negated by the reset switch, and some modems are disconnected.

This signal also goes to the DTR timer. The timer is not activated until the following conditions are met:

- DTR must be asserted.
- Port 49, bit 3 must be equal to 1.

When these conditions are met, the timer remains set for 128 cycles of the vertical synchronous from the CRTC, then asserts at the CTS A pin on the SCC, and inhibits DTR, the light, and RTS A. The timer is retriggered, or kept running, by any interrupt acknowledge by the CPU (IORQ-M1 cycle) that prevents timeout. This system provides for the event that the microcode ceases to function when operating in an interrupt environment, and the DTR is asserted. Microcode should provide a timer that prevents the RTS and DTR lines from being active too long, locking up the communication system or the telephone line. The DTR timer is provided to back up the microcode timer. The DTR timer does not function if port 49, bits 3 or 4 are negated, or the CRTC is not generating synchronous pulses. The CRTC must be activated (and vertical synchronous generated) when DTR is asserted and the timer is required.

Ring Indicator

The ring indicator (RI) signal, CCITT–125, is routed to the STI interrupt 0. This interrupt should be enabled only when RI is expected, since modems and other equipment generate spurious signals on this line.

Data Rate Select

The data rate select signal, CCITT–112, is routed to the SCC DTR/REQA pin 16. It serves as the RTS-B signal on the SPERRY terminal multiplexer interface. No other CCITT signals are supported on this interface except grounds 101 and 102. The RS–232 and V.24 cables provide for proper routing of the signals on this interface. The use of other cables is not necessarily nonfunctional, but may cause erratic operation on some equipment. The terminal multiplexer cable is suitable for communications with the SPERRY terminal multiplexer described in the following paragraphs.

2.2.3. Terminal Multiplexer

Terminal multiplexer operation is supported via terminal multiplexer cable feature F8217 (with exceptions noted in this section). Operation on the terminal multiplexer is allowed up to 5000 feet at 9600 baud, or 2000 feet at 19.2K baud. A 10000–foot operation is allowed at 4800 baud. The transmit data driver has an output impedance of about 50 ohms, allowing long distances for terminal multiplexer systems.

Handshake with the terminal multiplexer is facilitated by control of RTS A, RTS B, and transmit data. The inputs CTS A, CTS B, and transmit clock are latched in a register, and generate only one interrupt to the STI (0) when they are latched by an assertion from CTS A or CTS B. The data remains in the latch until both CTS A and CTS B negate and one of them asserts again. An interrupt occurs when both negate (1) if so programmed. The latch is strobed about 30 microseconds after the assertion of CTS A or CTS B. This guarantees that the state of send clock during this operation is valid, including cable and receiver skew. If a glitch occurs on CTS A or CTS B while it is asserting, the latch stores the status after the glitch, possibly losing the transmit clock status.

The lines transmit data, receive data, transmit clock, and receive clock are the same as the modem interface to the SCC.

Clear-to-send A and clear-to-send B are routed to an input I/O register at I/O port 42. Transmit clock is also routed to this port. These three lines are latched 30 microseconds after assertion of CTS A or CTS B. Also, at the 30 microsecond delay point, SCC interrupt input 7 is driven high, generating an interrupt and status, indicating that the terminal multiplexer has issued status and is available in port 42 (bit 0 = CTS A, bit 1 = CTS B, and bit 2 = transmit clock). When the terminal multiplexer has negated both CTS A and CTS B, the STI interrupt 7 input goes low. The status in the register is still valid. When the terminal multiplexer again asserts CTS A or CTS B, the status is relatched after 30 microseconds, generating a positive transition at STI interrupt 7. The microcode has a limited time to capture the status, before the terminal multiplexer reasserts the CTS lines. Request-to-send A is routed to the SCC RTSA pin 17.

Request-to-send B is routed to the SCC DTR/REQA pin 16.

Request status to the terminal multiplexer requires that the transmit data line be either marking or spacing (1 or 0) when RTS is asserted. The SCC internal register 5 can be used for this purpose. The output is marking (1) when the transmitter is not enabled; the output is spacing (0) when the transmitter is enabled and the command "send break" is issued.

NOTE:

The send break function of the SCC requires transmit clock. It cannot be guaranteed when break is required (on and off): the internal baud rate generator should be used during the break duration.

Modem lines not used for terminal multiplexer operation should have interrupts disabled to prevent noise from generating spurious interrupts.

2.2.4. Direct Connection

Operation up to 200 feet and 19.2K baud is provided by generating the required transmit and receive clocks from the SCC internal baud rate generator. The clocks are both routed from the SCC TRxC pin 14. The external circuits must also be programmed to drive the clocks. Set I/O port 49, bit 2 = 1. When operating in this mode, interrupts should be disabled for all modem control lines, since they are unterminated and generate spurious inputs. The SCC internal register 11 must be set to generate the clock at the TRxC pin. The clock input is 4.000 MHz, used in calculating the baud rate values for registers 12 and 13 in the SCC.

Direct connection at distances longer than 200 feet can be accomplished by reducing the baud rate or operating with a direct connection module (DCM). When the DCM is used, it generates the clocks. It does not necessarily terminate all control lines, so unused lines should not be interrupt enabled. J8 interfaces to the communications line via IC's 2892905, 2899272, 3007893, and 3007894.

2.3. MEMORY CONFIGURATION

Specifications regarding memory configuration are described in the following paragraphs.

Main Memory Architecture

The main memory consists of 128K bytes of dynamic RAM. The RAM is organized so that the CPU has access to all 128K of memory in a page-swapping scheme. The memory is divided into a main bank and a user bank, each 64K bytes deep, selected by address bit 16. Address bit 16 is generated using an 8-bit register (I/O port 4B). The bits in this register become address bit 16 (A16) for each of 8 address blocks, 8K bytes each, resulting in 16 pages. Eight at a time are selected. Address bits 15, 14, and 13 determine which bit is A16. If the three bits are zeros, port 4B bit 0 is A16. This relationship is shown in table 2-2.

Table 2-2. Memory Addressing Architecture

Register Bit		
Address Page	**Bank**	**(Port 4B)***
E000	user	D7 = 1
C000	main	D6 = 0
A000	main	D5 = 0
8000	user	D4 = 1
6000	user	D3 = 1
4000	main	D2 = 0
2000	user	D1 = 1
0000	main	D0 = 0

*Register 4B has been written to 9A.

Writing port register 4B will cause the RAM map to switch during the I/O instruction. Four control bits are provided to allow microcode to reside in any address space as well as access or jump to any other address space.

ROM

Read only memory (ROM) of 8K bytes is located onboard (addressed at 0000), independent of the state of the user bank control logic. The ROM can be switched in and out via I/O port 48, bit 0. If bit 0 = 0, ROM is selected. If bit 0 = 1, ROM is not selected. The RAM at the ROM address space can always be written, even if ROM is selected. A write to ROM actually goes to RAM. The bit is delayed until the beginning of the second M1 cycle, allowing programs to jump in and out of ROM from RAM at the same address. Since write is allowed, loading data into RAM while ROM is selected is simple. In order to read RAM at the ROM address using ROM code (POC), address bit 13 must be inverted via I/O port 4A, bit 1.

Non-volatile RAM

A RAM that stores data with power off is provided for parameter storage. This RAM is actually two memories in one IC, a conventional static RAM and an electrically alterable ROM (EEROM). The RAM data is transferred into EEROM when it receives a store command. This command is issued by hardware when a power fail warning occurs. This warning occurs at least 10 milliseconds before power drops, and the system is reset. An interrupt input to the SCC DCDA pin indicates that the warning has started a store cycle. The NVR is unavailable to the CPU for a maximum of 10 milliseconds. The EEROM is able to execute 5K store cycles before wearout. Wearout aging occurs only if a bit in EEROM must change states. Store cycles without changed bits do not age these bits. The 5K store cycles should be adequate for the life of the terminal, since the cycles occur only on power down. It is reasonable to use the NVR for error logs and control page parameters, as well as basic terminal parameters. The NVR is organized in 256 four-bit nibbles. If it is selected by setting I/O port 48, bit 7, it is addressed at the beginning of attribute display memory.

Address 16 Generation

Address 16 is needed by the CPU to access 128K of RAM. An ordinary I/O controlled paging mechanism is not suitable for all applications and a RAM mapping mechanism is not used. Instead, an 8-bit register with a bit assigned for each of 8 memory address blocks is incorporated. This register output goes to an 8:1 multiplexer. The output of the multiplexer is passed through a buffer and becomes address 16. The multiplexer is selected by address bits 13, 14, and 15. These select 8K blocks of memory. Thus, if the address bits point to bit 3 of the 8-bit register, that bit determines A16.

Two control bits in port 4A serve to enable the multiplexer, and therefore, address 16. One bit is delayed by 4 MI assertions and the other is immediate.

Address 13 Manipulation

Address 13 is inverted as a result of three possible conditions. Each is a special purpose function. One condition controlled by port 4A, bit 0 is used by the microcode to create a two-state machine for use in the screen bypass mode, where there are two virtual terminals created. This bit inverts address 13 for all addresses at the display location and above. Therefore all display information, pointers, and data buffers in RAM can be selected by changing the state of port 4A, bit 0 without changing or keeping track of addresses in the CPU. This bit has no other practical use.

Address 13 is also inverted conditionally on the state of address 16, in order to allow microcode to manipulate RAM data, regardless of the microcode location doing the manipulation. The inversion is accomplished by the Boolean expression:

A13 = A13c XOR ((A not < DSR x 4A0) + (A16 not x 4A1 x r) + (A16 x 4A2 x r))

where:

A13 = external address bit 13

A13c = CPU address 13

A = CPU address

DSR = address pointed to by the DSR

4A0 = port 4A bit 0, etc.

r = any address not in ROM space

2.4. AUXILIARY RS-232 ASYNCHRONOUS INTERFACE

This interface is connected both to the SCC channel B control lines and the STI serial channel. The STI does not buffer data, so it is not used for the keyboard which sends bytes in pairs. When the STI receiver is operating, it is important to read the data from the receive register before the next byte is assembled or data is lost. Thus, it may be necessary to force the keyboard to wait if the STI receiver is operating. The keyboard internally buffers up to eight keystrokes if the SCC channel B receiver still has data in it.

Aux transmit data (from the device) is routed to the STI serial input pin 38.

Aux receive data (to the device) is routed to the STI serial output pin 37.

The asynchronous clock input to STI pins 36 and 39 is driven by SCC TRxCB pin 26. Thus the SCC internal baud rate generator for channel B serves as the generator for the STI channel. The maximum baud rate for this channel is 9600, since the input to the STI cannot be driven at twice that speed.

Aux request-to-send data is routed to the SCC input DCDB pin 22.

Aux clear-to-send data is routed to the SCC output RTSB pin 23.

Aux data set ready is routed to the SCC output DTR/REQB pin 24. This line must be negated prior to power off to prevent the printer from accumulating garbage in its buffer, as one piece of junk might be a CR. The power fail interrupt can be used to negate the line.

Aux data terminal ready is routed to the SCC input SYNCB pin 29.

Aux K-ready is routed to the SCC input CTSB pin 22.

Aux carrier detect is not supported.

3. I/O Port Definitions

3.1. GENERAL PORT DEFINITIONS

The I/O port bit assignments used in the UTS 30 are described in the following table. For each bit position described, the true or assert condition is 1. In some cases, the negate or 0 state is identified.

Table 3–1. I/O Bit Port Assignments (Part 1 of 6)

Address	Direction	Function
00*	In/Out	SCC channel B control registers DTR line drives DSR (AUX) RTS line drives CTS (AUX) DCD line driven by RTS (AUX) CTS line driven by K-Ready (AUX) SYNC line driven by DTR (AUX) TRxC line drives printer clock into STI RTxC line driven by 615.38 KHz keyboard clock (9600X64) REQ/READY drives keyboard enable (ready-receive mode) (receive buffer not empty inhibits keyboard)
01*	In/Out	SCC channel B keyboard data
02*	In/Out	SCC channel A control registers DTR line drives RTS B/RATE RTS line drives RTS A DCD line driven by 1 = NVR store cycle started CTS line driven by 1 = DTR timer expired SYNC line not connected TRxC line driven by send clock (modem, terminal multiplexer) or by keyboard clock (19200X32) Drives connect clocks RTxC line driven by receive clock (modem, terminal multiplexer) or by STI channel D timer output REQ/READY not connected
03*	In/Out	SCC channel A communications data

* Ports 00 through 03 do not use address bits 2 or 3.

Table 3–1. I/O Bit Port Assignment (Part 2 of 6)

Address	Direction	Function
10	In/Out	STI control registers, actual addresses are 10 through 1F corresponding to register addresses 0 through F. Clock input is 615.38KHz. Timer outputs are half the channel interrupt rate. **Timer Output:** A – Not used (input intrpt 4 vert sync) B – Audio (reset to 0 after beep or click) C – 16X field Register 9 (OE)–Character field size, default to 15 D – Drives RTxC A (if enabled) **Interrupts:** 0 – Modem ring indicator 1 – 0=port B interrupt 2 – 0=port C (graphics) interrupt 3 – Cursor blink (8X cursor) 4 – Positive vertical synchronization 5 – Modem CTS 6 – 0=port A interrupt 7 – CTS assertion, check port 42
20*	Out	CRT controller register select port
21*	Out	CRT controller parameterization registers **Hex Value of Registers** Register 0 (68**) – Must set to 105 char. Register 1 (59**) – Horiz. displayed 80th char. Register 2 (59**) – Must set H sync to 85th char. Register 3 (8C**) – Must set sync width Register 4 (1C) – Vert total, 70 Hz Register 5 (00) – Vert adjust, 80 Hz Register 6 (19) – Vert displayed, default 25 lines Register 7 (1B) – Vert sync Register 8 (AO**) – Interlace/skew Register 9 (OP) – Character field size, default to 15 Register A (00) – Display separator width/blink (default) Register B (00) – One line at top of character row Register C(H) (08**) – Display start, set to 0800 Register D(L) (00)
	In/Out	Register E(H) – Display separator position, screen address referenced to screen physical start.**
	In/Out	Register F(L) – Display separator position, screen address referenced to screen physical start.**
	In	Register 10(H) – Graphics cursor position under closest display character, absolute address referenced to screen physical start.**
	In	Register 11(L) – Graphics cursor position under closest display character, absolute address referenced to screen physical start.**

*Ports 20 and 21 do not use address bits 1, 2, or 3.

**These values affect display stability and centering and should not be changed.

An address bit is used for timing the row counter. Add 08 to E and 10. The vertical value can be changed if the total number of scan lines including retrace is unchanged from that shown above (vertical total X character field size). The vertical synchronization position affects screen vertical centering. To turn off the display separator, write FF to register A.

Table 3-1. I/O Bit Port Assignments (Part 3 of 6)

Address	Direction	Function
60*	In/Out	Access of this port serves to reset the nonmaskable interrupt latch if the reset switch is not being actuated.
61*	In/Out	Access strobes port 42; data is arbitrary.
62*	In	Communications status, latches terminal multiplexer handshake lines. Strobed by assertion of CTS A, CTS B, or port 42. **Data Bit Status:** Bit 0 1=CTS A — { terminal multi- plexer handshake lines } or modem CTS Bit 1 1=CTS B — or modem CD Bit 2 1=Send clock or modem TxC Bit 3 1=Set up enabled keylock position Bit 4 1=Terminal locked keylock position Bit 5 1=Reset switch caused last system reset (0=power on) Bit 6 AHR parity bit Bit 7 1=Modem data set ready
63*	Out	Character generator, character holding register. Write character to be formed (i.e., ASCII "A", write 40).
64*	Out	Segment pattern for character in port 43, 16-bit I/O write using "OUT C, A" instruction, with 44 in CPU register C.
67*	Out	Strobes 4A bit 7. Address 16 will become valid after 4 memory cycles. Interrupts are inhibited between the time 4A bit 7 differs from the strobed and delayed 4A bit 7, if the strobe will change the state of the flip-flop.

*Ports 60 through 67 do not use address bit 3.

Table 3–1. I/O Bit Port Assignments (Part 4 of 6)

Address	Direction	Function
48*	Out	Display control register; cleared by power on. **Data bit status:** Bit 0 1=Disable ROM after 4th following memory cycle 0=Enable ROM after 4th following memory cycle ROM forced on by NMI flip-flop; bit not affected Bit 1 1=RSR(H)-RSR(L)=4K Screen attributes; 4K above characters 0=RSR(H)-RSR(L)=32K Screen attributes; 32K above characters. If bit=1, changes CPU A12 and CPU A15 to RAM Bit 2 1=CG write mode; set before port 44, Display=junk 0=Display mode; characters displayed Bit 3 1=Blank characters with blank bit in CG Bit 4 1=Reverse-video screen (black on light) Bit 5 1=Attribute bit 3 selects upper 256 CG characters 0=Attribute bit 3 selects strike-through character attribute Bit 6 Not used Bit 7 1=Enable non-volatile RAM overlay at DSR 0=NVR is disabled
49*	Out	Communications control port; cleared by power on. **Data bit status:** Bit 0 1=Invert CPU address bit 13 above DSR (screen bypass) Bit 0 1=TRxC A driven by keyboard clock (19.2K x 32) 0=TRxC A driven by line receiver Bit 1 1=Select long distance receiver hysteresis (multiplexer, DC) Bit 2 1=Select direct connect clock driver Bit 3 0=Invert Xmit clock receiver for loopback plug; inhibit punt timer Bit 4 1=Data terminal ready (modem), and maintenance light. Bit 5 1=RTxC A driven by STI channel D output 0=RTxC A driven by line receiver Bit 6 1=Enable peripheral ports 1 and 2 0=Reset ports Bit 7 1=Invert RAM write parity; reset cursor counter (test)

*Ports 48 and 49 do not use address bit 3.

Table 3–1. I/O Bit Port Assignments (Part 5 of 6)

Address	Direction	Function
4A*	Out	Memory address control; cleared by power on or interrupt ACK. **Data bit status:** Bit 0 1=Invert CPU address bit 13 above DSR (screen bypass) Bit 1 1=Invert CPU address bit 13, if bit 16=0, except ROM Bit 2 1=Invert CPU address bit 13, if bit 16=1, except ROM Bit 3 1=Enable user RAM bank logic, if bit 7 delayed is 1 Bit 4 1=Enable attribute holding register between DSR and DER Bit 5 1=Enable attribute holding register read operation 0=Hold contents of attribute holding register Bit 6 1=Inhibit interrupt ACK from clearing register Bit 7** 1=Enable RAM bank 2 (A 16) into delay latch, see port 67. 0=Disable RAM bank 2**
4B*		Memory bank register (A16 control); cleared by power on only. Eight bits select address blocks where A16=1 in 64K address space. Bits 0 through 7 enable blocks 0 through 7. Blocks are 2000 hex bytes. Block 0 is at 0000 through 1FFF; block 7 is at E000 through FFFF, etc.
4C*		Port 4A bits 1 and 2 swap the even and odd blocks. Display start and display end registers (DSR and DER) cleared by power on only. **CPU display addr if port 48 bit 1=0–A16 A14 A13 A12 A11** **CPU display addr if port 48 bit 1=1–A16 A15 A14 A13 A11** **Display counter address (and RSR) A15 A14 A13 A12 A11** **DSR register bits----\D7—D6—D5—D4/** **DER register bits----------\D3—D2—D1—D0/** A. DSR and DER limit address range of the attribute holding register. The AHR effectiveness is from DSR to DER+07FF. B. DER determines the address of the RSR(L); RSR(H)=RSR(L)+1000 OR 8000. The RSR(L) is at DER+07E0. C. DSR determines the address of the NVR which is at DSR, if selected. D. DSR determines the address above which port 4A bit 0 operates. This carries into the user bank, if DSR is in the main bank. When port 4A bit 0=1, DSR bits 4 and 5 must be 0 or a fault will result. The most significant address bit in the RSR determines the logical bank where the display resides (A15). If it is set, that display line is accessed from the user bank. DER bit 7 determines the bank (user or main) where the RSRs are located.

*Ports 4A through 4C do not use address bit 3.

**These bits take effect in both directions at the end of the indicated M1 cycle after the output instruction. NMI has no effect. NMI latch must not be cleared until all registers are initialized.

Table 3–1. I/O Bit Port Assignments (Part 6 of 6)

Address	Direction	Function
4D	Out/In	Generates interrupt pulse to external port A.
4E*	Out/In	Generates interrupt pulse to external port B.
4F*	Out/In	Generates interrupt pulse to external port C (graphics).
50-5F	In/Out	Addresses 50 through 5F select ports on the graphics board. These ports generally serve to override the graphics processor for fundamental initialization and RAM access control.
80-8F		Port addresses 80 through 8F are reserved for devices that reside in the external ports 1, 2, or 3. Port 3 uses this space in the event that addresses 50 through 5F are full or unavailable. The program cartridge load device is I/O accessible only via any I/O input above 80.

*Ports 4E through 4F do not use address bit 3.

NOTE:

The I/O port addresses below 40 are located on the internal bus, and the remaining ports are located on the external bus. All ports are affected by read or write. Input of an output port will cause that port to receive indeterminate data.

The following port addresses are decoded but not used in the UTS 30. Changes to this list may occur at a later date.

30 - 3F

65, 66 (next in line for use)

70 - 7F

3.2. I/O PORT 4A

A detailed description of the bit definitions for I/O port 4A follow. This port controls the memory addressing by the CPU.

Bit 0

Bit 0, when set to 1, causes address bit 13 to be inverted at all addresses (including A16) that exceed that value loaded in the 4-bit DSR register (4C).

This condition is used by the screen bypass function and is very specific. This function is limited. If DSR bit 4 is not 0, erratic memory addressing or undefined AHR operation may result.

Bit 1

Bit 1, when set to 1, causes address bit 13 to be inverted for all addresses where A16 = 0.

This bit is used when the microcode is resident in the user bank. This bit has no effect on ROM addressing, but does affect RAM that is written at the ROM address.

Bit 2 Bit 2, when set to 1, causes address bit 13 to be inverted for all addresses where A16 = 1.

This bit is used when the microcode is resident in the main bank. This bit has no effect on ROM addressing, but does affect RAM that is written at the ROM address. Bits 0, 1, and 2 are logically ORed before the inversion takes place (setting two bits does not cause a double inversion).

Bit 3 Bit 3 must be set in order for address bit 16 to be set.

This bit changes during the I/O output instruction and is handy for selecting the pages enabled via port 4B. Bit 7 delayed must also be 1 for this bit to operate.

Bit 4 Bit 4 enables the attribute holding register (AHR).

When this bit is 1, the byte in the AHR transfers to RAM if the address of the write is in display space (see port 4C).

If bit 5 is set, a read of RAM in display space causes that data to be loaded into the AHR. The data that is affected by the AHR is that byte corresponding to the attribute of the data character being written.

For example, if a character is read from display space by the CPU, its corresponding attribute is loaded into the AHR. If the CPU then writes to another location in display space, the byte in the AHR is also written into the attribute location for that character. Display attributes are fixed at an address offset from the data character. This offset is either +2000 or +8000 (see port 48, bit 1).

Bit 5 Bit 5 is used to enable the read function that causes the AHR to be loaded with the memory data that is in the attribute space. This bit operates only if bit 4 is set.

If bit 5 = 0, the AHR serves as a holding register to propagate attributes for each display space write. If bit 5 = 1, the AHR assists in moving data around in display space as the attribute is automatically transferred. When bit 4 is enabled, other reads or writes in display space must be avoided or the attribute space and the AHR contents may inadvertently change. The AHR carries parity and is discussed in the following paragraphs.

Bit 6 Bit 6 is reset to 0 if interrupts are to be handled in only the main bank.

Interrupt acknowledge clears all bits in port 4A. This causes the vector and instructions to be accessed in the main bank regardless of the contents of port 4B or the previous value of port 4A. If bit 6 = 1, interrupts do not affect the memory configuration.

Bit 7

Bit 7, when set to 1, enables address bit 16, as defined by port 4B.

This bit is delayed until after Port 67 is written and 4 memory cycles have occurred. This allows the microcode to jump from one bank to the other and properly return from interrupts if bit 6 is reset. It allows for either a jump, branch, or call. This bit should be left in the 1 state, and bit 3 should be used for bank selection if this function is not to be delayed. Port 4B can also be used for immediate selection.

3.3. I/O PORT 67

Port 67 is used to delay the action of port 48 bit 7 so that microcode may recover the environment. A detailed description of the bit definitions for Port 67 follows. The four memory cycle instruction fetches (opcodes) use two cycles. The position of the out to 67 in relation to the call or return determines whether bank switching occurs before or after the stack operation interrupts are inhibited during the bank switching operation. When 4A bit 7 state disagrees with the delayed state of the 4A bit 7 strobed and delayed value, interrupts will be blocked. This action takes two instructions to activate. The following example may clarify this operation.

Assume that code is executing in the user bank and desires to switch to the main bank via a call.

1. Switch port 4A bit 7. After 2 more instructions, interrupts will be blocked.

2. Set up all registers as desired.

3. Write to port 67.

4. Call routine; stack will occur in the user bank. Interrupts will be enabled after the first instruction in the main bank.

NOTE:

Insert an instruction that uses two memory cycles, such as an NOP between step 3 and 4 if the stack is to be in the main bank.

Returns and jumps are implemented in a similar way. Do not use branch instructions.

3.4. I/O PORT 48

Port 48 controls the display parameters and the ROM. A detailed description of the bit definitions for I/O port 48 follows.

Bit 0

Bit 0 selects the ROM always addressed at 0000 through 1FFF.

If bit 0 = 0, the ROM is enabled, and overlays RAM at its address. The RAM at the ROM address can be written to when the ROM is selected, but all reads at the ROM address result in ROM data. If bit 0 = 1, the ROM is deselected. The ROM remains selected for the next four memory cycles after the output instruction to port 48, allowing a PC HL or jump to any address, including one at the ROM address. The ROM is not affected by any of the bits in port 4B or 4A except bit 0 in port 4A.

This bit causes ROM to be displaced by 2000 if the DSR is 00. ROM can also be deselected by activating the signal, inhibit ROM, available at J3. This is generally used by the maintenance panel or for test purposes.

The ROM is selected whenever the NMI latch is set; however, the NMI latch does not affect the state of bit 0. The NMI latch is set whenever RAM parity is read incorrectly from any source, or when the reset switch is activated. If the NMI latch is cleared (by accessing port 40) while bit 0 = 1 the ROM is immediately deselected. When set, the NMI latch generates one nonmaskable interrupt. All NMI handling must start in ROM at address 0066. The NMI latch can be set on power up, but no NMI occurs. Before exiting ROM, the latch must be reset or bit 0 is ignored.

Bit 1 Bit 1 determines the address separation of the display attributes and display data.

When bit 1 = 0, the attributes are 8000 hex greater than the display characters. The row start registers (RSR) are also split by the same 8000 bytes (RSR L + 8000 = RSR H). In other words, the low order byte of the row start address is in the display area, and the high order byte is in the attribute area.

When bit 1 = 1, the attributes are 1000 hex greater than the display characters (RSR L + 1000 = RSR H).

Bit 2 Bit 2 controls the character generator.

When bit 2 = 0, the character generator is in display mode. When bit 2 = 1, the character generator is in write or load mode; displayed characters are arbitrary. This bit must be set prior to loading the data via port 43 and 44.

Bit 3 Bit 3 controls the characters in the character generator having the blank bit set.

If bit 3 = 1, those characters are blanked. If bit 3 = 0, the characters are displayed normally. This is used either for special characters that must be conditionally displayed, or for characters that must be blinked at a rate different from that provided by the blink character timer.

Bit 4 Bit 4 controls the polarity of the video information.

When bit 4 = 1, light characters on a dark background are displayed. When bit 4 = 0, dark characters on a light background are displayed. This bit also affects graphics data.

Bit 5 Bit 5 selects the function of attribute bit 3.

When bit 5 = 0, attribute bit 3 causes a strike-through to appear over the corresponding display character. When bit 5 = 1, the alternate character set is selected. This bit must be used when loading the character generator and should be set prior to ports 43 or 44. If the alternate character set is not installed, the characters are generally displayed as a solid block. This is not guaranteed, but an indication of whether the feature is present. If it is present, random dots are displayed.

Bit 6 Bit 6 is not used.

Bit 7

Bit 7 selects the non-volatile RAM (NVR). When bit 7 = 1, the NVR overlays the beginning of display RAM as defined by the DSR. The NVR address is at DSR to DSR + 00FF. When bit 7 = 0, the main RAM is again selected. When this bit is set, the action of the AHR is inhibited at DSR to DSR + 0FFF.

3.5. I/O PORT 49

Port 49 controls some communications parameters and the parity generator. A detailed description of the bit definitions for I/O port 49 follows.

Bit 0

Bit 0 determines the driver of the TRxC A pin of the SCC IC.

When bit 0 = 0, the TRxC A pin is driven by the line receiver for online synchronous communications. This bit is not used when bit 2 of this port is set (direct connect mode).

Bit 1

Bit 1 is used to select the receiver sensitivity to control noise susceptibility. 1 = long distance; 0 = short distance cable less than 50 feet. When bit 1 = 0, it reduces the receiver sensitivity on some lines, improving noise immunity.

NOTE:

This function is not implemented, but may be in the future if the need arises. To allow for future implementation, this parameter should be supported in the microcode to avoid future code changes.

This bit is dependent only on cable length, not on communications mode.

Bit 2

Bit 2 is used to select the direct connect clock driver. When bit 2 = 1, it selects direct connect. The TRxC A pin of the SCC drives both the transmit and receive clocks on the communications interface. This bit must be set prior to setting the TRxC A pin to output, or driver conflict (because of noise generation) will result causing possible system malfunction.

Bit 3

Bit 3 controls the polarity of the transmit clock receiver.

When bit 3 = 0, the receiver is inverted for loopback testing with the loopback adapter, 2832471. In loopback, the transmit clock is inverted from the receive clock to reduce cable crosstalk. This bit also serves to disable the punt timer which is active when RTS is active. The punt timer serves no purpose in asynchronous applications so this bit can be left 0. When bit 3 = 1, it allows normal reception of the transmit clock signal and is used for synchronous communications. It also enables the RTS timer causing DTR and RTS to be negated if interrupts are ignored for longer than about 2 seconds.

Bit 4

Bit 4, when set to 1, activates DTR and lights the maintenance indicator on the top edge of the card.

This light should be on whenever the terminal is operating, indicating to a service operator that the system is operational from the point of view of the CPU. However, if the terminal is connected to an auto answer modem, the DTR line should only be active when a connection can be made. POC should not activate DTR except under special conditions.

Bit 5 Bit 5 determines the driver source for the SCC RTxCA pin.

If bit 5 = 0, the pin is driven by the communications receive clock line receiver. If bit 5 = 1, the pin is driven by the STI channel D output. This configuration prevents the use of timer D for system uses, but provides for dual asynchronous baud rates for transmit and receive during the video-text application.

Bit 6 Bit 6 resets the peripheral ports.

If bit 6 = 0, it resets peripheral ports A and B. If bit 6 = 1, it enables the ports.

Bit 7 Bit 7, when set, is a test bit that causes RAM write parity to be inverted and presets the cursor counter for more effective testing.

If bit 7 = 0, normal even parity is operating, and the cursor counter operates.

4. Interrupt Interface

4.1. GENERAL DESCRIPTION

CP/M Plus uses interrupts to control keystroke cycling, system clock, keyboard input, communications, and peripheral device control. This section describes the use of interrupts with CP/M Plus on the UTS 30.

Interrupts in the CP/M Plus system run in Z80 interrupt mode 2, with the interrupt address register in the Z80 set to memory page OFFH (hex). The mode and the page address must not be changed under any circumstances.

In order for an interrupt to be trapped, the interrupt vector must be saved, and the entry point address of the desired routine substituted in an address location associated with the interrupt vector addresses shown in table 4-1.

All interrupts are first handled by the operating system. Consequently, interrupts are disabled and the registers AF, BC, DE, and HL are saved prior to entering into the interrupt handler. After the interrupts have been processed, the interrupt handler restores all registers except AF, BC, DE, and HL, and then returns control to the operating system. The operating system then handles the remainder of the interrupt process (i.e. turns on the interrupts and returns back to the interrupted routine).

Since the operating system uses interrupts, an interrupt handler should return control to the operating system no later than 30 milliseconds after gaining control. The system timer interrupts every 34 ms. Interrupt mode 2 is in effect while CP/M Plus is loaded.

Keyboard input is driven by interrupts from SCC channel B. Asynchronous communications are driven by interrupts from SCC channel A.

The system timer is driven by interrupts from STI timer A. The printer interrupts are driven by interrupts from STI receive buffer full.

The interrupt vectors are found at memory locations OFF40H to OFF6FH (see table 4-1). The user may change any of the 24 vectors. The user's interrupt routine is the code to process the interrupt and a return.

The operating system turns on interrupts when the user interrupt routine is complete.

4.2. RESTRICTIONS

The following restrictions apply to the interrupt interface:

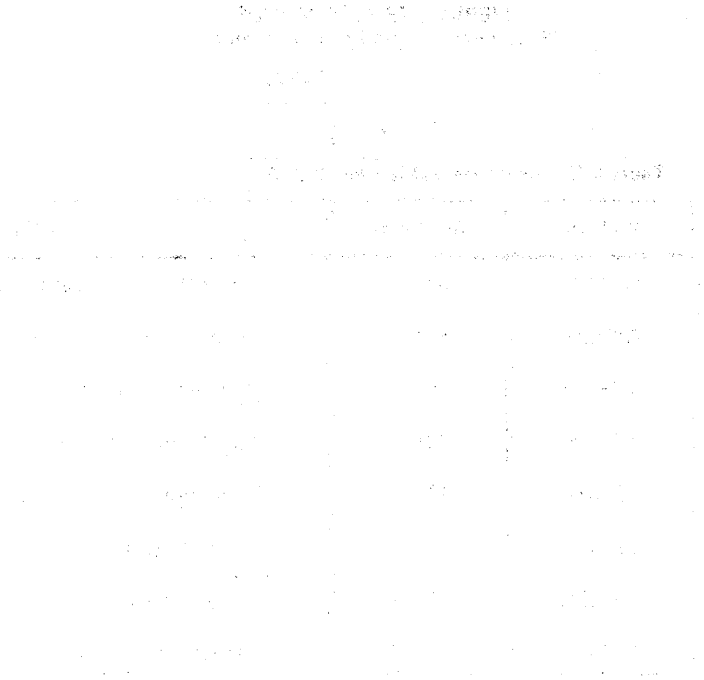
1. Address OFFH is contained in Z80 register I.
2. Interrupt mode 2 is in effect.
3. BDOS and BIOS cannot be called while the system is in interrupt routine.
4. When you exit the program, the vector table must be returned to the state that was in effect before the program was executed.
5. When using registers IX or IY in the interrupt handler, you must save the register contents.
6. Interrupts may not be turned on when the system is in another interrupt routine.
7. When returning control to the system from an interrupt routine, you must use the instruction RET.

Table 4–1. Interrupt Vector Table (Part 1 of 2)

Address	Controller	Function
OFF4OH	SCC B	Transmit buffer empty
OFF42H	SCC B	External/status change
OFF44H	SCC B	Keyboard input
OFF46H	SCC B	Keyboard input error
OFF48H	SCC A	Communications—transmit buffer
OFF4AH	SCC A	External/status empty
OFF4CH	SCC A	Communications input
OFF4EH	SCC A	Communications input error
OFF50H	STI IO	Modem ring indicator
OFF52H	STI I1	Port B interrupt
OFF54H	STI I2	Port C interrupt
OFF56H	STI I3	Cursor blink period (x8=time)
OFF58H	STI	Timer D drives RTxC A (if enabled)
OFF5AH	STI	Timer C 16x field blink, 8x cursor
OFF5CH	STI 14	Positive vertical sync. OFF 5EH
	STI 15	Modem CTS

Table 4-1. Interrupt Table (Part 2 of 2)

Address	Controller	Function
OFF60H	STI	Timer B audio (reset to zero)
OFF62H	STI	Peripheral transmit error
OFF64H	STI	Peripheral transmit buffer empty
OFF66H	STI	Peripheral receive error
OFF68H	STI	Peripheral receive buffer full
OFF6AH	STI	Timer A system timer
OFF6CH	STI I6	Port A interrupt
OFF6EH	STI 17	CTS assertion (check port 062H)



5. Keyboard Functions

5.1. CP/M PLUS MODE KEYS

The keys highlighted on the keyboard in figure 5-1 are the only operative keys in CP/M Plus mode. Keyboard definitions may be altered by using the two utilities MAKEKBD and LOADKBD. More information on these utilities is provided in the UTS 30 CP/M Plus operator's guide, UP-9839 (current version).

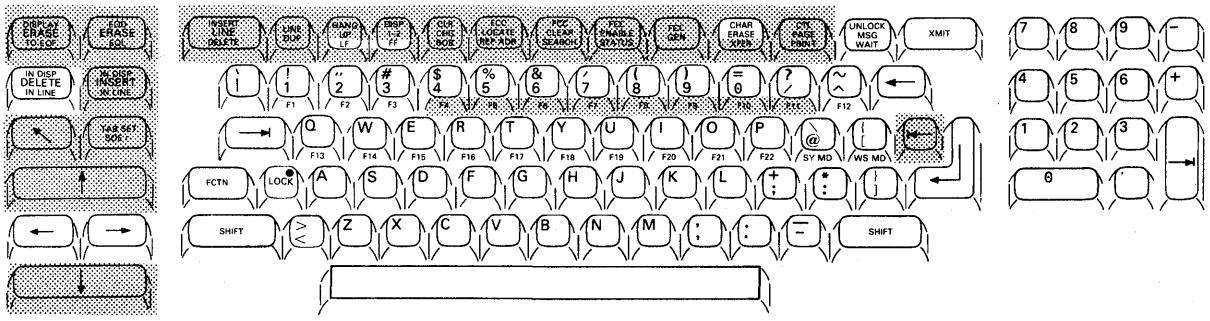
Keyboard entries made in CP/M Plus mode are of two general types: local and non-local.

- Local input is immediately interpreted by the terminal—a user program cannot interpret these keys.
- Non-local input is passed by the terminal to the operating system—a user program can interpret these keys. Other types of input, such as control sequences, require the simultaneous use of the FUNCTION key.

NOTE:

Pressing any key that is inoperative in CP/M Plus mode (those keys that are shaded in figure 5-1) will cause an invalid entry which is handled at the local level.

Figure 5–1. Keys Used in CP/M Plus Mode (Low-Profile Keyboard)



5.2. OPERATIONAL CHARACTERISTICS

Table 5-1 shows the operating characteristics of the various keyboard keys in CP/M Plus Mode.

Table 5–1. Normal Operating Characteristics of Keys in CP/M Plus Mode

Key		Local	Non-Local	Character/Code Generated	Key Function
UTS 40/ 400-Format	SPERRYLINK*				
A – Z (41 to 5A hex)	A – Z		●	A – Z	
a – z (61 to 7A hex)	a – z		●	a – z	
0 – 9 (30 to 39 hex)	0 – 9		●	0 – 9	
F1	1	●			Toggles between uppercase-only and upper/lowercase modes
F2	2	●			Toggles display output between normal and monitor modes
MSG WAIT	CANCEL/ STOP		●	Control–S	
XMIT	ENTER		●	Carriage return	
UNLOCK	CANCEL/ STOP		●	Control–C	
FF	L		●	Control–L	
LF	J		●	Control–J	
RETURN	RETURN		●	Carriage return	
Space bar	Space bar		●	Space	
SPACE			●	Space	
CHAR ERASE	SET TABS		●	Space	
BACK SPACE	BACK ERASE		●	Backspace	
TAB FORWARD	TAB		●	Control–I	
Special characters (starting at 20 hex to 7E hex)			●		
FUNCTION control sequences**			●		

*Keycap labels on the SPERRYLINK keyboard vary significantly from those on other keyboards available for the UTS 40. Refer to UP-9491 (current version).

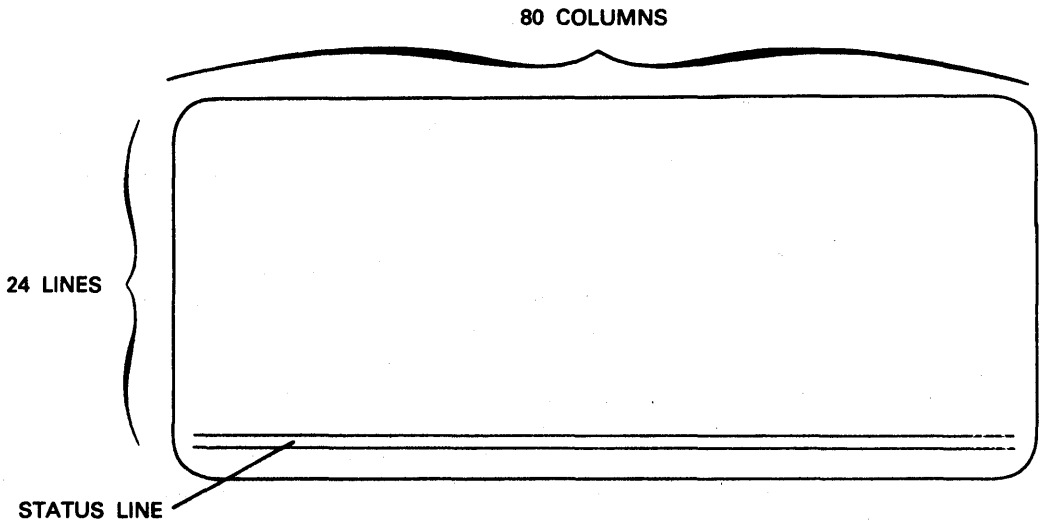
**Invalid FUNCTION sequences are handled at the local level.

6. Screen Functions for CP/M Plus Mode

6.1. SCREEN DISPLAY

The standard CP/M Plus screen display format, shown in figure 6-1, is 24 lines (rows) by 80 columns, for a total of 1920 character positions. A 25th row is used for status messages, and the display of keyboard character set names and CRT character font names.

Figure 6–1. CP/M Screen Display



Characters received from the keyboard are evaluated by the CP/M Plus system CCP or by a user program. The terminal places a character on the screen at the position occupied by the cursor, and the cursor is then advanced one position. Each character position is uniquely addressable as explained in 6.3.

The UTS 30 CP/M Plus screen functions, including escape codes, cursor positioning, screen control, and special emphasis characters, are discussed in the following paragraphs.

6.2. DEFINITIONS

The following list defines the basic elements of the escape and ANSI (American National Standards Institute) sequences used in CP/M Plus.

Default — Default is a value that is assumed for a particular function when no explicit value, or a value of 0, is specified.

Delimiter — A delimiter is a character (such as a semicolon) used to define the end of a data string or series.

FUNCTION Key — The FUNCTION key is pressed simultaneously with another key to enter a specific control character.

Escape (ESC) — ESC is a prefix used to introduce an escape sequence.

NOTE:

Since there is no ESC key on the UTS 30, ESC is generated by pressing the FUNCTION and [(left bracket) keys simultaneously.

Escape Sequence — The escape sequence is a sequence of characters, beginning with the ESC character, indicating that the sequence is to be interpreted according to a different coded character set. See figure 6–5 for an example.

Control Sequence Introducer (CSI) — CSI is a prefix used in ANSI control sequences to provide supplemental controls. It affects the interpretation of a limited number of contiguous characters. In Sperry CP/M Plus, the CSI is ESC[(ESCape followed by a left square bracket). See figure 6–4 for an example.

NOTE:

Since there is no ESC key on the UTS 30, ESC is generated by pressing the FUNCTION key and [(left bracket) key simultaneously. Therefore, CSI is generated by pressing the FUNCTION key and the [(left bracket) key simultaneously and then pressing the [key again.

Coded Character Set — A coded character set is a group of characters with predefined, one-to-one relationships between the individual characters and the codes they represent.

Final Character — The final character terminates an escape or ANSI control sequence.

Parameter — A parameter is a string of decimal characters (0 through 9) representing a single value. Leading zeros are ignored. The string may contain any number of decimal characters.

Numeric Parameter — A numeric parameter represents a number and is designated by Pn, Pr, Pc, Px, and Py.

String — A string is a series of contiguous characters.

Parameter String — A parameter string is a series of parameters separated by semicolons and designated by Ps.

Pn — Pn represents a general parameter specified by a decimal number.

Px Px represents a parameter which describes horizontal movement of the cursor along rows. It is the X-coordinate parameter.

Py Py represents a parameter which describes vertical movement of the cursor across columns. It is the Y-coordinate parameter.

NOTES:

1. *For row-related operations, the range of permissible values is from 1 to 24.*
2. *For column-related operations, the range of permissible values is from 1 to 80.*

Figure 6–2. ESC I — Escape Sequence for Reverse Line Feed

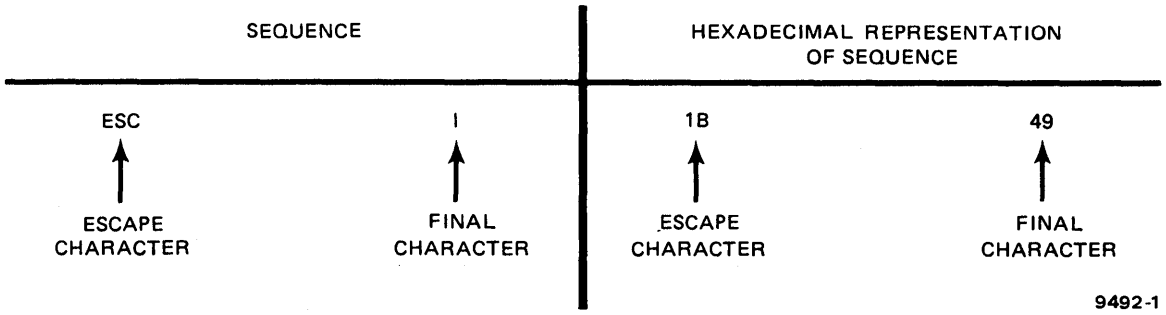
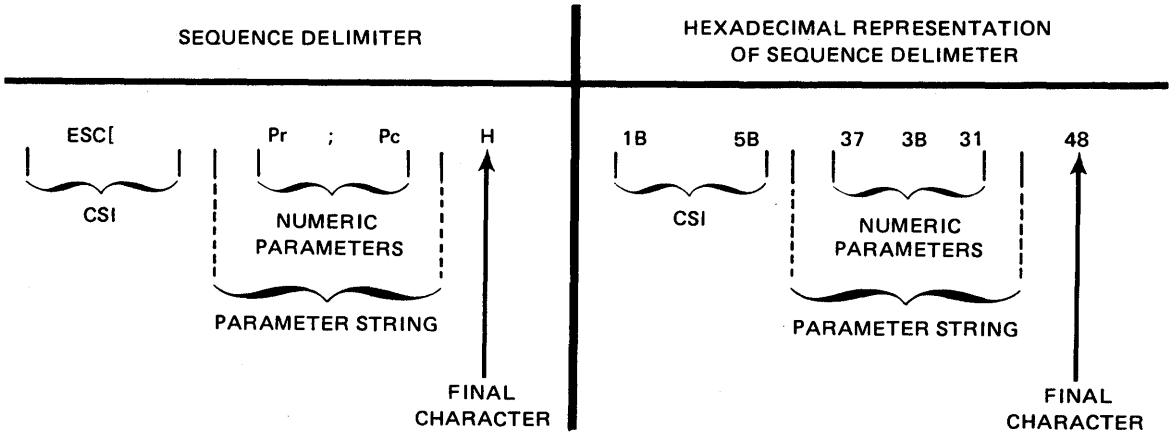


Figure 6–3. CSI Pr;Pc H — ANSI Sequence for Direct Cursor Positioning



6.3. CURSOR POSITIONING

Since text transmitted to the terminal is placed on the screen starting at the cursor position, cursor positioning is a vital feature. Several different methods may be used to position the cursor:

- Cursor Address Sequence — positions the cursor at any specified point on the screen according to row and column coordinates.
- Cursor-to-Home Code — places the cursor in the upper left corner of the screen.
- Cursor Return Code — positions the cursor relative to its last position.
- Cursor Scan Codes — position the cursor relative to its last position.

The cursor positioning sequences (summarized in table 6-1) are discussed in the following paragaraphs.

Table 6–1. Cursor Positioning Sequences

Function	Code	ANSI Standard
Cursor Address Sequence	ESC Y Pr Pc	CSI Py; Px H CSI Py; Px f
Cursor-to-Home	ESC H	CSI H
Cursor Return	CR	CR
Scan Up	ESC A	CSI Pn A
Scan Down	ESC B	CSI Pn B
Scan Right	ESC C	CSI Pn C
Scan Left	ESC D	CSI Pn D
Line Feed	LF	LF
Reverse Line Feed	ESC I	

6.3.1. Cursor Address Sequence

The format for direct cursor positioning is as follows:

ESC Y Pr Pc

CSI Py; Px H

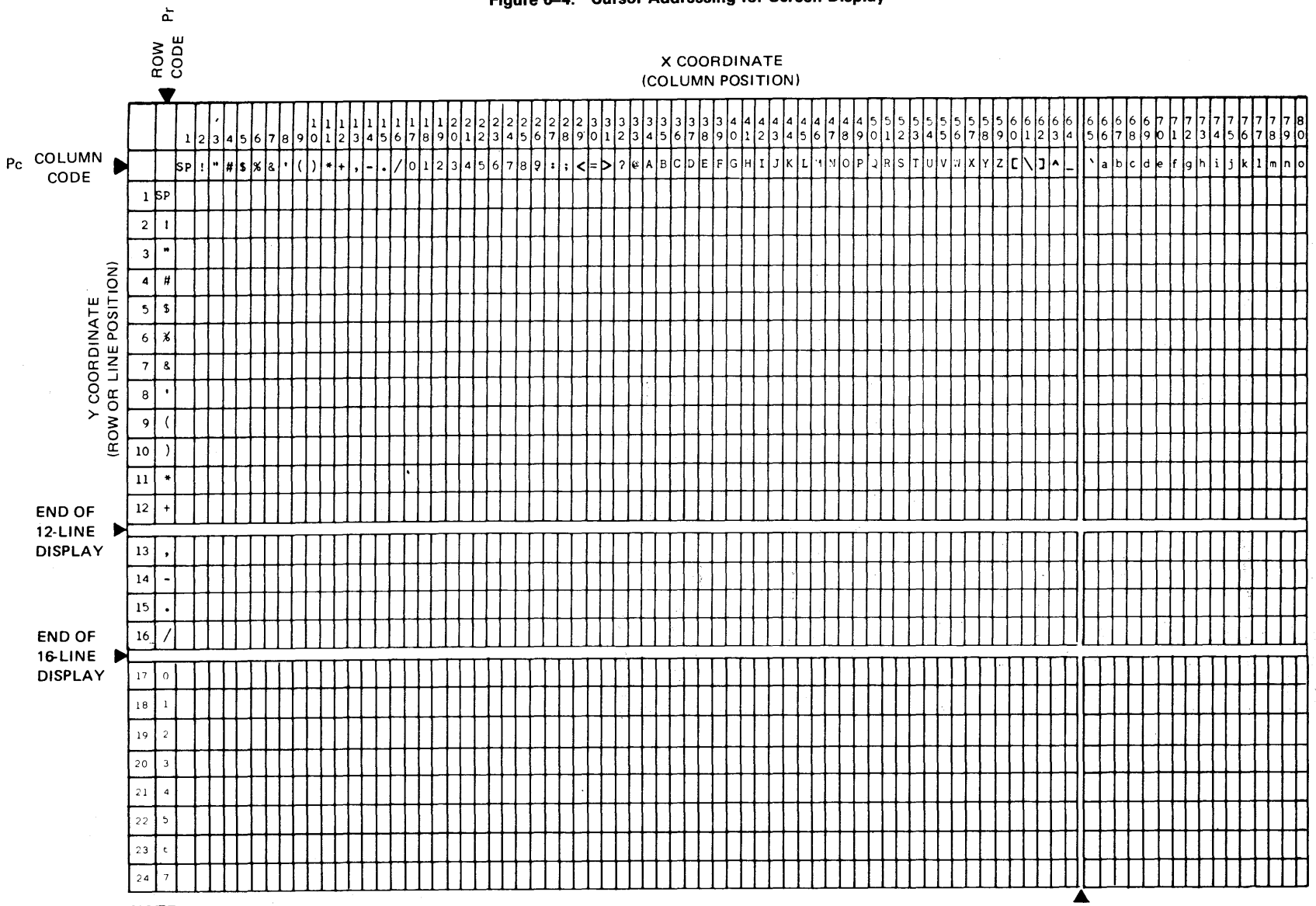
CSI Py; Px f

The following list defines the basic elements of the cursor address sequence.

- **ESC (Escape)** — prefix indicating that the following characters are part of an escape sequence.
- **CSI** (Control Sequence Indicator) — prefix indicating that the following characters are part of an ANSI sequence.
- **Y** — indicates that the next two characters contain the cursor address.
- **Pr** — Y coordinate identifying the horizontal line (row) on the screen where the cursor is to be placed.
- **Pc** — X coordinate identifying the vertical column on the screen where the cursor is to be placed.
- **Py** — Y coordinate identifying the horizontal line (row) on the screen where the cursor is to be placed (range of Py is 24).
- **Px** — X coordinate identifying the vertical column on the screen where the cursor is to be placed.
- **H** or **f** — indicates that the previous two characters, separated by a delimiter, are the cursor address.

The X coordinates (Pc and Px) and Y coordinates (Pr and Py) can be determined from figure 6-4. The row and column codes shown in figure 6-4 are derived from columns 2 through 6 of the ASCII code chart (appendix A). Columns 0 and 1 of the ASCII code chart contain control codes and are not used for cursor address codes. Column 7 is also not used because the terminal does not allow for an address greater than 80.

Figure 6–4. Cursor Addressing for Screen Display



NOTE:

The addressing sequence will always be:
Y position (lines 1 through 24),
then X position (columns 1 through 80).

6.3.2. Cursor Address Codes

Cursor-to-Home

ESC H or **CSI H** — moves the cursor to the first character position of the first line in the upper-left corner of the screen (row 1, column 1). This is the home position.

Cursor Return

CR — returns the cursor to the first character on the current line.

Line Feed

FUNCTION J — moves the cursor down one line in the same column (essentially the same as ESC B).

Reverse Line Feed

ESC I — moves the cursor up one line in the same column. If the cursor is at the top margin, an insert line function is performed (essentially the same as ESC A).

Scan Up

ESC A — moves the cursor up one line in the same column.

CSI Pn A (default value: 1) — moves the cursor upward within the same column. The number of lines moved is determined by the parameter Pn. If the parameter value is 0 or 1, the cursor moves one line upward. If the parameter value is n, the cursor moves n lines upward.

Scan Down

ESC B — moves the cursor down one line in the same column.

CSI Pn B (default value: 1) — moves the cursor downward within the same column. The number of lines moved is determined by the parameter Pn. If the parameter value is 0 or 1, the cursor moves one line downward. If the parameter value is n, the cursor moves n lines downward.

Scan Right

ESC C — moves the cursor one character position to the right. If the cursor is in the last position of a line, it moves to the first character position of the following line. If the cursor is in the last position of the last line, the screen is scrolled up one line and the cursor is moved to the first position of the new last line.

CSI Pn C (default value: 1) — moves the cursor to the right. The distance moved is determined by the parameter Pn. A parameter value of n moves the cursor n positions to the right. If the parameter moves the cursor past the end of a line, the cursor moves to the first character position of the next line and continues to the right until the cursor has moved a total of n spaces.

Scan Left

ESC D — moves the cursor one character position to the left. If the cursor is in the first position of a line, it moves to the last character position of the previous line. If the cursor is in the first position of the first line, the screen is scrolled down one line and the cursor is moved to the last position of the new first line.

CSI Pn D (default value: 1) — moves the cursor to the left. The distance moved is determined by the parameter Pn. If the parameter moves the cursor past the beginning of a line, the cursor moves to the last character position of the previous line and continues to the left until the cursor has moved a total of n spaces.

6.4. SCREEN CONTROL

CP/M Plus provides you with many additional types of screen control as well as basic data entry. The basic screen control functions include:

- clear
- delete
- insert

In addition to these screen functions, CP/M Plus provides special emphasis functions including:

- underline
- column separator (vertical line to the left of a character)
- strike-through
- normal
- blink
- reverse video

These functions are described in detail in the following paragraphs.

6.4.1. Basic Screen Control Functions

You can clear various portions of your screen by using the following clear functions.

Clear Screen

ESC E — erases the entire screen, resets the scroll range as lines 1 through 24, and places the cursor in the home position. Screen colors return to their originally configured (default) values.

Clear in Screen

ESC J — erases from the cursor (including the cursor position) to the end of the screen.

CSI Pn J (default value: 0) — erases all or some of the characters in the display according to the option parameter Pn.

Pn Value	Parameter Meaning
0	Erase from the cursor to the end of the screen, including the cursor position.
1	Erase from the beginning of the screen to the cursor, including the cursor position.
2	Erase the display; all lines are erased, and the cursor does not move.

Clear in Line

ESC K — erases from the cursor (including the cursor position) to the end of the line.

CSI Pn K (default value: 0) — erases some or all characters in the current line according to the option parameter Pn.

Pn Value	Parameter Meaning
0	Erase from the cursor to the end of the line, including the cursor position.
1	Erase from the beginning of the line to the cursor, including the cursor position.
2	Erase the entire line; the cursor does not move.

You can insert and delete lines and characters using the following insert and delete functions.

Delete Line

ESC L — deletes the line that the cursor is in, moves all the following lines up one line, and adds a blank line at line 24.

CSI Pn M (default value: 1) — deletes the line that the cursor is in, moves all the following lines up one line, and adds a blank line at line 24. If Pn is greater than 1, the procedure is repeated Pn times. Otherwise, the procedure is executed only once.

Delete Character

ESC M — deletes the character at the cursor position, shifts existing text to the right of the cursor (on the cursor line) one character position to the left, and adds a blank character at column position 80 of the current line.

CSI Pn P (default value: 1) — deletes the character at the cursor position, shifts existing text to the right of the cursor (on the cursor line) to the left, and adds blank characters at the end of the line. If Pn is greater than 1, the procedure is repeated Pn times. Otherwise, the procedure is executed only once.

Insert Line

ESC N — inserts a new blank line by moving the line that the cursor is in and all following lines, down one line. The cursor is moved to the beginning of the new blank line.

CSI Pn L (default value: 1) — inserts a new blank line by moving down the line that the cursor is in, as well as all following lines. If Pn is greater than 1, the procedure is repeated Pn times. The cursor is moved to the beginning of the new blank line.

Insert Character

ESC O — inserts a blank character at the cursor position by shifting existing text on the current line one character position to the right. The cursor does not move.

CSI Pn @ (default value: 1) — If Pn is greater than 1, blank characters are inserted at the cursor position by shifting existing text on the current line Pn character positions to the right. As each blank is inserted, the character at the end of the current line is lost. If Pn is not greater than 1, the sequence has the same effect as ESC O.

6.4.2. Special Emphasis Functions

The special emphasis functions include:

- normal display (no emphasis)
- reverse-video display
- underline
- strike-through
- column separator
- blink

Normal Display (no emphasis)

ESC P @ or **ESC Q** — places a bright character on a dark background. The corresponding ANSI sequence is **CSI 0 m** or **CSI m**.

Reverse-Video Display

ESC P H — places a dark character on a bright background. The corresponding ANSI sequence is **CSI 7 m**.

Underline

ESC P B — places an underline beneath a character. The underline is the full width of the character and connects with adjacent underlines to form a continuous line. The corresponding ANSI sequence is **CSI 4 m**.

Strike-through

ESC P D — places a horizontal line through the middle of a character. The strike-through is the full width of the character and connects with adjacent strike-throughs to form a continuous line. The corresponding ANSI sequence is **CSI 21 m**

Column Separator

ESC P P — places a vertical line to the left of a character. This line is the full length of the character. A continuous vertical line can be created by using column separators in contiguous rows of the same column. The corresponding ANSI sequence is **CSI 20 m**.

Blink

ESC P I — places a blinking foreground color character on the background color. The corresponding ANSI sequence is **CSI 5 m**.

6.4.3. Emphasis Character Sequence

Emphasis characters are transmitted to the terminal by a normal escape (ESC) or ANSI control sequence indicator (CSI) sequence in the following format:

ESC P <emphasis> CSI P1;...;Pn m

where:

<emphasis> = ASCII character or hexadecimal code corresponding to the desired emphasis (see table 6-2)

P1-Pn = 0 no emphasis
4 underline
5 blink
7 reverse video
20 column separator
21 strike-through

The special emphasis characters and the ASCII characters and hexadecimal codes which can be used in normal escape sequences are provided in table 6–2. An example showing the special emphasis byte is provided in figure 6–5.

Table 6–2. ASCII Characters Used in Emphasis Sequences (Part 1 of 2)

ASCII Char.	Hex	Binary Code 6543210	Special Emphasis
@	40	1000000	Normal intensity
A	41	1000001	Low intensity
B	42	1000010	Underline
C	43	1000011	Underline and low intensity
D	44	1000100	Strike-through
E	45	1000101	Strike-through and low intensity
F	46	1000110	Underline and strike-through
G	47	1000111	Underline, strike-through and low intensity
H	48	1001000	Reverse video
I	49	1001001	Dim background
J	4A	1001010	Reverse video and underline
K	4B	1001011	Dim background and underline
L	4C	1001100	Reverse video and strike-through
M	4D	1001101	Dim background and strike-through
N	4E	1001110	Reverse video, strike-through and underline
O	4F	1001111	Dim background, strike-through and underline
P	50	1010000	Column separator
Q	51	1010001	Column separator and low intensity

Table 6–2. ASCII Characters Used in Emphasis Sequence (Part 2 of 2)

ASCII Char.	Hex	Binary Code 6543210	Special Emphasis
R	52	1010010	Column separator and underline
S	53	1010011	Column separator, underline and low intensity
T	54	1010100	Column separator and strike-through
U	55	1010101	Column separator, strike-through and low intensity
V	56	1010110	Column separator, strike-through and underline
W	57	1010111	Column separator, strike-through, underline and low intensity
X	58	1011000	Column separator and reverse video
Y	59	1011001	Column separator and dim background
Z	5A	1011010	Column separator reverse video and underline
[	5B	1011011	Column separator, dim background and underline
]	5D	1011101	Normal intensity
^	5E	1011110	Normal intensity
—	5F	1011111	Low intensity

NOTE:

Bit 7 is always 0 to indicate the ASCII character set.

Figure 6–5. An Example of Expanded Special Emphasis Byte

Character	Hex Code	Binary	Meaning
B	42	100 0010 MSB LSB	Underline

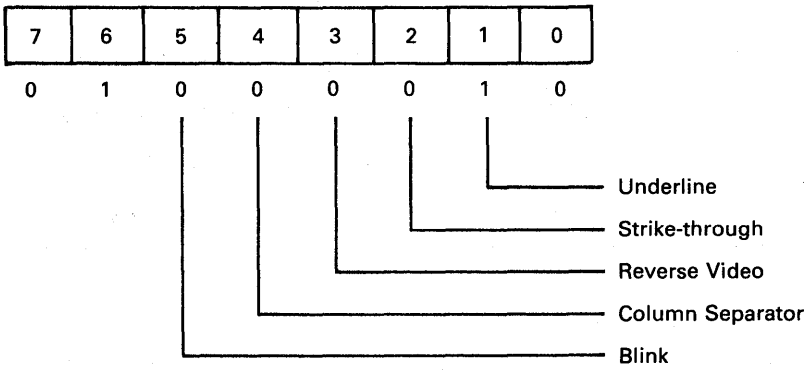


Figure 6–6 provides some examples of combined special emphasis characters.

Figure 6–6. Combined Special Emphasis Characters

	Input	Screen Display
Column Separator and Underline	ESC P R or CSI 4 ; 20 m	Makes a corner (⌊‾)
Column Separator and Strike-through	ESC P T or CSI 20 ; 21 m	Makes a sideways T (⊢—)

Changing Special Emphasis

To change any given emphasis, simply send another special emphasis sequence to the terminal.

Special emphasis instructions may also be changed by sending (ESC) sequences:

ESC a <emphasis> to set a specific emphasis

or

ESC b <emphasis> to clear a specific emphasis

where:

<emphasis> = a 1 byte code specifying the emphasis attributes to be turned on or off (see table 6–2).

The format of this byte is shown in figure 6–5. A bit set to one indicates that the emphasis is to be changed. A bit set to zero leaves the corresponding emphasis unchanged. These two escape sequences alter the existing emphasis.

NOTE:

Changing a particular emphasis has no effect if that emphasis has already been set to the value you desire. However, ESC P @, CSI m, ESC Q, and CSI 0 m are exceptions to the rule. These sequences clear all special emphasis functions.

6.4.4. Other Screen Functions

The following screen functions are also available.

Cursor Disable

ESC R — turns off the cursor. If the cursor is already turned off, the sequence has no effect.

Cursor Enable

ESC S — turns on the cursor. If the cursor is already turned on, the sequence has no effect.

Save Cursor Position

ESC W — saves the current cursor position, emphasis mode, and loadable character set status.

For example, assume that the cursor is at row 12 column 3 with the reverse video on. If the escape sequence to save the cursor position is input, the terminal will save the cursor position (row 12 column 3), the emphasis mode (reverse video), and the loadable character set status (standard).

A second save destroys the first save.

Restore Cursor to Last Save Position

ESC X — restores the cursor to the last saved cursor position. The sequence also restores the emphasis mode and the status of the loadable character set.

For example, assume that the current cursor position is at row 14 column 30 with normal emphasis. By sending the restored escape sequence to the terminal (using the example introduced under Save Cursor Position), the cursor will move immediately to row 12 column 3 with the reverse video emphasis in effect.

The restore escape sequence has no effect if a save escape sequence was not previously sent to the terminal. A restore sequence repeatedly restores the same save sequence until another save sequence is sent to the terminal.

Screen Alignment Display

ESC T — fills the entire screen with uppercase E's for screen focus and alignment.

Set Top and Bottom Margins

ESC U Pt Pb — sets the top and bottom margins. These margins define the scrolling region.

The first parameter (Pt) is the line number of the first line in the scrolling region; the second parameter (Pb) is the line number of the bottom line in the scrolling region. The minimum allowable size of the scrolling region is two lines.

In other words, the top margin must be less than the bottom margin. The cursor is always placed in the first position of the scroll area.

Pt is the Y coordinate identifying the horizontal line (row) where the top margin is to be set (see figure 5-3). Pb is the Y coordinate identifying the horizontal line (row) where the bottom margin is to be set (see figure 6- 3).

Reset Top and Bottom Margins

ESC V — sets the top margin to 1 and the bottom margin to 24. The cursor is placed in the home position.

Clear Status Line

ESC \ — blanks out the first 40 positions on the 25th line. This line shows up on the screen in the originally configured foreground/background colors.

Enter Data on Status Line

ESC] — saves the current cursor position and enters any characters received on the 25th line. When a carriage return is received, the terminal restores the saved cursor position. When the 40th character is received by the terminal, the cursor is restored to the saved position.

Select Alternate Character Set

ESC F — initiates the alternate character set. Subsequent displayable characters have their character fonts (displayable dot pattern) taken from the alternate loadable character font set (see UTS 30 CP/M Plus operator's guide, UP-9839, current version) for further explanation of the loadable character set function.

Return to Standard Character Set

ESC d — has an effect only after an **ESC F** has been used. **ESC F** disables strike-through emphasis as well as enabling the optional character set memory. **ESC d** reenables strike-through emphasis as well as disabling the displaying of the optional character set on the screen.

Select Standard Character Set

ESC G — causes subsequent displayable characters to be taken from the standard character font set.

Appendix A. ASCII Chart

CONTROL CHARACTERS				DATA CHARACTERS															
				64 CHARACTERS UPPERCASE								32 CHARACTERS LOWERCASE							
0		0		0		0		1		1		1		1		7			
0		1		1		1		0		0		1		1		6			
0		0		0		1		0		1		0		1		5/4	3	2	1
00 00 000	NUL	10 16 020	DLE	20 32 040	SP	30 48 060	0	40 64 100	@	50 80 120	P	60 96 140	`	70 112 160	p	0	0	0	
01 01 001	SOH	11 17 021	DC1	21 33 041	!	31 49 061	1	41 65 101	A	51 81 121	Q	61 97 141	a	71 113 161	q	0	0	0	1
02 02 002	STX	12 18 022	DC2	22 34 042	"	32 50 062	2	42 66 102	B	52 82 122	R	62 98 142	b	72 114 162	r	0	0	1	0
03 03 003	ETX	13 19 023	DC3	23 35 043	#	33 51 063	3	43 67 103	C	53 83 123	S	63 99 143	c	73 115 163	s	0	0	1	0
04 04 004	EOT	14 20 024	DC4	24 36 044	$	34 52 064	4	44 68 104	D	54 84 124	T	64 100 144	d	74 116 164	t	0	1	1	0
05 05 005	ENQ	15 21 025	NAK	25 37 045	%	35 53 065	5	45 69 105	E	55 85 125	U	65 101 145	e	75 117 165	u	0	1	0	1
06 06 006	ACK	16 22 026	SYN	26 38 046	&	36 54 066	6	46 70 106	F	56 86 126	V	66 102 146	f	76 118 166	v	0	1	0	1
07 07 007	BEL	17 23 027	ETB	27 39 047	'	37 55 067	7	47 71 107	G	57 87 127	W	67 103 147	g	77 119 167	w	1	1	1	1
08 08 010	BS	18 24 030	CAN	28 40 050	(	38 56 070	8	48 72 110	H	58 88 130	X	68 104 150	h	78 120 170	x	1	0	0	0
09 09 011	HT	19 25 031	EM	29 41 051	)	39 57 071	9	49 73 111	I	59 89 131	Y	69 105 151	i	79 121 171	y	1	0	0	1
0A 10 012	LF	1A 26 032	SUB	2A 42 052	*	3A 58 072	:	4A 74 112	J	5A 90 132	Z	6A 106 152	j	7A 122 172	z	1	0	1	0
0B 11 013	VT	1B 27 033	ESC	2B 43 053	+	3B 59 073	;	4B 75 113	K	5B 91 133	[	6B 107 153	k	7B 123 173	{	1	0	1	1
0C 12 014	FF	1C 28 034	FS	2C 44 054	,	3C 60 074	<	4C 76 114	L	5C 92 134	\	6C 108 154	l	7C 124 174	|	1	1	0	0
0D 13 015	CR	1D 29 035	GS	2D 45 055	-	3D 61 075	=	4D 77 115	M	5D 93 135	]	6D 109 155	m	7D 125 175	}	1	1	0	1
0E 14 016	SO	1E 30 036	RS	2E 46 056	.	3E 62 076	>	4E 78 116	N	5E 94 136	^	6E 110 156	n	7E 126 176	~	1	1	1	0
0F 15 017	SI	1F 31 037	US	2F 47 057	/	3F 63 077	?	4F 79 117	O	5F 95 137	_	6F 111 157	o	7F 127 177	DEL	1	1	1	1

NOTE: HEXADECIMAL — ASCII

DECIMAL — 41 A 65 101

OCTAL

READER'S COMMENT SHEET

Your comments on this manual will help us improve it. Please fill in the requested information.

Name of manual: _____

Manual number: UP-_____ revision number_____, including update numbers _____

Name of your company: _____

Address of company: _____

What is your position? _____

Your level of experience: Professional_____ Knowledgeable_____ Novice _____

With what system is the equipment used? _____

How do you use this manual?

As a reference source ☐ As a self-instructional text ☐

As a classroom text ☐ As _____ ☐

Please rate this manual

As a reference source:	Good ☐	Adequate ☐	Not adequate ☐
As a text:	Good ☐	Adequate ☐	Not adequate ☐
For other uses:	Good ☐	Adequate ☐	Not adequate ☐

Add your specific comments. Give page and paragraph references where appropriate.

Thank you for your cooperation.

PLEASE SEND US YOUR COMMENTS!

We feel that this manual is an essential part of our equipment. We want to be sure it is the best, most usable manual possible. Your comments will help us achieve this goal. When you have become familiar with the manual, please fill in the other side of this form and mail the form to us. Your reply will be carefully reviewed by the persons responsible for writing and publishing this manual.

FOLD

FIRST CLASS
PERMIT NO. 2540
SALT LAKE CITY,
UTAH

BUSINESS REPLY MAIL NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

CUT

POSTAGE WILL BE PAID BY

SPERRY

322 NORTH 2200 WEST
SALT LAKE CITY, UTAH 84116

ATTN: MANAGER, GSD PUBLICATIONS

FOLD

NOTE:

Requests for copies of this manual and other Sperry publications and for assistance in getting the most use out of your Sperry equipment should be directed to your local Sperry representative.