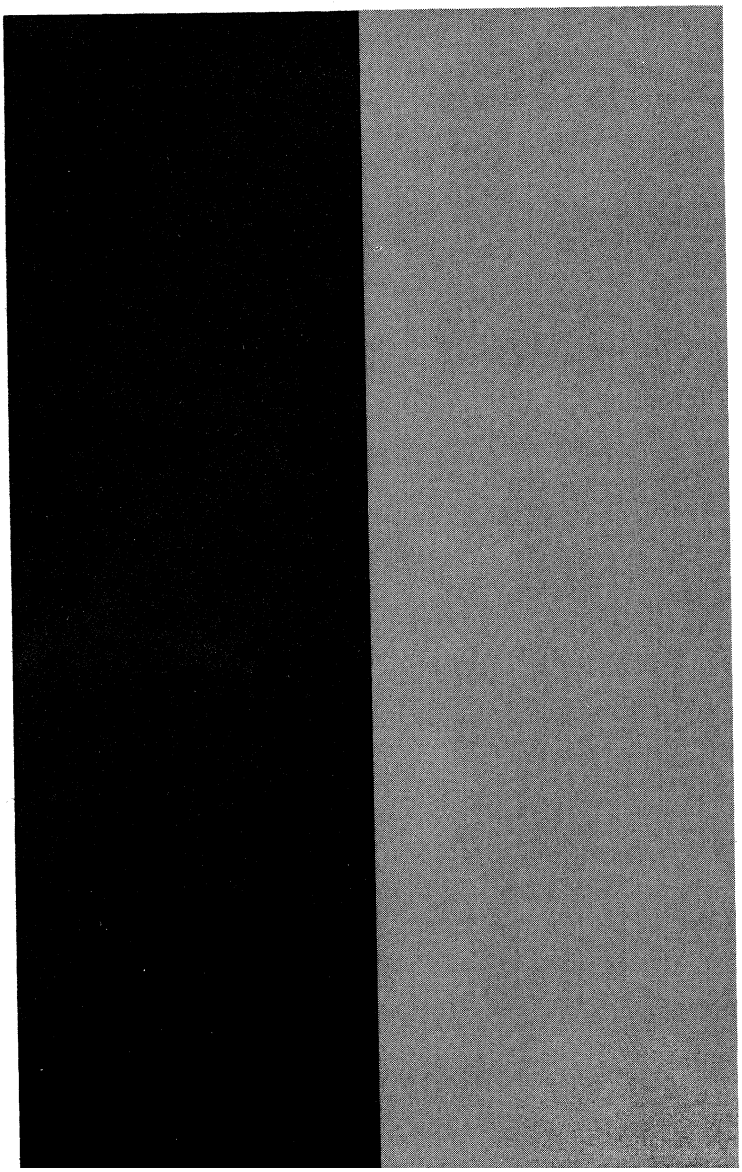
SPERRY RAND

UNIVAC

COMMUNICATIONS/ SYMBIONT PROCESSOR



SYSTEM
DESCRIPTION

UP-7850

This document contains the latest information available at the time of publication. However, the Univac Division reserves the right to modify or revise its contents. To ensure that you have the most recent information, contact your local Univac Representative.

UNIVAC is a registered trademark of the Sperry Rand Corporation.

Other trademarks of the Sperry Rand Corporation in this publication are:

UNISCOPE
UNISERVO

PRINTED IN U.S.A.

CONTENTS

APPENDIXES

FIGURES

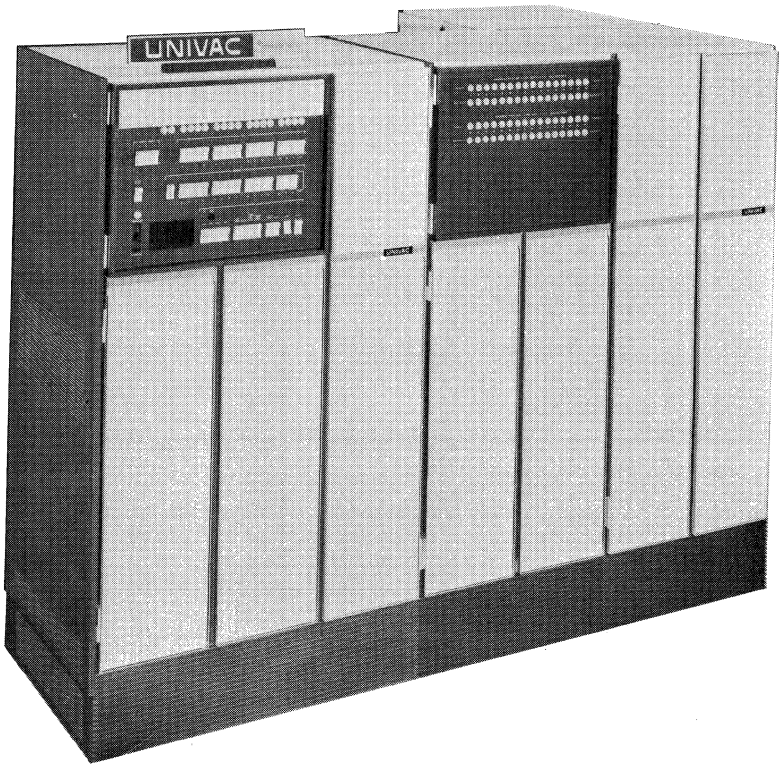
TABLES

I. INTRODUCTION

1.1. GENERAL

The UNIVAC Communications/Symbiont Processor (C/SP) is a high-performance, internally-programmed communications concentrator/multiplexer which is intended for use as an onsite communications subsystem. The C/SP (see Figure 1–1) incorporates a unique combination of hardware and software to control low-speed and high-speed communications lines and to provide an interface with a central computer. With the C/SP assuming control of the communications function, the host computer is relieved of the necessity of maintaining complex handlers and subroutines which are required when operating with a wide variety of terminal devices and communications lines.

Operating as an onsite multiplexer or preprocessor, the UNIVAC C/SP functions as a terminus for many communications lines over which data is transmitted to the central processor site. The characteristics of the lines may vary from the familiar telephone lines used for telegraphic messages at 10 characters per second, to the ultra-high speed, broad-band lines operating at rates of 50,000 bits per second. Each of these lines can have one or more terminal devices which transmit and/or receive data.



The individual line and terminal configurations operate under a variety of line procedures, code sets, control signals, and timing constraints. The C/SP, through a unique combination of software controlled hardware, has the capability of controlling these lines more efficiently than most small computing systems. The C/SP responds to the variable characteristics and requirements of each line to recognize control character sequences and perform data manipulation as required by the controlling user program. By performing these activities, the central computer is permitted to dedicate the major portion of its time to the primary task of problem solution (preprocessor mode).

When used as a symbiont processor, the C/SP controls the transfer of all data between peripheral subsystems and the host computer. This I/O control is accomplished through subroutines (symbionts) which run concurrently with the main program within the host computer. By using the C/SP in this application, the system interface to the I/O devices is simplified and all I/O symbionts are removed from the host computer. This results in a saving of both time and host computer primary storage and, consequently, a reduction in system overhead.

The C/SP is useful in any application which requires control of a large number of remote terminals or where there is a complex mixture of lines and terminals (for example, brokerage houses, communications companies, service bureaus, airlines, and large manufacturers).

The salient features of the C/SP are:

- Modular design
- Communications-oriented channels
- Communications and peripheral symbiont software
- 32K to 131K bytes of plated-wire storage
- 630 nanoseconds read/write storage cycle
- Storage protection
- Sixteen 32-bit word general purpose registers, external to storage
- Half-word (16 bits and 2 parity) basic data path
- 52 half-word and word instructions
- Multilevel interrupts

1.2. HARDWARE DESIGN

The C/SP hardware was designed with modularity and flexibility as primary goals. Realizing the wide implications of designing a multifunction subsystem, special emphasis was placed on high volume throughput. Special channels were designed to accommodate, with a high degree of efficiency, the varying needs of prime peripherals and of communications terminals. The C/SP configurator (see Figure 1–2) includes the following three units:

(1) C/SP Unit

- Processor
- 16 general purpose registers
- Maintenance panel
- Interval timer
- Special device channel (communications application)
- Multiplexer channel (symbiont processor application)
- 1100 series adapter channel
- Storage protection

(2) Storage Unit (one or two)

- 65K bytes maximum storage (each unit)

(3) General purpose communications channel unit (GPCC Unit)

- Asynchronous timing assembly (maximum of seven frequencies)
- Asynchronous timing assembly expansion
- 64 locations for communications line terminals (CLTs)
- Basic multiplexer positions (maximum of eight)
- Multiplexer position expansion (to 64 maximum in increments of eight)
- CLT types as required by the network
- Dialing adapters
- GPCC CLT expansion to 64 locations

Flexibility of the C/SP hardware and software permits operation with virtually any type code to eight bits. Integrated hardware and software control of buffer elements enables handling of messages varying in length, integrity check method, and control code recognition pattern. If growth involves devices and line disciplines which are already functioning, no additional software is required; if new types of devices or line disciplines are added, additional subroutines and software can easily be generated to accommodate the new line disciplines through provisions offered in the standard software. A user-oriented assembler with a control word compiler and a C/SP simulator is part of the standard software when used with a UNIVAC 1100 Series System.

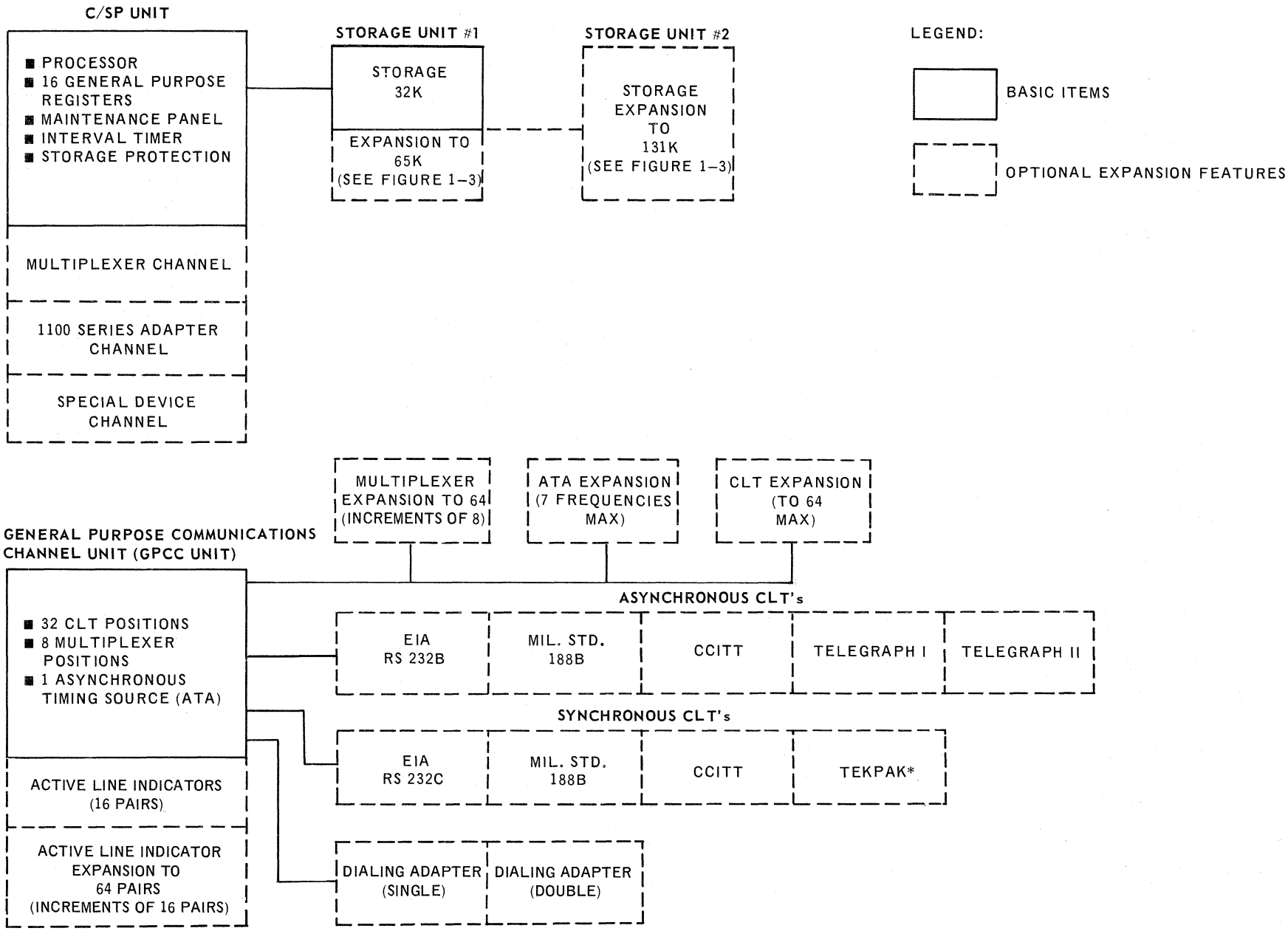


Figure 1–2. C/SP Configurator

*Registered trademark of AT&T.

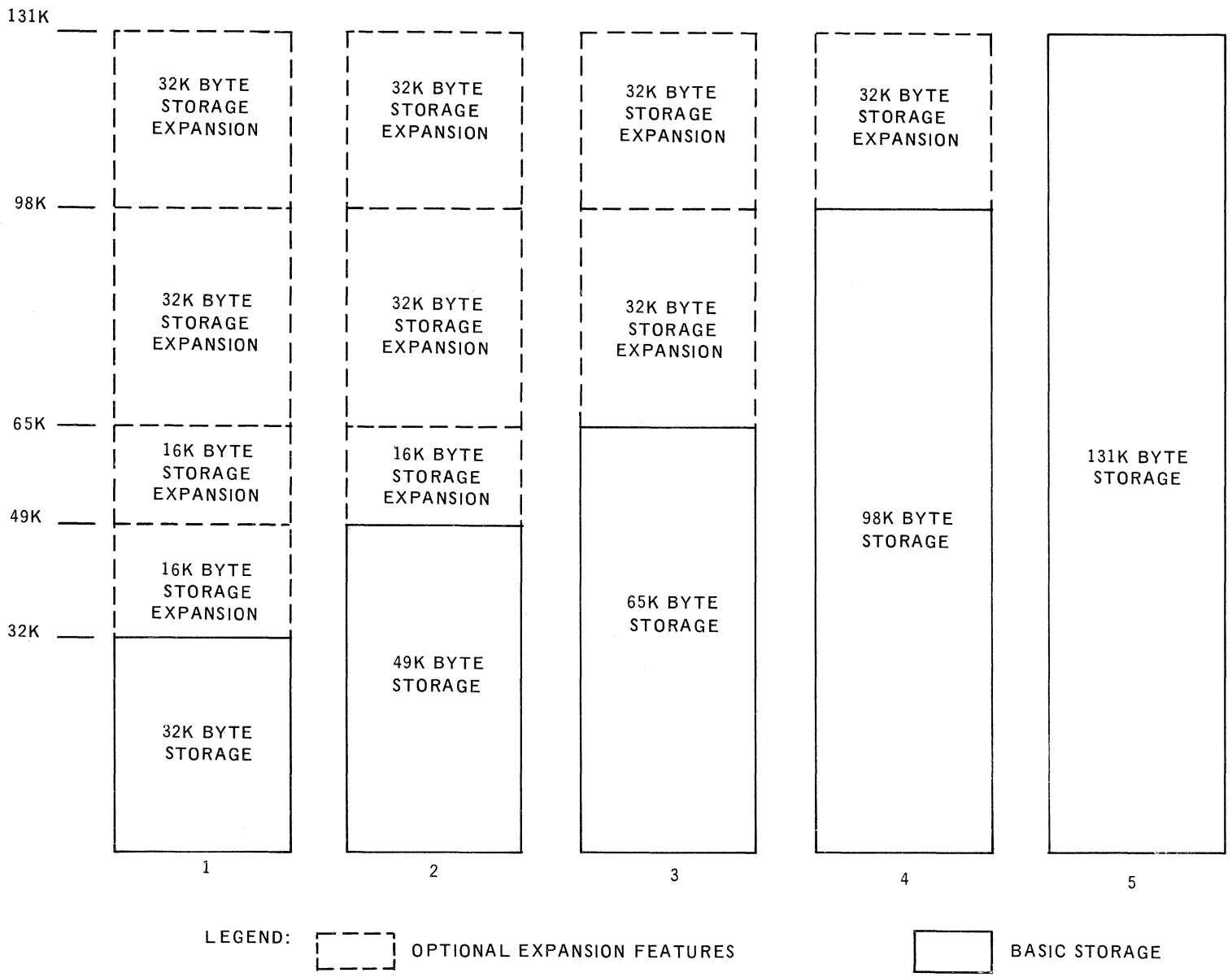


Figure 1–3. C/SP Storage Configurations

All information transfers into and out of the C/SP are handled by a maximum of seven channels (designated zero through six). A channel controls the operation of input/output devices and the transfer of data between devices and storage. Among the outstanding features of the C/SP channels are:

- Direct interface to storage
- Independent operation
- Simultaneous operation
- One or two general purpose communications channels (GPCC), each capable of controlling a maximum of 64 communications lines
- Priority interchangeability

The C/SP has four channel types:

(1) Special device channel (SDC) – The primary function of the SDC is to provide the means for local program loading and maintenance using a serial 80-column, 80-card per minute card reader device.

(2) 1100 series adapter channel (sometimes referred to as the intercomputer adapter (ICA) channel) – The 1100 series adapter channel provides an interface for direct connection of the C/SP to an I/O channel of an 1100 series system. The maximum transfer rate is in excess of 300,000 (36 bit) words per second.

(3) Communications channel – The communications channel is contained in the GPCC and is the link between the main storage and the communications line terminals (CLTs). Other elements in the GPCC perform necessary operations such as multiplexing the various CLTs, recognizing special characters, checking character parity, and coordinating all data transfers to and from storage.

The principal function of the GPCC is to multiplex a single channel of the C/SP by as many as 64 data paths, each connected to communications lines.

(4) Multiplexer Channel – The multiplexer channel provides an interface for eight control units associated with UNIVAC 9000 series computers. The maximum transfer rate is more than 80K bytes per second.

Overall characteristics of the C/SP are provided in Table 1–1.

CHARACTERISTICS	
Processor:	
Instruction set	52 (13-RR, 22-RX, 7-RS, 10-SI) (see 2.3.4)
Basic data path	Half-word (16 bits)
Basic add instruction (RX)	2.52 microseconds
Binary add instruction (RR)	1.26 microseconds
Addressable increments	8 bits, 16 bits, 32 bits
Registers:	
Number	16
Length	32 bits
Type	General purpose (hardware)
Internal word size	16 bits
Priority interrupt levels	5
Priority controls	Program execution, I/O interrupts, and data transfer
Storage:	
Type	Plated wire, nondestructive read; write
Data format	Byte – 8 bits, 1 parity Half-word – 2 bytes Word – 4 bytes
Parity	Odd (1 parity bit per byte)
Capacity	32,768 bytes min. 131,072 bytes max.
Cycle time	630 nsec. read/write cycle 630 nsec. partial write
Addressing	Direct and zero time indexing
Storage protection	Yes

Table 1–1. C/SP Characteristics (Part 1 of 2)

CHARACTERISTICS	
Channels:	
Number	7 (0 through 6)
Priority	0 through 6, consecutively, with 0 having the highest priority; channel 6 is dedicated to the SDC
Transfer rate	In excess of 1.5 MHz
Types:	
1100 Series adapter channel	
Interface	UNIVAC 1100 series computers
Transfer rate	300K words per second
Word size	36-bit
Transfer modes	Binary, byte, six-bit character
Special device channel (SDC)	
Interface	Card reader
Data rate	80 CPM
Card type	80 column
Multiplexer channel	
Interface	Byte-oriented peripherals
Number of devices	8 control units, 8 devices per control unit
Transfer rate	85K bytes per second
GPCC:	
Lines per controller	32 full duplex or 64 half duplex
Line type	Asynchronous or synchronous
Line rate	50K bps max.
Multiplexer rate	50K bytes/sec. max.
Number per system	Two
Priority levels	64 per GPCC
Interrupt control	Automatic tabling – 4 individually Controlled table segments

Table 1–1. C/SP Characteristics (Part 2 of 2)

1.3. SOFTWARE SUPPORT

Software support for the C/SP falls into the following three categories:

(1) programs which operate under control of the operating system of the UNIVAC 1100 series system;

(2) modifications and additions to the host computer operating system;

(3) C/SP resident programs which are assembled, collected, and loaded by the host computer at system generation time.

A listing of the software package available with the C/SP is provided in Table 1–2.

CATEGORY	PROGRAM OR ROUTINE
Programs under host computer operating system control	C/SP Assembler C/SP Element collector C/SP Program test simulator C/SP Diagnostic routines
Modification and addition of host computer operating system elements	C/SP Service routines including: (a) Initial load of C/SP (b) Program load of C/SP (c) Logging of data for C/SP (d) Console communications C/SP Symbionts including: (a) Input symbiont (b) Output Symbiont Interface (C/SP handler) Executive return additions C/SP Parameter table additions C/SP Interrupt answering routine C/SP Interrupt processing routine
C/SP resident programs	Intercomputer handler C/SP Operating system including: (a) Terminal management supervisor (b) Terminal mangaement control routine (c) Communications control routines (d) Message control program Peripheral control program Peripheral control routines: (a) Card read symbiont (b) Card punch symbiont (c) Print symbiont

Table 1–2. C/SP Software Package

The C/SP operating system is a resident program that maintains subsystem control of working programs, interrupt dispatching, timer services, and I/O operations. The C/SP operating system provides a simplified user interface for receiving or sending data to a remote device.

Message translation, editing, remote device acquisition, sign-on procedures for demand and remote batch operation, job stream or job control, and statement sequence checking are handled by the C/SP operating system. The host processor is thereby relieved of these details normally associated with communications.

The operating system is generated in MACRO form to enable the user to assemble only those MACROs needed to perform a required function. An element of the operating system called the message control program (MCP) controls a general class of terminals. The user may develop an MCP to operate concurrently, but independently, of the system MCP.

The operating system can be further tailored to the user's requirements by input and output own code options as well as user written contingency routines.

Estimates of storage requirements for individual elements of resident programs and routines are listed in Table 1–3. Actual user storage requirements can be computed only after factors such as terminal type, mixture of terminals, anticipated throughput, and amount of code for user's own code processing have been analyzed.

ELEMENT	APPROXIMATE NUMBER OF BYTES
C/SP operating system	15,000
Intercomputer handler	1,000
UNIVAC 1004/9000 control routine (RMS–1)	3,000*
UNIVAC DCT 2000 control routine (ASCII and XS-3)	3,000*
Teletypewriter and DCT 500 control routine	3,000*
UNISCOPE 100 control routine	4,000*
UNIVAC DCT 1000 control routine	3,000*
BSC** control routine	5,000*
Peripheral control routines	3,000
Peripheral control program	6,000
Storage and buffers (per line)	350
BCW's (per position)	16
Constants (per terminal)	10

*Reentrant code required only once per device type

**BSC – binary synchronous control

Table 1–3. C/SP Estimated Storage Requirements

1.4. EQUIPMENT CONFIGURATIONS

The following diagrams (Figures 1–4 and 1–5) contain examples of C/SP configurations. Figure 1–4 illustrates a communications concentrator configuration including eight asynchronous CLTs. Functioning as a line concentrator, multiplexed CLTs are connected to an 1100 series system through the 1100 series adapter channel. Figure 1–5 illustrates a typical onsite configuration with increased storage, eight asynchronous CLTs, eight synchronous CLTs, and a multiplexer channel. With the additional storage and multiplexer channel, this configuration may be used for symbiont processing as well as for line concentration.

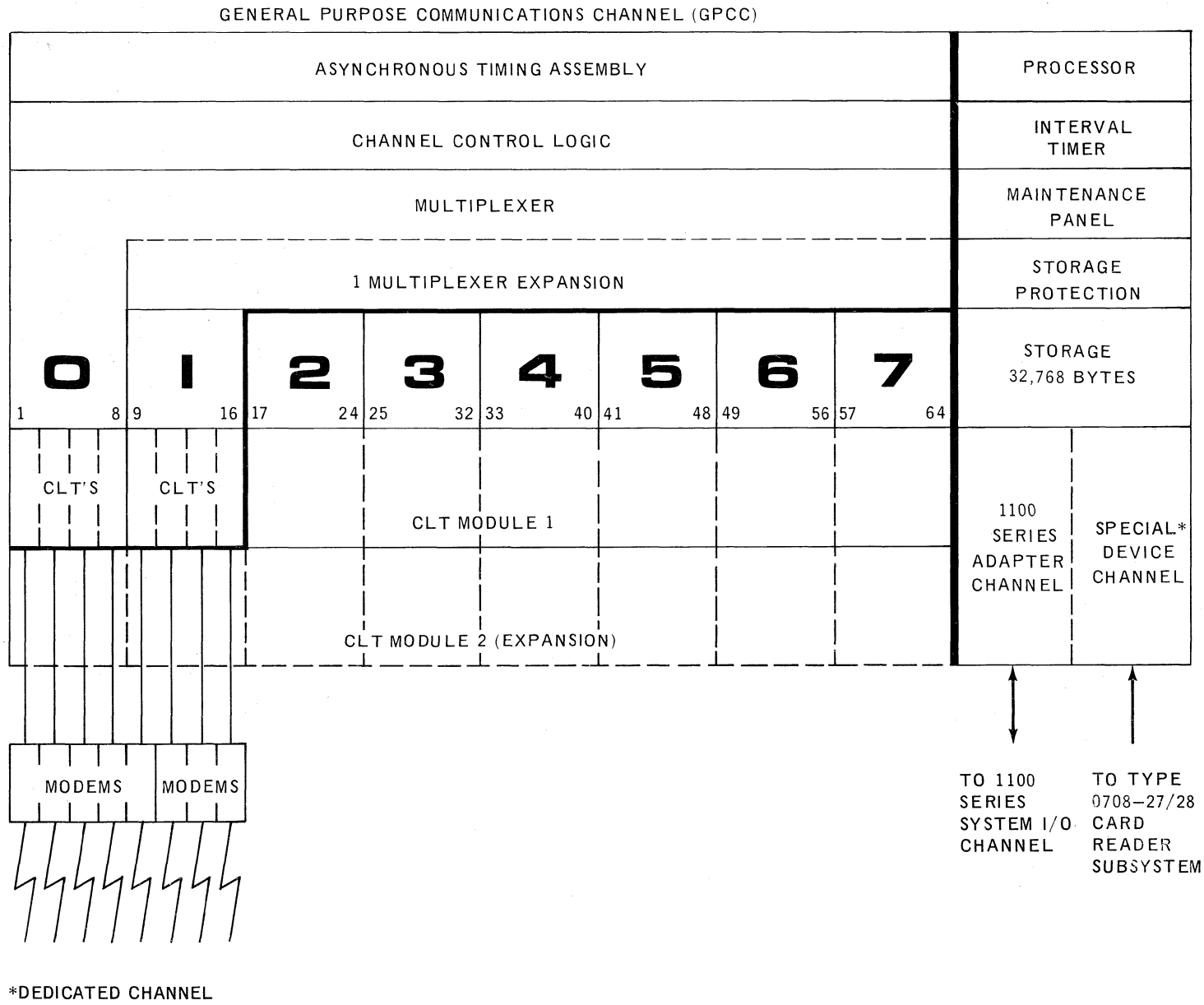


Figure 1–4. C/SP Configuration, Communications Concentrator

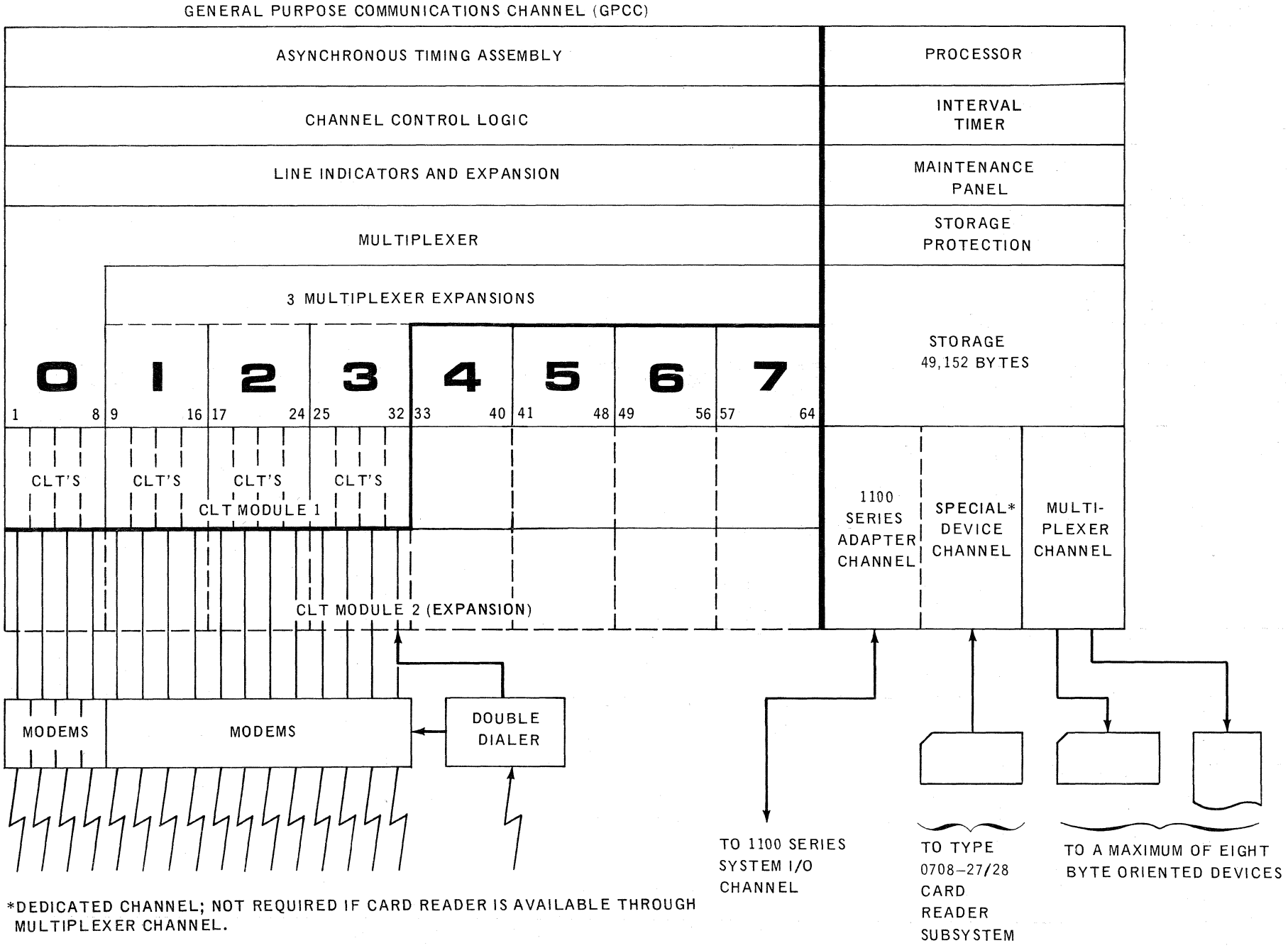


Figure 1–5. C/SP, Typical Configuration Including Symbiont Processing

2. HARDWARE COMPONENTS

2.1. GENERAL

The Communications/Symbiont Processor (C/SP) is a byte oriented, internally-programmed communications processor. The unique feature of the C/SP is the general purpose communications channel (GPCC) which has multiple buffer control words (BCW), sets of message discipline words (MDW), and interrupt tabling to permit software control of line and device type disciplines. Operating as an onsite data concentrator, the C/SP acts as a terminus for many types of lines over which data is transmitted to the host computer. The individual configuration of lines and terminals operates under a variety of line procedures, code sets, control signals, and timing constraints. The C/SP software/hardware provides the capability of controlling these configurations and presenting data that is ready for processing the host computer.

The C/SP is functionally divided into the following three major areas:

(1) Storage (contained in the storage unit)

(2) Processor (contained within the C/SP unit)

(3) I/O channels (contained within the GPCC unit with channel adapters located in C/SP unit)

2.2. STORAGE

A high performance, plated-wire storage is an integral part of the C/SP. Storage is located in one or two units depending upon the size of the storage capacity.

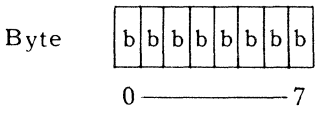
2.2.1. Main Storage Characteristics

Major features of main storage include the following:

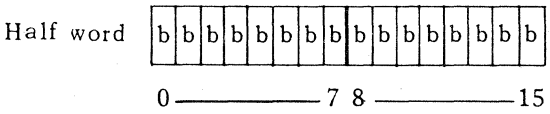
- Capacity – 32,768 bytes minimum; 131,072 bytes maximum
- Cycle time – 630 nanoseconds read/write cycle
- Operating mode – Nondestructive readout
- Storage data path – 18 bits wide (two 8-bit bytes and two parity bits)
- Addressing – Byte level addressing
- Storage protection – Program and I/O transfer
- Parity – Odd parity (1 parity bit per byte)

2.2.2. Storage Data Boundaries

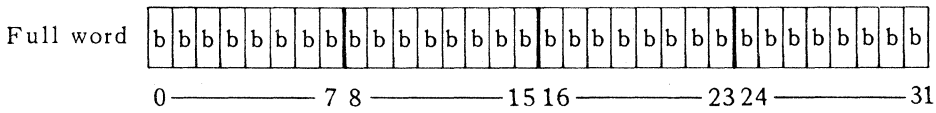
Bytes are addressed consecutively from 0 through 131,071. Bytes may be accessed separately or in groups. The address of a group of bytes is addressed by the left-most byte of the group. The bits within a byte are also numbered from left to right starting with 0.



Half-word data formats consist of two consecutive bytes.



Full-word data formats consist of four consecutive bytes.



Fixed-length fields, such as half words and full words, have integral boundaries. Fixed-length fields must be loaded into main storage so that the address is evenly divisible by the field length (in bytes). Thus, a half word must have an address that is a multiple of 2 and a full word must have an address that is a multiple of 4. The binary address of these fields must be in the form:

Half word XX...XX0

Full word XX...X00

All processor instructions are restricted to half-word boundaries.

2.2.3. Addressing

The addressing hardware accommodates a 17-bit address field which permits one cycle addressing of 131,072 bytes. While the address field permits the addressing of each byte, the least significant bit of the address is not used to access the data from storage.

On a read cycle, the storage presents two bytes to the processor. If the particular reference requires byte addressing, the processor selects the appropriate byte based upon the least significant bit of the address field.

The capability for partial write is provided; that is, one byte may be written without altering the other byte in the storage half word.

2.2.4. Storage Protection

In addition to the fixed storage assignment (see Figure 2–1), there may be several programs resident in C/SP storage at any one time. It becomes necessary to restrict storage accesses by a program to the storage limits assigned to the program.

Associated with C/SP storage is a maximum of 64 3-bit registers called key storage registers. The storage, beginning at address 0, is divided into a maximum of 64 blocks (each block contains 2048 bytes). A key storage register is assigned to each block. The six most significant bits of a storage address are used to define the address of the key storage register associated with the block containing the storage address.

Storage is segmented by grouping together all blocks whose associated key storage registers have the same setting. A maximum of eight storage segments can be defined.

When a program is loaded, the program is assigned a unique program number. This number is then loaded into the key storage register that is associated with each 2048 byte block assigned to the program.

Storage protection against improper storage accessing is provided during instruction execution and I/O transfers.

- Instruction Execution Protection

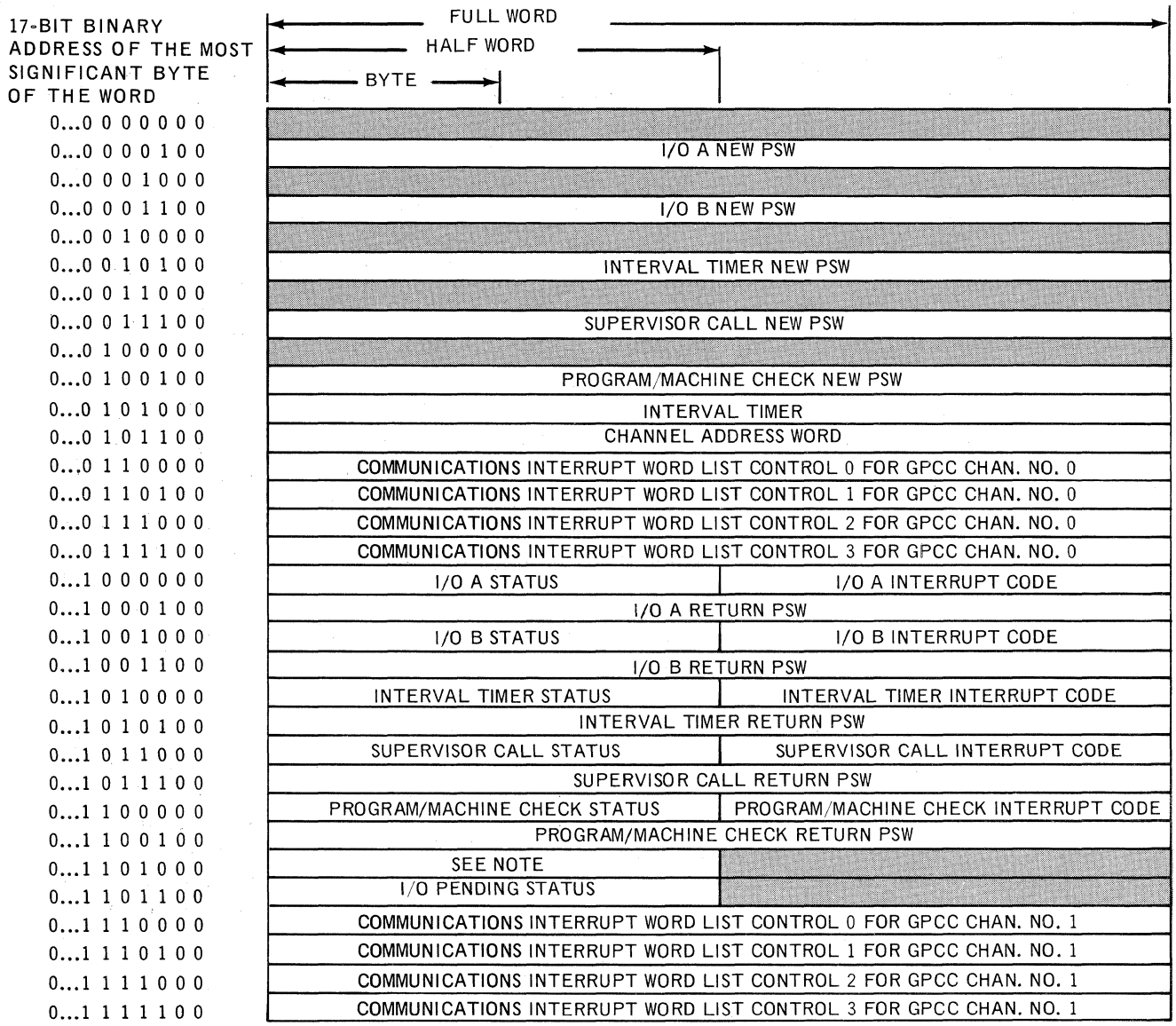
  When a program is scheduled for execution, the program number is loaded into a program status word (PSW) register. The program number uniquely identifies the program to be executed. On each access to storage during processing, the program number in this register is compared with the contents of the key storage register that is associated with the storage address. If a match is made, the storage access is allowed, otherwise an error interrupt occurs.

- I/O Transfer Protection

  The number of the program requesting the I/O transfer is presented to the I/O channel. Upon transfer, the contents of the key storage register that is associated with the address to which the transfer is to be made is checked against the program number associated with the I/O channel. Again, a match of the key storage register and program number permits the transfer to take place; otherwise, an error interrupt occurs.

2.2.5. Parity

The parity bit associated with each byte provides odd parity for that byte. Parity generation and checking are performed in storage. The parity bits are presented to the processor on a read cycle.



NOTE: Storage locations 68_{16} and 69_{16} are reserved for general purpose communications interrupt pointer list.

UNASSIGNED – CAN BE USED AS NORMAL STORAGE

Figure 2–1. Fixed Storage Assignments

2.3. PROCESSOR

The processor portion of the C/SP provides the flexibility that is required to control the I/O data flow and to perform message processing, as necessary, in a communications environment.

2.3.1. Processor Characteristics

Major features of the processor include the following:

- 52 half-word and full-word instructions;
- sixteen 32-bit general purpose registers, external to storage;
- attached processor maintenance panel;
- I/O interrupt and data priority controls;
- variable interval timer;
- half-word basic data path;
- multilevel interrupt;
- 630 nanosecond cycle time;
- basic binary add (RX) instruction time of 2.52 microseconds (four cycles);
- binary add instruction (RR) time of 1.26 microseconds (two cycles).

2.3.2. Control Section

The control section regulates the sequence in which instructions are executed, interprets and controls the execution of each individual instruction, initiates cycling of main storage, performs required storage address modification and indexing, and determines the different processor modes of operation. All of the hardware aspects of interrupt handling, error checking, and protection are also performed by the control section.

Many of these functions are accomplished through use of the program status word (PSW). There are three classes of PSW's:

- Current PSW
- New PSW
- Return PSW

The current PSW is the PSW currently in control of the processor and is located in the PSW register. Two PSW locations in fixed storage are associated with each class of interrupt (Figure 2–1). One of these is the return PSW, the other the new PSW. When an interrupt occurs, the current PSW is stored in the return PSW location of fixed storage that is associated with the interrupt class which causes the interrupt; the contents of the new PSW location for that interrupt class are placed in the current PSW register location.

2.3.3. Arithmetic Section

The arithmetic section of the processor performs all data manipulations including logical and numerical arithmetic, data comparisons, and shifting. The arithmetic section also performs single or double indexing of operand addresses. Arithmetic operations are performed in the twos complement form. A fixed-point arithmetic operand can be either a 32-bit full word or a 16-bit half word. The sign of a fixed-point operand is always the leftmost bit of the operand. When accessed from storage, a half-word fixed-point number is always expanded to a right-justified full word; the sign is extended to the left.

Logical operations on fixed-length operands are performed in registers. Logical operations include comparing, bit setting, and bit testing.

2.3.4. Instruction Set

The C/SP utilizes 52 basic instructions that vary in format and length. The format, in general, is dictated by the operation to be performed and the location of the operands. Operands may be located in storage, in general purpose registers, or in the instruction itself. The length of an instruction is determined by the format and is either a half word or full word. All processor instructions must be on half-word boundaries in storage. Operand addresses in storage are on byte, half-word or full-word boundaries, depending upon the instruction. For example, if an operand for a particular instruction is a full word, the operand address in storage must be on a full-word boundary. Appendix A lists all C/SP instructions together with mnemonic representations and OP codes.

Figure 2–2 illustrates the four basic instruction formats that are used in the C/SP, the formats are designated as follows:

RR (register to register) instructions

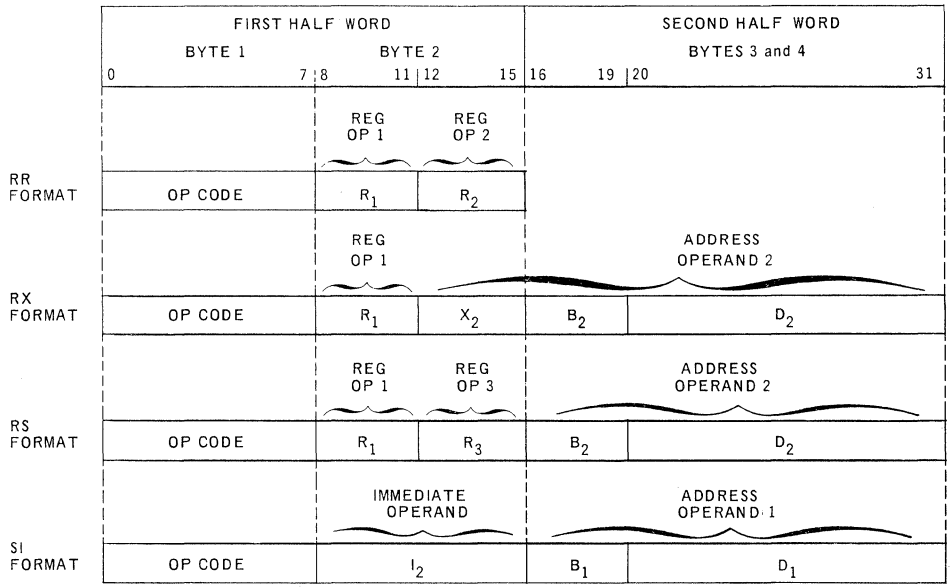
RX (register to indexed storage) instructions

RS (register to storage) instructions

SI (storage and immediate operand) instructions

Each format consists of an operation code (OP code) and two or more fields which specify the addresses of operand (in storage or in the general purpose registers). Each field is identified by a letter followed by a subscript numeral. The numeral denotes the operand (1, 2, or 3) to which the field applies.

The operation codes are expressed in hexadecimal (base 16); each code appears as two hexadecimal digits in the 8-bit OP code field of each instruction.



SYMBOL	MEANING
OP CODE	Instruction operation code
R_1	The number of the register address as operand 1, or a register which is the first register of a multiregister group
R_2	The number of the register addressed as operand 2
R_3	An expression representing a register which is the last register in a multiregister group
X_2	The number of the register to be used as an index for operand 2 of an RX instruction
I_2	The immediate data or device address used as operand 2 of a SI instruction
B_1	The base register for operand 1
B_2	The base register for operand 2
D_1	The displacement for operand 1
D_2	The displacement for operand 2
OP1	Operand 1
OP2	Operand 2
OP3	Operand 3 (extended mnemonic repertoire)
M1	Mask (extended mnemonic repertoire)

Figure 2–2. Basic Instruction Formats

2.3.4.1. Register to Register Instructions (RR)

The RR instructions are two bytes in length and process data contained only in registers. The maximum operand that can be handled is a full word of 32 bits. The operand may or may not be a signed binary number.

In this format, there are 13 instructions:

RR INSTRUCTIONS	MNEMONIC OP CODE
Add	AR
AND	NR
Branch and link	BALR
Branch on condition	BCR
Branch on count	BCTR
Compare	CR
Compare logical	CLR
Exclusive OR	XR
Load	LR
OR	OR
Set storage key	SSK (privileged instruction)
Subtract	SR
Supervisor call	SVC

2.3.4.2. Register to Indexed Storage Instructions (RX)

The RX instructions are four bytes in length and process data between registers and indexed storage. The maximum operand that can be handled is a full word of 32 bits. The operand may or may not be a signed binary number.

In this format, there are 22 instructions:

RX INSTRUCTIONS	MNEMONIC OP CODE
Add	A
Add halfword	AH
AND	N
Branch and link	BAL
Branch on condition	BC
Branch on count	BCT
Compare	C
Compare halfword	CH
Compare logical	CL
Divide halfword	DH
Exclusive OR	X
Insert character	IC
Load	L
Load address	LA
Load halfword	LH
Multiply halfword	MH
OR	O
Store	ST
Store character	STC
Store halfword	STH
Subtract	S
Subtract halfword	SH

2.3.4.3. Register to Storage Instructions (RS)

The RS instructions are four bytes in length and are used to perform branch and shift operations.

In this format, there are seven instructions:

RS INSTRUCTIONS	MNEMONIC OP CODE
Branch on index high	BXH
Branch on index low or equal	BXLE
Divide polynominal	DP
Shift left single	SLA
Shift left single logical	SLL
Shift right single	SRA
Shift right single logical	SRL

2.3.4.4. Storage and Immediate Operand Instructions (SI)

The SI instructions are four bytes in length and provide data that is used to control the processor and peripherals, and for logical operations.

In this format, there are 10 instructions:

SI INSTRUCTIONS	MNEMONIC OP CODE
AND	NI
Compare logical	CLI
Exclusive OR	XI
Halt and proceed	HPR (privileged instruction)
Load program status word	LPSW (privileged instruction)
Move immediate	MVI
OR	OI
Set system mask	SSM (privileged instruction)
Start I/O	SIO (privileged instruction)
Test under mask	TM

2.3.5. Extended Mnemonic Codes

Extended mnemonic codes are provided in the assembler language as a short-hand notation for writing branch instructions. A list of the extended mnemonic codes, along with their meaning, is shown in Table 2–1.

RR TYPE INSTRUCTIONS		RX TYPE INSTRUCTIONS		FUNCTION
MNEMONIC CODE	HEXADECIMAL OPERATION CODE M_1	MNEMONIC CODE	HEXADECIMAL OPERATION CODE M_1	
BR	07 F	B	47 F	BRANCH
NOPR	07 0	NOP	47 0	NO OPERATION
USED AFTER COMPARISON INSTRUCTIONS				
BHR	07 2	BH	47 2	BRANCH IF HIGH
BLR	07 4	BL	47 4	BRANCH IF LOW
BER	07 8	BE	47 8	BRANCH IF EQUAL
BNHR	07 D	BNH	47 D	BRANCH IF NOT HIGH
BNLR	07 B	BNL	47 B	BRANCH IF NOT LOW
BNER	07 7	BNE	47 7	BRANCH IF NOT EQUAL
USED AFTER TEST UNDER MASK INSTRUCTIONS				
BOR	07 1	BO	47 1	BRANCH IF ALL ONES
BZR	07 8	BZ	47 8	BRANCH IF ALL ZEROS
BMR	07 4	BM	47 4	BRANCH IF MIXED
BNOR	07 E	BNO	47 E	BRANCH IF NOT ALL ONES
BNZR	07 7	BNZ	47 7	BRANCH IF NOT ALL ZEROS
BNMR	07 B	BNM	47 B	BRANCH IF NOT MIXED
USED AFTER ARITHMETIC INSTRUCTIONS				
BOR	07 1	BO	47 1	BRANCH IF OVERFLOW
BZR	07 8	BZ	47 8	BRANCH IF ZERO
BMR	07 4	BM	47 4	BRANCH IF MINUS
BPR	07 2	BP	47 2	BRANCH IF POSITIVE
BNOR	07 E	BNO	47 E	BRANCH IF NO OVERFLOW
BNZR	07 7	BNZ	47 7	BRANCH IF NOT ZERO
BNMR	07 B	BNM	47 B	BRANCH IF NOT MINUS
BNPR	07 D	BNP	47 D	BRANCH IF NOT POSITIVE

Table 2–1. Extended Mnemonic Codes

2.3.6. Interval Timer

The interval timer, which utilizes a fixed-word location in storage, is a feature available in the processor which provides interval timing and time-of-day information. Interval timer requests for service are made every six milliseconds. Interval timer requests may be serviced only at the end of a processor instruction execution prior to the processor staticizing the next instruction.

2.3.7. Interrupts

The interrupt system provides an automatic means of alerting the C/SP processor to exceptional or unexpected conditions, such as the end of I/O operations, program errors, machine errors, and similar occurrences and directs the processor to the appropriate program routine following their detection. The system permits the interruption of any task to process a task of higher priority. Among the features of the interrupt function are:

- automatic tabling of communications channel interrupts;
- a dynamically alterable priority structure;
- automatic dispatch by interrupt class;
- automatic program switching by interrupt class.

The system is designed so that processing occurs on a priority basis by class of interrupt. There are five classes of interrupts, which are listed below:

- Program/machine check (PMC) interrupt – is caused by any of the following conditions:
  - Invalid operation code
  - An attempt to execute a privileged instruction while operating in problem mode.
  - Protection exception. (This occurs when the operating program attempts to access an area of storage which is not assigned to the program.)
  - An attempt to address a location beyond the storage size installed.
  - Parity error upon a transfer of data to or from storage.
- Supervisor call interrupt – results from the execution of a supervisor call instruction. Status information provides the operating system with a link to parameter information in the calling program.
- Interval timer interrupt – occurs when a program settable interval counter reaches 0. The interval setting may vary from a count of 1 to 256; the count is decremented by 1 every 6 milliseconds.
- I/O B interrupt – occurs when one of the channels or channel adapters operating on channels 1 through 6 accepts or terminates an I/O function, or detects a device malfunction, or a data parity error.
- I/O A interrupt – is generated by a variety of conditions monitored by the GPCC in control of communications activity. These conditions include data parity error, loss of transmission carrier, and recognition of program specified control characters. Special interrupt tabling accommodations are provided for the GPCC.

  Since all the processing that is required for communications interrupts is not equally time critical, the C/SP provides four interrupt lists in fixed storage.

2.3.8. Maintenance Panel

The processor maintenance panel comprises controls and indicators which permit personnel to monitor and operate the C/SP in various modes; executing processor operations, performing initial loads, and altering and displaying the contents of storage for maintenance purposes. The maintenance panel contains controls which are restricted to use of Univac customer engineers in addition to operator's controls. During maintenance, the C/SP communications line terminals can be connected in a back-to-back mode of operation and checks can be made of the equipment.

2.4. I/O CHANNELS

All information transmission in and out of the C/SP is handled by channels. A channel controls the operation of input/output devices and the transfer of data between devices and storage.

The C/SP contains a maximum of seven channels (0 through 6). Channel 0 has the lowest priority for data transfers; the priority for channels 1 through 6 increases in descending channel number order with channel 1 having the highest priority. Priority assignments can be changed, subsequent to initial installation, to allow for future expansion of facilities. For example, various adapter channels are assigned as follows:

Channel	Assignment
0	General purpose communications channel (GPCC)
1	Reserved for GPCC expansion
2	Not used
3	Multiplexer channel
4	1100 series adapter channel
5	Not used
6	Special device channel (SDC) (fixed channel assignment)

2.4.1. Channel Types

The C/SP can be equipped with the following four channels.

- Special device channel (SDC) – The primary function of the SDC is to provide the means for local program loading and maintenance of the C/SP using a serial 80-column, 80-card per minute, card reader device.

- 1100 series adapter channel – The 1100 series adapter channel (intercomputer adapter channel) provides an interface for direction communication of the C/SP to an I/O channel of a UNIVAC 1100 series computer. The maximum transfer rate is 300,000 words (36 bits each) per second.

- Multiplexer channel – This channel provides the capability of attaching eight currently available UNIVAC 9000 series peripheral devices which operate on a corresponding channel to the C/SP. In addition, the high speed card reader and the ASCII printer can be connected through this channel.

- General purpose communications channel (GPCC) – The GPCC and associated components are described in 2.4.2.

2.4.2. General Purpose Communications Channel (GPCC)

The GPCC performs such functions as multiplexing the various communications line terminals (CLTs) so that one CLT may be serviced at a time, recognizing special characters and sequences of characters, checking character parity, coordinating all data transfers to and from storage, and executing other necessary operations.

The CLTs perform the function of assembly and disassembly of data characters for proper reception from and transmission to a communications line; detection of certain conditions of the communications line such as loss of carrier, a ringing indication, and others; and establishment of character synchronization.

The CLTs handle a wide range of communications with rates up to 50 kilobytes per second. However, the CLTs must be selected so that the total combined rate of service requests (one per byte) is no more than 50,000 per second. When the GPCC is operated at this maximum rate, somewhat less than 40 percent of all available storage cycles are utilized for this purpose.

The GPCC is the link between storage and the CLTs and provides the data path and control for CLTs as they communicate with storage. The single data path can be time-shared by as many as 64 positions, which need not have identical CLTs. A full duplex CLT uses two positions; a half duplex or simplex CLT utilizes only one position.

The GPCC is equipped to analyze each data character or sequence of characters which is transmitted through the GPCC and to act upon these characters in a manner that is a program-changeable function of the line to which the GPCC is connected. The GPCC also interfaces with the C/SP processor to service Start I/O (SIO) instructions and interrupts. Associated with the GPCC is a display panel which contains two active line indicators (output and input) for each CLT. The indicator is on when the corresponding CLT data line is in a spacing condition.

The multiplexer portion of the GPCC accepts up to 64 simultaneously presented service requests from the CLTs plus an external function (XF) request from another portion of the GPCC. The multiplexer selects the highest priority request and connects the selected CLT to the GPCC. When all necessary information has been interchanged, the multiplexer is cleared and can immediately accept another request. The multiplexer can accommodate a maximum of 32 full duplex CLTs or 64 half duplex CLTs. An area in storage called a buffer control word (BCW) is associated with each position on the multiplexer and is used to store status, control, and data address information for the particular position.

When information is to be transferred during an I/O operation, the CLT requests service from the GPCC and (when priority permits) sends the CLT address to the GPCC. The GPCC uses the address to select the associated BCW (which is loaded into the channel) and now controls the channel operation until the information is transferred. Then the BCW, in general changed by the channel operation (for example, address incrementation), is returned to main storage and the GPCC facilities are released. The BCWs then retain the current state of the various positions.

The C/SP is intended to operate in an environment which can involve many different line discipline procedures. Many such procedures have been long established and must be handled unchanged by the C/SP. To avoid a multiplicity of tailored CLTs operating through the GPCC, the BCW is permitted to access a message discipline word (MDW). A chain of MDWs can be considered as a description of a given procedure, and the currently active MDW represents the position reached by a given line within that procedure. The chain of MDWs should not be modified once it has been loaded. Hence, all lines with the same line discipline procedure may share a common chain of MDWs. An MDW contains, for example, parameters which control character parity checking, special character recognition (single or multiple), special character insertion, and other operations.

At various points throughout a procedure, it is necessary to present certain information to the controlling program. Since this information is dynamically changing, it must be stored at the desired point immediately. A normal interrupt is not sufficient since the processor is not always in a condition to accept an interrupt request. Four interrupt lists are provided in storage where communications interrupt words (CIWs) are stored. These lists are controlled by CIW list controls which are in fixed storage locations. A list is selected by a BCW or MDW and is usually assigned on a priority basis.

The GPCC consists of the following:

- Control logic
- Communications line terminals (CLT)
- Dialer CLT
- Timing assembly
- CPU–I/O channel interface
- Multiplexer

2.4.2.1. GPCC Control Logic

The control logic section of the GPCC distinguishes the C/SP from other communications control processors. The GPCC control logic interfaces and controls the operation of the CLTs by communicating with the CLTs through the multiplexer. Hardware registers, loaded under software control and contained within the logic, assist in these control functions.

Control of the CLTs is exercised through the following four control words:

Channel address word (CAW)

Buffer control word (BCW)

Message discipline word (MDW)

Communications interrupt word (CIW)

(1) Channel Address Word (CAW)

The CAW is accessed from storage by the GPCC when a processor start input/output (SIO) instruction with the GPCC's channel number is detected. The GPCC stores the 16 most significant bits of the CAW in its hardware registers. This stored command is called an external function (XF).

(2) Buffer Control Word (BCW)

A BCW exercises primary control over the flow of data between an individual CLT and storage. Each multiplexer position within the GPCC has a corresponding BCW assigned to it.

Each BCW is four machine words in length (designated words 0 through 3) and must begin on a word storage boundary which is a multiple of four words. The BCWs for the GPCC occupy, in the maximum case, a table 256 words long (4 × 64 CLT positions). Individual BCWs are selected from this table on the basis of the 64 multiplexer position (effectively, CLT position) addresses.

The four-word BCW concept, together with the GPCC hardware buffer toggling feature, greatly enhances the C/SP's performance. The GPCC maintains the status of the data buffers in use and when an input buffer is filled or an output buffer emptied, the GPCC sends an interrupt to the processor. The buffer then toggles (or switches) control to another segment of the BCW thus directing new service requests to a new area containing valid data. In this manner, the operating programs are given line activity time required to deplete a data buffer instead of the single character time that is typical of a system without multiple BCWs and buffer toggling capabilities.

(3) Message Discipline Word (MDW)

An MDW may be used to supplement the control provided by the BCW. The MDW may be up to four machine words in length and exercises control over the data transfer, relative to the control information contained in the data being transferred. Provisions are provided to compare each input character with a maximum of six unique codes. Among the functions under control of the MDW are:

- Recognize up to six control characters and:
  - store or not store character;
  - interrupt or not interrupt;
  - terminate or not terminate line.

- Check or generate parity (odd or even)
- Duplicate recognized characters on output
- Control use of a special data buffer
- Effect rapid line reversal
- Force a unique code to output line
- Link to a new MDW control word
- Select a priority interrupt table

The BCW specifies whether an MDW will be accessed, and if so, which MDW will be accessed. All time variant constants and the field that designates the currently active MDW are in the current BCW for each line. An MDW list may be used by all lines that operate under the same message discipline.

(4) Communications Interrupt Word (CIW)

A CIW is stored whenever a condition is recognized which requires the attention of the controlling program. The control supplied by the BCW/MDW has been designed to allow the program to respond to interrupt conditions; therefore, the advantages of a semisoft interrupt scheme is realized. Interrupts are tabled in the form of CIWs and an interrupt request is generated. Additional interrupts will be tabled, even if the previous request has not been honored; thus, efficiency in examining CIWs by the program is provided.

2.4.2.2. Communications Line Terminal (CLT)

A communications line terminal (CLT) is placed in a GPCC terminal position to interface a modem on the communications line. A CLT operating in full-duplex mode requires two positions, while a CLT operating in a half-duplex mode requires one position. A variety of CLT models are provided to effect compatibility with varied communications facilities and communications lines with rates to 50 kilobits per second operating in either synchronous or asynchronous modes (see Table 2–2).

CLT TYPE	SPECIFICATION
SYNCHRONOUS	E.I.A. RS–232B MIL STD 188B CCITT
ASYNCHRONOUS	E.I.A. RS–232B MIL. STD. 188B CCITT AT&T 20–75 MIL (Neutral) WV 20–40 MIL (Polar)
DIALER	E.I.A. RS–232B

Table 2–2. CLT Compatibility Chart

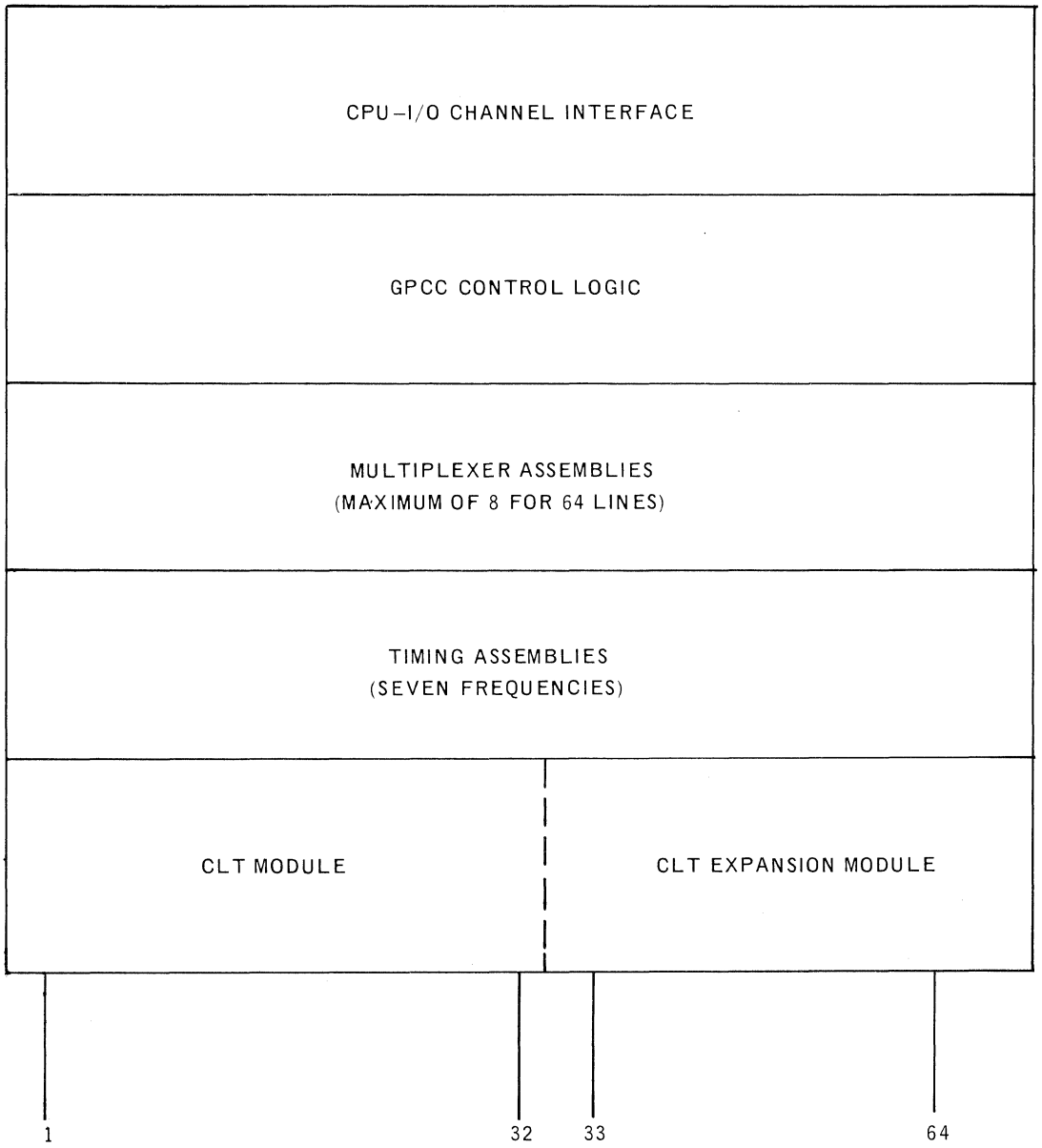


Figure 2–3. General Purpose Communications Channel (GPCC)

A CLT performs the function of inputting or outputting data and control characters. This function includes:

- accepting control commands from the GPCC;
- requesting service from the C/SP to input or output a character;
- storing one character in each of the input and output sections of the CLT;
- transferring the character to or from the data set (modem) in a bit-serial form;
- manipulating and interpreting the control lead signals to the data set;
- monitoring the lines for error conditions, such as loss of carrier;
- presenting CLT status to the GPCC.

All CLTs have the ability to operate in a test mode; this mode provides a means of checking the operation of the CLT without involving the data set or the communications lines.

(1) Synchronous CLT

Synchronous CLTs perform the following:

- Receive bit-serial data from the data set and test each bit for a time synchronizing code bit combination.
- Upon recognition of a synchronizing code, the CLT enters the data mode by initiating service requests to the GPCC for each subsequent group of bits defined by the number of levels in the code being used (programmer option).

(2) Asynchronous CLT

Asynchronous CLTs perform the following:

- Recognize and extract the start and stop bits associated with each character received from the data set.
- Generate the start and stop bits associated with each character transferred to the data set.

2.4.2.3. Dialers

The dialer is a device which permits the GPCC to control an automatic calling unit (ACU) for the purpose of originating calls automatically on a switched telephone network. The dialer performs the following functions:

- accepts control commands from the GPCC;
- requests a dialing digit from the GPCC;
- transmits a dialing digit to the ACU in 4-bit parallel form;
- manipulates and interprets the control lead signals of the ACU;
- presents dialer status to the GPCC.

2.4.2.4. Asynchronous Timing Assembly

Timing sources, required for operation of the asynchronous CLTs, are provided by the asynchronous timing assembly (ATA). One timing source is basic to each GPCC and is specified by frequency selection. A maximum of three timing sources are available with each ATA; each GPCC can accommodate two ATAs. Frequency selections which are not specified on initial orders but subsequently required, can be easily incorporated into the ATA. The following frequencies are available:

45.45 Hz	133.00 Hz	1050.00 Hz
50.00 Hz	150.00 Hz	1200.00 Hz
74.20 Hz	200.00 Hz	1800.00 Hz
75.00 Hz	300.00 Hz	
100.00 Hz	600.00 Hz	
110.00 Hz		

2.4.2.5. CPU – I/O Channel Interface

The CPU–I/O channel interface section of the processor provides the interface between the C/SP processor and the GPCC control logic. This interface recognizes I/O instructions intended for the GPCC, provides direct access to main storage, and provides the logic paths for interrupts from the GPCC.

2.4.2.6. Active Line Indicators

The active line indicators are located on the GPCC active line indicator display panel and includes two indicators for each CLT position to monitor the receive and transmit lines. When the line is active, the indicator for that line is illuminated by a spacing condition on that line. A CLT operating in a full-duplex mode has both indicators lit simultaneously; a CLT in half-duplex mode or simplex mode has one indicator illuminated.

When an indicator is associated with a single dialer in a CLT position, the indicator monitors the call request using the output indicator. A double dialer in a CLT position utilizes the input indicator to monitor the call request.

Indicators are available in groups of 32, which service 16 lines and are numbered at factory assembly. The GPCC active line indicator display panel can accommodate a maximum of 128 indicators covering two GPCCs and are numbered consecutively.

2.5. REMOTE PERIPHERAL SUBSYSTEMS

The remote peripheral subsystems perform diverse functions that are complementary to the central computer and are connected to the central computer through common carrier facilities and the C/SP. The remote devices included in this section are standard UNIVAC products and adhere to the accepted interface standards for serial data transmission, EIA specification RS–232. These devices include:

- UNIVAC UNISCOPE 100
- UNIVAC Data Communications Terminal DCT 500
- UNIVAC Data Communications Terminal DCT 1000
- UNIVAC Data Communications Terminal DCT 2000
- UNIVAC 9200/9200 II Systems
- UNIVAC 9300/9300 II Systems.

2.5.1. UNISCOPE 100 Display Terminal

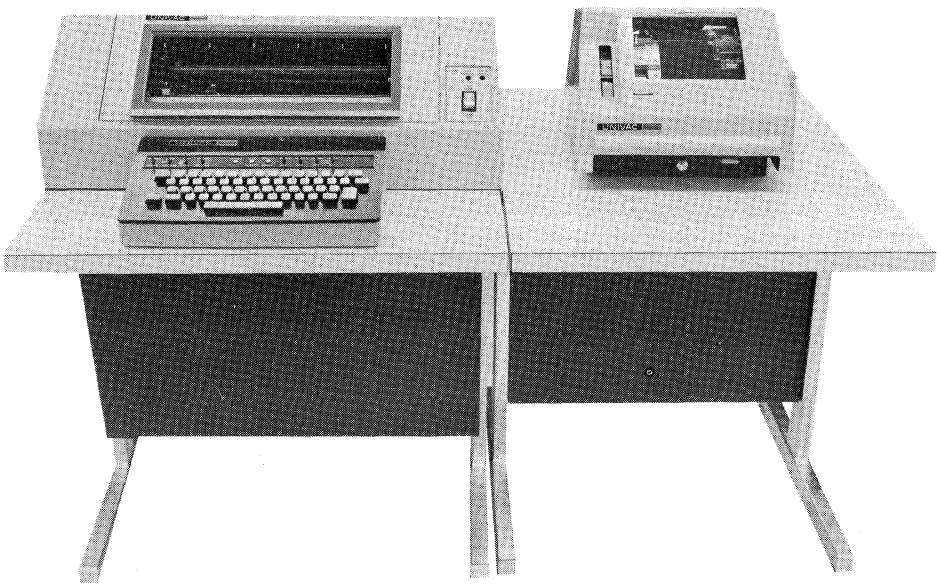
The UNISCOPE 100 Display Terminal is a low-cost alphanumeric display that is designed for a broad range of applications which require direct operator interaction with a centralized computer system. Due to its modular construction, the UNISCOPE 100 terminal operates either as a data entry or as a display device. It can be conveniently located at the central computer site or at a remote station connected to the system by way of telephone lines.

The UNISCOPE 100 terminal is a self-contained unit consisting of a cathode-ray tube display screen, refresh memory, character generator, control logic, operator keyboard, and communications interfaces. Special interfaces for direct computer connection and hard copy output are also available. A variety of presentation formats are offered which provide a total display capacity of 480, 512, 960, or 1024 ASCII characters. The complete ASCII set of 96 characters can be displayed (includes upper and lower case alphabetics). Hardware editing capabilities enable the operator to completely edit any message prior to transmitting the message to the computer.

Sixteen UNISCOPE 100 terminals may be connected to a single communications line modem or to a computer input/output channel by means of multiplexers. The general purpose multiplexer is available with all the communications line interfaces available on the UNISCOPE 100 terminal, thus permitting a mixture of single units and multiple units on one communications system. The multiplexer also permits broadcasting output messages to multiple devices.

CHARACTERISTICS	
Display	
Viewing area	10 inches wide x 5 inches high
Capacity	480, 512, 960, 1024 characters
Format	32 characters per line x 16 lines 64 characters per line x 16 lines 80 characters per line x 6 lines 80 characters per line x 12 lines
Refresh rate	60 cycles per second
Scan method	Digital
Character set	64 or 96 symbols (ASCII)
Character generation	Closed stroke, maximum 8 per character
Keyboard	Numeric, alphanumeric Cursor control keys Editing keys
Memory	
Type	Magnetic core
Cycle time	8 microseconds
Code	7-bit ASCII with parity bit
Interface	Synchronous – RS-232C, MIL STD 188B, CCITT Asynchronous – RS-232C, MIL STD 188B, CCITT CTMC and DCS direct Multiplexer interface Auxiliary interface (printer connection)

2.5.2. UNIVAC Data Communications Terminal DCT 500



The UNIVAC Data Communications Terminal DCT 500 is a versatile, low-cost terminal designed to accommodate modern computer systems. With an asynchronous keyboard/printer that operates at two or three times the speed of conventional teletypewriter terminals, the DCT 500 produces superior hard copy output with full 132 column width. The DCT 500 is modular in design and is available in three configurations. The basic DCT 500 contains an unbuffered printer that operates in a receive only (RO) mode and is similar in operation to a standard teletypewriter. The basic terminal can be expanded to include a keyboard and a paper tape reader/punch which permits keyboard send/receive (KSR) and automatic send/receive (ASR) operation. KSR operation permits transmission of data as well as reception; ASR operation permits print speeds up to 30 characters per second to be achieved using the paper tape reader/punch. While operating in either a full-duplex or half-duplex mode, communication speeds of 110, 150, and 300 baud are obtained in ASR operation.

CHARACTERISTICS

Communications	
Line type	Switched Nonswitched (private) TWX* exchange
Mode	Full duplex or half duplex, 2 or 4 wire, asynchronous
Transmission speed	10 characters at 110 baud 15 characters at 150 baud 30 characters at 300 baud
Code	ASCII, 7 bits plus parity bit
Interface	EIA RS-232 (asynchronous) CCITT
Printer	
Speed	30 characters per second
Print positions	132 (adjustable)
Character set	63
Character size (typical)	H: 0.090 to 0.110 inch W: 0.060 to 0.085 inch
Print spacing	Horizontal: 10 characters per inch
Ink colors	Black (standard), red, green, violet (optional)
Paper feed rate	30 lines per second
Form size	W: 3 5/8 to 14 7/8 inches (overall) L: 6-part continuously sprocketed, or 3-part carbonless
Form quantity	1 original and 5 carbons (maximum pack thickness .0155 inch recommended)
Paper Tape Reader/Punch	
Tape size	1 inch
Codes	5 or 6 level, 8-level ASCII
Speed	30 characters per inch online 50 characters per inch offline

*Trademark of AT&T Co.

2.5.3. UNIVAC Data Communications Terminal DCT 1000

The UNIVAC Data Communications Terminal DCT 1000 is a reliable, high performance, low-cost terminal equipment capable of handling large quantities of information. The basic DCT 1000 contains an incremental printer that produces hard copy and has a maximum speed of 30 characters per second. A full 132-column line may be printed using a 63 character set. The printer operates at two to three times the speed of conventional teletypewriters and provides up to five carbon copies of printed material. The basic terminal can be expanded to include a keyboard and control panel, card reader, card punch, paper tape reader/punch, and an auxiliary printer.

The keyboard and control panel have 48 alphanumeric character keys and are capable of using a 128 character set. In addition to the character keys, the keyboard contains keys and switches required for full operational control.

The card reader is a desk type unit employing standard 80-column cards. Cards are fed through the card reader in a straight line from an input hopper at a rate of 40 cards per minute through the read section, then into the output stacker. Translator selections permit reading of punched cards using the most common card codes. Cards that have been punched in binary code may be read by adding the binary read feature.

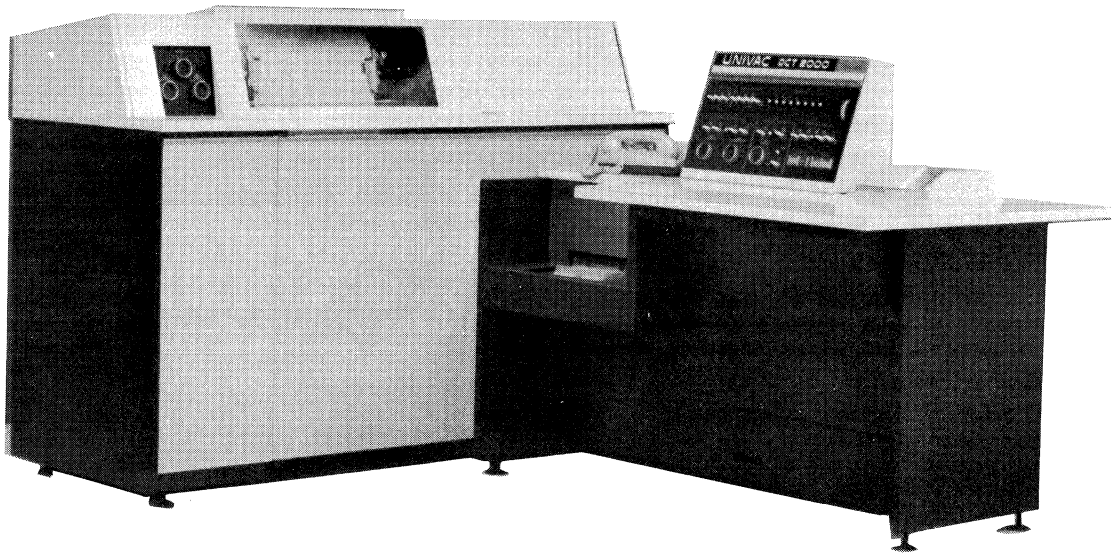
A standard UNIVAC 1701 Verifying Punch (VP) or the UNIVAC 1710 Verifying Interpreting Punch (VIP) is used as the card punch for the DCT 1000. The card punch is capable of operating independently as a separate entity but is interfaced to the DCT 1000 by means of a punch adapter. Reliable performance and instantaneous speed are assured by the use of integrated circuits, main storage, and other modern devices. When operating the DCT 1000 in an offline mode or when the card punch is not selected for normal operation, the card punch may be used as a normal VP or VIP keypunch.

The paper tape reader punch is a desk type unit that is capable of reading and punching standard 1-inch paper tape. The unit can read or punch 5-level or 6-level codes and an 8-level ASCII code at a speed of 50 characters per second. Parity checking is provided on both reader and punch as an optional feature. An even parity is checked on the paper tape reader. In the event of a repeated error, the reader stops operating. An echo parity check is provided on the paper tape punch. If a parity error occurs during punching, the erroneous character is overpunched with a delete character.

CHARACTERISTICS

Communications	
Line type	Private or switched
Private line	Two or four wire
Transmission mode	Synchronous or asynchronous
Transmission speed	Synchronous up to 4800 bauds Asynchronous 300, 1200, or 1800 bauds
Interface	EIA RS-232 (synchronous or asynchronous) MIL STD 188B (synchronous)
I/O channels	Direct to 1100, 400, or 900 series I/O channel by means of a terminal multiplexer
Printer	
Speed	Up to 30 characters per second
Print positions	132
Character set	63
Print spacing	6 lines per inch; 10 characters per inch
Character size (typical)	H: 0.090 to 0.110 inch W: 0.060 to 0.085 inch
Form size	Width: 3 5/8 to 14 7/8 inches Length: up to 11 inches
Form quantity	One original and five copies (maximum pack thickness .0155 inch recommended)
Card Reader	
Card type	Standard 80-column cards
Speed	Up to 40 cards per minute
Hopper capacity	500 cards
Stacker capacity	500 cards
Translator selections	EBCDIC/ASCII H (scientific) code A (business) code
Optional feature	Binary read mode
Card Punch	
Card type	Standard 80-column cards
Speed	Up to 35 cards per minute
Hopper capacity	600 cards
Stacker capacity	600 cards
Translator selections	EBCDIC/ASCII H (scientific) code A (business) code
Optional feature	Binary punch mode
Paper Tape Reader/Punch	
Tape size	1 inch wide
Codes	5 and 6 levels 8-level ASCII
Speed	50 characters per second
Optional feature	Parity check

2.5.4. UNIVAC Data Communications Terminal DCT 2000



The UNIVAC Data Communications Terminal DCT 2000 provides half-duplex, medium-speed data communications with a central computer or with another DCT 2000 using voice-grade facilities. The input/output methods include 80-column card reading and punching, and 80-column line printing (128-column printing optional). The DCT 2000 is available in two code versions: ASCII, with 8-bit characters including parity, and XS-3 (compatible with UNIVAC 1004 DLT 1 or DLT 3), with 7-bit characters including parity. The DCT 2000 uses either a private line connection at a maximum rate of 2400 bits per second, or a dial facility at a maximum rate of 2000 bits per second. 250 blocks of data per minute can be handled by the DCT 2000.

DCT usage is simplified by the fact that no programming is required at the DCT 2000 location. In addition, the operator need not be concerned with the character codes and message delimiters (framing characters) employed, since the DCT 2000 automatically codes characters and adds message delimiters for output messages, and decodes characters and removes message delimiters for input messages.

The bar printer offers low-cost quality performance while producing highly legible hard copy at a maximum rate of 250 lines per minute. Through engineering innovations, particularly in the printing method, the significant advantage of cost reduction is offered to DCT 2000 users. Furthermore, a variety of cost avoidance features such as simplicity of operation and ease of maintenance are inherent in the basic design. In addition, several optional printer features are available.

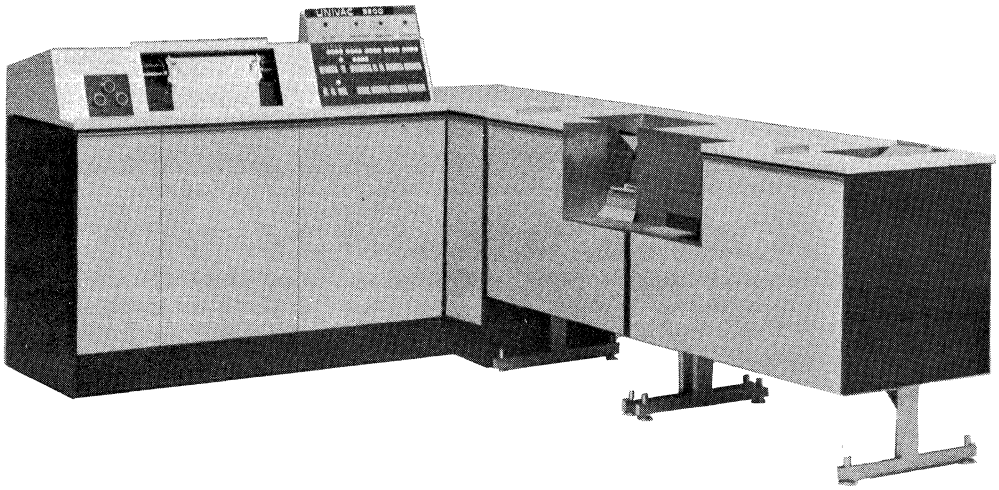
The reader/punch unit feeds, reads or punches, and stacks 80-column cards. The unit can either read or punch a particular card, but cannot perform both operations on the same card during the same cycle. Card punch operations are performed at a rate of 75 cards per minute; the card read rate is 200 cards per minute.

Cards to be read are passed from the input hopper ready station directly to the read station where the punches are sensed photoelectrically. The cards then go to the output stacker without interruption.

CHARACTERISTICS

Communications	
Line type	Private or switched
Transmission mode	Synchronous or asynchronous
Transmission speed	2400 bits per second, private line 2000 bits per second, switched line
Interface	EIA RS-232B MIL STD 188B
I/O channel	Facilities for four peripheral devices
Printer	
Speed	250 lines per minute maximum
Print positions	80 columns
Character set	63
Method	Removable bar
Paper speed	25 lines per second (form skip speed)
Paper spacing	6 lines per inch
Form size	22 inches by 14 inches (maximum one original, five carbons)
Card Reader/Punch	
Card type	Standard 80-column cards
Read speed	200 cards per minute maximum
Punch speed	75 cards per minute maximum (80 column)
Read method	Photoelectric read station
Punch method	Two columns at a time
Input hopper capacity	1200 cards
Output stacker capacity	700 cards
Translator selections	EBCDIC/ASCII H (scientific) code A (business) code

2.5.5. UNIVAC 9200/9200 II or 9300/9300 II Systems



The UNIVAC 9200/9200 II systems or the UNIVAC 9300/9300 II systems may be used as remote subsystems to the central computer. The UNIVAC 9200/9300 systems are linked to transmission facilities through the GPCC CLT, with the transmission facilities being connected to the central computer through the C/SP.

The remote UNIVAC 9200/9300 systems provide inexpensive and efficient means for transmitting data or job stream information to the central computer. Editing and data manipulation features of the UNIVAC 9200/9300 systems provide powerful tools in the preparation of data, thereby saving valuable transmission time.

The UNIVAC 9200 system is a compact, low-priced, internally programmed computer system which is card oriented. The computer is equipped with all functions for the execution of instructions, including arithmetic and input/output control. An integral card reader, card punch, and line printer are standard peripheral units. A multiplexer channel is available for use as the communications channel.

Storage is organized into bytes consisting of eight data bits plus one parity bit. Minimum storage size is 8192 bytes and maximum storage size is 16,384 bytes. Storage cycle time is 1.2 microseconds. The UNIVAC 9200 II system includes all the basic advantages of the UNIVAC 9200 system in addition to a selector channel suitable for disc input and greater storage potential.

The UNIVAC 9300 system is an internally programmed computing system which offers both a powerful 80-column card processing capability and a high-speed magnetic tape system. The computer is equipped with all functions for execution of instructions including arithmetic and input/output control. The integral card reader, card punch, and line printer offer higher speeds than those available on the smaller UNIVAC 9200 system. The multiplexer I/O channel of the UNIVAC 9300 system can accommodate up to eight peripheral subsystems. Maximum storage size is 32,768 bytes of plated-wire storage with a cycle time of 600 nanoseconds.

A valuable feature of the configuration is that the UNIVAC 9200/9300 systems may be used as independent computers when not engaged in remote transmissions. Another outstanding feature is that the systems are compatible, being members of the UNIVAC 9000 series which includes the UNIVAC 9400 system, providing for expansion and growth from smaller to larger systems and configurations.

The UNIVAC 9300 II system contains all the basic features and advantages of the UNIVAC 9300, and includes a selector channel suitable for disc input and larger main storage.

CHARACTERISTICS

	9200 SUBSYSTEM	9200 II	9300 SUBSYSTEM	9300 II
ONLINE AND OFFLINE				
MAIN STORAGE	8192 8-bit bytes	same as for 9200	8192 8-bit bytes	16,384 8-bit bytes
NUMBER OF CHANNELS	One multiplexer I/O subchannel	One multiplexer I/O subchannel One selector channel	One multiplexer I/O subchannel	One multiplexer I/O subchannel One selector channel
PRINTING	250 lpm (63-character bar) or 250/500 lpm (48-character bar)	same as for 9200	600 lpm (63-character font) or 1200 lpm (16 character font)	same as for 9300
CARD READING SPEED	400 cpm	same as for 9200	600 cpm	same as for 9300
STUB CARDS	Stub card read (51 and 66 column)	same as for 9200	Stub card read (51 and 66 column)	same as for 9300
UNIVAC 1001 SUBSYSTEM	1000/2000 cpm	same as for 9200	1000/2000 cpm	same as for 9300
CARD PUNCHING SPEED	75 to 200 cpm (column)	same as for 9200	75 to 200 cpm (column) or 200 cpm (row)	same as for 9300
INSTRUCTION EXECUTION	104 microseconds add decimal instruction time (two 5 digit fields)	same as for 9200	52 microseconds add decimal instruction time (two 5 digit fields)	same as for 9300
OFFLINE ONLY				
PREREAD	Preread in punch feedpath	same as for 9200	Preread in punch feedpath	same as for 9300
STACKER SELECT	Punch stacker select control	same as for 9200	Punch stacker select control	same as for 9300
MAGNETIC TAPE	Not available	same as for 9200	UNISERVO VI-C Magnetic Tape Subsystem	same as for 9300
MAIN STORAGE CAPACITY	16,384 8-bit bytes maximum	32,768 8-bit bytes	32,768 8-bit bytes maximum	same as for 9300
DIRECT ACCESS STORAGE	Not available	same as for 9200	UNIVAC 8410 Subsystem	same as for 9300
COMMUNICATIONS	Data Communications Subsystem	same as for 9200	Data Communications Subsystem	same as for 9300
PRINTER LINE SPACING	8LPI	same as for 9200	8LPI	same as for 9300
REQUIRED FEATURES				
PRINT POSITION REQUIREMENT	132 print positions	same as for 9200	Print position expansion	same as for 9300
INTERCONNECTION REQUIREMENT	Channel Adapter 9200/9300	same as for 9200	Channel Adapter 9200/9300	same as for 9300

3. PROGRAMMED SYSTEMS SUPPORT

3.1. GENERAL

The software support provided for the C/SP is designed to provide complete flexibility for implementing communications configurations of all types of terminal hardware while maintaining an expedient user interface. Coding efficiency is achieved by the utilization of a powerful instruction set at the assembly level. System macros are also provided to facilitate the user's requirements. Symbionts available with the UNIVAC 1110 system permit efficient use of card readers, card punches, and printers.

Software to integrate the C/SP effectively with the host computer system is supplied; an assembler and simulator to write and debug user own code on the larger system are included.

The software package is divided into three segments:

(1) resident programs and routines;

(2) programs to operate under the host computer executive system;

(3) modification to host computer elements.

Each of these segments is discussed in further detail in the following paragraph

3.2. RESIDENT PROGRAMS

The resident program software elements are defined as programs which reside entirely or partially in C/SP main storage during their execution. Resident programs are included in:

- Operating System
- Diagnostic Routines
- Intercomputer Handler

3.2.1. Operating System

The C/SP operating system comprises various program modules which are specified by the user at system generation. When supplied elements are used, the following are included:

- Terminal management supervisor (TMS)
- Message control program (MCP)
- Terminal management control routine (TMCR)
- Communications control routines (CCR)
- Peripheral control program (PCP)
- Peripheral control routines (PCR)

System interface is provided for inclusion or addition of user versions to any element specified under the operating system. Figure 3–1 illustrates the functional relationship of these elements.

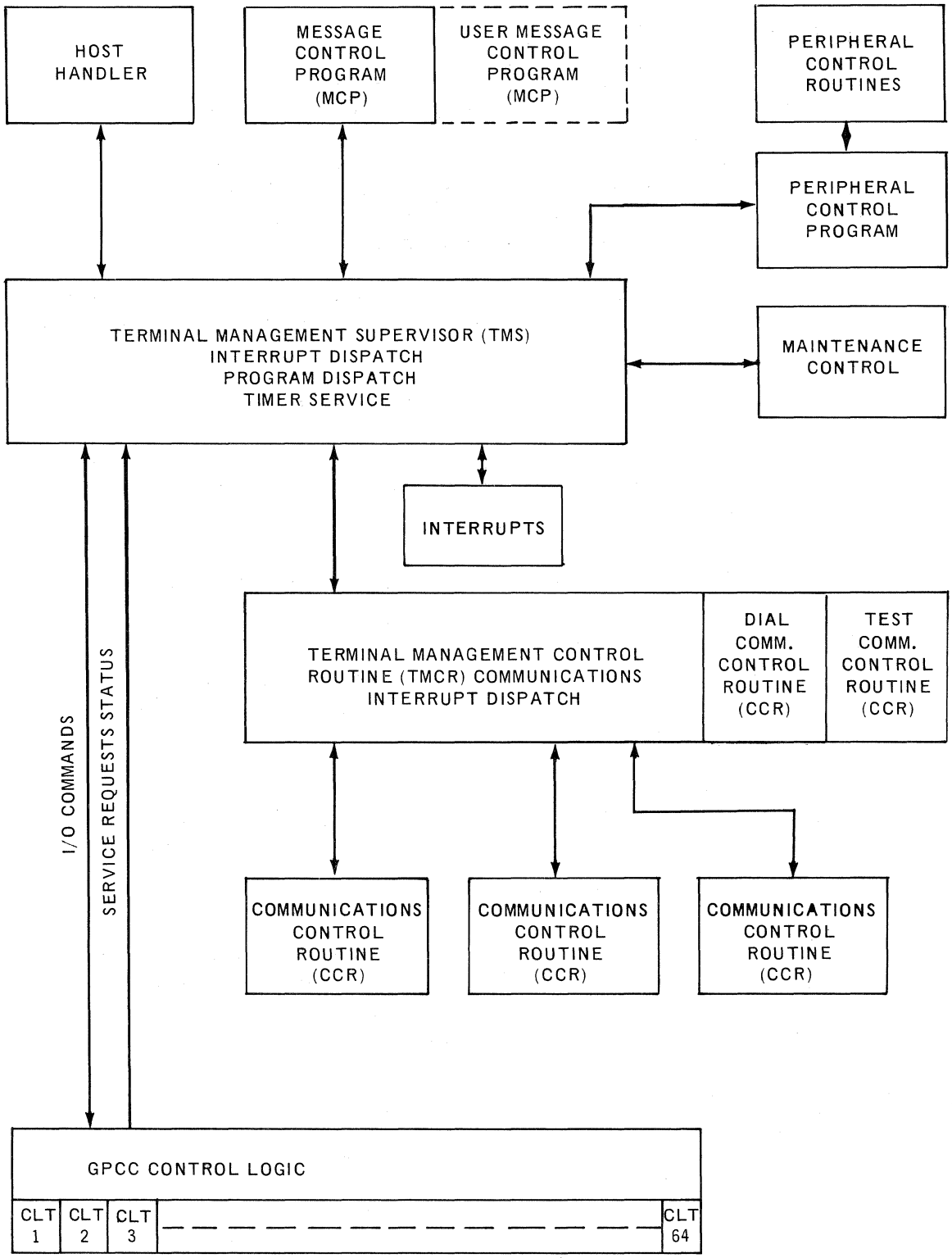


Figure 3–1. C/SP Operating System, Overview

3.2.1.1. Terminal Management Supervisor (TMS)

The TMS is the nucleus of the C/SP operating system and controls all program switching, I/O queueing, and interrupt handling. The modular design of the TMS is based upon the functional divisions within the scope of the TMS activity; this modularity permits the system to be custom made at system generation time for a particular installation. The major areas of the TMS are:

- Program Switching, Dispatching, and Subtasking

  Program switching is based on the following factors: a program of higher priority requires control; expiration of a time slice; and, voluntary release through the WAIT macro instruction.

  Switching is accomplished by executing the highest priority program currently able to run. All programs of equal priority are given equal length time slices. A maximum of seven problem programs, including the MCP, is supported on the C/SP. Within each problem program there are two levels of control or subtasking. These two levels are referred to as primary and secondary activities. The problem program has complete control on the activation and queueing of secondary activities within the program.

- Program Loading and Initialization

  Program loading, together with the concept of dynamic memory allocation, assigns an area of storage into which the program originating from the host computer is loaded.

  The program to be loaded is associated with a relocation dictionary that is produced by the collector on the host computer. This relocation dictionary is used to relocate the program at the origin specified by the dynamic allocator.

- Program Termination

  Problem programs are terminated upon normal request by the user or upon an error condition such as an unsolicited program check without a contingency routine or an unrecoverable I/O error. The program terminator also provides postmortem memory dumps as requested by the user. These and other termination messages are sent back to the host computer.

- Error Logging

  All error statistics are gathered and stored in a buffer. The statistics are transmitted back to the host computer when full. The host computer in turn stores these statistics on a file which is periodically listed on the printer. The error statistics are used for hardware error diagnostic checking and analysis.

- Interrupt Dispatching

  The general entry routine saves the contents of registers in their respective save areas whenever an interrupt occurs. Depending on the type of interrupt, control is passed to the processing routine for that interrupt.

- Contingency Control

  A problem program may request that control be regained upon the following three types of contingencies: I/O contingency, program check contingency, and interval timer contingency. Upon occurrence of the contingency, control is passed to the respective contingency routine provided by the problem program.

- Timer Control

  The timer control routine controls all access to the timer request queue. All time outs requested by the problem program and all time slices requested by the dispatcher are interleaved in the timer request queue on a time-of-day basis.

  The interval timer word (ITW) is loaded dynamically according to the shortest current timer request. If the timer has been loaded and a request is received for a timer interrupt of shorter duration than the one which has been loaded, the queue is updated and the timer is reloaded with the shorter interval.

- Supervisor Call Control

  All supervisor call processing routines are accessed through a jump table which contains entry points for the supervisor modules. These supervisor functions include timer control, I/O requests, and so forth.

- I/O Control

  The I/O control routine determines the source of each I/O interrupt. Control is then passed to the routine associated with the respective channel control routine including the intercomputer adapter routine and the terminal management control routine.

  Dynamic unit assignment is provided whereby problem programs are related to their associated devices by means of a logical number. At I/O execution time, this logical number is translated to a physical device address; thus, a base is provided for device independence.

3.2.1.2. Message Control Program (MCP)

The MCP uses facilities of the TMS, CCRs, and host handler, to accomplish message transmission between remote sites and the host computer.

The MCP is responsible for remote initialization, accomplishing line connection on dial-in and initiation of dial-out, remote sign-on, message routing, message editing, message translation, function dispatching, initiation of polling, and initiation of communications hardware diagnostic programs. Dynamic buffering is performed to support I/O operations with the remote site or with the host computer.

The system MCP is designed to support batch, real time, and demand functions as currently implemented on the UNIVAC 1108 system. In addition, certain user own-code options are available to enable a user to implement various preprocessing capabilities within the framework of the MCP. The user also has the capability to implement user coded MCPs that operate concurrently with the system MCP as a separate problem program.

The MCP is loaded at system initialization time following the system boot from the host computer. The MCP operates in a supervisory state under a different protect key from the supervisor, and operates as the highest priority problem program of the system.

- Remote Initialization

  The MCP initializes a remote line either upon answering a request from the remote site or upon request from the host computer. Where a line is initialized because of the remote request for service, the initialization is fixed. The request for initialization from the host computer is more flexible and permits the user to specify the block size whether or not translation or message editing is to be performed. The device assignment to the host computer is made during initialization and is a pseudo assignment, since the device is actually assigned to the MCP. The MCP dynamically assigns and releases the pseudo assignment to service the needs of the host computer.

- Establishing the Line Connection

  The data line connection (in the case of DATA-PHONE* or TWX* networks) is performed in one of two ways. The first method is the dial-out which is initiated from the host computer. The host computer passes a dial request, along with the dial digits, to the MCP. The MCP then issues the DIAL macro to TMCR to accomplish the actual dial function.

  The second method of establishing a line connection is by the remote site operator placing a call to a DATA-PHONE attached to the C/SP on the GPCC. As the DATA-PHONE rings, an interrupt is generated and passed to the MCP by the supervisor. The MCP gets control at the idle line monitor where a dispatching chain monitored by the MCP is manipulated to instruct the MCP to perform an initial I/O to the CLT. The idle line monitor then relinquishes control and the MCP (upon receiving control) dispatches the appropriate I/O routine to initiate a sign-on sequence with the remote site.

- Remote Sign-on

  After establishing the line connection, the MCP goes through a sign-on procedure if the terminal is batch or demand. The procedure varies according to the mode of operation but will essentially be for the purpose of identification (ID) verification and notification to the host computer that a site is signed on and must be assigned to a symbiont by the executive system.

- Message Routing

  The message routing concerns passage of the data message from the remote site to the host computer or from the host computer to the remote site. The message may be edited for removal or addition of control characters, scanned for significant control images, and optionally translated to or from the remote site or the host computer codes.

Trademarks of A.T.&T. Co.

- Polling

  Polling of multipoint networks is accomplished by the CCR. The request to poll is issued by the MCP. A polling list describing the hardware IDs of the various terminals on the circuit is issued with the poll. When a terminal has traffic to transmit, the data will be accepted by the C/SP when the poll is made. The poll function replaces the idle line monitor for nondialup networks when establishing initial communications with the remote site.

- Dynamic Message Buffering

  I/O messages are transmitted from buffers that are smaller than the size of the image being transmitted. This enables the MCP to acquire storage from a pool as needed to service input functions or to release storage to the pool when it is no longer required on output, thereby reducing the storage requirements for buffers.

- Function Dispatching

  Function dispatching to the various functions in the MCP is accomplished by a line activity dispatcher routine, which determines the sequence of dispatching to the various communications service routines based on priority and first-in first-out requirements. The line activity dispatcher determines if there is any work to be done in the MCP and dispatches to the appropriate routine or releases control of the CPU to the supervisor.

- Communications Hardware Diagnosis

  The MCP, upon detecting unrecoverable errors, initiates diagnosis of a communications path by marking a CLT pair "down" in the line control table describing the CLT. A program load of a diagnostic is initiated as a subtask to the MCP. After completion of the diagnostic, activity is resumed on the good line of the pair.

- Function of the Line Control Table

  The MCP consists of a group of reentrant routines that perform I/O operations with remote terminals in the communications system. Since processing of a message is a periodic function, there may be many messages in various stages of processing by the MCP. The mode of operation requires the MCP to save the message status when a point is reached where no further processing can be done until an event has occurred.

3.2.1.3. Terminal Management Control Routine (TMCR) and Communications Control Routines (CCR)

The routines or programs for controlling the communications terminals on the C/SP are divided into a general handler, TMCR, and the individual CCRs.

TMCR is a series of routines which perform functions common to all CCRs. Some of the functions performed by the TMCR are dial and test mode, CIW preprocessing, and interfacing with the MCP and TMS.

CCRs are designed to compensate for the peculiarities of a given communications service or procedure. The CCRs are reentrant and may support many devices of the same type. Where sufficient sameness in line procedures exists, the CCR may support more than one type of device such as DCT 1000 terminal and UNISCOPE 100 terminal.

In the normal operation of the communications devices attached to the C/SP, any MCP executes I/O packets in the form of supervisor calls (SVC instructions pointing to a packet). The C/SP TMS passes control to TMCR to determine which CCR the packet refers to. The appropriate CCR executes the indicated I/O functions desired and passes back appropriate status information.

Since many of the operations performed are common to all CCRs, the routines necessary for their execution are contained in TMCR.

The TMCR maintains the CIW list and receives control from the TMS whenever an interrupt occurs on the GPCC. After initial processing of the interrupt, control is passed to the appropriate CCR. A master status block (MSB) is maintained for each type of device, service, or procedure which is established at the time of system generation. TMCR and CCRs update information in this block.

The CCRs establish and control the message discipline words (MDW), buffer control words (BCW), and the line status blocks (LSB). Actual hardware I/O control is performed by the CCRs and status information is returned to the MCP which issued the packet.

The following communications devices, services, and procedures are supported with standard CCRs:

- UNISCOPE 100 Display Terminal
- Teletypewriter/UNIVAC Data Communications Terminal DCT 500
- UNIVAC Data Communications Terminal DCT 1000
- UNIVAC Data Communications Terminal DCT 2000
- UNIVAC 1004/9000 Systems
- Binary Synchronous Communications Devices

3.2.1.4. Peripheral Control Program (PCP)

The PCP provides the host computer with an efficient interface to onsite paper peripheral devices. The interface is established to input system run streams and to output system display information. The I/O functions are performed by symbiont routines; the input routines prepare the run stream data for remaining in main storage while the output routines prepare display data for output on print/punch subsystems.

The symbiont complex is external to the C/SP supervisor, operating under a private protect key. The supervisor provides symbiont support routines which interface with the peripheral devices and with the host computer, and which perform various data manipulation functions. Table and buffer areas required for the symbionts are dynamically acquired from and returned to a storage pool within the symbiont complex. The following symbionts are included in the PCP:

- Probe Routine

  The probe routine maintains descriptor items for each onsite input device, probes each inactive device, constructs device control tables, and activates input symbionts.

  The probe of an input device is directed by a probe table which contains an item describing each particular device. When a device is probed and found to contain data for input, the device status becomes active, a device control table is constructed, and the input symbiont which controls input from the device is activated.

- Input Symbiont

  The input symbiont routine directs the device I/O, analyzes the I/O status, and passes data images to the input control routine.

  When input is sensed at the device associated with the symbiont; the symbiont is activated and data images are read from the device. The input buffer is attached to the device control table. A character input mode is standard; however, a column binary mode may be specified by a control statement within the run stream. An interlock condition results in input terminations when no data remains; this interlock condition is marked in the device control table.

- Input Control Symbiont

  The input control symbiont searches the input stream for run start, transmits the run start to the host computer when it is detected, analyzes run images received from the input symbiont, composes data blocks, transmits the run finish control image to the host computer when detected, and transfers data blocks to the host computer.

  Initially, a search is made of the input images for a run control image (run search mode). When a run control image is detected, the following activities are initiated:

  (1) Construction of a run block for use by the host computer to record the new run;

  (2) Run block transfer to the host computer;

  (3) A wait for reply from the host computer pending analysis of the run card image and run identification.

Run images, including the run control image, are translated into Fieldata or ASCII, converted to standard data file (SDF) format, packed into data blocks, and transferred to the host computer. When C/SP storage is limited, partial data blocks may be transferred to the host computer; mass storage may also be used to extend storage capacity, thus enabling the transfer of entire blocks of data.

The run stream is scanned for certain control statement images in order to construct the run file correctly and also for the finish control image which specifies the end of the run. Detection of the finish control image initiates the following activities to terminate the run:

(1) End-of-file sentinel is inserted in last data block;

(2) Finish parameter is set in the data block;

(3) Last data block is transferred to the host computer.

When a run stream is exhausted, the inactive status is set in the probe table and the input symbiont is shifted into an inactive state.

- Output Symbiont

The output symbiont is activated in response to a host computer request for the symbiont to print or punch an output file. Each output image is obtained from the output control symbiont. An output control table is constructed to control the output activity on the particular device and an output buffer is attached to each table.

The output image is then written on the print or punch device with the format (spacing, heading, etc.) controlled by entries in the device control table.

- Output Control Symbiont

The output control symbiont transfers data blocks from the host computer, decomposes data blocks into images, passes output images to the output symbiont, analyzes control images to control output format, and transmits output termination status to the host computer.

The routine prepares print and punch images for an output symbiont and controls the transfer of data blocks from the host computer.

Partial blocks may be requested when C/SP storage is limited; mass storage may also be used to extend storage capacity, thus enabling entire blocks to be transferred. Initially, standard display header images are passed to the output symbiont. The data blocks are separated into images (print, nonprint, end-of-file).

Print images are translated into the output device code and passed to the output symbiont; nonprint images are analyzed to determine if format action (spacing, header, form change, etc.) is required. The format action is indicated to the output symbiont by setting the appropriate switch in the device control table. The image flow is maintained at a level that, when the last print image in a block is passed to the output symbiont, the next data block is ready to be processed. An end-of-file sentinel initiates the following activities to terminate the file processing:

(1) Standard display trailer images are passed to the output symbiont.

(2) Construction of a drop block for use by the host computer to release the output file and mark the device inactive or to initiate another output request for a waiting output file.

(3) Drop block transfer to the host computer.

(4) Deactivate the output symbiont.

- Director Symbiont

  The director symbiont dispatches processing among the various symbiont routines, responds to solicited or unsolicited messages from the host computer, and disperses data blocks received from the host computer.

  Since the symbiont complex controls many onsite I/O devices and interfaces with the host computer executive, it is necessary to dispatch processing time among the various symbiont routines to achieve and maintain efficient performance. The director symbiont activates symbiont routines to maintain efficient performance on each I/O device. Information from the host computer, whether it is output data or system control data, is analyzed and referred to the appropriate symbiont routine. Information originating from the host computer is managed through use of control tables and data queues.

3.2.1.5. Peripheral Control Routines (PCR)

The peripheral control routines are designed to control the I/O associated with devices utilizing the multiplexer channel and the special device channel (SDC) of the C/SP. The types of devices controlled through the PCR include:

- card punches
- card readers
- printers
- paper tape units
- keyboard devices

I/O requests initiated by problem programs, including the symbiont control program, are queued by the channel scheduler in the channel queue. Interrupts are monitored for specific device errors and functions (operator communications requests, intervention required, and others). These peripheral control routines may be considered as a subfunction of the channel scheduler and are responsible for formulating the channel address word (CAW) and device address to be used by the start I/O routine.

Basic error handling routines associated with identifying such errors as card jams, stacker full, forms out, not ready, and so forth are provided by the supervisor. The program is responsible for the channel programs necessary to handle each device.

3.2.2. Diagnostic Routines

Diagnostic routines are included to function with the C/SP operating system and provide an indication of the cause of a malfunction through the host computer primary output stream.

3.2.3. Intercomputer Adapter (ICA) Handler

The ICA handler is resident within the C/SP and controls all commands between the C/SP and the host computer.

The ICA handler uses the first portion of each buffer to send/receive data oriented control information. This information includes site identification and data descriptors. Error status is communicated through data transmissions.

- Initial Load (Boot)

  The C/SP boot function initiates an I/O command containing the boot function, instead of a read or write function, with the host computer. This command causes the host to read data from the C/SP TMS file and transmit the data to the C/SP, a record at a time, until the initial load is completed.

- Load a Program (C/SP)

  At times it may become necessary to load a program into the C/SP that is stored on the host computer mass storage. The request to acquire the program is made through an output command containing the nonsite indicator. The host computer recognizes the nonsite indicator and scans the input data to determine the special function that is requested. Data contains the load program indicator and the program name (file name) of the desired program. The program load routine will retrieve the program and transmit it to the C/SP.

- Write to Host Computer (Mass Storage Device)

  The TMS or user program has the ability to output information to a mass storage device of the host computer. The record to the host computer device includes a nonsite indicator which is set and the file name of the data.

  The user must open and close the file as he would with any host computer device file by using control word options. The control word is transferred along with the data.

- Read from Host Computer (Mass Storage Device)

  The read operation is controlled in the same manner as the write operation, with the requester sending a control record to the host computer device requesting a read of the specified file. The requester must then submit a read command to receive the data when it is transmitted from the host computer.

- Communicate with Host Computer Executive System

  The TMS and host computer are required to communicate information such as line-up and line-down status, console message, and unrecoverable errors. Communication is accomplished by outputting control records, with the appropriate indicators set, to initiate the specific functions requested.

3.3. PROGRAMS UNDER HOST COMPUTER

The following programs operate under control of the host computer operating system.

- C/SP assembler
- Element collector
- Program test simulator
- C/SP service routines
- C/SP symbionts

3.3.1. C/SP Assembler

The C/SP assembler is a one-phase, two-pass bootstrap assembler that operates under the host executive system. The assembler is an efficient easy to use language that satisfactorily handles most of the programming problems encountered by the user. Each machine instruction and data form has simple convenient representations in assembly language. The assembler translates this language into a form that can be executed by the computer. The rules that govern the use of the language are uncomplicated and can be easily applied by the programmer.

Assembly language coding form and operational characteristics are briefly summarized in the following.

- Mnemonic Operation Codes

  A fixed name, from one to four letters, is assigned to each machine instruction; each name suggests the nature of the instruction. Separate names are provided for each of the branch conditions as a further aid in writing the instructions.

- Symbolic Addressing and Automatic Storage Assignment

  Symbolic labels can be assigned to instructions or groups of data. An instruction then references the labeled data by label rather than by storage address. In many cases, other data required by the instruction (such as operand length) can be supplied automatically by the assembler. Another major task of the assembler is to keep an account of all storage locations used for a particular program and to assign all incoming instructions and data to specific locations. The assembler also performs base register and displacement calculations.

- Flexible Data Representation

  Data is represented in binary, hexadecimal, decimal, or character notation, allowing the programmer to choose the most suitable form for each constant.

- Relocatable Programs and Program Linking

  The assembler produces object modules in relocatable form. In this form, the actual storage locations to be occupied by a program need not be specified at assembly time but are determined when the program is loaded. Provisions are made for linking, loading, and executing as one program the results of separate assemblies, thereby reducing the machine time required to make changes to one part of a program. The input to one assembly can be divided into separate sections, each of which consists of a group of instructions and/or data occupying contiguous locations. The relative positions of the various sections can be declared at the time the program is linked.

- Macro Facility

  The assembler includes a macro facility which can reduce the effort required in writing patterns of coding which are repeated in one program or are common among several programs. One command to the assembler results in the inclusion in the object program of many instructions and/or constants, or merely results in establishing one or more values for use elsewhere within the program. The flexibility of the macro facility allows a macro to be written so that the pattern of coding generated can vary widely, depending upon parameters supplied with the call.

- Program Listing

  A printed listing of source and object codes is one output of the assembler. This listing includes error message flags, marking any errors detected by the assembler. Source code errors do not cause the assembler to halt. The assembler continues to process the remainder of the source code, performing its usual error checks, and thus minimizes the number of assemblies required to produce error free code.

3.3.2. C/SP Element Collector

The element collector is a host computer processor which provides a means of collecting independent relative binary elements to produce an absolute program for execution in the C/SP. A relative binary element is an output of the assembler (result of translating C/SP assembler instructions). An absolute program is a program unit with no unresolved references which can be relocated in C/SP storage as an executable program.

The collector operation is divided into three main phases. Each phase is designed to perform a certain function with the ultimate output an absolute program.

- Phase I

  Phase I of the collector processes all control statements and creates tables containing the necessary information for further processing.

- Phase II

  Phase II of the collector processes each input element's entry point table to satisfy all external references. An external reference symbol, from an element's external reference table is satisfied when another element's entry point table contains the same symbol. External references not satisfied through the standard way are satisfied by statements which give values to undefined symbols at collection time or by searching the standard run file or C/SP library file for elements with a matching entry point symbol. The external reference table is modified by adding the symbol value obtained from the entry point table.

- Phase III

  Phase III of the collector merges input elements thus producing an absolute program.

  The collector constructs the entity which may be loaded into C/SP storage and executed. The collection function facilitates combination and debugging of small parts of a large program and combines these individual parts for execution without recompiling the entire set of individual parts.

  A secondary function provided by the collector is the loader interface. The interface provides a means to load an absolute program and/or memory corrections into the C/SP.

  C/SP symbolic elements, relative binary elements, and absolute program elements are maintained under the host computer program file format. The host computer executive system utility routines manipulate these elements.

The C/SP collector is scheduled and activated in response to a host computer executive control statement in an input stream. The C/SP collector is directed by a set of control statements which follow the executive statement. These statements direct the collector in including relative binary elements, modifying collected elements, and storing the absolute program in a user defined file. The interface with the loader is an independent operation. These control statements indicate an absolute program and storage corrections to be loaded into the C/SP.

The functions of collector control statements follow:

- specify absolute program
- include element in collection
- provide value for undefined symbol
- specify corrections to an element
- specify program starting address
- define snapshot dump
- specify collector options
- end of collector language
- specify C/SP storage corrections
- specify loading absolute program

3.3.3. C/SP Simulator

The C/SP simulator is an 1100 series assembler language user program which runs under the host computer operating system. The simulator accepts C/SP object code, simulates execution on the C/SP processor, and provides diagnostic printout to aid in debugging the C/SP program.

The simulator provides means by which the C/SP supervisor or operating system may be developed and tested on the host computer.

The simulator consists of four basic sections: directive processing, instruction cycle simulation, instruction execution simulation, and general purpose communications simulation.

- Directive processing

  Reads directive statements entered through the host computer executive system run stream and provides for definition of the C/SP being simulated, loads and initializes the C/SP object program, and defines the diagnostic aids to be used.

- Instruction cycle simulation

  Provides simulation of machine cycles including interrupts, the interval timer, and the sequential loading of instructions.

- Instruction execution simulation

  Provides for the simulation of the C/SP object code instructions; includes setting the condition code, testing for valid operation codes and operand addresses, and updating total run time.

- General purpose communications simulation

  Provides simulation of one or two GPCCs. Includes the simulation of all GPCC functions such as BCW updating, message disciplines, and interrupt tabling. Timing for communications interrupts is also simulated.

Because the C/SP simulator is designed primarily as a diagnostic aid to C/SP programmers, a comprehensive set of diagnostic aids have been included in the simulator. These diagnostic aids are: post load storage patching, pre-execution and postmortem dumps, unconditional and conditional snapshot dumps, and instruction tracing. All of these options are set by the use of simulator directives of a job control language type which set flags and switches to indicate the simulation of these diagnostics.

3.3.4. C/SP Symbionts for EXEC 8

In order to maintain compatibility with the existing 1108 executive system (EXEC 8) several symbionts were added, preserving as much of the existing symbiont complex as possible without sacrificing the enhancements of the C/SP.

The input and output symbionts control all batch/demand traffic between the C/SP and host computer. These symbionts use the C/SP I/O handler to service the I/O requests and interrupt handling. The I/O symbionts are device independent and handle both batch and demand type runs through reentrant code and user run associated tables.

3.3.4.1. Input Symbiont

The input symbiont accepts batch or demand run streams from the C/SP. For each run stream submitted, a control table is generated to associate the random input records with the proper run stream.

When a record is received from the C/SP, the proper control table is found for the device handling this particular run. The record is associated with the table, and the table is associated with an activity through a switch list from the activity pool. The record is then ready to enter the system in the same manner that records enter from other input symbionts.

Batch records are placed on a drum or disc file to be processed in batch mode. Demand records are held in main storage for the demand program. If the demand program is not processing the input as it is being received, after a few backup records are received, the C/SP is instructed to stop processing input until notified that the backlog has been processed.

As each record is accepted by the system, the control table for that device is placed in an inactive status, and the activity used to process the record is returned to the activity pool to be used when needed.

3.3.4.2. Output Symbiont

The output symbiont processes output files for the C/SP using the standard symbiont interface in EXEC 8, and adding necessary control information needed by the C/SP.

Batch data is read from the drum or disc as fast as it is sent to the terminal by the C/SP. The C/SP sends an acknowledge to the host computer when each record has been sent to the terminal. This acknowledge causes the symbiont to prepare and send the next record.

Demand output resides in host storage until sent to the C/SP. When a backlog of records to be sent to a given device has been built up, the program generating output for that device is deactivated until the backlog is processed.

3.3.4.3. Intercomputer Routine

It is necessary at times to communicate from the C/SP executive system to the host executive system, or the host console. A special nonresident symbiont handles this type of request. The symbiont is loaded and activated by the handler whenever a special intercomputer record is received. The record is passed to the symbiont through the I/O buffer for symbiont processing.

When a request is made for console output, the message is output to the console and the symbiont released if no reply is requested. If a reply is requested, the symbiont is de-activated pending reply. When the reply is received, it is passed to the handler for output and the symbiont is released.

In case of a request for a C/SP system boot, all C/SP I/O is deactivated immediately. The intercomputer symbiont calls the supervisor from host storage and sends it to the C/SP, a record at a time, until the initial load is completed. All pending C/SP I/O is then terminated in error, allowing the C/SP system to initialize and recover.

If the request is to output to host mass storage, the appropriate file is found and the record written.

For a request to load a program to the C/SP, the program file is found on mass storage, and sent to the C/SP.

3.4. MODIFIED HOST COMPUTER ELEMENTS

The host computer software consists of modifications and additions to the host operating system to support the user interface with the C/SP. The software provides the user with the same basic communication philosophy that exists under the host computer executive along with the advantages and flexibility permitted by the C/SP.

The modifications and additions to the host operating system are listed as follows:

- Executive Return
- Interrupt Answering Additions
- Interrupt Processing Additions
- Configuration Additions
- Support Routines

3.4.1. Executive Return

The user interface with the C/SP software is in the form of executive requests (ER). The software necessary to process these user requests is added to the host operating system. The request may involve processing by nonresident portions of the system or processing by resident routines (requests are made for input or output).

The user controls the format and code of the data that is used in communications. When a device address is specified as a binary number, this binary number is available to the requester upon input and is specified by the requester upon output.

3.4.2. Interrupt Answering Additions

The intercomputer adapter operates on an ISI channel of the host computer. To make efficient use of the channel, additions are necessary to interrupt and perform a certain amount of preprocessing of C/SP interrupts. The interrupts that signify the necessity for further processing are queued for the interrupt processing routine.

3.4.3. Interrupt Processing Additions

The interrupt processing additions perform the task of processing the C/SP interrupts queued by the interrupt answering routine and the activation of the user to which the data belongs.

The user may specify action to be taken when an interrupt occurs through the use of completion activities. Activation of a completion activity may be inhibited, automatic, or conditional, depending upon the circumstances.

3.4.4. Configuration Additions

The C/SP represents a new peripheral device to the host computer system and therefore requires a series of tables and lists that can be used by the software to control the usage of the C/SP and the devices connected to the C/SP. These tables and lists may vary with different configurations; therefore, they are generated at system generation time, based upon values supplied by the individual system generator.

3.4.5. Support Routines

To maintain as much user program compatibility as possible, the C/SP is treated as an invisible intermediate with regard to the assignment and release of remote devices. This approach permits the use of existing assignment and release linkages within the host computer operating system, (with modifications) to inform the C/SP of the operation performed.

Modifications are made to the nonresident routines of the host computer operating system that control functions such as assignment, initialization, termination, dial, hangup, and the release of assignments.

The modifications are necessary to inform the C/SP of the operation requested, since the C/SP actually performs the communications with the hardware implied on the various requests.

APPENDIX A. C/SP INSTRUCTION REPERTOIRE

This appendix contains a summary of the C/SP instruction repertoire and includes operation (OP) codes, mnemonic representation, format, and execution time.

FUNCTIONAL DESCRIPTION	MNEMONIC	OP CODE HEXADECIMAL	FORMAT*	EXECUTION TIME (CYCLES)**
Add	A	5A	RX	4
Add half-word	AH	4A	RX	4
Add	AR	1A	RR	2
Compare	C	59	RX	4
Compare half-word	CH	49	RX	4
Compare	CR	19	RR	2
Divide half-word	DH	53	RX	22
Load	L	58	RX	4
Load half-word	LH	48	RX	4
Load	LR	18	RR	2
Multiply half-word	MH	52	RX	20
Shift left single	SLA	8B	RS	3 + N1 (Note 2)
Shift right single	SRA	8A	RS	3 + N1 (Note 2)
Subtract	S	5B	RX	4
Subtract half-word	SH	4B	RX	4
Subtract	SR	1B	RR	2
Store	ST	50	RX	5
Store half-word	STH	40	RX	4
LOGICAL				
AND	N	54	RX	4
AND	NI	94	SI	5 (Note 1)
AND	NR	14	RR	2
Compare logical	CL	55	RX	4
Compare logical	CLI	95	SI	5 (Note 1)
Compare logical	CLR	15	RR	2
Divide polynomial	DP	81	RS	3 + N (Note 2)
Exclusive OR	X	57	RX	4
Exclusive OR	XI	97	SI	5 (Note 1)
Exclusive OR	XR	17	RR	2
Insert character	IC	43	RX	3
Load address	LA	41	RX	4
Move	MVI	92	SI	5 (Note 1)

FUNCTIONAL DESCRIPTION	MNEMONIC	OP CODE HEXADECIMAL	FORMAT*	EXECUTION TIME (CYCLES)**
LOGICAL(continued)				
OR	O	56	RX	4
OR	OI	96	SI	5 (Note 1)
OR	OR	16	RR	2
Shift left single logical	SLL	89	RS	3 + N1 (Note 2)
Shift right single logical	SRL	88	RS	3 + N1 (Note 2)
Store character	STC	42	RX	4
Test under mask	TM	91	SI	5 (Note 1)
BRANCHING				
Branch & link	BAL	45	RX	4
Branch & link	BALR	65	RR	3
Branch on condition	BC	47	RX	2
Branch on condition	BCR	07	RR	1/2 NB/B
Branch on count	BCT	46	RX	4
Branch on count	BCTR	06	RR	2/3 NB/B
Branch on index high	BXH	86	RS	6
Branch on index low or equal	BXLE	87	RS	6
STATUS SWITCHING				
Halt & proceed	HPR	99	SI	***3 (If No Halt)
Load PSW	LPSW	82	SI	***5 (Note 1)
Set storage key	SSK	08	RR	***3
Set system mask	SSM	80	SI	***4 (Note 1)
Supervisor call	SVC	0A	RR	8
INPUT/OUTPUT				
Start I/O	SIO	9C	SI	***6–24

*RR – Register to register
RX – Register to indexed storage
RS – Register to storage
SI – Storage and immediate operand

**1 memory cycle = 630 nanoseconds; all RX instructions add one cycle if double indexing.

***Denotes Privileged Instruction

Note 1 – Number of cycles is one less if not indexing.

Note 2 – N = Number shifts to 16, N_1 = Number shifts to 31 (Modulo 16).

APPENDIX B. COMMUNICATIONS CODE CHARTS

The C/SP is capable of handling data in a variety of communications codes through software control of the CLTs. This appendix contains the following communications codes which are examples of the various levels of code which the C/SP can accept:

- BAUDOT 5-Level
- 7-Level ASCII
- 8-Level ASCII
- EBCDIC

BAUDOT 5-LEVEL CODE

BIT PATTERN	LETTERS	FIGURES (ALTERNATIVE SETS)		
		WEATHER SET	FRACTIONS SET	COMMUNICATIONS SET
00000	BLANK	–	BLANK	≈
00001	T	5	5	5
00010	CARRIAGE RET	CARRIAGE RET	CARRIAGE RET	CARRIAGE RET
00011	0	9	9	9
00100	SPACE	SPACE	SPACE	SPACE
00101	H	↓	BLANK	#
00110	N		7/8	,
00111	M	⑪	.	.
01000	LINE FEED	LINE FEED	LINE FEED	LINE FEED
01001	L	↖	3/4	)
01010	R	4	4	4
01011	G	↘	&	&
01100	I	8	8	8
01101	P	0	0	0
01110	C	◯	1/8	:
01111	V	ⓘ	3/8	;
10000	E	3	3	3
10001	Z	+	"	"
10010	D	↗	$	$
10011	B	⊕	5/8	?
10100	S	BELL	BELL	BELL
10101	Y	6	6	6
10110	F	→	1/4	!
10111	X	/	/	/
11000	A	↑	–	–
11001	W	2	2	2
11010	J	↙	,	,
11011	FIG. SHIFT	FIG. SHIFT	FIG. SHIFT	FIG. SHIFT
11100	U	7	7	7
11101	Q	1	1	1
11110	K	←	1/2	(
11111	LTR. SHIFT	LTR. SHIFT	LTR. SHIFT	LTR. SHIFT

Alternative Sets
① Weather Set
② Fractions Set
③ Communications Set

7-LEVEL ASCII CODE

	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	–	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

Example:

1 0 0	0 0 0 1

= A

$^{b}7,6,5 - 4,3,2,1$

8-LEVEL ASCII CODE

	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	NUL	DLE			space	0					@	P			`	p
0001	SOH	DC1			!	1					A	Q			a	q
0010	STX	DC2			"	2					B	R			b	r
0011	ETX	DC3			#	3					C	S			c	s
0100	EOT	DC4			$	4					D	T			d	t
0101	ENQ	NAK			%	5					E	U			e	u
0110	ACK	SYN			&	6					F	V			f	v
0111	BEL	ETB			'	7					G	W			g	w
1000	BS	CAN			(	8					H	X			h	x
1001	HT	EM			)	9					I	Y			i	y
1010	LF	SUB			*	:					J	Z			j	z
1011	VT	ESC			+	;					K	[			k	{
1100	FF	FS			,	<					L	\			l	|
1101	CR	GS			–	=					M	]			m	}
1110	SO	RS			.	>					N	^			n	~
1111	SI	US			/	?					O	_			o	DEL

Example:

1 0 1 0	0 0 0 1	= A

$^{b}7,6,X,5 - 4,3,2,1$

EBCDIC CODE

	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	NULL		DS		SP	&	–						PZ	MZ		0
0001			SOS				/		a	j			A	J		1
0010									b	k	s		B	K	S	2
0011			FS						c	l	t		C	L	T	3
0100	PF	RES	BYP	PN					d	m	u		D	M	U	4
0101	HT	NL	LF	RS					e	n	v		E	N	V	5
0110	LC	BS	EOB	UC					f	o	w		F	O	W	6
0111	DEL	IL	PRE	EOT					g	p	x		G	P	X	7
1000									h	q	y		H	Q	Y	8
1001									i	'	z		I	R	Z	9
1010			SM			:		`								
1011					.	&	,	#								
1100					<	*	%									
1101						)	—	'								
1110					(	;	>	=								
1111					|	¬	?	"								

Example:

1 1 0 0	0 0 0 1	= A

$^{b}0,1,2,3 - 4,5,6,7$


SPERRY RAND
UNIVAC