NAVSHIPS 0967-280-4010

VOLUME I

TECHNICAL MANUAL

for

DIGITAL DATA COMPUTER
CP-642B/USQ-20(V)

SECTIONS 1 THROUGH 3

SYLVANIA ELECTRIC PRODUCTS, INC.
SYLVANIA ELECTRONIC SYSTEMS
NEEDHAM HEIGHTS, MASS.

DEPARTMENT OF THE NAVY
NAVAL SHIP SYSTEMS COMMAND

Contract: NObsr 93272

Approved 22 SEPT 1967

LIST OF EFFECTIVE PAGES

PAGE NUMBERS	CHANGE IN EFFECT	PAGE NUMBERS	CHANGE IN EFFECT
Volume I		Volume III	
Title Page	Original	Title Page	Original
ii thru xxii	Original	ii	Original
1-i	Original	5-i thru 5-ii	Original
1-0 thru 1-20	Original	5-1 thru 5-48	Original
2-i thru 2-ii	Original	6-i thru 6-vi	Original
2-1 thru 2-22	Original	6-1 thru 6-166	Original
3-i thru 3-iii	Original	7-1 thru 7-216	Original
3-0 thru 3-50	Original	Volume IV	
Volume II		Title Page	Original
Title Page	Original	iii thru iv	Original
ii	Original	8-i thru 8-iv	Original
4-i thru 4-vi	Original	8-1 thru 8-356	Original
4-1 thru 4-260	Original	Volume V	
		Title Page	Original
		ii	Original
		9-i thru 9-ii	Original
		9-1 thru 9-972	Original

Errors found in this publication (other than obvious typographical errors), which have not been corrected by means of Temporary Corrections or Permanent Changes should be reported. Such report should include the complete title of the publication and the publication number (short title); identify the page and line or figure and location of the error; and be forwarded to the Electronics Publications Section of the Bureau of Ships.

All Navy request for NAVSHIPS electronics publications listed in the current issue of NAVSANDA Publication 2002 "Requisitioning Guide and Index of Forms and Publications", Cognizance Symbol I, or in a subsequent issue of the Electronics Information Bulletin should be directed to the appropriate Forms and Publications Supply Point.

RECORD OF CORRECTIONS MADE

CHANGE NO.	DATE	FIELD CHANGE NO.	SIGNATURE

LIST OF SECTIONS

INDEX OF ILLUSTRATIONS

A

B

INDEX OF ILLUSTRATIONS (CONT.)

C

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

INDEX OF ILLUSTRATIONS (CONT.)

LIST OF TABLES

SECTION 1

GENERAL INFORMATION

SECTION 2

INSTALLATION

SECTION 3

OPERATOR'S SECTION

SECTION 4

PRINCIPLES OF OPERATION

LIST OF TABLES (CONT.)

SECTION 5

TROUBLESHOOTING

SECTION 6

REPAIR

LIST OF TABLES (CONT.)

SECTION 7

PARTS LIST

SECTION 9

WIRE TABULATIONS

SECTION 1

GENERAL INFORMATION

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

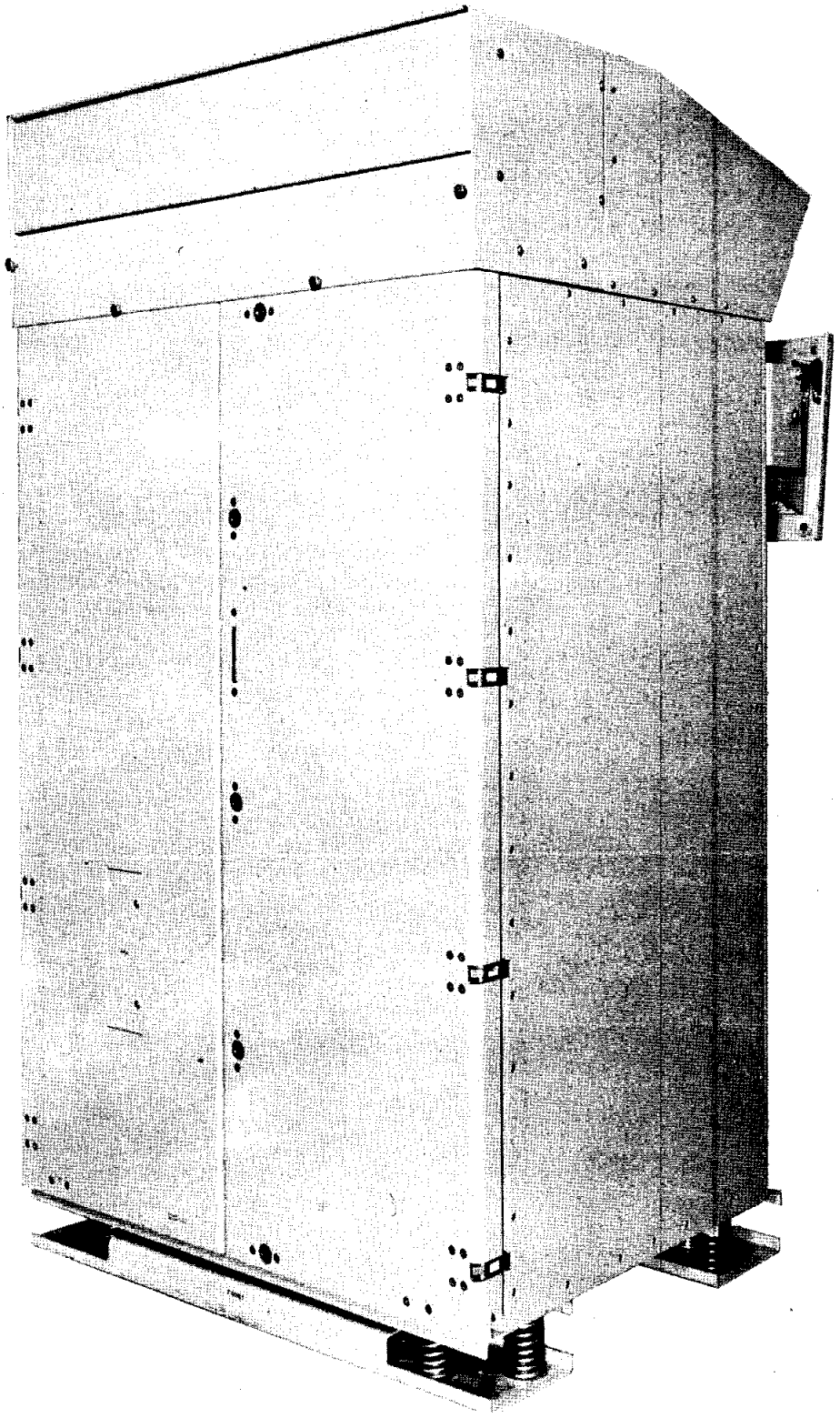


Figure 1-1. CP-642B/USQ-20(V) Digital Data Computer

SECTION 1

GENERAL INFORMATION

1-1. SCOPE

This Technical Manual is in effect upon receipt. Extracts from this publication may be made to facilitate the preparation of other Department of Defense publications.

1-2. GENERAL DESCRIPTION

This manual describes the Digital Data Computer, CP-642B/USQ-20(V) (figure 1-1). The computer is a general purpose, stored program, real-time, digital computer capable of processing a large quantity of complex data where heavy input/output communication is required. Major operational characteristics of the computer are listed in table 1-1.

The computer is suitable for such real-time applications as missile guidance, range safety, process monitoring, and tactical control and display. It may be connected simultaneously to a variety of military-qualified or commercial peripheral equipments. These include:

Teletypewriter units
Magnetic tape units
High-speed printer units
Card read/punch units
Display and display interface equipment
Radar and radar adaption interfaces
Paper tape units
Manual entry devices

The computer is also capable of communicating with a wide variety of other manufacturers' asynchronous external devices in real-time applications. In addition to the peripheral equipment listed, other compatible peripheral equipment include: video processors, various types of displays, digital-to-analog converters, analog-to-digital converters, X-Y plotters, and high-speed radio transmission links.

The computer is housed in a single cabinet, 37 inches deep, 38.5 inches wide, and 72 inches high. Thirteen chassis trays, seven logic, one control memory, and five main memory are horizontally mounted within the cabinet (figure 1-2).

Interconnections between computer chassis are by jacks mounted on the chassis and by movable plug racks. Connections between the computer and other equipment are by jacks mounted on the top of the computer.

Logic modules (figure 1-3) are encapsulated printed circuit modules which plug into the chassis trays (figure 1-4). Maintenance test points are located at the front of the trays.

Computer cabinet doors, closed during normal operation, can be opened for maintenance. The maintenance and control panel, built into the upper part of the cabinet, contains register indicators, set and clear pushbuttons, and operating switches.

TABLE 1-1. COMPUTER OPERATIONAL CHARACTERISTICS

ITEM	CHARACTERISTICS
Word Length Programs	30 bits or 15-bit half-words internally stored and data parity checking
Instruction	62 single address instructions with provisions for address modification via seven index registers
Arithmetic	Parallel ones complement, subtractive
Input/Output	4, 8, 12 or 16 input/output channels with provision for Highspeed or Lowspeed Interface (Lowspeed supplied)
Memories Main	Four microsecond cycle time storage of 32,672 30-bit words
Control	Thin film magnetic with a cycle time of 667 microseconds and capacity of 64 words
NDRO	Two Unifluxor* memories with a cycle time of 667 nanoseconds and a capacity of 32 words
Real-Time Clock	Internal seven day with provisions for an external real time clock

*Trademark of the Western Electric Company

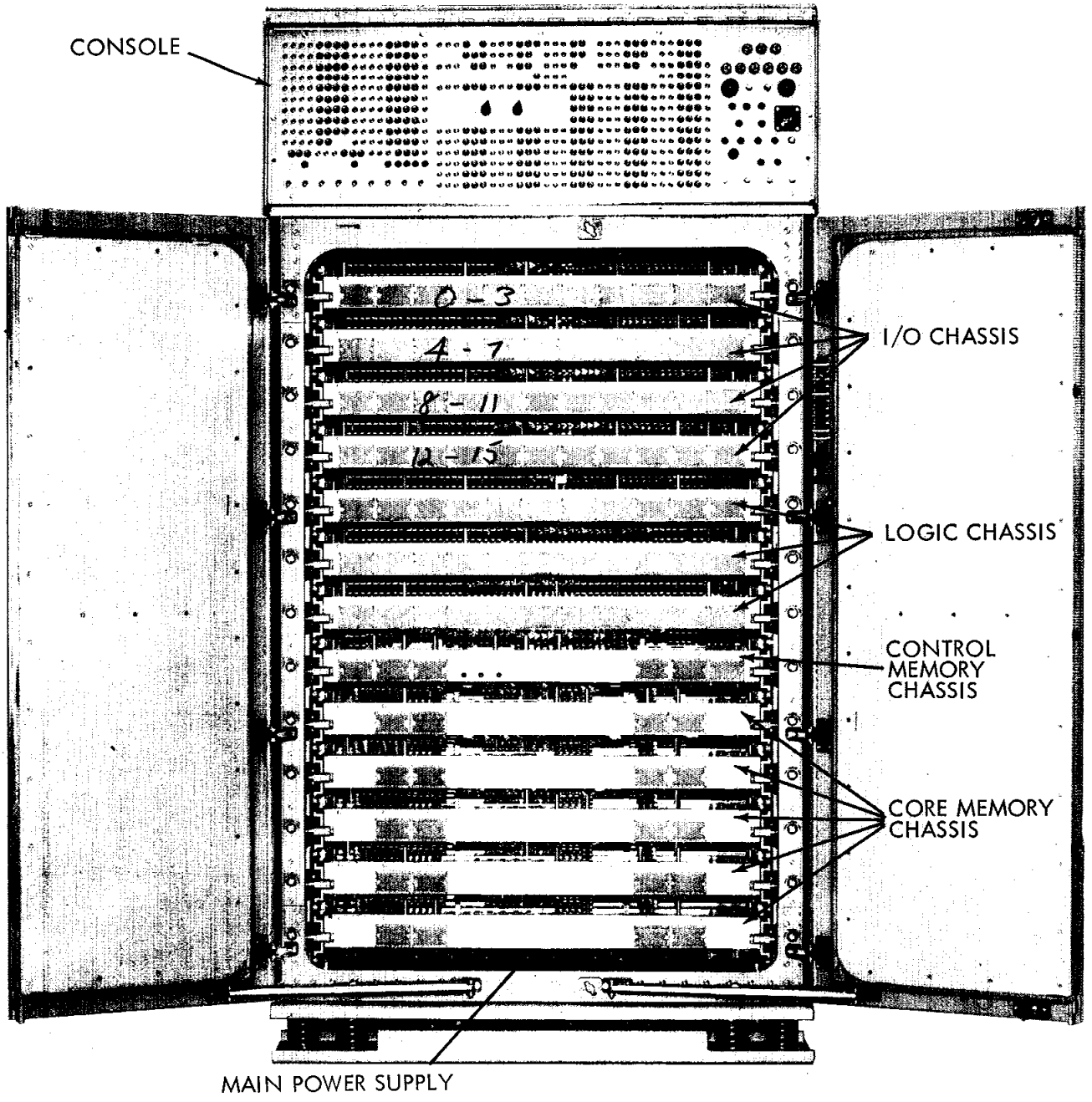


Figure 1-2. Computer, Doors and Console Open

The computer requires 400 cycle, three-phase, 115 volt regulated power for the logic circuits and 400 cycle, three-phase unregulated power for the blower motors.

The computer is equipped with a water-cooled heat exchanger and plenum assembly where ambient air is drawn over the heat exchanger fins and then cools the logic drawers. Interequipment cabling enters the computer at the top of the cabinet and may be run either overhead or through floor ducts.

The computer is designed and constructed to withstand severe shock and vibration. It may be installed aboard ship or in a trailer without modification.

1-3. FUNCTIONAL DESCRIPTION

a. INSTRUCTIONS. - A large repertory of instructions (table 3-4) provides the means for directing the computer to perform the mathematical operations involved in solving problems in real-time. Single address instructions are employed, most of which have an execution time of 8 - 12 microseconds. The computer can also be instructed to perform the data processing necessary for initiating and maintaining communications between, or control of, compatible external equipment.

The instructions are assembled into a routine (program) which is entered into the computer for storage in its memory section. These instructions are usually stored at sequential memory addresses. Short routines can be manually entered into the computer by operating the appropriate console controls. Lengthy routines are entered into the computer via a peripheral device such as a punched paper tape reader or a magnetic tape unit.

The internal storage of the computer consists of a 32,672-word, ferrite-core main memory. A complete cycle for storage of a 30-bit word requires four microseconds. An additional storage area, designated as control memory, provides 64 addressable locations with a read/restore cycle time of 667 nanoseconds. Fifty-six of these locations are special purpose to provide storage for input buffer control words, output command buffer control words, the real-time clock, and seven index registers. The other eight memory locations are used for data storage. Instructions cannot be run from the control memory; however, input/output transfers can take place to or from this memory, and any operand reference can be accomplished.

The instructions contained in the program provide the computer with constants, decision-making capabilities, and an input/output capability. During operation, the program instructions are usually obtained from memory and performed in a sequential manner. However, a program decision may direct the computer to either skip an instruction or to exit from the present routine and enter a subroutine. The routine is terminated when either a predetermined event or conclusion is reached, or when all the instructions have been performed. At this point of the routine, the results of the computer operations are made available to the appropriate external equipment.

Arithmetic and logical operations are performed in the parallel binary mode. For most operations, the result appears in a 30-bit accumulator register. Arithmetic is one's complement, subtractive, with a modulus $(2^{30}-1)$. Computer operation is controlled by a stored program capable of self-modification. Each program instruction contains a function code (six bits), an instruction operand designator (15 bits), and three execution modifiers (two, three or four bits). Execution modifiers provide for address incrementation, operand interpretation, branch condition designation,

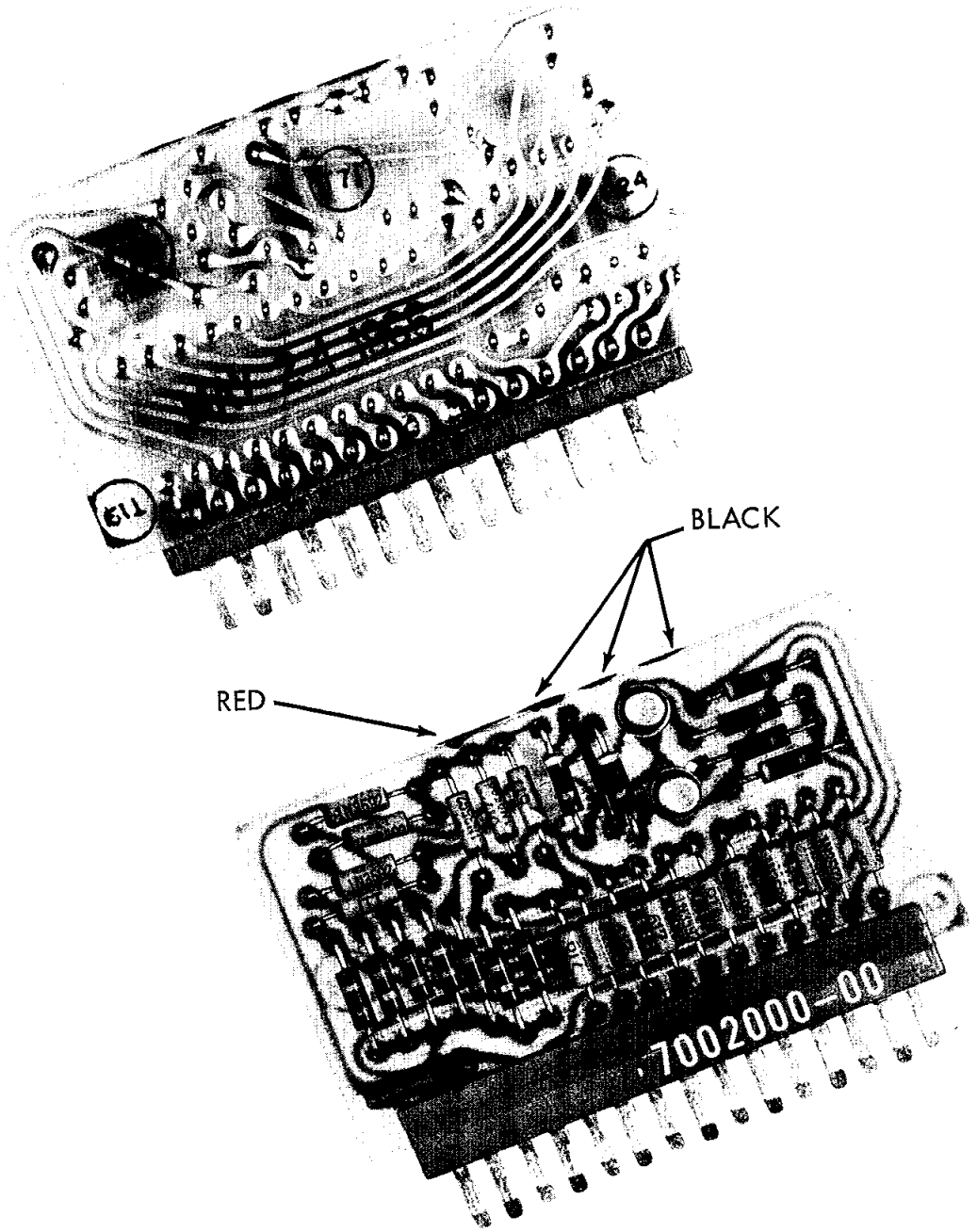


Figure 1-3. Printed Circuit Module, Front and Rear Views

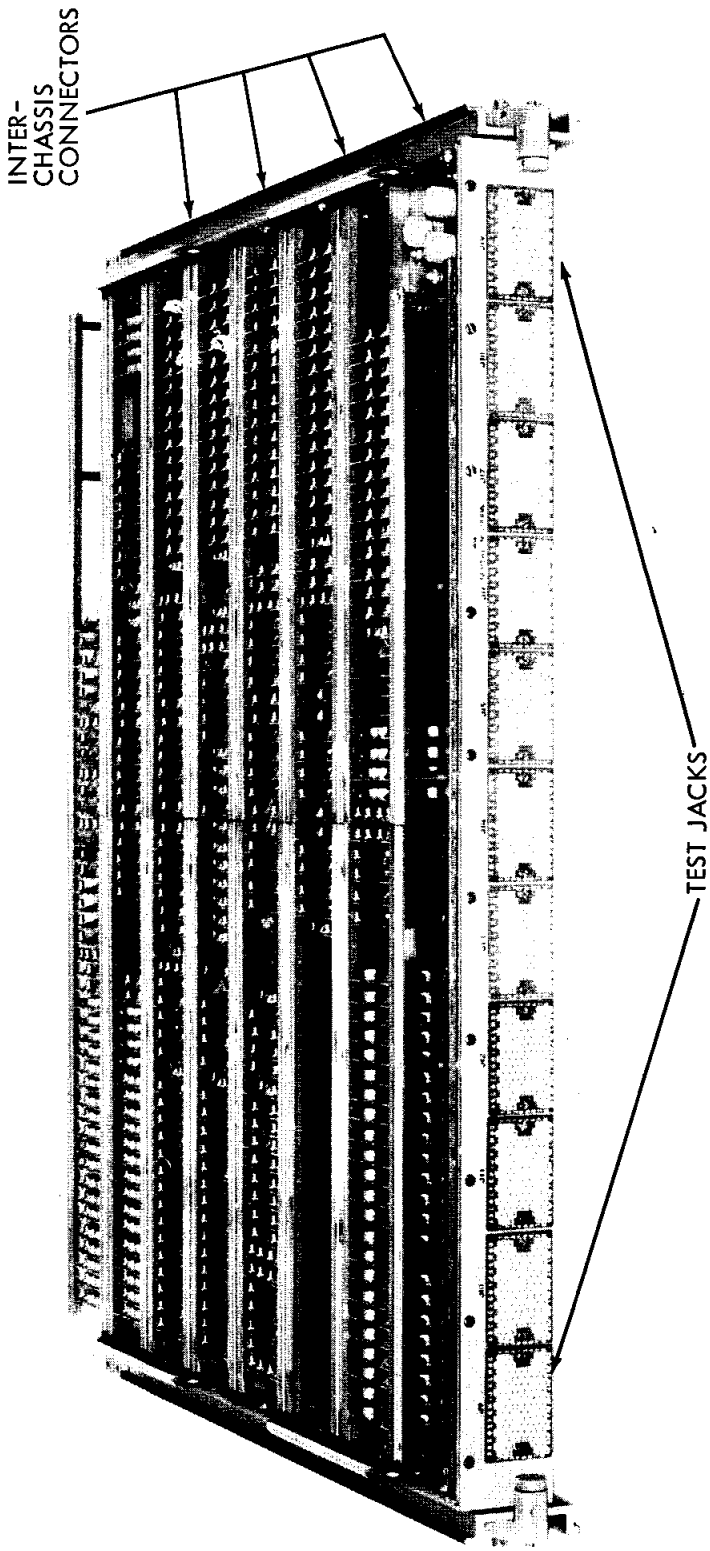


Figure 1-4. Typical Input/Output Chassis

input/output channel or minor function. The operand designator may be increased by the amount contained in any one of seven index (B) registers. The operand specified by the execution address may be interpreted as a 30-bit quantity, or as a 15-bit half-word with or without sign extension. The next sequential program step may be skipped when the branch condition designator places it under control of the contents of either the accumulator or the Q register.

Communication between the computer and its associated external equipment is normally accomplished by a buffered transfer of data, with timing under control of the external device. Operating asynchronously with the main computer program, such transfers of data have independent access to storage. The number of data words transferred is under program control by specifying the first and last memory address in the buffer.

A communication path is established by a sequence of request and response signals between external equipment and computer. The request signal may originate in either the computer or the external device. External request signals interrupt the main computer program and cause the computer to establish a communications channel. Once the communication link has been created, the computer returns to the main program sequence, and transfer of input or output data proceeds without program reference until completed.

Sixteen input and 16 output channels are provided in the computer; each channel consists of 30 parallel data lines plus control lines. Two different interface options are available; the computer is provided with two each Type I and Type II I/O chassis. A group of four channels (one I/O chassis) may be converted from Type I to Type II and vice versa by simply changing plug-in logic modules within the channel circuitry. The Type I input/output chassis (normal) is capable of communicating with peripheral equipment only. The Type II input/output chassis (special) is capable of communicating with either another computer, or by changing plug-in logic modules, with peripheral equipment. The input/output plug-in logic modules provided with the computer provide a low speed interface for the low speed 30-bit parallel mode. Plug-in logic modules are available for the input/output channel circuitry to modify it for a high speed mode.

In addition to data words, output channels carry external function words to the external equipment. These words specify the function that the external device is to perform. Control of the external function signal is accomplished in block transfers. This feature allows the computer to continue engaging an external device after the completion of each function.

Transfers of input and output data are controlled by priority and access control logic. This circuitry assigns access to control and core memory. If two or more requests for access to memory are received simultaneously, the priority and access circuitry evaluates the requests and assigns function priority according to an established sequence.

b. FUNCTIONAL ANALYSIS. - An analysis of computer operation is facilitated by grouping the various logic circuits into broad functional sections. The block diagram (figure 1-5) identifies the main circuits associated with each of the four functional sections within the computer. The timing control circuits are common to all four functional sections.

(1) CONTROL SECTION. - The control section consists of the registers and circuits necessary to procure, modify, and execute the instructions of the program.

The U register (30 bits) is the program control register. It holds the instruction word during execution of an instruction. The function code and the various execution modifiers are translated from appropriate sections of the register. The lower order 15 bits of the U register have additional properties, modulus $2^{15}-1$.

The 15-bit B registers, B1 through B7, store the quantities used for U_L modification. These B registers, also called index registers, occupy the lower 15 bits of the control memory addresses.

The R register (15 bits) has counting properties to increase the contents of a selected B register. The output of this register is made available to the control adder for modifying the lower 15 bits of the U register.

The B register (15 bits) is a nonaddressable control communications register. It holds the quantity added to the lower order 15 bits of the U register during address modification (i.e., the contents of the selected B register).

The P register (15 bits) holds the memory address of a computer instruction word, that of the current instruction, or a new instruction, as a result of a jump condition.

The S register (15 bits) holds the storage address during memory references. At the beginning of a memory cycle period, the address is transferred to the S register. The contents of the S register are then translated to activate the storage selection system.

The SØ register (seven bits) acts in the same manner as the S register except that it holds the address for control memory and NDRO memory during the memory cycle time.

The Z register (30 bits) is the core memory buffer register through which all information to and from a core location must pass. Because of the optional use of 15-bit half-words, Z is split into two 15-bit sections termed Z upper (Zu) and Z lower (ZL). The ZØ register (30 bits) is the memory buffer register for the control and Unifluxor memories. All information read from these memories must pass through this register. All information stored in the control memory locations must pass through this register. No storage is possible directly to Unifluxor locations.

Although the S, SØ, Z, and ZØ registers are shown in the control section they are closely allied with the memory section.

(2) ARITHMETIC SECTION. - The arithmetic section performs numeric and logical calculations.

The A register (30 bits) may be thought of, for programming purposes, as a conventional accumulator. Because of the logic employed, however, the A register is actually only the main rank of the accumulator; the D register serves as a second rank. The add operation is typical of the relationship between the A and D registers: the augend and addend are initially contained in the A and D registers. Before the addition is performed, the contents of the A register are transmitted to the X register. The values of X and D are combined by the add network to form the sum of the two numbers in a parallel manner and placed in the A register.

The Q register (30 bits) is used principally during multiply and divide operations. The contents of both A and Q may be shifted left or right, either individually or as one double-length, 60-bit word.

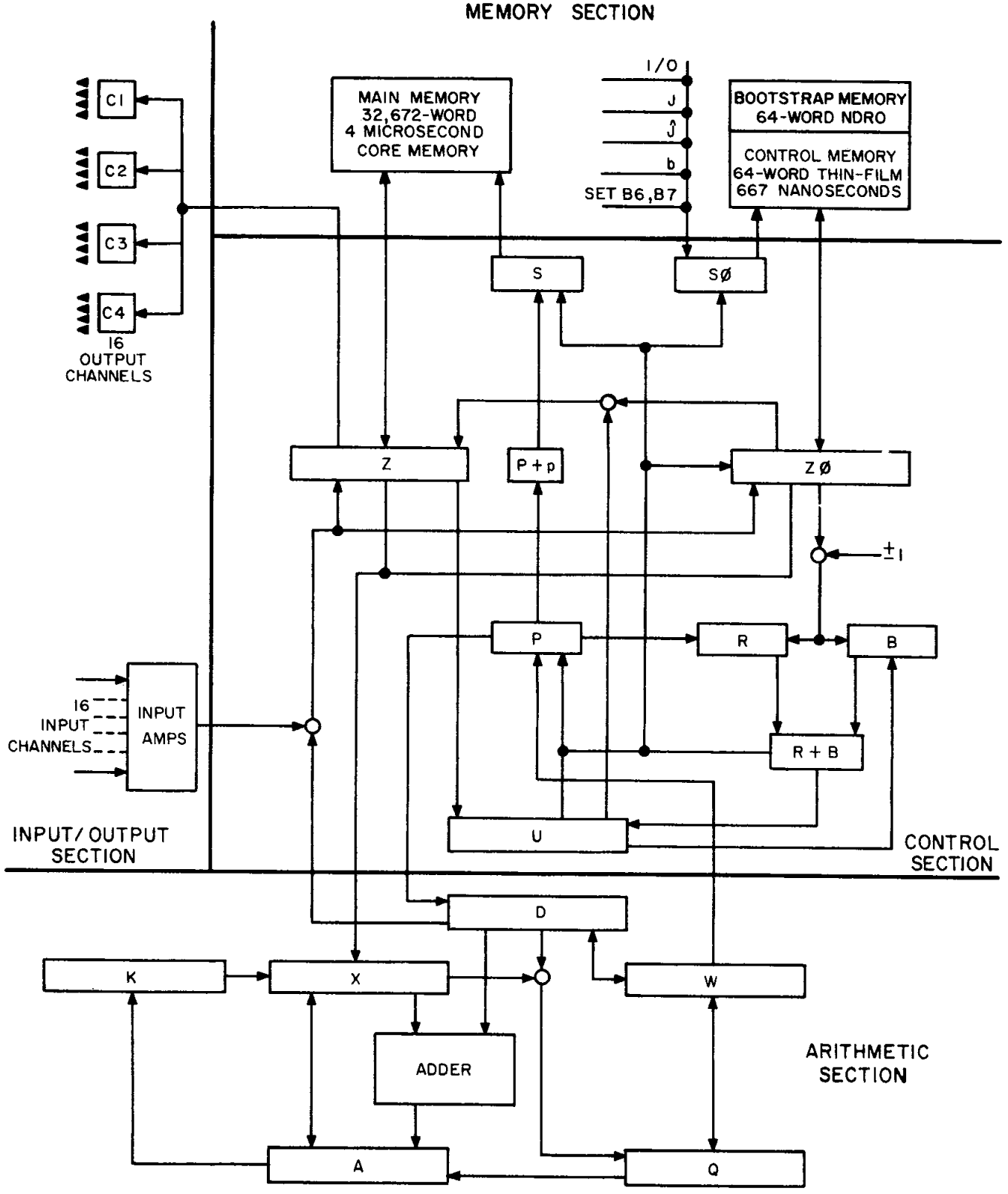


Figure 1-5. CP-642B/USQ-20(V) Computer Block Diagram

The X, D, and W registers are 30-bit, nonaddressable registers. These registers are used primarily for the exchange of data within the arithmetic section and for communicating with the remaining sections of the computer. The W register is not displayed on the control panel of the computer; the A, Q, X, and D registers have indicators which allow the operator to inspect the contents of these registers during debugging and maintenance operations. The A and Q registers are addressable arithmetic registers.

The K registers (K1, K2, and K3) function as a shift counter for all arithmetic operations that involve shifts. Other instructions employing the K registers are multiply, divide, and square root.

(3) MEMORY SECTION. - The memory section consists of three basic memories: main memory, control memory, and Bootstrap memory.

The main memory, constructed of modular arrays of ferrite cores, has a capacity of 32,672 words of 30 bits. It is coincident-current driven and is addressed via the address translator. The memory operates in the destructive readout mode. Time required for the read/restore cycle is four microseconds.

The control memory, constructed from magnetic thin film elements, can store 64 words of 30 bits. This memory stores the seven B-indexing registers, the control words for the input/output section, and the real-time clock. Address locations of the control memory are numbered from 00100 to 00177 and may be used for both read and store operations. No instruction can be run directly from control memory; if this is attempted, the computer will fault.

The storage media of the magnetic thin-film memory consists of spots of a Permalloy ferromagnetic material, deposited upon a thin glass substrate. The Permalloy spots are 50 mils in diameter and 1,000 angstroms thick. The geometry of these spots permits the magnetic state of a spot to be switched in billionths of a second with only a small amount of power applied. Since these spots have only two stable states of magnetization, they can readily store binary information. Cycle time for read/restore of data in the control storage is 667 nanoseconds (2/3 microsecond).

The Unifluxor storage is a nondestructive readout (NDRO) type of memory used in the computer for automatic program recovery or bootstrap programs. This storage area is capable of reading 64 words with a read cycle time of 2/3 microsecond per word. Either one of the two 32-word bootstrap programs in the Unifluxor storage may be selected by a switch. The Unifluxor memory may be entered from any point in a program, and the exit from this memory area requires no special instruction.

(4) INPUT/OUTPUT SECTION. - Communication with peripheral equipment is carried on in a 30-bit parallel mode through the input/output channels. The computer is equipped with 16 channels, numbered decimally 0 through 15. These channels are assembled on four chassis, each of which contains four identical input/output channels. Two different types of chassis can be used with the computer: Type I (normal chassis) communicates with peripheral equipment, and Type II (special chassis) communicates either with peripheral equipment or with another computer. The conversion from a peripheral equipment channel to an intercomputer channel on Type II chassis involves the changing of one plug-in jumper logic module per channel. Either of the two types of chassis can be provided with a high or a low speed interface. The low speed interface provides communications transfer rates

of up to a nominal 40 kc per channel. The high speed interface will provide transfer rates of up to 125 kc per channel.

In the computer, the transfer of input and output data words is asynchronous with the computer program, but the program maintains synchronous control over the issuance of external function words. Transmission of external function words is handled the same as data transmission. The peripheral equipment sets a line indicating it is capable of accepting a control word from the computer; therefore, the transmission of the word is not synchronous with the computer program. Provision has been made, however, to achieve synchronization of program and I/O control to be compatible with existing peripheral equipment.

The four C registers (C1, C2, C3 and C4) hold information for peripheral equipment during output data or external function transfers. Each is 30 bits in length and acts as a buffer register for four output channels. Groups of four channels may be adapted to Type I or Type II communication.

(5) TIMING CONTROL. - Timing control of the computer operations is provided by internal master clock and real-time clock circuits.

(a) MASTER CLOCK. - The master clock consists of a delay line oscillator developing a four-phase output with a cycle time of approximately 680 nanoseconds. The duration of each clock pulse is approximately 130, plus or minus 15, nanoseconds.

(b) INTERNAL REAL-TIME CLOCK. - The internal real-time clock consists of a tuning fork-controlled oscillator circuit operating at a frequency of 1024 cycles per second. The real-time clock is used to advance the count in memory address 00160 and is accurate to plus or minus two cycles in 10,240 cycles.

(c) EXTERNAL REAL-TIME CLOCK. - Provision is made for connecting an external real-time clock to the computer. The connection is made through a three-conductor connector located on the input/output connector panel. In order to use the external timing device, a logic module located on chassis A5 must be relocated at a different coordinate on the chassis. (See Section 5.)

c. MAINTENANCE AND CONTROL CONSOLE. - The maintenance and control console (see figure 3-1) located on the upper front of the computer, includes indicator lamps which display a detailed report of the internal status of the computer and controls to permit manual initiation of various operations. It is not necessary during normal operations, however, to monitor the console.

Each register is represented on the console by a row of indicator-switch modules each of which can be used to enter a "one" manually into the corresponding bit position. A clear switch is associated with each register and is used to enter "zeros" manually into all bit positions of the register. Many of the registers are involved only in the mechanics of executing instructions and are not directly accessible to the program.

The maintenance and control console provides manual controls for selecting the following special modes of operation:

Execution of one program instruction for each depression of a switch

Execution of consecutive program instructions at a low controllable rate

Execution of one master clock phase each time a switch is pressed

Execution of consecutive master clock phases at a low, controllable rate

Operation that is normal except that the computer does not stop for a programmed STOP instruction. (Such operation is called abnormal high-speed operation.)

The console also provides manual controls that may be used either to disable the real-time clock, to inhibit the decrementing of the B7 register, or to inhibit incrementing of the program address (P) register. These options enable the operator to suspend normal operation temporarily without affecting subsequent operation. Such suspensions could include stopping the computer or operating temporarily in one of the special modes.

1-4. QUICK REFERENCE DATA.

Quick reference data for the computer is summarized in table 1-2.

1-5. EQUIPMENT SUPPLIED.

The equipment normally supplied under a computer contract is tabulated in table 1-3.

1-6. EQUIPMENT AND PUBLICATIONS REQUIRED BUT NOT SUPPLIED.

Equipment, test equipment, tools, and publications required but not normally supplied with the computer are tabulated in table 1-4.

1-7. FACTORY OR FIELD CHANGES.

This manual includes all factory and field changes to the CP-642B/USQ-20(V) computer up to the publication date of this manual.

1-8. EQUIPMENT SIMILARITIES.

The CP-642B/USQ-20(V) computer, covered in this manual, is similar to the CP-642B computer and is an improved model of the CP-642/USQ-20(V) and CP-642A/USQ-20(V) computers. The similarities and differences of these units are tabulated in table 1-5.

1-9. SHIPPING DATA.

Information pertinent to shipping may be extracted from table 1-2.

1-10. MODULE COMPLEMENT.

(See figure 1-3.) The basic building unit of the computer is a printed-circuit module containing the necessary transistors, resistors, diodes, etc., to perform logic functions. Several different types of these modules are used in the computer. These module types and quantities are tabulated in table -16.

TABLE 1-2. QUICK REFERENCE DATA

ITEM	CHARACTERISTICS
POWER INPUT	
LOGIC (DC SUPPLIES)	
VOLTS	115 (±2%) VAC (Line to line)
FREQUENCY	400 (±5%) cps
PHASES	3
WATTS	2500
PROTECTION	15 Ampere fuse each line
BLOWER	
VOLTS	115 (±10%) VAC (Line to line)
FREQUENCY	400 (±5%) cps
PHASES	3
WATTS	2000
PROTECTION	15 Ampere fuse each line
OPERATING TEMPERATURE RANGE	0° - 50°C (32° - 104°F)
COOLING	
WATER TEMPERATURE	21.1° ± 2.8°C (70° ± 5°F)
WATER RATE	6.3 GPM
OVERTEMPERATURE FEATURES	
OVERTEMP WARNING	Alarm Horn and Light at 46°C (115°F)
OVERTEMP SHUTDOWN	Shutdown at 60°C (140°F)
SIGNAL CHARACTERISTICS	
INTERNAL	
"1"	-4.5 VDC
"0"	0 VDC
I/O LINES	LOW SPEED INTERFACE / HIGH SPEED INTERFACE
"1"	0 VDC / 0 VDC
"0"	-13.5 VDC / -3 VDC
INPUT/OUTPUT CHANNELS	
PERIPHERAL I/O	0, 4, 8, 12, or 16
INTERCOMPUTER I/O	0, 4, 8, 12, or 16
OPERATION	30 bit, parallel mode
INTERNAL FEATURES	
MAIN MEMORY	Ferrite cores
CAPACITY	32,672 thirty-bit words
CYCLE TIME	four microseconds
CONTROL MEMORY	Thin Film devices
CAPACITY	64 thirty-bit words
CYCLE TIME	667 nanoseconds

TABLE 1-2. QUICK REFERENCE DATA (CONT.)

ITEM	CHARACTERISTICS
BOOTSTRAP	Unifluxor devices
CAPACITY	64 thirty-bit words
CYCLE TIME	667 nanoseconds
INSTRUCTION REPERTOIRE	64 Function Codes (two of which are invalid)

MAIN POWER SUPPLY CHARACTERISTICS			
DC OUTPUT	VOLTAGE RANGE	RIPPLE VOLTS (MAX. P. TO P.)	FUSE (AMPS)
-15 Volts	-13.5 to -16.5	0.20	VARIES BY CHASSIS
+15 Volts	+13.5 to +16.5	0.20	
+10 Volts**	Adjustable from +9 to +11	0.10	
-4.5 Volts	-4 to -5	0.20	
+18 Volts	+16.2 to +19.8	0.10	

CONSOLE POWER SUPPLY CHARACTERISTICS			
-90.5 Volts (single phase half-wave rectified)	-80 to -100 (peak-to-peak)	--	*1
-54 Volts	-48 to -60	--	*1
-15 Volts	-13.5 to -16.5	--	
-26.5 Volts	-23.8 to -29.2	--	1

* Fused together in power supply primary

** Output from voltage regulator module located at J37E on each memory chassis (A9 thru A13)

TABLE 1-3. EQUIPMENT SUPPLIED

QUANTITY PER EQUIPMENT	NOMENCLATURE		OVER-ALL DIMENSIONS*					TECHNICAL MANUAL
	NAME	DESIGNATION	HEIGHT	WIDTH	DEPTH	VOLUME	WEIGHT	
1	Digital Data Computer	CP-642B /USQ-20(V)	72	38.5	37	102,564	2,150	NAVSHIPS

*Dimensions in inches, cubic inches, and pounds.

TABLE 1-4. EQUIPMENT AND PUBLICATIONS REQUIRED BUT NOT SUPPLIED

QUANTITY PER EQUIPMENT	NOMENCLATURE		REQUIRED USE	REQUIRED CHARACTERISTICS
	NAME	DESIGNATION		
1	Analyzer	Multimeter AN/PSM-4C or equivalent	Measure AC and DC voltages and measure resistances	Provide facilities for checking primary power, and for general troubleshooting
1	Oscilloscope	Tektronix 545 or equivalent with CA dual trace plug-in preamplifier, or equivalent	Check voltage waveforms	Provide facilities for adjustments and for general troubleshooting
1	Tool Set	Test-Tool Set AN/USM-3 or equivalent	General purpose	
1	Hand Stripper	Ideal Industries No. 45-171	Strip insulation from wire	
1	Hand Crimper	Aircraft Marine Products No. 47566	Crimp wire in taper pins	Provide facilities for making taper pin connections if necessary

TABLE 1-4. EQUIPMENT AND PUBLICATIONS REQUIRED BUT NOT SUPPLIED (CONT.)

QUANTITY PER EQUIPMENT	NOMENCLATURE		REQUIRED USE	REQUIRED CHARACTERISTICS
	NAME	DESIGNATION		
1	Hand Insertion Tool	Aircraft Marine Products No. 380306 with No. 394042 insertion tip	Insert taper pins in connectors	
1	Wrapping Tool	14R2 (8130-132)	Make wire wrap connections	
1	Power Pack	503885 (8130-151)	Battery for wrapping tool	
1	Unwrapping Tool			
	Dual for 30 ga.	505084	Unwrap wire-wrap	Unwraps left or right.
	Dual for 20-26 ga.	A31478	Connections	Hand wrapped connections
1 ea.	Bits for 30 ga.Sq. Pin 30 ga. Rect.Pin 24 or 26 ga.	504221) 505561(8130-144) 26263 (8130-121)	Adapts wire wrapping tool to various wire sizes	
1 ea.	Sleeves for 30 ga. Sq. Pin 30 ga. Rect.Pin 24 or 26 ga.	505350 17611-2 (8130-129) 18840 (8130-128)	Adapts wire wrapping tool to various wire sizes	

TABLE 1-5. EQUIPMENT SIMILARITIES

ITEM	CP-642B or CP-642B/USQ-20(V)	CP-642A/USQ-20(V)	CP-642/USQ-20(V)
DIMENSIONS			
HEIGHT	72	72	65-1/2
WIDTH	38.5	37-3/8	37-3/8
DEPTH	37	36-7/8	36-7/8
WEIGHT	2,315	2320	2320
VOLUME	95,536	95,536	85,700
MOUNTING	Shock Mounted	Shock Mounted	Shock Mounted
POWER			
LOGIC	115 (±2%) VAC, 3 phase 400 (±5%) cps, 2500 watts	115 (±2%) VAC 3 phase 400 (±5%) cps, 2500 watts	115 (±2%) VAC 3 phase 400 (±5%) cps, 2500 watts
BLOWER	115 (±10%) VAC, 3 phase 400 (±5%) cops 2000 watts	115 (±10%) VAC 3 phase 400 (±5%) cps, 2000 watts	115 (±10%) VAC, 3 phase 400 (±5%) cps, 2000 watts
COOLING	Recirculating air, water cooled	Recirculating air, water cooled	Recirculating air, water cooled
OVERTEMPERATURE			
WARNING	46°C (115°F)	46°C (115°F)	46°C (115°F)
SHUTDOWN	60°C (140°F)	60°C (140°F)	60°C (140°F)
SIGNAL CHARACTERISTICS			
INTERNAL			
"1"	-4.5 VDC	-3 VDC	-3 VDC
"0"	0 VDC	0 VDC	0 VDC
I/O LINES			
"1"	0 VDC 0 VDC	0 VDC	0 VDC
"0"	-13.5 VDC -3 VDC	-13.5 VDC	-13.5 VDC
STORAGE			
MAIN MEMORY	Ferrite Cores	Ferrite Cores	Ferrite Cores
CAPACITY	32,672 30-bit words	32,768 30-bit words	32,768 30-bit words
CYCLE TIME	four microseconds	eight microseconds	eight microceconds
CONTROL MEMORY	thin film		
CAPACITY	64 30-bit words		
CYCLE TIME	667 nanoseconds		
BOOTSTRAP MEMORY	Unifluxor devices	Wired jumper plugs	Wired jumper plugs
CAPACITY	64 30-bit words	16 30-bit words	16 30-bit words
CYCLE TIME	667 nanoseconds	eight microseconds	eight microseconds

TABLE 1-5. EQUIPMENT SIMILARITIES (CONT.)

ITEM	CP-642B or CP-642B/USQ-20(V)	CP-642A/USQ-20(V)	CP-642/USQ-20(V)
INPUT/OUTPUT CHANNELS	16 CP-642BCP-642B/USQ--20(V)	14	14
PERIPHERAL I/O	16 Max. 14	12	12
INTERCOMPUTER I/O	16 Max. 2	2	2
MASTER CLOCK PHASE	4	4	4
CYCLE TIME	680 nanoseconds	1.6 microseconds	1.6 microseconds
CLOCK PULSE DURATION	150 nanoseconds (±10 nsec)	0.4 microseconds	0.4 microseconds
CONSOLE LOCATION	Top of cabinet	Top of cabinet	Inside of doors
REPERTORY OF INSTRUCTIONS	64_{10}	64_{10}	64_{10}

TABLE 1-6. MODULE COMPLEMENT

MODULE TYPE	CHASSIS 1	2	3	4	5	6	7	8	*9-13	TOTALS	REMARKS
**0210											
1110											
1120											
1200											
**2000					60	2	150	45		257	
2011	21	21	24	24	32	39	41	24	1	231	
2020	106	106	117	117	58	107			6	641	
2030	35	35	37	37	22	23	1			190	
2040	3	3	5	5	25	35	86			162	
2050			1	1	48	45	73			168	
2060	3	3	15	15	56	90		6	2	198	
2070	13	13	15	15	35	39	27	11	1	173	
2080	4	4	4	4	4	9	10			39	
2090	68	68	68	68	16					288	
2100					5					5	
2110					2				1	7	
2120					1				1	6	
2130	16	16	16	16						64	
2140	30	30	30	30						120	
2150	23	23	23	23	32	33	40	15		212	
2160									6	30	
2170									8	40	
2180									16	80	
2191									10	50	

TABLE 1-6. MODULE COMPLEMENT (CONT.)

MODULE TYPE	CHASSIS									TOTALS	REMARKS
	1	2	3	4	5	6	7	8	*9-13		
2200									4	20	
2210									16	80	
2350									1	5	
2360					4					4	
2370								1		1	
2380								5		5	
2390								8		8	
2400								8		8	
2410								2		2	
2420								4		4	
2430								1		1	
2440								2		2	
2450								8		8	
2460								8		8	
2470								8		8	
2480								2		2	
2490								2		2	
**2500								4		4	
2510								39		39	
2520								12		12	
2530								2		2	
2540								4		4	
2550								15		15	
2560								32		32	
2570								30		30	
2580								30		30	
2590								8		8	
2600								1		1	
5000									8	40	
2620									16	80	
2631									16	80	
2640									1	5	
2650			4	4						8	
2660			4	4						8	
2720	1	1	1	1						4	
2740								1		1	
2750								1		1	

TABLE 1-6. MODULE COMPLEMENT (CONT.)

MODULE TYPE	CHASSIS 1	2	3	4	5	6	7	8	*9-13	TOTALS	REMARKS
2760					1					1	
2770					1					1	
2850									24	120	
2880					1					1	
2890					1					1	
3180									13	65	
5000					1					1	
250210											
251110										9	Mounted on Console
251120										1	Mounted on Console
251200					1					1	
TOTALS	323	323	364	364	406	422	428	339	151	3724	

* Quantity for one chassis; multiply by five for memory total.
** For complete module type, prefix each number with 25 for gold-plated connector or 700 for solder plated connector.
*** For complete module type, prefix each type number with 422 for gold-plated connectors or 700 for solder plated connectors

SECTION 2

INSTALLATION

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 2

INSTALLATION

2-1. UNPACKING AND HANDLING.

The computer will arrive at the site completely assembled. If shipped by truck, the computer requires no crating. When it is shipped by other means of transporation, the computer requires the protection of a standard crate. In either case, a skid bolted to the bottom of the computer permits the use of a forklift truck when handling the computer. To uncrate the unit, complete the following steps.

NOTE

Omit steps 1, 2 and 3 if the computer arrives uncrated.

STEP 1. Remove the fasteners from the top of the shipping crate.

STEP 2. Remove the top section of the crate.

STEP 3. Pull away the four sides of the crate to expose the equipment.

STEP 4. Move the computer to the desired area.

STEP 5. Remove the bolts which hold the equipment to the skid.

STEP 6. Lift or slide the computer from the skid.

STEP 7. Place the computer in the area as shown by the installation plans.

STEP 8. Perform a brief visual inspection of the equipment; note and report any damage.

2-2. SITE SELECTION.

The placement of the computer is optional and is dependent upon the installation site. Conditions must be suitable to protect the computer from damage. The equipment is self-contained but requires floor and wall supporting structures. There must be sufficient area around the back of the cabinet to allow access to the electrical and coolant connectors and to facilitate maintenance and repair procedures.

2-3. POWER REQUIREMENTS AND DISTRIBUTION.

The total power dissipation of the computer circuits and the centrifugal blower is 4,500 watts. The computer logic circuits require 2,500 watts of 3-phase power at 115 volts ($\pm$1 percent) and 400 cps ($\pm$5 percent) with a current requirement of 14 amperes. The centrifugal fan requires 2,000 watts of 3-phase power at 115 volts ($\pm$10 percent) and 400 cps ($\pm$5 percent) with a current requirement of 25 amperes starting and 9 amperes running.

2-4. INSTALLATION LAYOUT.

The installation layout for the computer and peripheral equipment is a function of the installing activity. When planning the installation layout, it must be

remembered that the maximum distances between the computer and peripheral units are limited by the type of interconnecting cables and the interface speeds (high speed or low speed interface) which are used. Refer to paragraph 2-5 for recommended cables.

2-5. INSTALLATION REQUIREMENTS

a. FLOOR AND SPACE REQUIREMENTS (See figure 2-1). - The floor area upon which the computer is to be mounted must be flat within 0.03 inch and unobstructed. There must be at least 17 inches clearance between the computer sides and walls or other equipment to allow space for the doors to be opened. There must be at least 23 inches clearance between the front of the computer and the wall or other equipment to allow space for removing the chassis. There must be at least seven inches head room to allow adequate air circulation and work space. There must be at least four inches clearance between the rear of the computer and the wall to allow space for cables.

b. COOLING REQUIREMENTS. - The cooling coil on the back of the computer requires 6.3 gallons-per-minute of water at approximately 21°C (70°F). The water supply inlet is a 3/4-inch NPT fitting located near the back on the right side of the plenum section. When the computer is installed, the inlet is approximately 33 inches from the floor and approximately 5-1/2 inches from the wall behind the computer. The water return outlet from the computer is a similar pipe fitting located 4-1/2 inches directly above the inlet. Flexible lines should be used to allow variation of 0.5-inch movement in any direction after installation.

c. DIMENSIONS AND WEIGHT. (Approximate, uncrated.) - The computer is 72 inches high (including shock mounts), 38.5 inches wide, and 37 inches deep. Cubic content (volume enclosed by over-all dimensions) is approximately 102,564 cubic inches. The total weight of the computer is 2150 pounds. This includes the weight of the cabinet, one power supply, heat exchanger, base and shock mounts totaling 1189 pounds, and weight of the seven chassis, A1 through A7, 588 pounds (each chassis 84 pounds); chassis A8, 73 pounds; chassis A9 through A13, 300 pounds (each chassis 60 pounds).

NOTE

The computer is provided with four eyebolts to facilitate lifting during installation. The eyebolts are located on top of the chassis frame and are accessible only when the top cover is removed.

d. INTERCONNECTING DIAGRAMS. - Connections between the chassis of the computer are made via movable plug-racks on the computer's sides and jacks mounted on the chassis. Figure 2-2 shows these plug designations. Connections between the computer and external equipment are via the jacks located on the top of the computer (see figure 2-3). Functions of these jacks are listed in table 2-1. Power connections to the computer are shown in figure 2-4. Power connections internal to the computer are shown in figure 2-5.

NOTE

Interconnecting cable types and the conductor grouping may vary among installations. Refer to the applicable ship or station plans to determine the correct cabling for any installation.

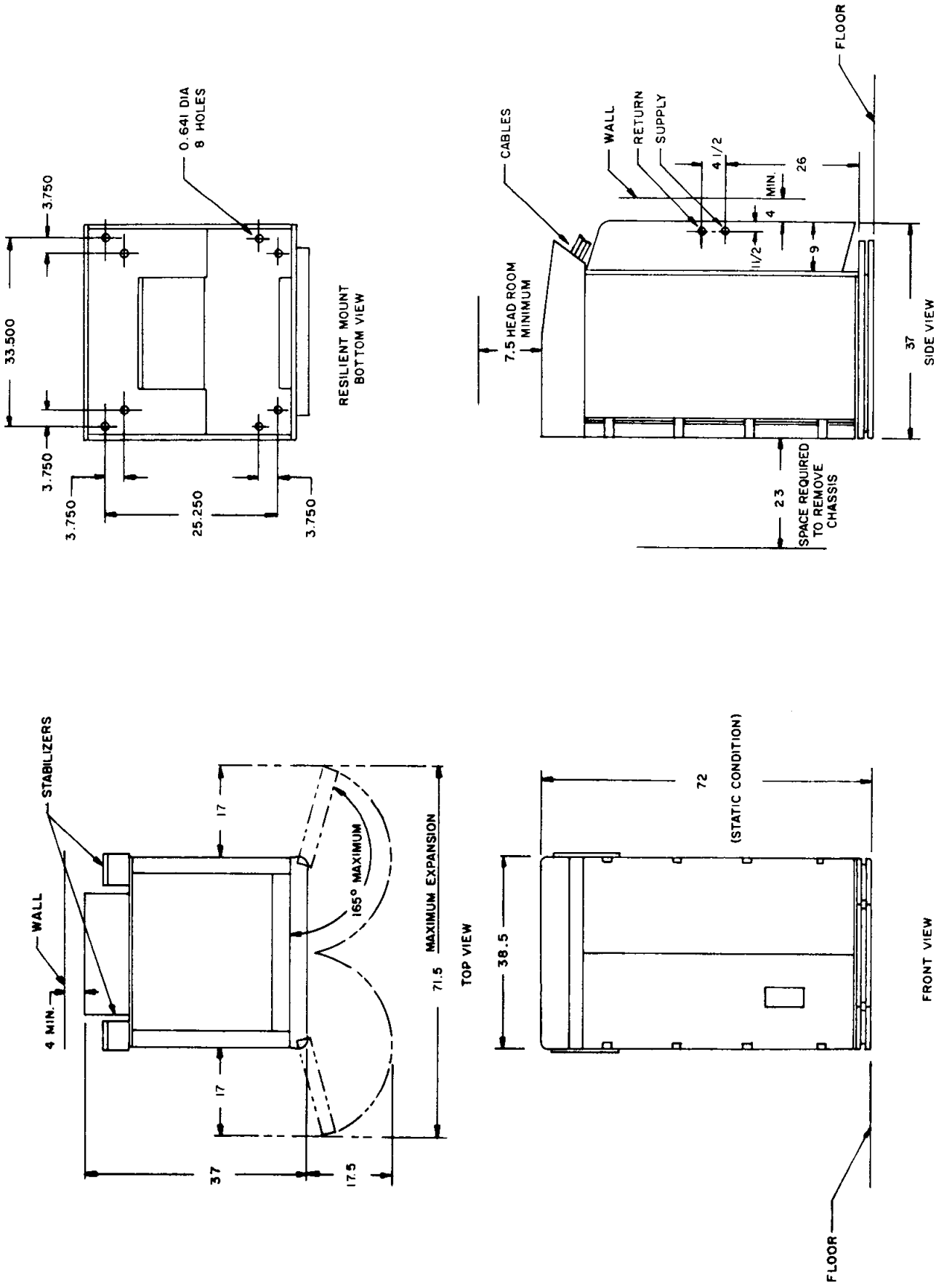


Figure 2-1. Computer Installation and Dimensional Diagram

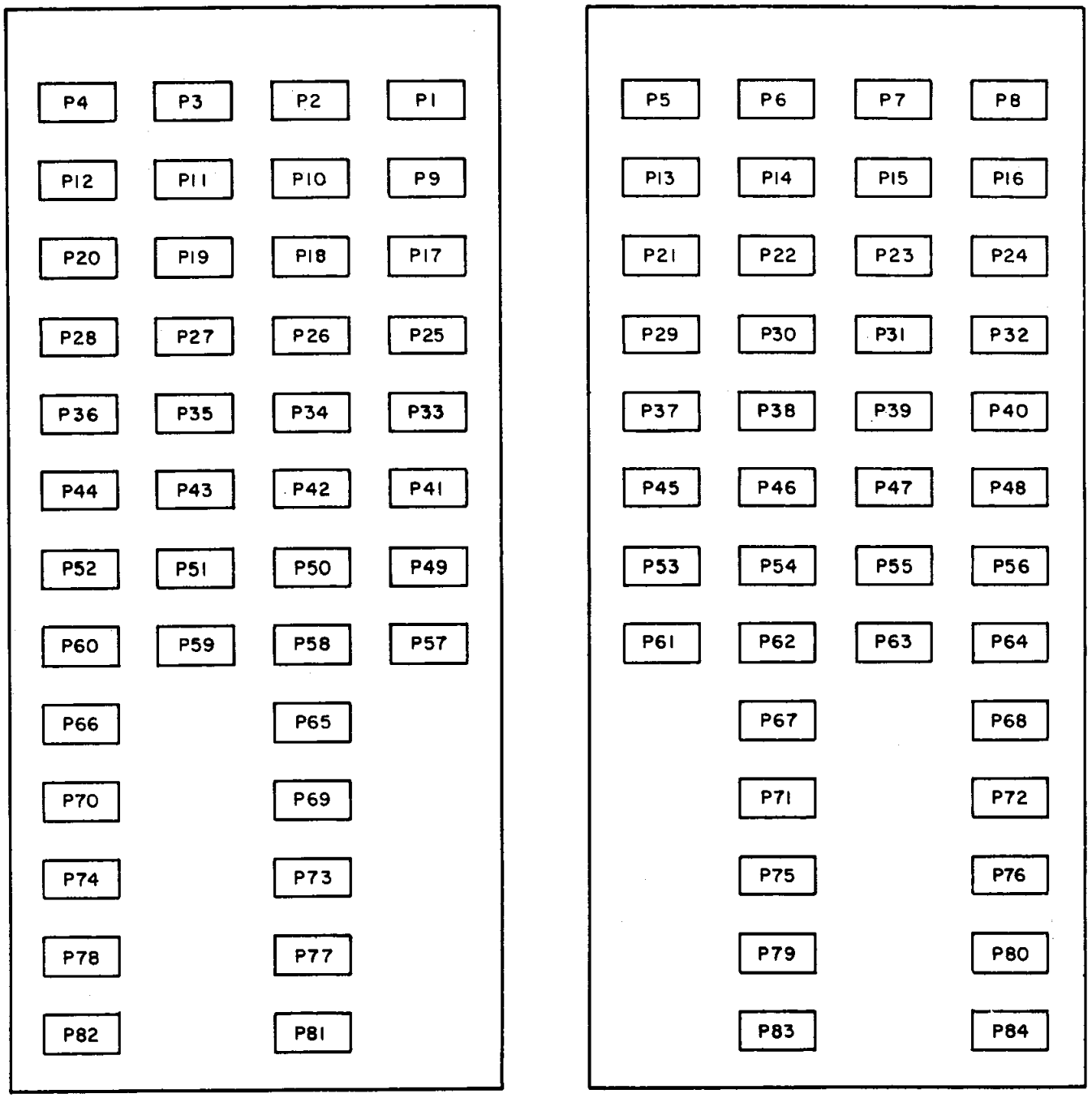


Figure 2-2. Interassembly Plug Rack, Diagrams of Plug Location

TABLE 2-1. CABINET INPUT/OUTPUT CONNECTORS

CABINET CONNECTOR		FUNCTION
INPUT DATA	OUTPUT DATA	
J1	J17	Channel 0
J2	J18	Channel 1
J3	J19	2
J4	J20	3
J5	J21	4
J6	J22	5
J7	J23	6
J8	J24	7
J9	J25	8 10
J10	J26	9 11
J11	J27	10 12
J12	J28	11 13
J13	J29	12 14
J14	J30	13 15
J15	J31	14 16
J16	J32	15 17
J53		Regulated 400 cps Power
J54		Unregulated 400 cps Power
J42		External Real-Time Clock

e. GROUNDING.

(1) GENERAL. - The method of grounding the computer assumes extreme importance for two reasons: first, the safety of operating and maintenance personnel; and, secondly, a DC system is used for input/output communications. When large distances separate various pieces of equipment, a difference of potential may exist between the ground points. Such a difference of ground potential has a direct effect on the equipment's ability to distinguish between a logical "1" or a logical "0" on the input/output lines.

(2) CABINET GROUND. - The computer cabinet is grounded to the system ground bus through a branch cable. The branch cable should be of minimum practical length. Its size is determined by its length and is dependent upon the requirements of the system. (See table 2-2.)

(3) INTERCABINET GROUND. - The ground bus, which has a very low impedance at low frequencies, has a relatively high impedance at the signal frequencies used. Because of its low inductance, the ground return wire of each signal-twisted pair will carry virtually all the signal return current. Recommended connections for ground return lines are shown in figure 2-6. Ideally, a twisted pair is carried from an output circuit in one cabinet to an input circuit in another cabinet. If necessary, a common ground return may be used exclusively on the output circuit end of the cable. This is required since the current supplied by an output circuit during transition from "zero" to "one" state, or vice versa, includes the line charging current, which is much greater than the DC steady state current received by the input circuit.

Figure 2-6 shows that the cable shield connects to pins 45 and 69 of the connector; these in turn are grounded at both ends to equipment cabinets. This arrangement allows the shield ground to be disconnected at one end if later desired. The aluminum braid armor is primarily for mechanical protection, serving no direct function in the grounding system, although some protection from radiation is realized. Circuit ground is shown connected to an internal bus which is isolated from cabinet ground. Provision is made for strapping circuit ground to cabinet ground at a point near the ground lug. This is an idealized case and provides maximum flexibility. Internal circuit considerations, however, determine the type of internal circuit grounding for specific cabinets.

f. CABLE ASSEMBLY. - This paragraph provides a general description of the service-type input/output cables suggested for use with the computer. The computer will operate with the cables (or equivalent) and the corresponding cable lengths and interface speeds specified in table 2-2 when the mating connectors [see paragraph 2-5f(3)] are connected to the cable ends.

TABLE 2-2. INPUT/OUTPUT CABLE CHARACTERISTICS

CABLE TYPE IN ACCORDANCE WITH -	MAXIMUM CABLE LENGTH (FEET)	
	HIGH SPEED INTERFACE	LOW SPEED INTERFACE
*DS 5177	200	300
*DS 5192	150	90
**SCD 7956068	200	300
**SCD 7956091	200	300

* Manufacturer's Design Specification
**Manufacturer's Specification Control Drawing

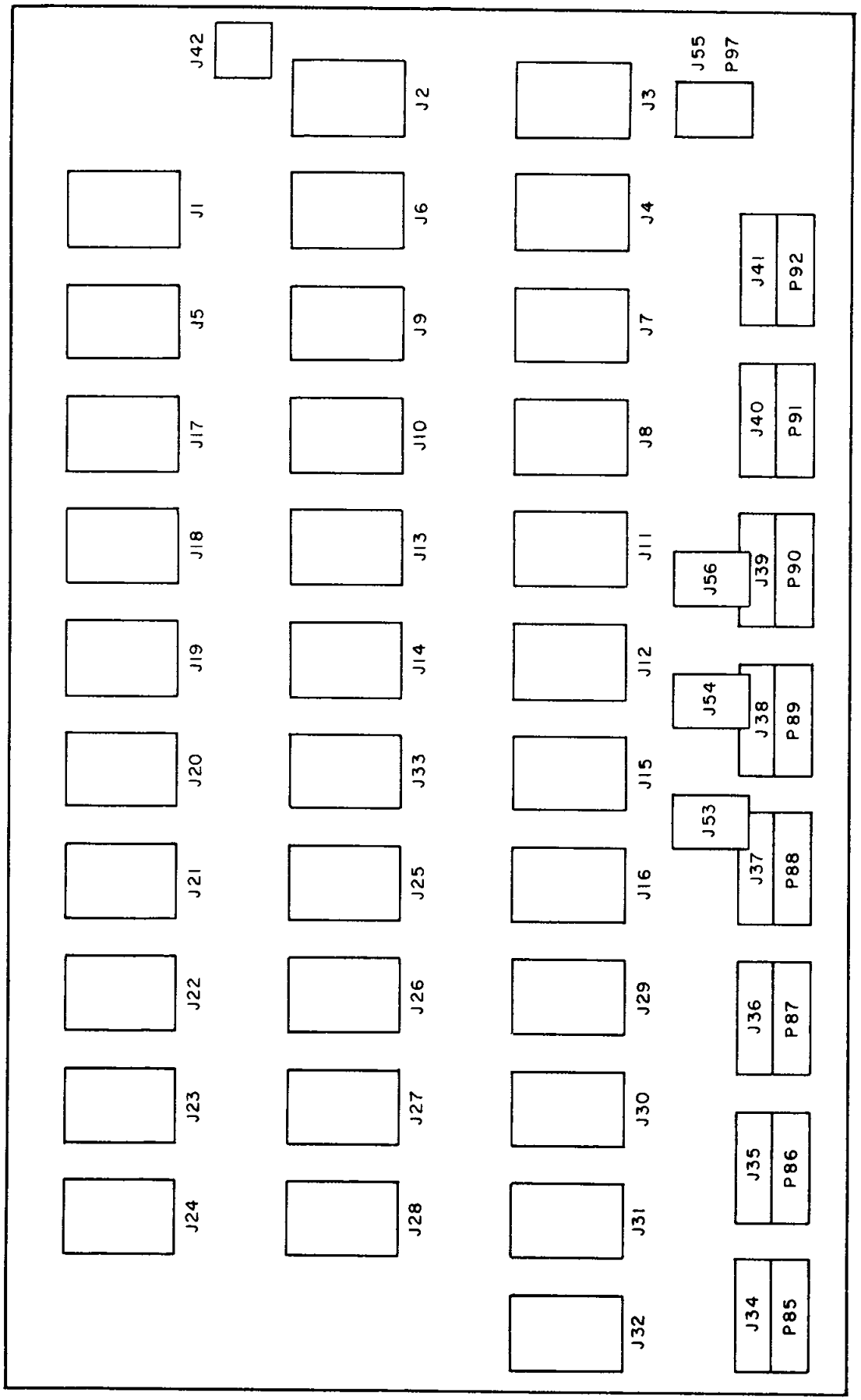
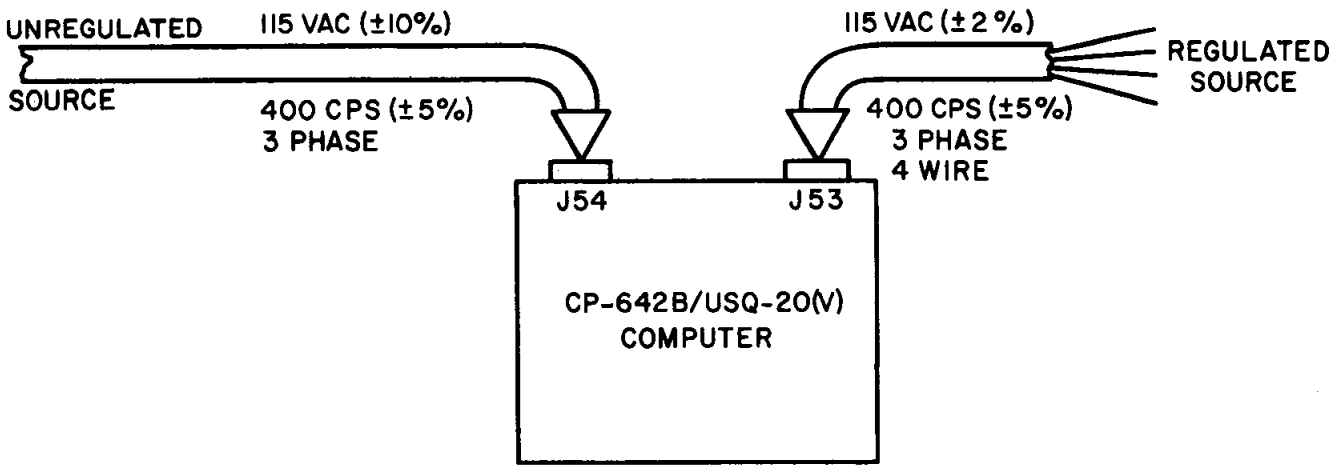


Figure 2-3. Input/Output Connectors (Top of Computer), Jack Orientation Diagram



NOTES

1. REGULATED SOURCE POWER CABLE - USE A FOUR-CONDUCTOR (14 AWG) CABLE AND A MS-3106A24-22SW CONNECTOR TO MATE WITH J53. CONNECT AS FOLLOWS:

FROM PWR SOURCE	TO CONN. PIN	FUNCTION
Ø1	A	115 VAC Ø1 400 CPS REG.
Ø2	B	115 VAC Ø2 400 CPS REG.
Ø3	C	115 VAC Ø3 400 CPS REG.
GND	D	SAFETY GROUND

2. UNREGULATED SOURCE POWER CABLE - USE A FOUR-CONDUCTOR (14 AWG) CABLE AND A MS-3106A24-22S CONNECTOR TO MATE WITH J54. CONNECT AS FOLLOWS:

FROM PWR SOURCE	TO CONN. PIN	FUNCTION
Ø1	A	115 VAC Ø1 400 CPS UNREG.
Ø2	B	115 VAC Ø2 400 CPS UNREG.
Ø3	C	115 VAC Ø3 400 CPS UNREG.
GND	CLAMP	SAFETY GROUND

Figure 2-4. Computer Power Connection Diagram

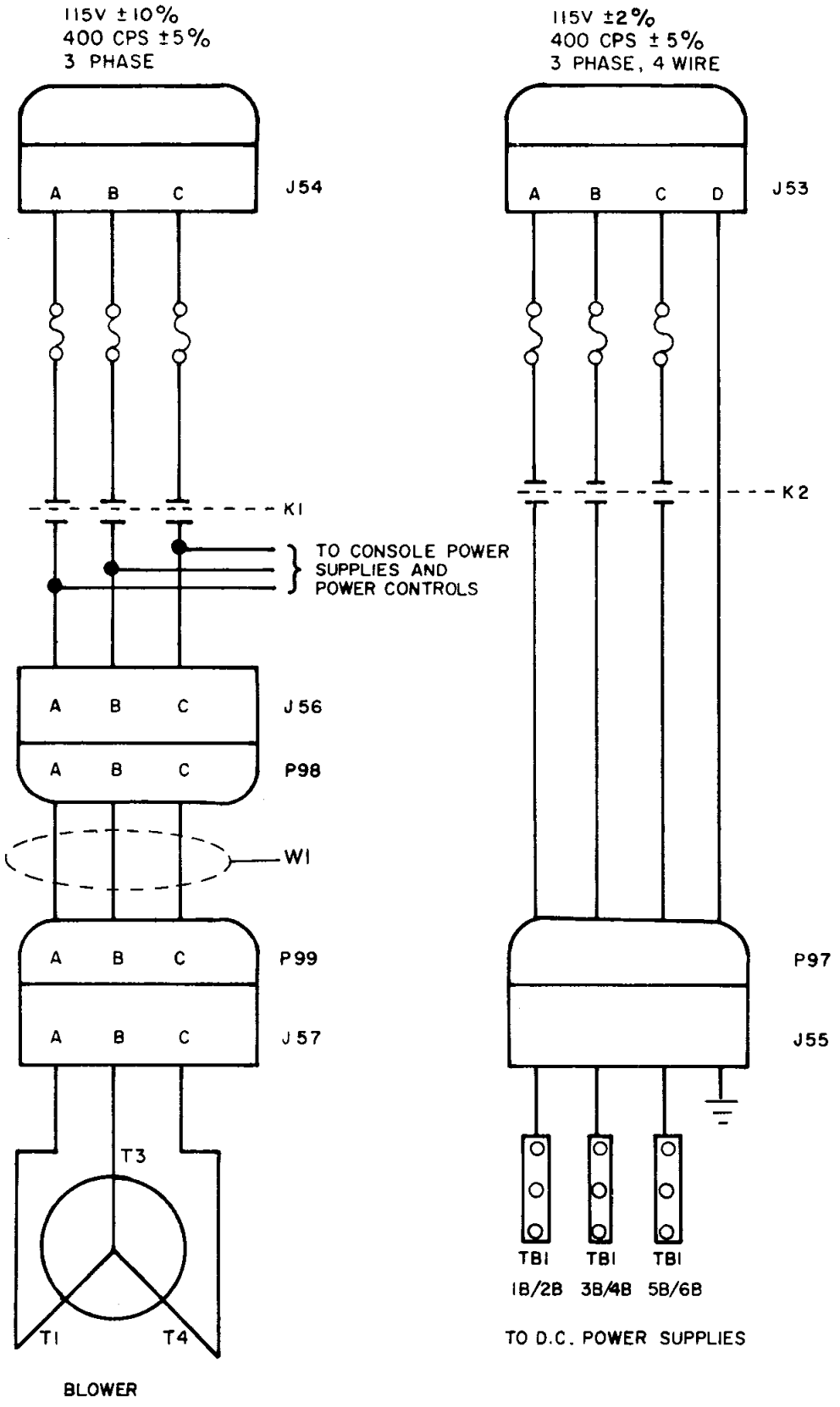


Figure 2-5. Computer Internal Power Distribution Schematic

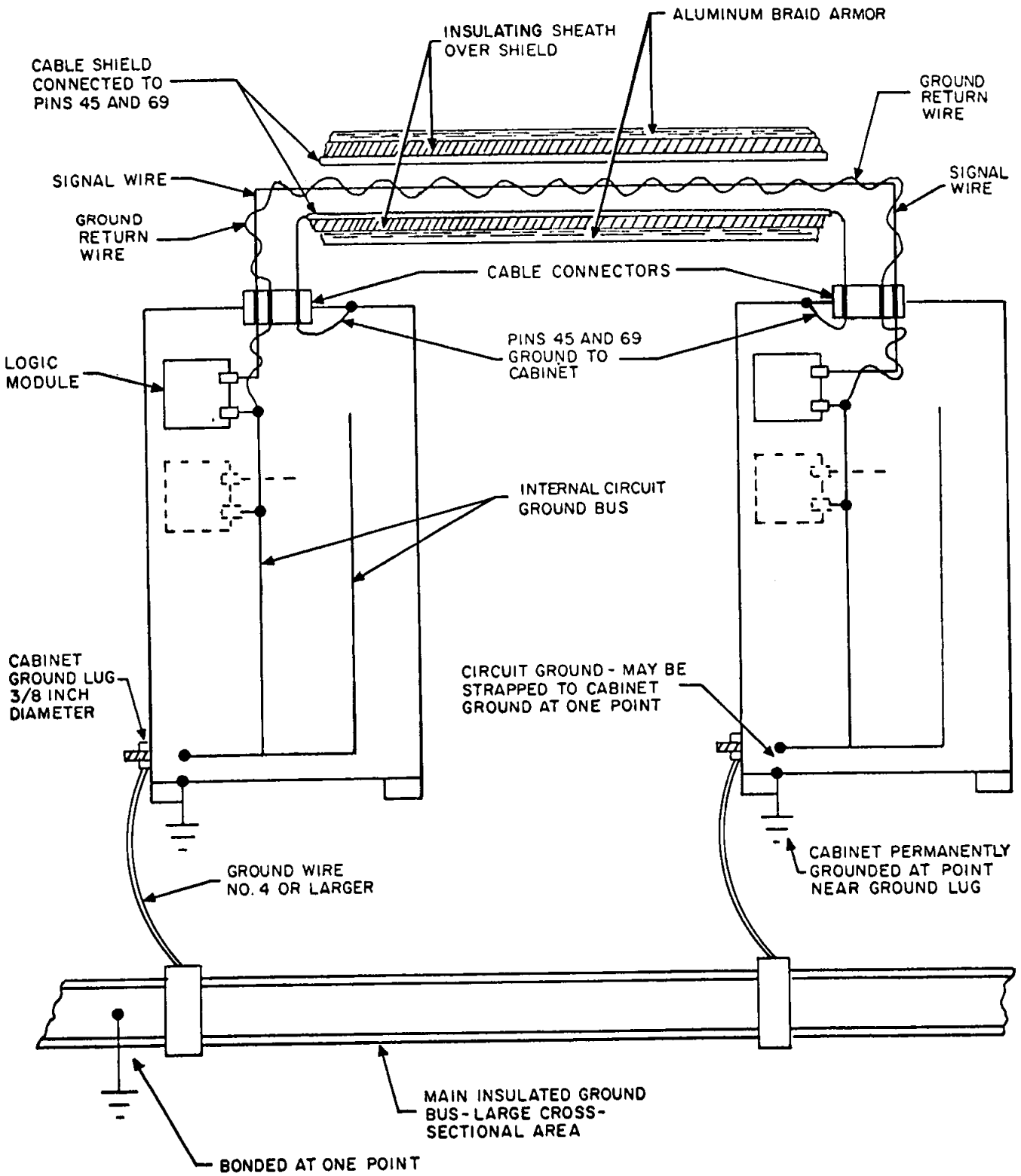


Figure 2-6. Computer Idealized Grounding Diagram

(1) CONSTRUCTION AND MATERIALS. - Individual conductors are composed of strands of tinned, soft-annealed copper. Primary insulation is extruded, high-molecular weight polyethylene with an average thickness among all conductors no less than 0.025 inch. This primary insulation is covered with a sheath of extruded nylon.

Conductor color coding conforms to Military Standard MIL-STD-104. Conductors are twisted into pairs, and cabled with the necessary fillers to form a core of circular cross-section. Cabling sequence conforms to Military Specification MIL-C-915 for twisted pair telephone cable.

A shield of braided, tinned, soft-annealed copper is applied over the binder to provide a minimum of 88 percent coverage. A black polyvinyl chloride sheath is extruded over the shield. The sheath is covered with an armor braid of aluminum alloy wire, containing not more than 0.2 percent copper and not less than 92.5 percent aluminum. The complete cable is painted, by the immersion process, with aluminum paint.

(2) CONNECTOR PIN AND WIRE ASSIGNMENTS. - Tables 2-3 and 2-4 provide the connector pin and wire assignments. The male connector pin arrangement is illustrated in figure 2-7, the female connector pin arrangement in figure 2-8. Comparing the tables with the pin arrangement of the figure reveals that, for a signal carried by a pin in a particular horizontal row, the associated ground return is carried by a pin in the row beneath and slightly to the right. For example, a control signal is assigned to pin 1; ground return for that signal is assigned to pin 11. Thus, a row of signal pins alternate with a row of ground return pins. Note the distinction between spare pins (wires connected but carrying no intelligence) and unused pins (no wires connected). The cable shield connects to pins 45 and 69, outside pins.

(3) CONNECTORS. - Cable connectors are of the rectangular, 90-pin type, featuring solderless taper-pin wire connections. Contacts are gold-plated, assuring continued low resistance over extended periods in corrosive atmospheres. Male connectors are cabinet-mounted; female connectors are attached to both ends of interconnecting cables. This arrangement minimizes the possibility of damage to male pins when installing or removing cable connectors. Exposed male pins present no safety hazard since only relatively low voltages are present.

(a) MALE CONNECTOR. - The physical configuration of the 90-pin male connector is illustrated in figure 2-7.

(b) FEMALE CONNECTOR. - The physical configuration of the 90-pin connector is illustrated in figure 2-8.

(4) CONNECTOR SHELLS. - The fittings on the female connector are termed a connector shell. Three connector shell variations allow for various angles of cable entry, zero degrees (cable axis parallel to connector pins), 90 degrees, and 135 degrees. Thus, flexibility is provided in connector panel positioning, relative to the direction of cable entry. Figure 2-9 illustrates a 90-degree connector shell.

A clamp, shown in the figure 2-9, functions as a locking device to secure the connector. The metal strap slips over a notch on the connector shell and a locking screw holds the connector in place.

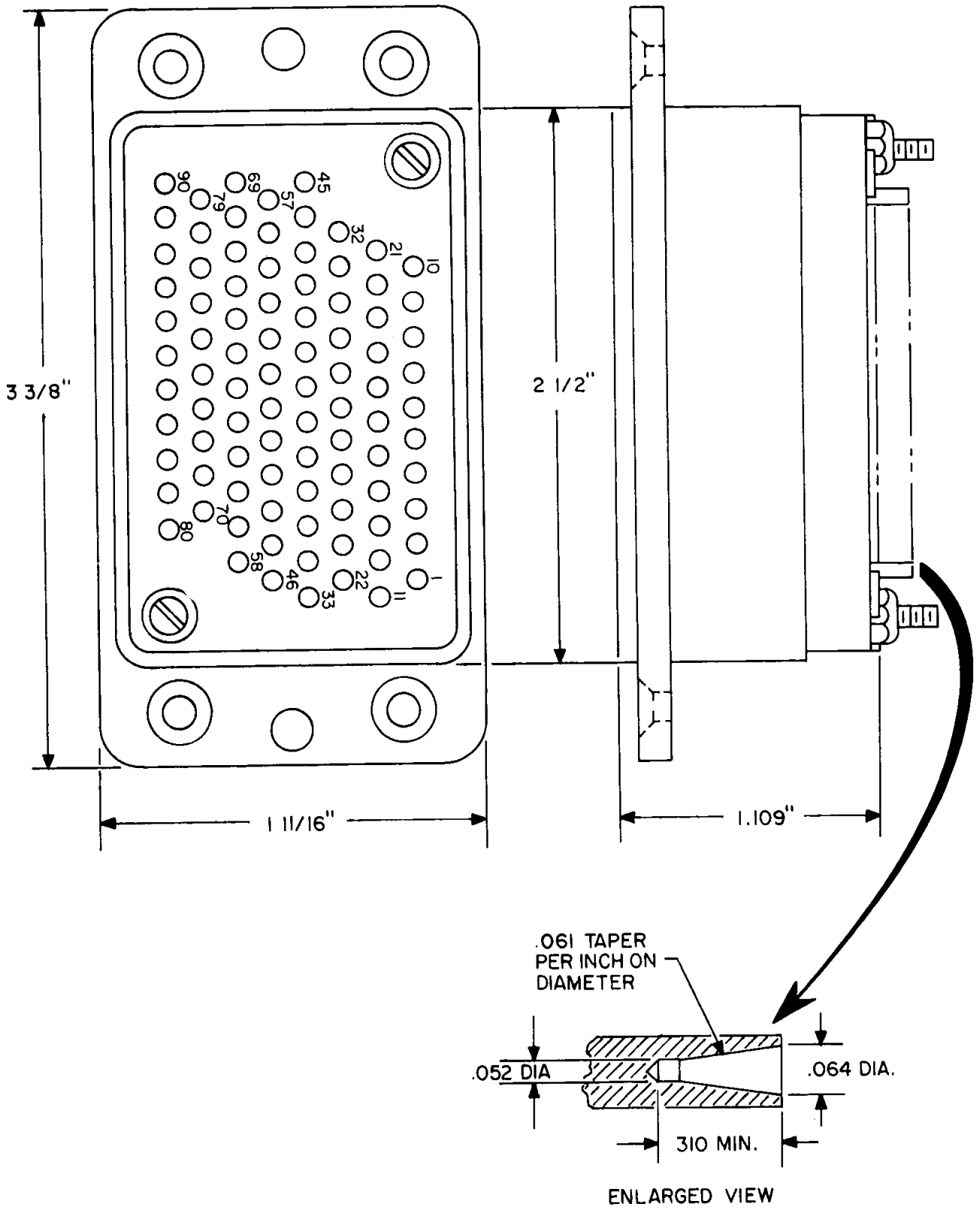


Figure 2-7. Male Connector, 90-Pin, Detailed View

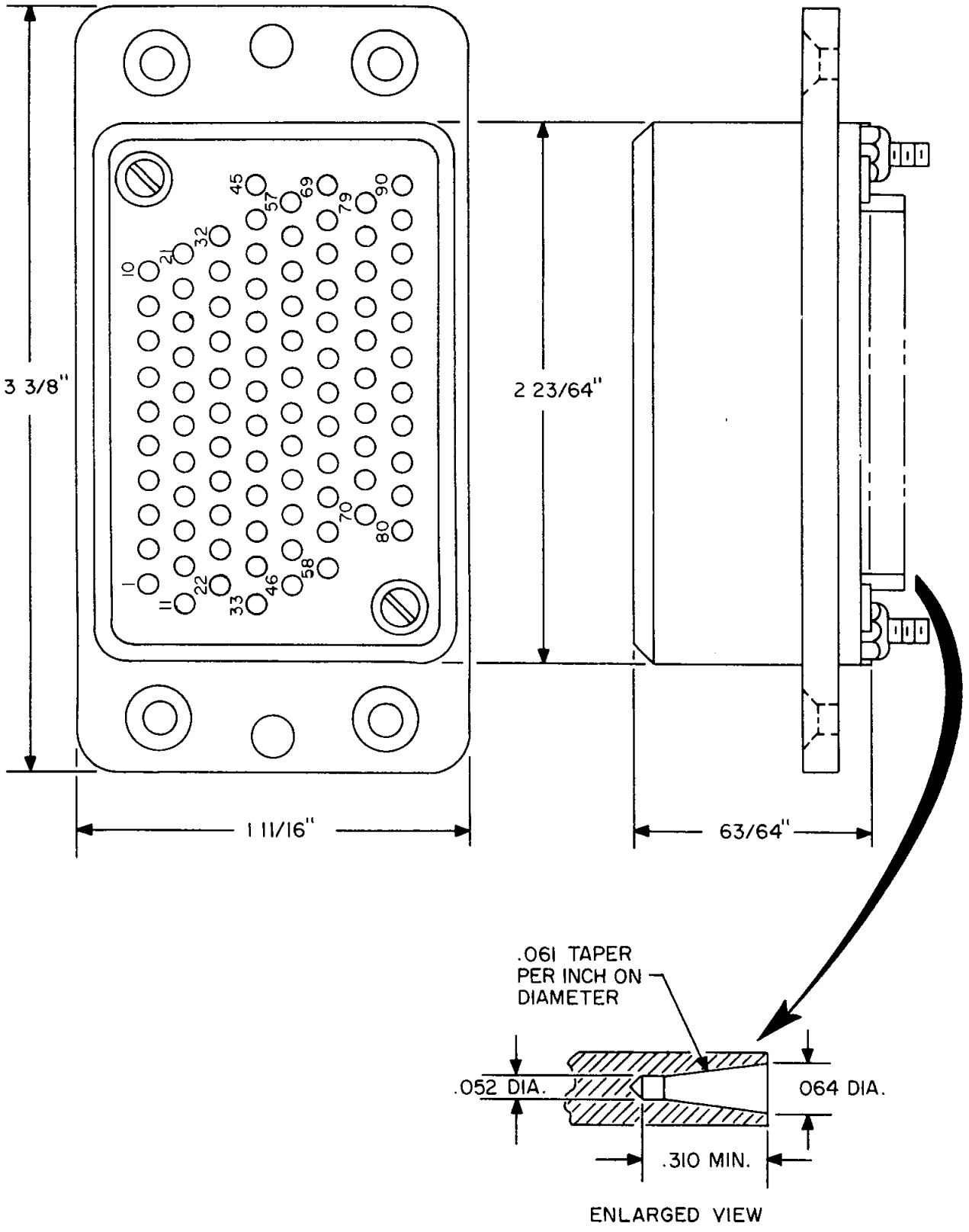


Figure 2-8. Female Connector, 90 Pin, Detailed View

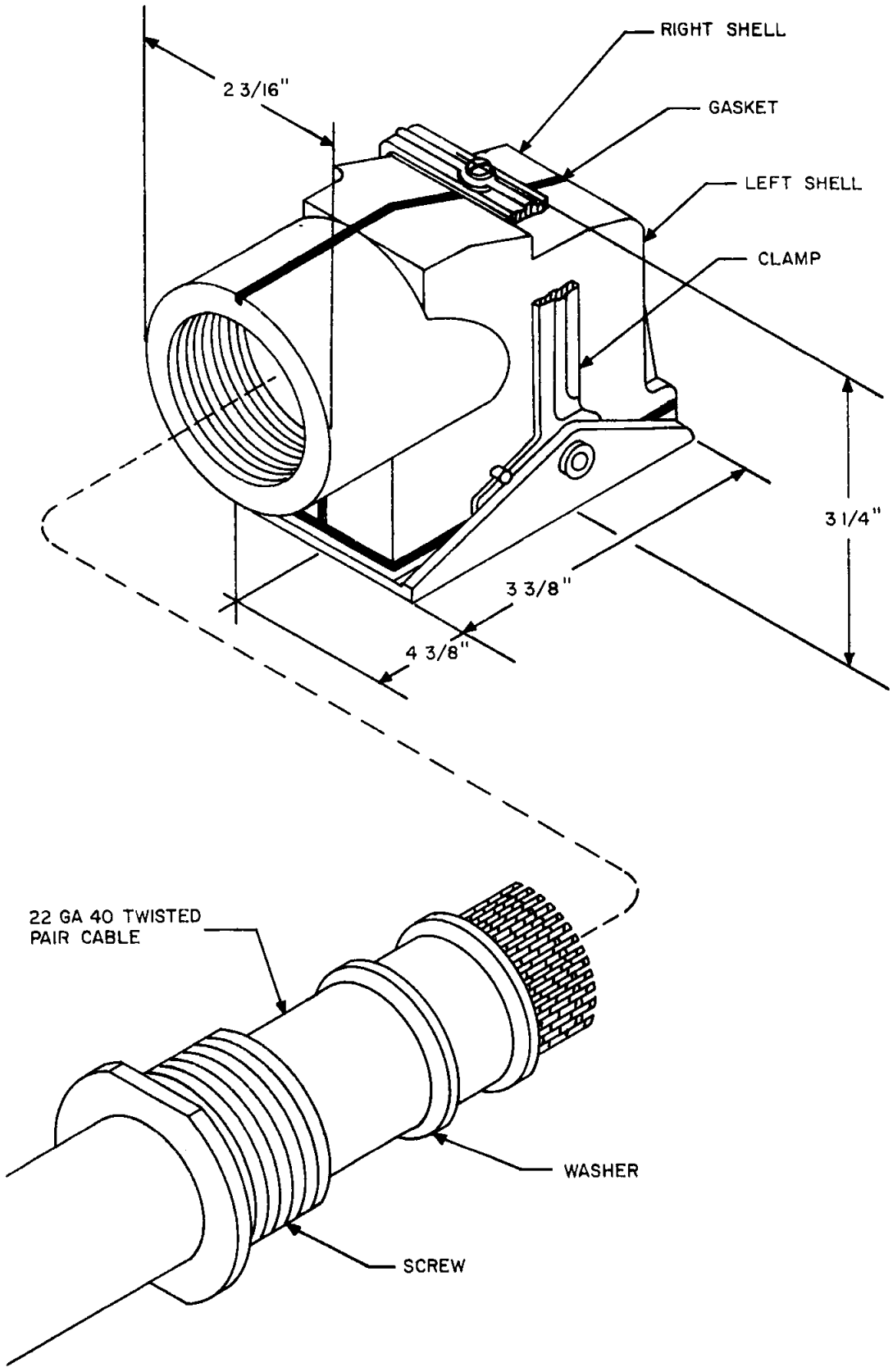


Figure 2-9. Connector Shell, 90-Degree, Detailed View

TABLE 2-3. CONNECTOR PIN ASSIGNMENTS

CONNECTOR PIN NUMBER	PERIPHERAL EQUIPMENT INPUT CHANNEL	INTERCOMPUTER INPUT CHANNEL	PERIPHERAL EQUIPMENT OUTPUT CHANNEL	INTERCOMPUTER OUTPUT CHANNEL
1	Input Data Request	Input Data Request	Output Data Acknowledge	Ready
2	Input Data Acknowledge	Input Data Acknowledge	Output Data Request	Resume
3	External Interrupt	External Interrupt	External Function	External Function
4	External Interrupt Request	External Interrupt Request	External Function Request	External Function Request
5	Spare	Spare	Spare	Spare
6	Spare	Spare	Spare	Spare
7	Spare	Spare	Spare	Spare
8	Spare	Spare	Spare	Spare
9	Data Bit 2^0	Data Bit 2^0	Data Bit 2^0	Data Bit 2^0
10	Data Bit 2^1	Data Bit 2^1	Data Bit 2^1	Data Bit 2^1
11	Input Data Request r	Input Data Request r	Output Data Acknowledge r	Ready r
12	Input Data Acknowledge r	Input Data Acknowledge r	Output Data Request r	Resume r
13	External Interrupt r	External Interrupt r	External Function r	External Function r
14	External Interrupt Request r	External Interrupt Request r	External Function Request r	External Function Request r
15	Spare	Spare	Spare	Spare
16	Spare	Spare	Spare	Spare
17	Spare	Spare	Spare	Spare
18	Spare	Spare	Spare	Spare
19	Data Bit 2^0r	Data Bit 2^0 r	Data Bit 2^0 r	Data Bit 2^0 r
20	Data Bit 2^1 r	Data Bit 2^1 r	Data Bit 2^1 r	Data Bit 2^1 r
21	Unused	Unused	Unused	Unused
22	Data Bit 2^2	Data Bit 2^2	Data Bit 2^2	Data Bit 2^2
23	Data Bit 2^3	Data Bit 2^3	Data Bit 2^3	Data Bit 2^3
24	Data Bit 2^4	Data Bit 2^4	Data Bit 2^4	Data Bit 2^4
25	Data Bit 2^5	Data Bit 2^5	Data Bit 2^5	Data Bit 2^5
26	Data bit 2^6	Data Bit 2^6	Data Bit 2^6	Data Bit 2^6

TABLE 2-3. CONNECTOR PIN ASSIGNMENTS (CONT.)

CONNECTOR PIN NUMBER	PERIPHERAL EQUIPMENT INPUT CHANNEL	INTERCOMPUTER INPUT CHANNEL	PERIPHERAL EQUIPMENT OUTPUT CHANNEL	INTERCOMPUTER OUTPUT CHANNEL
27	Data Bit 2^{7}	Data Bit 2^{7}	Data Bit 2^{7}	Data Bit 2^{7}
28	Data Bit 2^{8}	Data Bit 2^{8}	Data Bit 2^{8}	Data Bit 2^{8}
29	Data Bit 2^{9}	Data Bit 2^{9}	Data Bit 2^{9}	Data Bit 2^{9}
30	Data Bit 2^{10}	Data Bit 2^{10}	Data Bit 2^{10}	Data Bit 2^{10}
31	Data Bit 2^{11}	Data Bit 2^{11}	Data Bit 2^{11}	Data Bit 2^{11}
32	Data Bit 2^{12}	Data Bit 2^{12}	Data Bit 2^{12}	Data Bit 2^{12}
33	Data Bit 2^{2} r	Data Bit 2^{2} r	Data Bit 2^{2} r	Data Bit 2^{2} r
34	Data Bit 2^{3} r	Data Bit 2^{3} r	Data Bit 2^{3} r	Data Bit 2^{3} r
35	Data Bit 2^{4} r	Data Bit 2^{4} r	Data Bit 2^{4} r	Data Bit 2^{4} r
36	Data Bit 2^{5} r	Data Bit 2^{5} r	Data Bit 2^{5} r	Data Bit 2^{5} r
37	Data Bit 2^{6} r	Data Bit 2^{6} r	Data Bit 2^{6} r	Data Bit 2^{6} r
38	Data Bit 2^{7} r	Data Bit 2^{7} r	Data Bit 2^{7} r	Data Bit 2^{7} r
39	Data Bit 2^{8} r	Data Bit 2^{8} r	Data Bit 2^{8} r	Data Bit 2^{8} r
40	Data Bit 2^{9} r	Data Bit 2^{9} r	Data Bit 2^{9} r	Data Bit 2^{9} r
41	Data Bit 2^{10} r	Data Bit 2^{10} r	Data Bit 2^{10} r	Data Bit 2^{10} r
42	Data Bit 2^{11} r	Data Bit 2^{11} r	Data Bit 2^{11} r	Data Bit 2^{11} r
43	Data Bit 2^{12} r	Data Bit 2^{12} r	Data Bit 2^{12} r	Data Bit 2^{12} r
44	Unused	Unused	Unused	Unused
45	Cable Shield	Cable Shield	Cable Shield	Cable Shield
46	Unused	Unused	Unused	Unused
47	Data Bit 2^{13}	Data Bit 2^{13}	Data Bit 2^{13}	Data Bit 2^{13}
48	Data Bit 2^{14}	Data Bit 2^{14}	Data Bit 2^{14}	Data Bit 2^{14}
49	Data Bit 2^{15}	Data Bit 2^{15}	Data Bit 2^{15}	Data Bit 2^{15}
50	Data Bit 2^{16}	Data Bit 2^{16}	Data Bit 2^{16}	Data Bit 2^{16}
51	Data Bit 2^{17}	Data Bit 2^{17}	Data Bit 2^{17}	Data Bit 2^{17}
52	Data Bit 2^{18}	Data Bit 2^{18}	Data Bit 2^{18}	Data Bit 2^{18}
53	Data Bit 2^{19}	Data Bit 2^{19}	Data Bit 2^{19}	Data Bit 2^{19}
54	Data Bit 2^{20}	Data Bit 2^{20}	Data Bit 2^{20}	Data Bit 2^{20}
55	Data Bit 2^{21}	Data Bit 2^{21}	Data Bit 2^{21}	Data Bit 2^{21}
56	Data Bit 2^{22}	Data Bit 2^{22}	Data Bit 2^{22}	Data Bit 2^{22}
57	Data Bit 2^{23}	Data Bit 2^{23}	Data Bit 2^{23}	Data Bit 2^{23}
58	Data Bit 2^{13} r	Data Bit 2^{13} r	Data Bit 2^{13} r	Data Bit 2^{13} r

TABLE 2-3. CONNECTOR PIN ASSIGNMENTS (CONT.)

CONNECTOR PIN NUMBER	PERIPHERAL EQUIPMENT INPUT CHANNEL	INTERCOMPUTER INPUT CHANNEL	PERIPHERAL EQUIPMENT OUTPUT CHANNEL	INTERCOMPUTER OUTPUT CHANNEL
59	Data Bit 2^{14} r	Data Bit 2^{14} r	Data Bit 2^{14} r	Data Bit 2^{14} r
60	Data Bit 2^{15} r	Data Bit 2^{15} r	Data Bit 2^{15} r	Data Bit 2^{15} r
61	Data Bit 2^{16} r	Data Bit 2^{16} r	Data Bit 2^{16} r	Data Bit 2^{16} r
62	Data Bit 2^{17} r	Data Bit 2^{17} r	Data Bit 2^{17} r	Data Bit 2^{17} r
63	Data Bit 2^{18} r	Data Bit 2^{18} r	Data Bit 2^{18} r	Data Bit 2^{18} r
64	Data Bit 2^{19} r	Data Bit 2^{19} r	Data Bit 2^{19} r	Data Bit 2^{19} r
65	Data Bit 2^{20} r	Data Bit 2^{20} r	Data Bit 2^{20} r	Data Bit 2^{20} r
66	Data Bit 2^{21} r	Data Bit 2^{21} r	Data Bit 2^{21} r	Data Bit 2^{21} r
67	Data Bit 2^{22} r	Data Bit 2^{22} r	Data Bit 2^{22} r	Data Bit 2^{22} r
68	Data Bit 2^{23} r	Data Bit 2^{23} r	Data Bit 2^{23} r	Data Bit 2^{23} r
69	Cable Shield	Cable Shield	Cable Shield	Cable Shield
70	Data Bit 2^{24}	Data Bit 2^{24}	Data Bit 2^{24}	Data Bit 2^{24}
71	Data Bit 2^{25}	Data Bit 2^{25}	Data Bit 2^{25}	Data Bit 2^{25}
72	Data Bit 2^{26}	Data Bit 2^{26}	Data Bit 2^{26}	Data Bit 2^{26}
73	Data Bit 2^{27}	Data Bit 2^{27}	Data Bit 2^{27}	Data Bit 2^{27}
74	Data Bit 2^{28}	Data Bit 2^{28}	Data Bit 2^{28}	Data Bit 2^{28}
75	Data Bit 2^{29}	Data Bit 2^{29}	Data Bit 2^{29}	Data Bit 2^{29}
76	Spare	Spare	Spare	Spare
77	Spare	Spare	Spare	Spare
78	Unused	Unused	Unused	Unused
79	Unused	Unused	Unused	Unused
80	Data Bit 2^{24} r	Data Bit 2^{24} r	Data Bit 2^{24} r	Data Bit 2^{24} r
81	Data Bit 2^{25} r	Data Bit 2^{25} r	Data Bit 2^{25} r	Data Bit 2^{25} r
82	Data Bit 2^{26} r	Data Bit 2^{26} r	Data Bit 2^{26} r	Data Bit 2^{26} r
83	Data Bit 2^{27} r	Data Bit 2^{27} r	Data Bit 2^{27} r	Data Bit 2^{27} r
84	Data Bit 2^{28} r	Data Bit 2^{28} r	Data Bit 2^{28} r	Data Bit 2^{28} r
85	Data Bit 2^{29} r	Data Bit 2^{29} r	Data Bit 2^{29} r	Data Bit 2^{29} r
86	Spare	Spare	Spare	Spare
87	Spare	Spare	Spare	Spare
88	Unused	Unused	Unused	Unused
89	Unused	Unused	Unused	Unused
90	Unused	Unused	Unused	Unused

NOTE: "r" denotes ground return side of twisted pair.

TABLE 2-4. WIRE ASSIGNMENT FOR DIGITAL DATA CABLE

COLOR	PIN	PAIR
Black White	28 39	1
Black Red	49 60	2
Black Green	52 63	3
Black Orange	53 64	4
Black Blue	30 41	5
Black Brown	29 40	6
Black Gray	27 38	7
Black Yellow	50 61	8
Black Purple	51 62	9
Black Tan	54 65	10
Black Pink	55 66	11
White Red	56 67	12
White Green	31 42	13
White Orange	10 20	14
White Blue	9 19	15
White Brown	7 17	16
White Gray	5 15	17
White Yellow	25 36	18
White Purple	26 37	19
White Tan	24 35	20

TABLE 2-4. WIRE ASSIGNMENT FOR DIGITAL DATA CABLE (CONT.)

COLOR	PIN	PAIR
White Pink	23 34	21
Red Green	48 59	22
Red Orange	72 82	23
Red Blue	22 33	24
Red Brown	47 58	25
Red Gray	70 80	26
Red Yellow	71 81	27
Red Purple	73 83	28
Red Tan	74 84	29
Red Pink	75 85	30
Green Orange	76 86	31
Green Blue	77 87	32
Green Brown	57 68	33
Green Gray	32 43	34
Green Yellow	8 18	35
Green Purple	6 16	36
Green Tan	4 14	37
Green Pink	3 13	38
Orange Blue	2 12	39
Orange Brown	1 11	40

TABLE 2-4. WIRE ASSIGNMENT FOR DIGITAL DATA CABLE (CONT.)

COLOR	PIN	PAIR
UNUSED	21 44 46 78 79 88 89 90	
Cable Shield	45 69	

2-6. INSPECTION AND ADJUSTMENT.

a. PHYSICAL INSPECTION. - Make certain that all power is disconnected at the main power panel before making any inspection. Check the tightness of the bolts on the base of the computer to insure that the base is well secured. Remove the top cover and the side panels of the computer. Make a thorough visual check for loose or broken taper pins, broken or bared wires, and possible broken or damaged connectors. Be sure that all connectors are set firmly in place on every chassis because there is no interlock system.

The following checkout procedure is used to check the mechanical operation of the doors and the chassis.

(1) COMPUTER DOORS. - Release the five door locks by turning locks in a counterclockwise direction. Open the doors and check for binding, dents, condition of gaskets, etc. When the computer doors are open, they are held in either of two positions by a detent mechanism (locking mechanism) on the door guides. Once the doors are locked in position, the detent must be released manually. Do not attempt to force a door out of a detent position. When a chassis is to be removed from the computer, the doors must be locked in their outermost detent.

(2) CHASSIS. - The chassis are normally shipped installed in the computer. Each chassis should be removed and given a thorough visual inspection before power is applied. Check for any damage to jacks, printed circuit modules, or wiring.

Using the special wrench provided, turn, to OPEN, the plug assembly lock-release mechanism on each side of a chassis (see figure 2-10). Turn only far enough so that the assembly clears the side of the chassis. This disengages the plug assembly from the jacks on the side of the chassis. Manually release the chassis locks on each side of a chassis. Hook the two provided chassis pullers over the locks; the chassis should slide forward easily. When the chassis is stopped by the chassis stops, push the release rods on each side of the chassis and remove the chassis.

WARNING

Do not attempt to remove a chassis alone. Get assistance. Personnel disregarding this warning could very easily receive a serious injury.

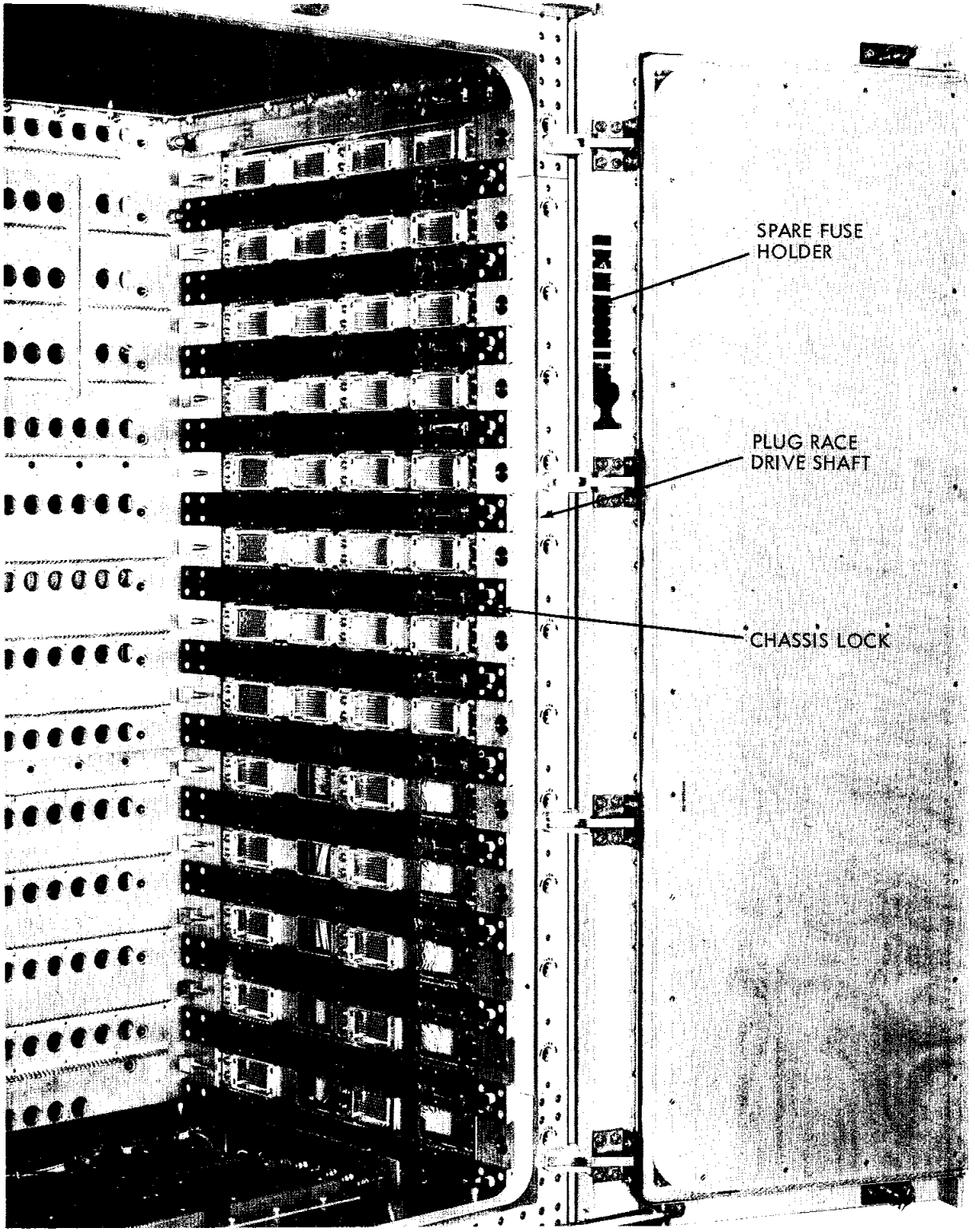


Figure 2-10. Chassis Removal Mechanisms

When all chassis have been removed, inspect the inside of the computer. Pay particular attention to the alignment of each plug assembly and the condition of the nylon rollers. Unless alignment is required, do not turn a plug assembly drive shaft when a chassis is removed from the computer.

CAUTION

Chassis A8 contains switches exposed on the front face of the chassis. Do not "stand" chassis A8 on its front edge.

If alignment of the plug assembly is necessary, remove the chassis. Turn the lock-release mechanism in the locking direction until the assembly is disengaged from the drive gears. Reposition the assembly and feed it evenly back into the drive gears while turning the mechanism in the RELEASE direction.

To install a chassis in the computer, reverse the removal procedure described in paragraph 2-6a(2).

b. INITIAL TURN ON PROCEDURE. - Energize the computer in the following manner:

STEP 1. Operate the BLOWER POWER ON-OFF switch up to ON (indicator glows green).

STEP 2. Operate the COMPUTER POWER ON-OFF switch up to ON (indicator glows green).

STEP 3. Press the PHASE STEP MODE and OP STEP MODE indicator-switches and then the MASTER CLEAR pushbutton.

The computer power on sequence is completed.

c. NONLOGIC CHECKS. - Perform all nonlogic checks listed in paragraph 5-4.

d. LOGIC CHECKS. - Perform all logic checks listed in paragraph 5-3.

e. ADJUSTMENTS. - Perform all adjustments as required in accordance with the adjustment and timing procedures in paragraph 6-2.b.

2-7. INTERFERENCE REDUCTION.

If properly installed no interference by or to other equipment is possible.

2-8. PREPARATION FOR RESHIPMENT.

To prepare the equipment for reshipment, perform the following steps.

STEP 1. Disconnect all cables attached to the equipment.

STEP 2. Ensure that all chassis are secured in the cabinet and the doors are shut.

STEP 3. Reverse the procedure given in paragraph 2-1. Include the applicable technical manuals with the computer and mark the box "TECHNICAL MANUALS INSIDE".

SECTION 3

OPERATOR'S SECTION

TABLE OF CONTENTS

TABLE OF CONTENTS (CONT.)

LIST OF ILLUSTRATIONS

LIST OF TABLES

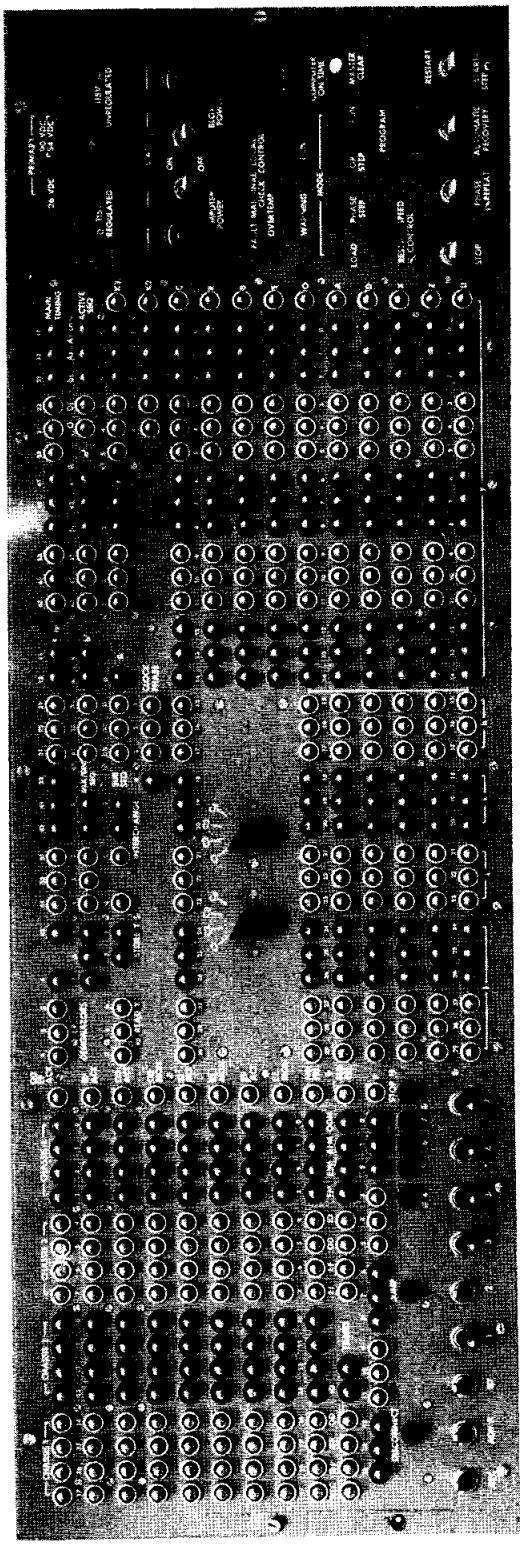


Figure 3-1. Computer Control Console

SECTION 3

OPERATOR'S SECTION

3-1. FUNCTIONAL OPERATION.

The CP-642B/USQ-20(V) computer is a program-controlled device designed to process data and disperse the resulting information. Once initiated, it performs all task requirements with a minimum of supervision or assistance.

a. OPERATING CONCEPT. - Since the computer is automatic, the operator need only have limited experience in its operation. After initial starting, the computer operates automatically until the program has completed the tasks assigned or until the computer is stopped through manual intervention.

b. OPERATING CHARACTERISTICS. - Prior to any operation of the computer and its program, preliminary selections must be made in a prescribed sequence. Before the computer is actually started, it must be powered and the desired mod of operation selected. Once defined by mode selection, the computer will run in that mode until otherwise directed by different selections.

3-2. PREPARATION FOR USE.

The computer, when installed, inspected, and adjusted according to procedures in Section 2, requires no further preparation for use.

3-3. OPERATING PROCEDURES.

Although the computer is basically automatic, there are provisions for manual control. There are switches which are used to affect the entire computer operation, control parts of the computer operations, provide certain jump or stop conditions, and govern speed of operation. There are also indicator-switches by which a single stage can be set and switches which clear an entire register. The various controls are covered in this manual by first describing the function performed and then the procedure the operator must follow for correct operation.

a. CONTROLS AND INDICATORS. - (See figure 3-1 and table 3-1.)

(1) BLOWER POWER ON/OFF. - Operating the BLOWER switch to the ON position energizes a contactor which provides power to the blowers and lights the green indicator adjacent to the switch.

(2) COMPUTER POWER ON/OFF. - Operating the COMPUTER POWER switch to the ON position energizes a contactor which provides power to the logic circuits and lights the green indicator. An interlocking device prevents computer primary power distribution when a temperature or blower fault condition occurs.

(3) MODE SWITCHES. - The computer is operative in one of several modes. The indicator-switch controls (LOAD MODE, PHASE STEP MODE, OP STEP MODE, and RUN MODE) allow the operator to select a specific computer operating mode. Pressing the LOAD MODE control locks out all interrupts, enables the AF sequence, and enables a jump to address 00540. Pressing the PHASE STEP MODE indicator-switch allows the computer to operate on a one-clock-phase-at-a-time basis. Pressing the OP

TABLE 3-1. SUMMARY OF OPERATOR CONTROL FUNCTIONS

* Type of Control
(M) Momentary
(L) Locked
Lit - Indicates Neon Indicator ON

CONTROL	TYPE			OPERATION	POSITION	ACTION	REFERENCE PARAGRAPH 3-3a
	LEVER SWITCH	PUSHBUTTON SWITCH	INDICATOR				
BLOWER POWER ON-OFF	*		Green	Operated	ON (M)	Applies 400 cycle power to centrifugal blower	(1)
				Operated	OFF(M)	Removes 400 cycle power from blower and logic	
COMPUTER POWER ON-OFF	*		Green	Operated	ON (M)	Applies 400 cycle power to logic circuits	(2)
				Operated	OFF(M)	Removes 400 cycle power from logic circuits	
LOAD MODE		*	Amber	Pressed		Permits Load mode of operation when computer starts	(3)
OP STEP MODE		*	Amber	Pressed		Permits Op Step mode of operation when computer starts	(3)
PHASE STEP MODE		*	Amber	Pressed		Permits Phase Step mode of operation when computer starts	(3)
RUN MODE		*	Amber	Pressed		Permits Run mode of operation when computer starts	(3)
START-STEP/ RESTART	*			Operated	Down(M) (START-STEP)	Starts computer at normal high-speed in Run mode	(4)
						Generates one clock phase in Phase Step mode	(4)
						Allows one instruction execution in Op Step mode	(4)
						Initiates Load mode if selected	(4)

TABLE 3-1. SUMMARY OF OPERATOR CONTROL FUNCTIONS (CONT.)

CONTROL	TYPE			OPERATION	POSITION	ACTION	REFERENCE PARAGRAPH 3-3a
	LEVER SWITCH	PUSHBUTTON SWITCH	INDICATOR				
START-STEP/ RESTART	*			Operated	Up (L) (RESTART)	Generates clock phases at a repetition rate controlled by the low-speed oscillator in Phase Step mode.	(4)
						Allows instruction initiation at a time controlled by the low-speed oscillator in Op Step mode.	(4)
						Provides self-recovery from program stops if in Run mode	(4)
RESTART SPEED CONTROL	Rotary			Rotated	Right	Increases frequency of low-speed oscillator	(5)
					Left	Decreases frequency of low-speed oscillator	(5)
STOP	*			Operated	Down(M)	All computer operation stops	(6)
PHASE REPEAT	*			Operated	Up (L)	Forces repetition (at high-speed) of one selected clock phase if Phase Step mode is active	(7)
AUTOMATIC RECOVERY	*			Operated	Up (L)	Directs computer after program fault condition to jump to address 00540 (Bootstrap)	(8)
						Locks out all interrupts	
				Operated	Neutral (L)	Directs computer after program fault condition to jump to address 00000. Locks out all interrupts.	(8)
					Down	Not Used	

TABLE 3-1. SUMMARY OF OPERATOR CONTROL FUNCTIONS (CONT.)

CONTROL	TYPE			OPERATION	POSITION	ACTION	REFERENCE PARAGRAPH 3-3a
	LEVER SWITCH	PUSHBUTTON SWITCH	INDICATOR				
MASTER CLEAR				Pressed		Clears all computer circuits Sets Run mode	(9)
RUN			Green	Pressed		Indicates high-speed operations Clears Run flip-flop	(10)
PROGRAM I and II		* (2)	Green	Pressed		Selects the bootstrap program to be entered in Load mode or indicates the bootstrap program entered during program fault	(11) (8)
FAULT			Red			Indicates program fault condition	(13)
MARGINAL CHECK			Red			Indicates the computer is in a memory margin check condition	(14)
LOCAL CONTROL			Red			Indicates computer is under control of its own operating console	(15)
CLOCK PHASE		* (4)		Pressed		Enables selected (1,2,3, or 4) clock phase to be repeated during Phase Repeat mode.	(16)
OVERTEMP WARNING		*	Lit	Pressed		Indicates a particular clock phase is being issued Indicates excessive temperature 46°C (115°F) Silences overtemperature alarm horn	(17)
COMPUTER ON TIME			Meter			Records the time computer logic power is applied	(18)

TABLE 3-1. SUMMARY OF OPERATOR CONTROL FUNCTIONS (CONT.)

CONTROL	TYPE			OPERATION	POSITION	ACTION	REFERENCE PARAGRAPH 3-3a
	LEVER SWITCH	PUSHBUTTON SWITCH	INDICATOR				
C1, 2, 3, and 4	Rotary (2)			Rotate	(4)	Allows the contents of the selected C register to be displayed on the indicators	(19)
DISCONNECT			Red			Indicates one or more of the disconnect switches have been activated	(20)
RTC	*			Operated	Up (L)	Inhibits increment of real-time clock register	(20)
ADV P	*			Operated	Up (L)	Inhibits the incrementing of the P register	(20)
B7	*			Operated	Up (L)	Inhibits the decrementing of the B7 register in Repeat mode	(20)
JUMP			Red			Indicates one or more of the JUMP switches have been activated	(21)
1	*			Operated	Up (L)	Permits jump to specified address for f = 61 or 65, j = 1	(21)
2	*			Operated	Up (L)	Permits jump to specified address for f = 61 or 65, j = 2	(21)
3	*			Operated	Up (L)	Permits jump to specified address for f = 61 or 65, j = 3	(21)
STOP (SELECTIVE)			Red			Indicates one or more of the STOP switches have been selected	(22)

TABLE 3-1. SUMMARY OF OPERATOR CONTROL FUNCTIONS (CONT.)

CONTROL	TYPE			OPERATION	POSITION	ACTION	REFERENCE PARAGRAPH 3-3a
	LEVER SWITCH	PUSHBUTTON SWITCH	INDICATOR				
4			Red			Indicates an unconditional computer program stop, f = 61 or 65, j = 4	(22)
5	*		Red	Operated	Up (L)	Permits jump to specified address and stop computation, f = 61 or 65, j = 5	(22)
6	*		Red	Operated	Up (L)	Permits jump to specified address and a stop computation, f = 61 or 65, j = 6	(22)
7	*		Red	Operated	Up (L)	Permits jump to specified address and a stop computation, f = 61 or 65, j = 7	(22)
BIT INDICATORS (INDIVIDUAL)		*	Lit	Press		Sets the stage of individual bit selected	(23)
CLEAR (INDIVIDUAL)		*		Press		Clears associated register or designator	(23)

STEP MODE indicator-switch allows the computer to execute one instruction at a time. Pressing the RUN MODE indicator-switch allows the computer to operate at normal high-speed. When pressed, each amber MODE indicator switch glows.

(4) START-STEP/RESTART. - Once a mode selection has been made, the computer is started by operating the START-STEP/RESTART switch. The function of the computer is dependent on the mode selected and the initiation of the START-STEP/RESTART switch. Table 3-2 summarizes the action performed by the various selections.

(5) RESTART SPEED CONTROL. - Operating the RESTART SPEED CONTROL varies the frequency of the low-speed oscillator. Its functions are noted in table 3-1. The low-speed oscillator frequency varies between two and 200 cycles per second.

(6) STOP. - The STOP switch, when operated, disables the high-speed operation of the computer. It extinguishes the RUN indicator [see paragraph 3-3a(10)] and discontinues the operations of the run circuits.

(7) PHASE REPEAT. - The operation of the PHASE REPEAT switch forces the repetition of the selected clock phase [see paragraph 3-3a(16)] at a high-speed operation rate. The PHASE REPEAT switch is active only if the phase step mode is active.

(8) AUTOMATIC RECOVERY. - The AUTOMATIC RECOVERY switch, when selected, gives direction to the computer's activity after a program fault condition arises. Positioning the switch in the up position causes all interrupts to be locked out and initiates a jump to address 00540 (if a program fault occurs) and the automatic execution of the Bootstrap program. With the switch in the center position, a jump to 00000 and action appropriate to the program at that address occur after a program fault. For additional information, refer to paragraphs 3-3a(12) and 3-10.

(9) MASTER CLEAR. - Pressing the MASTER CLEAR pushbutton clears all registers and timing sequences and results in the setting of the RUN MODE indicator and its associated circuitry (if the computer is not in high-speed operation). Operating the pushbutton during high-speed operations clears only the FAULT indicator.

(10) RUN. - The green RUN indicator-switch glows when the computer is in high-speed operation. Operating the STOP switch during high-speed running disables the operation and extinguishes the RUN indicator.

(11) PROGRAM I AND II. - The operation of the PROGRAM I or II indicator-switch indicates the Bootstrap program (see paragraph 3-10) that will be performed when the program is manually initiated under load mode. It also indicates the bootstrap utilized when the program is referenced by means of the AUTOMATIC RECOVERY switch [see paragraph 3-3a(8)]. The corresponding green PROGRAM I or II indicator glows when selected. For additional information refer to paragraph 3-3a(12).

(12) PROGRAM SELECTOR SWITCH (PROGRAM I/II). - The program selector switch provides manual selection of the program that is referenced when in the automatic recovery mode. The switch is located on logic module J35A on chassis A5. Refer also to paragraphs 3-3a(8) and (11).

(13) FAULT. - The red FAULT indicator glows whenever the computer encounters a program fault condition. Function codes 00 and 77 are fault conditions which, if executed, cause a fault interrupt within the computer. The FAULT indicator glows during this condition and can be extinguished only by operating the MASTER CLEAR pushbutton.

TABLE 3-2. SUMMARY OF FUNCTIONS OF THE START-STEP/RESTART SWITCH

SELECTION	MODE	COMPUTER ACTION
START-STEP	RUN	Initiates high-speed operation.
RESTART	RUN	Allows self-recovery from program stops.
START-STEP	OP STEP	Allows the execution of one instruction each time the switch is operated.
RESTART	OP STEP	Allows initiation of instructions at a repetition rate controlled by the RESTART SPEED CONTROL and low-speed oscillator.
START-STEP	PHASE STEP	Generates one clock phase each time the switch is operated.
RESTART	PHASE STEP	Generates clock phases at a rate controlled by the RESTART SPEED CONTROL and low-speed oscillator.
START-STEP	LOAD	Initiates high-speed operation starting at address 00540 (executes bootstrap).
RESTART	LOAD	Not used.

(14) MARGINAL CHECK. - The amber MARGINAL CHECK indicator glows when a memory chassis is in the marginal check condition.

(15) LOCAL CONTROL. - When the computer is under the control of its own operating console the amber LOCAL CONTROL indicator glows.

(16) CLOCK PHASE. - The CLOCK PHASE indicator-switch indicates generation of the corresponding Master Clock phase. Pressing the switches in conjunction with phase repeat mode [see paragraph 3-3a(7)] allows the repetition of the selected clock phase at a high-speed rate of operation.

(17) OVERTEMP WARNING. - The computer's temperature sensing circuits cause the alarm horn to sound and the red OVERTEMP WARNING indicator-switch to glow whenever the internal cabinet temperature exceeds 46°C (115°F). Automatic shutdown occurs at 60°C (140°F). Pressing the OVERTEMP WARNING indicator-switch silences the alarm horn.

(18) COMPUTER ON TIME. - The COMPUTER ON TIME meter cumulatively records the time that power is on for distribution to the computer logic circuits. The meter ranges from 0 to 9999.9 hours and cannot be reset.

(19) C1 - C2 - C3 - C4. - The two C register rotary switches allow switching between the four C register and the C register indicator-switches. Operating the

switches to the appropriate setting allows the selected register to be displayed. The setting of a specific stage (or stages) can then be accomplished if desired. The left switch permits the upper 15 bits to enter the indicator display and the right switch allows the lower 15 bits to enter the display.

(20) DISCONNECT. - The operation of the disconnect circuit involves an indicator and three switches. Selecting (operating to the up position) any of the three switches (RTC, ADV P, or B7) causes the red DISCONNECT indicator to glow. Selecting the B7 switch inhibits the decrementing of the B7 register during the repeat mode of operation. Selecting the ADV P switch inhibits the incrementing of the P register. Selecting the RTC switch inhibits the incrementing of the real-time clock register.

(21) JUMP (SELECTIVE). - The red JUMP indicator glows when any or all of the three Selective Jump switches (1, 2, or 3) are selected (operated to the up position). Selecting the switches allows manual selection or omission of predetermined program sections in conjunction with the 61 or 65 instructions.

(22) STOP (SELECTIVE). - The red STOP indicator glows when any or all of the Selective Stop switches (5, 6, or 7) are selected (operated to the up position). Selecting the switches allows program monitoring when used in conjunction with appropriate 61 and 65 instructions. The red STOP 5, 6, or 7 indicators glow when a corresponding switch is selected, and another indicator glows when the program stop occurs.

(23) BIT INDICATORS. - The remaining arrays of indicator-switches and clear switches are associated with the registers, designators, and sequences of the computer. Pressing any register indicator-switch sets that stage and lights the associated neon indicator. The entire register is cleared by operating the associated clear switch. The operation of the designator and sequence indicator-switches and the clear switches is identical to those of the various registers.

(24) OPERATOR CONTROL FUNCTION. - All controls, indicators, indicator-switches, and switches have prescribed functions to follow to obtain correct indications and maintain control over the computer.

b. TURN ON PROCEDURE. - To supply power to the computer operate the BLOWER POWER switch to the ON position. Power from the external source energizes a contactor which provides voltage to the centrifugal blower. This selection lights the green indicator adjacent to the switch. Protection against short circuits is provided by 15 ampere fuses. Once blower power is supplied, operate the COMPUTER POWER switch to the ON position. Power from the external source energizes a contactor which applies power to the computer logic circuits. This selection lights the green indicator adjacent to the switch. The computer power also has short circuit protection provided by 15 ampere fuses.

Releasing the BLOWER POWER or the COMPUTER POWER switch allows the switch to return to the neutral position. During the operation of the computer, a temperature or blower fault will remove regulated power. A temperature or blower fault during the initial turn on procedure prevents the application of regulated power to the computer.

c. TURN OFF PROCEDURE. - To remove power from the computer, operate the BLOWER POWER switch to the OFF position. This removes power from the interlocking relays. The relays open and remove all power from the computer.

d. MODES OF OPERATION. - To initiate a mode of operation for the computer, the prescribed routine must be followed.

To operate the computer in run mode, following power application, press the PHASE STEP MODE, OP STEP MODE and the MASTER CLEAR switches in order. The computer is cleared, and the amber RUN MODE indicator-switch glows, indicating that the run mode is enabled. If a starting address other than 00000 is desired, it must be set in the P register. Operate the START-STEP/RESTART switch to the START-STEP position, and the computer starts.

To initiate the run mode after another mode (op step, phase step, or load) has been selected, press the RUN MODE indicator-switch. This allows the computer to initiate high-speed operation when started and permits the execution of the former program, starting from the address at which it stopped, or a new program, by setting a different address in P. Operate the START-STEP switch down and the computer starts. This lights the green RUN indicator.

To operate the computer in the op step (Operation Step) mode, press the MASTER CLEAR switch and the OP STEP indicator-switch. Operating the START-STEP/RESTART switch down initiates the execution of one instruction. Operating the START-STEP/RESTART switch up (locked) causes the computer to initiate the instructions under control of the restart speed control circuits.

To operate the computer in the phase step mode and accomplish the issuance of one clock phase pulse, press the MASTER CLEAR switch, and the PHASE STEP MODE indicator-switch. Operate the START-STEP/RESTART switch to the START-STEP position, and computer operation is initiated. The computer then issues progressively (1, 2, 3, and 4) one clock phase each time the START-STEP/RESTART switch is operated to the START-STEP position. To generate progressive clock phases at a rate controlled by the restart speed control circuits, operate the START-STEP/RESTART switch up to RESTART (locked) position.

To operate the computer in the load mode, press the MASTER CLEAR and LOAD MODE switches, and operate the START-STEP/RESTART switch down to the START-STEP position. The computer automatically starts the program at address 00540 (bootstrap). Immediately upon starting, the LOAD MODE indicator-switch will extinguish and the green RUN and amber RUN MODE indicators will glow, signifying high-speed operation.

e. MANUAL FUNCTIONS. - The following procedures for manually reading and writing can be used in performing operations used in testing and debugging programs and in maintenance of the computer. In the following paragraphs, the Q register is utilized. Similar operations are accomplished using the A register and related instruction references.

(1) MANUAL READING. - To transfer information from one memory address location to the Q register, master clear the computer; press the Af En (Af Enable) indicator-switch to allow the generation of the proper command signals when the computer is started; press the appropriate indicator-switches in the U register for enter Q instruction (f = 10, j = 0, k = 3 and b = 0). The y portion of the U register is loaded with the memory address from which the information is to be extracted. (See figure 3-2a.) Once the U register is properly set, press the OP STEP MODE indicator-switch. Operate the START-STEP/RESTART switch to the START-STEP position. The computer operates, the word from memory is located in the Q register, and the computer stops.

a.

001 000	000	011	000	15 BITS (DESIRED ADDRESS)
f	j	k	b	y

b.

001 100	000	011	000	15 BITS (DESIRED ADDRESS)
f	j	k	b	y

Figure 3-2. U Register Bits for Manual Reading and Writing

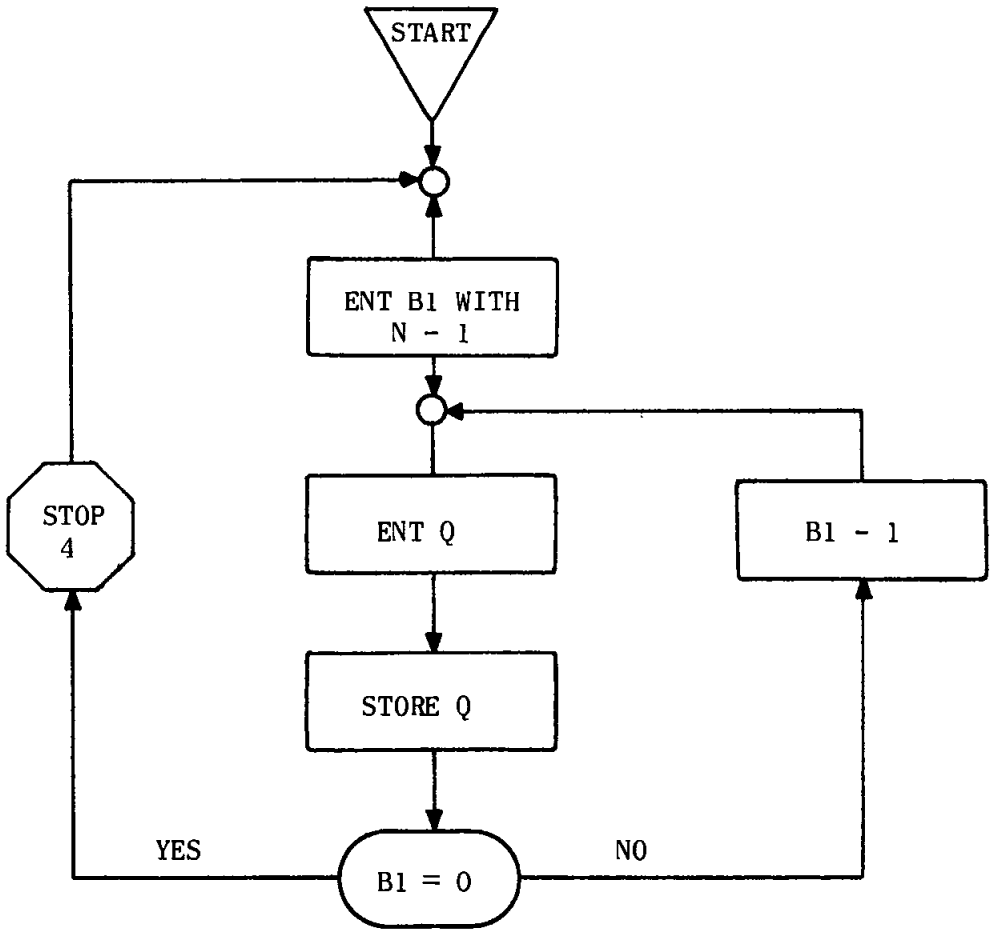
Information can be transferred from consecutive addresses to Q by setting up the repeat mode for enter Q. Master clear the computer and press Af En. Manually set the U register to 70100 00003. Operate the DISCONNECT B7 switch to the up position. Press the OP STEP MODE indicator-switch and then operate the START-STEP/RESTART switch to the START-STEP position. The repeat mode is now active. Clear U and manually set Uu to 10030 with UL equal to the address of the first desired memory reference. Repeated operations of the START-STEP/RESTART switch to the START-STEP position enter the contents of consecutive addresses in Q for inspection.

(2) MANUAL WRITING. - To write from the Q register into one of the storage locations, master clear the computer; press the Af En (Af Enable) indicator-switch to allow the generation of the proper command signals when the computer is started; press the appropriate indicator-switches in the U register for a store Q instruction (f = 14, j = 0, k = 3, and b = 0). The y portion of the U register is loaded with the memory address that is the destination of the word in Q (see figure 3-2b). Set the word to be stored in the Q register. Press the OP STEP MODE indicator-switch. Operate the START-STEP/RESTART switch to the START-STEP position. The computer operates and stops, and the desired word is located at the selected memory location.

Information can be stored in consecutive addresses from Q by setting up the repeat mode for the store Q instruction. Master clear and press the Af En indicator-switch. Set the U register to 70100 00003, operate the DISCONNECT B7 switch to the up position, press the OP STEP MODE indicator-switch, and then operate the START-STEP/RESTART switch to the START-STEP position. The repeat mode is now active. Clear U and enter Uu with 14030 and UL with the lowest address of the desired storage locations. Enter Q with the information to be stored. Operate the START-STEP/RESTART switch to the START-STEP position. The word in Q is stored in the address specified by UL. Load Q with new information and operate the START-STEP/RESTART switch to the START-STEP position. Consecutive addresses can be referenced for storage in this manner. The same word or different words can be stored.

(3) BLOCK TRANSFER. - Block transfer utilizes a small routine which must first be entered in computer memory. This routine can be loaded into the computer manually or via other input devices if the proper loading routine is stored in memory (see figure 3-3). The block transfer subroutine is shown in table 3-3.

When the block transfer has been loaded, perform the following sequence of manual operations: Master clear the computer. Set the P register equal to the initial address of the subroutine (see m1 of table 3-3). This permits the computer, when started, to perform the instruction contained at that location. Operate the START-STEP/RESTART switch to the START-STEP position; this starts operations at normal high-speed. When the transfer is completed, the computer will stop with the red STOP 4 indicator lighted.



Example:

Transfer words in addresses 01520 - 01525 to addresses 01630 - 01635.

1st cycle - Transfer word from y_1 (01520) + N(5) to y_2 (01630) + N(5) and B1 - 1 (N = 4).

2nd cycle - Transfer word from y_1 (01520) + N(4) to y_2 (01630) + N(4) and B1 - 1 (N = 3).

3rd cycle - Transfer 01523 to 01633 and B1 - 1 (N = 2).

4th cycle - Transfer 01522 to 01632 and B1 - 1 (N = 1).

5th cycle - Transfer 01521 to 01631 and B1 - 1 (N = 0).

6th cycle - Transfer 01520 to 01630 and jump (B1 = 0) and STOP 4.

Figure 3-3. Block Transfer Routine Flow Chart

TABLE 3-3. BLOCK TRANSFER SUBROUTINE

RELATIVE ADDRESS	INSTRUCTION	FUNCTION
m1	12100 N	Enter B1 with number of words to be transferred, less one (N).
m2	10031 y1	(y1 equals lowest address of block to be transferred.) Enters word from highest address to be transferred.
m3	14031 y2	(y2 equals lowest address of new storage location.) Stores word at highest address of new location.
m4	72100 m2	If B1 = 0, read NI; if B1 ≠ 0, B1 - 1 and jump to m2. (This operation stops transfer when all words have been transferred.)
m5	61400 m1	Jump to beginning of routine and STOP 4.

3-4. SUMMARY OF OPERATING PROCEDURES.

The following step-by-step procedures are those to be followed for the listed operations.

a. TURN ON.

STEP 1. BLOWER POWER switch to ON.

STEP 2. COMPUTER POWER switch to ON.

STEP 3. Press the PHASE STEP MODE and OP STEP MODE indicator switches, and the MASTER CLEAR pushbutton switch.

b. TURN OFF.

STEP 1. BLOWER POWER switch to OFF.

c. BOOTSTRAP LOAD.

STEP 1. Master clear the computer.

STEP 2. Press LOAD MODE.

STEP 3. Press PROGRAM I or II.

STEP 4. Operate the START-STEP/RESTART switch to START-STEP.

d. HIGH SPEED OPERATIONS.

STEP 1. Master clear the computer.

STEP 2. Set starting address in P register.

STEP 3. Operate the START-STEP/RESTART switch to START-STEP.

e. PHASE STEP OPERATIONS.

STEP 1. Master clear the computer.

STEP 2. Press PHASE STEP MODE.

STEP 3. Set the instruction in the U register.

STEP 4. Operate the START-STEP/RESTART switch to START-STEP (for one clock phase).

STEP 5. Operate the START-STEP/RESTART switch to START-STEP (for each additional clock phase desired).

or

STEP 1. Repeat steps 1 thru 3.

STEP 2. Operate START-STEP/RESTART switch to RESTART.

STEP 3. Rotate RESTART SPEED CONTROL switch to the desired rate of clock phase control.

STEP 4. Operate the START-STEP/RESTART switch to neutral to stop.

f. OPERATION STEP.

STEP 1. Master clear the computer.

STEP 2. Press OP STEP MODE.

STEP 3. Set starting address in P register.

STEP 4. Operate the START-STEP/RESTART switch to START-STEP (for one instruction execution.

STEP 5. Operate the START-STEP/RESTART switch to START-STEP (for each additional instruction execution desired).

or

STEP 1. Repeat steps 1 thru 3.

STEP 2. Operate the START-STOP/RESTART switch to RESTART.

STEP 3. Rotate RESTART SPEED CONTROL switch to the desired rate of instruction execution.

STEP 4. Operate the START-STEP/RESTART switch to neutral to stop.

g. STOP. - Operate STOP switch down at any time to stop operation.

h. PHASE REPEAT.

STEP 1. Master clear the computer.

STEP 2. Press PHASE STEP MODE.

STEP 3. Press desired CLOCK PHASE.

STEP 4. Operate the PHASE REPEAT switch to the up position. To advance the clock it is necessary to press the phase indicators in order.

STEP 5. Operate the PHASE REPEAT switch to neutral to stop.

i. CLEAR FAULT INDICATOR. - Press MASTER CLEAR.

j. SILENCE OVERTEMPERATURE ALARM HORN. - Press the OVERTEMP WARNING indicator.

3-5. EMERGENCY OPERATION.

There are no emergency operation procedures for the computer.

3-6. TEST PROCEDURES.

No test procedures are to be performed by the operator. See Section 5 for troubleshooting and test procedures.

3-7. OPERATOR'S MAINTENANCE.

See Section 5 for operator maintenance routines.

3-8. PROGRAMMING.

a. PROGRAMMING CONCEPTS. - Once the program is written and coded in an acceptable form, it is entered into the storage section of the computer. From this point, the computer, upon proper initiation, executes the instructions of the program. The instructions of the program are stored in the memory (storage) section of the computer in a sequential manner. The computer will first execute the instruction located at the lowest address of the program and proceed to the highest address. From its storage location, the instruction is moved to the control section, where the computer analyzes the instruction to determine the method of execution. (The instruction in its original form is not altered in memory due to this process.) Normally, the next instruction to be executed is located at the address that is, in value, one greater than the address of the previous instruction.

Any problem that can be solved by mathematical procedures can be solved by the computer. If it is determined that the problem can be best solved by the computer, it is then necessary to formulate the problem in the language of the computer. A program of instructions and the data needed for the solution of the problem must be devised.

(1) FLOW DIAGRAMS. - A flow diagram is helpful in facilitating the coding or programming of the problem. A flow diagram or flow chart indicates the flow, or series of steps, in the computation that leads to the solution of a particular problem. A basic flow diagram usually lists the series of simple arithmetic steps that are to be performed by the computer. It is imperative that the coder be familiar with the overall operations and the peculiarities of each computer instruction so that he can construct the outline with regard to the capabilities of the computer.

(2) BASIC PROCEDURES. - Usually, more than one flow diagram is formed for more complicated problems. The first flow outline may be equations in mathematical language written in the sequence in which they will be computed, together with brief explanations of the steps involved. The second flow chart formulates the flow of computation as the problem will be computed in the computer. This chart contains the instructions necessary for the data input, the instructions that operate on the input data to obtain the solutions, and the necessary instructions for the output of the results. Many problems are such that the second type of flow diagram will consist of many charts and/or diagrams, each a more detailed presentation of the preceding chart.

b. INSTRUCTION WORD. - The instruction word is placed in the U register and its outputs are applied to translator circuits which interpret the bits. The instruction word bits are illustrated in figures 3-4 and 3-5. Further simplification of the word is attained by using the octal numbering system because of the ease of conversion. The binary notation of an instruction could be as follows:

001 100 010 011 101 000 010 001 111 110

Octal notation of the same instruction would be 14 2 3 5 02176. The designators can be stated as f = 14, j = 2, k = 3, b = 5, and y = 02176. The translation of these designators determines the exact method of executing the instruction.

(1) FUNCTION CODE DESIGNATION. - The f designator appears in bit positions 29 through 24 of the U register or an instruction and designates the function to be performed by that instruction. All values of f other than 00 and 77 are defined in the instruction repertoire (see table 3-4). Codes 00 and 77 are program fault conditions; if executed, they cause a fault interrupt and a jump to address 00000, the fault entrance register, or address 00540 of memory, depending on the AUTOMATIC RECOVERY switch setting.

(2) BRANCH CONDITION DESIGNATOR. - The j designator appears in bit positions 23, 22, and 21 of the U register or an instruction; it is used in a majority of the instructions. The three primary uses of j are for jump and skip determination, for B register specification, and for repeat status interpretation. Appropriate interpretations of the j designator are listed either below or under the descriptions of the individual instructions.

For those instructions in which the j designator has no special interpretation, it specifies the condition under which the next sequential instruction in the program will be skipped. This permits branching from a sequence without executing a jump instruction, as would normally occur if a skip condition were not satisfied.

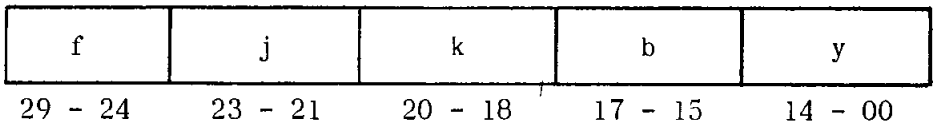


Figure 3-4. Normal Instruction Word Format

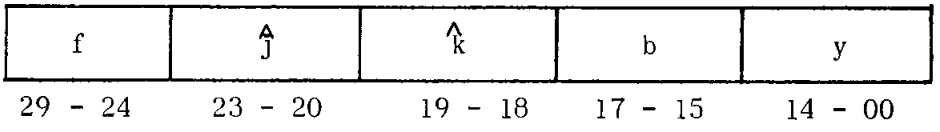


Figure 3-5. Input/Output Instruction Word Format

NOTE: $\hat{j} = C^n$ Input/Output Channel

A skip of the next sequential instruction is determined by the following rules in all instructions except 04, 12, 13, 16, 17, 26, 27, 60-67, and 70-76:

j = 0: Do not skip the next instruction.

j = 1: Skip the next instruction.

j = 2: Skip the next instruction if (Q) is positive.

j = 3: Skip the next instruction if (Q) is negative.

j = 4: Skip the next instruction if (A) is zero (positive zero).

j = 5: Skip the next instruction if (A) is nonzero.

j = 6: Skip the next instruction if (A) is positive.

j = 7: Skip the next instruction if (A) is negative.

When the branch condition involves the sign of the quantity in A or Q, the evaluation examines the sign bit of these quantities; hence, positive zero (all zeros) is considered a positive quantity, and negative zero (all ones) is considered a negative quantity.

For input/output instructions, the $\hat{j}$ designator appears in bit positions 23, 22, 21, and 20 of the U register or an input/output instruction and specifies the channel for the instruction. Bit 23 assumes a value of eight, bit 22 a value of four, bit 21 a value of two, and bit 20 a value of one; thus, the $\hat{j}$ designator provides accessibility to the sixteen (decimal) input/output channels (numbered $0\text{-}17_8$). Instructions 13, 17, 62, 63, 66, 67, 73, 74, 75, and 76 use the $\hat{j}$ designator configuration.

(3) OPERAND INTERPRETATION DESIGNATOR. - The k designator (three bits) appears in bit positions 20, 19, and 18 of the U register or an instruction. The $\hat{k}$ designator appears only in bit positions 19 and 18 since bit 20 is a portion of the $\hat{j}$ designator. Instructions 13, 17, 62, 63, 66, 67, and 73 through 76 use the $\hat{k}$ designator configuration since they perform input/output activities and require a $\hat{j}$ designator for channel specification. The $\hat{k}$ designator controls operand and instruction interpretation for the input/output instructions as noted in the description of the applicable instruction [see paragraph 3-8b(2)].

The k designator controls operand interpretation. Those instructions that read an operand but do not replace it after the arithmetic is performed are designated Read instructions. Those instructions that do not read an operand but store it are designated Store instructions. Instructions which both read and store operands are classified as Replace instructions.

The various values of k affect the operand except where noted otherwise under individual instruction description as follows:

Read instructions (01-12, 20-23, 26, 27, 30, 31, 40-43, 50-53, 60-65, 70-72):

k = 0: Yu = 0's; YL = Y

k = 1: Yu = 0's; YL = (Y)L

k = 2: Yu = 0's; YL = (Y)u

k = 3: Y = (Y)

k = 4: Yu = same bits as Y14; YL = Y

k = 5: Yu = same bits as Y14; YL = (Y)L

k = 6: Yu = same bits as Y29; YL = (Y)u

k = 7: Y = (A)

For instructions 22, 52, and 53, k = 7 is not used.

Store instructions (14-16, 32, 33, 47):

k = 0: Store operand in Q.*

k = 1: Store operand L in YL, leaving (Y)u undisturbed.

k = 2: Store operand L in Yu, leaving (Y)L undisturbed.

k = 3: Store operand in Y.

k = 4: Store operand in A.**

k = 5: Store complement of operand L in YL, leaving (Y)u undisturbed.

k = 6: Store the complement of operand L in Yu, leaving (Y)L undisturbed.

k = 7: Store the complement of operand in Y (storing the complement of Bj is the same complement as for a 30-bit register).

Replace instructions (24, 25, 34-37, 44-46, 54-57):

k = 0: Not used.

k = 1: Read portion - Yu = 0's; YL = (Y)L.
Store portion - Stores operand L in YL leaving (Y)u undisturbed.

k = 2: Read portion - Yu = 0's; YL = (Y)u
Store portion - stores operand L in Yu leaving (Y)L undisturbed.

k = 3: Read portion - Y = (Y).
Store portion - Stores operand in Y.

k = 4: Not used.

k = 5: Read portion - Yu = same bits as Y14: YL = (Y)L
Store portion - Stores operand L in YL leaving (Y)u undisturbed.

k = 6: Read portion - Yu = same bits as Y29; YL = (Y)u
Store portion - Stores operand L in Yu leaving (Y)L undisturbed.

k = 7: Not used.

Repeat instructions require special interpretation when followed by a replace instruction.

(4) INDEX DESIGNATOR. - The b designator appears in bit positions 17, 16, and 15 of the U register or an instruction; it specifies which of the B-index registers, if any, will be used to modify the operand address designator, y, to form

* A 1400000000 instruction complements (Q)
**A 1504000000 instruction complements (A)

Y = y + (Bb). This operation employs an additive accumulator; hence, a quantity consisting of all zeros cannot result unless the bits of both the operand designator, y, and (Bb) are all zeros. Use of a B-index register to modify y to form Y = y + (Bb) causes the higher-order 15 bits of the B register memory address to be made zero.

The effect of the various values of the b designator is as follows:

b = 0: Do not modify y.

b = 1: Add (B1) to y.

b = 2: Add (B2) to y.

b = 3: Add (B3) to y.

b = 4: Add (B4) to y.

b = 5: Add (B5) to y.

b = 6: Add (B6) to y.

b = 7: Add (B7) to y.

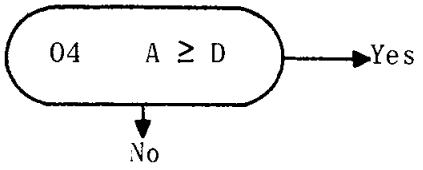
(5) OPERAND DESIGNATOR. - The y designator appears in bit positions 14 through 00 of an instruction. The operand or address of an operand Y is relative to y since Y = y + (Bb).

c. PROGRAMMING SYMBOLS. - To make it easier to formulate and understand flow diagrams, certain symbols are used. The following list of symbols gives the coder an example of the basic symbols used in drawing the second type of flow chart (see figure 3-6).

(1) LINES OF FLOW. - A solid line with an arrow touching the next element of the flow diagram usually is used to indicate the path to be followed by the computer, or, more precisely, the path to be followed by the coder who is formulating the computer instructions from the flow diagrams.

(2) OPERATION SYMBOLS. - The rectangular box usually contains a statement about a computer or mathematical operation. The content of the box may be a simple statement or a mathematical expression.

(3) DECISION SYMBOLS. - An oval is used to indicate a two-way decision. This symbol is sometimes written as:



The function code, 04, designates the use of the Compare instruction to make an evaluation in the computer.

(4) CONNECTORS AND REMOTE CONNECTORS. - To eliminate crossing flow lines on a diagram, remote connectors are used to indicate a destination not easily reached in the diagram. Thus, the flow can be broken at a convenient point by terminating it in an arbitrary symbol which can be used to initiate the flow in another region of the diagram.

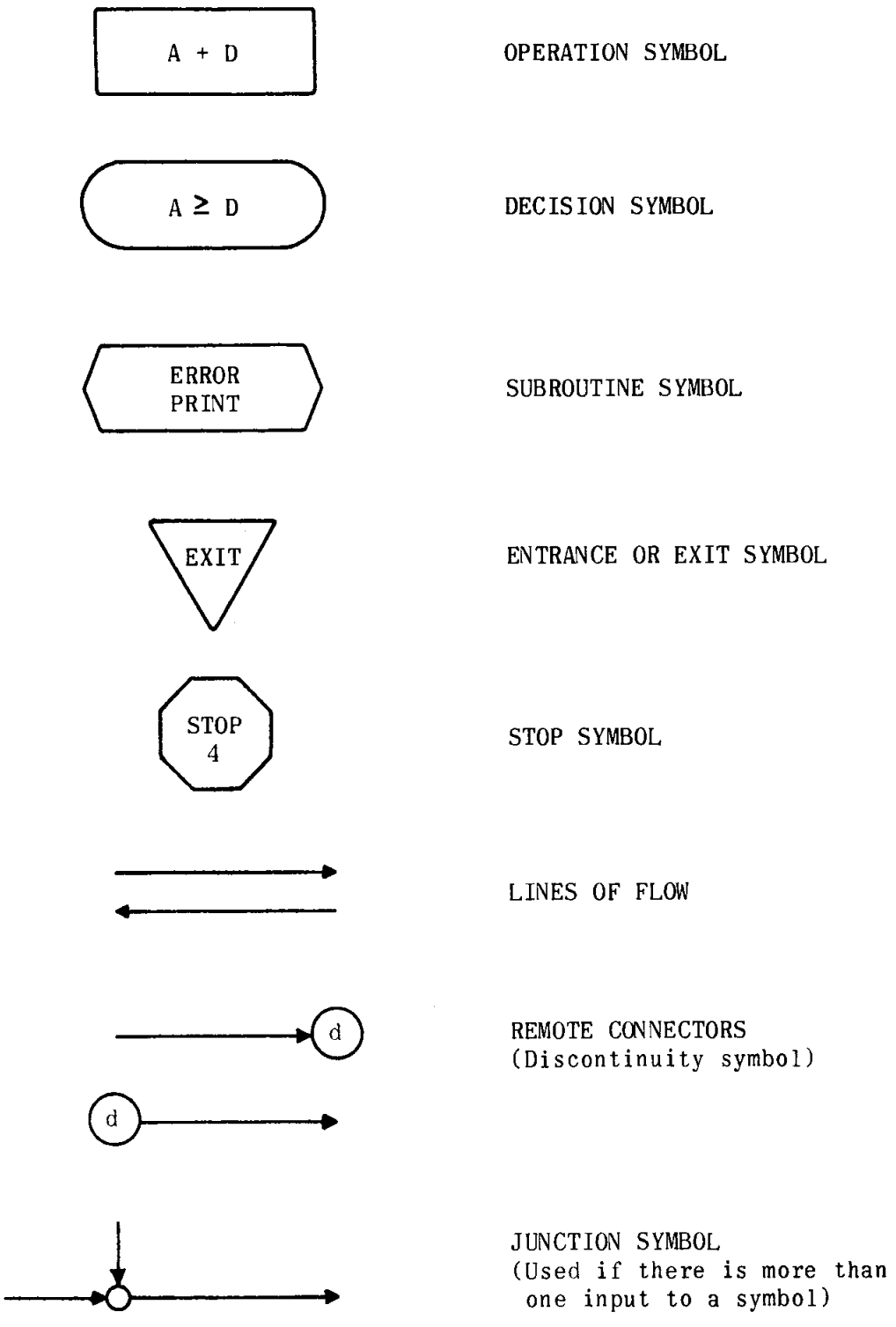


Figure 3-6. Programming Symbols

3-9. REPERTORY OF INSTRUCTIONS. - (See table 3-4.)

The CP-642B/USQ-20(V) computer is specified as a self-modifying, single-address computer. Although this means that one reference or address is provided for the execution of instruction, this reference can be modified automatically during a programmed sequence. The references are modified by using the B registers which contain any previously stored constants. In order to modify the address, the content of a selected B register is added to the operand address designator, y. A programmed address is coded using octal notation with each octal digit denoting three binary digits. The instructions are read sequentially from memory storage except after satisfied jump or skip instructions. Every instruction executed by the computer is transmitted from memory to the Z register and then to the U register. While the instruction is in the U register, its components are translated to determine the exact method of executing the instruction. The instructions and explanations of their common usage are listed below. Instruction execution times are tabulated in table 3-4.

01 - Right Shift Q. _ Shifts (Q) to the right Y bit-positions.* The higher-order bits are replaced with the original sign bit as the word is shifted. Only the lower-order six bits of Y are recognized for this instruction. The higher-order 24 bits are ignored.

Example of two executions of right shift in Q: Y = 1

Content of Q		Content of Q	
$(Q)_i$ (positive) =	0101	$(Q)_i$ (negative) =	1010
First shift	0010	First shift	1101
Second shift	0001	Second shift	1110

02 - Right Shift A . - Shift (A) to the right Y bit positions.* The higher-order bits are replaced with the original sign bit as the word is shifted. Only the lower-order six bits of Y are recognized for this instruction. The higher-order 24 bits are ignored. The over-all operation is analogous to the example given in instruction 01.

03 - Right Shift AQ. - Shift (A) and (Q) as one 60-bit register. The shift is to the right Y bit positions* with the lower bits of A shifting into the higher bit positions of Q. The higher-order bits of A are replaced with the original sign bit as the word is shifted. Only the lower-order six bits of Y are recognized for this instruction. The higher-order 24 bits are ignored.

Example of two executions of right shift in AQ: Y = 1

Content of AQ		Content of AQ	
$(AQ)_i$ (positive) =	01010011	$(AQ)_i$ (negative) =	10001010
First shift	00101001	First shift	11000101
Second shift	00010100	Second shift	11100010

* The maximum shift count shall be 63 decimal (D) places.

04 - Compare - Compare the signed value of Y with the signed value of (A) and (Q) or the value of either. It does not alter either (A) and (Q). The branch condition designator, j, is interpreted for this instruction as listed below:

j = 0: Do not skip the next instruction.

j = 1: Skip the next instruction.

j = 2: Skip the next instruction if Y is less than or equal to (Q).

j = 3: Skip the next instruction if Y is greater than (Q).

j = 4: Skip the next instruction if (Q) is greater than, or equal to Y, and Y is greater than (A).

j = 5: Skip the next instruction if Y is greater than (Q), or if Y is less than, or equal to (A).

j = 6: Skip the next instruction if Y is less than, or equal to (A).

j = 7: Skip the next instruction if Y is greater than (A).

05 - Left Shift Q. - Shift (Q) circularly to the left Y bit positions.* The lower-order bits are replaced with the higher-order bits as the word is shifted. Only the lower-order six bits of Y are recognized for this instruction; the higher-order 24 bits are ignored.

Example of two executions of left circular shift in Q:Y = 1

Content of Q		Content of Q	
$(Q)_i$ (positive) =	0011	$(Q)_i$ (negative) =	1100
First shift	0110	First shift	1001
Second shift	1100	Second shift	0011

06 - Left Shift A. - Shift (A) circularly to the left Y bit positions.* The lower-order bits are replaced with the higher-order bits as the word is shifted. Only the lower-order six bits of Y are recognized for this instruction; the higher-order 24 bits are ignored. The over-all operation is analogous to the example given in instruction 05.

07 - Left Shift AQ. - Shift (A) and (Q) as one 60-bit register. The shift is circular to the left Y bit positions.* The lower bits of A are replaced with the higher bits of Q and the lower bits of Q are replaced with the higher bits of A. Only the lower-order six bits of Y are recognized by this instruction; the higher-order 24 bits are ignored.

Example of left shift in AQ: Y = 1

Content of AQ		Content of AQ	
$(AQ)_i$ (positive) =	01010011	$(AQ)_i$ (negative) -	10001011
First shift	10100110	First shift	00010111
Second shift	01001101	Second shift	00101110

* The maximum shift count shall be 63 decimal (D) places.

TABLE 3-4. INSTRUCTION REPERTORY AND INSTRUCTION EXECUTION TIME

OCTAL CODE	INSTRUCTION	EXECUTION*	ABORT*	REPEAT*	SPECIAL CONSIDERATIONS
00	(Fault interrupt)	8-12	4	4-8	0-3, 8-11, 16-19, or 24-27 place shifts- 8 μsec 4-7, 12-15, 20-23, 28-63 place shifts- 12 μsec
01	RIGHT SHIFT Q	8-12	4	4-8	
02	RIGHT SHIFT A	8-12	4	4-8	
03	RIGHT SHIFT AQ				
04	COMPARE	8	4	4	
05	LEFT SHIFT Q	8-12	4	4-8	0-3, 8-11, 16-19, or 24-27 place shifts- 8 μsec 4-7, 12-15, 20-23, 28-63 place shifts- 12 μsec
06	LEFT SHIFT A	8-12	4	4-8	
07	LEFT SHIFT AQ	8-12	4	4-8	
10	ENTER Q	8	4	4	
11	ENTER A	8	4	4	
12	ENTER Bj	8	-	4	
13	EXTERNAL FUNCTION on $C_j^{\wedge}$	8	-	-	13 $\hat{k}$ = 3-max of 36 μsec 13 $\hat{k}$ = 0,1,2 - 8 μsec
14	STORE Q	8	4	4	
15	STORE A	8	4	4	
16	STORE $B_j^{\wedge}$	8	-	4	
17	STORE $C_j^{\wedge}$ or TEST EFB	8-12	-	-	17 $\hat{k}$ = 2-max of 36 μsec 17 $\hat{k}$ = 0, 1-8 μsec 17 $\hat{k}$ = 3-12 μsec
20	ADD A	8	4	4	
21	SUBTRACT A	8	4	4	
22	MULTIPLY	32-52	4	32-52	
23	DIVIDE	52	4	32	k = 7 Square Root Execution time = 52 μsec
24	REPLACE A + Y	12	0	8	
25	REPLACE A - Y	12	0	8	
26	ADD Q	8	4	4	
27	SUBTRACT Q	8	4	4	
30	ENTER Y + Q	8	4	4	
31	ENTER Y - Q	8	4	4	
32	STORE A + Q	12	0	8	
33	STORE A - Q	12	0	8	
34	REPLACE Y + Q	12	0	8	
35	REPLACE Y - Q	12	0	8	

* Time in microseconds

TABLE 3-4. INSTRUCTION REPERTORY AND INSTRUCTION EXECUTION TIME (CONT.)

OCTAL CODE	INSTRUCTION	EXECUTION*	ABORT*	REPEAT*	SPECIAL CONSIDERATIONS
36	REPLACE Y + 1	12	0	8	
37	REPLACE Y - 1	12	0	8	
40	ENTER LOGICAL PRODUCT	8	4	4	
41	ADD LOCIGAL PRODUCT	8	4	4	
42	SUBTRACT LOGICAL PRODUCT	8	4	4	
43	COMPARE MASK	8	4	4	
44	REPLACE LOGICAL PRODUCT	12	0	8	
45	REPLACE A + LOGICAL PRODUCT	12	0	8	
46	REPLACE A - LOGICAL PRODUCT	12	0	8	
47	STORE LOGICAL PRODUCT	8	4	4	
50	SELECTIVE SET	8	4	4	
51	SELECTIVE COMPLEMENT	8	4	4	
52	SELECTIVE CLEAR	8	4	4	
53	SELECTIVE SUBSTITUTE	8	4	4	
54	REPLACE SELECTIVE SET	12	0	8	
55	REPLACE SELECTIVE COMPLEMENT	12	0	8	
56	REPLACE SELECTIVE CLEAR	12	0	8	
57	REPLACE SELECTIVE SUBSTITUTE	12	0	8	
60	JUMP (Arithmetic)	8	-	-	If jump condition is not satisfied, execution time is 4 μsec
61	JUMP (Manual)	8	-	-	
62	JUMP ON $C\hat{j}$ ACTIVE INPUT BUFFER	8	-	-	
63	JUMP ON $C\hat{j}$ ACTIVE OUTPUT BUFFER	8	-	-	
64	RETURN JUMP (Arithmetic)	12	0	0	If return jump condition is not satisfied, execution time is 4 μsec
65	RETURN JUMP (Manual)	12	-	-	
66	TERMINATE $C\hat{j}$ INPUT BUFFER OR ENABLE, DISABLE INTERRUPTS	4	-	-	
67	TERMINATE $C\hat{j}$ OUTPUT BUFFER, ALL BUFFERS, OR TERMINATE Cj EXTERNAL FUNCTION BUFFER	4	-	-	
70	REPEAT	8	-	-	
71	B SKIP ON Bj	8	0	-	
72	B JUMP ON Bj	8	-	-	
73	INPUT BUFFER ON $C\hat{j}$ (without monitor mode)	8	-	-	
74	OUTPUT BUFFER ON $C\hat{j}$ without monitor mode)	8	-	-	
75	INPUT BUFFER ON $C\hat{j}$ (with MONITOR mode)	8	-	-	
76	OUTPUT BUFFER ON $C\hat{j}$ (with MONITOR mode)	8	-	-	
77	(Fault interrupt)				

10 - Enter Q. - Enter Y in Q.

11 - Enter A. - Enter Y in A.

12 - Enter Bj. - Enter Y in B register specified by j. The branch condition designator, j, is used to specify the selected B register for this instruction and is not available for its normal function.

13 - External function on $C\hat{j}$. - Establish a one word external function buffer via the output buffer channel $\hat{j}$ from storage address Y. The buffered word address is maintained in the lower order 15 bits of storage address 00140 plus $\hat{j}$.

This instruction is implemented as follows: For all $\hat{k}$ values, store Y in the upper- and lower-order half of storage location 00140 plus $\hat{j}$. The $\hat{k}$ values modify the instructions as follows:

k = 0: Establish a one word external function buffer with monitor and proceed to the next instruction. A monitor interrupt follows the completion of the buffering operation. Subsequent to this instruction the individual transfer shall be accomplished when requested by the external device.

k = 1: Establish a one word external function buffer with monitor and with force. The force considerations for $\hat{k}$ = 3 apply. A monitor interrupt follows the completion of the buffering operation.

k = 2: Initiate a one word external function buffer and proceed to the next instruction. Subsequent to this instruction the transfer shall be accomplished when requested by the external device.

k = 3: Establish a one word external function buffer with force (with force the instruction ignores the external function request from the external device). The program will hold until the external function word is transmitted. If the external devices cannot accept external functions executed consecutively, restrictions must be made in the programming of external function instructions to this equipment.

14 - Store Q. - Store (Q) at storage address Y as directed by the operand interpretation designator, k. If k = 0, complement (Q). If k = 4, store in A.

15 - Store A. - Store (A) at storage address Y as directed by the operand interpretation designator, k. If k = 4, complement (A). If k = 0, store in Q.

16 - Store Bj. - Store a 30-bit quantity whose lower order 15 bits correspond to the content of B register specified by j and whose higher order 15 bits are zero at storage address Y as directed by the operand interpretation designator, k. The branch condition designator, j is used to specify the selected B register for this instruction and is not available for its normal function.

17 - Store $C\hat{j}$ or Test EFB. - Performs a storage or jump function as specified below:

k = 1 or 0: Clear the program-address register P, and enter a new address in P for certain external-function buffer conditions on the channel designated by $\hat{j}$. If the buffer is active, the jump condition is

satisfied; then if $\hat{k} = 1$, Y L; or if $\hat{k} = 0$, Y becomes the address of the next instruction. If the buffer is inactive, the jump condition is not satisfied. The next sequential instruction in the current sequence shall be executed in the normal manner.

k = 2: Store the contents of the C channel specified by $\hat{j}$ at storage address Y. An input data acknowledge with force signal is then sent on the C channel. The program will hold until the word is read.

k = 3: Store the contents of storage address 00520 plus $\hat{j}$ at storage address Y. The external-interrupt-request line is reset on channel C_j.

20 - Add A. - Add Y to the previous content of A.

21 - Subtract A. - Subtract Y from the previous content of A.

22 - Multiply. - Multiply (Q) times Y, leaving a double-length product in AQ. If the factors are considered as integers, the product is an integer in AQ.

The branch condition designator, j, is interpreted prior to end correction permitting sensing of a product with (A)f = 0. If j = 4, a skip of the next instruction is made when (A)f = 0. (A)f $\neq$ + 0, a double-length product has been formed with significant bit(s) in the A register. However, if a skip does occur for j = 4, the multiply instruction can be re-executed with the same operand and with j = 2 or 3 to determine if Q29 contains a significant bit (a one) of the product.

In this instruction, k = 7 is not used.

23 - Divide. - Perform a divide or square root operation as specified by the k designator. If k $\neq$ 7, divide (AQ) by Y leaving the quotient in the Q register and the remainder in the A register. The remainder bears the same sign as the dividend. If k = 7, take the square root of (Q) leaving the root in the Q register and the remainder in the A register. The remainder a = n-(QxQ).

NOTE

If a divide overflow condition exists, no maintenance console indication is given. However, by coding each divide instruction with j = 3, a program test for the divide overflow is automatic. With this selection of j, a skip of the next instruction occurs if a divide overflow exists. The skip should be made to a jump instruction which provides a remedial means of noting the error or of correcting it. Therefore, the instruction which follows the divide instruction should have its j = 1 in order to preclude the jump instruction whenever the divide sequence culminates in a correct answer.

A divide overflow can be detected also if the divide instruction is executed with j = 2. In this case, a correct answer is indicated when a skip occurs.

24 - Replace A + Y. - Add Y to (A) and store the results in Y and A.

25 - Replace A - Y. - Subtract Y from (A) and store the results in Y and A. In this instruction k = 0 and k = 7 are not used.

26 - Add Q. - Add Y to (Q) and store the results in Q. The branch condition designator, j, has special meaning in this instruction as specified in instruction 27.

27 - Subtract Q. - Subtracts Y from (Q) and stores the results in Q.

In instructions 26 and 27, the branch condition designator, j, has the following meaning:

j = 0: Do not skip the next instruction.

j = 1: Skip the next instruction.

j = 2: Skip the next instruction if (A) is positive.

j = 3: Skip the next instruction if (A) is negative.

j = 4: Skip the next instruction if (Q) is zero.

j = 5: Skip the next instruction if (Q) is nonzero.

j = 6: Skip the next instruction if (Q) is positive.

j = 7: Skip the next instruction if (Q) is negative.

30 - Enter Y + Q. - Add (Q) to Y and enter the results in A.

31 - Enter Y - Q. - Subtract (Q) from Y and enter the results in A.

32 - Store A + Q. - Add (Q) to (A) and store the results at A and Y as directed by the operand interpretation designator, k.

33 - Store A - Q. - Subtract (Q) from (A) and store the results in A and Y as directed by the operand interpretation designator, k.

34 - Replace Y + Q. - Add (Q) to Y and store the results in Y and A. In this instruction k = 0 and k = 7 are not used.

35 - Replace Y - Q. - Subtract (Q) from Y and store the results in Y and A. In this instruction k = 0 and k = 7 are not used.

36 - Replace Y + 1. - Add 1 to Y and store the results in Y and A. In this instruction k = 0 and k = 7 are not used.

37 - Replace Y - 1. - Subtract 1 from Y and store the results in Y and A. In this instruction k=0 and k=7 are not used.

40 - Enter Logical Product. - Enter the bit-by-bit logical product of Y and (Q) in A.

In this instruction, the j designator is interpreted in a special way for the values j = 2 or 3. If j = 2, skip if the parity of (A)f is even. If j = 3, skip if the parity of (A)f is odd.

NOTE

Even parity - an even number of "ones" in the A register. Odd parity - an odd number of "ones" in the A register.

41 - Add Logical Product. - Add to (A) the bit-by-bit product of Y and (Q).

42 - Subtract Logical Product. - Subtract from (A) the bit-by-bit product of Y and (Q).

43 - Compare Mask. - Subtract from (A) the bit-by-bit product of Y and (Q), and perform the branch point evaluation for skip of next sequential instruction as directed by the branch condition designator, j.

This instruction results in no net change in the content of any operational register. It provides, through the branch condition designator, j, a comparison of a portion of Y with (A).

44 - Replace Logical Product. - Enter in A the bit-by-bit product of Y and (Q). Then store (A) at storage address Y.

In this instruction, the j designator is interpreted in a special way for the values j = 2 or 3. If j = 2, skip if the parity of (A)f is even. If j = 3, skip if the parity of (A)f is odd. In this instruction k = 0 and k = 7 are not used.

45 - Replace A + Logical Product. - Add to (A) the bit-by-bit product of Y and (Q) and enter the sum at storage address Y. In this instruction k = 0 and k = 7 are not used.

46 - Replace A - Logical Product. - Subtract from (A) the bit-by-bit product of Y and (Q) and enter the difference at storage address Y. In this instruction k = 0 and k = 7 are not used.

47 - Store Logical Product. - Store in address Y the bit-by-bit product of (A) and (Q) as directed by the operand interpretation designator, k.

50 - Selective Set. - Set the individual bits of A to "ones" corresponding to "ones" in Y leaving the remaining bits of A unaltered.

51 - Selective Complement. - Complement the individual bits of A corresponding to "ones" in Y leaving the remaining bits of A unaltered. If k = 4, this instruction shall enter A' in A when Y is 77777.

52 - Selective Clear. - Clear the individual bits of A corresponding to "ones" in Y leaving the remaining bits of A unaltered. In this instruction k = 7 is not used.

53 - Selective Substitute. - Substitute the individual bits of A with bits of Y corresponding to "ones" in Q leaving the remaining bits of A unaltered. In this instruction k = 7 is not used.

54 - Replace Selective Set. - Set the individual bits of A to "one" corresponding to "ones" in Y leaving the remaining bits of A unaltered and then store (A) at a storage address Y. In this instruction k = 0 and k = 7 are not used.

55 - Replace Selective Complement. - Complement the individual bits of A corresponding to "ones" in Y leaving the remaining bits of A unaltered and then store (A) at storage address Y. In this instruction k = 0 and k = 7 are not used.

56 - Replace Selective Clear. - Clear the individual bits of A corresponding to "ones" in Y leaving the remaining bits of A unaltered and then store (A) at storage address Y. In this instruction k = 0 and k = 7 are not used.

57 - Replace Selective Substitute. - Substitute the individual bits of A with bits of Y corresponding to "ones" in Q leaving the remaining bits of A unaltered and then store (A) at storage address Y. In this instruction k = 0 and k = 7 are not used.

60 - Jump (Arithmetic). - Clear the program address register, P, and enter a new program address in P for certain conditions of either the A or Q register content. The branch condition designator, j, is interpreted in a special way for this instruction and thus determines the conditions under which a jump in program address occurs. If the jump condition is not satisfied, the next sequential instruction in the current sequence is executed in a normal manner. If the jump condition is satisfied, as listed below, then Y becomes the address of the next instruction and the beginning of a new program sequence.

j = 0: No jump. Set interrupt enable to remove interrupt lockout, thus clearing bootstrap and interrupt modes. Continue with current program sequence.

j = 1: Execute jump. Set interrupt enable to remove interrupt lockout, thus clearing bootstrap and interrupt modes.

j = 2: Execute jump if (Q) is positive.

j = 3: Execute jump if (Q) is negative.

j = 4: Execute jump if (A) is zero.

j = 5: Execute jump if (A) is nonzero.

j = 6: Execute jump if (A) is positive.

j = 7: Execute jump if (A) is negative.

61 - Jump (Manual). - Clear the program address register, P, and enter a new program address in P for certain conditions of manual JUMP key selections. The branch condition designator, j, is interpreted in a special way for this instruction and thus determines the conditions under which a jump in program address occurs. If the jump condition is not satisfied, the next sequential instruction of the current sequence is executed in a normal manner. If the jump condition is satisfied, as listed below, then Y becomes the address of the next instruction and the beginning of a new program sequence.

Program execution may be stopped by certain STOP selections on execution of this instruction. The branch condition designator, j, specifies which key selections are effective.

j = 0: Execute jump regardless of key selections.

j = 1: Execute jump if JUMP 1 is selected.

j = 2: Execute jump if JUMP 2 is selected.

j = 3: Execute jump if JUMP 3 is selected.

j = 4: Execute jump. Stop computation.

j = 5: Execute jump. Stop computation if STOP 5 is selected.

j = 6: Execute jump. Stop computation if STOP 6 is selected.

j = 7: Execute jump. Stop computation if STOP 7 is selected.

62 - Jump On Cn Active Input Buffer. - Clear the program address register, P, and enter a new program address in P for certain input buffer conditions on the channel designated by $\hat{j}$. If the buffer is active, the jump condition is satisfied; then Y becomes the address of the next instruction. If the buffer is inactive, the jump condition is not satisfied. The next sequential instruction in the current sequence is executed in normal manner. $\hat{k}$ = 0, 1, 2, or 3 permitted.

63 - Jump on C$\hat{j}$ Active Output Buffer. - Clear the program address register, P, and enter a new address in P for certain output buffer conditions on the channel designated by $\hat{j}$. If the buffer is active, the jump condition is satisfied; then Y becomes the address of the next instruction. If the buffer is inactive, the jump condition is not satisfied. The next sequential instruction in the current sequence is executed in the normal manner. $\hat{k}$ = 0, 1, 2, or 3 permitted.

64 - Return Jump (Arithmetic). - Execute a return-jump sequence for certain conditions of either the A or Q register content. The branch condition designator, j, is interpreted in a special way for this instruction and determines the conditions under which the return jump sequence is executed. If the return jump condition is not satisfied, then the next sequential instruction in the current sequence is executed in a normal manner. If the return jump condition is satisfied, as listed below, then the following sequence is performed:

Store (P) + 1 in the lower half of memory address Y. Then jump to Y + 1.

j = 0: No action. Continue with the current program sequence.

j = 1: Execute return jump.

j = 2: Execute return jump if (Q) is positive.

j = 3: Execute return jump if (Q) is negative.

j = 4: Execute return jump if (A) is zero.

j = 5: Execute return jump if (A) is nonzero.

j = 6: Execute return jump if (A) is positive.

j = 7: Execute return jump if (A) is negative.

65 - Return Jump (Manual). - Execute a return jump sequence for certain conditions of manual key selections. The branch condition designator, j, is interpreted in a special way for this instruction and determines the conditions under which the return jump sequence is executed. If the return jump condition is not satisfied, the next sequential instruction in the current sequence is executed in a normal manner. If the return jump condition is satisfied, as listed below, then the following sequence is performed.

Store (P) + 1 in the lower half of memory address Y. Then jump to Y + 1.

j = 0: Execute return jump regardless of key selections.

j = 1: Execute return jump if JUMP 1 is selected.

j = 2: Execute return jump if JUMP 2 is selected.

j = 3: Execute return jump if JUMP 3 is selected.

j = 4: Execute return jump. Then stop computation.

j = 5: Execute return jump. Stop computation if STOP 5 is selected.

j = 6: Execute return jump. Stop computation if STOP 6 is selected.

j = 7: Execute return jump. Stop computation if STOP 7 is selected.

66 - Terminate $C\hat{j}$ Input Buffer or Enable, Disable Interrupts. - This instruction is variable, depending upon the $\hat{k}$ designator and b designator selections.

$\hat{k} = 0$: Terminate the input buffer on channel $\hat{j}$

$\hat{k} = 1$ & $b = 0$: Enable all interrupts

$\hat{k} = 1$ & $b \neq 0$: Disable all interrupts

$\hat{k} = 2$ & $b = 0$: Enable all external interrupts

$\hat{k} = 2$ & $b \neq 0$: Disable all external interrupts (The external interrupt request is removed from all channels)

$\hat{k} = 3$ & $b = 0$: Enable external interrupt on channel $\hat{j}$

$\hat{k} = 3$ & $b \neq 0$: Disable external interrupt on channel $\hat{j}$ (The external interrupt request signal is removed from the channel specified by $\hat{j}$)

The operand address designator, y, bits are not translated for this instruction.

The enable, disable interrupts, f = 66, $\hat{k} = 1$, b = o, $b \neq o$, do not affect the state of the external interrupt lock out flip-flops. Therefore, if all interrupts are disabled and subsequently enabled, the status of the external interrupt lock out flip-flops shall be identical to the status before the lock out all interrupts instruction was executed. Only the instructions pertaining to the external interrupts shall have any effect on the state of the external interrupt lock out flip-flops, that is f = 66, $\hat{k} = 2$ or 3.

67 - Terminate $C\hat{j}$ Output Buffer or all Buffers. - This instruction is variable, depending upon the $\hat{k}$ designator and $\hat{j}$ designator selections.

k = 0: Terminate the output buffer on channel $\hat{j}$.

k = 1: Terminate the external function buffer on channel $\hat{j}$. If the channel specified by $\hat{j}$ is involved with the use of an output register of an inter-computer group, a resume signal is simulated and sent to the register.

k = 2: Terminate all buffers, if an output register is being used for inter-computer communications, a resume signal is simulated and sent to this register.

For all values of $\hat{k}$ no output buffer monitor interrupt shall occur. The index designator, b, and the operand address designator, y, bits are not translated for this instruction.

70 - Repeat. - Initiate the repeat mode or if YL is zero, skip the next instruction. The repeat mode executes the instruction immediately following the repeat instruction YL times. B7 contains the number of executions remaining throughout the repeat mode.

If no skip condition is met for the repeated instruction, the repeat mode terminates and the instruction following the repeated instruction is executed. If the skip condition for the repeated instruction is met, the repeat mode

terminates and the instruction following the repeated instruction is skipped. Following the repeat mode termination, the count remains in B7.

In no way does the repeat mode alter a repeated instruction as stored in memory.

The three low order bits of the r designator (from j of instruction 70) affects the operand indexing as follows:

r = 0: Do not modify the operand address of the repeated instruction after each individual execution.

r = 1: Increase the operand address of the repeated instruction by one after each execution of the repeated instruction.

r = 2: Decrease the operand address of the repeated instruction by one after each execution of the repeated instruction.

r = 3: Repeat the initial B register modification of the repeated instruction before each execution.

r = 4: Do not modify the operand address of the repeated instruction after each individual execution. If the repeated instruction is a replace instruction the operand address is incremented by (B6) for the store portion of the replace instruction.

r = 5: Increase the operand address of the repeated instruction by one after each execution of the repeated instruction. If the repeated instruction is a replace instruction the operand address is incremented by (B6) for the store portion of the replace instruction.

r = 6: Decrease the operand address of the repeated instruction by one after each execution of the repeated instruction. If the repeated instruction is a replace instruction the operand address is incremented by (B6) for the store portion of the replace instruction.

r = 7: Repeat the initial B register modification of the repeated instruction before each execution. If the repeated instruction is a replace instruction the operand address is incremented by (B6) for the store portion of the replace instruction.

NOTE

Instruction 70 j designator establishes the repeat mode r designator since j is transmitted to r.

71 - B Skip On Bj. - Selectively skip the next instruction depending upon a comparison of B register j and YL. If the content of B register j is equal to YL, skip the next instruction in the current sequence and proceed to the following instruction. Clear B register j.

If the content of B register j is not equal to YL, proceed to the next instruction in the sequence in a normal manner. Increase the content of B register j by one.

The branch condition designator, j, is used to designate the selected B register in this instruction and is not available for its normal function. Only the lower order 15 bits of Y are used in these comparisons.

72 - B Jump on Bj. - Selectively execute a program jump to address Y depending upon the value of B register j. If the content of B register j is nonzero, execute a jump in program address to address Y. Reduce the content of B register j by one.

If the content of the B register j is zero, proceed to the next instruction in a normal manner. Do not alter the content of B register j.

The branch condition designator, j, is used to designate the selected B register in this instruction and is not available for its normal function.

73 - Input Buffer on $C\hat{j}$ (without monitor mode). - Establish an input buffer, via input buffer channel $\hat{j}$, to storage with an initial storage address Y. Subsequent to this instruction individual transfers will be executed at a rate determined by an external device. The storage address initially established by this instruction will be advanced by one preceding each individual transfer. The next current address will be maintained throughout the buffer process in the lower order 15 bits of control register 00100 plus $\hat{j}$. This mode will continue until it is superseded by a subsequent initiation or termination of an input buffer via the same input channel or until the higher order half and the lower order half of control register 00100 plus $\hat{j}$ contain equal quantities, whichever occurs first.

k = 0: Store Y in the lower order half of storage location 00100 plus $\hat{j}$ leaving the higher order half undisturbed.

k = 1: Store the lower order 15 bits of Y in the lower order half of storage location 00100 + $\hat{j}$ leaving the higher order half undisturbed.

k = 2: Not permitted.

k = 3: Store Y in storage location 00100 + $\hat{j}$.

74 - Output Buffer on $C\hat{j}$ (without monitor mode). - Establish an output buffer. (If $k \neq 2$, output data buffer; if k = 2 external function buffer) via output buffer channel $\hat{j}$ from initial storage address Y. Subsequent to this instruction the individual transfers will be executed at a rate determined by an external device. The storage address initially established by this instruction will be advanced by one preceding each individual transfer. The next current address will be maintained throughout the buffer process in the lower order 15 bits of control register, if $\hat{k} \neq 2$, 00120 plus j; if $\hat{k} = 2$, 00140 plus $\hat{j}$. This mode will continue until it is superseded by a subsequent initiation or termination of an output buffer via the same output channel or until the higher order half and the lower order half of control register (if $\hat{k} \neq 2$, 00120 plus $\hat{j}$; if $\hat{k} = 2$, 00140 plus $\hat{j}$) contain equal quantities, whichever occurs first.

k = 0: Store Y in the lower order half of storage location 00120 + $\hat{j}$ leaving the higher order half undisturbed.

k = 1: Store the lower order 15 bits of Y in the lower order half of storage location 00120 + $\hat{j}$ leaving the higher order half undisturbed.

k = 2: Store Y in storage location 00140 + $\hat{j}$.

k = 3: Store Y in storage location 00120 + $\hat{j}$.

75 - Input Buffer on $C\hat{j}$ (with monitor mode). - Establish an input buffer, via input buffer channel $\hat{j}$, to storage with an initial storage address Y. Subsequent to this instruction, the individual transfers will be executed at a rate determined by an external device. The storage address initially established by this

instruction will be advanced by one, preceding each individual transfer. The next current address will be maintained throughout the buffer process in the lower order 15 bits of control register 00100 plus $\hat{j}$. This mode will continue until it is superseded by a subsequent initiation or termination of an input buffer via the same input channel or until the higher order half and the lower order half of the control register contain equal quantities, whichever occurs first. The initiation of this input buffer selects the input channel $\hat{j}$ and establishes a buffer monitor on input channel $\hat{j}$. A monitor interrupt to address 00040 + $\hat{j}$ shall follow completion of the buffering operation.

$\hat{k}$ = 0: Store Y in the lower order half of storage location 00100 + $\hat{j}$.

$\hat{k}$ = 1: Store the lower order 15 bits of Y in the lower order half of storage location 00100 + $\hat{j}$ leaving the higher order half undisturbed.

$\hat{k}$ = 2: Not permitted.

$\hat{k}$ = 3: Store Y in storage location 00100 + $\hat{j}$.

76 - Output Buffer on C$\hat{j}$ (with monitor mode). - Establish an output buffer (if $\hat{k} \neq 2$, output data buffer; if $\hat{k} = 2$, external function buffer) via output buffer channel $\hat{j}$ from initial storage address Y. Subsequent to this instruction the individual transfers will be executed at a rate determined by an external device. The storage address initially established by this instruction will be advanced by one preceding each individual transfer. The next current address will be maintained throughout the buffer process in the lower order 15 bits of control register, if $\hat{k} \neq 2$, 00120 plus $\hat{j}$; if $\hat{k} = 2$, 00140 plus $\hat{j}$. This mode will continue until it is superseded by a subsequent initiation or termination of an output buffer via the same output channel or until the higher order half and the lower order half of the control register contain equal quantities, whichever occurs first. The initiation of this output buffer selects the output channel $\hat{j}$ and establishes a buffer monitor on output channel $\hat{j}$. A monitor interrupt (if $\hat{k} \neq 2$ to address 00060 + $\hat{j}$; if $\hat{k} = 2$ to address 00500 + $\hat{j}$) follows the completion of the buffering operation.

$\hat{k}$ = 0: Store Y in the lower order half of storage location 00120 + $\hat{j}$ leaving the higher order half undisturbed.

$\hat{k}$ = 1: Store the lower order 15 bits of Y in the lower order half of storage location 00120 + $\hat{j}$ leaving the higher order half undisturbed.

$\hat{k}$ = 2: Store Y in storage location 00140 + $\hat{j}$.

$\hat{k}$ = 3: Store Y in storage location 00120 + $\hat{j}$.

3-10. BOOTSTRAP PROGRAMS

a. DESCRIPTION.

(1) GENERAL. - The bootstrap program is a sequence of 32 computer instructions fabricated as a permanent part of the computer memory. These instructions can be executed by the computer but cannot be altered by it in any way. The purpose of the bootstrap is to provide automatic loading of other programs into the computer. Programs to be loaded may originate from three different sources:

Magnetic Tape
Another CP-642B/USQ-20(V) Computer
Paper Tape

A separate and different bootstrap program is required for each of these sources.

The bootstrap programs, being limited to 32 instructions, require that the programs to be loaded follow a definite format. In addition, there are a number of restrictions pertaining to the use of certain memory addresses in the computer. Programs to be loaded by the bootstraps should not in general occupy memory addresses between 00000 and 00617. These memory addresses are normally permanently assigned for such special purposes as buffer control, B-register storage, bootstrap program storage, and interrupt entrance and status word storage.

(2) PHYSICAL. - Each computer contains two bootstrap programs. These are located in the Non Destructive Read-Out (NDRO) memory and are assigned addresses 00540 through 00577. Although both bootstrap programs are assigned these same 32 addresses, only one bootstrap can be used at a time. The particular bootstrap program from the NDRO memory is determined by the position of manually operated switches on the computer. These switches are the PROGRAM I and PROGRAM II push button indicator switches located on the computer control console, and the program selector switch located on logic module J35A on chassis A5.

Program I is the magnetic tape bootstrap and is selected by depressing the PROGRAM I indicator switch or by placing the program selector toggle switch in the down position. Program II is either the inter-computer bootstrap or the paper tape bootstrap. (See paragraph 3-10c(2) & (3).) Program II is selected by depressing the PROGRAM II indicator switch or by placing the program selector toggle switch in the up position.

b. BOOTSTRAP OPERATION. - Bootstrap operation normally occurs in one of two ways. The first and most common is inititated when initially loading the computer. The second method is the automatic recovery operation initiated when the computer experiences a fault condition(fault condition is when the computer tries to execute an instruction having an 00 or 77 function code) during the execution of a program.

c. BOOTSTRAP PROGRAMS AND OPERATING PROCEDURES

(1) MAGNETIC TAPE BOOTSTRAP.

(a) PROGRAM. - This bootstrap program will load in the computer memory the first record (properly formatted)from transport 1 of an RD-243 magnetic tape unit. It will checksum the record loaded to ensure correct transmission and then jump to the new program. Summarizing, the magnetic tape bootstrap issues the following commands:

Disable all external interrupts and terminate all buffers. Take control of magnetic tape. Store a Release-Interrupt-Lockout (RIL) instruction in all external interrupt addresses. Store a jump to the magnetic tape interrupt routine in the external interrupt address applicable to the RD-243 magnetic tape unit. Set the return address in B1. Send a rewind command to transport 1 of the RD-243 magnetic tape unit. Enable the external interrupt applicable to the RD-243 magnetic tape unit. Upon receipt of the external interrupt from the RD-243 magnetic tape unit indicating that transport 1 is rewinding, initiate an input buffer and send a read command to transport 1 of the RD-243 magnetic tape unit. Set the return address in B1. Upon receipt of the external interrupt from the RD-243 magnetic tape unit indicating that the first record on tape

transport 1 has been read, the record loaded is checked against the checksum to ascertain correct transmission. If a checksum error is found, the bootstrap is re-executed. If the checksum is correct, jump to the program entrance address. Each external interrupt from the RD-243 magnetic tape unit is checked. If the interrupt indicates an error has occurred, the bootstrap is re-executed.

(b) OPERATING PROCEDURES. - The magnetic tape to be read into memory must be manually positioned on transport number 1 of the magnetic tape unit.

MANUAL INITIATION

The computer must be operating in the stop condition.

Step 1. - Depress the PROGRAM I push button indicator switch.

Step 2. - Depress the LOAD MODE push button indicator switch.

Step 3. - Operate the START-STEP/RESTART switch to the START-STEP position.

The computer will immediately begin reading the magnetic tape and, upon completion, will jump to the program just loaded.

AUTOMATIC INITIATION

Step 1. - Place the AUTOMATIC RECOVERY toggle switch in the up position.

Step 2. - Place the program selector switch, located on logic module J35A on chassis A5, to the down position.

If the computer attempts to execute a 00 or 77 function code the bootstrap operation will be initiated.

The proper format for the RD-243 magnetic tape unit is shown in figure 3-7. The program is listed in figure 3-8 and flow-charted in figure 3-9. The program as listed is for use on channel 11 (octal). It may be used on any channel by changing the operators TAPE and CHAN appropriately. The bootstrap requires the following memory locations which are outside the loaded programs:

ADDRESS	PURPOSE
00020-00037	Interrupt Preset
00107	Storage for program entrance address and first address
00110	Storage for checksum

(2) INTER-COMPUTER BOOTSTRAP.

(a) PROGRAM. - This bootstrap program will load a program into memory from one of three other computers via an inter-computer channel. It will checksum the program to ensure correct transmission and, upon completion, it will jump to the program entrance address. The use of the inter-computer bootstrap program presupposes that the other computer contains a program capable of communicating with the bootstrap program. Summarizing, the inter-computer bootstrap issues the following commands:

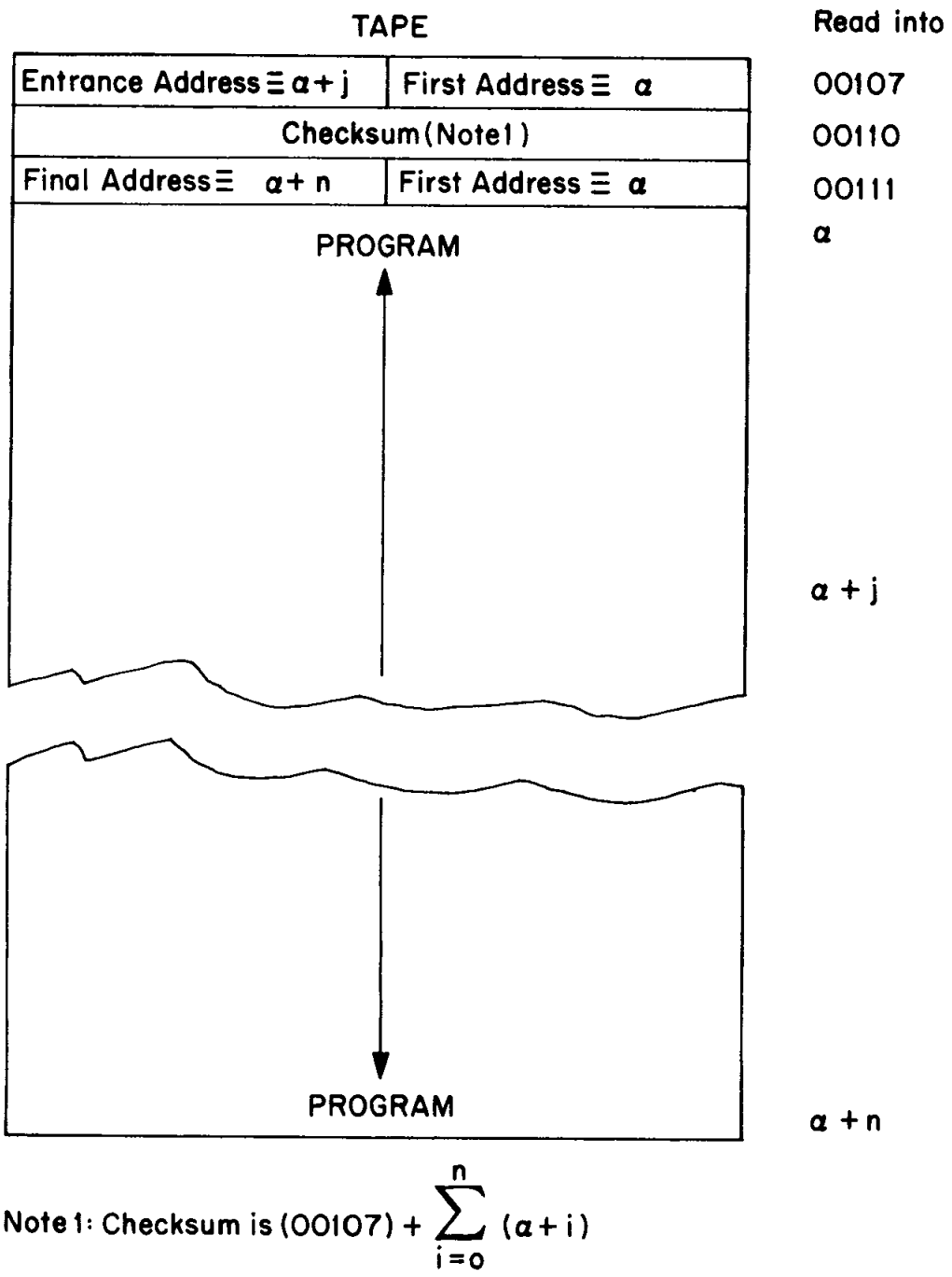


Figure 3-7. Magnetic Tape Bootstrap Program Record Format

ADDRESS	INSTRUCTION WORD	CODE	FUNCTION
00540	66027 00013	SIL-EX·ALL	LOCKOUT EXTERNAL INTERRUPTS, CONSTANT
00541	67020 10100	TERM·ALL	TERMINATES ALL BUFFERS,CONSTANT
00542	13570 00540	EX-FCT·TAPE·W(540)	TAKE CONTROL OF TAPE UNIT
00543	10000 60000	ENT·Q·60000	STORE RIL INSTRUCTION IN
00544	70100 00020	RPT·20·ADV	ALL EXTERNAL INTERRUPT
00545	14020 00020	STR·Q·U(20)	ENTRANCE ADDRESSES
00546	10030 00574	ENT·Q·W(574)	
00547	14030 00033	STR·Q·W(20+CHAN)	SET UP TAPE EXTERNAL INTERRUPT ADDRESS
00550	12100 00554	ENT·B1·554	SET UP INTERRUPT RETURN ADDRESS
00551	13470 00541	EX-FCT·TAPE·W(541)	SEND REWIND COMMAND
00552	66470 14420	RIL-EX·TAPE	ENABLE TAPE INTERRUPT,CONSTANT
00553	60100 00553	RILJP·553	WAIT FOR INTERRUPT
00554	73470 00560	IN·TAPE·W(560)	INITIATE INPUT BUFFER
00555	13470 00552	EX-FCT·TAPE·W(552)	SEND READ COMMAND
00556	12100 00560	ENT·B1·560	SET UP INTERRUPT RETURN ADDRESS
00557	61000 00557	JP·557	WAIT FOR INTERRUPT
00560	12110 00107	ENT·B1·L(100+CHAN-2)	FIRST ADDRESS TO B1
00561	11030 00110	ENT·A·W(100+CHAN-1)	CHECKSUM TO A
00562	21030 00107	SUB·A·W(100+CHAN-2)	SUBTRACT ENTRANCE ADDRESS WORD
00563	21031 00000	SUB·A·W(B1)	SUBTRACT PROGRAM WORD
00564	71120 00111	BSK·B1·U(100+CHAN)	IS CHECK SUMMING COMPLETED
00565	61000 00563	JP.563	NO, CONTINUE
00566	60500 00572	JP·572·ANOT	YES, JUMP IF CHECKSUM ERROR
00567	61020 00107	JP·U(100+CHAN-2)	JUMP TO PROGRAM ENTRANCE ADDRESS
00570	11760 00533	ENT·A·UX(520+CHAN)·ANEG	STATUS TO A AND TEST FOR IMPROPER
00571	52400 00777	SEL·CL·777·AZERO	TEST STATUS FOR READ ERROR
00572	61000 00540	JP·540	ERROR, START OVER
00573	60101 00000	RILJP·B1	STATUS GOOD, RETURN TO B1
00574	61000 00570	JP·570	TAPE INTERRUPT INSTRUCTION
00575	00000 00000	0·0	SPARE
00576	00000 00000	0·0	SPARE
00577	00000 00000	0·0	SPARE

NOTE: TAPE MEANS·C11
CHAN EQUALS·11

001 011 100 1

Figure 3-8. Magnetic Tape Bootstrap Program

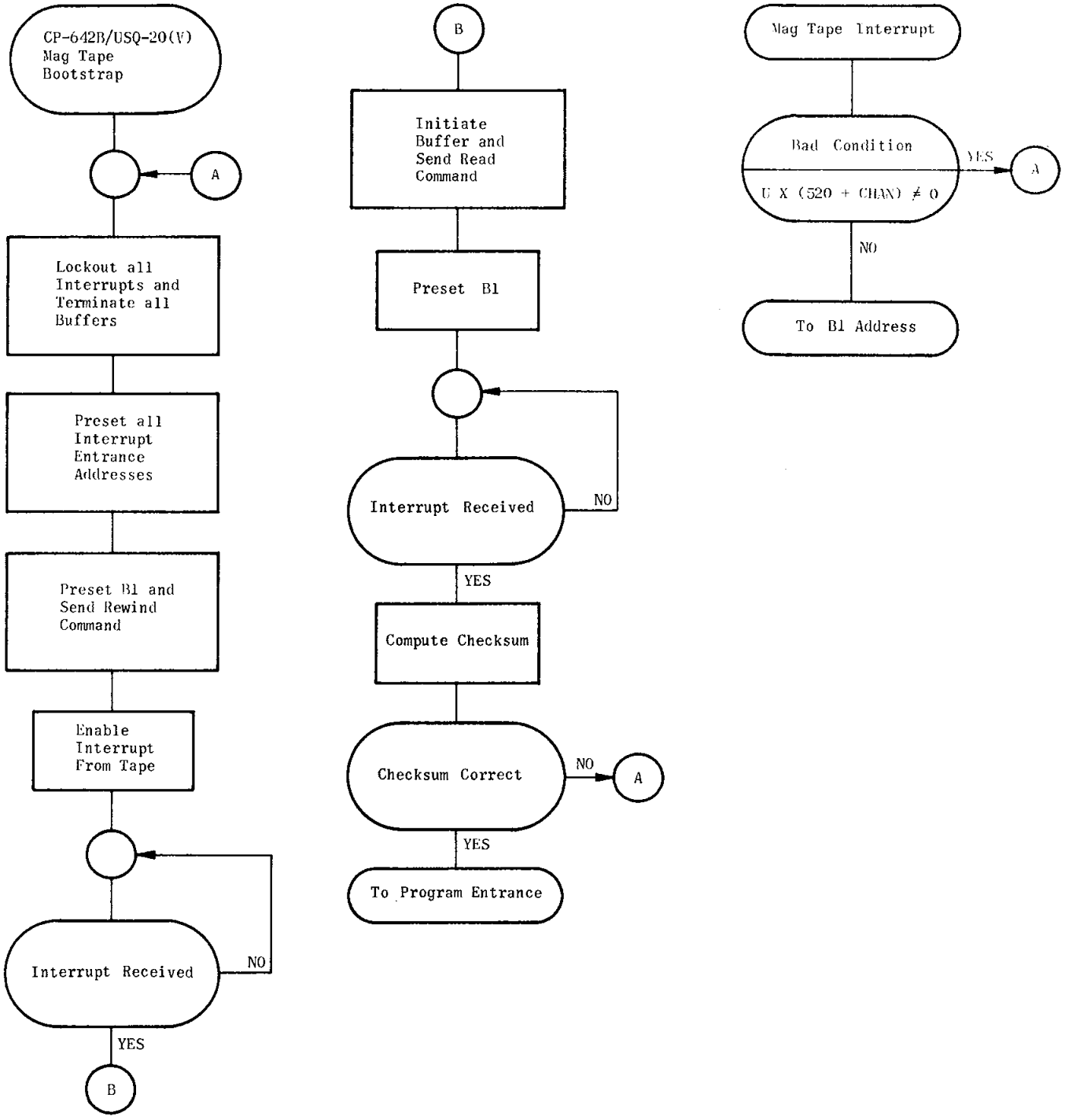


Figure 3-9. Magnetic Tape Bootstrap Program Flow Chart

Disable all interrupts and terminate all buffers. Store a Release-Interrupt-Lockout (RIL) instruction in all interrupt addresses. Send an external function on the three inter-computer channels requesting any computer that is on-line to reply. Monitor the interrupt word storage address to ascertain if any computer replied. Terminate all buffers and clear upper fifteen bits of B7. Send an external function on the three inter-computer channels requesting program transfer from the computer that replied first. Initiate an input buffer on the three inter-computer channels. When the input buffer terminates for the computer from which program transfer was requested, the loading is complete. Terminate all buffers. The program loaded is checked against the checksum to ascertain correct transmission. If a checksum error is found, the bootstrap is re-executed. If the checksum is correct, jump to the program entrance address.

(b) OPERATING PROCEDURES.

MANUAL INITIATION

The computer must be operating in the stop condition.

Step 1. - Depress the PROGRAM II push button indicator switch.

Step 2. - Depress the LOAD MODE push button indicator switch.

Step 3. - Operate the START-STEP/RESTART switch to the START-STEP position.

The bootstrap computer will request and accept a program from another computer and jump to that program.

AUTOMATIC INITIATION

Step 1. - Place the AUTOMATIC RECOVERY toggle switch in the up position.

Step 2. - Place the program selector switch, located on logic module J35A on chassis A5, in the up position.

If the computer attempts to execute a 00 or 77 function code the bootstrap operation will be initiated.

The inter-computer request-reply logic is summarized in table 3-5. The format of the inter-computer program record is shown in figure 3-10. The real-time clock must be enabled in the bootstrap computer. The program as listed is for use on channels 0, 1, and 2, but it may be used on any three consecutive inter-computer channels provided the operators BC, IC0, IC1 and IC2 are changed appropriately. The bootstrap requires the following memory locations which are outside the loaded program:

ADDRESS	PURPOSE
00020-00617	Interrupt Preset

The inter-computer bootstrap program is listed in figure 3-11 and flow-charted in figure 3-12.

(3) PAPER TAPE BOOTSTRAP.

(a) PROGRAM. - This bootstrap will load into memory a program from a manually positioned paper tape of the proper format and, upon completion, jump to that program

with a STOP 7 option. The paper tape bootstrap is installed in the computer as Program II (instead of the inter-computer bootstrap). The paper tape bootstrap is only for diagnostic testing during the installation and checkout of a new data processing system. When the installation is complete and the system is operating properly this bootstrap is removed and the inter-computer bootstrap is installed in its place. The paper tape bootstrap issues the following commands:

Disable all interrupts and terminate all buffers. Enable the paper tape reader. Store a Release-Interrupt-Lockout (RIL) instruction in all external and internal monitor interrupt addresses. Look for the beginning of information on the paper tape. Read the first and last address from the paper tape. Read each program data word from the paper tape and store it in memory as defined by the first and last address. When all program data words and the two checksums have been loaded into memory, the loading is complete. Display the first and last address in the Q register and jump to the first address. Stop of STOP 7 toggle switch is in the up position.

(b) OPERATING PROCEDURES. - The paper tape to be read into memory must be manually positioned on the paper tape reader.

MANUAL INITIATION

The computer must be operating in the stop condition.

Step 1. - Depress the PROGRAM II push button indicator switch.

Step 2. - Depress the LOAD MODE push button indicator switch.

Step 3. - Operate the START-STEP/RESTART switch to the START-STEP position.

The computer will immediately begin reading the paper tape. If the STOP 7 toggle switch is in the down position the computer will jump to the input program upon completion of the loading operation. If the STOP 7 switch is in the up position the computer will stop when the program is loaded. Operating the START-STEP/RESTART switch to the START-STEP position will cause the computer to jump to the input program.

AUTOMATIC INITIATION

Step 1. - Place the AUTOMATIC RECOVERY toggle switch in the up position.

Step 2. - Place the program selector switch, located on logic module J35A on chassis A5, in the up position.

If the computer attempts to execute a 00 or 77 function code the bootstrap operation will be initiated. As above, the position of the STOP 7 switch will determine whether the computer stops or transfers to the input program upon completion of the loading operation.

The proper format for the paper tape is the normal bioctal format consisting of leader, bioctal tape identifier (76), first address, last address, contents of all address (from first to last address), checksum of all upper half words, checksum of all lower half words and trailer (see figure 3-13). Note that the bioctal tape identifier must be the first non-zero frame on the tape or the tape must be so positioned. The last address should not exceed 77775 because two locations beyond

TABLE 3-5. INTER-COMPUTER REQUEST-REPLY LOGIC

WORD FORMAT	SENT BY	RECEIVED BY	MEANING
66010.XXXXX	Bootstrap Computer	All computers	Please reply if you are there
00000.XXXXX	All Computers capable of replying	Bootstrap Computer	I am here
00000.00000	Bootstrap Computer	All computers	Will the computer connected to the Bootstrap computers Channel 0 please send program.
or 00000.00001	Bootstrap Computer	All computers	Will the computer connected to the Bootstrap computers Channel 1 please send program.
or 00000.00002	Bootstrap Computer	All computers	Will the computer connected to the Bootstrap computers Channel 2 please send program.

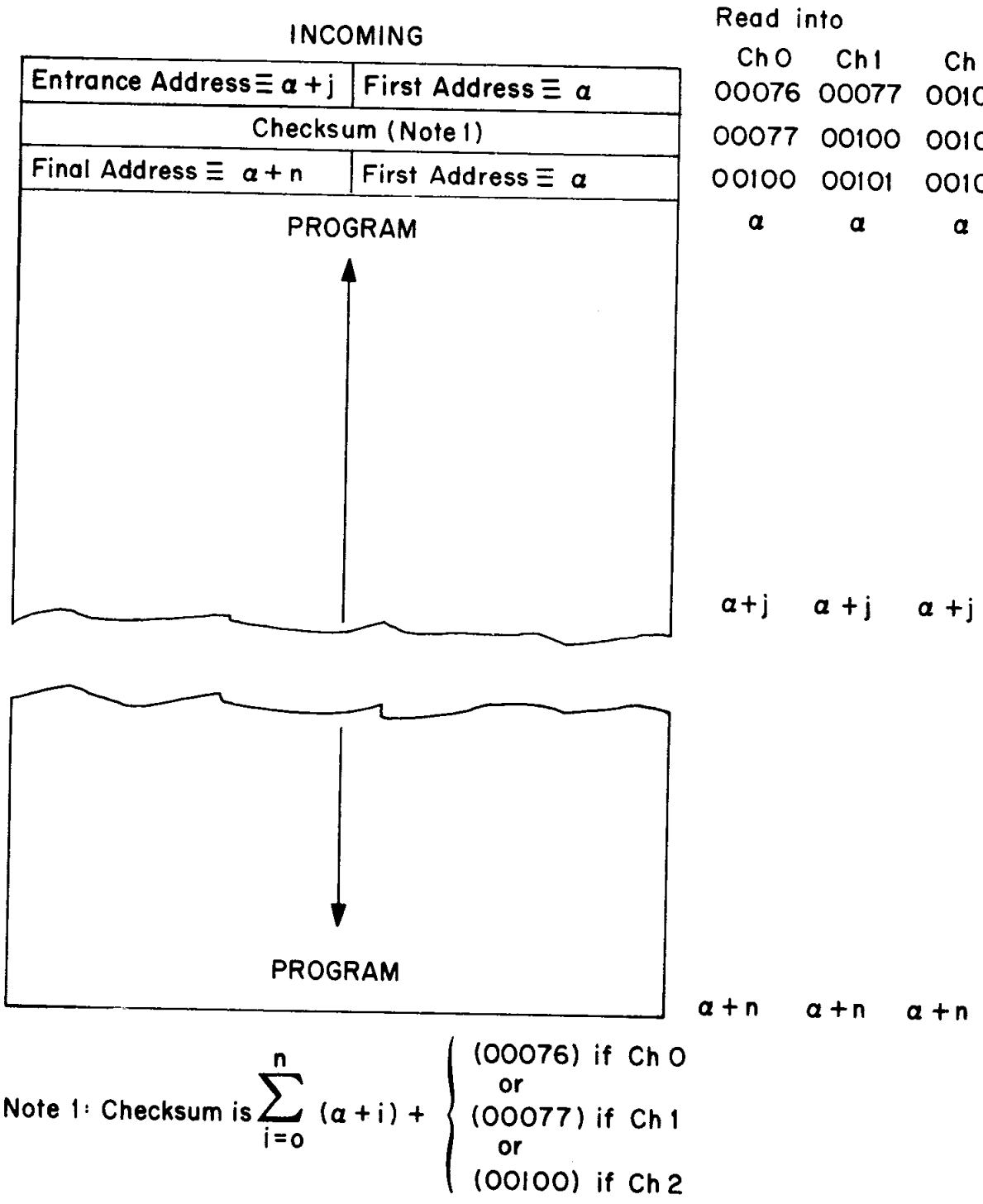


Figure 3-10. Inter-Computer Bootstrap Program Record Format

ADDRESS	INSTRUCTION WORD	CODE	FUNCTION
00540	66017 00076	SIL·ALL	LOCKOUT ALL INTERRUPTS,CONSTANT
00541	67020 00077	TERM·ALL	TERMINATE ALL BUFFERS,CONSTANT
00542	10000 60000	ENT·Q·60000	STORE RIL INSTRUCTION
00543	70100 00600	RPT·600·ADV	IN ALL ADDRESSES
00544	14020 00020	STR·Q·U(20)	FROM 00020 to 00617
00545	66010 00100	RIL·ALL	ENABLE ALL INTERRUPTS,CONSTANT
00546	13020 00545	EX-COM·IC0·W(545)	INTERROGATE
00547	13060 00545	EX-COM·IC1·W(545)	THE THREE SUCCESSIVE
00550	13120 00545	EX-COM·IC2·W(545)	INTERCOMPUTER CHANNELS
00551	70200 00003	RPT·3·BACK	DID ANY
00552	11427 00517	ENT·A·U(BC+517+B7).AZERO	COMPUTER REPLY
00553	61000 00546	JP·546	NO, RE-INTERROGATE
00554	67020 00000	TERM·ALL	YES,(B7)INDICATES REPLYING COMPUTER
00555	16020 00167	CL·U(167)	CLEAR UPPER HALF OF B7 STORAGE ADDRESS
00556	13020 00167	EX-COM·IC0·W(167)	SEND PROGRAM REQUEST, (B7),
00557	13060 00167	EX-COM·IC1·W(167)	ON THE THREE SUCCESSIVE
00560	13120 00167	EX-COM·IC2.W(167)	INTERCOMPUTER CHANNELS
00561	73030 00540	IN·IC0·W(540)	INITIATE INPUT BUFFER
00562	73070 00541	IN·IC1·W(541)	ON THE THREE SUCCESSIVE
00563	73130 00545	IN·IC2·W(545)	INTERCOMPUTER CHANNELS
00564	11027 00100	ENT·A·U(100+B7)	IS INPUT BUFFER ON REQUESTED CHANNEL
00565	21717 00100	SUB·A·L(100+B7)·ANEG	TERMINATED (LOWER GREATER THAN UPPER)
00566	61000 00564	JP·564	NO,CONTINUE TERMINATION CHECK
00567	67020 00000	TERM·ALL	YES,TERMINATE OTHER TWO INPUT BUFFERS
00570	12617 00076	ENT·B6·L(100+B7-2)	FIRST ADDRESS TO B6
00571	11037 00077	ENT·A·W(100+B7-1)	CHECKSUM TO A
00572	21037 00076	SUB·A·W(100+B7-2)	SUBTRACT ENTRANCE ADDRESS WORD
00573	21036 00000	SUB·A·W(B6)	SUBTRACT PROGRAM WORD
00574	71627 00100	BSK·B6·U(100+B7)	IS CHECK SUMMING COMPLETED
00575	61000 00573	JP·573	NO, CONTINUE
00576	60500 00540	JP·540·ANOT	YES, JUMP IF CHECKSUM ERROR
00577	61027 00076	JP·U(100+B7-2)	JUMP TO PROGRAM ENTRANCE ADDRESS

NOTE: BC EQUALS·0
ICO MEANS·C0
IC1 MEANS·C1
IC2 MEANS·C2

Figure 3-11. Inter-Computer Bootstrap Program

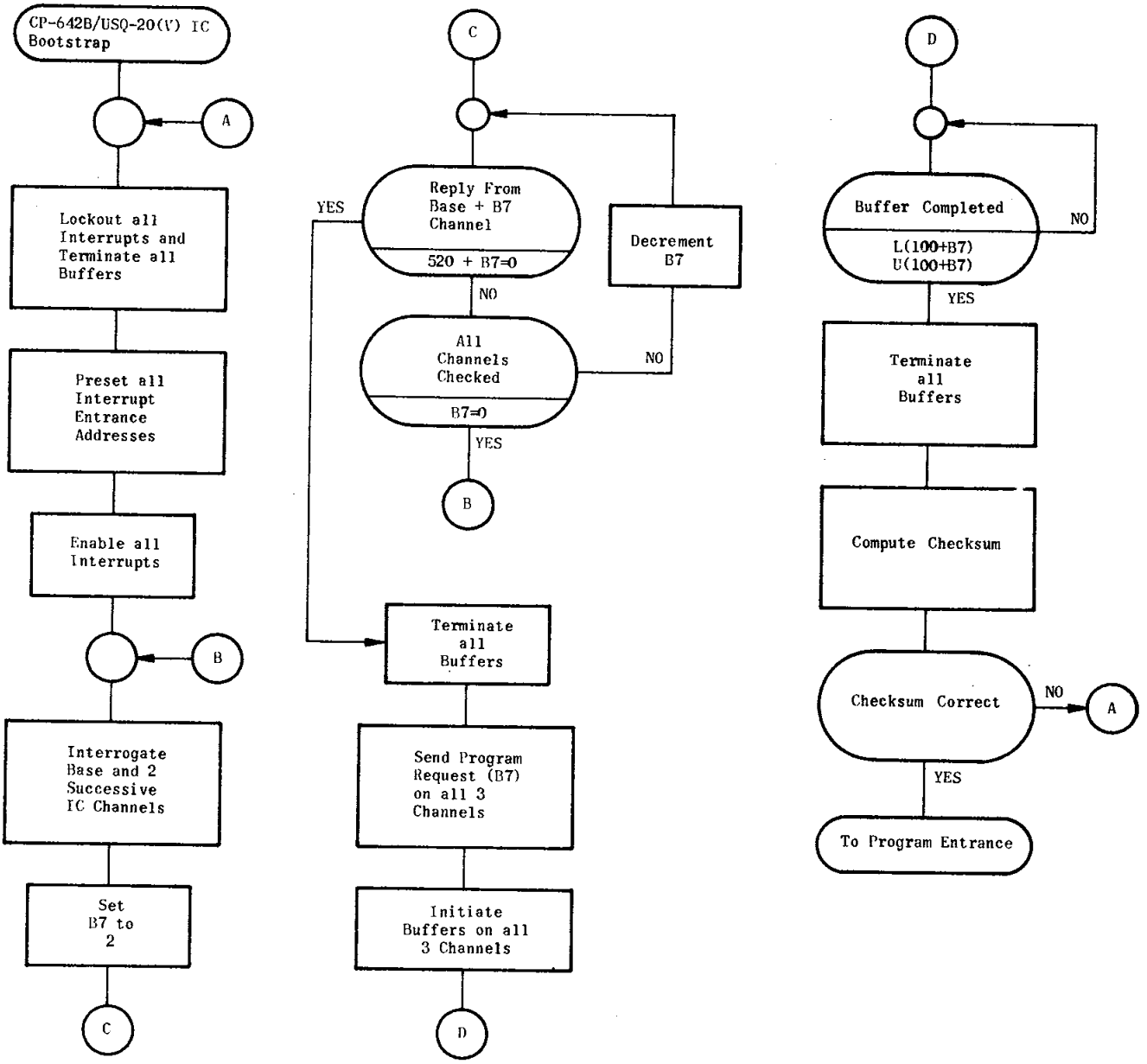


Figure 3-12. Inter-Computer Bootstrap Program Flow Chart

the last address will be used for storage of the checksums. The program loaded should ascertain proper loading by checking the checksum. The address limits to be checked are in the Q register when the program is entered; the first address in Q lower and the last in Q upper. The checksum words are stored in the two locations beyond the last address as explained above. The program as listed is for Channel 4 but the program can be used on any channel if the instructions which contain the operators PT and PTCHAN are changed appropriately.

The paper tape bootstrap program is flow-charted in figure 3-14 and listed in figure 3-15. This bootstrap requires the following memory locations which are outside the loaded program.

ADDRESS	PURPOSE
00104	Input buffer control word and input store
00124	Storage for first and last address
Last address +1	Storage for checksum of all upper half words
Last address +2	Storage for checksum of all lower half words

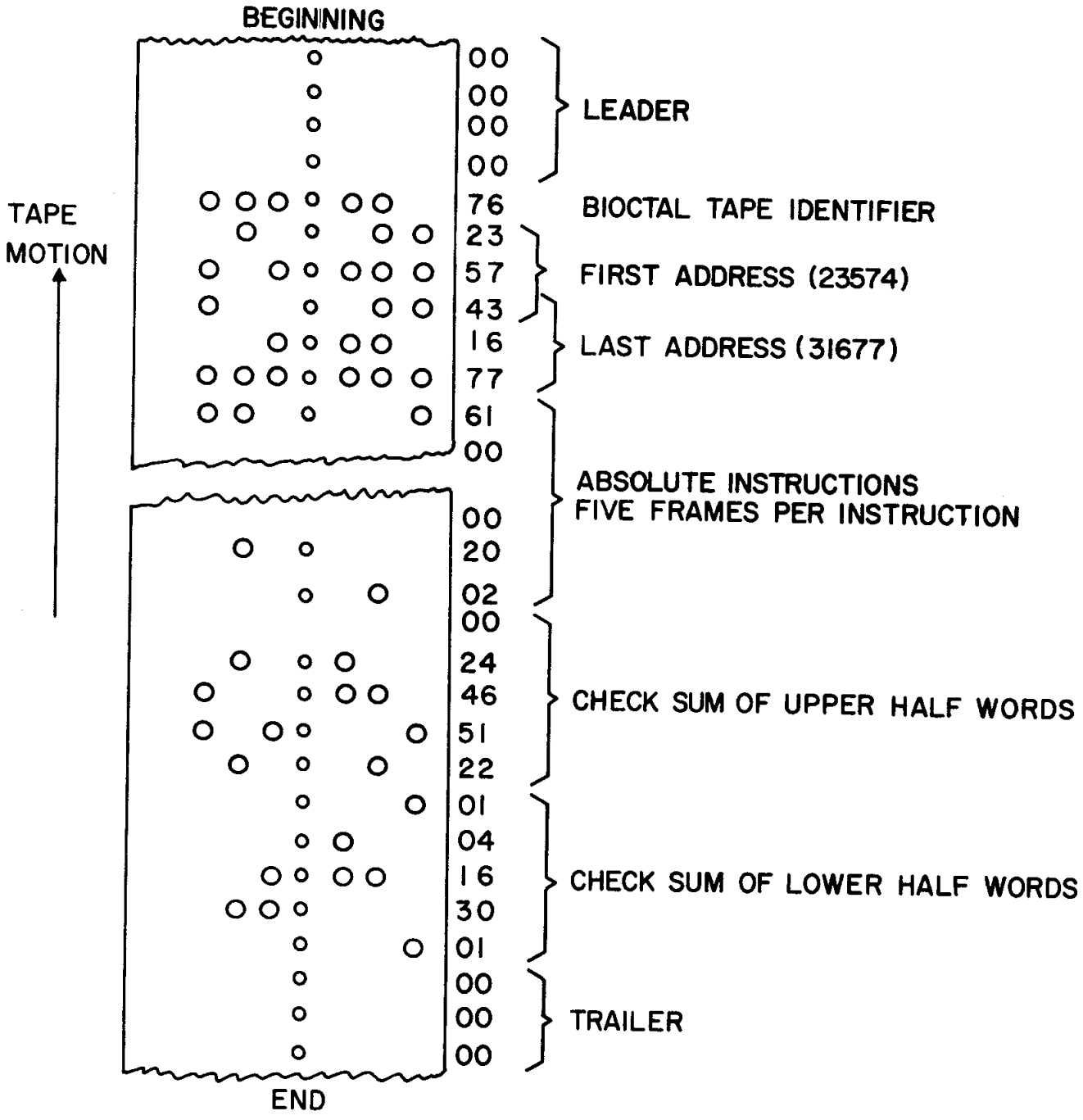


Figure 3-13. Example of Bioctal Paper Tape

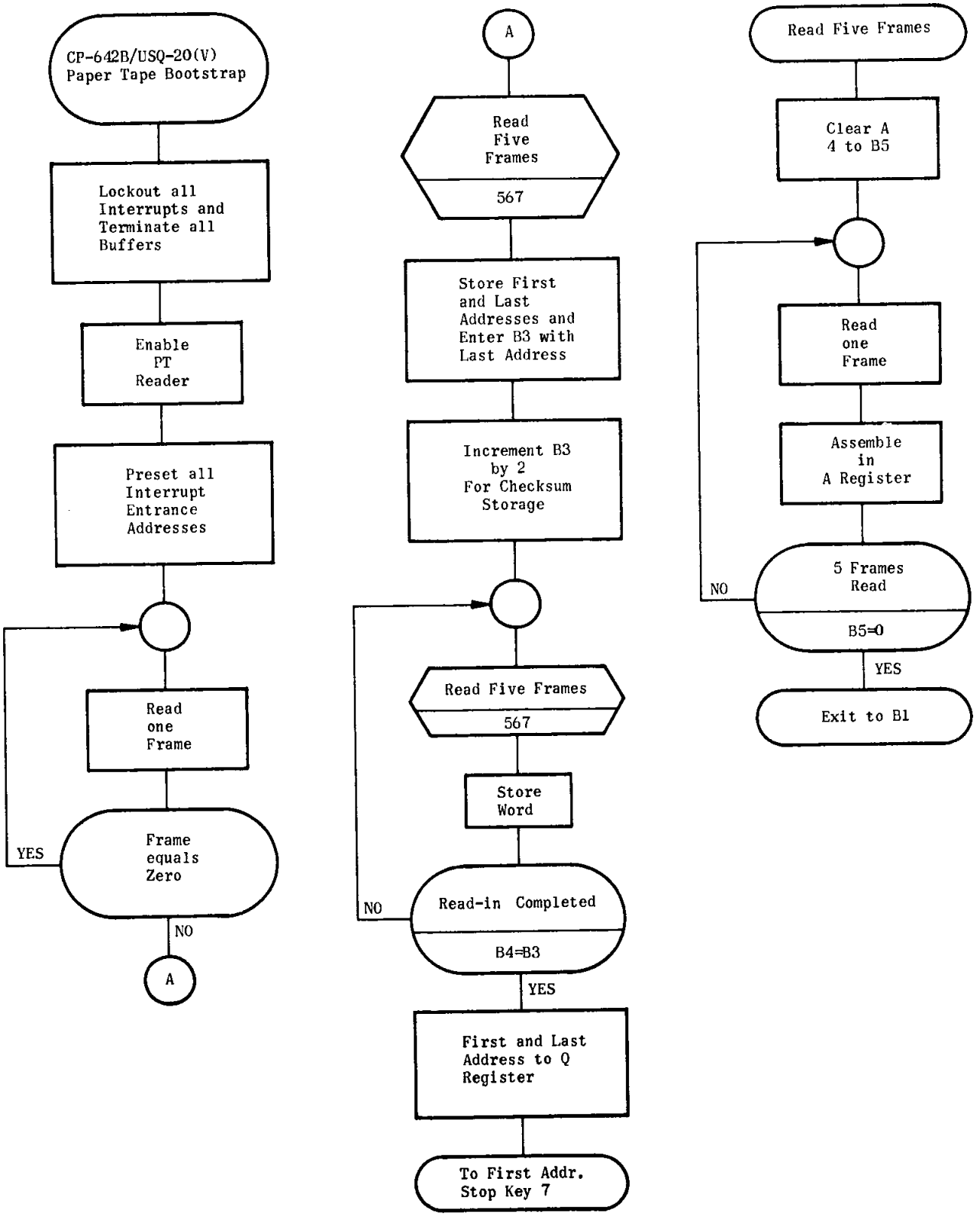


Figure 3-14. Paper Tape Bootstrap Program Flow Chart

ADDRESS	INSTRUCTION WORD	CODE	FUNCTION
00540	66011 00040	SIL·ALL	LOCKOUT ALL INTERRUPTS,CONSTANT
00541	67020 00000	TERM·ALL	TERMINATE ALL BUFFERS
00542	13230 00540	EX-FCT·PT·W(540)	ENABLE READER
00543	10000 60000	ENT·Q·60000	STORE RIL INSTRUCTION
00544	70100 00060	RPT·60.ADV	IN ALL ADDRESSES
00545	14020 00020	STR·Q·U(20)	FROM 00020 TO 00077
00546	73230 00577	IN·PT·W(577)	INITIATE ONE FRAME INPUT BUFFER
00547	62200 00547	JP·547·PT·ACTIVEIN	WAIT FOR BUFFER TO COMPLETE
00550	11010 00104	ENT·A·L(100+PTCHAN)	EXAMINE FRAME
00551	60400 00546	JP·546·AZERO	JUMP IF STILL ON LEADER
00552	12100 00554	ENT·B1·554	SET UP RETURN ADDRESS
00553	61000 00567	JP·567	READ AND ASSEMBLE 5 FRAMES
00554	15030 00124	STR·A·W(120+PTCHAN)	SAVE FIRST AND LAST ADDRESS
00555	12310 00124	ENT·B3·L(120+PTCHAN)	LAST ADDRESS TO B3
00556	12303 00002	ENT·B3·2+B3	LAST ADDRESS + 2 TO B3
			(CHECKSUM STORAGE)
00557	12420 00124	ENT·B4·U(120+PTCHAN)	FIRST ADDRESS TO B4
00560	12100 00562	ENT·B1·562	SET UP RETURN ADDRESS
00561	61000 00567	JP·567	READ AND ASSEMBLE 5 FRAMES
00562	15034 00000	STR·A·W(B4)	STORE ASSEMBLED WORD IN MEMORY
00563	71403 00000	BSK·B4·B3	IS READ-IN COMPLETED
00564	61000 00560	JP·560	NO, CONTINUE
00565	10030 00124	ENT·Q·W(120+PTCHAN)	YES,FIRST AND LAST ADDRESS TO Q
00566	61720 00124	JP·U(120+PTCHAN)·STOP 7	JUMP TO FIRST ADDRESS, STOP IF
			KEY 7 SET
00567	12500 00004	ENT·B5·4	SET UP TO READ 5 FRAMES
00570	11000 00000	CL·A	CLEAR ASSEMBLY REGISTER
00571	73230 00577	IN·PT·W(577)	INITIATE ONE FRAME INPUT BUFFER
00572	62200 00572	JP·572·PT·ACTIVEIN	WAIT FOR BUFFER TO COMPLETE
00573	06000 00006	LSH·A·6	MAKE ROOM FOR FRAME RECEIVED
00574	50010 00104	SEL·SET·L(100+PTCHAN)	SET FRAME IN ASSEMBLY REGISTER
00575	72500 00571	BJP·B5.571	JUMP IF LESS THAN 5 FRAMES READ
00576	61001 00000	JP.B1	RETURN TO B1
00577	00104 00104	U-TAG·100+PTCHAN·100+PTCHAN	

NOTE: PT MEANS·C4
PTCHAN EQUALS·4

Figure 3-15. Paper Tape Bootstrap Program