B 1900 SERIES
CENTRAL SYSTEM

TECHNICAL MANUAL
VOLUME 1:

OPERATION and MAINTENANCE

Burroughs

FIELD ENGINEERING

F.E. Dist. Code BB

Printed in U.S. America

July 1980

For Library Binder 304A
Form 1127388

Burroughs believes that the information described in this manual is accurate and reliable, and much care has been taken in its preparation. However, no responsibility, financial or otherwise, is accepted for any consequences arising out of the use of this material. The information contained herein is subject to change. Revisions may be issued to advise of such changes and/or additions.

Correspondence regarding this document should be addressed directly to Burroughs Corporation, P.O. Box 4040, El Monte, California 91734, Attn: Publications Department, TIO–West.

LIST OF EFFECTIVE PAGES

Page	Issue
Title	Original
ii	Original
iii	Original
iv	Blank
v thru xi	Original
xii	Blank
1-1 thru 1-71	Original
1-72	Blank
2-1 thru 2-8	Original
3-1 thru 3-4	Original
4-1 thru 4-13	Original
4-14	Blank

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont)

TABLE OF CONTENTS (Cont)

TABLE OF CONTENTS (Cont)

TABLE OF CONTENTS (Cont)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1
FUNCTION AND OPERATION

INTRODUCTION

This FETM provides information on function and operation (Section 1), installation (Section 2), documentation and components (Section 3), and maintenance techniques (Section 4) for B 1900 central systems.

SYSTEM DESCRIPTION

Two basic system series are included: the B 1955 and the B 1905. Each includes a microprogrammed processor with a local, high-speed memory (Cache), a dynamic main memory (S-Memory), and a soft I/O subsystem. With the exception of a few circuits that contain discrete elements, system circuitry is entirely comprised of TTL (transistor-transistor logic) integrated circuits. TTL/CTL conversion is provided for interfacing to units with CTL circuitry (primarily I/O controls).

These systems utilize "low boy" cabinets. The top of the cabinet functions as a console, with built-in keyboard and CRT display for the operator display terminal (ODT) function. Both series include an operator control panel (Op panel) and a diagnostics/maintenance control panel (D/M panel). The D/M panel includes a magnetic tape cassette drive. An expansion cabinet may be added to the B 1955 for additional I/O controls on independent backplanes.

Figure 1-1 shows a B 1905 central system mainframe; Figure 1-2 shows a B 1955 with an expansion cabinet.

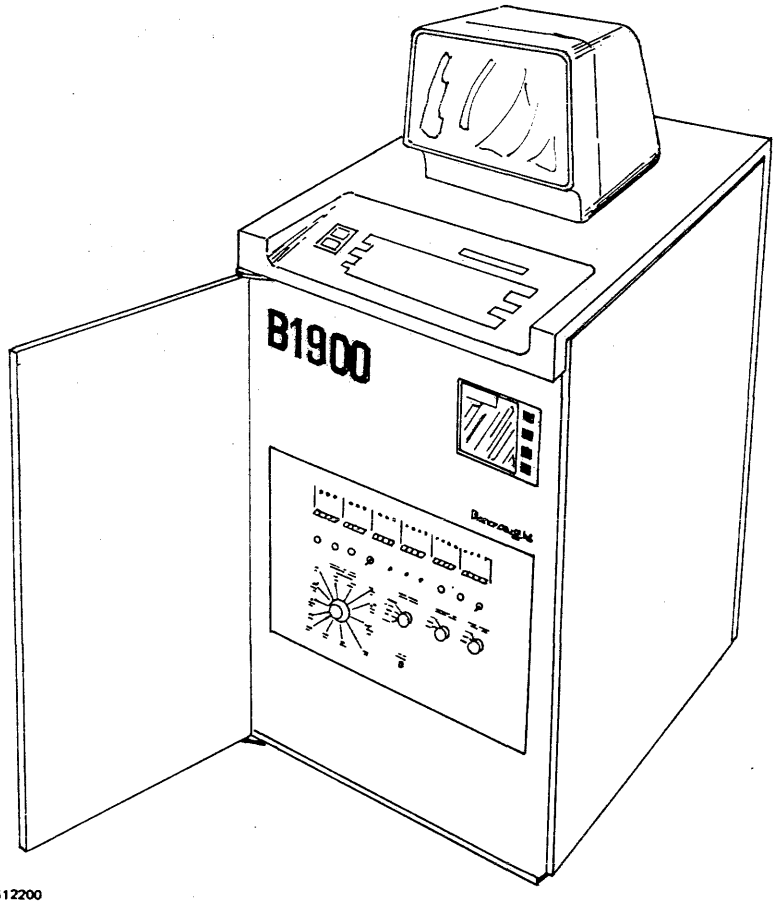


Figure 1-1. B 1905 Central System

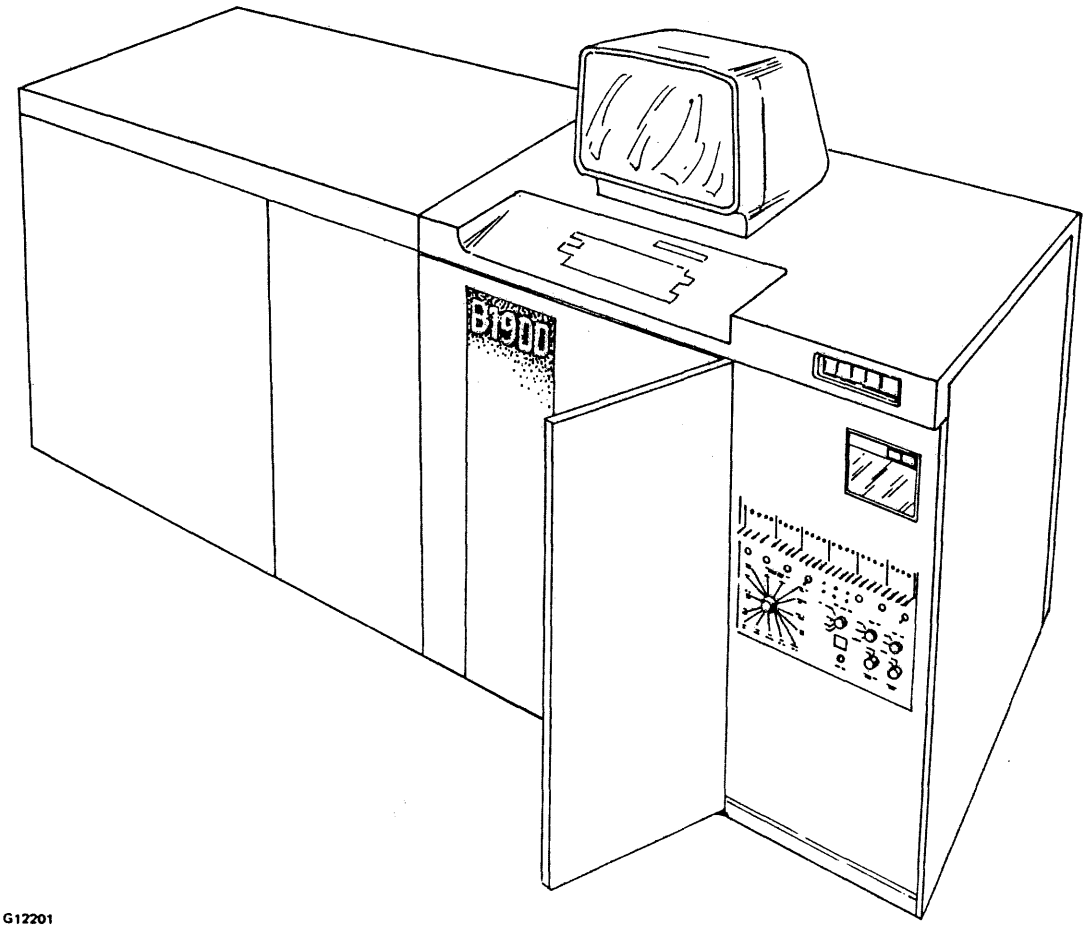


Figure 1-2. B 1955 Central System with Expansion Cabinet

Features and Options

1. Variable micrologic design, which allows the processing environment to be optimized (by software) for a variety of programming languages. Forty distinct microinstructions are available.
2. A hardware-managed Cache memory for in-processor storage of microinstructions (micros). The management scheme minimizes the need for the processor to access main memory for micros. Cache memory capacity is 4K words. (Each word is a 16-bit micro.)
3. A three-phase microprogram handling structure in which the fetch, decode, and execute functions are handled separately but concurrently. This is called the micro pipeline.

   Associated with this structure is a nanoregister for storage of nanoinstructions (control signals, called "nanos") for the execute phase.

4. Stored logic in the form of programmable read-only memories.
5. Modular IC main memory (S-Memory) with error detection and correction capability. S-Memory is expandable in 128-KB increments to a maximum of 512 KB in the B 1905 and 2 MB in the B 1955.

6. An independent, expandable I/O subsystem ("soft I/O") employing fully buffered control logic. The following types of peripheral devices are available:

   card (80 & 96 column)
   disk (cartridge, pack, and head-per-track)
   magnetic tape
   line printer
   reader/sorter
   data communications

   The B 1905 handles up to six I/O controls; the B 1955 handles up to fifteen. Both series include an integrated disk drive electronics controller (DDEC).

7. An optional host adapter that allows independent access to S-Memory by certain devices such as multiline data communications subsystems. This is called the "hard I/O" subsystem.

8. The following program products:

   A Master Control Program (MCP) for management of system resources.

   Compilers for COBOL, FORTRAN, RPG, BASIC, SDL, MIL, NDL, and UPL.

   Special emulator and translator programs for direct execution or machine translation of programs written in code for certain other systems.

9. Multiprogramming under MCP control and a virtual memory that is bit-addressable, allowing code compaction of variable-length operands.
10. A 4-MHz (both series) or 6-MHz (B 1955) clock rate.

Subsystems

Four subsystems comprise the basic (direct-connect) B 1900 central system: (1) the processor, (2) the memory subsystem (S-Memory), (3) the input/output subsystem (I/O), and (4) the console (control panels). The relationships among these subsystems are shown in Figure 1-3. B 1900 central systems may also be configured in a port-connect version. This configuration includes a host adapter with four ports. Figure 1-4 is a block diagram of a port-connect subsystem utilizing two of these ports, one for the processor and the other for the hard I/O subsystem which, in this example, consists of a multiline control and its associated units.

The processor contains the arithmetic and control portions of the central system as well as the associated registers and storage devices, including Cache.

The console, actually part of the processor, is more conveniently discussed as a separate subsystem.

The S-Memory subsystem includes logic for transferring information between the processor (or any other memory-accessing unit) and the storage boards. In a direct-connect system, the memory-accessing unit is the processor. I/O transfers to and from memory are through the processor. In a port-connect system, the memory-accessing unit is the host adapter. The processor is one of several possible port devices that contend for memory; therefore, transfers between the I/O and memory subsystems are through the processor and the host adapter.

The basic I/O subsystem utilizes individual controls to interface directly with the processor. The optional hard I/O subsystem utilizes multiline controls to the host adapter, which communicates with the processor and S-Memory.

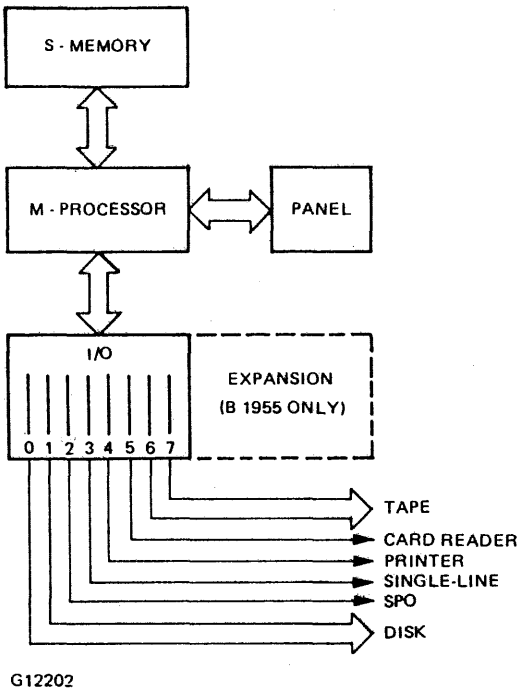


Figure 1-3. Block Diagram: B 1900 Central System, Direct Connect

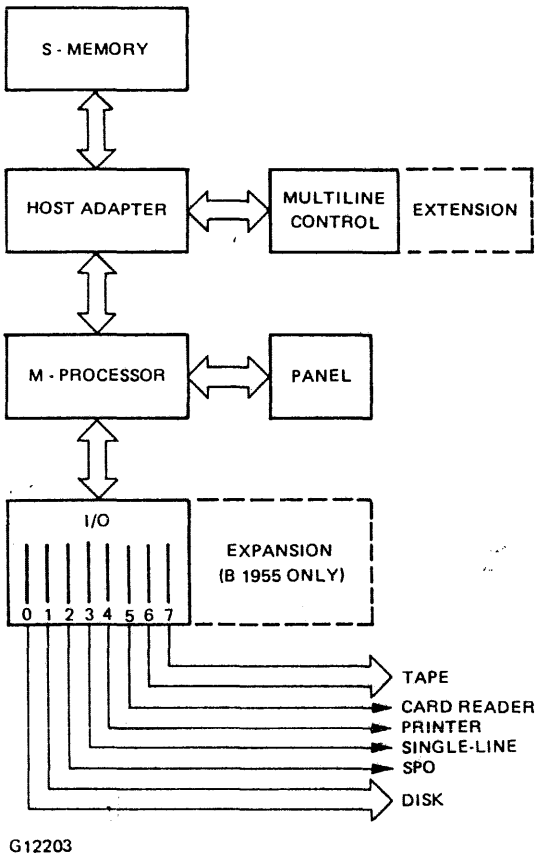


Figure 1-4. Block Diagram: B 1900 Central System, Port Connect

CONTROL PANELS

The B 1900 is provided with two control panels, one for normal operational use and the other for diagnostic and maintenance purposes. Figures 1-5 and 1-6 show these panels.

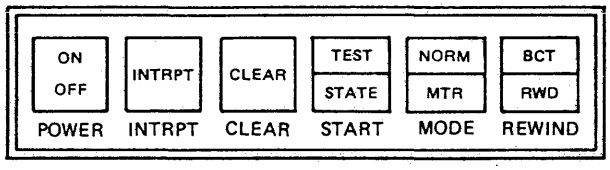


Figure 1-5. B 1900 Operator Panel

Operator Control Panel

The operator control panel (Op panel) includes six indicator push buttons. In the B 1955, these are located in an insert strip in the right front corner of the mainframe cabinet. In the B 1905, two of these push buttons (POWER, INTRPT) are on the left side of the ODT keyboard and the other four are to the right of the cassette unit, on the D/M panel. All the push button and indicator functions are described in the following subsections.

POWER Push Button, ON/OFF Indicator

This push button controls ac power to both the mainframe and expansion cabinets. When power is applied, the ON indicator is lighted.

INTRPT Push Button/Indicator

The INTRPT indicator is lighted whenever the interrupt bit (CC register, bit 0) is set. Pressing the INTRPT push button with the system in RUN generates a pulse that sets this bit.

CLEAR Push Button

Pressing CLEAR while the processor is halted or pressing CLEAR and HALT together while the processor is running resets the system's control state to idle and clears (resets to all zeroes) the following processor registers: A, BR, CC, CD, M, PERM, PERP. Cache memory is also cleared unless the MICRO SOURCE switch on the D/M panel is at C (Cache only).

START Push Button, TEST STATE Indicator

Pressing START causes the processor to start running if it is halted and has no effect if the processor is running. The indicator (TEST STATE) is not functionally connected to the push button. It lights to show the existence of one of the following conditions:

The INTERRUPT switch on the D/M panel is on (up). This causes a processor interrupt on every cycle.

The SINGLE MICRO/NORMAL switch on the D/M panel is at SINGLE MICRO (up).

The MICRO SOURCE switch on the D/M panel is at a position other than NORMAL.

MODE Push Button, NORMAL/MTR Indicator

MODE, a complementing push button, is pressed while the processor is halted to select one of two possible micro sources: NORMAL, in which micros are fetched from the source designated by the MICRO SOURCE switch, or MTR, in which micros are obtained from the cassette (which must be at BOT). The appropriate indicator shows the source selected.

BOT/RWD Push Button and Indicator

Pressing BOT/RWD provides a signal to rewind the magnetic tape cassette. Both indicators are off if the tape is not rewinding or at the beginning; RWD is on while rewinding is in progress; BOT goes on when the tape reaches the beginning-of-tape marker and stays on until the tape moves off the marker.

Diagnostic/Maintenance Panel

The D/M panel, a hinged panel behind a hinged cover at the front right corner of the mainframe cabinet, includes the switches and indicators described in the following subsections, as well as the magnetic tape cassette unit. Figure 1-6 shows this panel. (Sheet 1: B 1905, sheet 2: B 1955.) All functions described in the following subsections are applicable to both series of systems.

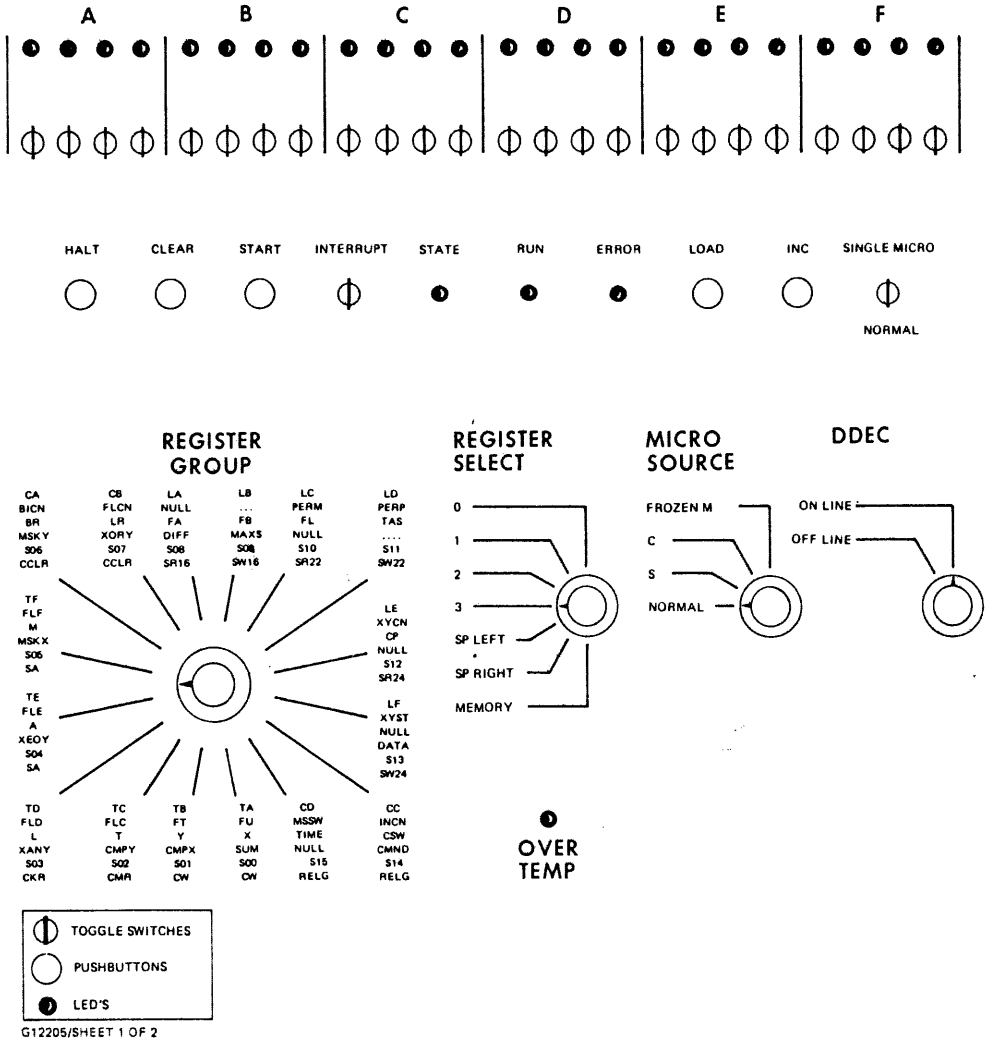


Figure 1-6. B 1900 Diagnostic/Maintenance Panel (Sheet 1 of 2)

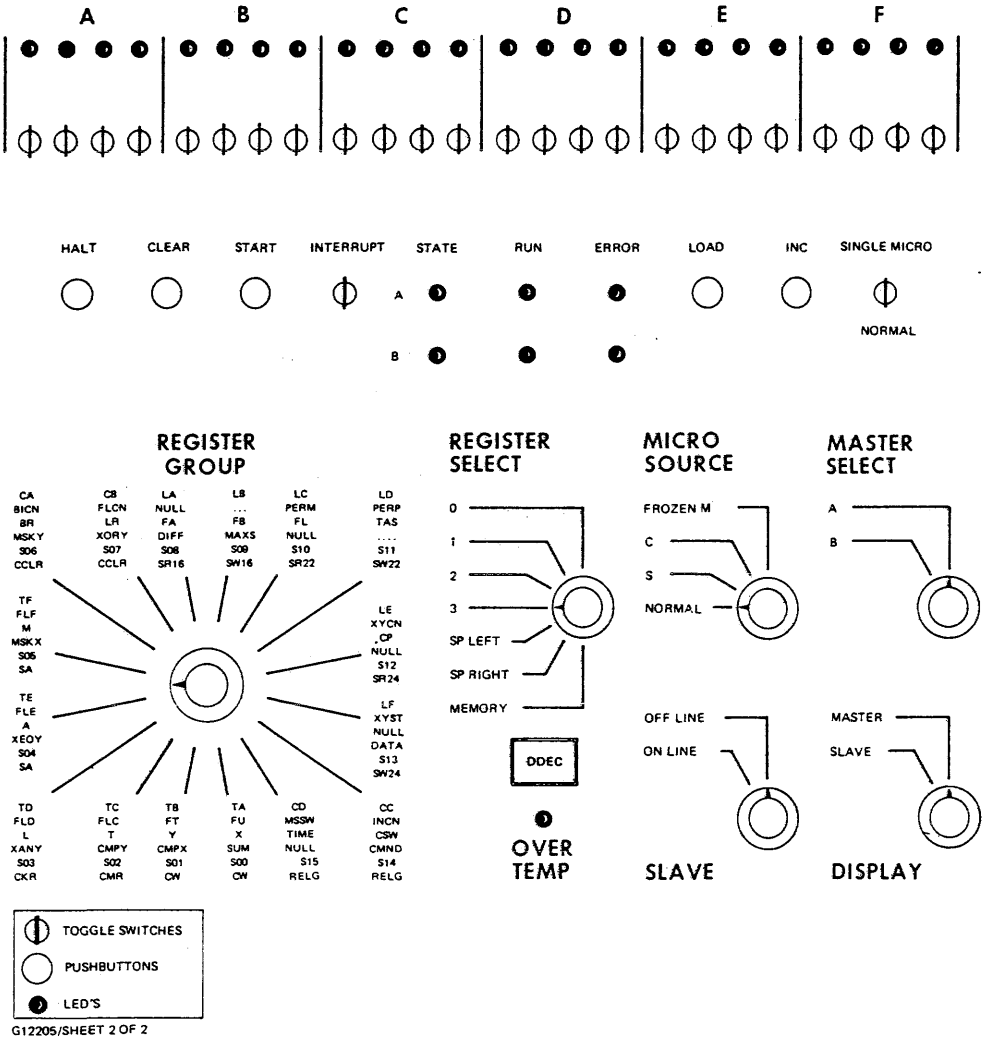


Figure 1-6. B 1900 Diagnostic/Maintenance Panel (Sheet 2 of 2)

Console Lamps

During RUN mode, the 24 console lamps either monitor the leftmost 7 bits of the processor main exchange (MEX) and the 17 output bits of Cache (the micro plus parity) or display the pattern selected by the Load Lamps (7F) micro.

When the processor is halted, the console lamps can be used to display the contents of registers or memory locations selected by the REGISTER GROUP and REGISTER SELECT switches on the D/M panel.

Console Switches

When the processor is halted, these 24 two-position switches can be used to define data to be loaded into a selected register, a scratchpad word, or Cache or S-Memory. The switches are loaded by setting the individual switches (up to represent binary 1, or down to represent 0), the destination is specified by setting REGISTER GROUP and REGISTER SELECT, and LOAD is pressed.

These switches can also be used to specify masks that enable processor halts. (See subsection titled REGISTER GROUP and REGISTER SELECT Switches.)

HALT Push Button

This push button causes the processor to execute the current micro, fetch the next micro to the M register, and halt. If the processor fails to halt, pressing the CLEAR and HALT buttons simultaneously causes the processor to halt immediately.

CLEAR Push Button, START Push Button

These momentary push buttons are identical in their actions to the CLEAR and START push buttons on the Op panel.

INTERRUPT Switch

If the system is in RUN, NORMAL mode, and this toggle is moved to the up position, the interrupt bit (CC register, bit 2) is set. The down position has no effect. The interrupt bit may be cleared programmatically or by means of the CLEAR switch, but it is set again on each clock as long as it is up. INTERRUPT is not active in HALT.

STATE Indicator

When lit, STATE indicates that the most significant bit in the CC register is true. Typically, this bit is made true by software to indicate that the processor is executing an idle loop.

RUN Indicator

RUN is on whenever the processor is running.

ERROR Indicator

When lit, ERROR indicates that at least one bit in the PERM or PERP register is true. These bits are normally set on detection of an error in S-Memory, Cache, or cassette tape. PERM and PERP are 4-bit registers that reset to zero each time the processor is switched from HALT to RUN or whenever the LOAD button is pressed with the processor in HALT and the REGISTER SELECT switch at MEMORY.

REGISTER GROUP and REGISTER SELECT Switches

These two rotary switches are used together to provide selection of a total of 112 (16 X 7) possible conditions. REGISTER GROUP has 16 functional positions. (The position labeled TA is considered position 0; clockwise rotation takes the switch through positions 1 to 15.) REGISTER SELECT has seven positions (0 through 6 reading down). Table 1-1 lists the specific entities and functions available with each combination of the REGISTER GROUP and REGISTER SELECT switch settings.

Positions 0, 1, 2, and 3 of REGISTER SELECT in conjunction with 0 through 15 of REGISTER GROUP define all the addressable registers. (The binary values of the REGISTER GROUP/REGISTER SELECT combinations are used in certain micros to address specific registers.) While the processor is in HALT, the contents of the particular register selected are displayed in the console lamps, and the register is available for loading from the console switches.

Positions 4 (SP LEFT) and 5 (SP RIGHT) of REGISTER SELECT and 0 through 15 of REGISTER GROUP address each of the 16 left and 16 right scratchpad words.

Position 6 (MEMORY) of REGISTER SELECT and 0 through 15 of REGISTER GROUP enable certain Cache and S-Memory operations, as well as examination of the Error Log register (ELOG).

Table 1-1. REGISTER GROUP/REGISTER SELECT Switch Combinations

REGISTER GROUP Switch Position	REGISTER SELECT Switch Position						
	0	1	2	3	SP LEFT	SP RIGHT	MEMORY
0	TA	FU	X	SUM	S00A	S00B	CW
1	TB	FT	Y	CMPX	S01A	S01B	CW
2	TC	FLC	T	CMPY	S02A	S02B	CMR
3	TD	FLD	L	XANY	S03A	S03B	CKR
4	TE	FLE	A	XEOY	S04A	S04B	SA
5	TF	FLF	M	MSKX	S05A	S05B	SA
6	CA	BICN	BF	MSKY	S06A	S06B	SA
7	CB	FLCN	LR	XORY	S07A	S07B	CCLR
8	LA	NULL	FA	DIFF	S08A	S08B	SR16
9	LB	...	FB	MAXS	S09A	S09B	SW16
10	LC	PERM	FL	NULL	S10A	S10B	SR22
11	LD	PERP	TAS	....	S11A	S11B	SW22
12	LE	XYCN	CP	NULL	S12A	S12B	SR24
13	LF	XYST	NULL	DATA	S13A	S13B	SW24
14	CC	INCN	CSW	CMND	S14A	S14B	RELG
15	CD	MSSW	TIME	NULL	S15A	S15B	RELG

LOAD Push Button

While the processor is halted and depending on the settings of REGISTER SELECT and REGISTER GROUP, pressing LOAD causes one of the actions listed in Table 1-2. Tables 1-3 and 1-4 summarize certain other console functions.

Table 1-2. LOAD Push Button; Console Switch Contents Loading

RS	RG	ID	Contents of the console switches are loaded to
0-3	all	**	* The specified register. Exceptions: DATA, CMND
3	13	DATA	* The I/O Bus; RC is issued.
3	14	CMND	* The I/O Bus; CA is issued.
4, 5	all	–	* The Scratchpad location selected.
6	0-1	CW	The Cache location specified by the A register. (Switches contain parity + 16 data bits.)
6	2	CMR	(See Table 1-3.)
6	3	CKR	(See Table 1-3.)
6	4-5	SA	(See Table 1-4.)
6	6-7	CCLR	(See Table 1-4.)
6	8-9	SW16	The S-Memory location specified by FA. (Switches contain 16 data bits.)
6	10-11	SW22	The S-Memory location specified by FA. (Switches contain ECC + 16 data bits.)
6	12-13	SW24	The S-Memory location specified by FA. (Switches contain 24 data bits.)
6	14-15	RELG	(See Table 1-3.)

NOTES

Loading takes place when LOAD is pressed with the processor in HALT.

RS = REGISTER SELECT switch
RG = REGISTER GROUP switch
ID = Applicable RG switch label at the setting specified

* The contents of these locations are displayed in the console lamps as long as they are selected.

** The following cannot be loaded: BICN, FLCN, XYCN, XYST, INCN, TAX, CSW, TIME, SUM, CMPX, CMPY, XANY, XEOY, MSKX, MSKY, XORY, DIFF, MAXS, U.

Table 1-3. Displaying Cache, S-Memory and ELOG After Address is Loaded Using LOAD

RS	RG	ID	Console lamps display
6	2	CMR	The micro plus parity in the Cache location specified by the A register. (Rightmost 17 lamps.)
6	3	CKR	The Cache Key in the Cache location specified by the A register. (Bits 22, 20, 8-0.)
6	8-9	SR16	16 bits (data) of FA-specified S-Memory location.
6	0-11	SR22	22 bits (ECC + data) of FA-specified S-Mem.0 loc.
6	12-13	SR24	24 bits (data) of FA-specified S-Memory location.
6	14-15	RELG	ELOG contents. ELOG is cleared; display remains.

NOTES

Display is visible with processor in HALT.

RS = REGISTER SELECT switch
RG = REGISTER GROUP switch
ID = Applicable RG switch label at the setting specified

Table 1-4. Actions Other than Load and Display

RS	RG	ID	Action
6	4-5	SA	Stop on A. When these settings are selected and the processor is running, a HALT occurs when the contents of the A1 register become equal to the contents of the leftmost 20 console switches.
6	6-7	CCLR	Cache CLEAR. With processor in HALT, pressing LOAD clears Cache memory.

NOTES

RS = REGISTER SELECT switch
RG = REGISTER GROUP switch
ID = Applicable RG switch label at the setting specified

INC (Increment) Push Button

With the processor halted and REGISTER SELECT set to MEMORY, this momentary push button causes the A or FA register to be incremented as follows:

RG	A	FA
0-3	+16	–
8-11	–	+16
12-13	–	+24

INC is not active while the processor is in the RUN state.

SINGLE MICRO/NORMAL Switch

With this two-position toggle switch in the SINGLE MICRO position, pressing START puts the processor into the RUN state and causes it to execute one microinstruction and then return to the HALT state.

With the switch at NORMAL and the processor in HALT, pressing START causes the processor to enter the RUN state and execute micros continuously until a HALT occurs.

If this switch is changed from NORMAL to SINGLE MICRO with the processor in RUN, the processor completes execution of the current micro and halts.

MICRO SOURCE Switch

MICRO SOURCE is a 4-position rotary switch that permits the operator to specify the source of micros. NORMAL specifies that the micro source is Cache memory but if the desired micro is not present, an automatic load of four micros from S-Memory takes place. S specifies that the micro source is S-Memory only and C specifies Cache memory only. FROZEN M means that the micro in the M register does not change and is executed on each cycle.

DDEC Switch Indicator

This on-line/off-line switch for the disk drive electronics control (DDEC) is a two-position rotary switch on the B 1905 panel and an indicating push button on the B 1955 panel.

OVER TEMP Indicator

This indicator goes on if the cooling air flow rate is reduced.

Switches for Two-Processor Systems

The following two-position rotary switches apply to two-processor systems only.

MASTER SELECT enables either Processor A or Processor B to be designated as master.

SLAVE processor designated as slave to be placed on-line or off-line.

DISPLAY allows selection of the processor to be displayed in the console lamps.

Console Uses in Central System Operation

The B 1900 central system may be operated in either an on-line or off-line manner. During on-line operation, control of system activities is by program, with actions proceeding at normal machine speed. The operator communicates with the system through the ODT. Off-line operation, controlled through the two control panels, is usually with the processor halted.

The two control panels represent a physical interface with the processor's micro decoding and control state logic as well as with the 24-bit main exchange (MEX). The panels include on/off, start/stop and operational mode selection functions as well as provisions for examining and manually entering data into the various registers and storage elements.

Basic uses of each panel are summarized in the next two subsections.

Operator Panel; Basic Functions

The Op panel is used primarily for control when the system is on line. The functions provided are therefore restricted to initiating and terminating operations, selecting operational modes, and controlling the cassette tape drive.

The MODE push button is used to select the system's operational mode. NORMAL operational mode implies that the micros are being fetched from the source specified by the MICRO SOURCE switch on the D/M panel. MTR mode specifies that the micro source is the cassette.

The CLEAR push button is used to initialize the system to a predefined state, called CLEAR, in preparation for operation. This is normally done only when the machine is halted. The CLEAR button is duplicated on the D/M panel.

The START push button, used when the processor is in HALT state, causes the execution of micros to begin. The START button is duplicated on the D/M panel.

The INTRPT (interrupt) push button is used to set the interrupt bit (CC register, bit 2). The state of this bit is tested by the executing program, which may specify the action to be taken. Usually, setting this bit results in the processor coming to an orderly halt. The interrupt function is duplicated on the D/M panel.

The RWD (rewind) button enables the cassette tape to be rewound. The processor may be running or halted. The integral BOT indicator signals that beginning-of-tape has been sensed and the tape is ready for use or removal.

Diagnostic/Maintenance Panel; Basic Functions

D/M panel controls are usually used for analysis and troubleshooting. The individual controls and indicators are detailed later in this document.

Individual registers and memory locations may be selected for viewing on the console lamps or for loading from the console switches by means of the REGISTER GROUP (RG) and REGISTER SELECT (RS) switches. To select an entity, RG is set at the position which contains the name of the entity, and RS is set at one of the six specific positions: 0-3 for registers, SP LEFT or SP RIGHT for scratchpad, and MEMORY for S or Cache memories. (The processor must be in HALT.)

The significance of the various bit positions of the display (or of the switches) depends on the unit addressed. Registers are right-justified when displayed individually and occupy the number of bits appropriate to the specific register. Memory data, on the other hand, may occupy any of several positions, depending on the memory access mode selected.

An individual console switch corresponds to the console lamp directly above. Moving a switch up does not alter the contents of the entity; that is done by a load action.

Displaying a Register

If the entity selected is a register (RS positions 0, 1, 2, and 3) the act of selection causes a hardware-forced 1C (Register Move) micro to be executed. The specified register is the source and the CSW register is the destination. The console lamps display the register contents. If the entity is a memory location, displaying the contents requires a read operation. This procedure is described later.

Loading a Register

To load a register, the selection procedure described above is performed first, then the desired bit pattern is entered into the console switches. Now, pressing the LOAD push button causes a hardware-forced 1C (Register Move) micro to be executed and the contents of the switches are gated to the register. Note that the entire contents of the register are affected by the loading process. Therefore, bit states to be saved must be entered into the console switches before LOAD is pressed.

Displaying S-Memory

Displaying the contents of a memory location differs from the process of displaying registers just described because an actual read of the desired memory location must be done. To accomplish this for an S-Memory location, the FA register (refer to Figure 1-7) is selected via RS and RG, the bit address of the desired location is entered via the console switches, and LOAD is pressed. Next, the RS switch is set to the MEMORY position and the desired one of three possible memory read modes (SR16, SR22, SR24) is selected on the RG switch. (Refer to Figures 1-8, 1-9, and 1-10.)

SR16 Read 16 bits of S-Memory beginning at the bit location specified in FA. The data is right justified in the console lamps.

SR22 Read 16 bits of data beginning at any even 16-bit boundary in memory and the corresponding six bits of the ECC (error correction code). The 16 data bits are right justified, and the parity bit appears as the 17th bit. The six ECC bits are left justified. For 22-bit reads, the contents of the least significant four bits of FA are ignored (forcing the use of 16-bit boundaries).

SR24 Read 24 bits beginning at any desired bit location in memory. The data occupies all 24 console lamps.

After selecting the desired memory read mode, the read operation is initiated by pressing the LOAD button.

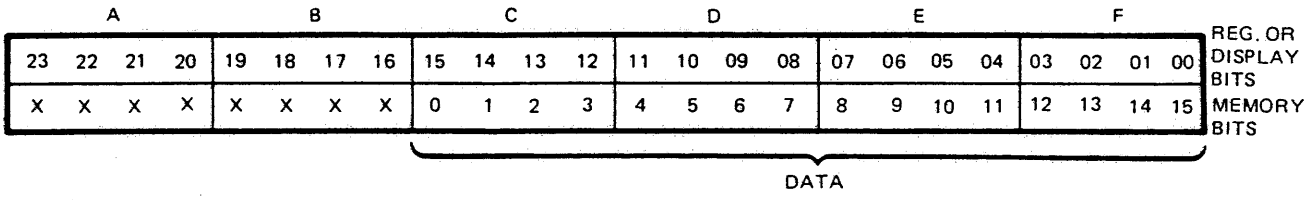
Pressing INC (increment) causes the address in FA to be incremented by an amount appropriate to the read mode selected. FA is incremented by binary 16 for 16-bit or 22-bit reads and by binary 24 for 24-bit reads.

Performing a console read does not destroy the data stored at the addressed location.

FA	FB	FC	FD	FE	FF	
23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0	BIT

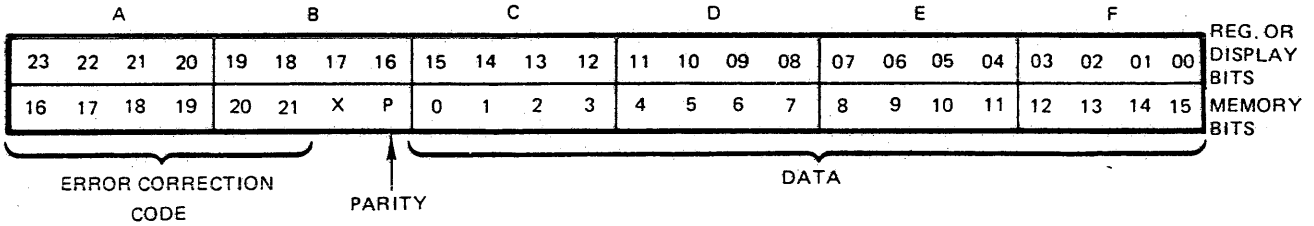
G12206

Figure 1-7. FA Register in Lamps or Switches



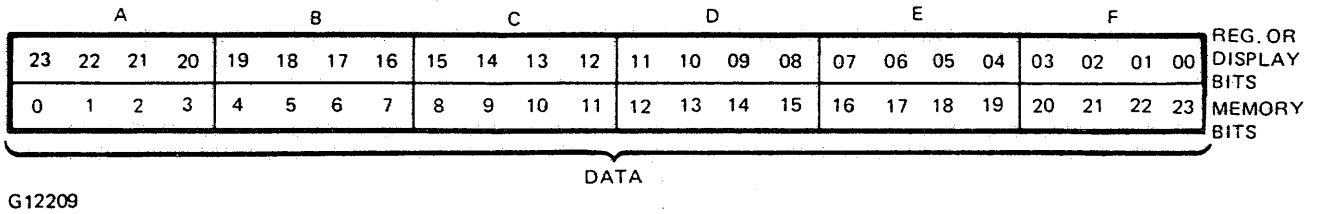
G12207

Figure 1-8. S-Memory in Lamps or Switches for SR16 or SW16



G12208

Figure 1-9. S-Memory in Lamps or Switches for SR22 or SW22



G12209

Figure 1-10. S-Memory in Lamps or Switches for SR24 or SW24

Writing into an S-Memory Location

To write the bit configuration selected on the console switches to be gated to S-Memory, the desired address is first loaded into the FA register, as described above. Then, MEMORY is selected on the RS switch, and the desired memory write mode (SW16, SW22, SW24) is selected on the RG switch.

The memory write modes are analogous to the memory read modes listed above with the following exceptions. (Both exceptions are applicable to SW22.)

1. The parity bit is not used.
2. It is not practical to manually create the proper error-correcting code for a given data field. Therefore, the 22-bit write operation can only be used to verify proper functioning of storage locations used for the ECC bits.

After the bit configuration is entered into the console switches, the actual write is done by pressing the LOAD button.

Pressing INC causes the address in the FA register to be incremented by the appropriate value (16 bits for SW16 and SW22, 24 bits for SW24).

Note that console writes are fixed field length operations. Data to be saved must be reentered via the console switches before LOAD is pressed.

Displaying a Cache Location

To display a specific location in Cache, the Cache address (8-bit key, 10-bit index, and 2-bit word) must first be loaded into the A register. The procedure described earlier for loading a register is used, with the 20-bit A register selected via the RG and RS switches. (Refer to Figure 1-11.) Next, RS is set to MEMORY, RG is set to CMR (Cache Memory Read), and LOAD is pressed. If a matching key is found in Cache, the read is successful (a hit), and the Cache word is displayed in bits 16 through 0 of the lamps. This is not an associative read; read data is also returned on a miss. (Refer to Figure 1-12.)

Parity is not checked on the Cache read data, since the data bypasses the M register. However, a parity check is performed on the Cache key. (No indication is given on the Cache data read.)

The A register is not automatically incremented by the read operation, but it can be incremented by 16 bits by by pressing INC. Thus, the next micro can be read by pressing LOAD again.

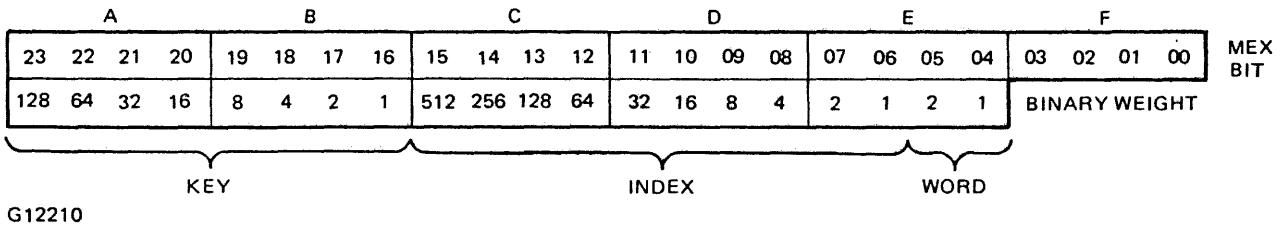


Figure 1-11. A Register in Lamps or Switches

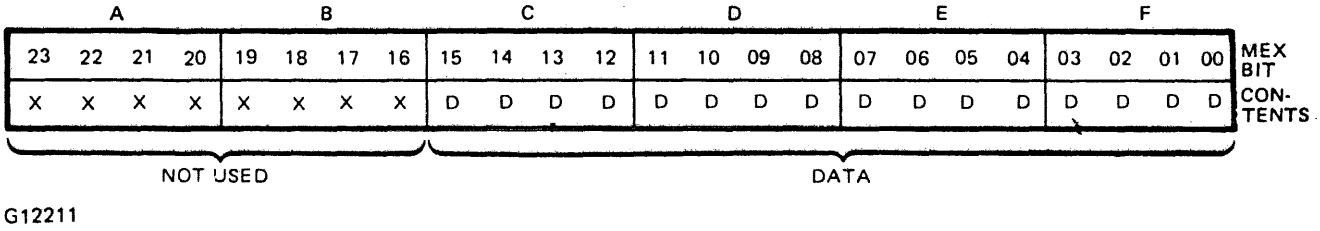


Figure 1-12. Cache Word (Data) in Lamps or Switches

Displaying a Cache Key

To perform a Cache key read, the index portion (bits 11-2) of the A register must first be loaded with the desired index, using the register loading procedure described earlier. (Refer to Figure 1-11.) Next, RS is set to MEMORY, RG to CKR (Cache key read), and LOAD is pressed. The display shows the current key parity at bit 22, the hit indicator at bit 20 (1 = hit), the contents of the key address at bits 8 through 1, and validity at bit 0 (1 = valid). (Refer to Figure 1-13.) This is a nonassociative read; data is always returned, but the hit indicator shows whether or not a match has been made with the key portion of A. This indication is of significance only if an address is placed in the key portion of A before A is loaded.

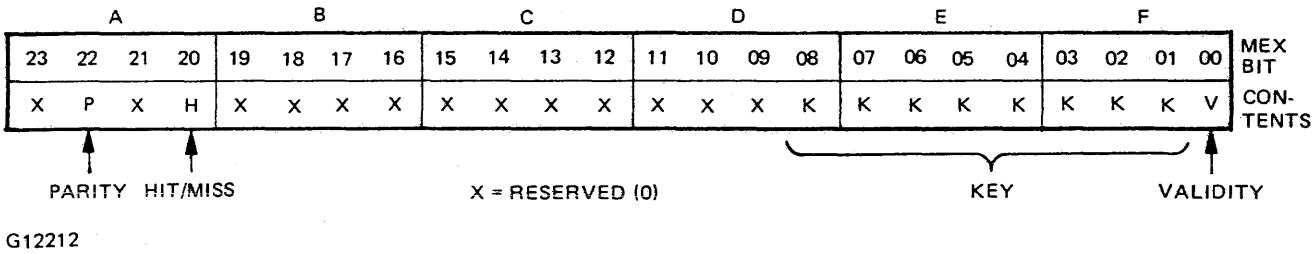


Figure 1-13. Cache Key in Lamps

Reading the Cache key contents does not change the stored addresses or other existing conditions (validity and parity bits). Key parity is checked, an error is reported, and the D/M panel ERROR lamp lights. The parity and validity bit states are returned in the display.

Writing into a Cache Micro and Key Location

To write into Cache, the Cache address must first be loaded into the A register. This procedure is the same as the procedure for displaying Cache. Next, RS is set to MEMORY and RS to CW (Cache Write). The desired data is then entered into console switches 15 through 0, and odd parity, calculated as the odd sum of the data bits plus the parity bit, is entered into bit 16. The write operation is implemented by pressing LOAD.

The A register may be incremented by pressing the INC button.

The eight Cache key locations (bits 23-16) may also be written into by means of this operation.

PROCESSOR

The basic B 1900 processor is designated M-processor-5A (M-5A).

NOTE
Actual processor types include the M-5, M-5A, and M-6. For this document, the M-5A designation is sufficient, and the term "processor" is generally used.

The processor consists of nine logic cards (A-H, J) and a clock card (K). Also treated as part of the processor are the console controls (on the Op panel and the D/M panel) and the cassette tape reader.

Functional Layout

The processor includes the following functional sections:

1. Working registers, for storage of operands, addresses, literals, and control signals having significance to the operations being performed.
2. Arithmetic/logic units (ALUs) and function boxes, for performing arithmetic, logical, and data repositioning (shift, rotate, extract) operations on selected fields of data. These are the portions of the processor in which the actual data processing occurs.
3. A, M, and A-Stack registers, for controlling the source of microinstructions, and providing a pathway to the decoding and execution logic.
4. Cache Memory, for in-processor storage of microinstructions and rapid access to microprogram material in current use.
5. Microinstruction fetch, decode, and execute circuitry and nanoinstruction generation circuitry, for timing and implementing the actions specified in the microprogram.
6. Scratchpad and stack memory for internal processor storage reguirements.
7. PROM and ROM memories for specific logical functions.
8. Console interface for operational and diagnostic control of the system.
9. S-Memory interface, for communication with the systems main memory either directly or via a host adapter.
10. I/O interface, for communication with peripheral devices through individual I/O controls.
11. Data paths. A 24-bit main exchange (MEX) is the bidirectional, internal and external, path for data, and a 16-bit microoperator (MOP) line is the internal, essentially unidirectional, path for microinstructions.

Processor Block Diagram

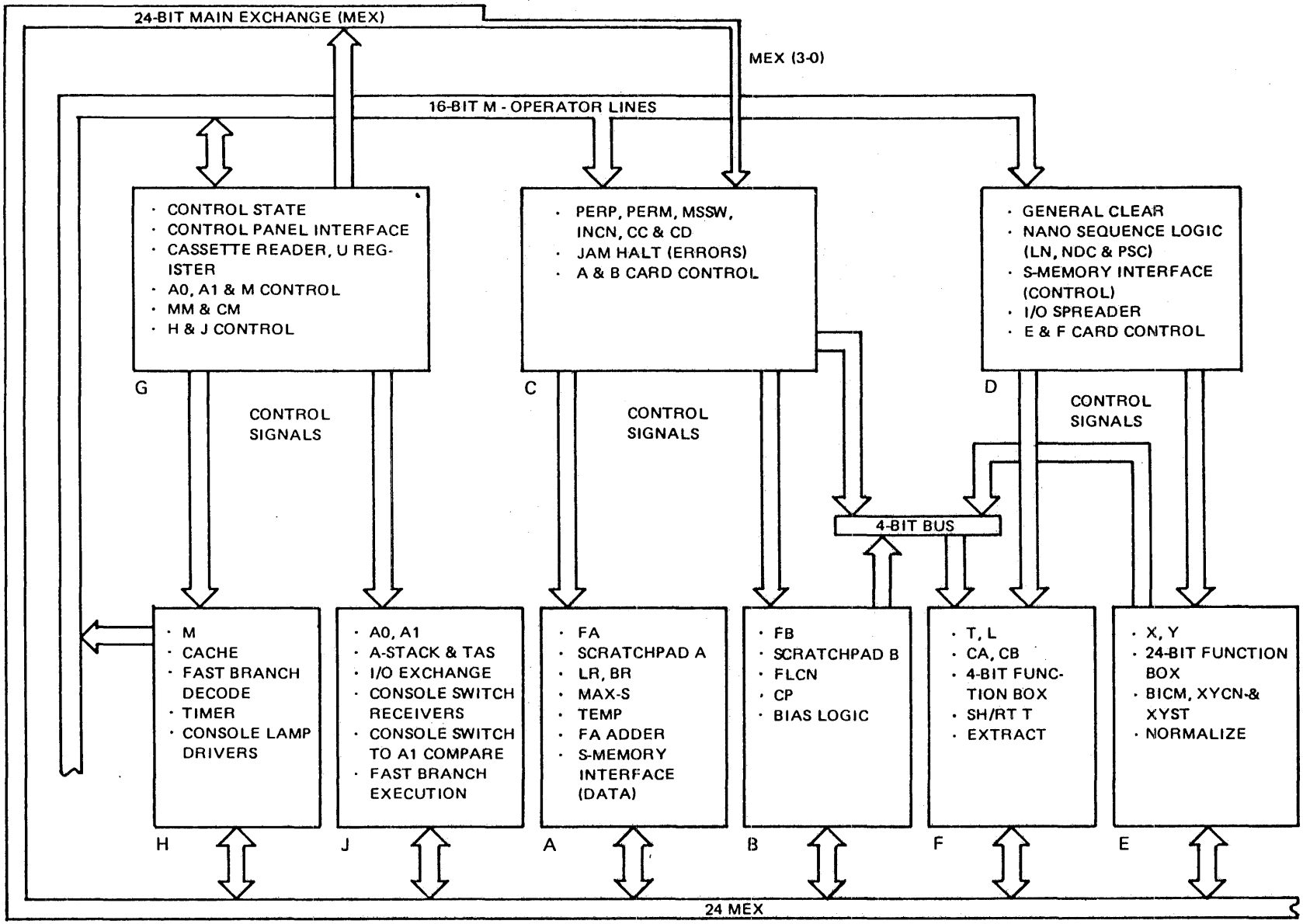
These functional sections are physically distributed throughout the cards that make up the processor and can be visualized by a diagram that groups these cards and shows their relationships and contents. (See Figure 1-14.)

The diagram is segmented in two ways: (1) in terms of the fetch, decode, and execute functions, and (2) by card groups, three cards each, with each group further subdivided in terms of control and data cards. Card group 1 consists of cards G, H, and J; card group 2 consists of cards C, A, and B; and card group 3 consists of cards D, E, and F.

The fetch function involves card group 1 (cards G, H, and J). The decode and execute functions include all three card groups.

G, C, and D are the control cards of their respective groups.

In the subsections that follow, the contents of the individual cards are briefly described.



G12213

Figure 1-14. B 1900 Processor Block Diagram

Card Group 1 (G, H, J)

Card G, the control card, includes the following logical functions:

1. Buffering of the MOP lines from cards G and H, and decoding of the micros as they appear on the MOP lines.
2. Generation of control signals (from PROMs) for the H and J cards.
3. Loading of PROM outputs, resulting from the decoding of micros, to the nanoregister. (The nanoregister is distributed to all three control cards, holds the control states required to fulfill the actions specified by the microinstruction.)
4. processor HALT and RUN state logic (RUN/HALT flip-flop and associated logic.)
5. Micro sequence (pipeline) control logic. Fetch, decode, and execute functions are sequenced, depending on processor state and mode.
6. Micro sequence alteration control logic. This function deals with micros that change the normal micro sequence.
7. MTR (cassette tape) mode control logic. When the processor is in MTR mode, data (each micro) from the cassette is fed bit by bit to the U register. When the micro is completely assembled in U, it is loaded into the M register for decoding and execution.
8. Cache memory fill control (FHS flip-flop and associated logic). If MICRO SOURCE is NORMAL and the needed micro is missing, an automatic Cache-fill operation takes place under control of this logic.
9. Processor initialization. A general CLEAR (GPCLEAR) is generated (1) when the system is powered up, (2) when CLEAR is pressed with the processor in HALT state, or (3) when HALT and CLEAR are simultaneously pressed with the processor in RUN state.
10. H and J card control logic.
11. CM and MM register control logic. CM holds the micro generated to implement D/M panel displaying or loading of registers and memory. MM is used to re-execute the 9C micro if a fetch is needed between the first and second words specified by 9C, and is also used to hold the 8-bit literal portions of the 8C and 9C micros before they are moved to the MEX.
12. Console (Op and D/M panel) interface logic.
13. U register and cassette control logic, including mechanism control, state sequencer, single-micro signal generation, and syndrome register and error correction logic.

Card H includes :

1. Cache memory, including Cache key store and parity generation, the M register, and related logic.
2. Fast-branch decode and address logic.
3. Timer logic.
4. Console lamps register (latches and drivers).

Card J includes :

1. The A register and related logic. This the microprogram address register. It consists of two parts, A0 and A1.
2. The logic for comparison of the console switches and the A1 register.
3. Fast-branch execution circuitry.
4. The A-Stack and the associated TAS (top of A-Stack) register.
5. TTL/CTL and CTL/TTL conversion for the 24-bit path from the MEX to the I/O exchange.

Card Group 2 (C, A, B)

This group deals primarily with S-Memory data control but includes logic and data paths that can be used for general-purpose computing.

Card C (the control card) includes the following functions:

1. Buffering of the MOP lines from cards G and H, and decoding of the micros as they appear on the MOP lines.
2. Generation of the control signals (from PROMs) for the A and B cards.
3. Loading of PROM outputs resulting from the decoding of micros to the nanoregister. (The nanoregister, a pseudoregister that is distributed to all three control cards, holds the control states required to fulfill the actions specified by the microinstruction.)
4. Scratchpad address generation and write control.
5. System status and error information, using the following 4-bit registers: PERM, PERP, CC, CD, and INCN.
6. Micro source information (MSSW register).
7. 4-bit register source select logic. Any of seven 4-bit registers (six FB subregisters plus FLCN) located on card B may be selected for input to the 4-bit function box located on the F card.
8. Error halt detection logic.
9. Mask amount generation logic.
10. Memory write protection logic when a memory address is outside the limits of memory.

Card A provides a 24-bit bidirectional path for memory data as well as MBU address and control signals. Card A includes:

1. The FA register for memory addressing and the BR and LR registers for memory address limit checking. All three of these registers also have general uses.
2. The MAXS register, a jumper chip used to specify the physical size of S-Memory.
3. Scratchpad A, the left half (24 bits) of scratchpad memory.
4. A 24-bit arithmetic/logic unit (ALU) with inputs from FA, LR, BR, TEMP, scratchpad A, and the LCPL (literal or CPL) value. This ALU output is to the MEX.
5. A temporary register used by the BIND (4F) and LOAD LAMPS (7F) micros.

Card B includes:

1. The CP register.
2. The FB register.
3. Bias and FLCN logic.
4. Multiplexing logic (4 bits) with inputs from the six 4-bit subregisters of FB and the 4-bit FLCN rgister.
5. Scratchpad B, the right half (24 bits) of scratchpad memory.
6. A 24-bit ALU with inputs from FL, scratchpad B, and LCPL.

Card Group 3 (D, E, F)

Card D includes the following functions:

1. Buffering of the MOP lines from cards G and H, and decoding of the micros as they appear on the MOP lines.
2. Generation of the control signals (from PROMs) for the E and F cards.
3. Loading of PROM outputs resulting from the decoding of micros to the nanoregister. (The nanoregister, a pseudoregister that is distributed to all three control cards, holds the control states required to fulfill the actions specified by the microinstruction.)
4. Timeout logic and control signals for the clock (K) card.
5. Nano-complete logic, to count the nanoinstruction sequence for the processor. This is required for micros that decode into more than one nano.
6. General CLEAR (GPCLR) logic.
7. I/O spreader logic for assuring sufficient time for I/O data transfer.
8. The S-Memory control interface.

Card E includes:

1. The X and Y registers and related logic.
2. The 24-bit function box, an ALU for the X and Y registers, with BCD sum correction and variable-width output masking under control of CPL.
3. BICN, XYCN, and XYST logic for generating the results of X/Y comparisons.
4. Mask generation and binary mask (binary to BCD) logic.

Card F includes:

1. Rotate and masking logic for the T register.
2. The CA, CB, T, and L registers.
3. Skip and branch logic and a nanoregister.
4. A 4-bit function box.

System Clock (Card K)

Card K includes I/O distribution functions and clock functions. Either of two basic clock frequencies are available depending on the system series. These frequencies are 6 MHz, which provides a basic operating cycle of 167 nanoseconds, or 4 MHz, which provides a cycle of 250 nanoseconds.

The card provides all clock outputs needed by the processor, Cache Memory, S-Memory, and the I/O subsystems and controls.

Registers

The B 1900 processor includes registers that are addressable by (or through the action of) microinstructions. Included in the term "register" are (1) actual storage units of specific sizes and configurations that can be used as sources and sinks, (2) subregisters, which are independently addressable segments of certain larger registers, (3) pseudoregisters, which are usually sources only or destinations only, and (4) two specialized memory units, the A-Stack and the scratchpad.

All the registers described in the following subsections can be addressed by a 4-bit group number in combination with a two-bit select number. These numbers form the 16 rows (0-15) and the first four columns (0-3) of Table 1-1.

The register descriptions that follow are grouped according to basic type of function:

1. General purpose registers, used solely as storage media and inputs to logical, transformational, and arithmetic sections of the processor.
2. Addressing and execution registers that hold the current microinstruction, point to instructions in memory, address data fields in memory, and provide memory protection through boundary checking.
3. Arithmetic/logical registers that may contain the results of certain defined manipulations of data in the general-purpose registers.
4. Interrupt registers and processor status registers.
5. Miscellaneous registers.
6. Scratchpad.

General-Purpose Registers

L Register (Including LA, LB, LC, LD, LE, and LF)

The L register is a 24-bit general-purpose register generally used to hold logical flags for the microprogram code. L and each of its 4-bit subregisters are addressable as sources or destinations. Since the L register is addressable in 4-bit groups, the micros 4-Bit Manipulate (3C), Bit-Test-Branch (4C, 5C), and Skip When (6C) can operate on L register data. The layout of L is identical to that of T, shown in Figure 1-15.

TA	TB	TC	TD	TE	TF	
23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0	BIT

G11210

Figure 1-15. T Register and Subregisters

The Dispatch (1E) micro uses the L register as the source or destination for 24-bit messages (usually addresses) that are stored in or read from S-Memory location zero.

The L register is one of the four registers (X, Y, L and T) used in the Read/Write Memory (7C), Diagnostic Read/Write Memory (11D), and Read/Write Cache (7E) micros.

The L register is also used in conjunction with the Bind (4F) micro.

T Register (Including TA, TB, TC, TD, TE, and TF)

The T register is a 24-bit general purpose register used primarily for the interpretation of S-Language instructions. The T register and each of its 4-bit subregisters (see Figure 1-15) are addressable as sources and destinations. Since the T register is addressable in 4-bit groups, micros 3C, 4C, 5C, and 6C can operate on its data. The T register also has unique functions implemented by the Shift/Rotate T (10C) and Extract From T (11C) micros.

The T register is one of the four registers (X, Y, L and T) used in the 7C, 11D, and 7E micros.

Dispatch operations use the least significant seven bits of T as the source or destination for port and channel information.

The T register is also used in conjunction with the Bind (4F) micro.

X Register, Y Register

The X and Y registers are 24-bit general purpose registers used to hold the two operands of the arithmetic and combinatorial logic units. Both are addressable as sources and destinations and both can be shifted or rotated, together (in concatenation) or separately. The X register can also be normalized.

The X and Y registers are two of the four registers (X, Y, L and T) used in the 7C, 11D, and 7E micros.

The X and Y registers are compared in the Cassette Control (2E) micro to cause either a cassette and processor halt or a microinstruction skip, depending on the variant.

TAS (Top Of A-Stack) Register

The 24-bit TAS register is the top of the A-Stack (see next subsection) and is addressable as a source and a destination. References to TAS cause the stack to be automatically pushed or popped.

A-Stack

The A-Stack is a 24 (wide) by 32 (deep) memory with no automatic hard overflow interrupt. It operates as a push-down stack with a last-in, first-out (LIFO) structure. Its common use is for the storage of A register addresses to allow fast subroutine calling, but it also can be used to store data. Together with the TAS register, the A-Stack provides a virtual stack 33 words deep. A push stores the information in TAS into the location specified by the Stack pointer, and puts new information in TAS. A pop reads the information in TAS and moves the contents of the indicated Stack location into TAS.

Addressing and Execution Registers

FA (Field Address) Register

The 24-bit FA register (Figure 1-16) is addressable as either a source or a destination. FA is used primarily to hold an absolute bit address for S-Memory and, with one exception, can directly address any bit in S-Memory starting at any point. The exception is the inability to address the last bit in the memory in a negative direction.

FA can be counted up or down by a literal in a microinstruction or by the value contained in the CPL register. It can be incremented or decremented by a value in the left scratchpad word, and also can be loaded, stored, or exchanged, along with the FB register, into a double scratchpad word.

Neither overflow nor underflow of the FA register is detected; its value can go above @FFFFFF@ and below @000000@ and wrap around.

Variants of the Read/Write Cache (7E) micro use the FA register as the source for Cache addresses.

In Read/Write Memory (7C) and Swap Memory (2D) operations, FA is checked to assure that its value lies between the limits set by the values in the BR and LR registers.

FB Register

The 24-bit FB register is functionally divided into three subregisters: FU (field unit), 4 bits; FT (field type), 4 bits; and FL (field length), 16 bits. FB is further divided into four 4-bit subregisters FLC,

FLD, FLE, and FLF. FB as well as the seven subregisters are addressable as sources and as destinations. FB is shown with FA in Figure 1-16.

FU holds the length of the unit that makes up a field in S-Memory, FT holds field information of interest to software operations, and FL holds the total length of the field. FL can describe fields of up to 65,536 bits (8K bytes); this value can be adjusted up or down by a literal in a micro or by the value contained in the CPL register.

Overflow of FL is not detected and causes wrap-around. Underflow is detected, however, and no wrap-around occurs; a value of zero is left in FL after underflow.

Since FB is addressable in 4-bit groups, the 3C, 4C, 5C, and 6C micros can operate on its data. FB can be loaded, stored or exchanged along with FA into a double scratchpad memory word.

FU and FL, along with corresponding portions (SFU, SFL) of word 0 of scratchpad A, are used to set various conditions of the FLCN and CP registers.

BR (Base Register) and LR (Limit Register)

BR and LR are 24-bit registers used for S-Memory protection. Each can be addressed as a source or destination.

S-Memory protection is provided by comparing the memory address in FA with the limits in LR and BR. Any address outside these bounds is flagged in the CD register (bit 0 for Write or Swap, bit 1 for Read). Read is permitted if the address in FA is outside the bounds, but Write and Swap are permitted in such cases only if CD register bit 2 (the override bit) is true.

If the FA register points to an in-bounds bit but the field length extends past the boundary, the whole field (including the out-of-bounds bits) may be accessed.

A count operation, specified by the Count FA/FL (6D) micro, is fulfilled as specified even if FA is out of bounds.

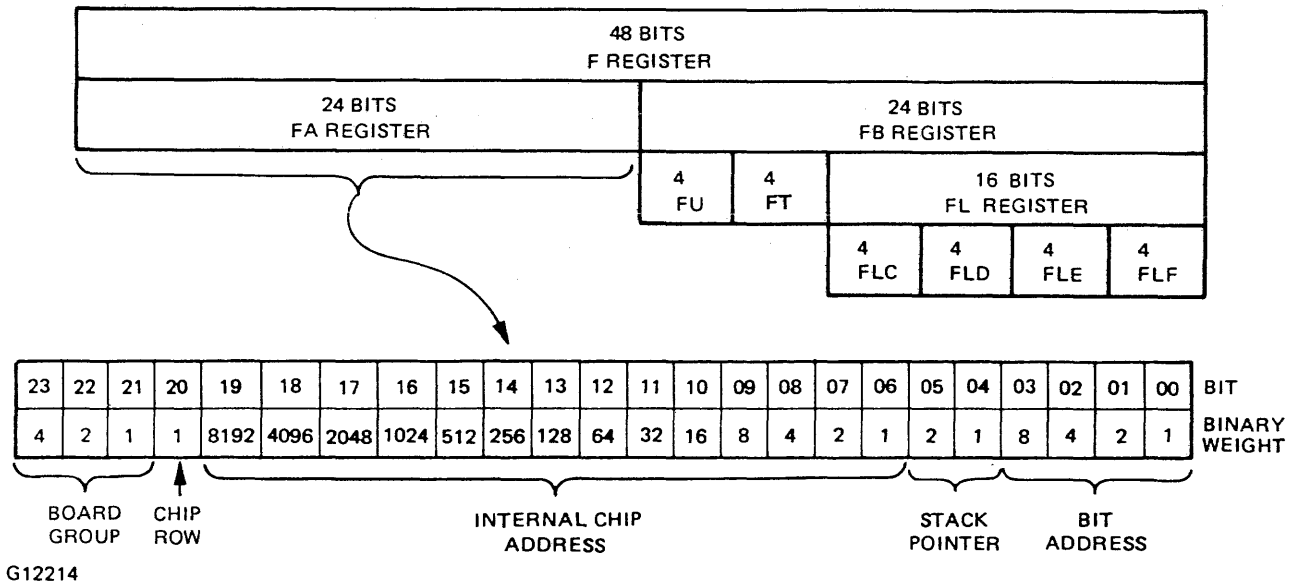


Figure 1-16. FA and FB Registers

M (Microinstruction) Register

The 16-bit (plus odd parity bit) M register holds the micro that is decoded into various control signals within the processor. These are the signals that perform the operations specified by the micro.

The M register (Figure 1-17) is divided into four fields (C, D, E, F), structured so that a maximum of 61 distinct micros can be decoded.

M is addressable as a destination. Data moved to M are bit-ORed into the next micro.

The 17th bit is the parity bit, and a parity check is performed on the contents of the M register and its parity bit after each fetch. Parity checks are not performed after a move to M nor on inputs from the console switches or the cassette tape.

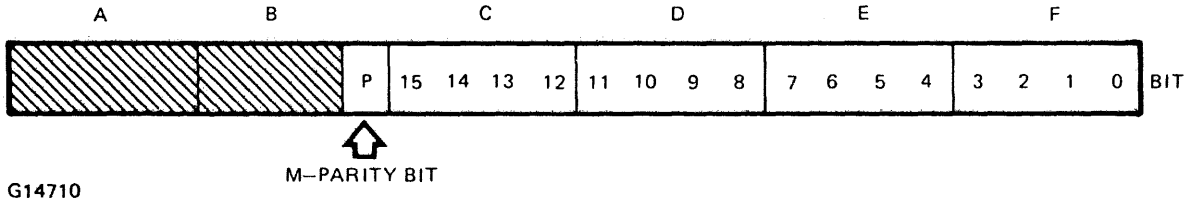


Figure 1-17. M Register

U Register

The 16-bit U register, accessible as a source only, accumulates bit-by-bit input from the cassette tape. Reading of cassette tape is done with the system in MTR mode.

Access to U is delayed until 16 bits have been transferred from tape to U; then, the contents of U are automatically moved to the M register for execution.

If a micro that references U as a source is executed, the contents of U are moved directly to the destination register. If M is the destination, the U's contents are bit-ORed with the current contents of M. After the micro is complete, the next micro goes to M for execution, as before.

When a micro specifying a branch type of action occurs, the branch is taken and the contents of the A register may be affected, but the next micro to be executed is the next one from the cassette tape.

In NORMAL mode, if the U register is addressed following a Cassette Control (2E) micro specifying a cassette halt, the results are undefined.

A (Address) Register

The A register (see Figure 1-11) is a 20-bit register that addresses micros in either Cache or S-Memory. The A register can be addressed as a source or as a destination. As a source, four low order zero bits are appended to its contents. As a destination, the four low order bits of the source are lost.

The A register automatically increments by one on each processor cycle in step or continuous operation (SINGLE MICRO, CONT), but does not increment in MTR mode or if FROZEN M is specified on the MICRO SOURCE switch. Any micro calling for a skip or a branch modifies A.

Values from 0 through 4,095 (12-bit length) can be added to or subtracted from the address in the A register.

The Bind (4F) micro addresses A as a destination, and variants 0-3 of Read/Write Cache (7E) use A as an address register.

Wrap-around of the A register can occur and is permitted.

The A register is described in more detail in the subsection entitled Microprogramming.

MAXS (Maximum S-Memory Size) Register

MAXS (Figure 1-18) is a 24-bit pseudoregister that can be adjusted (strapped) to give the size of S-Memory installed in the system.

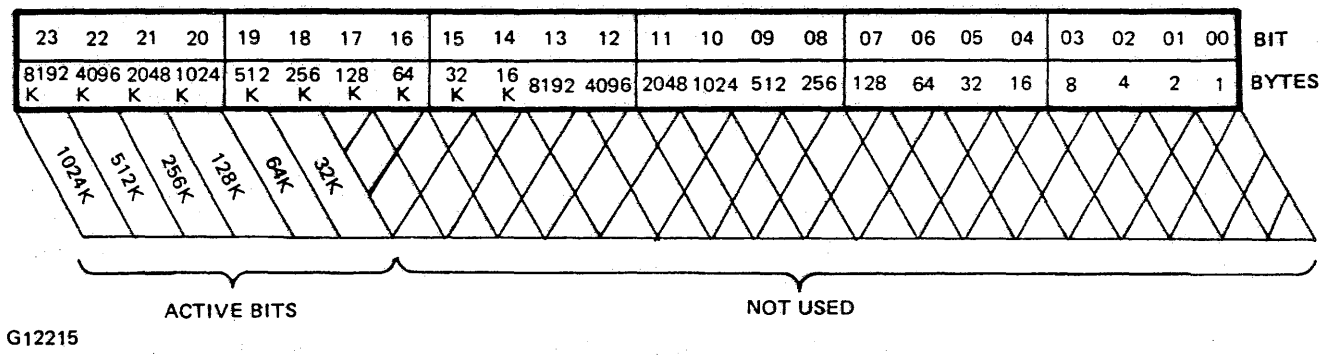


Figure 1-18. MAXS Register

Arithmetic and Logical Registers

SUM Register

SUM, a pseudoregister addressable as a source only, is equal to the sum of the X, Y, and CYF registers (X + Y + CYF). Zero bits are produced in the higher order positions of the 24-bit result when the length set in the CPL register is less than 24; CPL values greater than 24 yield undefined results.

The carry-out level is generated from the output bit position, as specified by CPL. If CPL equals 1, the carry-out level is generated from the least significant bit of the ALU.

If bit 5 of the CPU register equals 0, the binary sum is produced. If bit 5 of CPU equals 1, the decimal sum is produced by considering each of the X and Y inputs as from one to six 4-bit binary coded decimal (BCD) nybbles. Results are not defined for non-BCD nybbles. (Bit 6 of CPU is a "don't care" bit.)

DIFF (Difference) Register

DIFF, a pseudoregister addressable as a source only, is equal to the difference of the X, Y, and CYF registers (X – Y – CYF). Zero bits are produced in the high order bit positions of the 24-bit result when the length set in the CPL register is less than 24. CPL values greater than 24 yield undefined results.

The borrow level, generated from the static comparison of all 24 bits of X and Y, is true if X < Y or if X = Y and CYF is true.

If bit 5 of CPU register equals 0, the binary difference is produced. If bit 5 of CPU equals 1, the decimal difference is produced by considering each of the X and Y inputs as up to six 4-bit binary coded decimal (BCD) nybbles. Results are undefined for non-BCD nybbles.

A negative result is in twos-complement form in the binary case and in tens-complement form in the decimal case.

XANY, XORY and XEOY (Boolean Functions)

XANY (X AND Y), XORY (X OR Y), and XEOY (X Exclusive-OR Y) are 24-bit pseudoregisters that provide the results of the AND, OR and exclusive-OR logical functions. Each is addressable as a source only.

Zero bits are produced in the higher-order positions of the 24-bit result when the length set in the CPL register is less than 24. CPL values greater than 24 yield undefined results.

CMPX (Complement of X), CMPY (Complement of Y)

CMPX and CMPY are 24-bit pseudoregisters. They are addressable as sources only, and produce, respectively, the ones complement of X and of Y. Zero bits are produced in the higher order positions of the 24-bit result when the length set in the CPL register is less than 24. CPL values greater than 24 yield undefined results.

MSKX (Mask of X), MSKY (Mask of Y) Registers

MSKX and MSKY are 24-bit pseudoregisters, addressable as sources only. Each produces the contents of its associated register (X, Y) with the high order positions masked as specified by bits in the CPL register. When CPL = 24, all bits are gated through X or Y. When CPL is less than 24, the number of low order (rightmost) bits specified by the CPL value are gated. When CPL = 0, all bits are masked off. CPL register values greater than 24 yield undefined results.

BICN (Binary Conditions) Register

BICN, a 4-bit pseudoregister addressable as a source only, contains the following:

Bit 3 True if LSUY (least significant unit of Y) is true and CPU = 00 or if the least significant nybble of Y = 1001 and CPU = 01.

Bit 2 CYF. Reflects the state of the carry flip-flop in the CP register.

Bit 1 CYD (borrow-out level). A function of X, Y, and CYF. (See DIFF Register.)

Bit 0 CYL (carry-out level). A function of X, Y, CYF, CPL, and CPU. (See SUM Register.)

XYCN (XY Conditions) Register

XYCN, a 4-bit pseudoregister addressable as a source only, provides information on the most significant bit of the X register as well as certain relational conditions between the X and Y registers.

Bit 3 MSBX. Reflects the state of the most significant bit of X. The width of X is specified by CPL, and can range from 0 to 24. MSBX = 0 if CPL = 0; MSBX = the state of the most significant bit of X if CPL = 1 through 24.

Bit 2 True if X = Y.

Bit 1 True if X < Y.

Bit 0 True if X > Y.

NOTE

All 24 bits of X and Y are included in the comparisons specified by XYCN bits 2, 1, and 0 regardless of the values of CPL and CPU.

XYST (XY States) Register

XYST, a 4-bit pseudoregister addressable as a source only, provides the following information:

Bit 3 LSUX (least significant unit of X) is true when the least significant bit of X is true and CPU = 00 or when the least significant nybble of X = 1001 and CPU = 01.

Bit 2 INT (interrupt) is a flag and is true if any of the following conditions are true. (See appropriate subsection for more information.)

Bit	Condition
INCN(3)	Missing port device
INCN(1)	Port interrupt
CC(2)	100-ms real-time clock interrupt
CC(1)	I/O bus service request interrupt
CC(0)	Control panel interrupt
CD(3)	Memory error interrupt
CD(0)	Memory write/swap address out-of-bounds interrupt

Bit 1 True if all 24 bits of Y = 0.
Bit 0 True if all 24 bits of X = 0.

Interrupt and Processor Status Registers

C Register

The 24-bit C register is not addressable as an entity. It is segmented into one 8-bit portion, CP, and four 4-bit portions, CA, CB, CC, and CD. CP is further segmented into CYF, 1 bit, CPU, 2 bits, and CPL, 4 bits. (See Figure 1-19.)

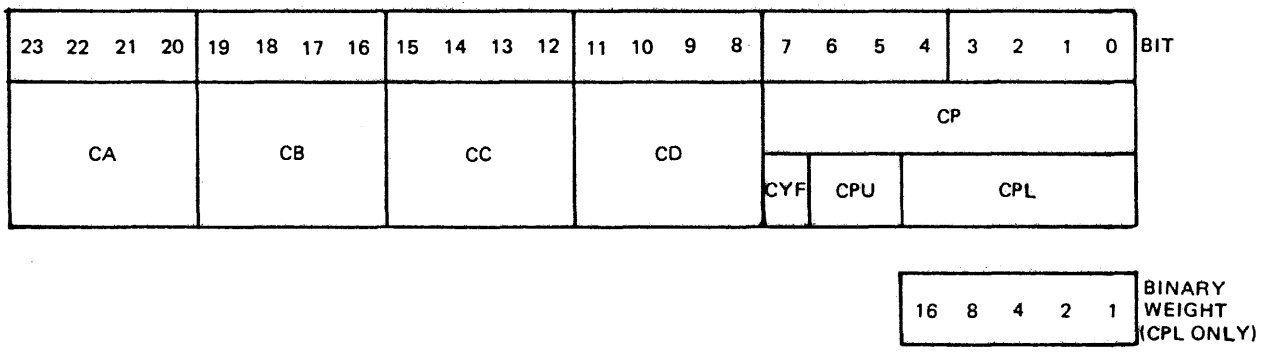


Figure 1-19. C Register and Subregisters

CA AND CB Registers

The CA and CB registers are available as general purpose 4-bit registers and have no specific functional assignment. Manipulate (3C), Bit Test (4C, 5C), and Skip (6C) micros are applicable to CA and CB.

CC Register

CC, addressable as a 4-bit source or destination, is used to store four processor states and conditions:

Bit 3 D/M Panel STATE lamp flip-flop. When TRUE, this bit causes the STATE lamp to be on.

Bit 2 Timer. Set every 100 ms by the real-time clock interrupt signal.

Bit 1 I/O Bus service request interrupt. Set at each clock time if one or more of the I/O controls connected to the I/O Bus is requesting service.

Bit 0 Control panel interrupt. The processor must be in the RUN state to allow the INTERRUPT switch (D/M panel) or the INTRPT switch (Op panel) to set this; in HALT, these switches have no effect. This bit also drives the lamp behind the INTRPT switch on the Op panel.

The Manipulate (3C), Bit Test (4C, 5C), and Skip (6C) micros are applicable to the CC register. When a CC bit is reset by a micro, the bit remains false for at least one clock period. A Test micro executed in that clock period will find the bit false even if the condition responsible for setting the bit has occurred. If the condition then ends before reset time (timer interrupt), the bit never gets set for that occurrence of the condition.

CD Register

CD, addressable as a 4-bit source or destination, is used to store four processor states and conditions:

Bit 3 Memory error interrupt flag. Set at next clock following the setting of any PERM or PERP bit, but not reset with resetting of PERM or PERP.

Bit 2 Memory Write/Swap address out-of-bounds override (BR/LR check). If set, the execution of a Write or Swap outside the BR/LR limits is permitted, but the setting of an out-of-bounds interrupt or flag bit is not affected. (See BR, LR Registers.)

Bit 1 Memory Read address out-of-bounds flag (BR/LR check). See Bit 0, below.

Bit 0 Memory Write/Swap address out-of-bounds interrupt (BR/LR check). Bits 1 and 0 indicate memory out-of-bound conditions derived from comparisons of FA with BR and LR on all memory data accesses.

The Manipulate (3C), Bit Test (4C, 5C), and Skip (6C) micros are applicable to the CD register.

CP Register, including CYF, CPU, CPL

The 8-bit CP register includes the following subregisters: CYF (carry flip-flop), 1 bit; CPU (unit control), 2 bits; CPL (data length control), 5 bits. Only CYF is addressable individually (by the Set CYF (6E) micro). The CP register as a whole is addressable as either a source or destination. CPU and CPL may be addressed individually by Bias (3E).

The value in CYF can be read (but not changed) as bit 2 of BICN.

The value of bit 5 of CPU determines the unit of the input field to the ALU; 0 = binary and 1 = BCD.

The value in CPL determines the length of the inputs to the ALU. This can vary from 0 through 24 (CPL = 00000-11000).

The Bias micro is also applicable to the CP register.

INCN (Interrupt Conditions) Register

INCN, 4-bit pseudoregister addressable as a source only, relates to port operations. Bit 3 is affected by dispatch operations (see micro 1E). Bits 2, 1, and 0 reflect the states of certain interface lines between the processor and the host adapter in port-connect systems. In direct-connect systems (no host adapter), INCN bits 2, 1, and 0 are false.

Bit 3 Missing port device.

Bit 2 High-priority port interrupt.

Bit 1 Port interrupt.

Bit 0 Port lockout.

PERM (Parity Error in Memory) Register

PERM, a 4-bit register that is addressable as either a source or a destination, holds information on memory error conditions. If any PERM bit goes true: (1) CD bit 3 goes true on the next clock, and (2) ERROR (D/M panel) lights during HALT. PERM is cleared (all bits reset) by (1) powering up, (2) pressing CLEAR or START, or (3) setting REGISTER SELECT to MEMORY and pressing LOAD.

PERM holds the following information:

Bit 3 Micro timeout. This bit is set if a timeout occurs during the execution of any micro except Cassette Control (2E)

Bit 2 S-Memory out-of-bounds. This bit is set when the actual memory size, as specified by the jumper chip in the MBU, is exceeded during a memory operation. (In this situation, all zeroes are returned for out-of-bounds bits but there is no error-correcting action and, hence, no parity error indication. During a fetch, however, this error halts the processor.) In a write or swap operation, the out-of-bounds signal may be received as late as seven clock periods after the end of the operation.

Bit 1 Change in memory error log (ELOG). This bit is set when any ELOG bit is changed from 0 to 1. If the change is due to a processor access, the error signal can be received as late as seven clock periods after the end of the operation.

Bit 0 Uncorrectable S-Memory error during a processor operation. This bit is set when an uncorrectable S-Memory error is detected. During a micro fetch, this error halts the processor.

PERP (Parity Error in processor) Register

PERP, a 4-bit register addressable as a source and a destination, provides but is not limited to processor parity error reporting. PERP also reflects certain error conditions detected by the processor during micro fetches and cassette tape read operations. If any PERP bit goes true: (1) CD bit 3 goes true on the next clock, and (2) ERROR (D/M panel) lights during HALT. PERP is cleared (all bits reset) by (1) powering up, (2) pressing CLEAR or START, or (3) setting REGISTER SELECT to MEMORY and pressing LOAD.

Bit 3 Always reset (0).

Bit 2 Cache key parity error. This bit is set if a parity error occurs while reading a Cache key (8 bits plus parity). The processor halts on detection of this parity error except during a diagnostic read of Cache.

Bit 1 Parity error on word fetched to M. This bit is set (and the processor halts) when a parity error is detected on a micro fetched to M from Cache or S-Memory. Parity error reporting is disabled when loading M from cassette tape, from the console switches, or via a MOVE micro.

Bit 0 Uncorrectable cassette read error. This bit, when set, indicates that an uncorrectable error occurred while the cassette tape was being read. If MODE is MTR, the processor halts.

MSSW (MICRO SOURCE Switch) Register

MSSW is a 4-bit register addressable as a source and a destination. Bit 3 = 0 indicates that the processor is a "master", Bit 3 = 1 indicates that the processor is a "slave". In a dual-processor environment, Bit 2 = 0 indicates the A processor and Bit 2 = 1 indicates the B processor. Bits 1 and 0 function as follows.

The two-bit signal from the MICRO SOURCE switch is bit-ored with the contents of MSSW whenever MSSW is (1) designated as the source by a micro, or (2) used to control the source of micros. Table 1-5 explains the results of this action.

MSSW does not control the micro source if the MODE push button is set to MTR. In this case, the micro source is the cassette tape.

Table 1-5. Effect of MSSW Bit Positions 1 and 0

MSSW Bits 0, 1	MICRO SOURCE Switch	Interpretation
00	NORMAL	Micro source is Cache. If a miss occurs, micros are obtained from S-Memory.
01	S	Micro source is S-Memory only.
10	C	Micro source is Cache only. A miss causes the processor to halt.
11	FROZEN M	Micro source is the M register. The micro in M is executed repeatedly.

Miscellaneous Registers

DATA Register

DATA, a 24-bit pseudoregister, is used to transfer data to and from the I/O Bus. DATA can be addressed as either a source or destination. When DATA is a source, the processor generates an RC (Response Complete) signal to the interface and accepts the 24 bits of data from the bus. When DATA is a destination, the processor generates the RC signal to the interface and the 24 bits of data are moved from the designated source to the bus. Hardware prevents the RC from occurring fewer than eight clock periods after a previous RC or fewer than three clock periods after a Command Active (CA) signal.

CMND (Command) Register

CMND, a 24-bit pseudoregister addressable as a destination only, is used to transfer commands to devices on the I/O Bus. The processor generates a CA signal to the interface and moves the contents of the designated source to the bus. Hardware prevents a CA from following an RC any sooner than four clock periods.

NULL

Registers designated as NULL on the REGISTER GROUP dial may be considered as 24-bit pseudoregisters that are addressable as sources or destinations. As a source, NULL supplies all zeroes; as a destination, data addressed to NULL is lost. A typical application is the use of NULL as a destination when the TAS register must be popped without affecting other registers.

CSW (Console Switch) Register

CSW, 24-bit pseudoregister addressable as a source only, provides the data represented by the settings of the console switches.

FLCN (Field Length Condition) Register

FLCN, a 4-bit pseudoregister addressable as a source only, holds the result of a comparison of the FL subregister and the corresponding portion of the first word of the right scratchpad, called SFL. Bit meanings, when set, are as follows:

Bit 3 FL = SFL

Bit 2 FL > SFL

Bit 1 FL < SFL

Bit 0 FL = 0

TIME Register

TIME, a 24-bit register, increments continuously every 0.5 microsecond when power is applied. It wraps around when it passes its highest possible value.

TIME can be read by the Register Move (1C) micro and cleared by designating it as a destination in the Move 8-Bit Literal (8C) micro. Information cannot be moved into it.

Scratchpad Memory

Scratchpad is a memory with a capacity of 16 pairs of 24-bit words. (See Figure 1-20.) Scratchpad holds field descriptors during the iteration of operands, and may also hold S-Language stack pointers and other processor registers that are under constant manipulation.

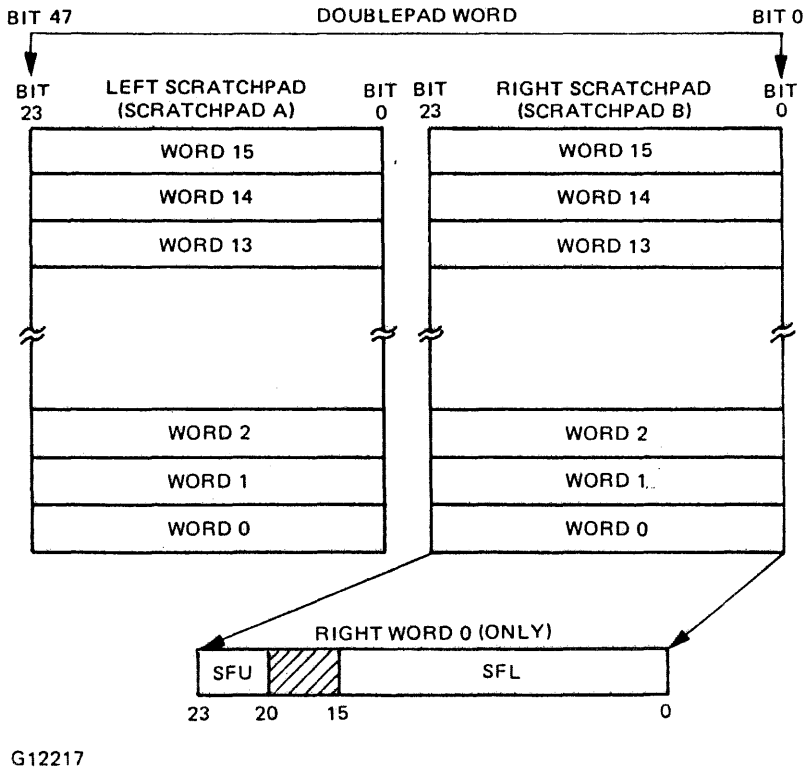


Figure 1-20. Scratchpad Memory

Each of the pairs of words is considered as a left (A) word and a right (B) word. A and B together are called a doublepad word. The sixteen A (left) scratchpad words are numbered S00A through S15A, and the sixteen B (right) words are numbered S00B through S15B on the REGISTER GROUP switch. The FU and FL registers, along with corresponding portions (SFU and SFL) of the first location (S00B) of the left scratchpad, are used to set various conditions of the FLCN and CP registers. SFU and SFL refer respectively to the most significant and least significant four bits of S00B.

Storing information into the scratchpad is a two-step operation. Each step requires one clock to execute: data is trapped into a latch on the first clock and written into scratchpad on the second clock. Another micro can be executed during the second clock if it is not a read from scratchpad, which is delayed by hardware until the store into scratchpad operation is complete.

Microprogramming

The B 1900 central system performs its operations by means of a set of 40 low-level microinstructions (micros) which are fetched from Cache or S-Memory. The micros are decoded within the processor and the outputs thereby produced activate the appropriate registers, pseudoregisters, logic and arithmetic sections for performing (executing) specific activities.

This scheme, called microprogramming, provides several advantages. One advantage is the ease with which modification of machine operations can be made; that is, an operation can be changed through modification of software rather than modification of the system's logic. Another advantage is the ability to tailor the manner in which the hardware is exercised to the requirements of the specific programming language (COBOL, FORTRAN, etc.) being used at the time.

The three-phase (fetch, decode, execute) process takes place in a "pipeline" fashion; that is, the processor normally does not wait for one micro to go through the three phases before fetching the next micro. At any given time, while one micro (say, micro A) is executing, the next one (micro B) is being decoded and a third one (micro C) is being fetched. Pipelining, coupled with the ready availability of most micros from Cache, expedites processor operation.

Micro Access and Execution Logic

A micro fetch is accomplished as follows: the micro is obtained from the location specified by the A register and gated to the M register for decoding and execution. The location of the micro is specified by an S-Memory address, even though the micro is usually obtained from Cache memory. (Cache essentially duplicates a portion of S-Memory within the processor to provide rapid access to micros.) If the desired micro cannot be found in Cache when it is needed, an automatic loading feature is used to obtain it from S-Memory. Operation of the micro fetch, decode, and execute structure involves the M register, the A register, Cache Memory, and nano sequencing. These topics are discussed in turn in the subsections that follow.

M Register

The M register (see Figure 1-17) is a 17-bit (16 data bits, 1 parity bit) register that holds the micro to be executed. The contents of M are fed to the micro decoding logic and used to produce the various enabling signals which initiate the specified operation. M is segmented into four fields for decoding. In the B 1900, decoding involves the addressing of stored logic in the form of programmable read-only memories (PROMs).

Normal input to the M register is from Cache, and a direct path is provided. The M register can also receive input from the main exchange (this path is used when micros are obtained directly from S-Memory), from cassette tape via the U register, and from the console switches via the CM register. The M register may also serve as a source or sink for Register Move operations. When used as a sink, the input data is bit-ORed with the incoming micro.

A Register

The A register is a 20-bit microinstruction address register which is used to hold the address of the micro currently being accessed (fetched) for execution. The address contained in A defines a location in S-Memory where the desired micro is to be found. However, actual micro fetching, in most cases, is performed from Cache memory, which is a duplicate of a portion of S-Memory.

The address in A specifies a location in Cache where the desired micro is to be found, provided it was previously placed there by the hardware function for doing so. Normally, the desired micro is in Cache when needed (a hit). However, if it is not present in Cache (a miss), the same A address is used to fetch it from S-Memory. Concurrent with a fetch from S-Memory, a hardware controlled Load Cache routine is performed which results in writing the addressed micro plus the three following micros into Cache. Micro execution with fetches from Cache then continues until another miss is encountered or an exception condition arises.

The A register is a two-stage register composed of sections known as A0 and A1. Two stages are required to hold the address of the micro during the first two stages of the fetch-decode-execute sequence that is performed for each micro. Since there are succeeding micros in each phase of the pipeline at any time, separate address storage facilities are needed to retain the contents of the A register in the case they are needed (for reference purpose) when the micro is decoded. Therefore, the contents of A0 are known as the fetch address, and those of A1 as the decode address. Transfer of the contents of the A0 to A1 occurs concurrently with the move of the fetched micro to the M register.

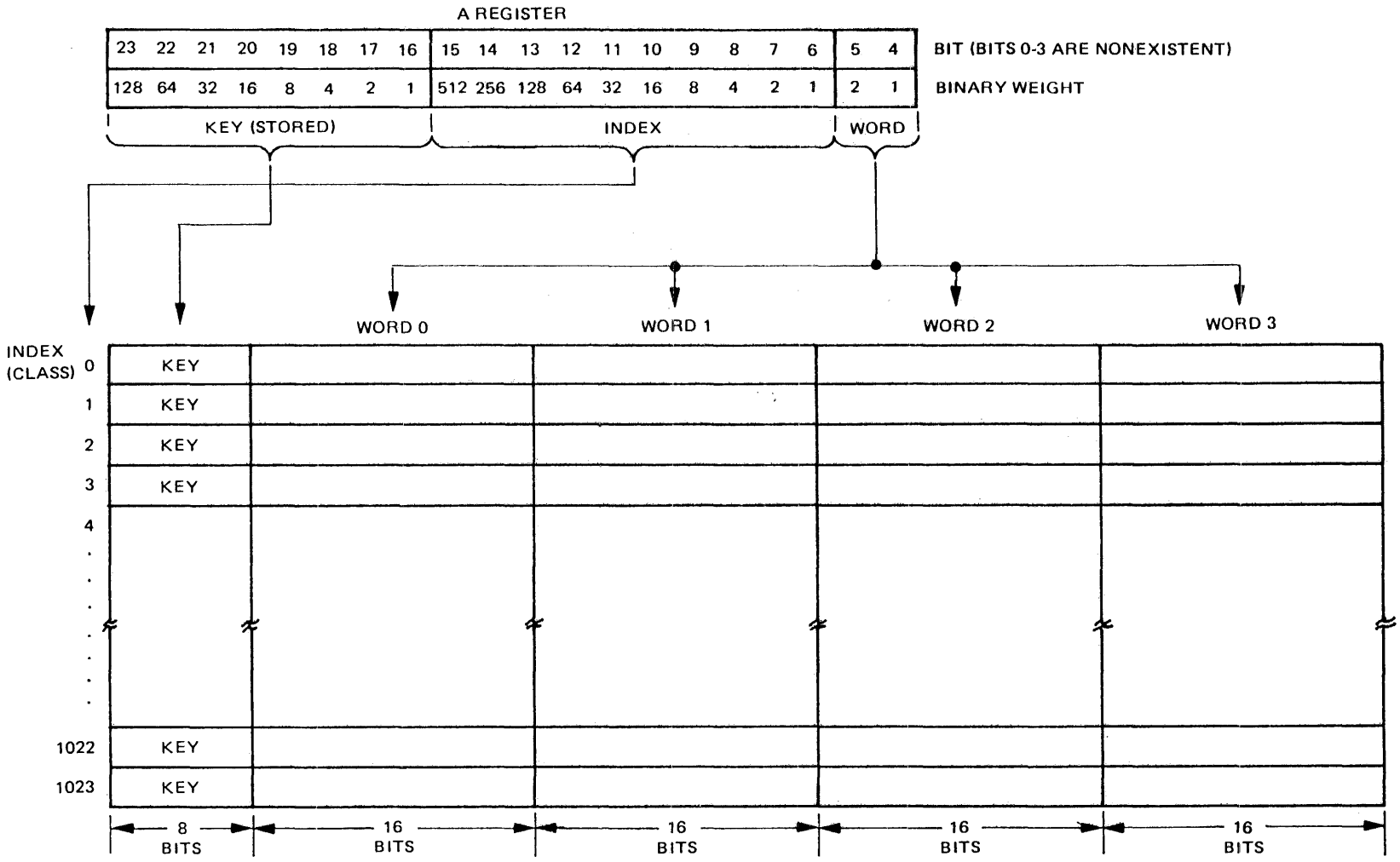
The contents of the A0 register may be modified either by incrementing by 1 or adding/subtracting the contents of a 12-bit or 24-bit field. Incrementing of A0 by 1 occurs automatically with each fetch when the processor is in RUN, NORMAL (continuous) or SINGLE MICRO (step) mode, but does not occur automatically when the processor is in MTR mode. Adding or subtracting some externally originated value is part of a Skip or Branch operation. Wrap-around can occur and is permitted.

The A register can be addressed as either source or sink. The A0 register receives the main exchange, but does not drive it. Conversely, A1 drives the main exchange but does not receive it. A0 and A1 are viewed as a single register from a software standpoint, and micros that manipulate the A register do not reference the A0 or A1 register. When the A register is a source, the rightmost four bits received from A register are always equal to zero (0); when the A register is a sink, the rightmost four bits of the source are lost.

Cache Memory

Cache, a high-speed memory located in the processor, is used to store micros in readiness for fetching. The basic scheme of Cache, including the relationship of the A register to it, is shown in Figure 1-21.

Cache operates to deliver a micro to the M register. The contents of the A register specify the address of the micro, and, if the addressed micro is not present in Cache, an automatic load operation obtains it from S-Memory. Furthermore, each such load operation involves the desired micro plus the three following micros stored in S-Memory. This four-micro load provision is sufficient to ensure that the core of any given program is loaded within a short time after execution. In fact the reentrant nature of most programs allows a 98% success rate in accessing micros from Cache once stabilized operation is attained.



NOTE:

KEY STORAGE IS ACTUALLY 10 BITS IN WIDTH: 8 BITS ADDRESS, 1 VALIDITY BIT AND 1 PARITY BIT. LIKEWISE, MICRO STORAGE IS ACTUALLY 17 BITS WIDE: 16 DATA AND 1 PARITY.

G12218

Figure 1-21. Cache Memory Organization and Relation to A

Cache memory is organized for the exclusive purpose of storing micros and has the capacity to store 4,096 (4K) micros. Cache is addressed associatively; this means that the physical location occupied by a given micro is only partially determined by the contents of the A register. In this scheme the least significant 10 bits of the A register are used to determine the word and index at which the micro is to be stored. The most significant bits of the A register (know as the KEY) are stored along with the micro (actually four micros, corresponding to the same storage configuration in main memory) for identification.

Fetching micros from Cache requires comparison between the contents of the key portion of the A register with the key portion stored at the location specified by the index portion of the A register. On a match of the two values (Hit), the desired micro is gated to the M register for execution. The absence of a match (Miss) causes initiation of the four-micro load routine previously described. The incoming four micros are loaded into the Cache location specified by the A address.

A validity bit accompanies each stored block to indicate whether or not that block contains any valid information (a true validity bit indicates good data). The validity bit is set (=1) when the four micros are loaded from S-Memory. However, validity bits may also be manipulated by writing in Cache from the console (sets validity bit), by executing the Clear Cache micro (5F), or by pressing the CLEAR button.

Parity generation and checking are provided for both the Cache word and Cache key storage. Micro parity is generated in the S-Memory base and checked at the M register. Cache key parity is both generated and checked within the Cache logic. Parity errors of both types are reported by way of the PERP register.

Nano Sequencing

The transition between the decode and execute phases within the processor involves the use of PROMs to generate the control signals (nanoinstructions or nanos) used by the processor for the execute phase. In brief, micros are decoded into a series of control lines used for PROM addressing. These in turn enable the proper PROM outputs for use by the nano register.

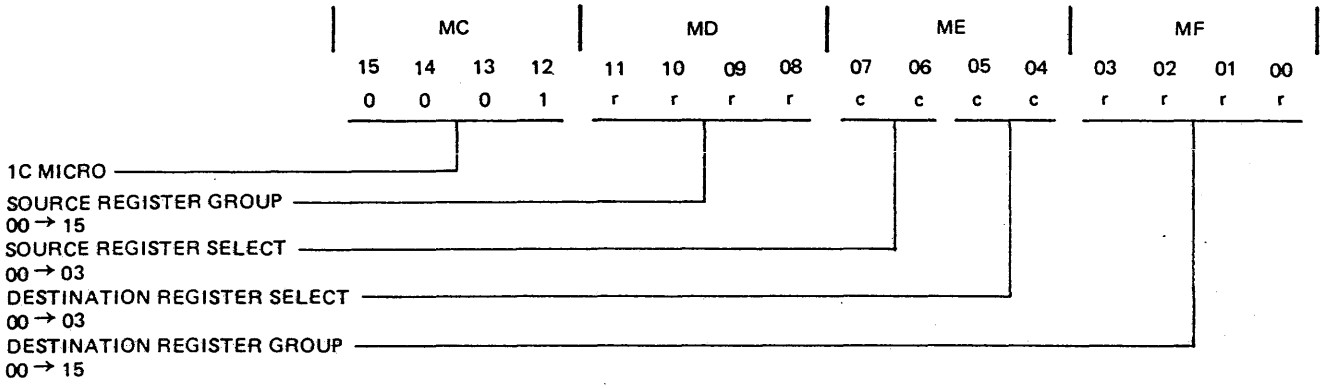
Most micros are decoded into individual nanos, one nano for one micro. There are, however, certain special micros which generate more than one nano, each of which requires one or more clocks to execute. To keep track of micros and the nanos they generate, and to keep the pipeline functioning properly, a nano sequence counter is used.

Sequence counter functions begin within the decode structure before the corresponding nanos are actually generated. Each micro is associated with a predefined count called the last sequence number (LSN). The LSN reflects the number of nanos that are required to fulfill the micro. When a micro with LSN greater than 1 occurs, pipeline operation is suspended in order to allow the micro to complete wthout conflict with upcoming micros. Each nano generated by the micro is associated with a count called the present sequence number (PSN). PSN starts at zero and is counted up with each nano. When PSN = LSN, the micro is complete and the pipeline resumes operation.

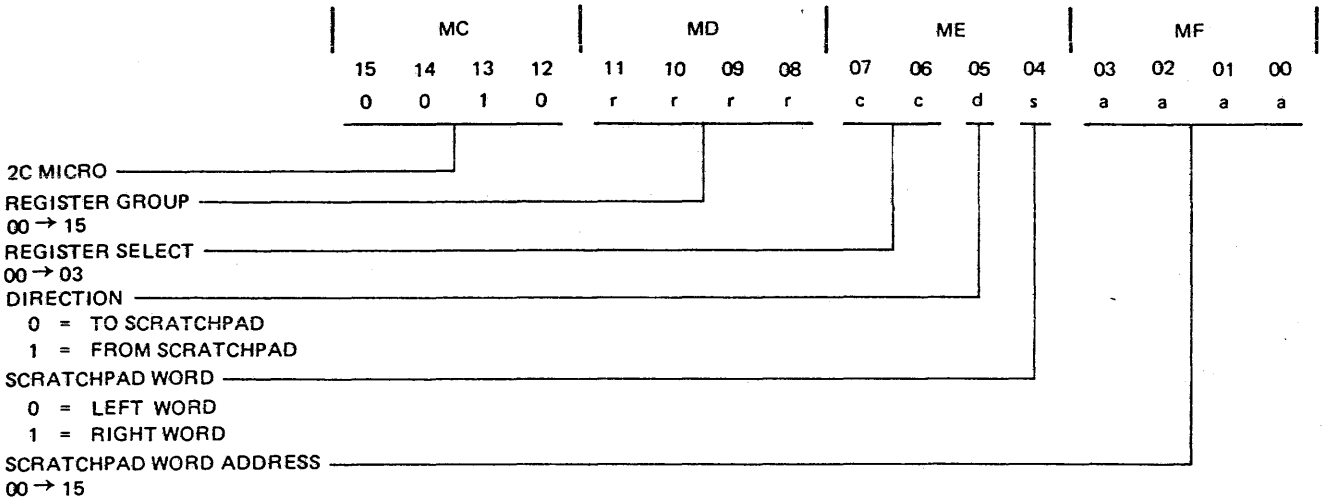
Microinstructions

Microinstructions for the B 1900 are described in the subsections that follow. All micros are tabulated in Figures 1-22, 1-26, 1-27. and 1-28. The micros are grouped by the letter designation, which refers to the pertinent portion of the M register as it appears in the console lamps; the figures appear at the start of each group.

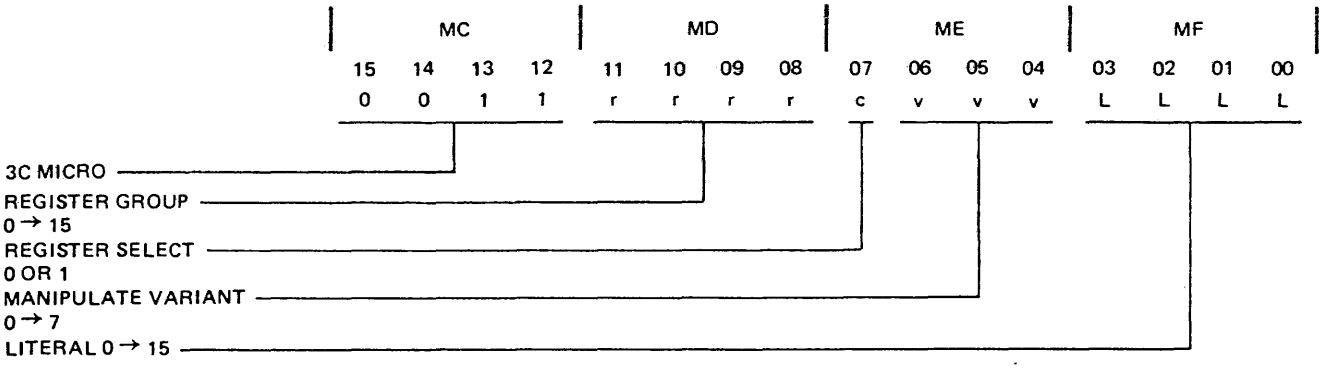
1C Register Move



2C Scratchpad Move



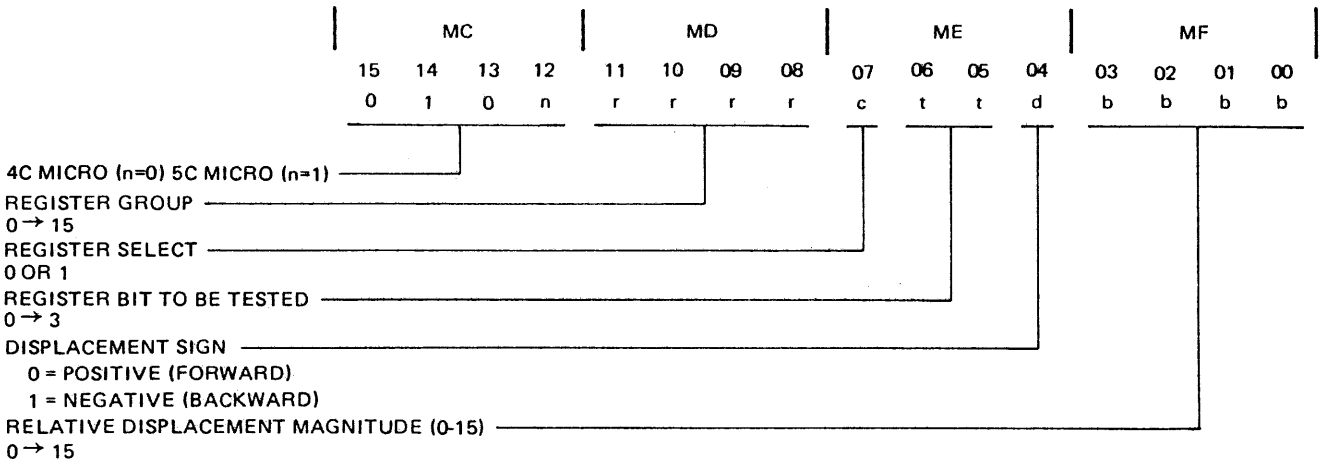
3C 4-Bit Manipulate



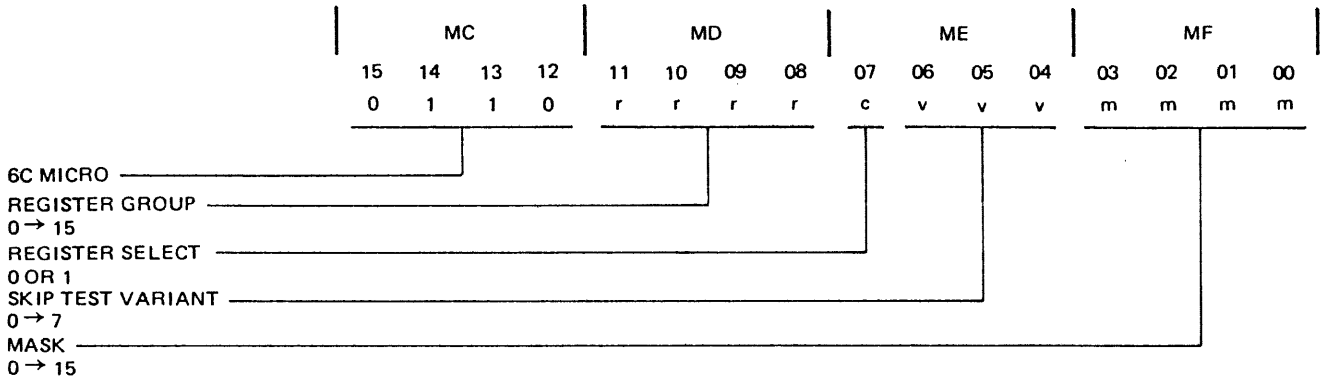
G12219/SHEET 1 OF 4

Figure 1-22. Microinstructions 1C through 15C (Sheet 1 of 4)

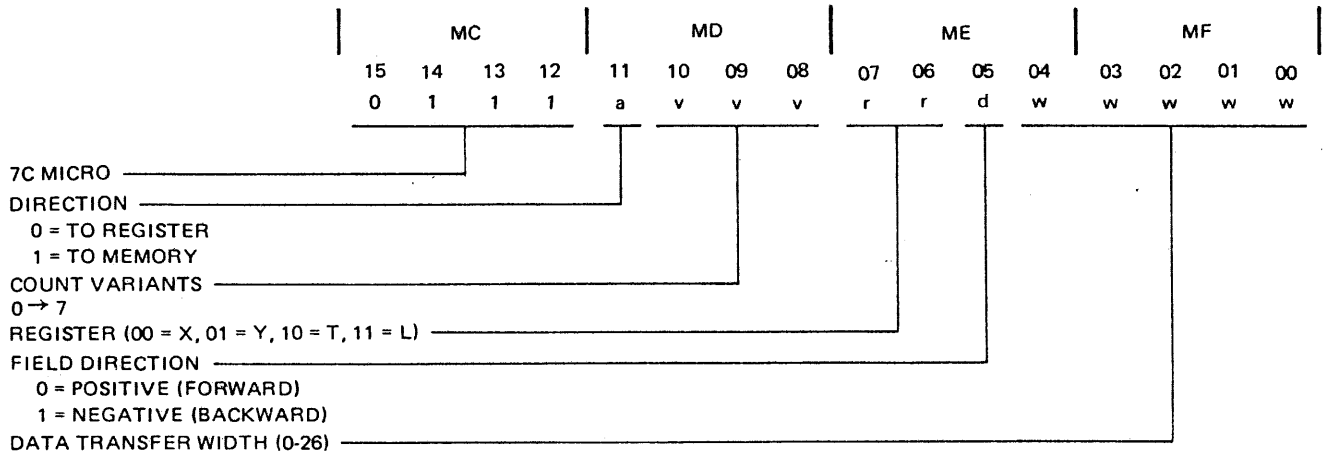
4C/5C Bit Test Relative Branch



6C Skip When



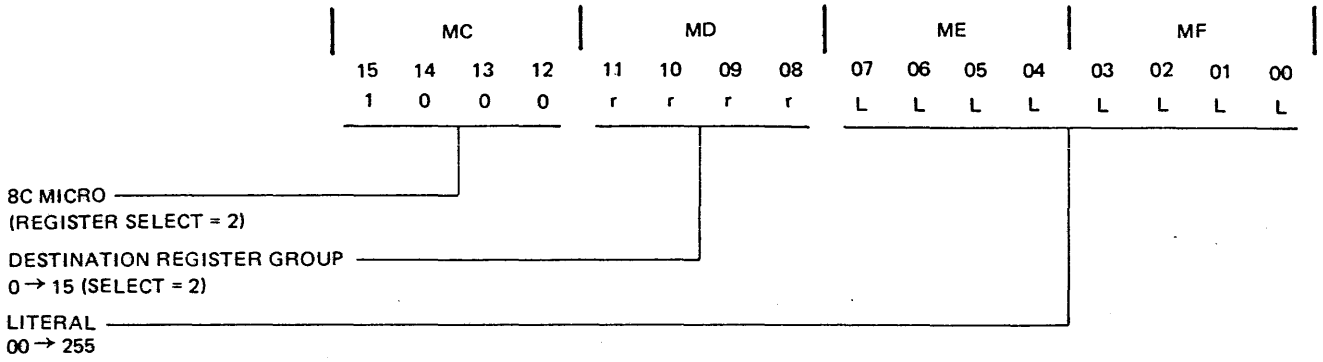
7C Read/Write Memory



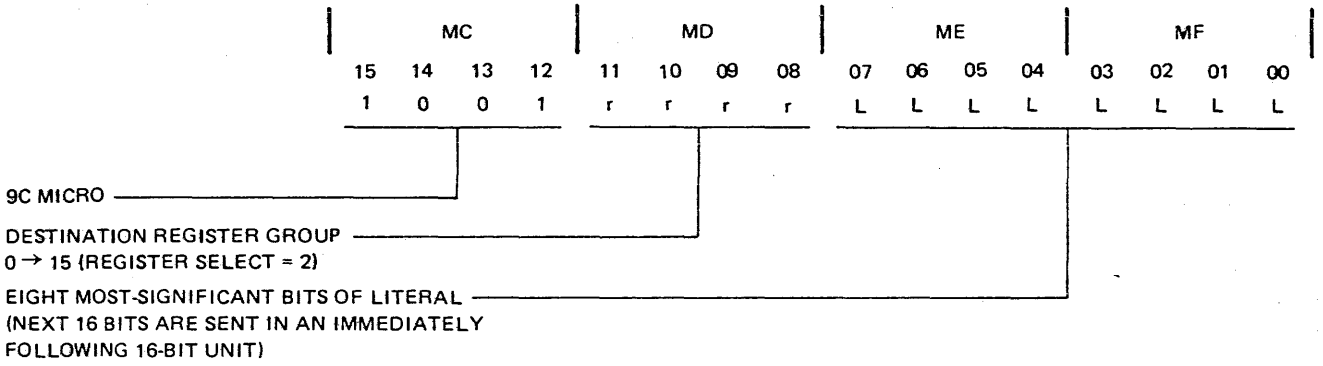
G12219/SHEET 2 OF 4

Figure 1-22. Microinstructions 1C through 15C (Sheet 2 of 4)

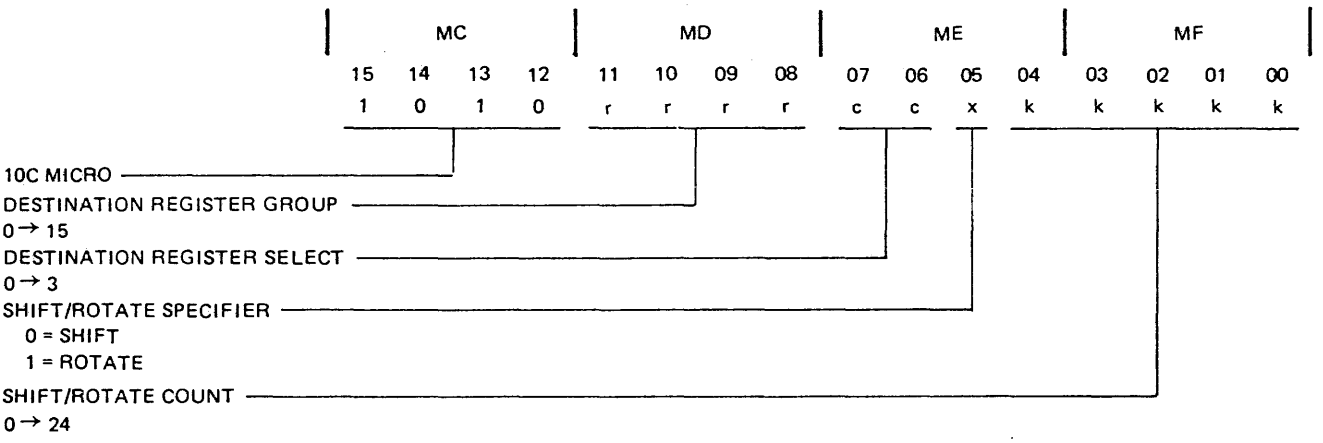
8C Move 8-Bit Literal



9C Move 24-Bit Literal



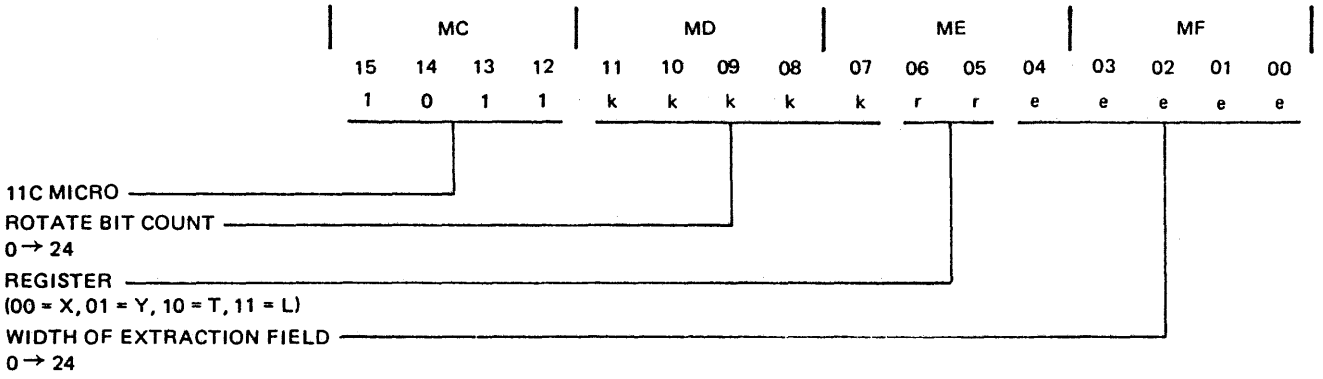
10C Shift/Rotate T Register Left



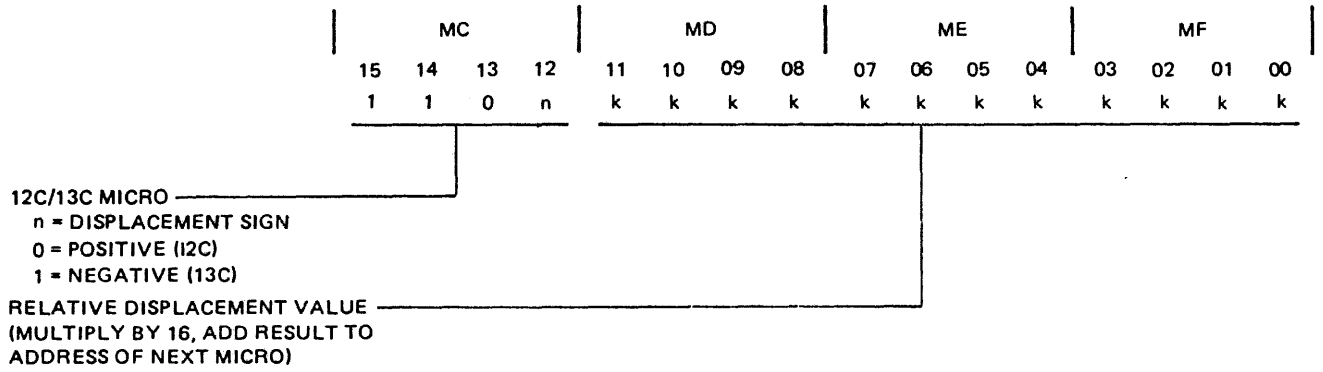
G12219/SHEET 3 OF 4

Figure 1-22. Microinstructions 1C through 15C (Sheet 3 of 4)

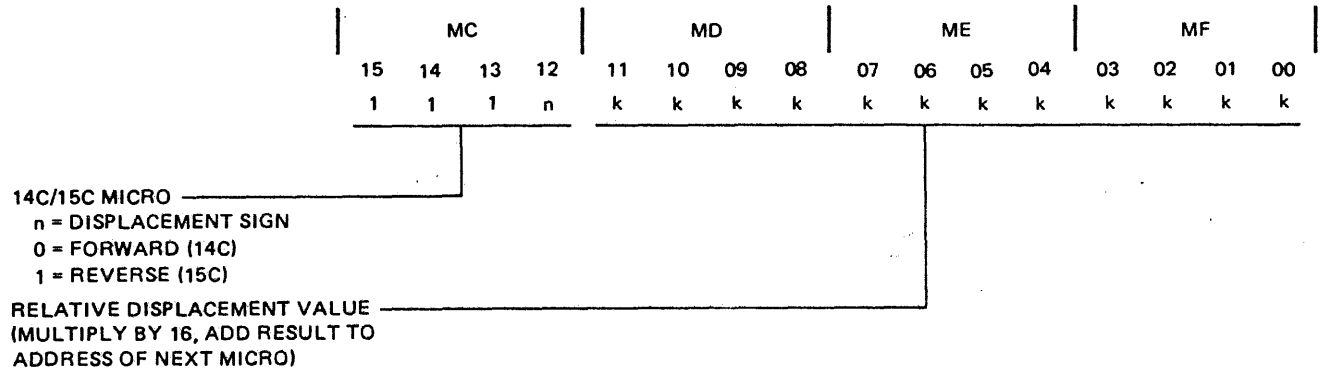
11C Extract from T-Register



12C/13C Branch Relative



14C/15C Call



G12219/SHEET 4 OF 4

Figure 1-22. Microinstructions 1C through 15C (Sheet 4 of 4)

1C Register Move

The contents of the source register are moved to the destination register, with left zero fill or left truncation as appropriate if the two registers differ in size. The source remains unaffected.

If the destination register is M, the source register is bit OR-ed with the next micro fetched to M, and the result is the micro that is executed.

CMND and M are excluded as sources. U is excluded as a source if the processor is in SINGLE MICRO state.

BICN, CMPX, CMPY, CSW, DIFF, FLCN, INCN, MAXS, MSKX, MSKY, SUM, TIME, U, XANY, XEOY, XORY, and XYST are excluded as destinations. When DATA is the source, CMND and DATA are excluded as destinations.

2C Scratchpad Move

The contents of the source are moved to the destination. Source and destination are determined by the direction variant; the doublepad word-half to be used is specified by the scratchpad Word variant. Left zero fill or left truncation is performed as appropriate if source and destination lengths differ. Contents of the source are unchanged. If the destination is M, the scratchpad word-half is bit-ORed with the next micro fetched to M.

CMND, TIME, and M are excluded as sources. U is excluded as a source if the processor is in SINGLE MICRO state.

BICN, CMPX, CMPY, CSW, DIFF, FLCN, INCN, MAXS, MSKX, MSKY, SUM, TIME, U, XANY, XEOY, XORY, and XYST are excluded as destinations.

3C 4-Bit Manipulate

A logical operation is performed with a literal and a 4-bit register as operands. Valid operations are ASSIGN, AND, OR, EXCLUSIVE-OR, SUM, and DIFFERENCE. SUM and DIFFERENCE include the option of skipping the next micro if a carry or borrow results in the most significant position.

The pseudoregisters BICN, FLCN, XYST, XYCN, and INCN are not changed, but a carry or borrow is produced if the appropriate variant is specified, permitting the next micro to be skipped.

Manipulate variant (bits 6, 5, and 4) definitions follow:

vvv	Set register to:
000	The value of the literal.
001	The logical AND of the register and literal.
010	The logical OR of the register and literal.
011	The logical EXCLUSIVE-OR of the register and literal.
100	The binary sum (modulo 16) of the register and literal.
101	The binary sum (modulo 16) of the register and literal; if a carry is produced, skip the next micro.
110	The binary difference (modulo 16) of the register and literal.
111	The binary difference (modulo 16) of the register; if a borrow is produced, skip the next micro.

4C, 5C Bit Test Branch Relative

A specific bit in a register is tested for true/false condition. The bit is specified by the test bit variant (micro bits 6, 5). The state of this bit (false in 4C, true in 5C) determines whether a branch is taken or not; that is, whether the A register (micro address pointer) is changed or left pointing to the next-in-line micro. Branch magnitude is specified by the displacement Value variant (micro bits 3, 2, 1 and 0); branch direction is specified by the forward/backward variant (bit 4).

6C Skip When

Bits in the specified 4-bit register that are referenced by the 4-bit test mask are tested for any ones, all ones, or identity with the mask. The test and the action (skip or no skip, clear) are specified by the skip test variant (vvv).

Test	Action	
Mask vs. Referenced Bits	Skip	Do Not Skip
Any referenced bit = 1	000	100
All referenced bits = 1	001 011*	101 111*
Register = mask	010	110
Mask = 0000	001 011+ 100	000 101+ 111

* Also clear (reset to 0) referenced bits.
\+ Tests but does not clear BICN, XCN, XYST, INCN.

7C Read/Write Memory

The contents of the specified register are moved to memory, or vice versa. The contents of the source are unaffected. The field width of the data to be transferred is specified by the width variant or, if this is 0, by CPL. For widths less than 24, data is right justified and left zero filled in the register on Reads and left truncated in memory on writes.

Field widths (as specified by the variant or CPL) of 25 and 26 are truncated to 24 and, on Writes, error writing and reporting are suppressed. The error correcting code (ECC) is forced to be correct if field width is 25, and forced to be inverted if field width is 26.

The count variant works as follows.

vvv	Count Up	Count Down
000		(No count)
001	FA	–
010	FL	–
011	FA	FL
100	FL	FA
101	–	FA
110	–	FL
111	–	FA & FL

FL is not allowed to underflow.

8C Move 8-Bit Literal

The 8-bit literal in the micro is moved to the destination register in the group specified by REGISTER SELECT = 2. If the move is to a register with greater than 8-bit capacity, the 8 bits are right justified and left zero filled; if the register is smaller, truncation is from the left.

An 8C move to TIME clears TIME to zero. On moves to M, the literal is bit-ORed with the next micro to be executed. The micro source (Cache or S-Memory) is not affected.

9C Move 24-Bit Literal

A 24-bit literal, comprised of the 8-bit literal in the micro and an immediately following 16-bit unit, is moved to the destination register in the group specified by REGISTER SELECT = 2. Truncation, if required, is from the left.

CSW, M, TIME, and all pseudoregisters are excluded as destinations.

10C Shift/Rotate T

The contents of the T register are rotated left 0 to 24 positions as specified by the count variant, and the 24-bit result is moved to the destination register. The T register remains unchanged unless it is the destination. Truncation, if required, is from the left. If count is zero, the value in CPL is used.

BICN, CMPX, CMPY, DIFF, FLCN, INCN, MAXS, MSKX, MSKY, SUM, TIME, U, XANY, XEOY, XORY, XYCN, and XYST are excluded as destinations. Moves to M are bit-ORs with the next micro to be executed. The micro source (Cache or S-Memory) is not affected.

Shifting and rotating are illustrated in Figure 1-23, 1-24, and 1-25.

11C Extract from T

The contents of T are rotated left by the rotate bit count, and the rightmost number of bits specified by the extract bit count are moved, right justified, to a destination register and zero filled on the left. The destination register is X, Y, T, or L. The source (T register) is unaffected unless it is also the destination. A rotate bit count of 24 is equivalent to a count of 0.

Extracting is illustrated in Figure 1-25.

12C, 13C Branch Relative Forward, Reverse

The next micro is fetched from the address obtained by adding the signed relative displacement value to the address of the next-in-line micro. Bit 12 of the OP code specifies positive or negative displacement (forward or backward jump). Note that the relative displacement value is in words (or micros) for convenience.

14C, 15C Call Relative Forward, Reverse

The contents of the A register (the address of the next-in-line micro) are pushed on to the TAS; the signed displacement value is then added to A0 and the next micro is fetched from the address now at TAS.

At the end of the called routine, a Register Move (1C) from TAS to the A register enables the return to the micro following the Call.

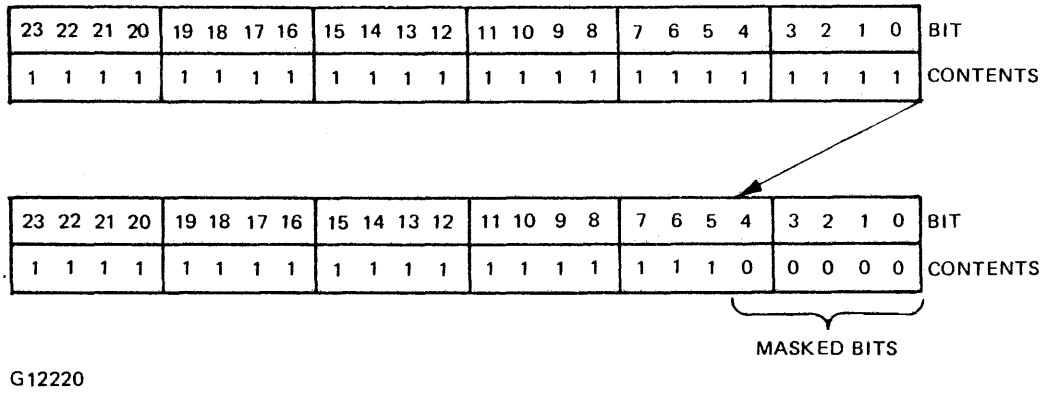


Figure 1-23. Example of Shift

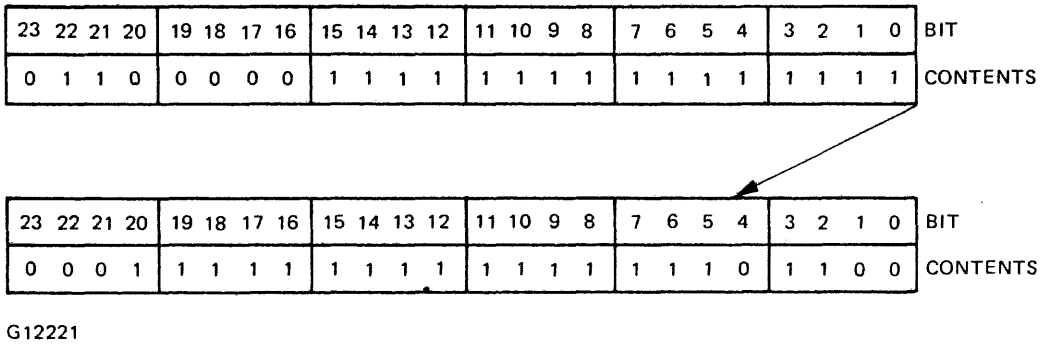


Figure 1-24. Example of Rotation

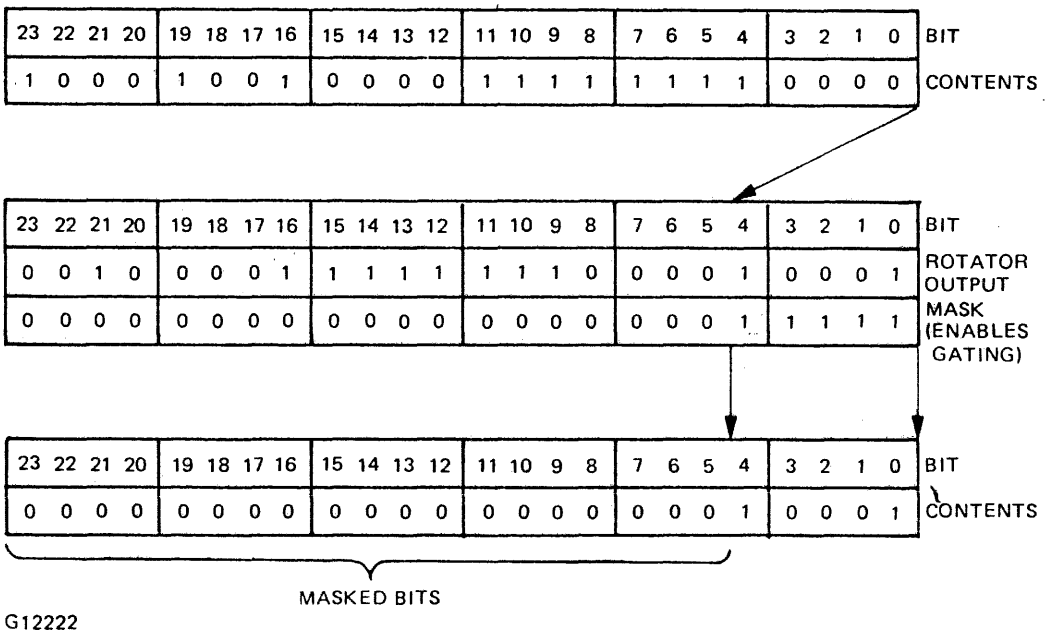
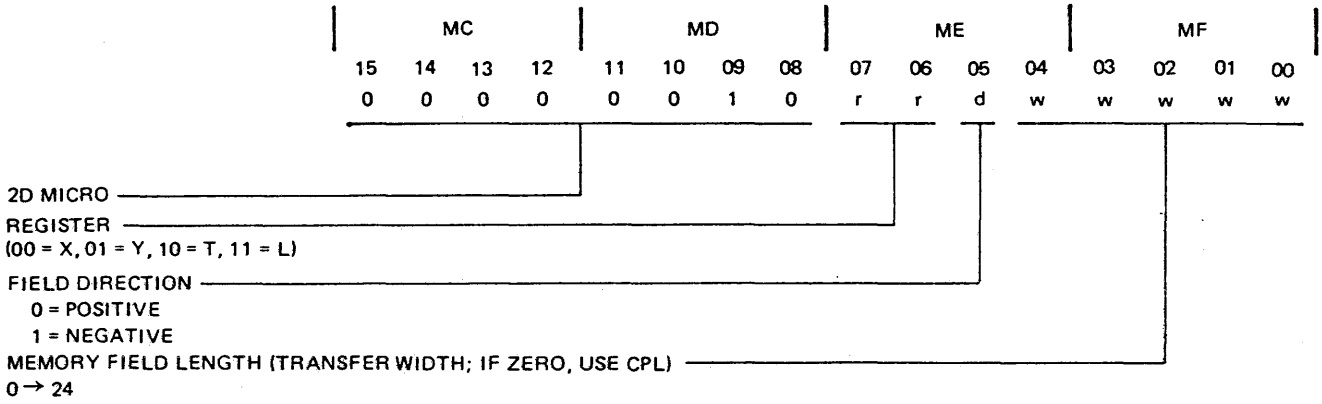
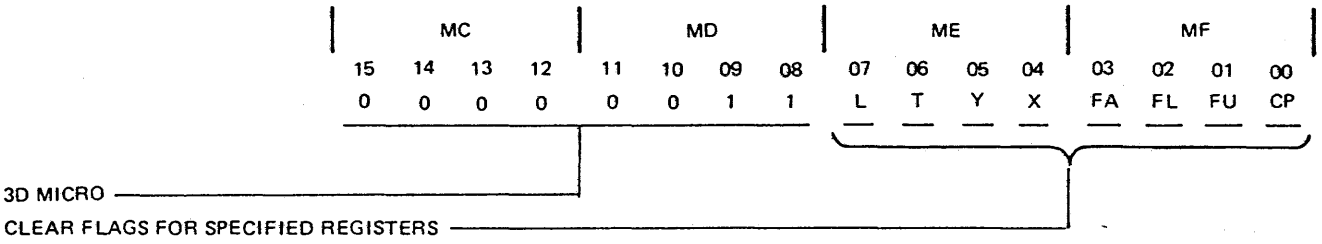


Figure 1-25. Example of Extraction

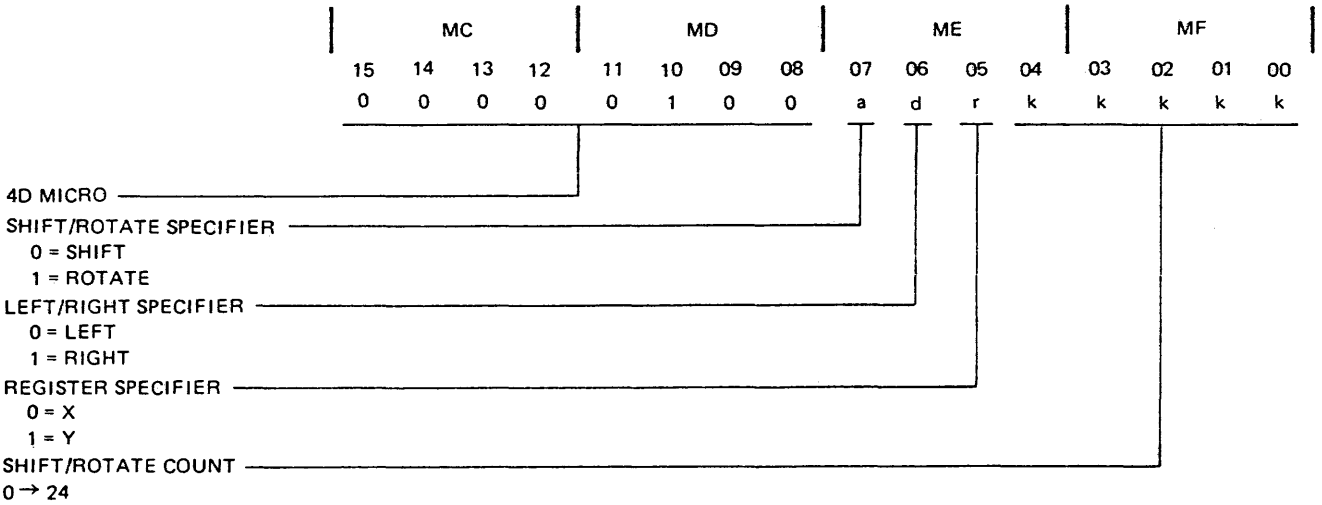
2D Swap Memory



3D Clear Registers



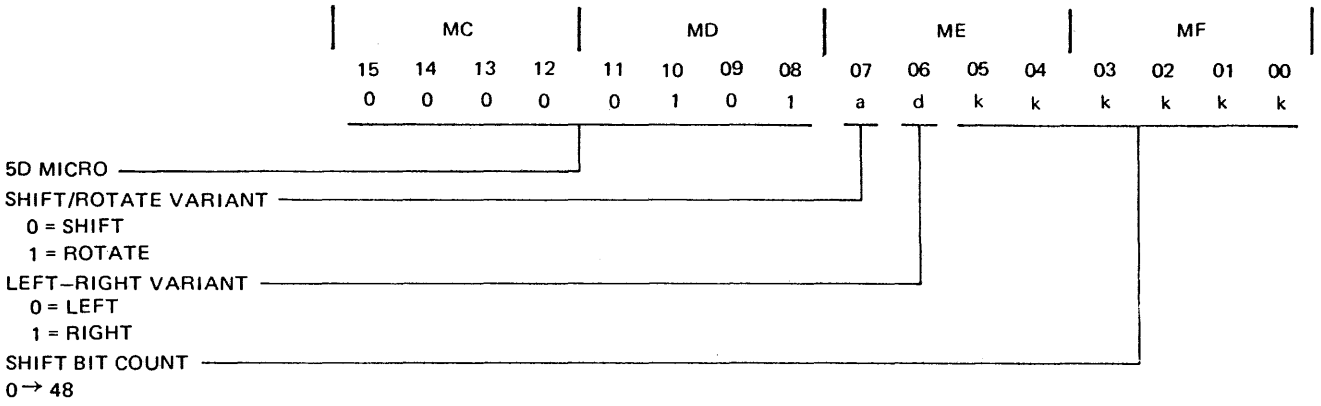
4D Shift/Rotate X or Y



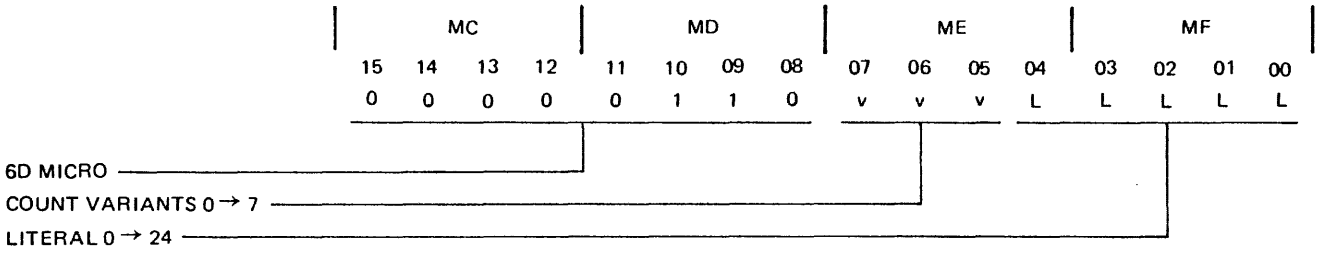
G12223/SHEET 1 OF 3

Figure 1-26. Microinstructions 2D through 12D (Sheet 1 of 3)

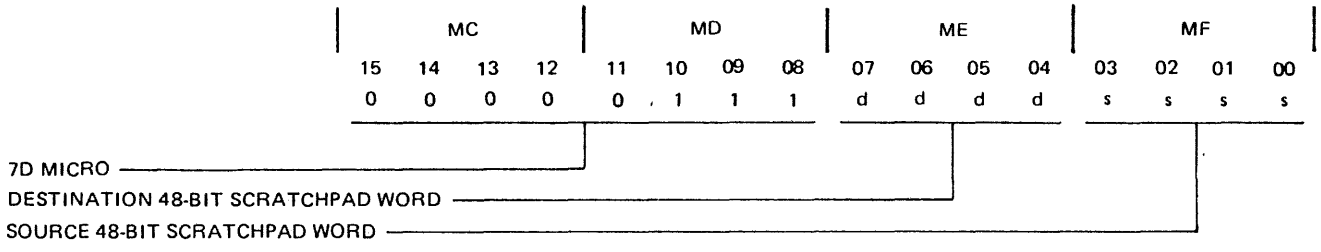
5D Shift/Rotate X and Y



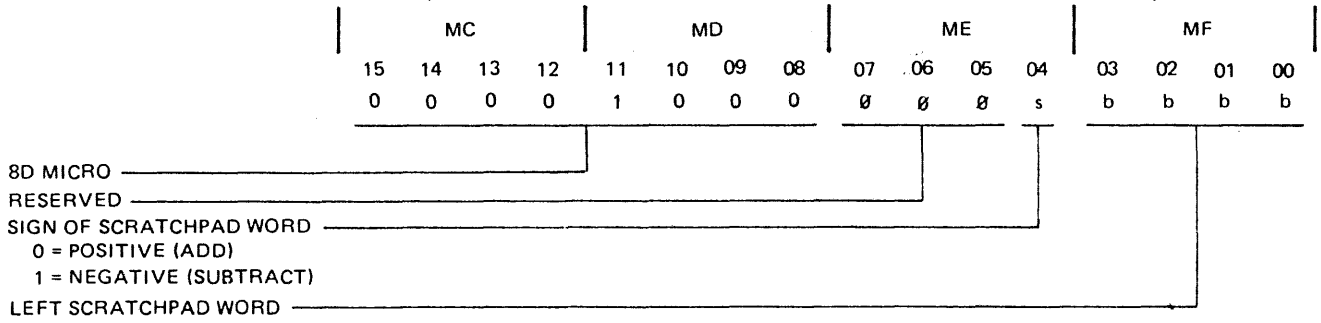
6D Count FA/FL



7D Exchange Doublepad Word



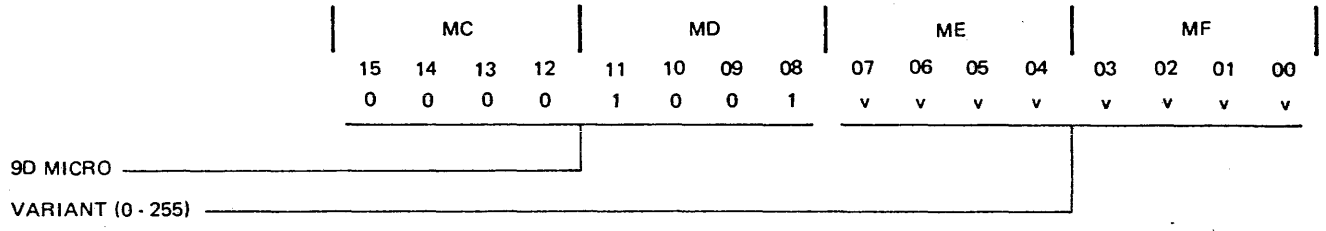
8D Scratchpad Relate FA



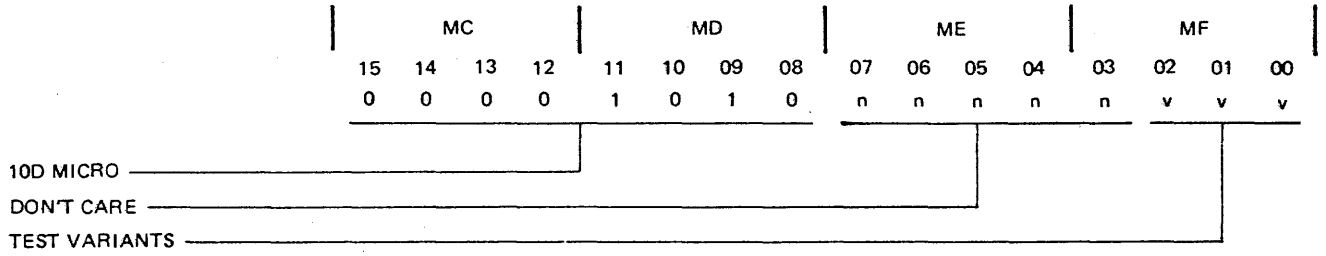
G12223/SHEET 2 OF 3

Figure 1-26. Microinstructions 2D through 12D (Sheet 2 of 3)

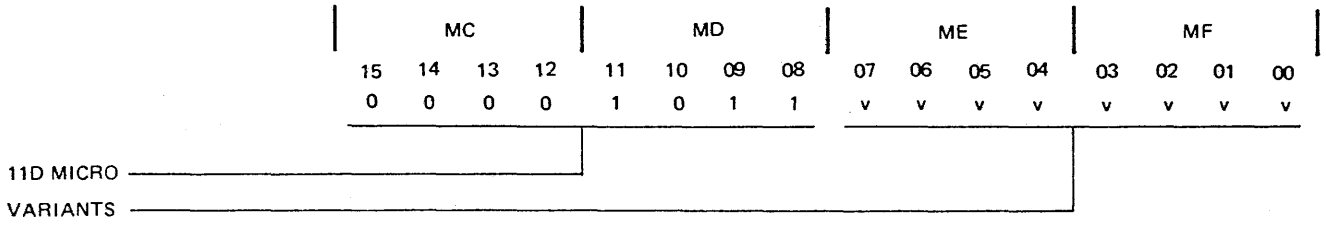
9D Monitor



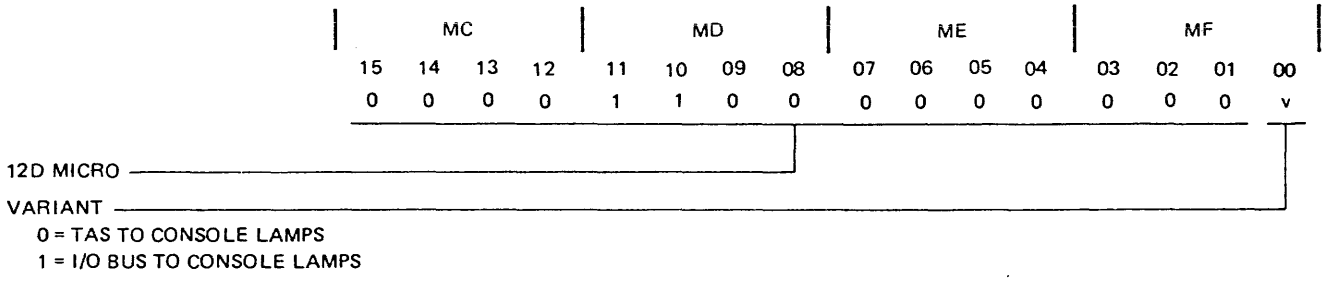
10D Diagnostic Test



11D Diagnostic Read/Write Memory



12D Display TAS or I/O Bus



G12223/SHEET 3 OF 3

Figure 1-26. Microinstructions 2D through 12D (Sheet 3 of 3)

2D Swap Memory

This micro swaps data in S-Memory with data in the specified register (X, Y, L, or T). If transfer width is less that 24, data is right justified and left zero filled in the register and left truncated in memory. If transfer width is zero, the value in CPL is used. The S-Memory address, in FA, is the first bit address if direction is forward (0), or the last bit address plus one if direction is backward (1).

3D Clear Registers

None, some, or all of the following registers are zeroed by setting to ones the appropriate clear flags: L, T, Y, X, FA, FU, CP.

4D Shift/Rotate X or Y

X or Y is shifted left or right by the number of bits specified by the count. On a left shift, truncation is from the left (MSB) and zero fill is on the right (LSB). On a right shift, truncation is from the right, zero fill on the left.

5D Shift/Rotate X and Y

The 48-bit entity formed by X concatenated with Y is shifted/rotated left or right. Note that X is the leftmost (more significant) register. On a shift operation, truncation and zero fill are as stated for the 4D micro.

6D Count FA/FL

FA and/or FL registers may be counted up or down in various combinations by the value of the literal or, if this is zero, by the value in CPL. A literal (or CPL) value greater than 24 is truncated to 24. FA and FL may overflow and wrap around, and FA may underflow and wrap around, without detection. FL may underflow but goes to zero rather than wrapping around. FL underflow is not detected by the processor.

The count variant works as follows.

vvv	Count Up	Count Down
000	(no count)	
001	FA	–
010	FL	–
011	FA	FL
100	FL	FA
101	–	FA
110	–	FL
111	–	FA & FL

7D Exchange F with Doublepad Word

The contents of the 48-bit F register are moved to a temporary area. The 48-bit doublepad word specified by the Source field is moved to F register, and the contents of the temporary area are moved to the doublepad word specified by the destination field. Note that source and destination may be (and generally are) identical; the result is a swap of the contents of the 48-bit double scratch-pad word with the contents of F.

8D Scratchpad Relate FA

The value of FA is added to the signed left scratchpad word. The binary sum is stored in the FA register. Overflow and underflow of the FA register are neither restricted nor detected.

9D Monitor

This micro causes certain signals to go true for one clock period. Inclusion of this micro at various points in the software enables hardware monitors to measure system performance.

The signal MONIT.C1 is true whenever MONITOR is executed. It may be monitored at the Card C, frontplane connector $Y, along with the MOP lines.

10D Diagnostic Test

This micro allows programmatic checks of various data paths within the processor. Actions depend on the value of the variant (bits 2, 1, 0).

vvv	Action
000	Move I/O bus to Y register.
001	Move FA register to Y latch.*
010	Reserved.
011	Move Y latch to Y register.
100	Reserved.
101	Echo S-Memory cable.
110	Test timeout. (A timeout is forced.)
111	Set I/O data path.

NOTE

* The term "Y latch" is a coined word used to describe the bidirectional hardware gates isolating Exchange X from Exchange Y. These gates can function as latches for diagnostic operation only, and can be echoed. To echo the Y latch, three 10D micros must be executed sequentially with the processor in NORMAL (not SINGLE MICRO), and no fetches, misses or halts between the micros. The 10D micro variants are, in sequence: 111, 001, and 011. The procedure returns zeroes on the B 1905.

11D Diagnostic Read/Write Memory

This micro performs several distinct types of diagnostic operations, specified by the variant (Table 1-6), to allow low-level testing of the processor/memory interface and memory control. The destination register is always the Y register and the information is returned in the format shown in Table 1-7.

See the subsection titled S-Memory for more information.

Table 1-6. Variant Definitions, 11D Micro

7 6 5 4 3 2 1 0	Significance
0 0 0 0 0 0 0 0	READ 22-bit word (16 data + 6 ECC) to Y
r r 0 0 0 1 0 0	WRITE 22-bit word from X, Y, T or L to memory
r r 0 0 1 0 0 1	ECHO write data from X, Y, T or L to Y
0 0 0 0 1 0 1 0	ECHO modified address forward from FA to Y
0 0 1 0 1 0 1 0	ECHO modified address backward from FA to Y
0 0 0 0 1 0 1 1	READ AND CLEAR error log to Y
0 0 0 1 1 0 0 0	READ port data latch to Y
0 0 0 1 1 0 0 1	READ port adapter interface (always false) to Y
0 0 0 1 1 0 1 0	READ host adapter multiplexor (always false) to Y
0 0 0 1 1 0 1 1	READ host adapter tristate bus (always true) to Y
r r 1 1 1 0 0 1	ECHO through Port 2 from X, Y, T or L to Y
r r 1 1 1 0 1 0	ECHO through Port 3 from X, Y, T or L to Y
r r 1 1 1 0 1 1	ECHO through Port 4 from X, Y, T or L to Y

NOTES

rr is the source register indicator: 00 = X, 01 = Y, 10 = T, 11 = L.

Bit patterns not listed are undefined.

Table 1-7. Y Register Format for Diagnostic Data

Bits	Meaning
23-18	ECC bits
17	Filler bit (always zero)
16	Hardware-generated parity bit (read only; ignored on write)
15-0	Data

READ 22-Bit Word to Y

All 22 bits (16 data and 6 ECC) of the memory word designated by the address in FA are read to Y. (Accesses are always forward on stack boundaries.) Single-bit errors are not corrected but all errors are reported. If a single-bit error is detected within the 16 data bits, the odd parity bit is complemented causing even parity.

WRITE 22-Bit Word to Memory

Twenty-two bits (16 data, 6 ECC) are written from the specified register into the memory address specified by FA. (Accesses are always forward on stack boundaries.) The source register must be in the format shown in Table 1-7; errors are neither corrected nor reported.

ECHO Write Data

Data (24 bits) is sent from the source register to the MBU which returns it to the Y register.

ECHO Modified Address

A 24-bit address is sent from FA to the MBU which loads certain bits of the address into the Error Log register (ELOG) after adding or subtracting 16, returns the contents of the ELOG to Y, and clears the ELOG. (ELOG format is shown in Table 1-8 and the related syndrome summary in Table 1-9.)

The LSB of FA is used as a flag; if FA LSB = 1, the MBU refresh cycle is disabled. Thus, the processor, after writing to a given memory location, can disable the refresh cycle, delay for any period of time, and then read the same location to check for data lost.

READ AND CLEAR Error Log

The ELOG is read to Y. ELOG is then cleared.

READ Port Data Latch

The current contents of the port data latch are read to Y. This information is the most recent of the following: (1) a memory address from the last port device write cycle, (2) read data from the last port device read cycle, or (3) echo data from the last echo cycle.

READ Port, Host Adapters

The port adapter interface (all bits false), the host adapter's multiplexor (all bits false), or the host adapter's tristate bus (all bits true) is read to Y.

ECHO Through Port

Data (24 bits) from the designated source is sent to be captured in the host adapter's data latch. The data is then returned to Y via one of four paths, one entirely within the host adapter and the other three through one of the three possible ports.

Table 1-8. ELOG Format

Bit	Meaning (if set unless otherwise noted)
23	Duplicate error of the same type.
22	Uncorrectable error during processor access.
21	Uncorrectable error during nonprocessor access.
20	Correctable single-bit error.
19	0 = read operation, 1 = write operation.
18-17	Always 11 (MBU 7/8).
16	Reserved (always zero) for 64 K RAM.
15-6	Address bits in sequence: 19, 18, 17, 16, 23, 22, 5, 4, 21, 20.
5-0	Syndrome S1-S6 (see Table 1-9).

NOTE

On multiple errors, priority is such that information on error location and syndrome for an uncorrectable processor error replaces that of an uncorrectable nonprocessor error, which, in turn, replaces that of a correctable single-bit error.

Table 1-9. Syndrome Summary

Syndrome No.	Syndrome Pattern*	Interpretation (Error Type)	Report	Correct
0	000000	No error	No report	–
1	100000	ECC bit	Single	ECC bit 6
2	010000	ECC bit	Single	ECC bit 5
3	110000	Multiple even	Multiple	–
4	001000	ECC bit	Single	ECC bit 4
5	101000	Multiple even	Multiple	–
6	011000	Multiple even	Multiple	–
7	111000	Multiple odd	Multiple	–
8	000100	ECC bit	Single	ECC bit 3
9	100100	Multiple even	Multiple	–
10	010100	Multiple even	Multiple	–
11	110100	Data bit	Single	Data bit 15
12	001100	Multiple even	Multiple	–
13	101100	Data bit	Single	Data bit 14
14	011100	Data bit	Single	Data bit 13
15	111100	Multiple even	Multiple	–
16	000010	ECC bit	Single	ECC bit 2
17	100010	Multiple even	Multiple	–
18	010010	Multiple even	Multiple	–
19	110010	Data bit	Single	Data bit 12
20	001010	Multiple even	Multiple	–
21	101010	Multiple odd	Multiple	–
22	011010	Data bit	Single	Data bit 11
23	111010	Multiple even	Multiple	–
24	000110	Multiple even	Multiple	–
25	100110	Data bit	Single	Data bit 10
26	010110	Data bit	Single	Data bit 9
27	110110	Multiple even	Multiple	–
28	001110	Data bit	Single	Data bit 8
29	101110	Multiple even	Multiple	–
30	011110	Multiple even	Multiple	–
31	111110	Multiple odd	Multiple	–
32	000001	ECC bit	Single	ECC bit 1
33	100001	Multiple even	Multiple	–
34	010001	Multiple even	Multiple	–
35	110001	Data bit	Single	Data bit 7
36	001001	Multiple even	Multiple	–
37	101001	Data bit	Single	Data bit 6
38	011001	Data bit	Single	Data bit 5
39	111001	Multiple even	Multiple	–
40	000101	Multiple even	Multiple	–
41	100101	Data bit	Single	Data bit 4
42	010101	Multiple odd	Multiple	–
43	110101	Multiple even	Multiple	–
44	001101	Data bit	Single	Data bit 3
45	101101	Multiple even	Multiple	–
46	011101	Multiple even	Multiple	–
47	111101	Multiple odd	Multiple	–

Table 1-9. Syndrome Summary (Cont)

Syndrome No.	Syndrome Pattern*	Interpretation (Error Type)	Action Report	Correct
48	000011	Multiple even	Multiple	–
49	100011	Data bit	Single	Data bit 2
50	010011	Data bit	Single	Data bit 1
51	110011	Multiple even	Multiple	–
52	001011	Data bit	Single	Data bit 0
53	101011	Multiple even	Multiple	–
54	011011	Multiple even	Multiple	–
55	111011	Multiple odd	Multiple	–
56	000111	Multiple odd	Multiple	–
57	100111	Multiple even	Multiple	–
58	010111	Multiple even	Multiple	–
59	110111	Multiple odd	Multiple	–
60	001111	Multiple even	Multiple	–
61	101111	Multiple odd	Multiple	–
62	011111	Multiple odd	Multiple	–
63	111111	Multiple even or memory absent	Multiple	–

NOTES

* ELOG bit sequence (left to right): 0-1-2-3-4-5; syndrome pattern (left to right): S1-S2-S3-S4-S5-S6.

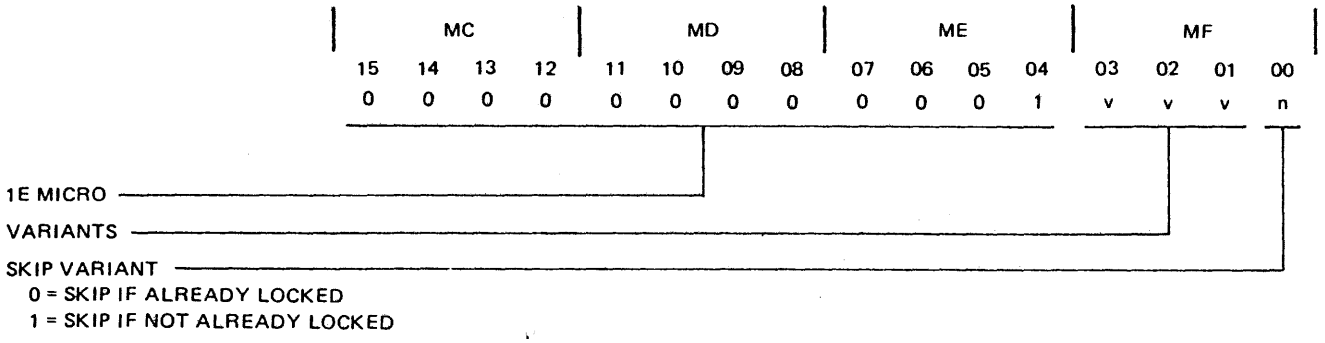
Multiple even or odd means an even or odd number of bits in error were detected.

12D Display TAS or I/O Bus

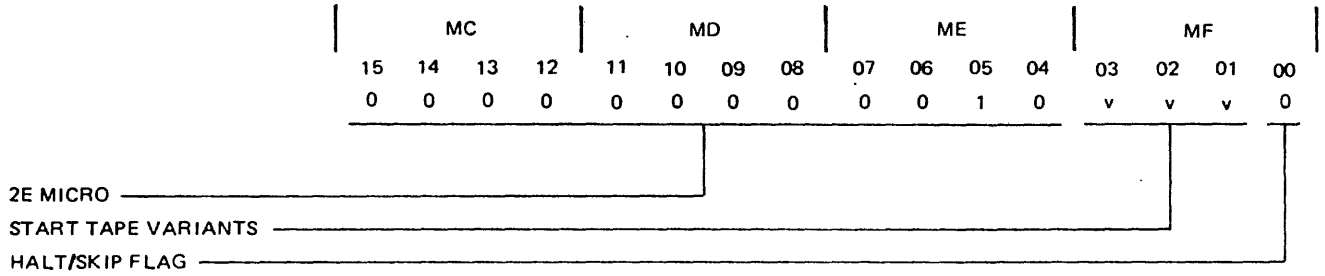
This micro is active when either TAS or CMND is selected via the D/M panel with the processor in halt. When TAS is selected, this micro executes to display the contents of TAS, but no pushes or pops of the A-Stack occur. When CMND is selected, the micro executes to display the I/O bus, but CA and RC are inhibited.

The micro is not used in microprograms; if it is specified, it executes as a no-op.

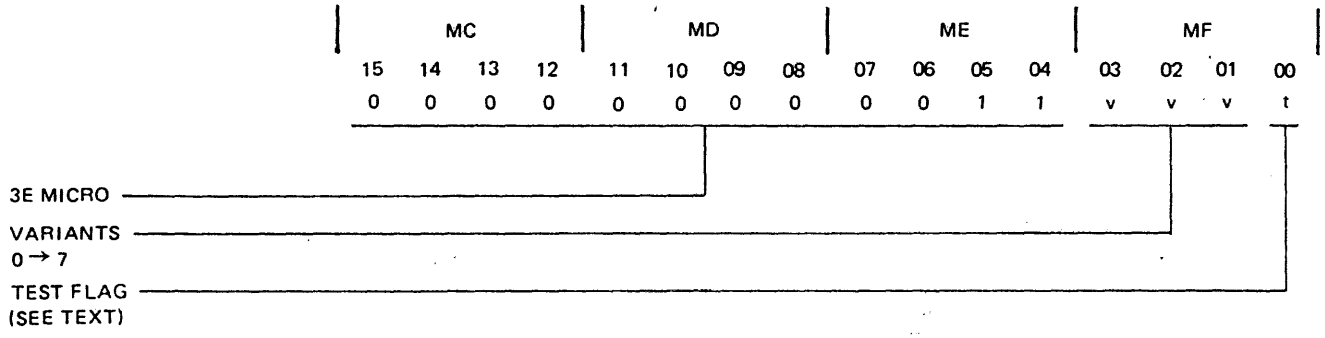
1E Dispatch



2E Cassette Control



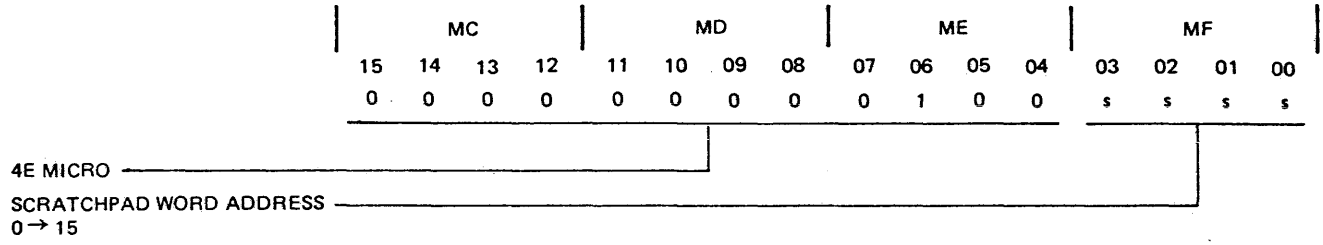
3E Bias



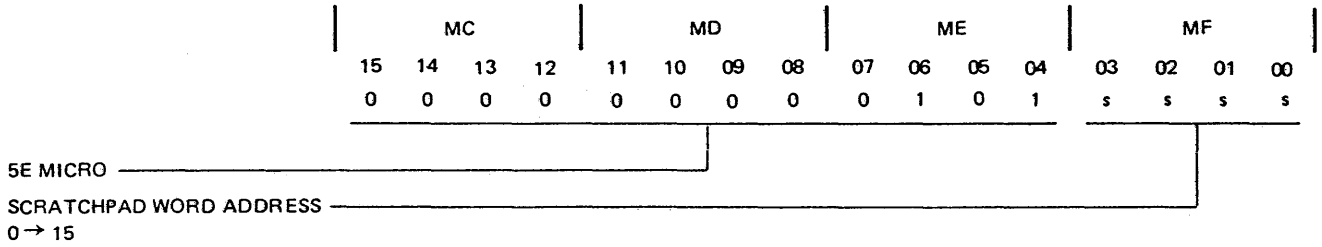
G12224/SHEET 1 OF 2

Figure 1-27. Microinstructions 1E through 7E (Sheet 1 of 2)

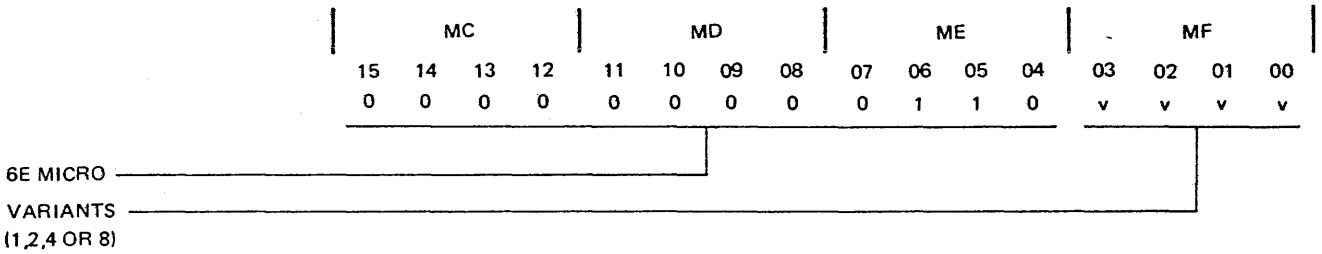
4E Store F in Doublepad Word



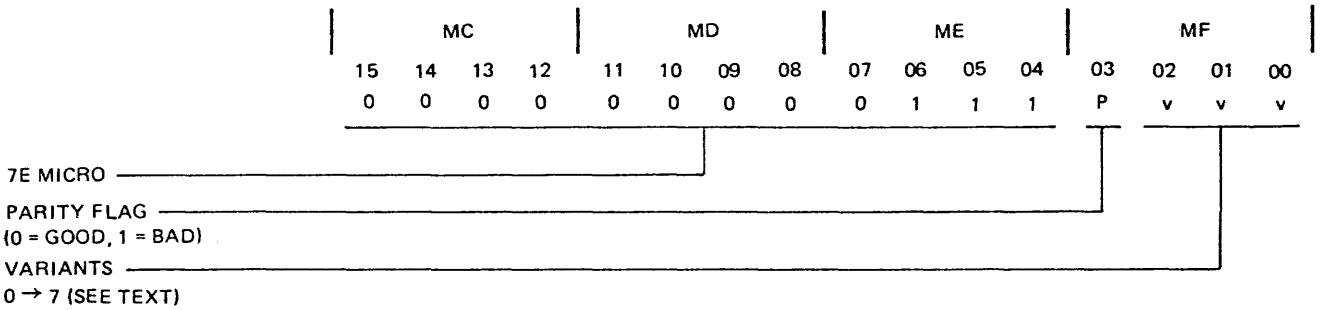
5E Load F From Doublepad Word



6E Carry Flip-Flop Manipulate



7E Read/Write Cache



G12224/SHEET 2 OF 2

Figure 1-27. Microinstructions 1E through 7E (Sheet 2 of 2)

1E Dispatch

This micro allows the processor to send and receive interrupt information from ports other than port 7 (the "soft I/O" port). Since all ports share a common interrupt system, the processor needs to get control of the interrupt system with a successful Dispatch Lockout before executing a Dispatch Write.

A B 1900 direct-connect system is wired such that Dispatch options are limited: (1) Lockout (000) always skips; (2) Write Low (001) always sets INCN bit 3 (missing port device); (3) Read and Clear (011) always resets INCN bit 3; (4) no changes occur in the L and T registers; and (5) INCN bit 3 can can change, as noted in (2) and (3), but INCN bits 2, 1, and 0 are always strapped false.

The meanings of the variants follow. (Variants 101, 110, and 111 are undefined.)

000 Lockout. (The skip flag, bit 0, applies only to this variant.) The lockout flip-flop in the host adapter's Dispatch register is set. This allows or prohibits skipping of the next micro, depending on the setting of the skip flag. If the skip flag = 1, skipping is allowed if Lockout is successful. (Successful Lockout means that the Lockout bit was not previously set.)

001/100 Write Low/Write High. The Write High and Write Low variants set the Lockout and Interrupt bits in the host adapter's Dispatch register. (The bits represent flip-flops in the host adapter.) The contents of the L register are written into absolute memory locations 0 through 23. The least significant 7 bits of the T register, representing the destination port and channel, are written into an identification register in the host adapter. Write High causes a High Interrupt flip-flop in the host adapter to be set; Dispatch Read resets this condition.

010 Read. The Read variant stores the contents of absolute memory locations 0 through 23 into the L register and the contents of a port and channel identification register in the host adapter into bits 6 through 0 of the T register. (Bits 23 through 7 of T are left unchanged.)

011 Read and Clear. With the Read and Clear variant, all functions of a Read are performed. In addition, the four flip-flops in the host adapter comprising the INCN pseudoregister are cleared.

2E Cassette Control

This micro controls the start/stop action of the cassette tape. The skip flag (bit 0) is not operative; 0 and 1 produce identical results. Variant (bits 3, 2, 1) definitions follow (011, 100, and 111 are reserved).

vvv	Meaning
000	Start tape.
010	Stop tape if X $\neq$ Y.
101	Stop tape.
110	Stop tape if X = Y.
111	Rewind.

NOTE

Software must stop the tape prior to rewind and wait at least 100 ms (two timer interrupts) before issuing a rewind. Otherwise, physical damage to the tape can occur.

3E Bias

This micro sets values into the CPU and CPL registers and skips or does not skip the micro following according to the rules given in Table 1-10.

Table 1-10. Skip/No-Skip Rules for Micro 3E (BIAS)

vvv	4 or 8 in FU	4 or 8 in SFL	Set CPU to	Set CPL to	Test Flag = 1.
000	Yes	–	1	FU	No skip
000	No	–	0	FU	Skip possible **
001	Yes	–	1	*FL, 24	Skip possible **
010	No	Yes	1	*SFL, 24	Skip possible **
010	No	No	0	*SFL, 24	Skip possible **
011	Yes	–	1	*FL, SFL, 24	Skip possible **
011	No	–	0	*FL, SFL, 24	Skip possible **
100	–	–	NC	NC	Undefined
101	Yes	–	1	*FL, CPL, 24	Skip possible **
101	No	–	0	*FL, CPL, 24	Skip possible **
110	–	–	NC	NC	Undefined
111	–	–	NC	NC	Undefined

NOTES

* Set to smallest.
** Skip next micro if Test Flag = 1 and final contents of CPL are not equal to 0.
NA = Not applicable.
NC = No change.

4E Store F Into Doublepad Word

The contents of the F register are moved into the specified doublepad word, but are not changed in F. (Doublepad refers to the 48-bit word formed from the concatenation of the Scratchpad A and B words.)

5E Load F From Doublepad Word

The contents of the specified doublepad word are moved into the F register but are not changed in the scratchpad.

6E Set Carry Flip-Flop

CYF, the carry flip-flop, is set to either an unconditional zero or one, or to the carry-out or borrow-out flip-flops CYL or CYD. There are three inputs to the 24-bit ALU:, X, Y, and CYF; thus, the latter two cases provide means for feeding any overflow past the most significant unit on one calculation and into the ALU on the next calculation.

7E Read/Write Cache Memory

This micro is one of the two ways to access Cache memory. (The other way is the 5F micro.) The variants (vvv) specify the actions taken and are explained in the subsections that follow. Variants 000-011 are executed from the console only, by pressing LOAD after setting REGISTER SELECT and REGISTER GROUP as required. Variants 100-111 are executed via software.

vvv = 000 and 001: Console Write Cache (RS = MEMORY, RG = CW)

A Cache word consisting of (1) the 8-bit Cache key field from the A register, (2) one key parity and one Cache validity bit, both hardware-generated, and (3) 17 data bits (data plus odd parity) from the console switches, is written into Cache at the location specified by the Cache index and Cache word fields of the A register.

The A register is not automatically incremented. It can be incremented manually by selecting it via REGISTER SELECT and REGISTER GROUP and pressing INC.

vvv = 010: Console Read Cache Data (RS = MEMORY, RG = CMR)

If there is a hit, the Cache micro (including parity; not checked) at the location specified by the A register is read into the leftmost 17 console lamps (bits 16-0). Key parity is checked; on a key parity error, PERP bit 2 is set which, in turn, sets CD bit 3 and the ERROR lamp on the D/M panel.

The A register is not automatically incremented. It can be incremented manually by selecting it via REGISTER SELECT and REGISTER GROUP and pressing INC.

vvv = 011: Console Read Cache Key (RS = MEMORY, RG = CKR)

The Cache key, its parity bit, the hit status bit and the validity bit of the Cache location specified by the A register are read into the console lamps in the following format (23-0 left to right):

Bit	Contents
23	0
22	Key parity
21	0
20	Hit status
19-9	All 0
8-1	Cache Key
0	Validity bit

If there is a key parity error, PERP bit 2 is set which, in turn, sets CD bit 3 and the D/M panel ERROR lamp.

The A register is not automatically incremented. It can be incremented manually by selecting it via REGISTER SELECT and REGISTER GROUP and pressing INC.

vvv = 100 and 101: Diagnostic Write Cache Micro

This variant of the 7E micro can be used to load four micros into Cache using a preloaded address in FA for the Cache location. FA bits 23-16 hold the Cache key and bits 15-6 hold the Cache index. Micro operation is as follows:

1. The current contents of the A register are stored in A1.
2. The contents of FA are moved to A.
3. The key field (bits 23-16) is moved to the Cache key field, the Cache validity bit is set to indicate the presence of a valid micro, and Key parity is written.
4. In sequence, the low-order 17 bits of registers X, Y, T, and L are moved to Cache words 0, 1, 2, and 3, respectively.
5. The temporarily stored contents of A are returned to A.

vvv = 110: Diagnostic Read Cache Micro

A micro (16 bits plus parity) is read from a Cache location (obtained from FA) to the Y register. Sequence of operation is as follows:

1. The current contents of the A register are stored in A1.
2. The contents of FA are moved to A.
3. The Cache word now specified in the A register is moved to the low-order 17 bits of the Y register. Key parity is checked; an error sets PERP bit 3 and, in turn, CD bit 2 and the D/M panel ERROR lamp. No halt occurs.
4. The temporarily stored contents of A are restored to A.

vvv = 111: Diagnostic Read Cache Key

The Cache key, its parity bit, the hit status bit and the validity bit of a Cache location specified by FA are read into the Y register in the same format as shown for reading into the console lamps, under vvv = 011: Console Read Cache Key.

If there is a key parity error, PERP bit 2 is set which, in turn, sets CD bit 3 and the D/M panel ERROR lamp. No processor halt occurs. The sequence of operations follows:

1. The current contents of the A register are stored in A1.
2. The contents of FA are moved to A.
3. The Cache key and the parity, hit status, and validity bits of the Cache location specified are moved to Y in the format given above.
4. The temporarily stored contents of A are returned to A.

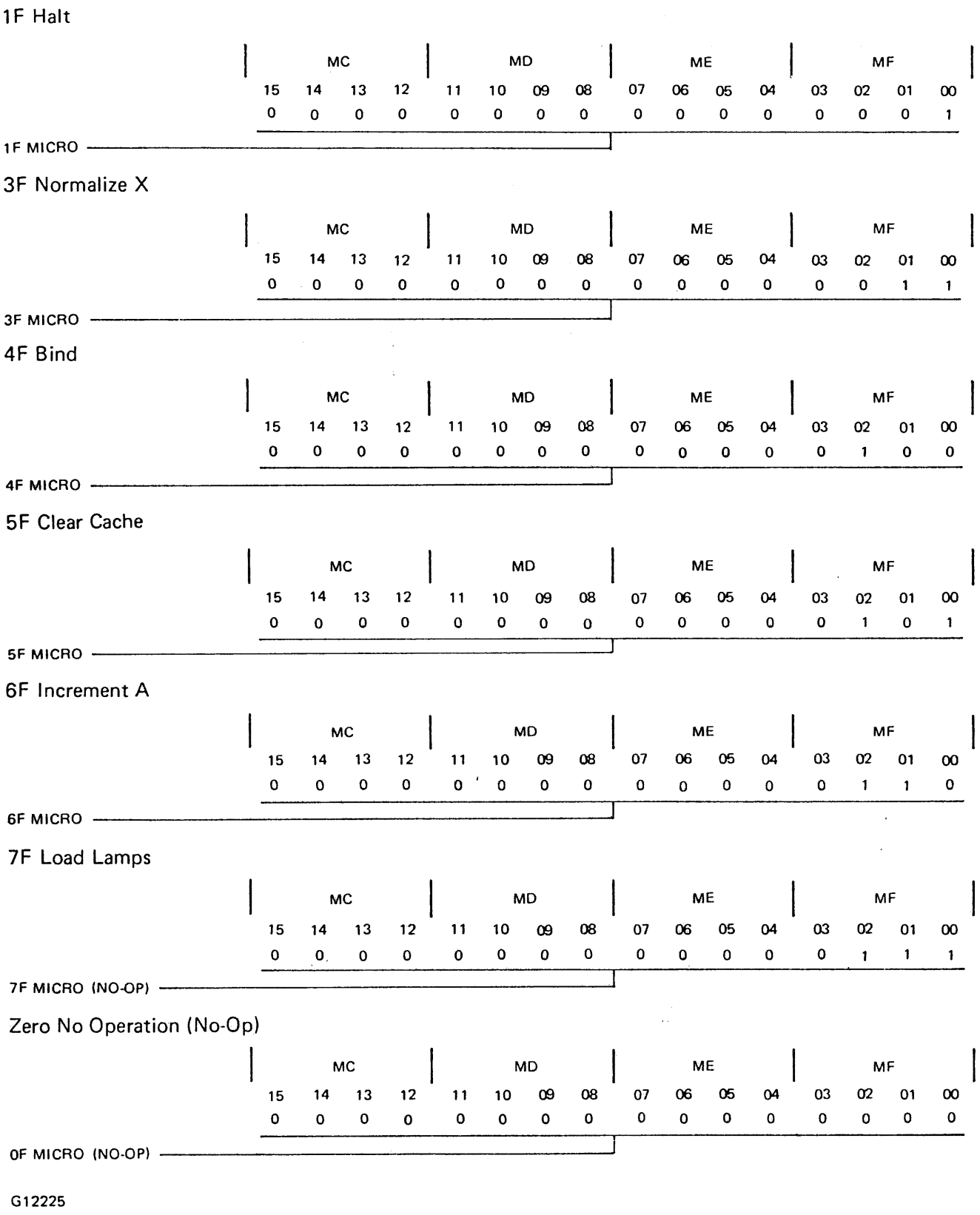


Figure 1-28. Microinstructions 1F through 7F and No-Op

1F Halt

This micro causes the execution of micros to terminate. If it is executed with the processor running in NORMAL mode, it takes effect after the next micro is fetched to the M register, with the A register pointing to the second-in-line micro. In MTR mode, the halt takes effect with the HALT micro in M; the next micro is not fetched.

3F Normalize X

For this micro, the value in CPL references a bit in the X register. CPL = 1 references the least significant (rightmost) bit and each higher value of CPL references the next more significant bit of X; CPL = 24 references the most significant (leftmost) bit of X. (CPL = 0 is undefined.)

The contents of the X register are shifted left, with right zero fill, while the FL register is counted down. Shifting stops when FL = 0 or when the CPL-referenced X bit = 1.

4F Bind

The 24-bit sum of registers L and T is moved to the A register. The four least significant bits of this 24-bit sum are truncated because A is a 20-bit register.

5F Clear Cache

All of Cache is cleared by resetting all 1024 Validity bits to zero. The Cache keys for indexes 0 through 511 are reset (0) and the key parity bits are set (1). @1FFF@ is stored in each word 0 of indexes 0 through 511. All other locations are unchanged.

6F Increment A

This micro may be used in the MTR mode to count up the A register by 1 word (16 bits).

7F Load Lamps

The console lamps normally monitor MEX bits 23-17 plus the 17 bits of Cache output. However, the first execution of this micro after entry into the RUN state turns off all 24 lamps until execution of another Load Lamps micro or a halt. Each subsequent Load Lamps micro causes the contents of the source designated in the preceding Move micro (1C or 2C) to be continuously displayed. If the preceding micro is not 1C or 2C, results are undefined.

This micro is defined for NORMAL and C (Cache only). Automatic Cache loading from S-Memory cannot take place between execution of the Move and Load Lamps micros.

No Operation

This micro causes a skip to the next-in-line micro.

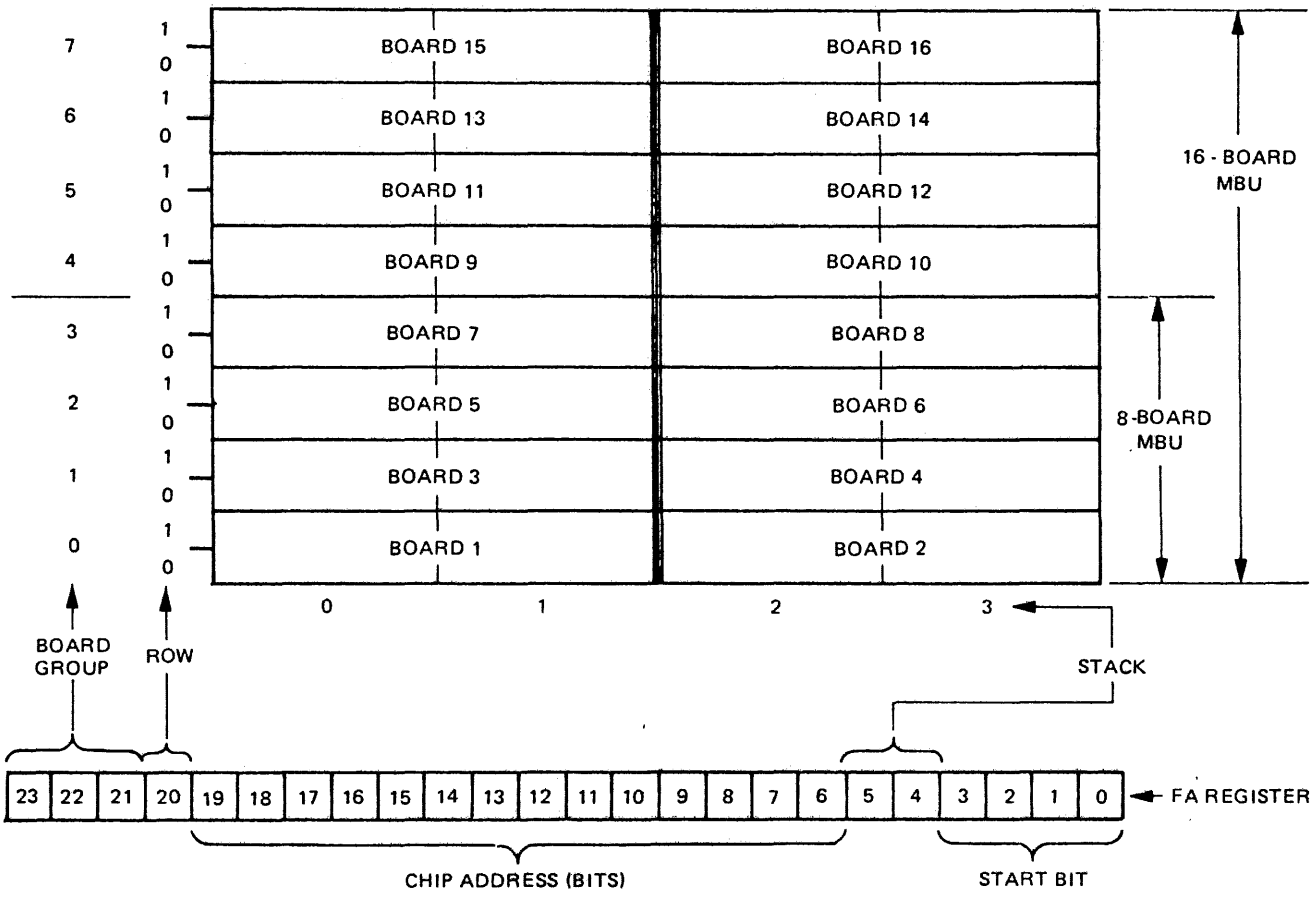
S-MEMORY

S-Memory, the system's main memory, is physically located in a memory base unit (MBU) that consists of two logic cards (R6 and S6) and from either two to eight storage boards (128 KB to 1 MB) or two to 16 storage boards (128 KB to 2 MB). The MBU, which provides memory control for storage access, interfaces directly with the memory-using devices (for example, processors or multiline controls).

S-Memory Storage Architecture

S-Memory storage consists of four blocks, or stacks, that can be randomly accessed. Each stack is 22 bits wide (the 22 bits represent one word). Each 22-bit word is organized as 16 data bits (two bytes) and six bits that contain the error correction code (ECC) used for error detection and correction on the 22-bit memory word.

S-Memory organization is shown in Figure 1-29.



1 BOARD = 128K BYTES
1ROW = 128K BYTES
1 STACK = 32K BYTES X NUMBER OF BOARDS IN STACK
1 WORD = 22 BITS (16 DATA + 6 ECC BITS)
1 BYTE = 8 BITS

G12226

Figure 1-29. S-Memory Organization

Error Correction Code

The MBU utilizes a 6-bit error correction code (ECC) to enable detection and correction of all single-bit errors within the 22-bit memory word. The following illustrates the 22-bit storage word.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	16 17 18 19 20 21
DATA	ECC

The six ECC bits permit the successful detection of the majority of multiple-bit errors. All errors involving two bits are detected; however, certain errors involving three or more bits can be erroneously detected as single-bit errors and incorrectly "corrected". Error detection and correction is accomplished via a 6-bit syndrome. Table 1-11 summarizes the ECC generation, Table 1-12 summarizes the syndrome generation, and Table 1-13 shows the syndrome patterns for single-bit errors.

Table 1-11. Check Bit Generation

16 = (0 + 1 + 2 + 3 + 4 + 5 + 6 + 7)/

17 = (0 + 1 + 2 + 8 + 9 + 10 + 11 + 12)/

18 = (3 + 4 + 8 + 9 + 10 + 13 + 14 + 15)/

19 = (0 + 3 + 5 + 6 + 8 + 11 + 13 + 14)/

20 = (1 + 5 + 7 + 9 + 11 + 12 + 13 + 15)/

21 = (2 + 4 + 6 + 7 + 10 + 12 + 14 + 15)/

\+ = Logical exclusive OR.

/ = Logical NOT.

Table 1-12. Syndrome Generation

S1 = (0 + 1 + 2 + 3 + 4 + 5 + 6 + 7 + 16)/

S2 = (0 + 1 + 2 + 8 + 9 + 10 + 11 + 12 + 17)/

S3 = (3 + 4 + 8 + 9 + 10 + 13 + 14 + 15 + 18)/

S4 = (0 + 3 + 5 + 6 + 8 + 11 + 13 + 14 + 19)/

S5 = (1 + 5 + 7 + 9 + 11 + 12 + 13 + 15 + 20)/

S6 = (2 + 4 + 6 + 7 + 10 + 12 + 14 + 15 + 21)/

\+ = Logical exclusive OR.

/ = Logical not.

Table 1-13. Syndrome Patterns for Single-Bit Errors

			Data Bits															ECC Bits					
ELOG	MBU	*	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
5	S1	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
4	S2	0	1	1	1	0	0	0	0	0	1	1	1	1	1	0	0	0	0	1	0	0	0	0
3	S3	0	0	0	0	1	1	0	0	0	1	1	1	0	0	1	1	1	0	0	1	0	0	0
2	S4	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	0	0	0	1	0	0
1	S5	0	0	1	0	0	0	1	0	1	0	1	0	1	1	1	0	1	0	0	0	0	1	0
0	S6	0	0	0	1	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	0	0	0	1

* No errors

NOTE

Any syndrome pattern not in the above table represents a multiple-bit error.

MBU Operations

The MBU is capable of performing 13 separate operations. Twelve of these are commanded by the host system and one (Refresh) is commanded internally by the MBU. The operations are included in Table 1-14.

Table 1-14. MBU Operations

Micro Bits 4/ 3/ 2/ 1/ 0/	REQ/	Mode Bits 3/ 2/ 1/ 0/	Operation
Don't care	1	X X X X	No request
Don't care	0	1 1 1 1	Reserved
Don't care	0	1 1 1 0	Refresh (Internal only)
Don't care	0	1 1 0 1	Micro fetch (four stacks)
Don't care	0	1 1 0 0	Reserved
Field length	0	1 0 1 1	Defined field read (7C)
Field length	0	1 0 1 0	Defined field write (7C)
Field length	0	1 0 0 1	Defined field swap (2D)
Don't care	0	1 0 0 0	Reserved
Don't care	0	0 1 1 1	22-bit read (11D)
Don't care	0	0 1 1 0	22-bit write (11D)
Don't care	0	0 1 0 1	Echo write data (11D)
Don't care	0	0 1 0 0	Echo address (11D)
Don't care	0	0 0 1 1	Read and clear ELOG (11D)
Don't care	0	0 0 1 0	No-OP (HA diagnostic)
Don't care	0	0 0 0 1	Dispatch read (1E)
Don't care	0	0 0 0 0	Dispatch write (1E)

X = Don't care.

NOTE

The values given for the various signals are those on the interface to the MBU.

Defined Field Memory Accessing

The basic memory operations are called defined field accesses and are implemented by micros 7C and 2D. The concept is illustrated in Figure 1-30.

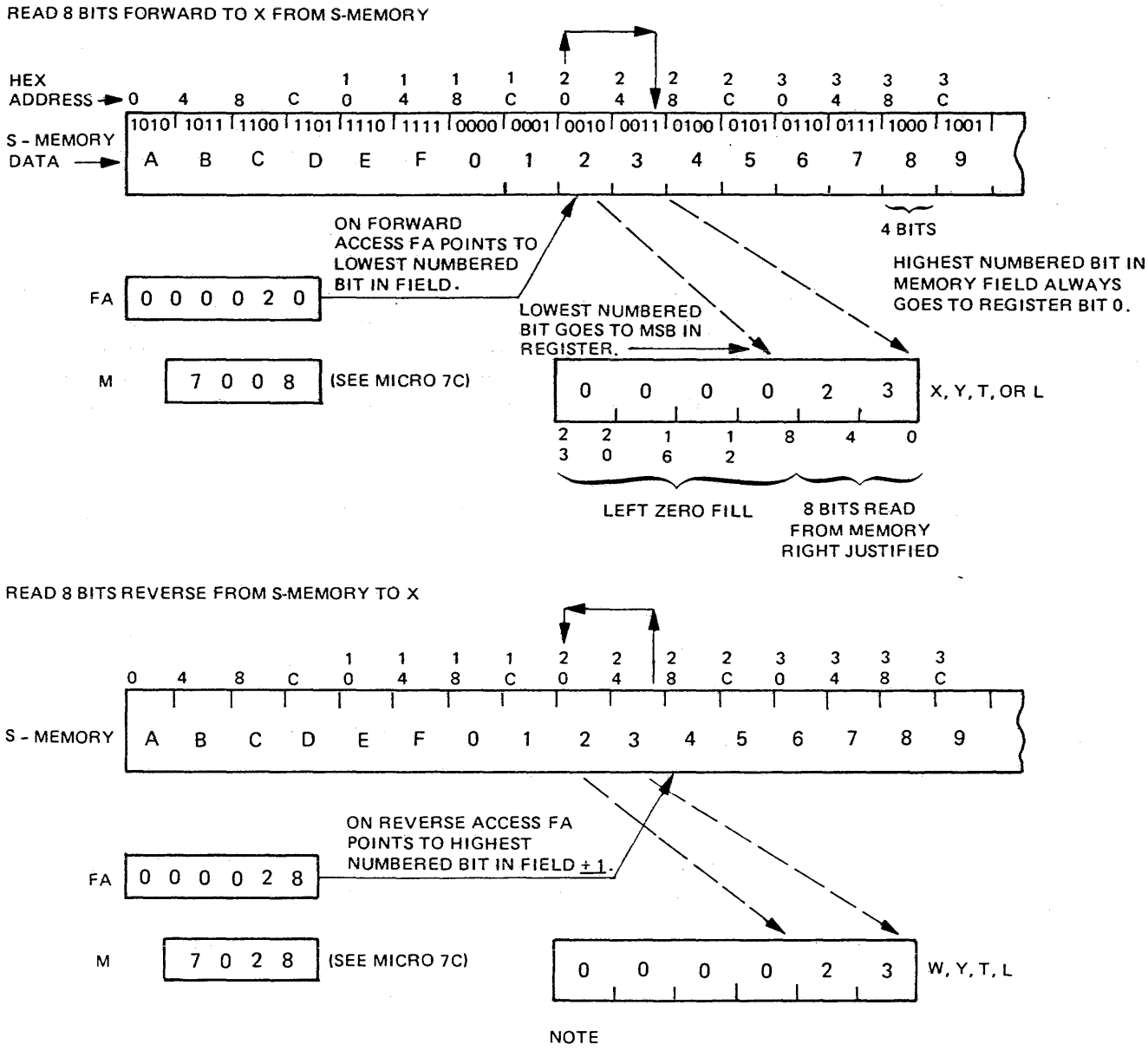


Figure 1-30. Defined Field Memory Accessing

Defined Field Read

This is the basic read operation from memory. Any field length of data from 0 to 24 bits can be accessed at any bit (from 0 to 15) within any stack (from 0 to 3) in either the forward or reverse

direction. All the above information must be supplied by the user. All single-bit errors are corrected and multiple-bit errors are reported. Since data can span from one to three stacks in memory, a read that accesses data from two stacks takes one clock longer than a read that accesses data from one stack, and a read that accesses data from three stacks takes one clock longer than a read that accesses data from two stacks. The one exception is the reverse read with the bit pointer pointing at bit 0. This is always at least a two-stack access even though the data may lie entirely within one stack.

On a forward access, FA points to the lowest-numbered bit in the accessed field in memory. On a reverse access, FA points to the highest-numbered bit plus one. In both cases, the highest-numbered bit in the memory field corresponds to bit 0 of the processor register specified and the lowest-numbered memory bit corresponds to the most significant bit in the register.

Defined Field Write

The Defined Field Write is the basic write operation to memory. Any field length of data (from 0 to 24 bits) can be written starting at any bit (from 0 to 15) within any stack (0 to 3) in either the forward or reverse direction. All the above information must be supplied by the user.

This is actually a Read-Modify-Write operation to the storage boards. Data is first read out (any single-bit error is corrected), then write data is substituted into the appropriate bit positions to create a new 16-bit data word. New ECC is then generated for the new 16-bit data word and the entire 22 bits is written into the storage boards.

If a multiple-bit error is detected when the data is read from the stack, it is reported and the new ECC generated for the new 16-bit data word is inverted before it is written into the storage board. This leaves an indication in memory that the word may still contain an error. (Inverting the correct ECC for a word causes an uncorrectable error to be detected whenever the word is reread.) If, however, the entire 16 data bits of the stack are updated with data from the user, any errors detected in the read data are ignored and correct ECC is written into memory for the 16 data bits.

If a transfer width of 25 or 26 is indicated by the user, it is truncated to 24 for data manipulation purposes, and special conditions are set for the operation. A transfer width of 25 causes correct ECC bits to be written into memory regardless of any error conditions encountered in the read-out phase for all stacks accessed.

A transfer width of 26 causes the correct ECC bits to be inverted before they are written into memory regardless of any error conditions encountered in the read-out phase of all stacks accessed. Transfer width requests of 27, 30, or 31 are interpreted by the MBU as a transfer width of 26. A transfer width request of 28 is interpreted as a transfer width of 24, and a transfer width request of 29 is interpreted as a transfer width of 25.

If the NOWRITE bit from the user also comes true for the write operation, the cycle continues with its normal timing except that all write enables to the storage boards are disabled and no data is written into memory.

Defined Field Swap

The Defined Field Swap command causes the MBU to perform a Defined Field Read immediately followed by a Defined Field Write in one uninterruptable operation. The paragraphs for the Defined Field Read and Defined Field Write apply to the Defined Field Swap.

Micro Fetch Four Stacks (Stream Mode)

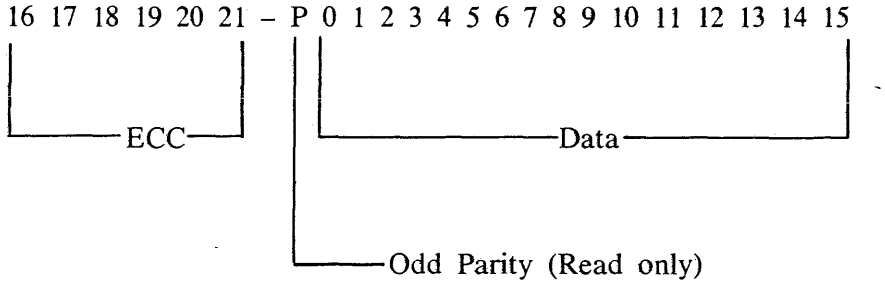
The MBU responds to a Stream Mode request by sending back four 16-bit data fields plus an odd parity bit for each field. This is done in successive clocks. The MBU forces the accesses to start in stack 0 at bit 0.

The accesses of the four micros are always in the forward direction. The bit pointer, stack pointer, and field direction sign do not affect this operation. Single-bit errors are corrected and multiple-bit errors are reported.

To ensure proper error handling, the value of micro bits 4 through 0, which the MBU interprets as the transfer width, must not be in the range of 25 through 31. Transfer width values from 0 to 24 have no effect on this operation.

22-Bit Read (Diagnostic Read)

The Diagnostic Read causes the entire 22 bits (16 data bits and 6 ECC bits) of a memory word to be read. At the MBU interface, the 16 data bits are right justified and the six ECC bits are left justified. An odd parity bit is supplied on the 16 data bits. Single-bit errors are not corrected but single and multiple errors are reported. The parity bit is always odd for the 16 data bits regardless of whether any errors are detected in the data. The following illustrates the Diagnostic Read/Write data format.



The MBU forces the access to start at bit 0 of the appropriate stack and forces the access to be one stack wide. Therefore, the bit pointer and field direction sign do not affect this operation.

In order to ensure proper error handling, the transfer width must not be in the range of 25 through 31. Values of transfer width from 0 to 24 do not affect this operation.

22-Bit Write (Diagnostic Write)

The Diagnostic Write updates the entire 22 bits (16 data and six ECC) of a memory word with data supplied by the user. At the MBU interface, the 16 data bits must be right justified and the 6 ECC bits left justified. The parity bit does not apply. There is no error correcting or reporting for this operation.

The MBU forces the write to occur on a stack boundary (starting at bit 0 of the appropriate stack). The transfer width, bit pointer and, field direction sign do not affect this operation.

Dispatch Read

The Dispatch Read is a Defined Field Read with the following parameters forced by the MBU:

Address = 0
Stack pointer = 0
Bit pointer = 0
Transfer width = 24

Data is accessed in the forward direction. All single-bit errors are corrected and all multiple-bit errors are reported.

Even though the MBU forces the transfer width to 24, a request by the user with the transfer width in the range of 25 through 31 interferes with proper error handling. Transfer widths in the range of 0 through 24 do not affect this operation.

Dispatch Write

The Dispatch Write is a Defined Field Write with the following parameters forced by the MBU:

Address = 0
Stack pointer = 0
Transfer width = 24

All data is accessed in the forward direction.

When the transfer width requested is in the range of 0 to 24 this operation is unaffected. (It is always a 24-bit operation.) However, if the transfer width is in the range of 25 to 31, the effect is the same as the Defined Field Write. NOWRITE also has the same effect as the Defined Field Write.

Diagnostic Read and Clear Error Log

This is not an access to the storage boards, but an access to the Error Log Register (ELOG) within the MBU. The MBU returns the contents of the ELOG at the end of the cycle.

The operation is completely defined by the simple request for Read and Clear ELOG; that is; there is no bit pointer, stack pointer, transfer width or field direction sign required.

This command instructs the MBU to return the address sent to it by the requesting device. The address is modified either in the forward direction or the reverse direction depending upon the field direction sign (FDS): FDS = 0 yields ADDR+16, FDS = 1 yields ADDR-16.

ADDR is changed by 16 because modification of the address in the MBU begins with the stack pointer (address bits 4 and 5).

The path for Echo Address is through the ELOG; therefore, not all bits of the address are returned to the requesting device.

Diagnostic Echo Write Data

This command instructs the MBU to return the write data sent to it by the user.

Refresh

This MBU-generated command causes the hardware to recharge the memory elements on the storage boards. The MBU generates a refresh request once every 16 microseconds. If the processor or another user requests a memory cycle at the same time, the refresh cycle is postponed, and the requestor is granted the cycle. After a refresh request is generated internally by the MBU, the actual refresh operation can be postponed up to a maximum of 12 microseconds. If a refresh request has not been honored within 12 microseconds, all requests from the users are locked out and a refresh cycle is performed. Setting the maximum postponement time at 12 microseconds ensures that the refresh request is honored before another refresh request is made. (Refresh requests are generated every 16 microseconds regardless of any previous refresh cycles being postponed.)

Refresh can be turned off and on either programmatically or by a jumper on the backplane. The Diagnostic Echo Address request with a "1" in the least significant bit of the FA register disables Refresh.

Either the Diagnostic Echo Address request, with a "0" in the least significant bit of the register, or clearing the system enables refresh. Grounding the RFSHEN.0 pin on the S6 card backplane disables refresh while leaving it disconnected enables refresh.

No-Op

The No-Op is executed whenever the processor issues a Host Adapter-3 diagnostic command. The only action taken by the MBU during a No-Op is to supply the processor with the required interface control signals.

SECTION 2
INSTALLATION

This section describes the initial set-up and test procedures for B 1900 central systems and provides instructions for field expansion. The information contained herein supplements the B 1900 Installation Planning Guide (IPG).

To install a system, proceed as directed in the following paragraphs.

PHYSICAL PREPARATIONS FOR OPERATION

Physical preparation involves unpacking, assembling, and making the necessary electrical connections. These procedures are described in the IPG.

SUBASSEMBLY CHECKLIST

Check to ensure the presence of all necessary subassemblies. Check the following items in the order listed.

1. Inspect the processor and I/O base frontplane to ensure that the logic and memory cards are installed in their proper locations.
2. Inspect the memory base frontplane to ensure that the proper number of logic cards and S-Memory storage cards are installed.
3. Check to ensure that all frontplane cables are installed in their proper locations.

NOTE
Cables connected to the I/O controls are not shown.

4. Check to ensure that all mechanical connections within the central system cabinet are secure, and that there are no loose wires, components, or other obvious comprises to the physical integrity of the unit.

CENTRAL SYSTEM OPERATIONAL CHECKOUT

When the physical installation of the central system has been completed and connection has been made to the electrical power source, an operational checkout must be made. This testing procedure is to ensure that the unit performs as expected. The operational checkout procedure is in three phases: (1) static tests with power off, (2) static tests with power on, and (3) dynamic tests.

CAUTION
The central system operational checkout concerns the processor and S-Memory only. Therefore, do not connect or attempt to utilize any peripheral devices until the checkout is completed.

Static Tests (Power Off)

These tests are to ensure that the central system does not fail due to shorts or open circuits.

1. Make sure that both the system and source circuit breakers are in the OFF position.
2. Using a Triplett 630 VOM or equivalent, make sure that no short exists between the +5.0V , -2.0V, +12V, or -12V outputs and ground or between each of these outputs and all others.
3. Check for continuity between the logic power supply outputs and the backplane pins listed in Table 2-1.

Table 2-1. Logic Power Continuity Test

Output Point Backplane Pin(s)	Card(s)	
+5.0V (Vcc)	0AX, 1AX	All logic cards.
-2.0V (Vee)	0ZX, 1ZX	All logic cards.
GND	1DX,1JX,1QX,1WK	All logic cards.
+12.0V	1LY	I/O Control cards only.
-12.0V	0ZX, 1AY	All logic cards.

Static Tests (Power On)

Static tests with system power applied are performed to ensure that the basic functions for proper system operation are present, and that the functions that are adjustable are set to the proper values. These tests involve three major areas: (1) power supplies, (2) clock circuits, and (3) the console. The tests are implemented by manipulation of the Op and D/M panel controls. (The panels are described in Section 1.)

Powering Up

Apply power to the system as follows:

Switch the source and system circuit breakers to ON. Switch POWER (in Op panel) to ON. When power is applied, the ON lamp lights and the blowers start.

Power Supply Tests

When the system has been successfully powered up, check the output voltages of the logic power supply. The proper voltages are shown in Table 2-2.

NOTE
Supply voltages are measured on the backplane.

Table 2-2. System Operating Voltages (Nominal Values)

Voltage	Tolerance
+5.00	±0.01
- 2.05	±0.01
+12.00	±0.10
-12.00	±0.10

Clock Circuit Tests

Perform the Clock Circuit Adjustment Procedure outlined in the Field Test and Reference (FT&R) documentation.

Console Tests

Before attempting operation under program control, ensure that all operational functions and capabilities of the central system logic are verified to be accessible and operable.

The 15 tests in the subsections that follow are performed in sequence. Before beginning, clear the system and set CP for a width of 24 as follows:

1. Press CLEAR.
2. Set REGISTER SELECT = 2, REGISTER GROUP = CP.
3. Set console switches = @000018@.
4. Press LOAD.

Following any procedure that specifies loading CP with a value other than 24 (@000018@), restore CP to 24.

Test 1 (Register Loading), Test 2 (Register Changing)

These tests verify the ability to (1) load and (2) change the contents of the registers X, Y, T, L, A, M, BR, LR, FA, FB, FL, and CP. (All these registers are selectable via the REGISTER GROUP switch when REGISTER SELECT is at position 2.)

Each register is tested in turn for loading (test 1). The register sequence is repeated for changing (test 2). Each result appears in the console lamps. Correct results are listed at the end of the instruction sequence that follows.

1. Set REGISTER SELECT = 2.
2A. (Load) Set console switches = @FFFFFF@ (all up).
2B. (Change) Set console switches = @000000@ (all down).
3. Set REGISTER GROUP = n. (For actual register, see list below. The simplest way to proceed is to start at register X and rotate the switch one position clockwise on each pass. Skip TAS, and stop at CP.)
4. Press LOAD.

Go through the sequence twice. First, use step 2A and compare the results as you go with those in the test 1 column, below. Then, use step 2B and compare the results with the test 2 column. (In the listing below, the register are sequenced left to right and top to bottom as they are encountered when the RG switch is rotated clockwise. TAS, which is between FL and CP on the dial, is omitted.)

Registers (n)	Test 1	Test 2
X Y T L M	@FFFFFF@	@000000@
A	@FFFFF0@	@000000@
BR LR FA FB	@FFFFFF@	@000000@
FL	@00FFFF@	@000000@
CP	@0000FF@	@000000@
M	@01FFFF@	@000000@

Test 3 (Addressing T and L)

This test verifies the ability to address the T and L registers as entities and in 4-bit increments.

Perform the following sequence:

1. Set REGISTER SELECT = 2.
2. Set REGISTER GROUP = T.
3. Set console switches = @ABCDEF@.
4. Press LOAD.
5. Set REGISTER GROUP = L.
6. Press LOAD.
7. Set REGISTER SELECT = 0.
8. Set REGISTER GROUP = TA, TB, TC, TD, TE, and TF and then LA, LB, LC, LD, LE, and LF in sequence. At each position, compare the console lamp reading with the following:

   TA, LA = @00000A@
   TB, LB = @00000B@
   TC, LC = @00000C@
   TD, LD = @00000D@
   TE, LE = @00000E@
   TF, LF = @00000F@

Test 4 (Addressing CA, CB, CC, CD)

This test verifies the ability to address registers CA, CB, CC, and CD as 4-bit destinations and as 4-bit sources.

Perform the following sequence:

1. Set REGISTER SELECT = 0.
2. Set console switches = @00000F@ (first pass).
3. Set REGISTER GROUP = CA.
4. Press LOAD. Verify that the value loaded from the console switches appears in the console lamps.

NOTE

When @000000@ is loaded and CC is read (second pass, step 6), the indicators show @000004@ because the real-time clock interrupt sets bit 2 of the CC register.

5. Set REGISTER GROUP = CB, repeat step 4, then go to step 6.
6. Set REGISTER GROUP = CC, repeat step 4, then go to step 7. On the second pass (following step 9), observe exception noted in step 4.
7. Set REGISTER GROUP = CD, repeat step 4, then go to step 8 if this is the first pass, or terminate the test if this is the second pass.
8. Set console switches = @000000@ (second pass).
9. Repeat steps 3-7 (second pass). Summary of values loaded and read:

Register	First Pass	Second Pass
CA	@00000F@	@000000@
CB	@00000F@	@000000@
CC	@00000F@	@000004@
CD	@00000F@	@000000@

Test 5 (Addressing FB)

Verify the ability to address the FB register in 4-bit increments.

1. Set REGISTER SELECT = 2.
2. Set REGISTER GROUP = FB.
3. Set console switches = @ABCDEF@.
4. Press LOAD.
5. Set REGISTER SELECT = 1.
6. In sequence, set REGISTER GROUP to the registers listed below and verify that the console lamps give the corresponding readings:

   FU = @00000A@
   FT = @00000B@
   FLC = @00000C@
   FLD = @00000D@
   FLE = @00000E@
   FLF = @00000F@

Tests 6, 7, 8, 9 (XY Conditions)

These four tests utilize the capabilities of the XYCN (XY conditions) register to verify that certain processor arithmetic logic unit (ALU) features are functioning properly.

Test 6 tests the X and Y registers for equality (X = Y) and tests 7 and 8 test for inequality (test 7: X > Y, test 8: X < Y). test 9 detects whether the most significant bit of X is true.

The following sequence is used for each test:

1. Set REGISTER SELECT = 2.
2. Set REGISTER GROUP = X.
3. Set console switches = hex value shown for X in the appropriate column in Table 2-3.
4. Press LOAD.
5. Set REGISTER GROUP = Y.
6. Set console switches = hex value shown for Y in the appropriate column in Table 2-3.
7. Press LOAD.
8. Set REGISTER SELECT = 1.
9. Set REGISTER GROUP = XYCN.
10. Compare result in console lamps with hex value shown for XYCN in appropriate column in Table 2-3.

Table 2-3. Loading and Results for Tests 6, 7, 8, and 9

	Test 6	Test 7	Test 8	Test 9
Load X with:	@000000@	@000001@	@000000@	@800000@
Load Y with:	@000000@	@000000@	@000001@	@800000@
Read XYCN:	@000004@	@000001@	@000002@	@00000C@

Test 6

XYCN = @000004@ means that the least significant four bits of XYCN = 0100, that is, that bit 2 is set. This is true when the contents of X are equal to the contents of Y (X = Y).

Test 7

XYCN = @000001@ means that the least significant four bits of XYCN = 0001, that is, that bit 0 is set. This is true when the contents of X are greater than the contents of Y (X > Y).

Test 8

XYCN = @000002@ means that the least significant four bits of XYCN = 0010, that is, that bit 1 is set. This is true when the contents of X are less than the contents of Y (X < Y).

Test 9

XYCN = @00000C@ means that the least significant four bits of XYCN = 1100, that is, that bits 3 and 2 are set. Bit 3 being set means that the most significant bit of the 24-bit X register is set, which is true because X was loaded with @800000@ (most significant four bits = 1000). Bit 2 being set means that X = Y (see test 6, above). X = Y because Y was also loaded with @800000@.

Test 10, 11 (XY States)

These two tests are similar to tests 6 through 9 except that the XYST (XY states) register is the object rather than the XYCN register. Test 10 verifies that the contents of the X register are not equal to zero; test 11 does the same for the Y register.

The following sequence is used for each test:

1. Set REGISTER SELECT = 2.
2. Set REGISTER GROUP = X for test 10, Y for test 11.
3. Set console switches = hex value shown for the register under test in the appropriate column in Table 2-4.
4. Press LOAD.
5. Set REGISTER SELECT = 1.
6. Set REGISTER GROUP = XYST.
7. Compare result in console lamps with hex value shown for XYST in appropriate column in Table 2-4.

Results are interpreted in the following subsections.

Table 2-4. Loading and Results for Tests 10 and 11

	Test 10	Test 11
Load X with:	@000001@	–
Load Y with:	–	@000001@
Read XYST:	@00000D@	@000006@

Test 10

XYST = @00000D@ means that the least significant four bits of XYCN = 1101, that is, bits 3, 2, and 0 are set. Bit 3 is set when the least significant bit of X is set (X was loaded with @000001@), bit 2 is set because a real-time clock interrupt has occurred, and bit 0 (the object of this test) is set because the contents of X are not equal to zero (X neq 0).

Test 11

XYST = @000006@ means that the least significant four bits of XYCN = 0110, that is, bits 2 and 1 are set. Bit 2 is set because a real-time clock interrupt has occurred. Bit 1 is set because the contents of the Y register are not equal to zero (Y neq 0).

Test 12 (A Register Incrementing)

Verify that the A register is incremented when the INC push button is pressed.

1. Press CLEAR. (CLEAR resets A to zero.)
2. Set REGISTER SELECT = MEMORY.
3. Set REGISTER GROUP = CMR (Cache memory read).
4. Press INC 7 times.
5. Set REGISTER SELECT = 2.
6. Set REGISTER GROUP = A.

Result in console lamps = @000070@. This means that the A register, which had been cleared to zero, was incremented seven times. Each increment of A has a binary weight of 16.

Test 13 (Writing into Cache)

Verify that a microinstruction (16 bits) can be written into Cache.

1. Press CLEAR.
2. Set REGISTER SELECT = MEMORY.
3. Set REGISTER GROUP = CW.
4. Set console switches = @00FFFF@.
5. Press LOAD.
6. Set REGISTER GROUP = CMR.
7. Press LOAD.

Result in console lamps = @FCFFFF@ (the microinstruction, including odd parity).

Test 14 (Incrementing FA)

Verify that the FA register is incremented when the INC push button is pressed.

1. Set REGISTER SELECT = 2.
2. Set REGISTER GROUP = FA.
3. Set console switches = @000000@.
4. Press LOAD. (Steps 1-4 clear FA to address zero.)
5. Set REGISTER SELECT = MEMORY.
6. Set REGISTER GROUP = SR16.
7. Press INC once.
8. Set REGISTER SELECT = 2.
9. Set REGISTER GROUP = FA.

Result in console lamps = @000010@, showing that FA was incremented by binary 16.

Test 15 (Writing Into and Reading From S-Memory)

Verify that 24 bits may be written into and read from an S-Memory address.

1. Set REGISTER SELECT = 2.
2. Set REGISTER GROUP = FA.
3. Set console switches = @000000@.
4. Press LOAD. (Steps 1-4 clear FA to address zero.)
5. Set REGISTER SELECT = MEMORY.
6. Set REGISTER GROUP = SW24.
7. Set console switches = @FFFFFF@.
8. Press LOAD. This writes the value in the switches into S-Memory address zero.
9. Set REGISTER GROUP = SR24.
10. Press LOAD.

Result in console lamps = @FFFFFF@. Steps 5-8 entered this value into the memory location specified by FA. Step 9 reads that location. If the ERROR indicator lights, it may be indicating that memory has not been flushed with correct ECC after power-up. This has no effect on the proper return of data in this test.

Dynamic Tests

Once it has been determined that the basic central system functions can be implemented under manual control, tests under program control can begin. The dynamic tests consist of software routines that exercise the processor and memory with microinstruction sequences. The tests are arranged to test various logic segments and obtain correct responses or error indications such as halt interrupts or specific bit patterns stored in various registers. These tests are also used for maintenance purposes.

The following are dynamic tests for proving the operational integrity of the central system:

1. MTR Mode processor Test.
2. Dynamic processor Test.
3. Host Adapter-3 Test. (For port-connect systems only.)
4. Dynamic S-Memory Test.
5. Dynamic Cache Test.

Each of these tests is a program on a cassette tape. Some are executed directly from the system's magnetic tape cassette unit; others are loaded into S-Memory from the cassette and then executed.

These tests are described in Section 4.

SECTION 3
DOCUMENTATION AND COMPONENTS

GENERAL

To assist the field engineer in performing maintenance work, a number of publications and diagnostic programs exist. These include documentation that is part of the system and other data available for reference purposes.

EQUIPMENT DOCUMENTATION

Each system when assembled is furnished with a basic issue of Field Test and Reference (FT&R) documentation. Included are schematics and logic diagrams for all portions of the system, assembly drawings, a backplane circuit list, diagnostic program listings, card test data, and the Hardware Rules book. Since these documents are used extensively during maintenance operations, a guide to the use of each is provided.

Logic Schematics

The logic schematics are used to present a graphic representation of the circuits that make up the system. In order to create documents of a practical size, it was necessary to sectionalize the schematics. The sections thus produced follow the physical rather than electrical divisions of the system with each individual schematic representing one of the logic cards. Where necessary to avoid crowding, several sheets are used to show the circuits on a single logic card.

Schematic Rules

To present a uniform appearance and avoid confusion, all schematics conform to a predetermined set of rules. Briefly, the rules for schematic layout are as follows:

1. Schematics depict the logic contained on a single card, and can consist of one or more pages.
2. Logic flow is from left to right and from top to bottom.
3. Signal connections can be either unidirectional or bidirectional as circuit requirements dictate but must be identified as such.
4. Unidirectional signals entering a card by way of frontplane connectors enter the schematic at the top of the page.
5. Unidirectional signals leaving a card by way of frontplane connectors leave the schematic at the bottom of the page.
6. Bidirectional signals using frontplane connectors may appear either at the top or the bottom of the page.
7. Backplane and interpage input signals enter the page from the left.
8. Backplane and interpage output signals exit the page to the right.
9. Signal destination (page and location) is denoted in a bubble.
10. Signal source (page and location) is denoted in a bubble.

Signal Names (Mnemonics)

To aid understanding of circuit functions and simplify tracing of signals through the various logic sections, the output of each active device has been assigned a mnemonic name. The mnemonic comprises an abbreviated description of the purpose and location of the signal it is associated with. Mnemonics consist of eight characters, and are composed of letters, digits, special symbols, and spaces.

Schematic Connection Symbols

To alleviate crowding and improve legibility, logic circuit schematic diagrams are drawn in a multiple-page format. To indicate the various types of connections that can exist between pages and to differentiate between physical and graphical circuit divisions, a system of symbolic notation was created. The schematic connection symbols consist of internal page symbols, backplane pin symbols, frontplane pin symbols, inter-page symbols, and special symbols. Each type is discussed in the following subsections.

Internal Page Symbols

Figure 3-1 illustrates internal page symbols. A numeral directly to the right of any of these symbols designates the numbers of places that the signal goes to on a particular page.

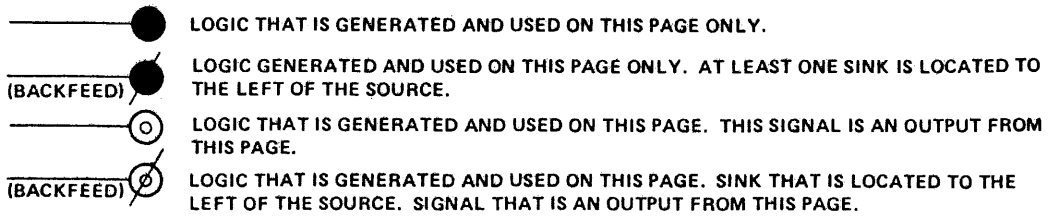


Figure 3-1. Schematic Internal Page Symbols

Backplane Pin Symbols

Backplane pins are identified by a three-character code that describes their location. The meaning of each character in the backplane pin code 0AX is described below.

0 = Component side of logic card (1 = solder side).
A = First pin of group. (Pin letters are A-Z, letter O is omitted.)
X = Upper half of card (Y = lower half).

Backplane pin codes are listed to the left of an input signal and to the right of an output signal. (Refer to Figure 3-2.)

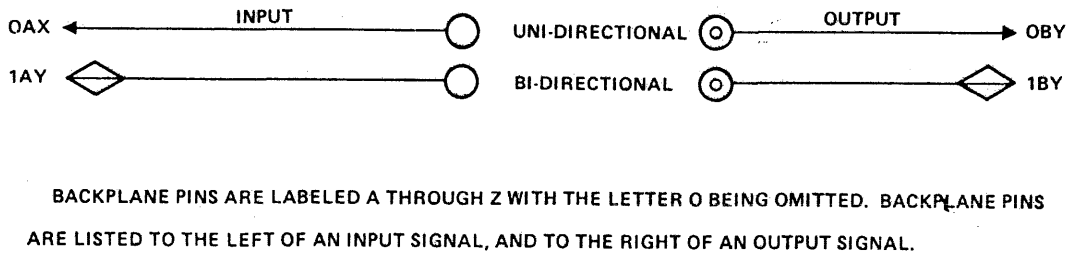


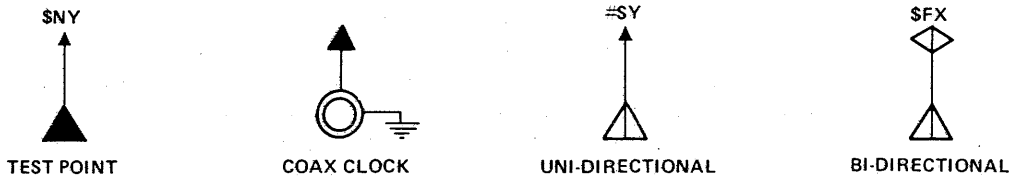
Figure 3-2. Schematic Backplane Pin Symbols

Frontplane Pin Symbols

Frontplane pins are identified in a manner similar to backplane pins, except that the pin groups are stacked vertically rather than being on opposite sides of the card. Since the frontplane can have the same number of pins as the backplane, two groups of pins per card half are required. The meaning of each character in the frontplane pin code $MY is described below.

$ = Upper group on card half (# = lower group).
M = 13th pin of group. (Pin letters are A-Z, O is omitted.)
Y = lower half of card (X = upper half).

Frontplane pin codes are listed above input signals and below output signals. (Refer to Figure 3-3.)

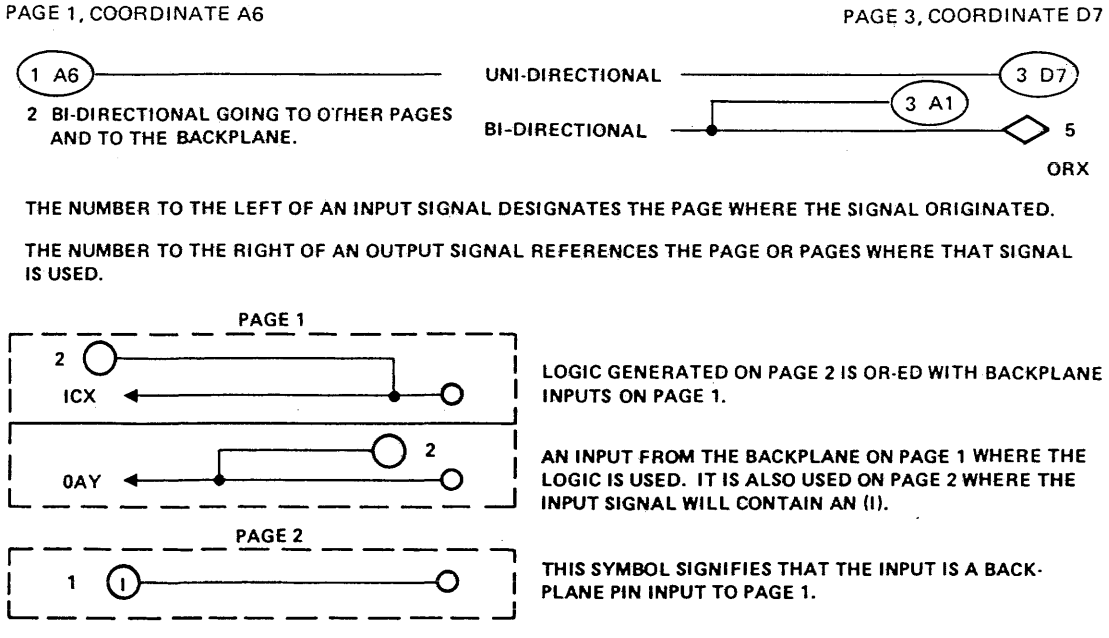


G11511

Figure 3-3. Schematic Frontplane Pin Symbols

Interpage Symbols

Where two or more schematic pages are needed to depict the logic circuitry on a single card, interpage connection symbols are used to indicate through-connections. Since no pin connections are involved, only the schematic page number(s) associated with that signal are shown. Refer to Figure 3-4.



G12229

Figure 3-4. Schematic Interpage Connection Symbols

Special Symbols

A triangle placed perpendicular to an output signal line indicates that a pull-down or load resistor is connected to this circuit. The accompanying code number shows the chip location and pin where this resistor is located.

An inverted triangle placed on an output line indicates a 150-ohm load resistor to ground on the solder side of the board. The code number shows the pin location to which the resistor is tied.

Hardware Rules Book

The integrated circuit elements employed in the B 1900 systems are identified by a special designator code. This code serves as an abbreviated specification for commonly used devices.

A compilation of the B 1900 series IC chips is contained in the Hardware Rules book (P/N 2209 6150) and is provided as part of the Field Test & Reference documentation.

The Hardware Rules book is arranged in alphabetical order by code designator, with the designators assigned in a manner roughly approximating abbreviated device operational functions. Device designator codes consist of four characters. The first two characters are always alphabetical. The third character can be either a letter or a number, depending upon whether one or more devices with the same general function (but with minor individual differences between them where there are two or more) exist. The last character of the code always appears as an N (meaning "number") in the references. Often such devices are packaged two or more to a chip, requiring that number be assigned to distinguish between them in actual applications. Such numbers appear only in the logic schematics and discussions which refer to them. Several examples of circuit element designator codes follow:

Designator Code	Chip Function
AFAN	Adder/Subtractor
CFAN	Comparator
FFAN	Flip-Flop
LFAN	Latch
RFAN	3-Bit Register

Backplane Circuit Lists

For each portion of the system that uses a wire-wrap backplane for interconnection of the plug-in logic cards, a backplane circuit list is provided. This is a complete listing, by signal name and pin numbers, of all point-to-point connections on the backplane. The listing complements the schematics by providing a means of tracing wiring circuits external to the logic cards.

Diagnostic Program Listing

The diagnostic programs provide system fault analysis by exercising suspected portions of the logic. Included with each such program is a listing that contains operating instructions and result analysis data. Refer to the Diagnostic Programs subsection in Section 4 for additional information.

SECTION 4
MAINTENANCE TECHNIQUES

INTRODUCTION

This section contains maintenance and troubleshooting procedures for B 1900 central systems. Included are discussions of general maintenance procedures and the aids that are available to assist in fault isolation.

Maintenance Concept

Requirements to maintain a central system in operating condition include proper preventive maintenance, installation and test of any recently released hardware changes, and, when required, the appropriate corrective action. Since there are few electromechanical devices in the central system itself, the need for continuing attention is minimal. However, certain precautions apply, and these are listed in the preventive maintenance discussion. When the central system does fail, the on-site repair effort is directed toward isolating the fault to a failing component, eliminating the fault, and restoring the system to operation. The direct substitution of subassemblies is avoided; this is reserved for cases where all other approaches fail.

A malfunction of the central system is characterized by its failure to perform one or more of the functions for which it was intended. B 1900 central system problems may be difficult to detect and isolate because control passes through several levels of software. The cause-and-effect relationships between hardware and software may not be apparent. On a typical maintenance call involving a logic fault, determining how the system is failing generally takes longer than finding and fixing the problem once that determination is made.

Test Equipment

In addition to common mechanical and electronic tools and work aids, the central system troubleshooting and maintenance procedures described herein require the following items:

Oscilloscope: Tektronix 465*
Multimeter: Triplett 630*
Digital voltmeter: Fluke 8000A*
Logic Card Extender: Burroughs M&E #2207 1237
Card Tester: Burroughs Field Card Tester

* or equivalent

Preventive Maintenance

Preventive maintenance for the B 1900 central system consists of inspection, operating voltage verification, and cleaning. Keeping the hardware in good order is important to maintain an operational and reliable system. The following items should be checked each time some portion of the system is serviced:

1. Ensure that all logic cards and connectors are firmly and accurately seated in their sockets.
2. Check power voltages and adjust if needed. (Refer to Special Troubleshooting Procedures, later in this section, for nominal voltage settings.
3. Inspect the interior and exterior portions of the central system cabinet and subassemblies for dust, foreign matter and corrosion. Take corrective action as needed.

4. Observe that all fans, including those in the logic power supply, rotate freely and are operational.
5. Check the air filter in each cabinet bay. Replace if necessary.
6. Check the ELOG for possible failing memory. Replace storage chips if needed.

Dynamic Troubleshooting

Dynamic troubleshooting involves the execution, in sequence, of six diagnostic programs (see Figure 4-1):

1. MTR Mode Processor Test
2. Dynamic Processor Test
3. Host Adapter Test
4. Dynamic S-Memory Test
5. Dynamic Cache Test

These programs exercise the central system logic in a progressively complex manner and can indicate faults down to the logic card level. When suspect cards have been identified, the Field Card Tester (FCT) is employed to further isolate the problems to failing circuits or IC chips. Faulty components are replaced as they are located, then the repaired cards are again checked with the FCT. Finally, the system is reassembled and diagnostic programs are again executed to verify the repairs.

Other Troubleshooting Methods

Sometimes a system fails under Master Control Program (MCP) control even though a diagnostic test does not indicate a hardware failure. The following is a list of conditions that could cause system failure but not diagnostic test failures:

1. Logic power supply voltages are out of tolerance or a logic power booster is inoperative.
2. System timings are out of tolerance.
3. An I/O control or peripheral device failure can cause the system to fail. The I/O diagnostic test provided with each control can determine if the I/O control is faulty and may also be useful for indicating a faulty peripheral device.
4. MCP failures can occur if required hardware or software changes have not been made. Therefore, the system should always be checked to assure that it is up to its required operational level (RIN/LIN).
5. Failures can be forced or isolated by using the diagnostic tests in conjunction with voltage margin testing.
6. Each logic power supply is designed to handle a specified maximum load (250 A). If the system is near its load limit, MCP failures may occur. Check to ensure that power supply loading is not in excess of the amounts permitted.
7. Installed S-Memory is insufficient for certain programs.
8. Required peripherals are not powered up and ready for operation.

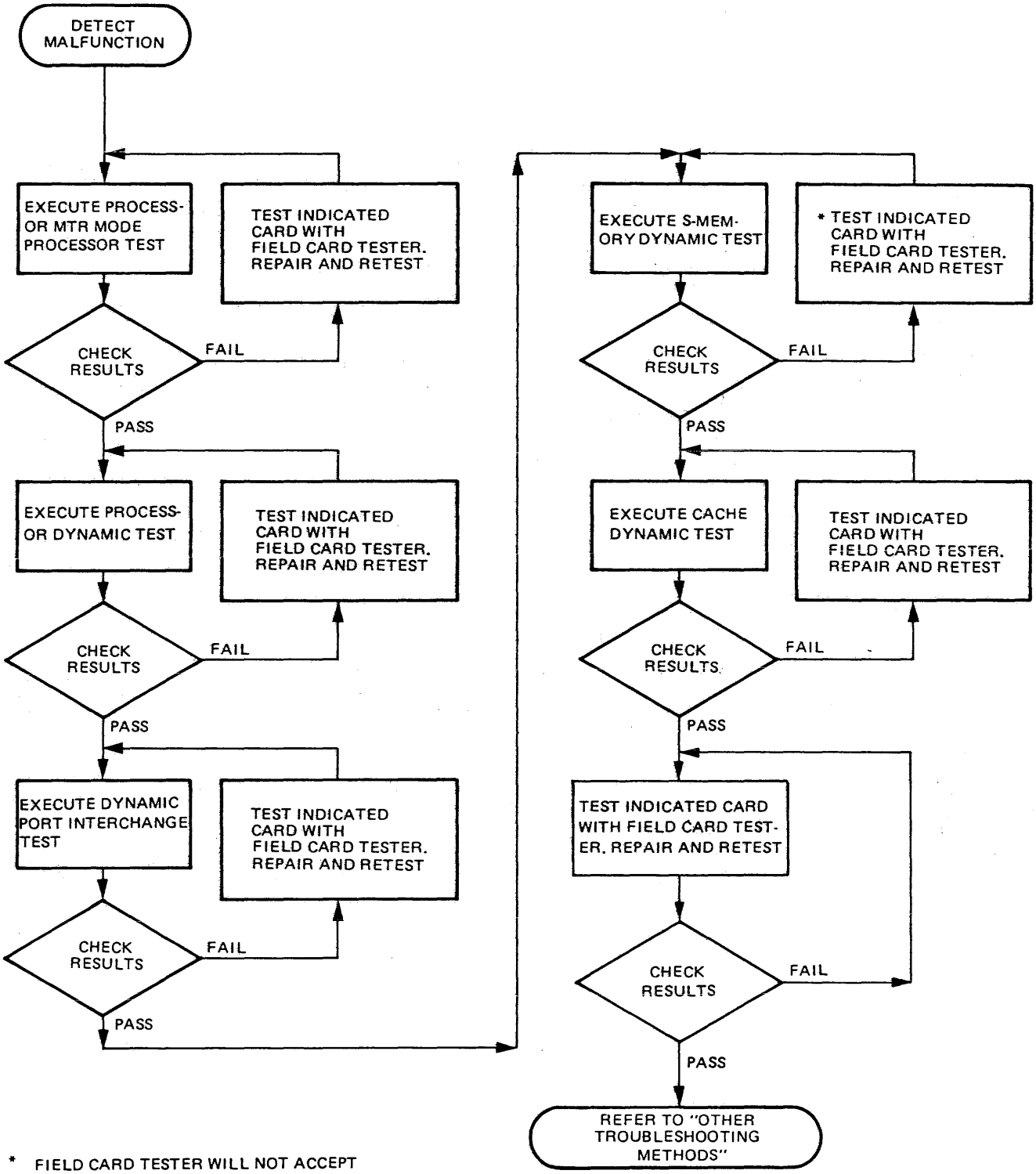


Figure 4-1. Flow Chart of the Dynamic Troubleshooting

DIAGNOSTIC PROGRAMS

The diagnostic programs listed earlier are available on cassette tapes. They exercise the processor with sequences of micros to produce specific, identifiable results. Each program includes a list of correct responses and a list of responses which indicate faults. (Response output is by way of the 24 console lamps or the ODT.) The program also includes instructions for running each program and analyzing the results. Procedures for using these programs are standardized and include halt codes, methods for entering program options, operator selectable modes of operation, register and scratch-pad usage, and loading procedures. The following paragraphs provide information on loading and use of the diagnostic programs.

NOTES

1. The following procedures deal with the processor, S-Memory, and Cache memory. The I/O subsystem, cassette tape subsystem, and power supplies are covered in other documents.

2. Halt codes written in hexadecimal are indicated by an at sign (@) preceding and following the code. Hexadecimal notation includes the numerals 0 through 9 and the letters A through F. Variants included with a hexadecimal code are indicated by lower case letters. Example: @xxAAAA@.

3. In references to the console switches, console lights, registers, and similar units made up of series of bit positions, software convention is used in order to conform with diagnostic test documentation. Therefore, sequences of bit positions reading left to right (MSB to LSB) are numbered in ascending (low to high) order. Thus, the leftmost console switch is bit 0 and the rightmost one is bit 23 instead of the opposite arrangement (23 to 0) which is the hardware convention.

Standard Loading

Loading procedures for all diagnostic tests except the MTR Mode Processor Test conform to the following general format:

1. Check that the cassette to be loaded into the reader has a part number and revision identification that matches that listed for the desired test program.
2. Set MODE switch (Op panel) to MTR.
3. Set MICRO SOURCE switch (D/M panel) to NORMAL.
4. Press CLEAR, then press START.
5. Select LR (the limit register) by setting RS to 2 and RG to LR. The processor halts and the console lights show @AAAAAA@ as the value in LR.
6. Set MODE switch to NORMAL.
7. Enter desired options. (Refer to program documentation.) No entry results in a default to standard execution.
8. Press START.

The test then runs under the chosen options until one of the following occurs:

1. A change of options is desired.
2. The end-of-test halt (LR = @EEEEEE@) occurs.
3. An error is detected.

Not all diagnostic programs use all the options provided. The program documentation specifies the valid options; any illegal entry is reported by LR = @DDDDDD@.

Halts

When a program halt occurs, the LR register contains a halt condition indicator value for interpretation by the operator. These halts (standard for all B 1900 central system programs) are described in the following two subsections, Standard Halts, which describes the halts that might occur during normal test execution, and Standard Executive Error Halts, which describes halts that might occur during test loading.

Standard Halts

@xxvnnn@ Error halt. xx denotes test section where error occurred; xx=00 indicates an executive or dynamic test MTR error. v, the halt variant, informs the operator that a specific diagnostic procedure is available to further diagnose the error; v=0 indicates that no procedure is available, v=1 indicates that the lockup loop is available, V=2 indicates that a diagnostic routine is available, and V=3 to V=F are undefined. The error number is specified by nnn.

@xxAAAA@ Begin test section halt. Occurs only when optionally selected and indicates that test number xx is loaded and will execute when START is pressed.

@AAAAAA@ Enter options halt. During loading, this display indicates the end of MTR mode operations. At that time, the operator may display the part and version numbers of the listing and the cassette and change certain options.

@BBBBBB@ Blow-up halt. An unexpected halt within a test section.

@DDDDDD@ Illegal operation halt. An illegal option was selected by the operator. To return to the enter options halt, START is pressed.

@EEEEEE@ End of test halt. All selected loops and passes are complete. To restart when the program is in S-Memory, START is pressed. To restart in Cache-Only mode, the tape is rewound and the procedure is begun again from operating instruction 1.

Standard Executive Error Halts

@000005@ MTR mode failure, MICRO SOURCE at C. This halt code is in LR during loading when Cache is the intended location of the executive program.

@000010@ Cassette read error detected after an executing program has read the cassette to load either S-Memory or Cache Memory.

@000015@ MTR mode failure, MICRO SOURCE at S. This halt code is in LR during loading when S-Memory is the intended location of the executive program.

@000020@ Correctable S-Memory error detected while still in MTR mode.

@000030@ Run mode failure, MICRO SOURCE at C. If possible, this halt code is placed in LR upon a failure if the micro source is Cache.

@000035@ Run mode failure, MICRO SOURCE at S or NORMAL. If possible, this halt code is placed in LR upon a failure if the micro source is NORMAL or S.

@000040@ Cache key parity error detected (PERP bit 1 = 1) while reading Cache key.

@000045@ Cache micro parity error detected while checking micro code loaded in Cache.

@000050@ Correctable S-Memory error detected by a dynamic routine while checking loaded micro code in S-Memory.

@000055@ Uncorrectable S-Memory error detected by a dynamic routine while checking loaded or relocated micro code in S-Memory. This halt is used when the error is detected on a test of bit 3 of the PERM register. The ELOG is saved in S9A (SAVE.ELOG) for display.

@000060@ Uncorrectable S-Memory data error detected by a dynamic routine while checking loaded or relocated micro code in S-Memory. This halt is used when a "read not equal to write" condition is detected.

@000070@ Extraneous port device S-Memory access error. An extraneous access to S-Memory by a port device is detected by the return of an incorrect port/channel code after loading in the MTR mode.

Standard Execution Options

Operator-selected test options can be entered directly via the console switches or by preloading certain scratchpad locations.

Console Switches

The console switches are used for selection of test options during program execution. This feature allows the operator to select or modify options dynamically (without a manual restart procedure).

Console switch contents are periodically sampled by each test program to determine if a change has occurred since the last sampling. The sample is ignored by the program unless the leftmost (MSB) switch is set. This switch, when set, provides a validity bit to prevent sampling of inadvertent residual switch settings or partial entry settings. Thus, to modify options, the operator places this validity switch in the down (reset) position, enters the desired values in the other switches, and returns the validity switch to the up (set) position.

Scratchpad

Certain scratchpad locations are used for selection of test options before execution. These options are entered into scratchpad at a control halt (LR = @AAAAAA@ or @xxAAAA@). The options can be overridden by switching the leftmost (validity) switch up.

To change a preselected option, a dynamic start-of section halt (or enter options halt) must be selected on the console switches. This action causes a control halt (LR = @xxAAAA@ or @AAAAAA@). The desired scratchpad location may now be loaded with the selected option, and the program may be restarted.

The significance of the switch positions for loading scratchpad are explained below and tabulated in Figure 4-2.

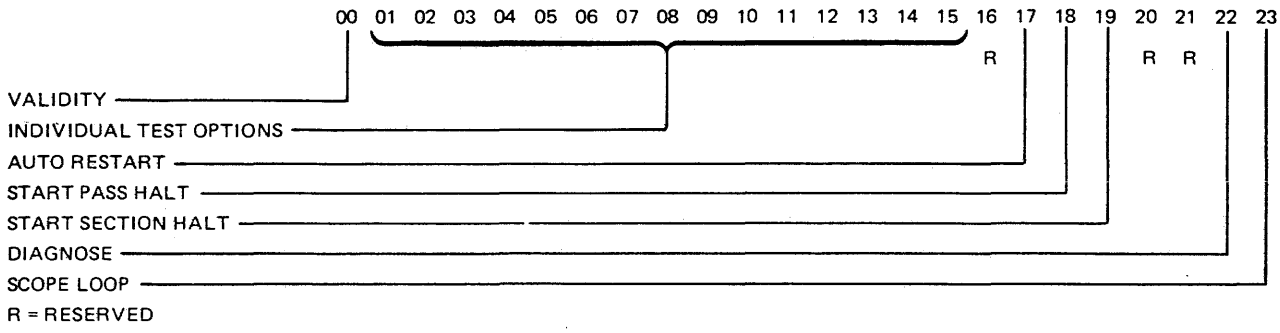


Figure 4-2. Test Execute Option Selection

Bit Significance for Left Scratchpad (S1A) Options

Bit 0 — Validity. If set (up), console switch contents are used; if reset (down), left scratchpad (S1A) contents are used.

Bit 1-15 — Individual test program options.

Bit 16 — Reserved.

Bit 17 — Automatic restart. This bit is checked at least every 10 seconds to determine if a program restart has been requested. If set, the executing program is discontinued and the LR = @AAAAAA@ halt is entered. If the bit is reset, normal execution continues.

This option allows the operator to do an orderly restart from even a long test section without going through the manual sequence.

Bit 18 — Start-of-pass halt. If set, the LR = @AAAAAA@ halt is entered after the test has completed all selected test sections the specified number of times. If reset, this halt does not occur.

Bit 19 — Start-of-section halt. If set, the LR = @AAAAAA@ halt is entered prior to the execution of each test section. If reset, this halt does not occur.

Bit 20-21 — Reserved.

Bit 22 — Diagnose. If diagnosis is possible on an error, the v digit in LR = @xxvnnn@ is set to some defined value. At this time, if bit 22 is manually set, the program will attempt to further isolate the fault. (This may destroy the original conditions relating to the failure.) After the diagnosis routine, the program will reenter the halt, permitting other options to be selected. These options are (1) re-diagnosis (bit remains set), (2) continue program (bit is reset), and (3) lockup for scoping (bit 23 is set).

Bit 23 — Scope loop. If this bit is set following an error halt, the program enters a tight loop (without halts) with sync points for triggering an oscilloscope. The failing condition is repeatedly executed until this bit is reset, at which point the program resumes.

NOTE

Bit 23 is not to be set unless a scope loop is desired because the program will lock up when the first test case is encountered.

This option is not available in all test cases; availability is indicated by the v digit in LR = @xxvnnn@.

Standard Scratchpad Options

Certain scratchpad locations are used to store option designators for use during the test routines. The scratchpad options can be entered during either the LR = @AAAAAA@ or LR = @xxAAAA@ halt. Since scratchpad locations be accessed from the D/M panel, it is expedient to enter the desired options directly. The capabilities provided include selection of all standard run options.

If a test section is deselected at the LR = @xxAAAA@ halt, that section is not executed if it has a test number higher than xx. If the deselected section is equal to or less than xx, it is not reexecuted on the present pass and is skipped on subsequent passes. Modifying Pass Counter, Pass Limit, Loop Counter, or Loop Limit has no effect until execution of the currently selected section (xx) is completed. The scratchpad locations designated Section Number and Run Mode may not be modified by the operator.

The option designators and corresponding scratchpad locations that have common definitions in all central system diagnostic test programs follow.

S0A Run Mode. This scratchpad location holds a value that is ORed with the contents of the MICRO SOURCE switch. This value is not to be altered unless the tape is rewound and a complete reload is to be done. The Run Mode designators are (1) @000000@, S-Memory to Cache (normal execution); (2) @000001@, run from S-Memory only; and (3) @000002@, run from Cache only. Default: depends on test program.

S1A Preselected option storage designator. This scratchpad location holds option bits that are used when the validity bit (bit 0 in the console switches) is reset. Default (no options specified):
XXXX XXXX XXXX XXXX 0000 XXXX.

S2A Test Section Designators. These specify the test sections to be run. Each bit position matches the corresponding decimal test number (bit 9 for test section 9, and so on). Each TRUE bit selects the corresponding test section. Default: run all valid test sections.

S3A Test Section Number Indicator. Indicates the test section currently executing: @xxAAAA@, where xx is the test section number.

S4A Test section loop limit designator. Specifies (in binary) the executions per selected test section to be performed prior to proceeding to the next selected test section. Default: @000001@.

S5A Present Section Loop Counter. Indicates the present loop (the executing test sections) in effect. When this counter matches S4A, the test proceeds to the next selected section.

S6A Test Pass Limit Designator. Specifies (in binary) the number of executions desired for all test sections. test executes continuously (infinite number of passes). Default: @000001@.

S7A Present Test Pass Counter. Indicates the present pass (of the complete test cycle) in effect. When the binary value in S7A matches the value in S6A, the program terminates, with LR = @EEEEEE@ (end-of-test halt).

S8A Cache Block Designator. Specifies the block of Cache (if any) being utilized by the test. This scratchpad location contains the BR value that was set prior to test loading and should not be altered during testing. Default: @000000@.

S9A ELOG Value After Test Case. Contains the ELOG value saved following a specific test case related to S-Memory within the program. If no error condition occurred, the contents of S9A = 0.

NOTES

S10A and higher scratchpad locations are used for special options determined by each specific test program.

Several options may not be preselected through scratchpad and are ignored if so specified. These are: (1) Automatic Restart, (2) Diagnose, and (3) Scope Loop.

Test Descriptions

The six diagnostic programs are described below. Areas of responsibility, test type, special optional features, deviations from the standards (explained earlier), and isolation and diagnostic capabilities are covered. The programs are listed in the sequence in which they would be run in a system "bring-up" environment.

MTR Mode Processor Test

The purpose of the MTR Mode Processor Test is to establish confidence in the basic integrity of those functions needed by the Dynamic Processor Tests to examine all other processor functions. The MTR Mode Test also checks error conditions and D/M panel operations. These latter checks are performed in the MTR Mode Test because they are inconvenient or impossible to accomplish during the Dynamic Processor Tests. The requirements and restrictions inherent in a bring-up, self-test program require a specialized, step-by-step format. For this reason, the execution procedure for the MTR Mode Test differs considerably from the standard test procedure previously described. Because of this departure from standards, a separate set of instructions are provided in the program listing.

All test cases in the program are diagnostic in nature. The general rules to be applied are as follows:

1. Test a function's control logic by means of the Diagnostic Test (10D) micro.
2. Check that the function's result is as expected. When either test fails, the processor is halted. The error indications identify the function being tested, the previously untested control/data paths involved, and the location of this logic. In many cases, enough information is available to indicate which portion of the logic has failed, down to a single card or even a single function on the card.

A minimal amount of Cache is tested in the MTR Mode Test. The purpose of the test is to verify Cache storage needed to execute the Dynamic Processor Test. Successful completion of the tests proves that the processor meets a confidence level at which the Dynamic Processor Test can be executed under default conditions. Option termination at this point may be selected if it is desired to proceed directly to the Dynamic Processor Tests. Otherwise, the MTR program can be continued to test the following:

1. Read/Write S-Memory.
2. Execution from S-Memory (MICRO SOURCE = S MEM).
3. S-Memory error detection and fetch error halts.
4. Console switch compare functions.
5. Execution in the NORMAL mode.

Dynamic Processor Test

The Dynamic Processor Test, in combination with the MTR Mode Test, is intended to verify all processor functions and logic. Building on the level of confidence provided by the MTR Mode Test, verification continues with diagnostic, confidence, and reliability testing. These tests are performed on previously untested as well as partially tested processor logic.

Overlap testing between this program and the MTR Mode Test is handled in two ways:

1. Where the MTR Mode Test contains a comprehensive verification of a logical area, the Dynamic Processor Test considers this logic to be functional and performs only a cursory check prior to using it. In the event of a failure, the error description directs the user to the MTR Mode Test for isolation.
2. Where the MTR Mode Test has checked only part of a function (specific cases) of a micro, comprehensive tests are provided in the Dynamic Processor Test. Failure isolation is then provided for each test case.

The Dynamic Processor Test can be executed using any one of three different Run Modes (micro sources). The default condition is Cache only, in which case each test section is loaded one micro at a time from cassette to Cache and executed from Cache. The user may choose to alter the Run Mode when initially loading the test by setting MICRO SOURCE = S or MICRO SOURCE = NORMAL. At either of these settings, the entire test is loaded from cassette to S-Memory. However, when MICRO SOURCE = S, micros are executed directly from S-Memory, but when MICRO SOURCE = NORMAL, they are fetched from S-Memory, loaded to Cache as needed (that is, on misses, when the needed micro is not already in Cache), and executed from Cache.

The Dynamic Processor Test program is divided into logical areas based on hardware functions and micro groups. Each division is tested in a similar manner as follows:

1. The Diagnostic Test (1OD) micro is used to test the basic functions of any previously untested micros in a group. Some test groups have no new (untested) micros.
2. All variants of each micro are tested, and the associated registers and data paths are tested with basic test patterns on a one-time basis.
3. The logic area is tested with reliability type tests where applicable.

In all cases, detected failures are diagnosed to a functional area on a card. All other operations and procedures are as specified earlier.

Dynamic Host Adapter Test

The Host Adapter Test utilizes a latch on the host adapter (HA-3) card that allows data to pass through this card and a port adapter card, and then echo back to the processor. This program is dynamic and uses the reliability testing methods previously defined. The Host Adapter Test program uses the standard hardware test rules and the following additional options:

1. Scratchpad location S10A is defined as Test Card and designates the card to be tested in Section 2. If the contents of the scratchpad location are all zeroes (that is, not defined by the user) the program attempts to test all port locations sequentially, resulting in errors at each location in which no port adapter card is present.
2. Scratchpad location S11A is defined as User Data and is used to store data entered by the operator for use in test sections where this option is permitted. This stored data is used when specified by the dynamic USE DATA switch. (Bit 1 on the console switches = dynamic; bit 1 in the switch settings scratchpad = preselected.)

The Host Adapter Test program deviates from the standard in the following respects:

1. The Run Mode specifies Cache only during both loading and execution. Section 4 is an exception, it causes an I/O descriptor to be constructed and dispatched from S-Memory.
2. Defaults to the correct Run Mode are made without reporting.

This program, by design, overlaps some functions of both the Dynamic Processor Test and Dynamic S-Memory Test. Refer to either of these programs for further error resolution.

Dynamic S-Memory Test

The Dynamic S-Memory Test verifies the integrity of the storage cards, MBU, portions of the processor/memory interface, and the higher level aspects of all memory related micros. The test is primarily a reliability verification program, but contains both diagnostic and confidence test sections.

The default operating mode of this test is to load each test section into Cache and execute from Cache only. In this case, only one test pass is required to test all of S-Memory.

MICRO SOURCE = NORMAL is an option. In this case, the complete test is loaded into S-Memory starting at location 0. The test micros are then fetched to Cache and executed from Cache. In this mode, S-Memory is tested above the program residence area, then the program is moved higher in S-Memory and the original program residence area is tested. When the program is residing in other than its original area, the D/M panel STATE lamp is on. One pass of this test is defined as testing all selected S-Memory. Each test section executes twice, once with and once without the STATE lamp on, before incrementing the loop counter.

The option of running with MICRO SOURCE = S (S-Memory only) is not available in this program.

As an additional option, the operator may select base and limit addresses for testing. All other features of this test are consistent with the standards described earlier.

The general purpose of the Dynamic S-Memory Test program is to test with the goal of separating the MBU from the storage cards. This is accomplished by means of the diagnostic constructs available for exercising the MBU (11D) micro. The initial memory tests consist of both diagnostic and confidence routines.

When a certain level of confidence in MBU and processor operations has been established, storage card and addressing test sections appear. These test sections, for the most part, are reliability tests. The storage card test sections attempt to isolate the failure to a storage chip. When multiple bits are incorrect, further analysis of the error condition is attempted to determine which logical area on the storage card or in the MBU is at fault.

Dynamic Cache Test

The Dynamic Cache Test serves to verify performance of Cache storage elements and the hit/miss/replace logic in the processor. The test is loaded into S-Memory from cassette, with most test sections being executed exclusively from S-Memory (MICRO SOURCE = S). However, some portions switch automatically to the NORMAL mode (load to S-Memory, execute from Cache). MICRO SOURCE = C (Cache only) is not used.

Proper execution of this test requires prior verification of (1) all non-Cache processor functions, (2) S-Memory storage of data, and (3) S-Memory-only execution.

Program execution proceeds as follows: First, in the S-Memory Only mode, Cache key storage and key-related functions are tested. Following this, the keys are used to address and exercise the micro store area of Cache. The program then enters the Normal mode and uses the tested keys and micro store to check the hit/miss/replace logic (resident primarily in the processor).

In checking the keys, diagnostic tests are run first to ensure that they can store data. Next come confidence tests which yield isolation between address and data lines. This same hierarchy is used in testing the micro store elements. The hit/miss/replace logic tests follow; these begin at the confidence level because several new processor functions are utilized when a miss is encountered. When individual testing of the keys, micro store, and Cache related processor logic have been completed (one at a time to the confidence level), all three sections are exercised together in reliability tests.

FIELD CARD TESTER

The Field Card Tester is provided as a means of external (out of system) troubleshooting for B 1900 logic circuits. This device allows the logic on each plug-in card to be independently exercised and analyzed for proper responses. The Field Card Tester is a portable unit that obtains its operating power directly from the dc voltages present in the central system. Testing is accomplished by pre-programming the tester to generate control signals appropriate to a particular circuit, exercising that circuit, and examining the results. Result reports from the tests are in the form of 4-digit codes (called signatures or node counts) that are compared with published specifications.

Complete operating instructions for the Field Card Tester are contained in the Field Card Tester Technical Manual, Form Number 2101380. Card test data for specific logic cards is a part of the system FT&R documentation.

MANUAL TROUBLESHOOTING

When machine-assisted troubleshooting is not possible or not desirable, several alternate approaches may be employed. These approaches include but are not limited to the following:

1. Console tests by manual manipulation of the controls.
2. In-circuit tests utilizing test equipment (primarily voltage measurements and signal tracing with the aid of an oscilloscope).
3. Physical examination of suspected components, including static testing.

Guidelines

The methods used in manual troubleshooting may vary, but a defined objective and a planned sequence characterizes most successful repair efforts. A typical approach should contain the following elements:

1. Define the problem. This is the most important single requirement and is likely to be the most time consuming. Determine exactly what the system is doing wrong or failing to do. If, for example, the 1C (Register Move) micro is not performed correctly, check the operation of both the source and sink registers, the intermediate data path, and the logic that decodes and implements execution of the micro. Check also to ensure that the micro is reaching the decoding logic properly.
2. Look for obvious solutions. Make sure that a malfunction has actually occurred. Relate problems to recent events such as cleaning, servicing, or the installation of new or replacement components. Look for improperly set controls, accidental disconnections of cable, and other mechanically caused circuit discontinuities. Consider miscellaneous temporary failures such as mechanical jamming of peripheral equipment.
3. Locate the trouble area. Isolate the fault to a functional area of the system by a process of elimination.
4. Analyze the isolated area. Use the logic schematics and appropriate test equipment to identify the faulty circuit element.
5. Correct the fault through repair or replacement. Attempt to determine the cause of the failure, and take the necessary steps to prevent recurrence.
6. Restore the system to operation. Return all components, cables, and subassemblies to their proper locations. Correctly set all controls and verify the repair work by running appropriate test programs.

Voltage Margins

Occasionally, logic problems develop that are marginal and difficult to isolate. Operating the central system with logic voltages set at their upper or lower tolerance levels tends to induce failures at a greater frequency than at the nominal settings. Therefore, use of the voltage margin settings list in Table 4-1 is suggested as an aid in diagnosing intermittent logic problems.

It is usually desirable to adjust more than one supply voltage to the high or low margin at one time in order to stress system weaknesses and possibly obtain better failure indications.

Table 4-1. Voltage Margins

Nominal Settings	Margins	
	High	Low
+5.00 ±0.01	+5.25	+4.75
−2.05 ±0.01	−2.15	−1.95
+12.00 ±0.10	+12.50	+11.50
−12.00 ±0.10	−12.50	−11.50

Documentation Evaluation Form

Title: B 1900 Series Central System, Volume 1: Operation and Maintenance

Form No: 1127388

Date: July, 1980

Burroughs Corporation is interested in receiving your comments and suggestions regarding this manual. Comments will be utilized in ensuing revisions to improve this manual.

Please check type of Suggestion:

☐ Addition ☐ Deletion ☐ Revision ☐ Error

Comments:

From:

Name _____

Title _____

Company _____

Address _____

_____

Phone Number _____ Date _____

Remove form and mail to:

Burroughs Corporation
Documentation Dept., TIO - West
P.O. Box 4040
El Monte, CA 91734
U.S.A.