3A CENTRAL CONTROL AND CONTROL PANEL FUNCTIONAL DESCRIPTION NO. 2B ELECTRONIC SWITCHING SYSTEM

CONTENTS PAGE

CONTENTS PAGE

CONTENTS PAGE

CONTENTS PAGE

TABLES

CONTENTS PAGE

FIGURES

CONTENTS PAGE

1. GENERAL

1.01 This section provides a functional description of the 3A Central Control (3A CC) and the 3A CC control panel. The 3A CC contains all the necessary logic required to direct and control the processing and handling of data within the No. 2B Electronic Switching System (ESS).

3A CENTRAL CONTROL PURPOSE

1.02 The 3A CC (Fig. 1) is an 18-bit (16 data bits and 2 parity bits) switching central control designed for electronic switching systems. The 3A CC is used in a duplex system configuration in the No. 2B ESS. The duplex configuration (Fig. 2) involves the duplication of system components for continuous real-time operation with a high degree of system reliability. One 3A CC always has active control over the system while the other 3A CC operates in a standby mode. A 3A CC, the associated memory, the 2B input/output control circuit, and processor frame power unit form a control unit (CU) which is a single switchable entity. Therefore, each CU is a separate and complete unit capable of controlling the peripherals and system actions. The on-line 3A CC keeps both the on-line and off-line memory up-to-date so that the standby 3A CC can assume control of the system as required.

CHARACTERICS

1.03 The major characteristics of the 3A CC are:

- ***1A Technology:*** This technology, which provides standardized logic gates implemented with silicon intergated circuits (SICs) mounted on ceramics which use automated packaging and interconnections techniques, enables the 3A CC to be small in size, economical in price, and to function at a very fast speed when compared to the CCs of earlier ESS systems.

- ***Self-Checking Circuits:*** The 3A CC uses self-checking circuits to give immediate detection of faults. These circuits eliminate the process of synchronous operation and match comparison between two control units while still providing rapid detection of failures.

- ***Microprogram Control:*** The internal sequencing of actions is controlled by a microprogram structure which results in a highly flexible means of implementing the instruction set and basic control functions. Each instruction is performed by a sequence of microinstructons within the microprogram control. The microcycle time (time to perform one microinstructon) is 150 nanoseconds.

- ***Asynchronous Communication to Memory and the Periphery:*** Additional flexibility is allowed by the asynchronous nature of both the 3A CC-to-memory and 3A CC-to-periphery communication. This means that after ordering the memory or periphery to accomplish a task, the 3A CC can proceed to other tasks without having to further control those units in their performance. At a later time, the 3A CC returns for the memory or peripheral response.

- ***Flexible Input/Output (I/O) Communication:*** A modular, loosely coupled, I/O structure provides the capability of handling a wide variety of I/O devices. This I/O facility is growable by increments of 20 subchannels up to a maximum of 360 high-speed (6.67 megabits per second) serial I/O subchannels.

- ***Large Addressing Capability:*** The 3A CC may directly access up to 1,048,576 words of storage by using a 20 bit address.

- ***Sixteen General Registers:*** General purpose registers in the 3A CC provide flexibility in data handling and processing.

- ***Manual Control and Display Panel:*** The 3A CC has a control panel by which maintenance personnel can gain access to the unit for testing or performing manual operations.

BROAD FUNCTIONAL OVERVIEW

1.04 A block diagram in Fig. 3 shows the functional sections within the 3A CC. The functional sections are:

- ***Microprogram Control:*** The microprogram control is the heart of the 3A CC operation. It directs and controls the operation of the other functional sections within the 3A CC. Since the internal sequencing of actions is controlled by a microprogram structure, a highly flexible means of implementing the instruction set as well as control functions is provided. Each main memory or control function instruction is performed by a sequence of microinstructions executed by the microprogram control. Each microinstruction performs an elementary function such as gating between two registers, loading the data manipulation logic or activating a control signal. The proper sequencing of these elemental functions results in the execution of a main memory instruction (i.e., macroinstruction) or one of the control functions required by the 3A CC operation (i.e., initialization).

- ***System Clock:*** The system clock supplies the basic timing pulses necessary to control system actions.

- ***General Registers:*** The general registers provide a quick access storage medium for storing data being used in the current data processing operation.

- ***Data Manipulation:*** The data manipulation section performs all arithmetic and logic operations upon one or two operands.

- ***Interrupt Facility:*** The interrupt facility provides the means of interrupting the program flow so that a timed or a demanded task may be performed.

- ***Main Memory Control:*** The interface by which information is transmitted to or received from the main store.

- ***I/O Channel and Controller:*** The interface by which information is transmitted to or received from the other units of the 2B processor.

- ***Maintenance Channel and Controller:*** The maintenance channel and controller provide the means by which information is transmitted between the duplicated 3A CCs for maintenance purposes.

- ***Control Panel and Interface:*** The control panel provides one of the means of

communication between the 3A CC and the maintenance personnel.

- ***Gating Bus and Bus Parity Checker:*** The gating bus is the communications path within the 3A CC. The bus parity checker tests the parity of the information placed on the gating bus to ensure its accuracy.

- ***Miscellaneous:*** This section contains a group of special registers and cable receivers. The cable receivers provide a means for receiving information from the system status panel. The special registers are buffers mainly for maintenance, control, status, and error information.

GENERAL DESCRIPTION OF OPERATION

1.05 The 3A CC comprises a large number of registers and their associated control logic circuitry. The 3A CC requests a sequence of commands from the main store (MAS) and translates them into action. This action usually results in the movement of data between the registers in the 3A CC or between the register in the 3A CC and the MAS. The MAS instruction points to the starting address in the microstore of a sequence of microinstruction that will cause the microprogram control to perform a required action. The first microinstruction of an instruction sequence is also typically used to issue a control function to request the next word to be accessed from the MAS. The action is performed by a sequence of microinstruction steps. The initiation of a microinstruction sequence consists of reading a word out of the MAS. This word indicates the location of the first microinstruction to be performed as well as the address of the next microinstruction to be performed.

1.06 At the completion of the sequence of microinstructions, the microprogram control interrogates the data ready flip flop to determine whether the fetch for the next instruction has been completed from the MAS. If not, the microprogram control goes into a loop in which it constantly checks to see if the MAS cycle is complete. When the MAS cycle is complete, the next instruction is loaded into the 3A CC and a new microinstruction sequence is initiated.

1.07 The 3A CC also has the ability to write into MAS. Writing is normally performed in the temporary storage portion of the MAS. The

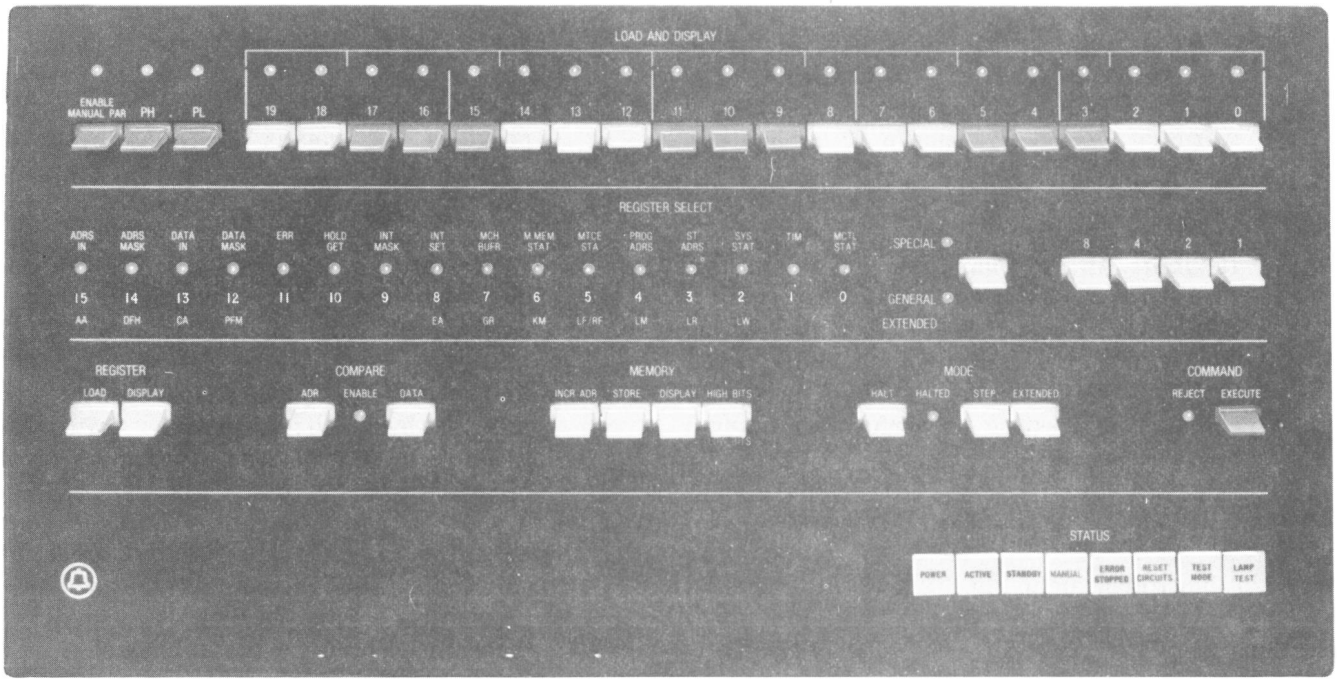


Fig. 1—3A Central Control

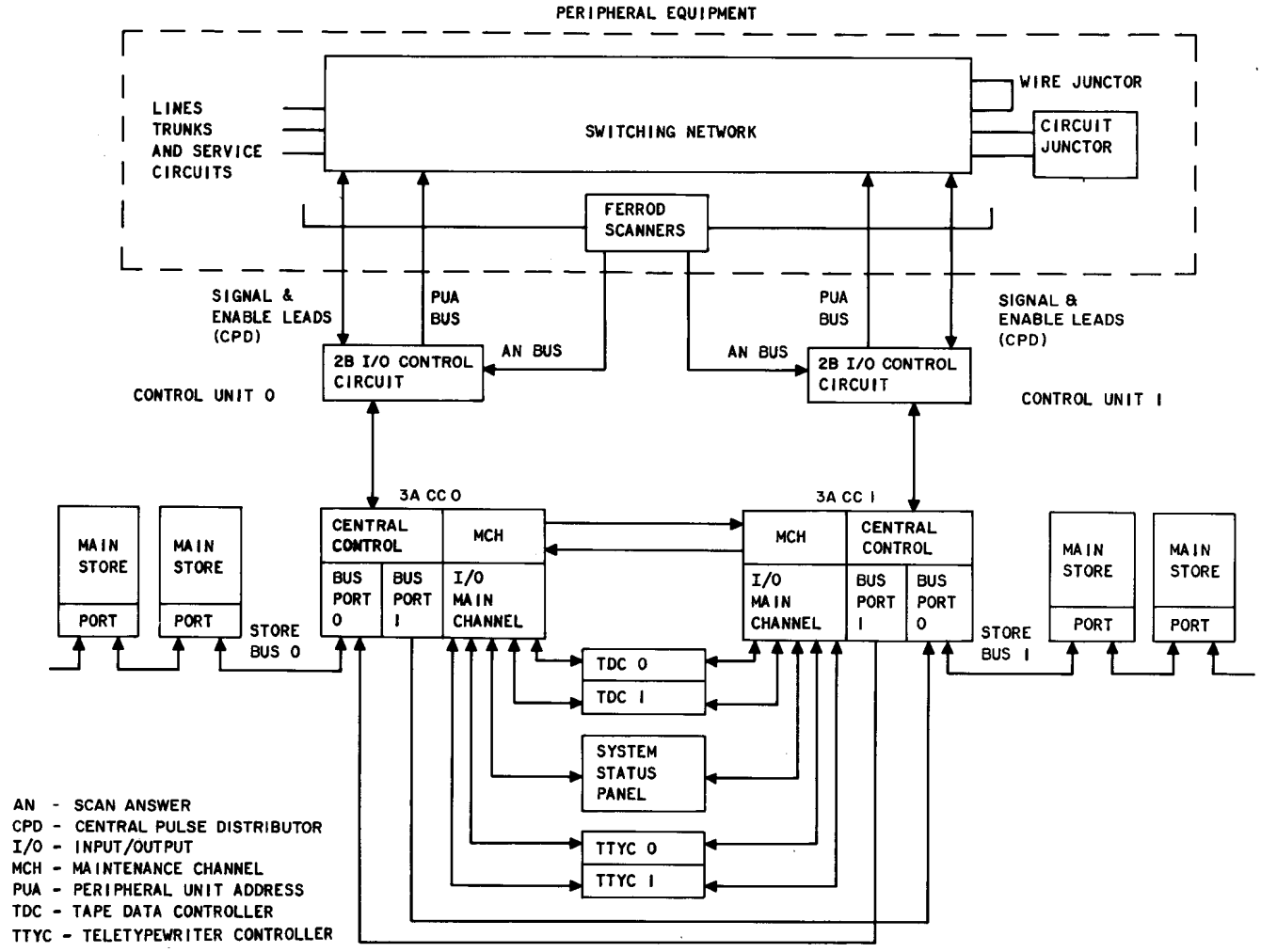


Fig. 2—Block Diagram of No. 2B ESS

other portions of the MAS are write-protected and an ordered sequence of operations is required to allow the 3A CC to write into protected areas of the MAS.

INTERFACE WITH NO. 2B ESS PERIPHERY

1.08 The 3A CC accesses and controls the No. 2B ESS periphery through the 3A CC input/output (I/O) main channel and the 2B I/O control circuit, which is a part of the 2B processor.

1.09 The I/O main channel (see Fig. 2) is one means by which information is transmitted to or received from the other units of the 2B processor. The I/O main channel provides serial AC data at 6.67 mega bits per second and has a 20-subchannel capacity. An I/O subchannel is provided to each main store controller (MASC), tape data controller (TDC), teletypewriter (TTY) controller and system status panel (SSP) (Fig. 4).

1.10 The 2B I/O control circuit performs the interfacing function between the 3A CC and the relatively low-speed peripheral equipment (Fig. 5). The 2B I/O control circuit is the buffer circuit through which inputs are received into the processor and from which outputs are transmitted to the peripheral equipment. Refer to Sections 232-309-101 2B Processor Description, and 232-309-108, Functional Description of 2B I/O Control Circuit,

for more information on the interface functions of the 2B I/O control circuit.

2. PHYSICAL DESCRIPTION

2.01 The 3A CC is located in the upper midsection of each bay of the processor frame (Fig. 6). The position on the frame provides the operator convenient access to the keys and switches of the 3A CC control panel. The 3A CC is 23-1/2 inches wide, 12 inches high, and approximately 14 inches deep. Basically, the 3A CC consists of the logic unit and the control panel.

2.02 All interconnections between the 3A CC and other units are accomplished by one of the following types of cabling techniques (Fig. 7). The first type of cable is a 30-gauge 31-conductor, flat ribbon cable; the second consists of coaxial cable. Both types of cable require a connector and paddleboard assembly at each end. Coaxial cable with a subminiature RF-type connector may also be used in the interconnection of certain units.

3A CENTRAL CONTROL LOGIC UNIT

2.03 A 12-inch mounting plate provides the necessary structure for mounting eight 80A apparatus housings. These housings hold the 3A CC logic unit circuit packs. The 3A CC uses four types of 1A circuit packs (FA, FB, FC, and ED types). Figure 8 shows the arrangement of the circuit packs in the 3A CC.

2.04 The FA-type circuit packs (Fig. 9) may contain a maximum of 52 silicon integrated circuit (SIC) chips mounted on a ceramic substrate which is approximately 3-1/4 by 4 inches in size. The ceramic substrate is mounted on a removable circut board (4 by 7-3/4 inches) with an 82-pin connector. In the 3A CC there are 54 FA-type circuit packs with an average of 43 SIC chips on each (300 gates) for a total of approximately 16,000 gates in the 3A CC.

2.05 The FB- and FC-type (Fig. 10) circuit packs are very similar to each other. Both types contain discrete devices and may contain hybrid integrated circuits (HICs). A HIC is a integrated circuit consisting of SICS or other chip components and thin film devices bonded to a ceramic substrate. The major difference between the FB- and FC-type packs is their means of external connections. The FB-type circuit packs use a 42-pin connector; the FC-type pack uses an 82-pin connector.

2.06 The microprogram store of the 3A CC consists of ED-type circuit packs which are very similar to the FC-type circuit packs. The ED-type circuit packs contain eight programmable read-only memory (PROM) devices, seven T_2L interfacing devices, and pull-up resistors. The PROM devices are 1K dual in-line packages (DIPs) arranged with 265 words by four bits. The ED-type circuit packs use 82-pin connectors.

3A CC CONTROL PANEL

2.07 The control panel consists of a 12-inch by 23-1/2 inch aluminum panel, silk-screened black with the appropriate nomenclature. The panel includes the following apparatus:

(a) Status indicator lamps and switches

(b) Light-emitting diodes (LEDs) which display register contents or main memory address or data information

(c) Register select switches for loading or displaying purposes

(d) Switches for selecting a particular manual function.

2.08 The panel is mounted to an aluminum frame and has a printed wiring board which supports all the apparatus and circuitry necessary for the control panel to function. The entire assembly is hinged to the side brackets of the 3A CC logic unit to provide access to the circuit packs within the unit. Most of the interconnections between the panel and the rest of the 3A CC are accomplished by means of a connectorized flat tape cable assembly. Some interconnections are by the coaxial cable type.

2.09 The panel is subdivided into the following areas:

(a) LOAD AND DISPLAY

(b) REGISTER SELECT

(c) REGISTER

(d) COMPARE

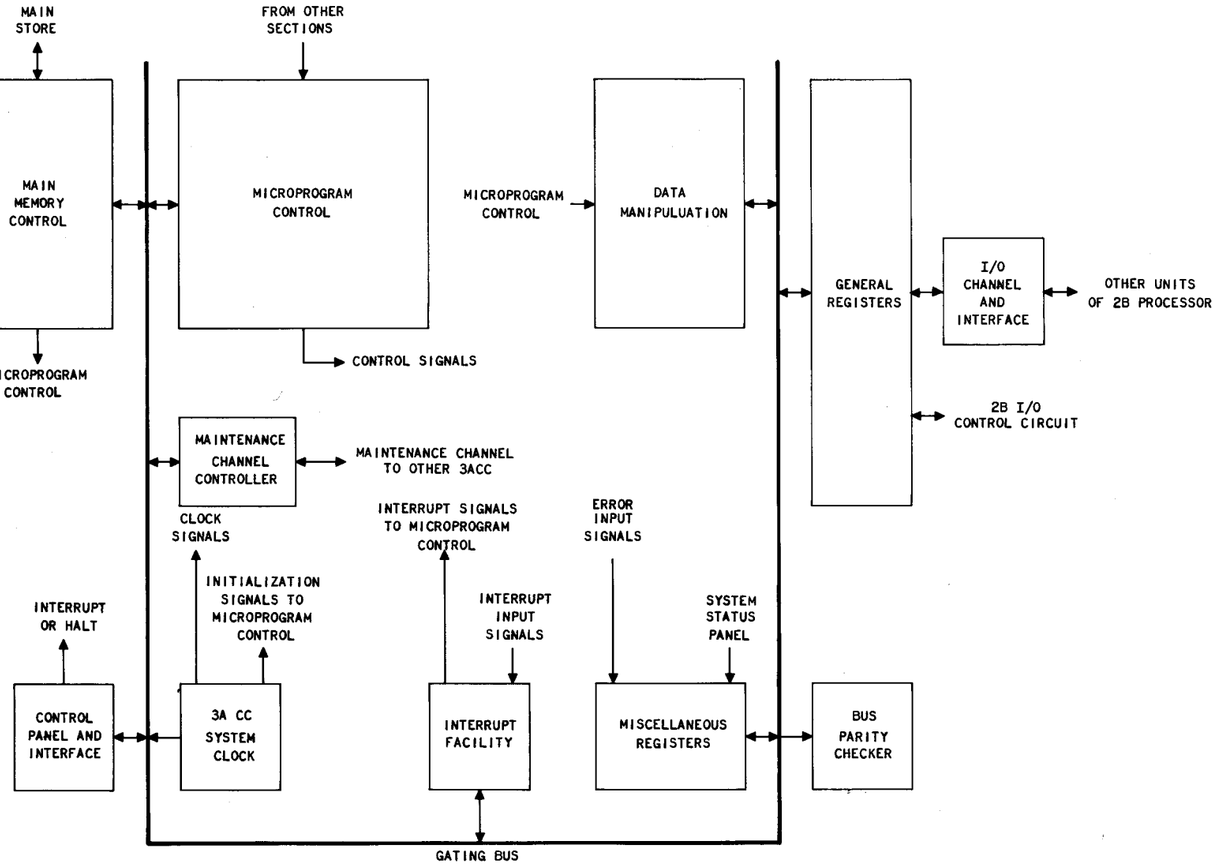


Fig. 3—Block Diagram of 3A CC

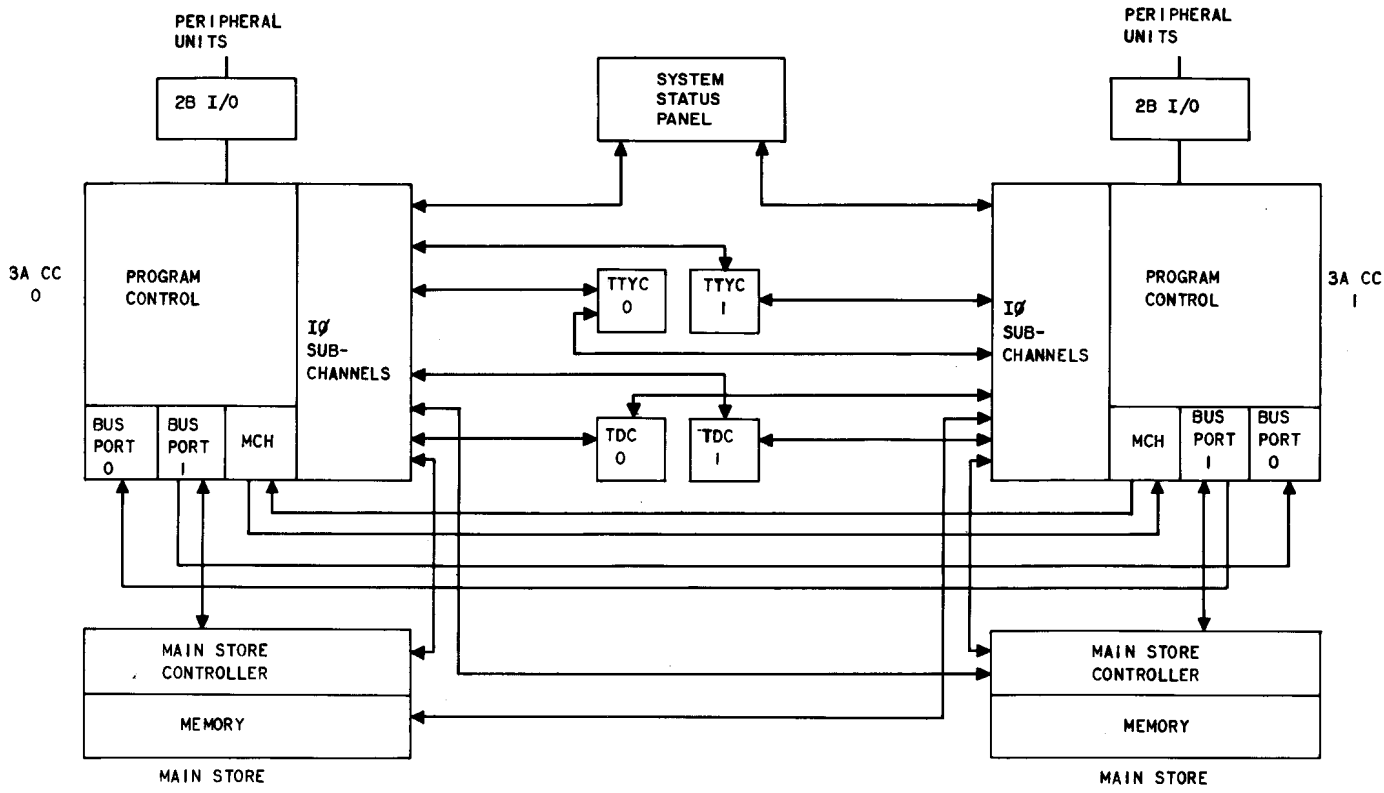


Fig. 4—Block Diagram of 2B Processor Illustrating 3A CC I/O Subchannel Interface

(e) MEMORY

(f) MODE

(g) COMMAND

(h) STATUS

The keys, lamps, and switches of these areas and their purpose are covered in detail in part 5.

3. TECHNOLOGY

BASIC CIRCUITS AND FUNCTIONS

3.01 The 3A CC performs logic and memory functions. Its basic building block is the NOT-AND (NAND) circuit (Fig. 11A) which is used as a gate for logic functions and is combined into flip-flops for memory functions. The output of the NAND gate is low only when all its inputs are high; its output is high whenever any one of its inputs is low. The NAND gate is implemented with a transistor-transistor logic gate (T_2L) as shown in Fig. 11B.

A. Logic Circuits

3.02 A circuit performs a logic function if it generates an output when certain specified input conditions exist. Examples of logic circuits are a single NAND used as a gate and a combination of NAND gates that provide a specified output only when specific input conditions exist. As shown in Fig. 12, NAND gates are combined to meet a specified condition. This circuit also utilizes a collector tie which performs an additional logic function without the use of any additional gates. The collector tie functions as an AND gate and is an attribute of 1A logic. The NAND gates and collector tie are combined to produce a high output at C only when inputs A and B are both high or both low.

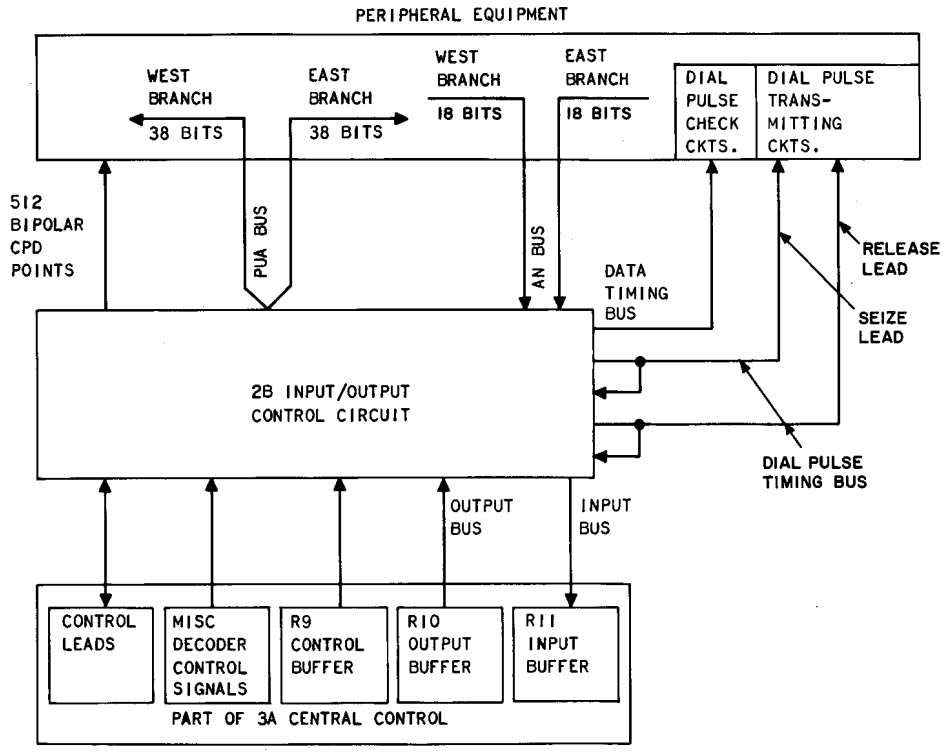


Fig. 5—2B I/O Control Circuit Bus Connections

B. Storage Circuits

3.03 A circuit which is stable in either of two states can perform a storage function and can remain in the state selected. The basic 3A CC storage circuit is a R-S flip-flop which consists of two NAND gates connected so that the output of each gate is an input to the other gate (Fig. 13). This flip-flop has two inputs, set (S) and reset (R), and two outputs, 1-side (1) and 0-side (0). It can acquire and retain either of two states, set or clear, in response to a momentary low signal on input S or R, respectively.

C. R-S Flip-Flop Operation

3.04 Assume that inputs S and R are both high and that the circuit is in the clear state (output 0 high and output 1 low). Since both inputs of the S gate are high, its output is low. This, in turn, keeps one of the inputs of the R gate low and insures that its output is high. Thus, the flip-flop is stable in the clear state.

3.05 If the S input is pulsed low, the output of the S gate goes high. This high is applied to one input of the R gate, and since both of its inputs are now high, the output of the R gate goes low. Since the low output of the R gate is fed back to the input of the S gate, it insures that the output of the latter stays high even when the S input goes high again. Thus, the flip-flop is now stable in the set state and will remain in this state until the R input goes low.

D. Binary Storage

3.06 In order to store a binary 1, a flip-flop is set by the application of a low voltage (binary 0) to its S input. Therefore, its 1-side

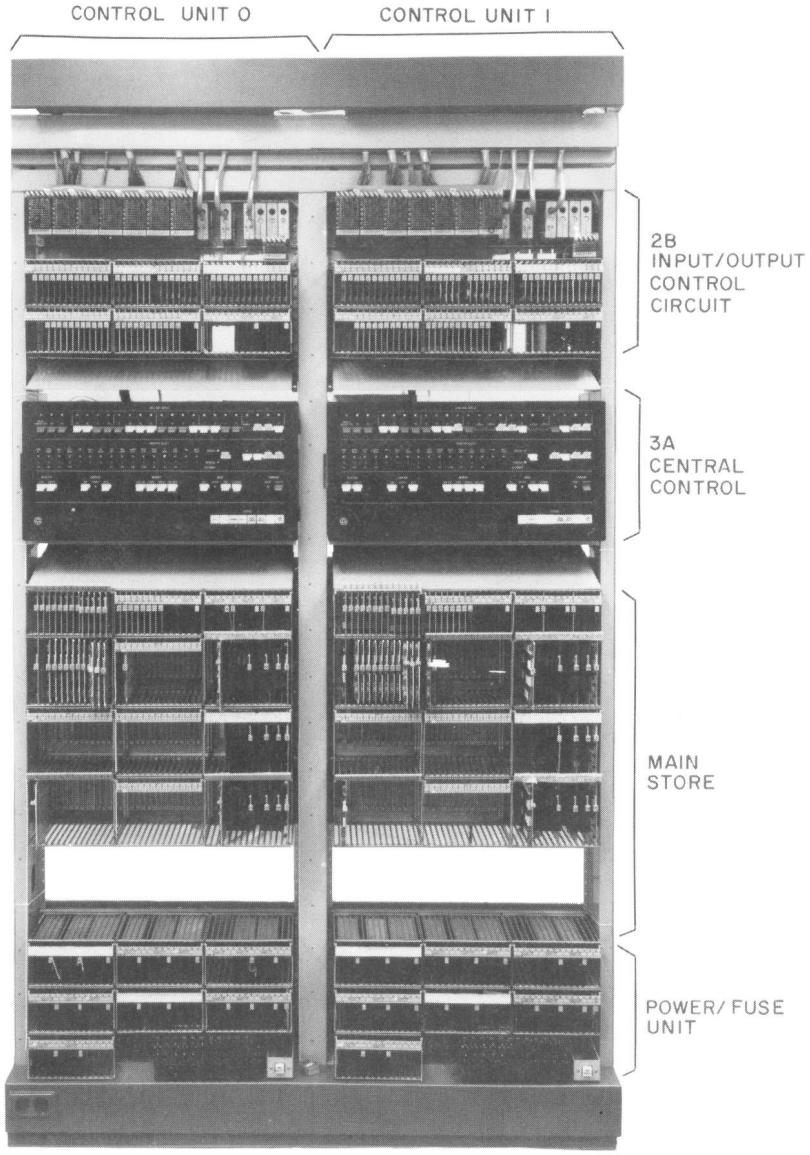


Fig. 6—2B Processor Frame

output is high (H), and its 0-side output is low (L). Similarly, to store a binary 0, a flip-flop is cleared by the application of a low voltage to its C input; its 1-side output is low, and its 0-side output is high.

4. NO. 2B ESS INSTRUCTION SETS

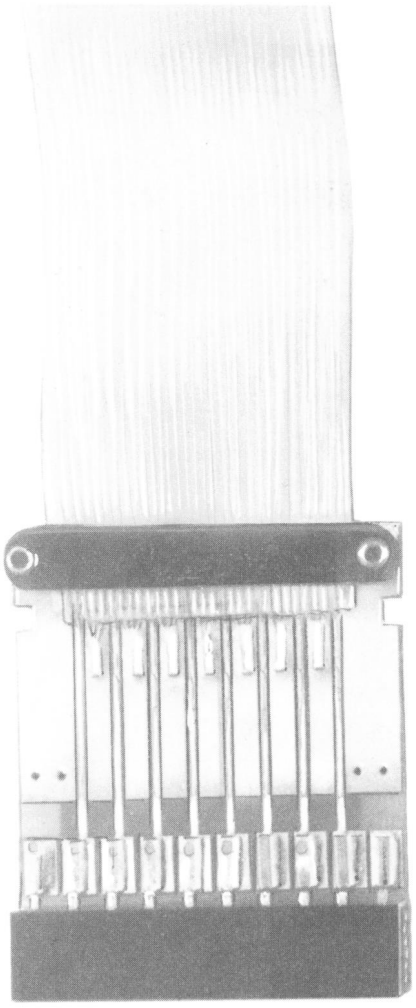
4.01 The No. 2B ESS utilizes two basic instruction sets:

- 3A instruction set
- 2B instruction set

The 3A instructions are used to control the internal sequencing, diagnostics and maintenance of the 3A CC. The 2B instructions are used for the execution of call processing programs, peripheral unit diagnostic and maintenance programs, and unique 2B programs. The 2B instruction set consists of new 2B instructions and approximately 75 percent of the No. 2 ESS instructions. The new 2B instructions are combined with both the No. 2 and 3A instructions for the unique 2B programs.

3A INSTRUCTIONS

4.02 The 3A instructions consist of single (full) word instructions and double word instructions (Fig. 14). Single word instructions are the most commonly used; however, double word instructions are used when either 16 bits of data or a 20-bit address is required in an instruction.



31-CONDUCTOR RIBBON CABLE AND PADDLEBOARD CONNECTOR

COAXIAL CABLES AND PADDLEBOARD CONNECTOR

Fig. 7—Cabling and Connectors

4.03 The 3A CC instructions are general purpose in nature to enable reading from or writing into any of the general registers. Since most of the instructions allow any general register to be used, it is not necessary to move the data to a special register to perform a function.

4.04 The general formats (Fig. 15) for the 3A CC instruction are:

- RR — Register to Register
- RN — Immediate Operand to Register
- RXR — References memory by adding an index register to an address register
- RXN — References memory by adding N to an address register pair
- RI — Register and Immediate Data
- SL—Specified 20-bit data to reference memory and sometimes load a register pair.
- SS — Specified 8-bit OFFSET in branch operation

All are single word type instructions except RI and SL. Each instruction format contains two parity bits, one branch allowed (BA) bit, and a 7-bit operation (OP) code.

REPRESENTS 12-INCH MOUNTING PLATE

REPRESENTS 80C APPARATUS HOUSING

I/O CHANNEL #00

1	2	3	4	5	6	7	8	9	10	11	12	13	14
FC 21	TS 1	FA 1038	FA 1039	FC 201	FC 201								
+3V REGULATOR	I/O CHANNEL 0 TERMINAL STRIP	I/O CHANNEL B#1	I/O CHANNEL B#2	10 I/O SUBCHANNELS	10 I/O SUBCHANNELS								

16	17	18	19	20	21	22	23	24	25	26	27	28	29
TS 2	TS 3	TS 4	TS 5	TS 6	TS 7	FA 1028	FA 1027			TS 8	TS 9	FB 152	FC 21
I/O CHANNEL 2 TERM. STRIP (IN NO. 3 ESS, NO PACK USED)	SYSTEM STATUS, TTYC, TDC, I/O UNIT TERMINAL STRIP	TEST SET CONN (2B PROCESSOR)	TEST SET CONN (2B PROCESSOR)	CONTROL PANEL TERMINAL STRIP	CONTROL PANEL TERMINAL STRIP	ERROR REGISTER AND DISPLAY #2	ERROR REGISTER AND DISPLAY #1			TEST SET CONN (MICROSTORE)	TEST SET CONN (MICROSTORE)	+12V REFERENCE AND CONTROL PANEL POWER TERMINAL STRIP	+3V REGULATOR

1	2	3	4	5	6	7	8	9	10	11	12	13	14
FC 21	FC 21		FA 1031	FB 6	FA 1034	FB 6	FB 6	FA 1010	FA 1010	FA 1010	FA 1010	FA 1010	FA 1010
+3V REGULATOR	+3V REGULATOR (IN NO. 3 ESS, NO PACK USED)		4-OUT-OF-8 CHECKER (TO)	PROTECTION CKT	CONSOLE AND 3A CC INTERFACE	PROTECTION CKT	PROTECTION CKT	BIT SLICE B#1 (6)					

16	17	18	19	20	21	22	23	24	25	26	27	28	29
FA 1010	FA 1010	FA 1010	FA 1020	FA 1026	FA 1013	FA 1012	FA 1015	FA 1012	FA 1012	FA 1015	FA 1012	FA 1014	FB 486
BIT SLICE B#1 (3)			MICROCONTROL B#5	BUS PARITY CHECKER	DML 1 (4)				DML 0 (4)				CRYSTAL OSCILLATOR

31	32	33	34	35	36	37	38	39	40	41	42	43	44
FA 1025	FA 1019	FA 1011	FA 1045	FA 1021	ED4C 063	ED4C 065		ED4C 069	ED2HI 040	ED2HI 042	ED2HI 044	ED2HI 046	FC 21
CLOCK (3A CC)	MICROCONTROL B#4	4-OUT-OF-8 DECODER (FROM)	4-OUT-OF-8 DECODER (TO)	MICROCONTROL B#6	MICROSTORE (BITS 16-31)								+3V REGULATOR

1	2	3	4	5	6	7	8	9	10	11	12	13	14
FC 21	FA 1031		FB 6	FA 1040	FA 1040	FA 1040	FA 1024	FA 1024	FA 1024	FA 1024	FA 1024	FA 1024	FC 21
+3V REGULATOR	4-OUT-OF-8 CHECKER (FROM)		PROTECTION CIRCUIT	SIR1 SDR1 INTERFACE FOR 26-BIT STORE			BIT SLICE B#2 (6)						+3V REGULATOR

16	17	18	19	20	21	22	23	24	25	26	27	28	29
FA 1024	FA 1024	FA 1024	FA 1030	FA 1030	FA 1029	FA 1029	FA 1046	FA 1033	FA 1032	FC 202	FA 1037	FA 1036	FA 1035
BIT SLICE B#2 (3)			MISCELLANEOUS DECODER AND BIT SLICE HIGH		MAIN MEMORY CONTROL		DOUBLE STORE READ	PROGRAM AND TIMER COUNTERS	MISCELLANEOUS CKTS	MAINTENANCE INTERFACE	MTCE CHANNEL B#3	MTCE CHANNEL B#2	MTCE CHANNEL B#1

31	32	33	34	35	36	37	38	39	40	41	42	43	44
FA 1023	FA 1022	FA 1018	FA 1017	FA 1016	ED4C 062	ED4C 064		ED4C 068	ED2HI 039	ED2HI 041	ED2HI 043	ED2HI 045	FC 21
MICROCONTROL B#8	MICROCONTROL B#7	MICROCONTROL B#3	MICROCONTROL B#2	MICROCONTROL B#1	MICROSTORE (BITS 0-15)								REGULATOR

Fig. 8—3A CC Circuit Pack Locations

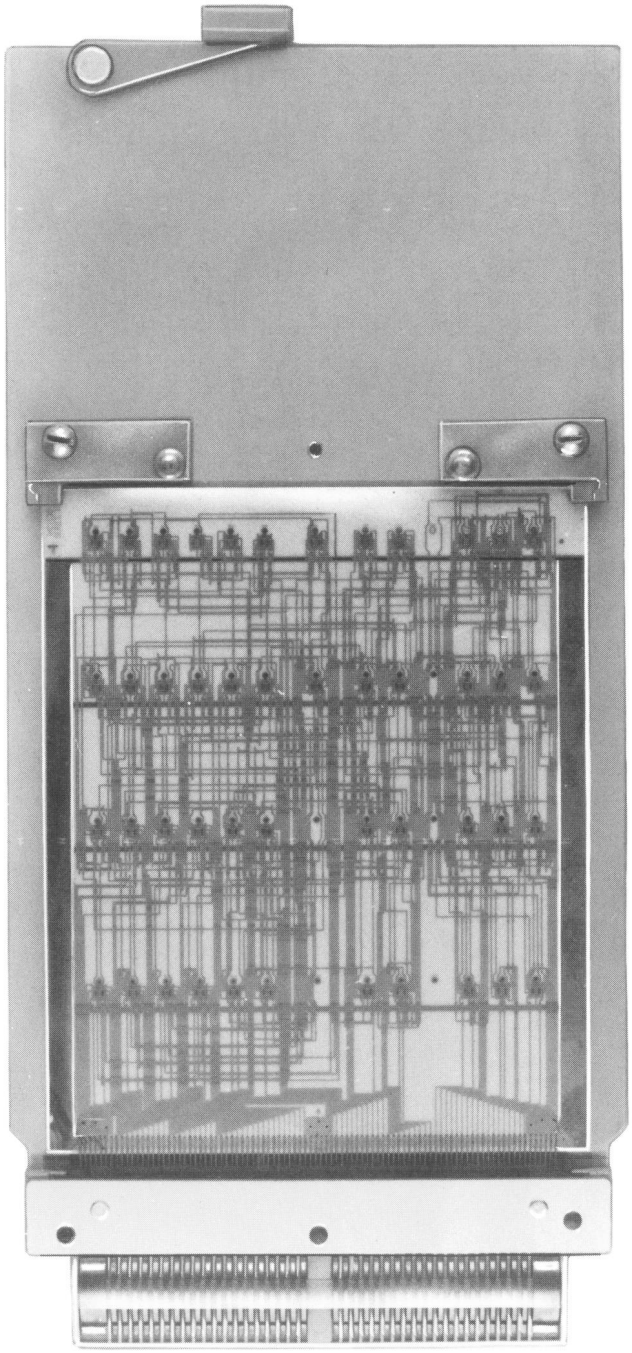


Fig. 9—FA-Type Circuit Pack

4.05 All 3A instructions have odd parity with two parity bits over each 16-bit word. The two parity bits of each word are parity (PL) on the low eight bits (0 through 7) and parity (PH) on the high eight bits (8 through 15).

4.06 The BA bit is used in the program transfer or branch process. When a branch instruction occurs, a hardware check ensures that the BA bit of the next instruction is set. If for some reason the BA bit is not set, an error is indicated in the 3A CC error register.

4.07 The 7-bit OP code field in all instructions specifies the function to be performed. The OP code is used to access a set of microinstructions which accomplish the function indicated by the instruction.

4.08 The remaining bits of both single word and double word instructions contain different arrangements and types of information depending on the particular instruction.

NO. 2B ESS EMULATED INSTRUCTIONS

4.09 The No. 2B ESS emulated instructions consists of full word instructions and half word instruction (Fig. 16). The half word instructions contain two OP codes and associated address fields and are the most commonly used; however, full word instructions are used when 16 bits of data or a 16-bit address is required in an instruction.

4.10 The basic instruction words are 24 bits long. The full word instructions contain one instruction with a 7-bit OP code, 16-bit address and a transfer allowed (TA) check bit. This check bit is used for detecting improper transfers because of an equipment fault or an error in program. The number of full word instructions is rather small. They are used for absolute program transfers and also for supplying constants for various functions.

4.11 The two half word instructions consists of two 12-bit instructions, each with a 7-bit OP code and a 4-bit address and a TA bit. The 4-bit address is used to denote a value or a modifier. For example, a value associated with a rotate instruction specifies the amount of rotation. A modifier associated with a gating operation specifies the data path from one register to another. The OP code is used to access a set of microinstructions which accomplishes the function indicated by the instruction.

4.12 The instruction mapping of half words into the main store is illustrated in Fig. 17. The mapping is not bit for bit. The emulated OP

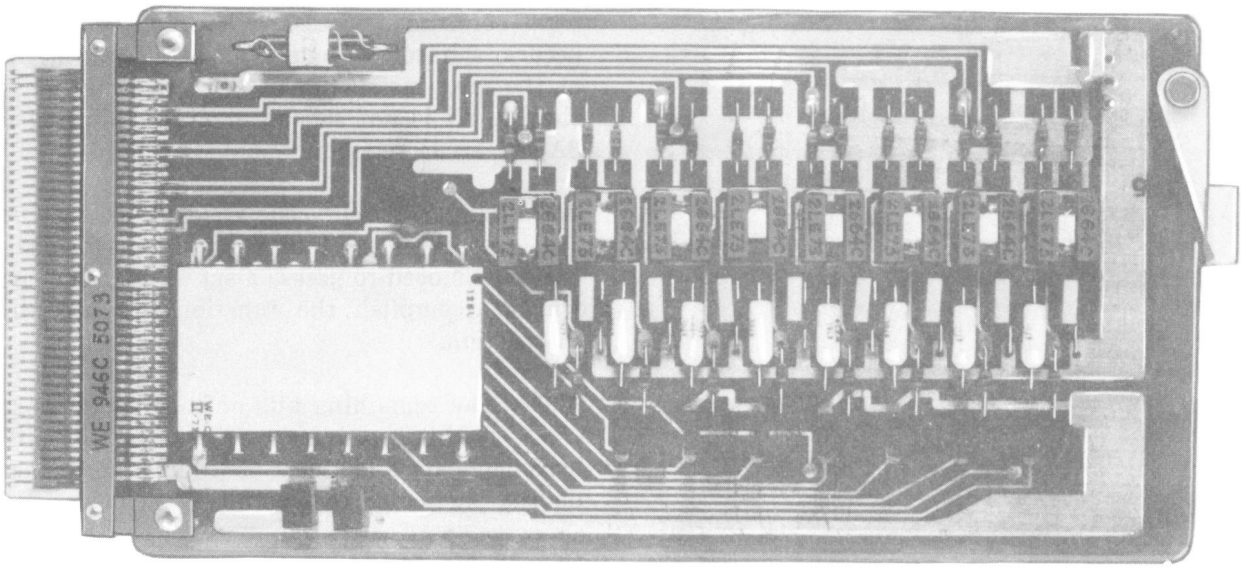
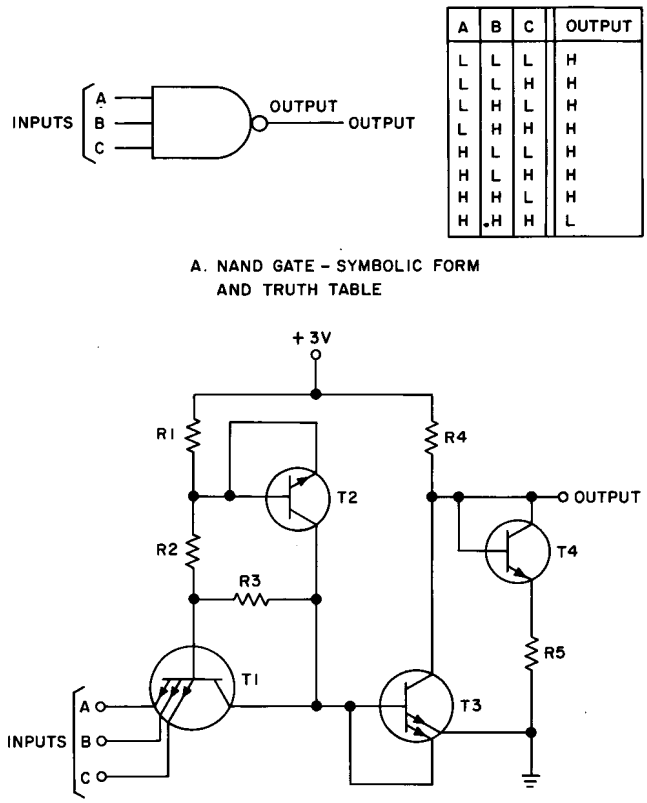
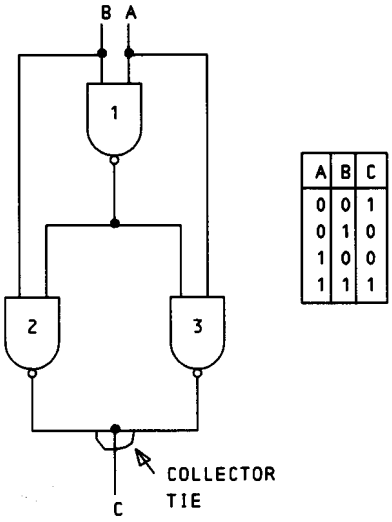


Fig. 10—FC-Type Circuit Pack



A	B	C	OUTPUT
L	L	L	H
L	L	H	H
L	H	L	H
L	H	H	H
H	L	L	H
H	L	H	H
H	H	L	H
H	H	H	L

Fig. 11—NAND Circuit



A	B	C
0	0	1
0	1	0
1	0	0
1	1	1

Fig. 12—Example of Logic Circuit

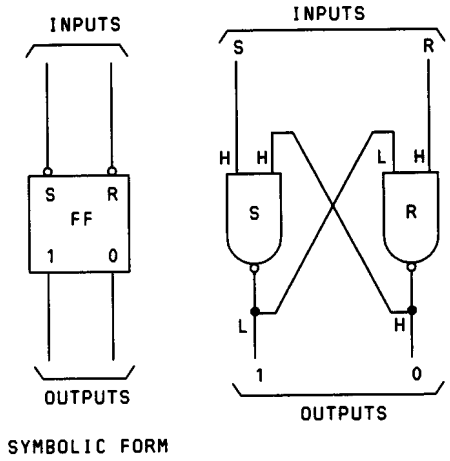


Fig. 13—Basic Flip-Flop Circuit

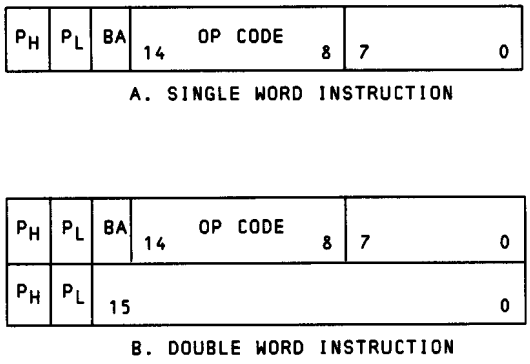


Fig. 14—3A Instructions

codes are different bit patterns than the old No. 2 OP code bit patterns. Figure 18 illustrates the mapping of a full word instruction into a 26-bit store word.

UNIQUE 2B INSTRUCTIONS

4.13 The unique 2B instructions consist of full word and half word instructions similar to the No. 2B ESS emulated instructions (Fig. 16). The OP code and address fields of the 2B instructions have the same bit length as the corresponding fields in the emulated instructions. The 2B instructions are formatted and executed exactly like emulated No. 2B instructions.

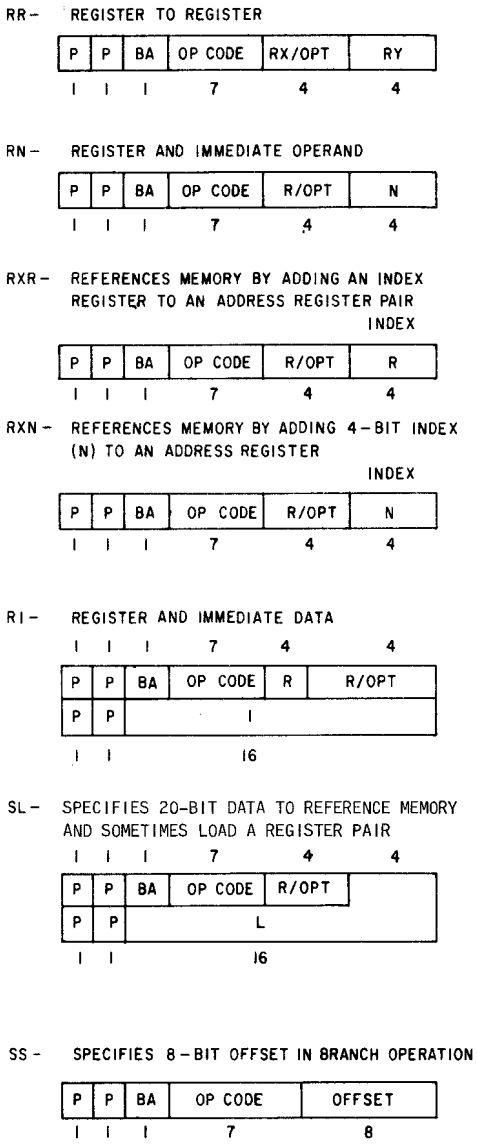


Fig. 15—General Format of 3A Instruction Set

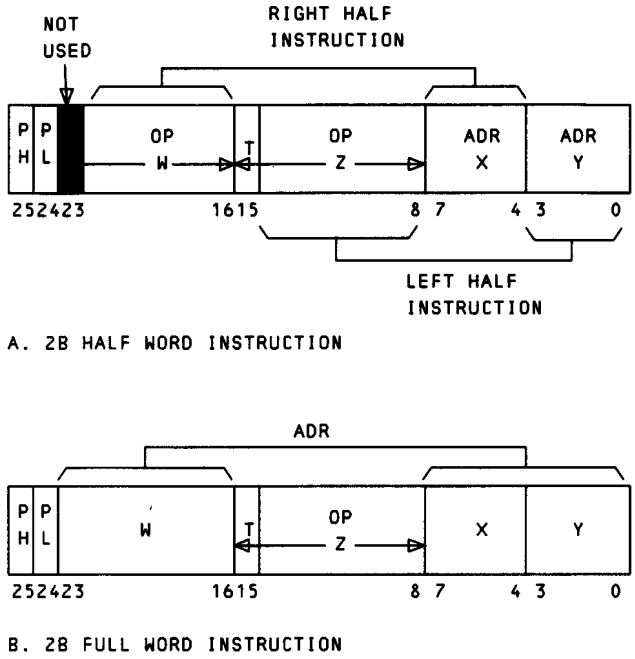


Fig. 16—Unique 2B Instructions and No. 2B Emulated Instructions

5. FUNCTIONAL DESCRIPTION

3A CC FUNCTIONAL RELATIONSHIP TO OTHER SYSTEM UNITS

5.01 The 3A CC is the controlling unit of the 2B processor and the entire system. The 3A CC is duplicated (as well as some other system units) to provide continuous real-time operation with a high degree of system reliability. The 3A CC uses the program instructions and translation data stored in main store to direct and control calls through the office as well as aid in detecting and analyzing improper performance of the equipment involved in this task. One 3A CC normally has active control over the system while the other 3A CC operates in a standby mode. Each 3A CC has its own dedicated main store. The on-line 3A CC keeps both the on-line and standby memory up-to-date so that the standby 3A CC can assume control of the system if necessary with an up-to-date storage area.

5.02 Since the 3A CC via hardware and software controls the operation of the office, it must be able to communicate with various units within the system. This communication involves the sending and receiving of information to and from other units of the 2B processor and certain peripheral units.

A. Communication Functions of 3A CC Within the 2B Processor

5.03 The store bus (See Fig. 2) is provided for communication between the 3A CC and the main stores. The functions performed by the 3A CC in relation to the main stores are the reading from or writing into a memory location via the

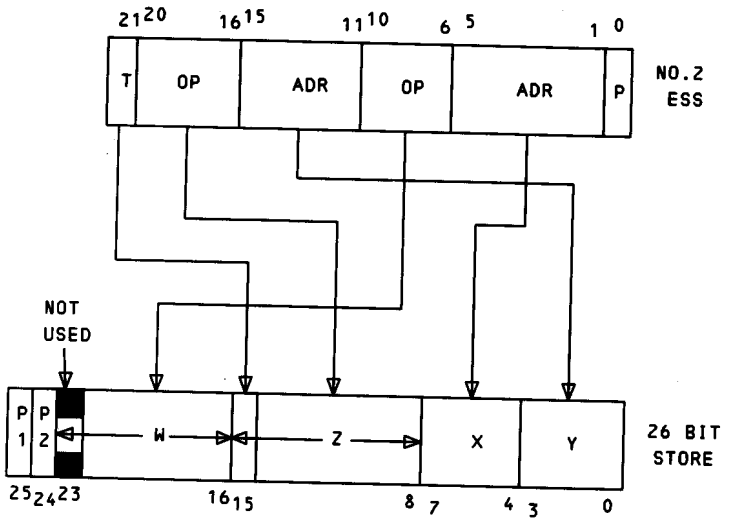


Fig. 17—Mapping of Two Half Word Instructions Into a 26-Bit Store Word

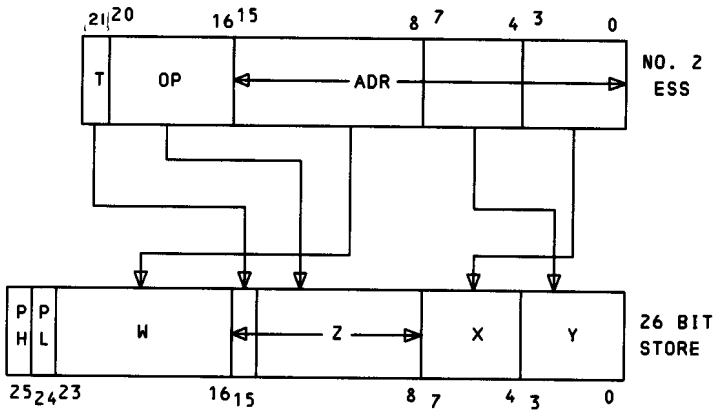


Fig. 18—Mapping of a Full Word Instruction With a 16-Bit Address Into a 26-Bit Store Word

store bus. The 3A CC main memory control provides the interface between the 3A CC and the store bus. The main store control is covered in detail in 5.48 through 5.64.

5.04 The 3A CCs must be able to communicate with each other, since the 3A CC is duplicated for system reliability. The maintenance channel (Fig. 19) provides this communication for diagnostic and control unit switching purposes via the maintenance channel (MCH) controller. The MCH is an asynchronous, semiautonomous data transfer system capable of serial ac data transfers at a rate of 6.67 megabits per second. It provides a half-duplex mode (one-way transmission at a time) communication between the duplicated 3A CCs. The communication is necessary for one 3A CC to determine the state of the other 3A CC and for the on-line 3A CC to exercise the other 3A CC as well. The maintenance channel controller is covered in detail in 5.92 through 5.105.

5.05 The teletypewriter (TTY) and system status panel (SSP) provide an interface between the operating personnel and the system. The 3A CC via a TTY controller communicates with the TTY to perform the functions of outputting characters to the TTY and receiving input characters from the TTY. The 3A CC must also communicate with the SSP to perform the functions of sending

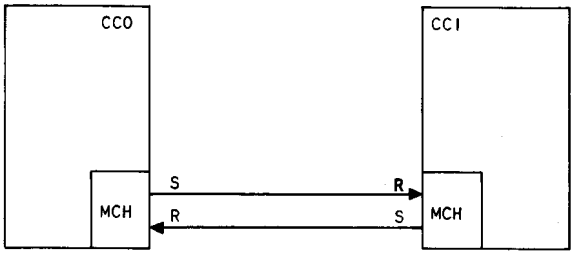


Fig. 19—Maintenance Channel Interconnections

status information to the SSP and receiving manually requested panel operations. Most functions for both TTY and SSP are performed over input/output (I/O) subchannels (see 5.78 through 5.86), however, some SSP functions are hardwired, e.g. FORCE (forces desired CU active).

5.06 The tape data units of the No. 2B ESS provide a backup image of the program and translation data stored in the main stores in case a failure should mutilate the store contents. The 3A CC must communicate with the tape unit to perform the functions of reading data from the tape or writing information on tape. These functions are performed between the 3A CC and the tape unit over an I/O subchannel.

B. Communication Functions of 3A CC in Relation to Periphery

5.07 The 2B I/O control circuit (Fig. 5), performs the interfacing function between the high-speed processor and the relatively low speed peripheral equipment. The 3A CC has nine miscellaneous decoder control signals and three general purpose registers associated with I/O operations. The miscellaneous decoder control signals are generated by the microprogram memory in the 3A CC. These control signals are used in the 2B I/O control circuit to enable gating paths, set or reset flip-flops, or initiate the execution of central pulse distributor or peripheral unit address pulses to control system units which provide service. The 2B I/O control circuit is the buffer circuit through which inputs are received into the processor and from which outputs are transmitted to the peripheral equipment. Every function performed by the I/O control circuit is initiated by microprogram control leads from the 3A CC. The functions of the 2B I/O control circuit are covered in detail in Section 232-309-108.

3A CC LOGIC UNIT

5.08 The functional schematic of the 3A CC logic unit is shown in Fig. 20. The schematic has been divided according to functions into the following areas:

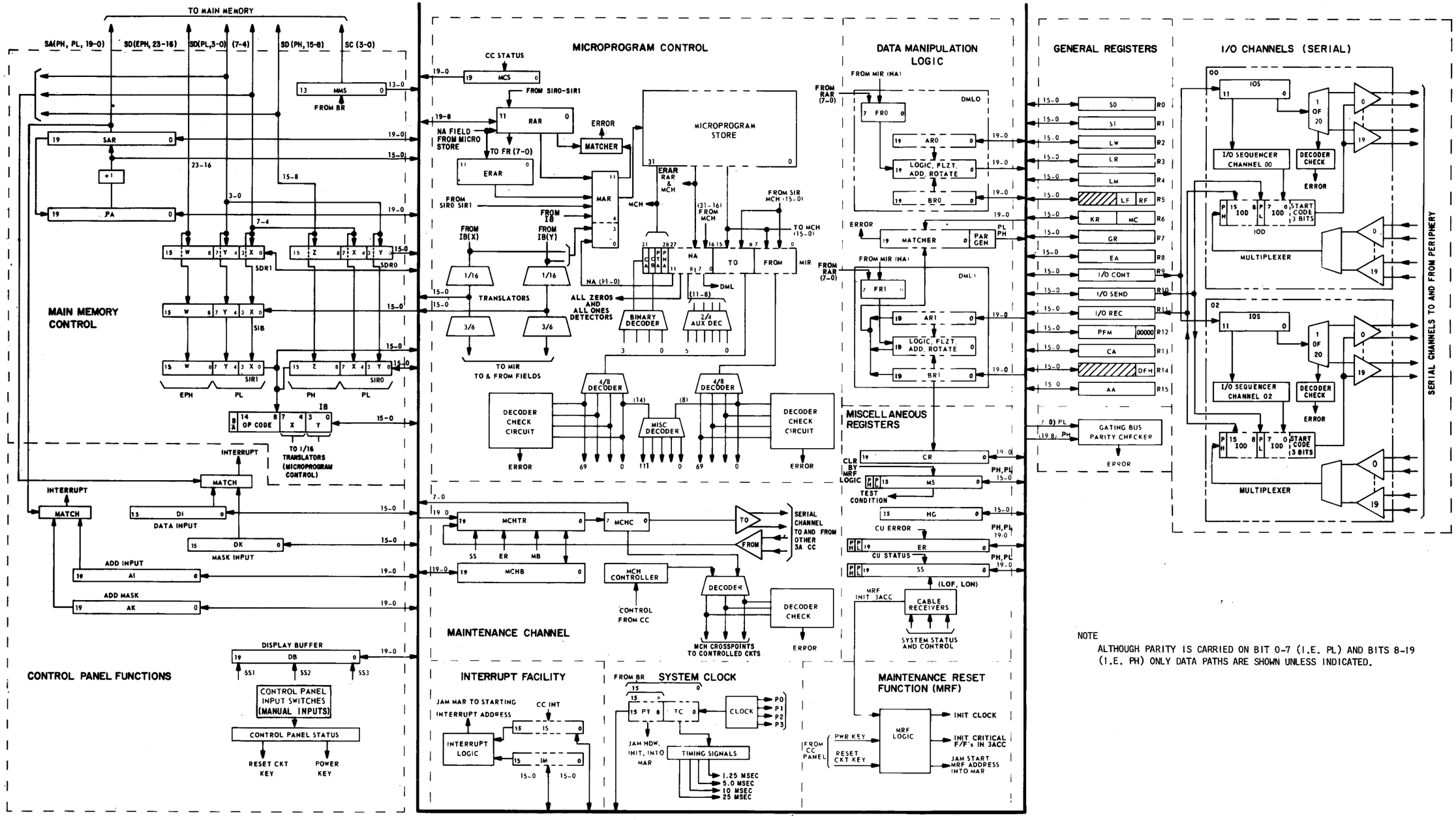
- System clock
- 3A CC registers
- Microprogram control
- Main memory control
- Gating bus and bus parity checker
- Data manipulation logic
- I/O channel and controller
- Interrupt facility
- Maintenance channel and maintenance channel controller
- Control panel and control panel functions
- Miscellaneous

The different functional areas will be discussed in a logical sequence which will aid in understanding their relationship of one to another.

A. System Clock

5.09 The system clock supplies the basic timing pulses necessary to control system functions such as data timing, gate, control, synchronization of events, etc. The system clock generates timing pulses for the 2B I/O control circuit and four sections of the 3A CC:

- Microprogram control



NOTE

ALTHOUGH PARITY IS CARRIED ON BIT 0-7 (I.E. PL) AND BITS 8-19 (I.E. PH) ONLY DATA PATHS ARE SHOWN UNLESS INDICATED.

Fig. 20—Functional Schematic of 3A CC

- Main store control
- Maintenance channel controller
- I/O channel sequencer.

5.10 The basic timing signal is generated by a standard crystal oscillator and squaring circuit. The crystal oscillator generates a sine wave with a period of 37.50ns. The sine wave is passed through a squaring circuit to generate a square wave. By the use of a couple of flip-flops and combinational logic, four clock phases are generated (Fig. 21). To provide some additional timing intervals a 10-bit counter is incremented every 150 ns by clock phase P2.

5.11 The prescaler, timing counter (TC), and program timer (PT) used in the 3A CC are shown in Fig. 22. Bit 6 of a 10-bit counter (located in the clock) increments the 4-bit prescaler by one every 19.2 usec. The prescaler converts the 19.2 usec clock into a 153.4-usec output which is inputted into the TC to generate the 5 msec timed interrupt. The 25-msec interrupt signal is used to increment the PT. Only the 5-msec signal is wired as an interrupt. The 25-msec signal is used to drive the PT but the 25-msec interrupt is software generated. The 8-bit PT is used in the switching and recovery process of the 3A CC to detect any fault (hardware or software) which causes the 3A CC to stray from the normal execution of its main loop. Normally, the PT is updated (reset) before timing to bit 14. It is a function of the software to routinely reset the program timer to prevent initializations. However, if a time-out to bit 14 occurs, the action is dependent on whether the 3A CC is on-line. This time-out may be caused typically by either software or undetected hardware failures.

5.12 If the 3A CC is on-line, time-out to bit 14 of the PT causes the 3A CC to stop and send a switch message to the standby 3A CC via the maintenance channel (MCH). The standby 3A CC verifies the switch message, goes on-line, and initializes itself.

5.13 If the 3A CC is in standby, time-out to bit 14 causes the standby 3A CC to initialize and restart itself. After restarting, the standby 3A CC checks the status of the on-line 3A CC. If the on-line 3A CC has stopped without sending a switch message, the standby 3A CC will switch on-line and start processing data.

5.14 If the PTs are not reset and a time-out to bit 15 occurs in either the on-line or off-line 3A CC, the 3A CC does an initialization and restart. This time-out would occur if the off-line 3A CC was incapable of running and the on-line 3A CC had stopped and sent a switch message to the off-line 3A CC.

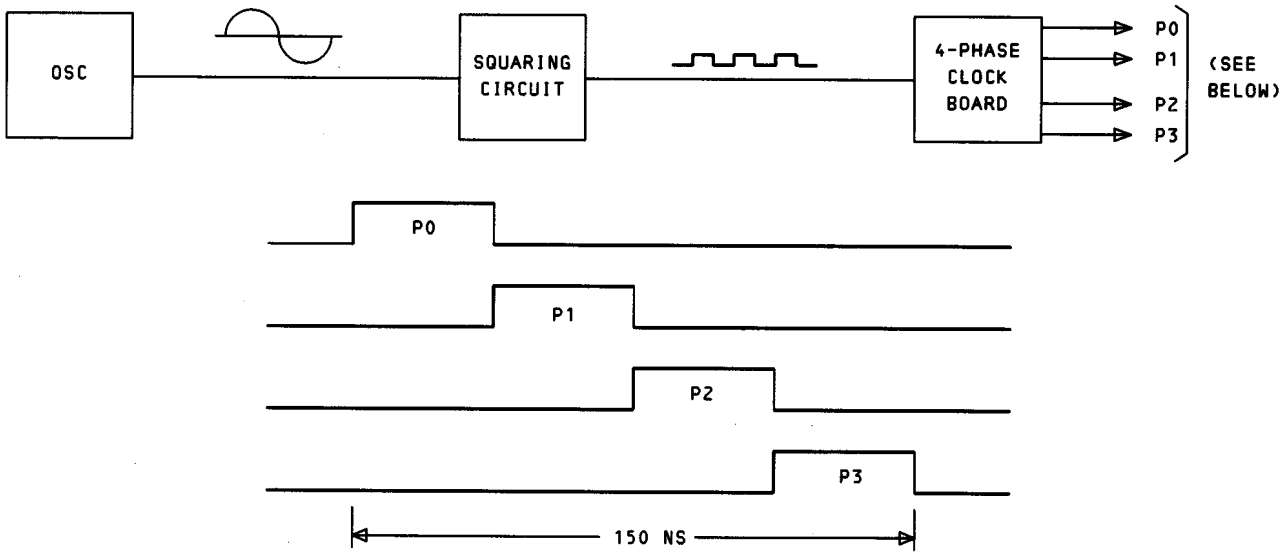


Fig. 21—Clock Pulse Waveforms

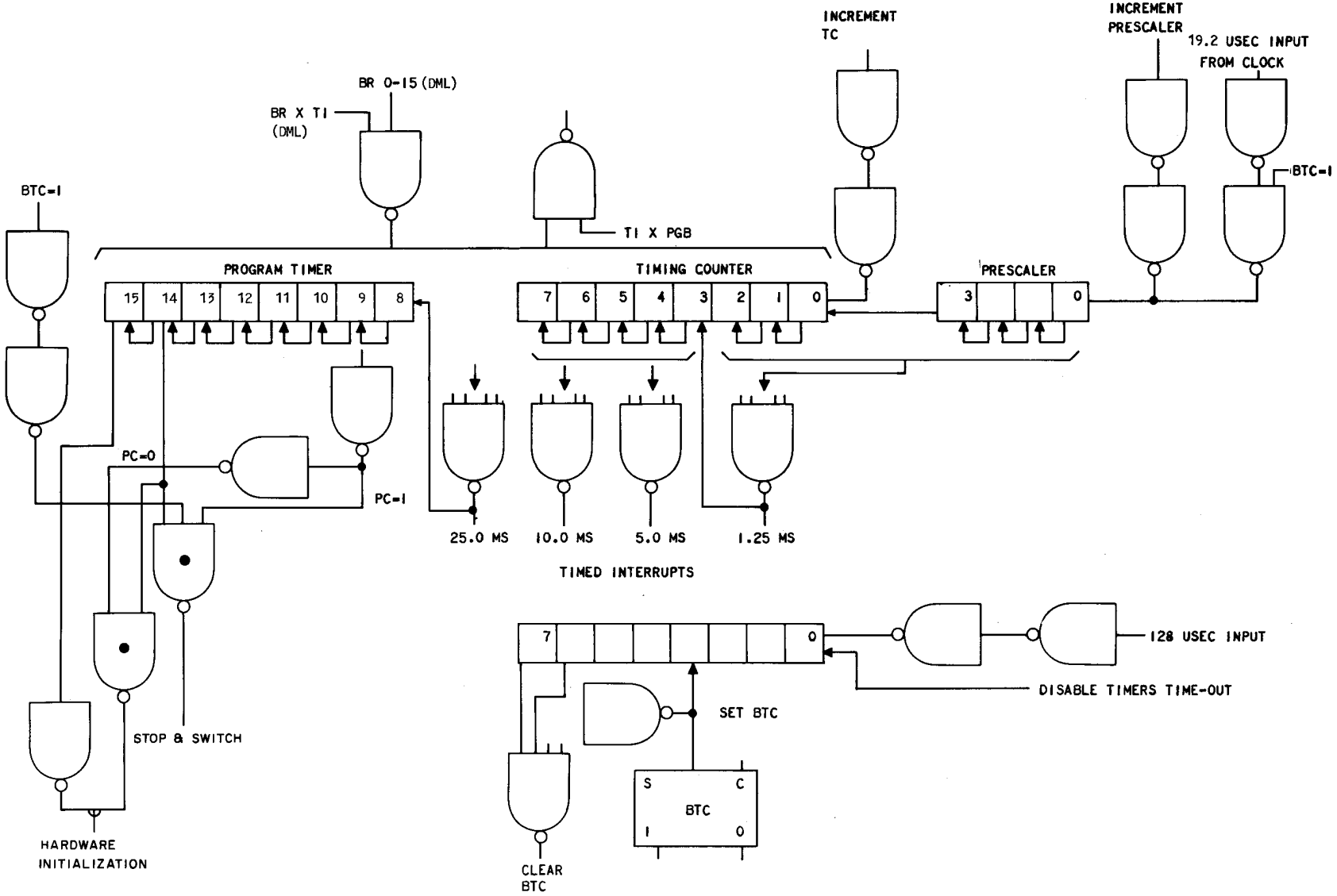


Fig. 22—Prescaler, Timing Counter, and Program Timer

5.15 If the on-line 3A CC is executing the main loop of its program properly, the PT is set after every completon of the main loop. The on-line PT is loaded with a constant so it will reach its first time-out in 300 msec. At the same time that the on-line 3A CC sets its PT, a command is sent to the standby 3A CC to set its PT. The off-line PT is cleared and will reach its first time-out in 1.6 sec. The on-line 3A CC is set to a constant while the standby 3A CC is set to 0.

B. 3A CC Registers

5.16 A register consists of a number of flip-flops which are combined to provide a means of storing data. Each flip-flop can store one bit of information. An example of a 2-bit register cell implemented with 1A logic is shown in Fig. 23. Nine 2-bit register cells are used to form an 18-bit register.

5.17 The two basic types of registers in the 3A CC are as follows:

- General registers
- Special registers.

5.18 The general register organization provides for flexibility in data handling and processing. The general registers are designated R0 through R15. Each general register is a 16-bit register with 2 parity bits (parity on bits 0 through 7 and parity on bits 8 through 15). These registers provide a set of general purpose program addressable registers that are used for high-speed buffer storage for data being stored in a current data processing operation.

5.19 To formulate 20-bit memory address information, a pair of general registers are utilized. Either registers R12 (bits 0 through 3) and R13 (bits 0 through 15) or R14 (bits 0 through 3) and R15 (bits 0 through 15) can be used in this way.

5.20 General registers R9, R10, and R11 serve particular functions concerning the I/O:

(1) R9 as a buffer for the control information to be sent to the I/O facility from the 3A CC

(2) R10 for the data to be sent to the I/O facility

(3) R11 for the results or data received from the external units via the I/O facilities.

When these registers are not being used for their I/O functions, they are used as general registers.

5.21 The special registers are dedicated to specific functions and depending on that function, may vary in length e.g., 20-bit store address register (SAR) and 16-bit store data register (SDR). Only 16 of the special registers may be displayed, loaded, or accessed by operation of the control panel. These procedures are covered in Section 232-306-301. The special registers are used for:

(1) Microprogram control and sequencing

(2) Interface to the main store

(3) Buffering data for microprogram control operations

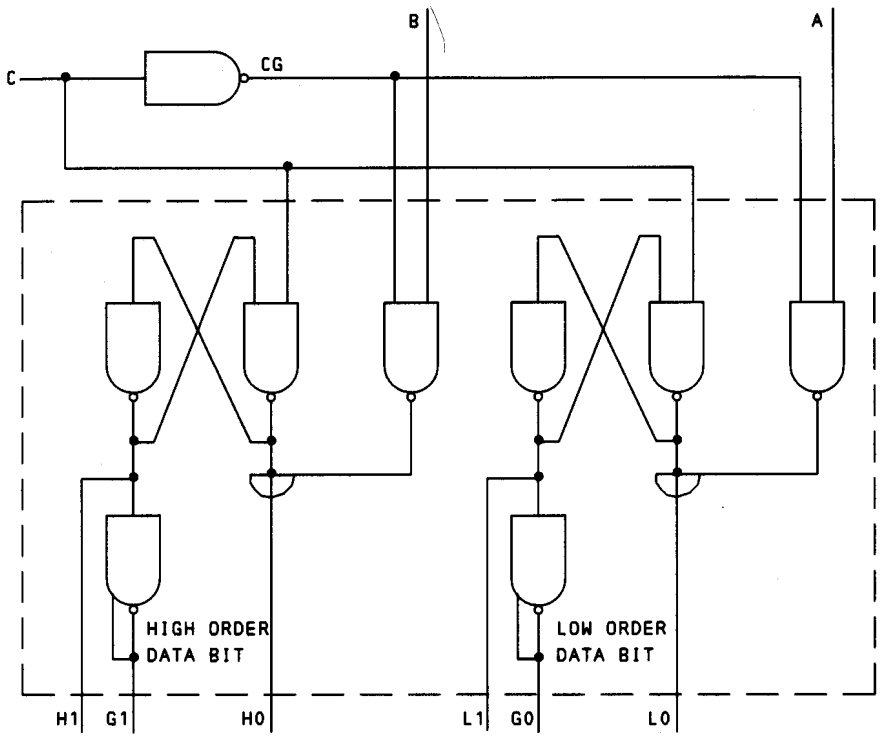
(4) Maintenance purposes.

Special registers and their checking schemes are discussed later in this section under their individual applications.

5.22 The general registers and most of the special registers are 2-bit partitioned or 2-bit sliced. This means that two bits of each register are on a single circuit board (Fig. 24), e.g., the first circuit pack contains bits 0 and 8 of every general register. Partitioning is used so that a fault will affect at most only two bits of any register and therefore be detected by the two parity bits.

C. Microprogram Control

5.23 The microprogram control (Fig. 25) acts as a translator. When an instruction is read out of main store, the microprogram control accesses a particular area of microcode in the microstore. The operation (OP) code of the instruction fetched from main store points to the starting address of the sequence of microinstructions (microsequence). The number of microinstructions in a microsequence range from 2 to 25 depending on the complexity of the instruction. The microsequence translates the OP code of the instruction and generates the gating and sequencing required to execute the instruction. Most instructions involve taking information from a register, placing the information on a bus (usually the gating bus) and gating the



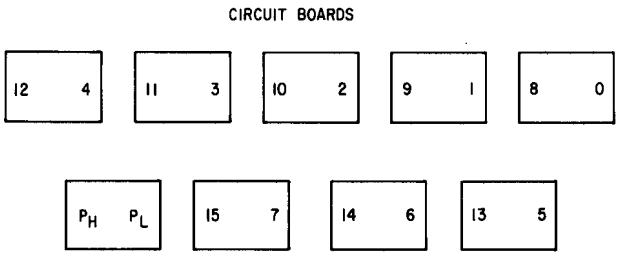
LEGEND

A = LOW ORDER BIT DATA LEAD
B = HIGH ORDER BIT DATA LEAD
C = COMMON CLEAR AND SET LEAD
CG = COMMON INPUT GATING LEAD
G0 = GATED OUTPUT FOR LOW ORDER BIT
G1 = GATED OUTPUT FOR HIGH ORDER BIT
H0 = HIGH ORDER BIT 0-SIDE OUTPUT
H1 = HIGH ORDER BIT 1-SIDE OUTPUT
L0 = LOW ORDER BIT 0-SIDE OUTPUT
L1 = LOW ORDER BIT 1-SIDE OUTPUT

INPUTS			OUTPUTS					
C	B	A	H1	H0	G1	L1	L0	G1
0	0	0	0	1	1	0	1	1
1	0	0	0	1	1	0	1	1
0	0	1	0	1	1	1	0	0
1	0	1	0	1	1	1	0	0
0	1	0	1	0	0	0	1	1
1	1	0	1	0	0	0	1	1
0	1	1	1	0	0	1	0	0
1	1	1	1	0	0	1	0	0

NOTE:
TO CLEAR OR SET EITHER BIT, THE C LEAD MUST BE PULSED LOW WITH THE APPROPRIATE SIGNAL ON THE DATA LEAD. COMMON INPUT GATING IS PROVIDED BY CG. TO STORE A BINARY 1 IN THE HIGH ORDER BIT AND/OR LOW ORDER BIT, THE APPROPRIATE DATA LEAD MUST HAVE A HIGH VOLTAGE WHEN A LOW VOLTAGE IS APPLIED TO THE C LEAD. SIMILARLY, TO STORE A BINARY 0, THE APPROPRIATE DATA LEAD MUST HAVE A LOW VOLTAGE WHEN A LOW VOLTAGE IS APPLIED TO THE C LEAD. THE COLLECTORS OF THE INPUT GATES ARE TIED DIRECTLY TO THEIR RESPECTIVE FLIP-FLOP ZERO OUTPUTS TO PROVIDE THE FAST SET FEATURE.

Fig. 23—2-Bit Register Cell



P_L = PARITY OVER BITS 0,1,2,3,4,5,6,7

P_H = PARITY OVER BITS 8,9,10,11,12,13,14,15

Fig. 24—Layout of General Registers

information to another register. The signals from the TO and FROM decoders control the data transfer operation. The microstore contains many microsequences used not only to translate commands but other functions such as, interrupts, panel operations, program loading, control functions, etc. The microprogram control is made up of the following:

- Microprogram address register
- Microprogram store
- Microinstruction register
- Return address register
- Error return address register

- Decoders and translators
- Microcontrol status register

5.24 ***Microprogram Address Register (MAR):*** The MAR is used to address the microstore. The address may be loaded into the MAR by one of the following (Fig. 26):

(a) The store instruction registers (SIR0/SIR1) after a main store instruction fetch is completed.

(b) The next address (NA) field of the microinstruction register (MIR).

(c) The return address register (RAR) when implementing a subroutine return.

(d) By incrementing the MAR. One is added to the MAR when the MIR control bits (CA and CB) indicate that the MIR NA field is to be used as data or auxiliary control (see 5.32 and Table A).

(e) By indexing the MAR by the IB x or y fields.

(f) Loaded to interrupt starting address.

(g) Jammed to stop address or initialization (MRF) address.

(h) The error return address register (ERAR) when implementing microsubroutine return after complement correction.

5.25 ***Microstore:*** The microstore is a programmable read only memory (PROM) with a maximum access time of 65 nanoseconds (ns). The microstore has a maximum size of 4096 32-bit words. Part of the microstore consists of six circuit packs for storing the 3A microcode. The other part consists of eight circuit packs for storing the 2B microcode. Each circuit pack contains 512 words (16 bits in length). Since each microprogram word is 32 bits in length, two circuit packs are addressed on each microstore cycle. The microstore is growable in increments of 512 words to a maximum size of 4096 words. For each 512 words added, two additional circuit packs are needed. Space is available within the 3A CC to equip the maximum size microstore.

5.26 The microcode stored in the microstore ***cannot be altered*** by the processor. The only way it can be altered is by replacing a circuit pack with one that is encoded with different information. The microstore is addressed by the MAR and the output is gated to the microinstruction register (MIR).

TABLE A

STATE AND USE THE TWO CONTROL BITS IN MICROINSTRUCTION REGISTER

CA	CB	STATE	USE
0	0	Null or Normal	Normal sequencing where no control function is required and the NA field is gated to the MAR.
0	1	Main Store Instruction Fetch	Initiates a new main store operation.
1	0	Data Control	Inhibits the normal sequencing and jams a one into bit zero of the last microstore address in the MAR. This is for data operations. The NA field contains data to be gated to some destination register other than the MAR.
1	1	Auxiliary Control	Enables an auxiliary decoder attached to the upper four bits of the NA field.

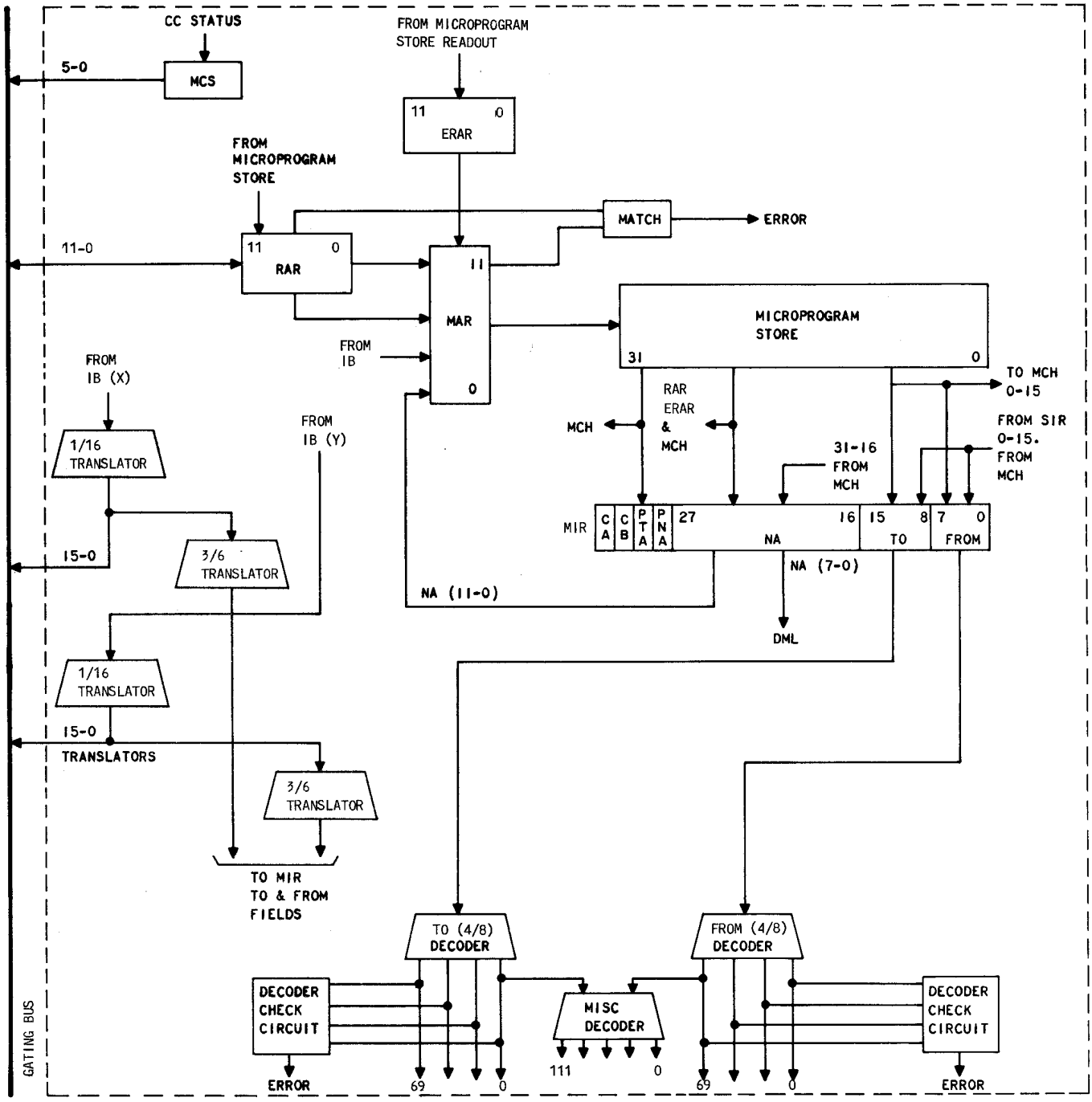


Fig. 25—Microprogram Control Block Diagram

5.27 ***Microinstruction Register (MIR):*** The MIR is a 32-bit register used to buffer microinstructions which are gated out of the microstore. The MIR contains a FROM field (bits 0 through 7), a TO field (bits 8 through 15), a NEXT ADDRESS (NA) field (bits 16 through 27), two parity bits (bits 28 and 29), and a two bit control field (bits 30 and 31).

5.28 The basic instruction set consists of register-to-register gating operations (Fig. 27). The register gating operation is implemented

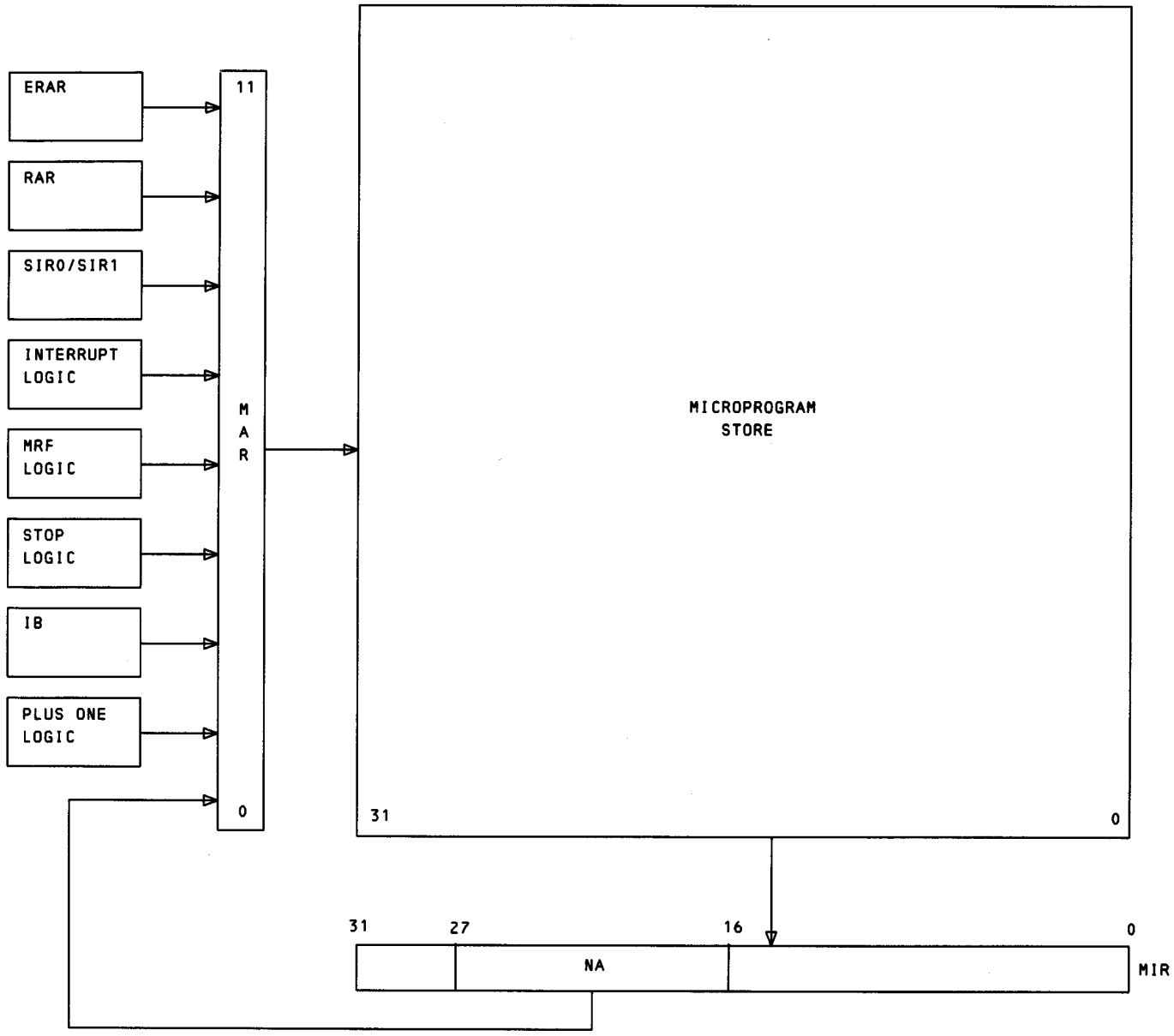


Fig. 26—Loading of Microaddress Register

by the TO and FROM fields while the NA field and parity P_{NA} are loaded in the MAR. The TO and FROM fields are decoded and used to enable a source and destination register. The FROM field enables a source register to gate information onto the data bus. The TO field enables a destination register to receive the data from the bus. Therefore, the control signals from the TO and FROM decoders perform the data transfer operation. After this operation, the address in the MAR is decoded and the next microinstruction is accessed and then loaded into the MIR.

5.29 The X and Y fields of the instruction buffer (IB) which is part of the main memory control (see 5.51) are used in data transfer operations involving the general registers and special registers. Normally, the X and Y fields correspond to the TO and FROM fields, respectively, (Fig. 28). The outputs of the X and Y fields are attached to 3-out-of-6 translators. The outputs of the translators are conditionally gated to the low six bits of the TO and FROM fields in the MIR. This conditional gating is enabled when all zeros are detected coming out of microstore in the same 6-bit positions.

Along with the all zeros, a 1-out-of-2 code is provided from microstore for the two high bits of each field. This code is used to select either the general register set or the one of 26 special register set in either the TO or the FROM field. The results gated into the MIR are the logical OR of the output from the 3-out-of-6 translators and the output of the microstore. This enables any one of the 16 general register or any one of the 16 special register to be either a source or a destination register. Note that not all special registers are accessed in this manner.

5.30 Occasionally, the roles of the X and Y fields are reversed so that the contents of various general registers may be exchanged or swapped (Fig. 29). As in the normal translation, the outputs of the translators are conditionally gated to the low six bits of the TO and FROM fields in the MIR. This conditional gating is enabled when all zeros are detected coming out of microstore in the same 6-bit positions. However, in this case, the code (11) from microstore for the two high bits of each field indicates a swap translation. Therefore, the results gated into the FROM field of the MIR are the logical OR of the output of the X field translator and the output of microstore bits 0 through 7. The results gated into the TO field of the MIR are the logical OR of the output of the Y field translator and the output of microstore bits 8 through 15. In each field, the high bit (bit 7 of the FROM field and bit 15 of the TO field) is cleared to meet the 4-out-of-8 code check.

5.31 The NA field is primarily used for the address of the next microinstruction in the microsequence. However, the upper four bits of the NA field can be decoded to give the MIR an additional control field. The use of the NA field for control is specified by the state of the two control bits of MIR (CA and CB). In this case the next address is formed by "oring" a one into bit zero of the MAR. A one is also "ored" into bit zero of the MAR for data operations when the NA field contains data to be gated to some destination register other than the MAR.

5.32 The binary codes and states of the two control bits (CA and CB) are shown in Table A. Two parity check bits (P_{NA} and P_{TA}) are used to check the address sequencing of the microstore. P_{NA} is the parity bit for the next address in the microstore and P_{TA} is the parity bit for the current address. The PTA and the two control bits (CA and CB) are matched with the parity bit of the MAR P_A to ensure that the correct microstore address was accessed. The relationship is expressed as:

$$P_A = CA \oplus CB \oplus P_{TA}$$

where $\oplus$ defines the exclusive or function. This means if the CA, CB bits are odd (0, 1 or 1, 0), the P_A and P_{TA} should mismatch and for CA, CB even (0,0 or 1,1), they should match. The parity for next address (P_{NA}) becomes the P_A for the next word access if it is a normal sequence, otherwise the PA is given the relationship as shown in Fig. 30.

5.33 ***Return Address Register (RAR):*** The RAR contains one parity bit and 12 bits of data used to store the return address in a microsubroutine. The RAR is usually matched with the contents of the MAR for error detection. Microsubroutine operation is covered in detail in 5.41.

5.34 ***Error Return Address Register (ERAR):*** The ERAR is provided for storing the microsubroutine return address when the complement correction microsubroutine must be invoked. The complement correction microsubroutine is invoked when bad parity is detected on a store read and the system is not in the update mode (see 7.15 through 7.17). The ERAR is required because the microcontrol may already be in a microsubroutine when the parity error is detected. The next address field coming from the microstore is gated to the ERAR during the complement correction subroutine. At the completion of the complement correction subroutine the contents of the ERAR are gated to the MAR.

5.35 ***Decoders and Translators:*** The TO and FROM fields of the MIR are decoded to generate the control signals required to implement the microinstruction. The outputs of the TO and FROM field drive a TO field (4-out-of-8) decoder which selects the source register for the gating operation. The information is normally gated from the source register onto the gating bus. The outputs of the TO and FROM field also drive a FROM field (4-out-of-8) decoder which selects the destination register for the information which was gated onto the gating bus.

5.36 The miscellaneous decoder is used to set and clear various flip-flops within the 3A

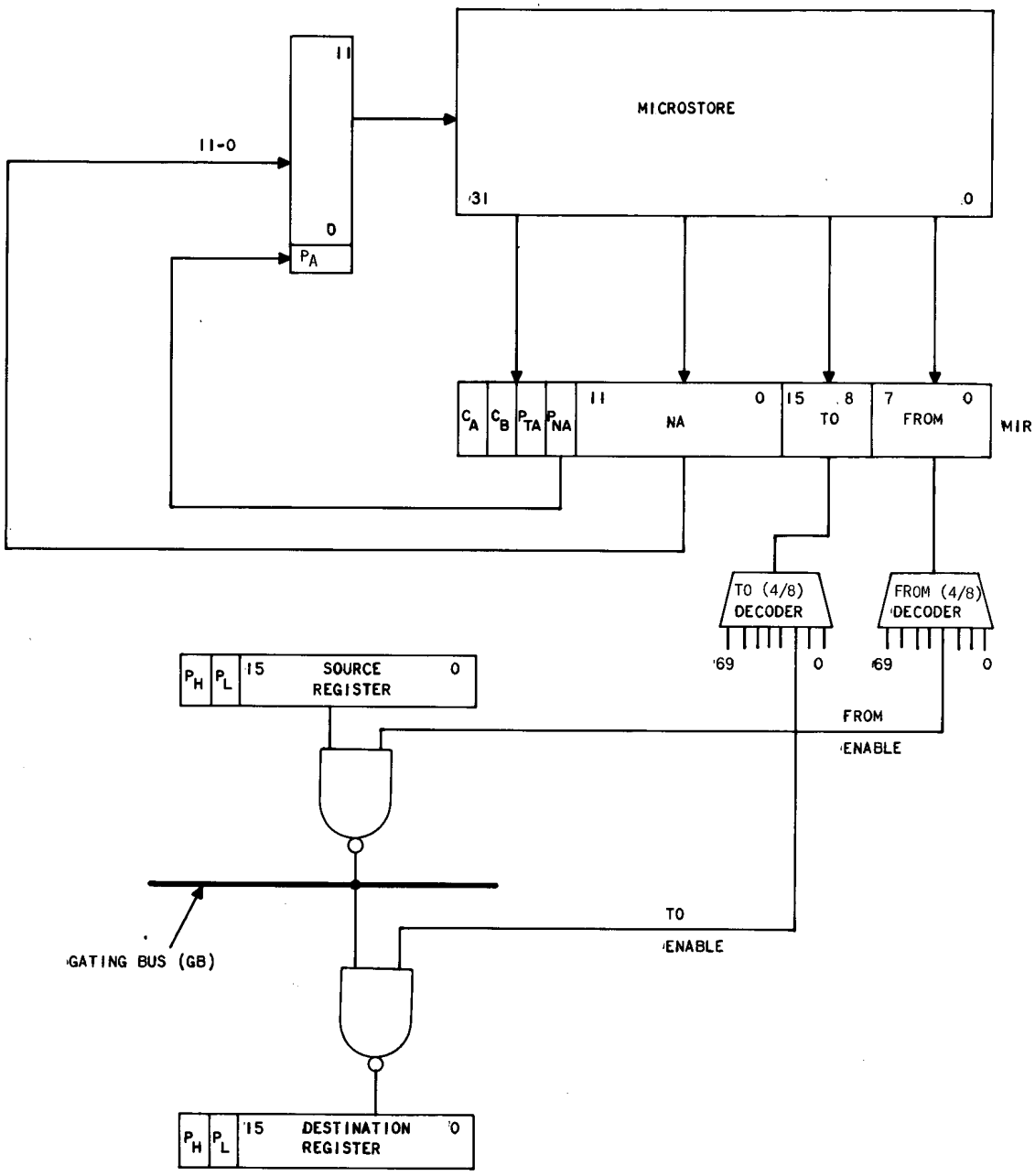


Fig. 27—Normal Microinstruction Execution

CC and to enable dedicated gating paths. The miscellaneous decoder is activated by 14 TO field outputs and 8 FROM field outputs. One output from each field will enable one of the possible 112 miscellaneous decoder outputs.

5.37 The TO and FROM decoder outputs are checked by use of an encoder and a self-checking m-out-of-n code check circuit. This checking is accomplished by encoding the outputs into the code complement of the original m-out-of-n code and then inputting the complement into the m-out-of-n checker. This m-out-of-n checker detects all errors of the decoder, and since it is self-checking, it also detects most errors created within the checker itself.

5.38 ***Microcontrol Status (MCS) Register:*** The MCS is a 20-bit register used to control certain functions, such as conditional transfers,

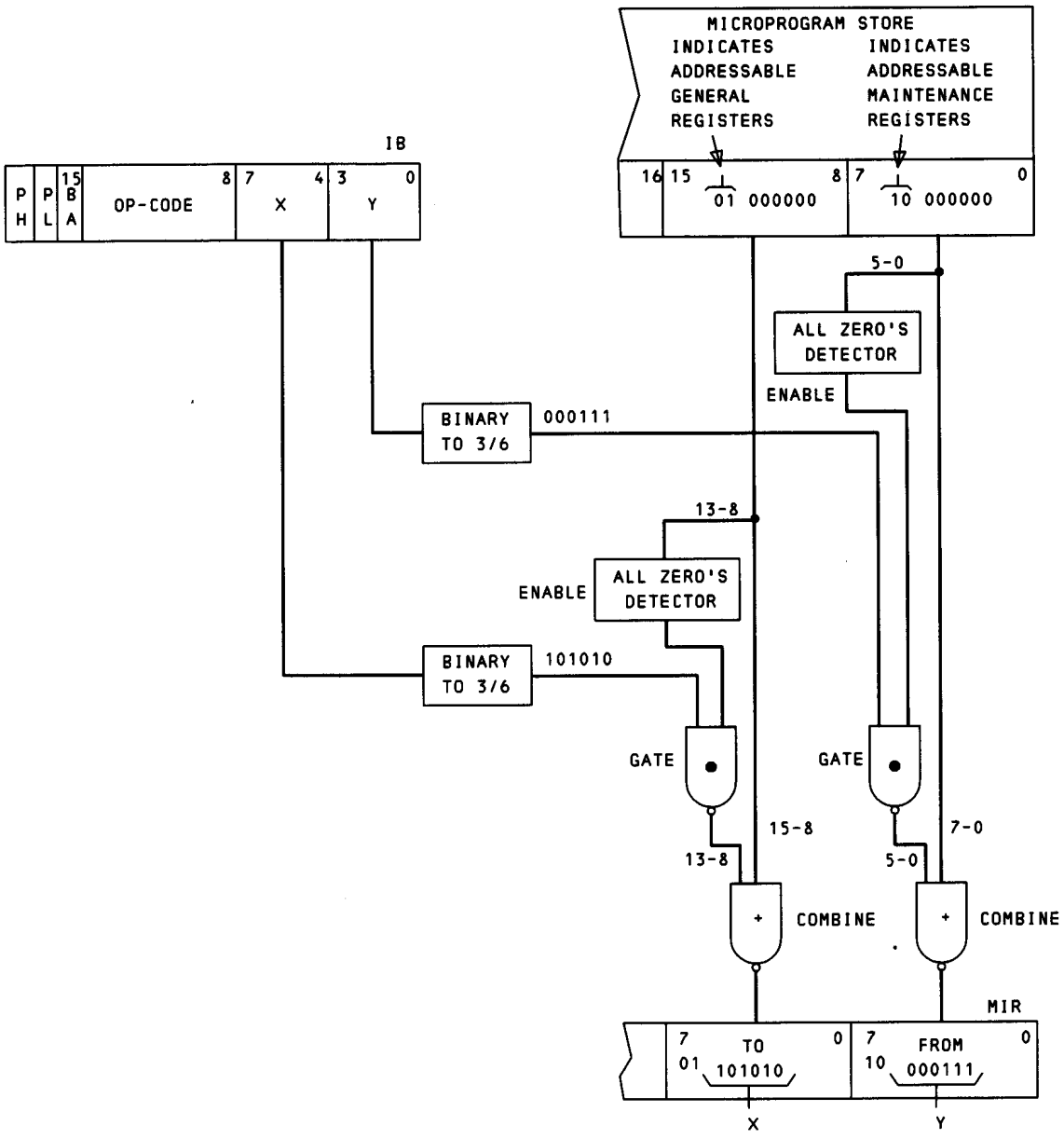


Fig. 28—Normal Operand Translation

operation of the RAR, selection of SIR0 or SIR1, and selecting which OP code set is to be executed (3A CC or 2B). The bits of the MCS are duplicated for error detection purposes. The duplicated bits are used in such a way so that a stuck at zero or one failure of either bit will be detected (i.e., the DS bits are duplicated and used for setting the MAR low order bit and complementing the MAR parity bit so that a stuck at zero or stuck at one failure of either DS bit will result in bad parity of the MAR). Table B contains the bit designations and functions fo the MCS register.

5.39 The OP code Fil (OPF) bit is necessary to select the 2B OP code set. The 2B OP code set is executed when OPF=1 and the 3A CC OP code set is executed when the OPF=0. There are 128 possible instructions for the 3A CC set and 128 for the 2B set. Each time a sequence of instructions being read out of main store changes instruction sets, an instruction must be executed to change the state of the OPF bit. Each instruction set is provided with an instruction to zero the OPF bit (ZOP) and an instruction to set the OPF bit (SOP). The microsequences for these instructions fire

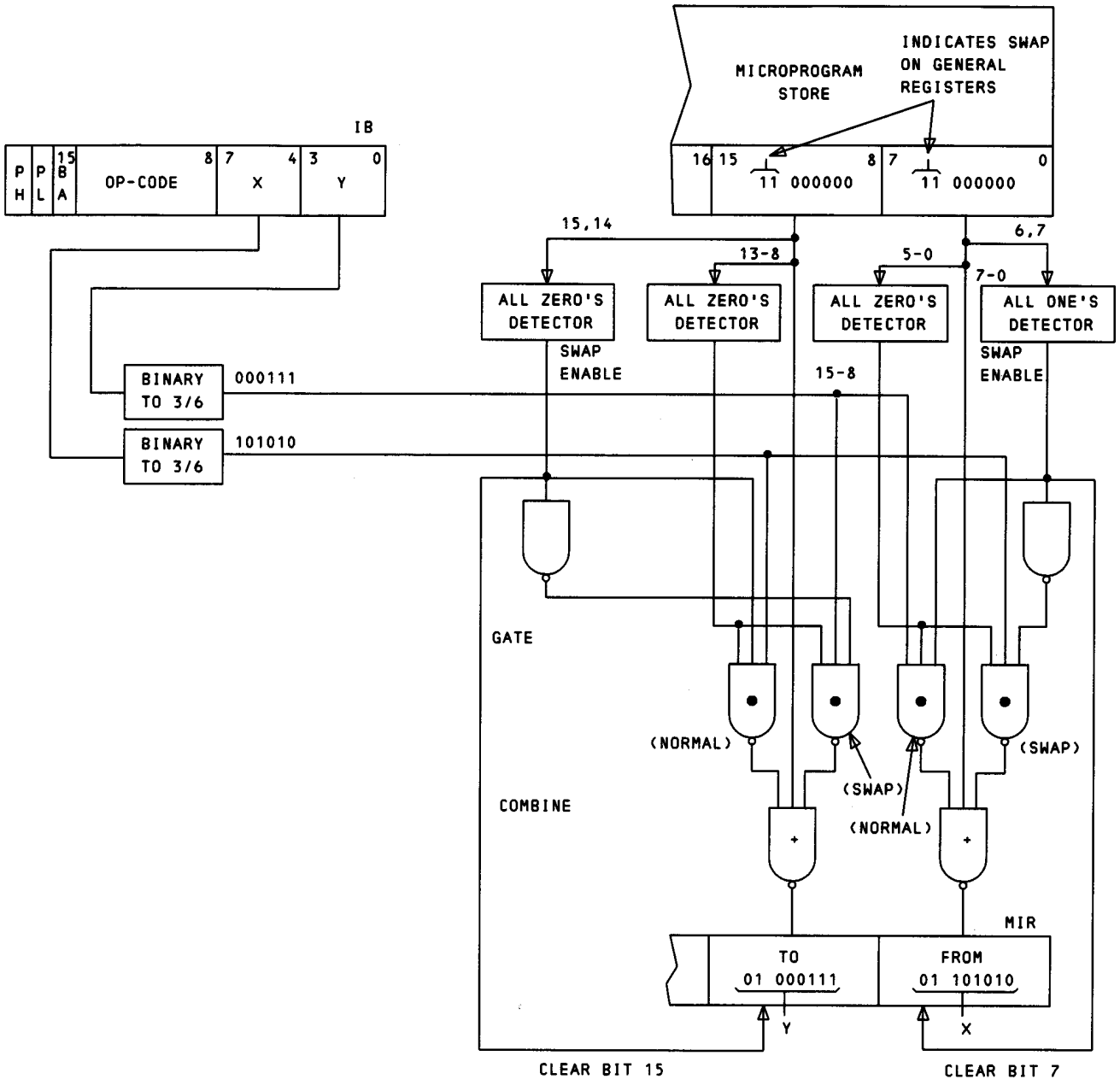


Fig. 29—Swap Operand Translation

separate miscellaneous decoder crosspoints to change the state of the OPF bit. The instruction will set the OPF bit if the next instruction to be loaded into the MAR is a 2B OP code. This jams MAR bit 11 with a one so that bits 11-7 equal 10000 to complete the 2B OP code address. This provides the 2B decoding information to start at address 4000 octal. If the next instruction is a 3A OP code, the OPF bit is cleared and MAR bit 8 is jamed to a one to complete the 3A OP code address. This provides the 3A decoding information to start at address 400 octal.

Common Microprogram Control Operations

5.40 Microsubroutine operation, conditional transfers and indexing are basic operations of the microprogram control which are common to many microsequences. Even though these are basic operations of the microprogram control, they are

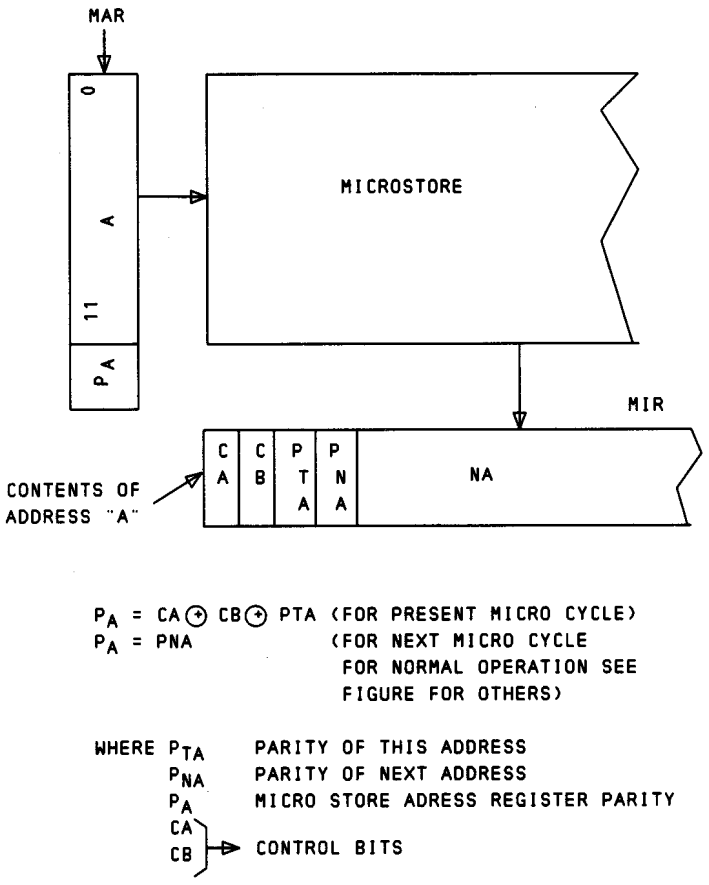


Fig. 30—Address and Control Parity Checking

discussed separately to simplify the general discussion of the microprogram control.

5.41 ***Microsubroutine Operation:*** The microsubroutine operation is used to save space in the microstore. This operation is used wherever a common set of microinstructions is performed. The microsubroutine is invoked by a data command. The return address is read from microstore into the MIR and RAR in the normal manner. However, the gating from MIR to MAR is blocked during this microcycle (C_A=1) and the MAR is incremented by one. In order to keep the RAR contents from being destroyed on the following microcycles, a microinstruction clears the RAR update (RU) bit. The contents of the RAR are thus saved, and the microinstruction sequencing continues until the return address is needed. The RAR is in a nonmatching mode with the MAR. To activate the microsubroutine return, a location is microstore that has an all ones NA field, is read into the MIR (Fig. 31). An all ones detector is activated which enables the gating of the RAR to the MAR. The all ones detector also sets the RU flip-flop which returns the MAR and RAR to their normal matching mode.

5.42 ***Conditional Transfers:*** The microcontrol status (MCS) register is provided to perform conditional transfers in the microprogram control sequencing logic as a function of various states of the 3A CC (Fig. 32). The results of functions performed in the data manipulation logic, in the I/O logic, in the main store control logic, etc., are gated to the MCS register with the use of various microinstructions. Six of the MCS register bits can be individually tested or conditionally transferred upon. These six bits and their primary functions are:

DS - Stores results of DML operations (i.e., adder overflows)

TABLE B

FUNCTION OF MICROCONTROL STATUS REGISTER BITS

BIT	DESIGNATION	FUNCTION
0	Condition Flip-Flop (CF)	Used as branch condition for microprogram control. Duplicate bits are used for setting the MAR low order bit and complementing the parity bit so that a stuck-at-zero (S-A-0) or S-A-1 failure of either duplicate bit will result in bad parity on the MAR.
1	CF (duplicate)	
2	Data Manipulation Logic Status (DS)	
3	DS (duplicate)	
4	Test Register 1 (TR1)	
5	TR1 (duplicate)	
6	Test Register 2 (TR2)	
7	TR2 (duplicate)	
8	Data Ready (DR)	Used to indicate that the last main memory operation has been completed. DR bit is set by a combination of store completion signals from both stores and reset by any signal which initiates a main memory operation. One copy controls gating of the new operation code into MAR and other controls same gating into RAR. Any failure of one of the copies of the DR bit will be detected by MAR-RAR mismatch.
9	DR (duplicate)	
10	RAR Update (RU)	Controls function of RAR. When RU=1, RAR is used as a duplicate of the MAR to check gating into it from MIR and micro-store. When RU=0, RAR is used to save a return address which will subsequently be branched to, e.g., return from a sub-routine. RU is set by same control signal that gates out of RAR and reset by a miscellaneous decoder crosspoint.
11	RU (duplicate)	
12	I Flip-Flop (IFF)	Determines from which SIR the new OP code is derived (SIR0 for the low 16-bits of memory and SIR1 for the upper bits of memory field).
13	IFF (duplicate)	

TABLE B (Cont)

FUNCTION OF MICROCONTROL STATUS REGISTER BITS

BIT	DESIGNATION	FUNCTION
14	OP Code Fil (OFF)	Used when expanding the number of operation codes from 128 to 256. If OPF bit=0, then the 7 bit OP field maps into a block of 128 OP codes starting at micro-store address 256. If OPF bit=1, then the OP code field maps into a block of an additional 128 OP codes starting at 2048.
15	OPF (duplicate)	
16	MARP	Used for internal check function. In general,reflects the parity of the MAR. MARP is not duplicated but appears twice to preserve MCS register parity.
17	MARP	
18	Error Return Address Update (ERU)	Controls function of ERAR. When ERU=0, the NA field coming from micro-store is gated to ERAR. ERU is not duplicated but appears twice to preserve MCS register parity.
19	ERU	

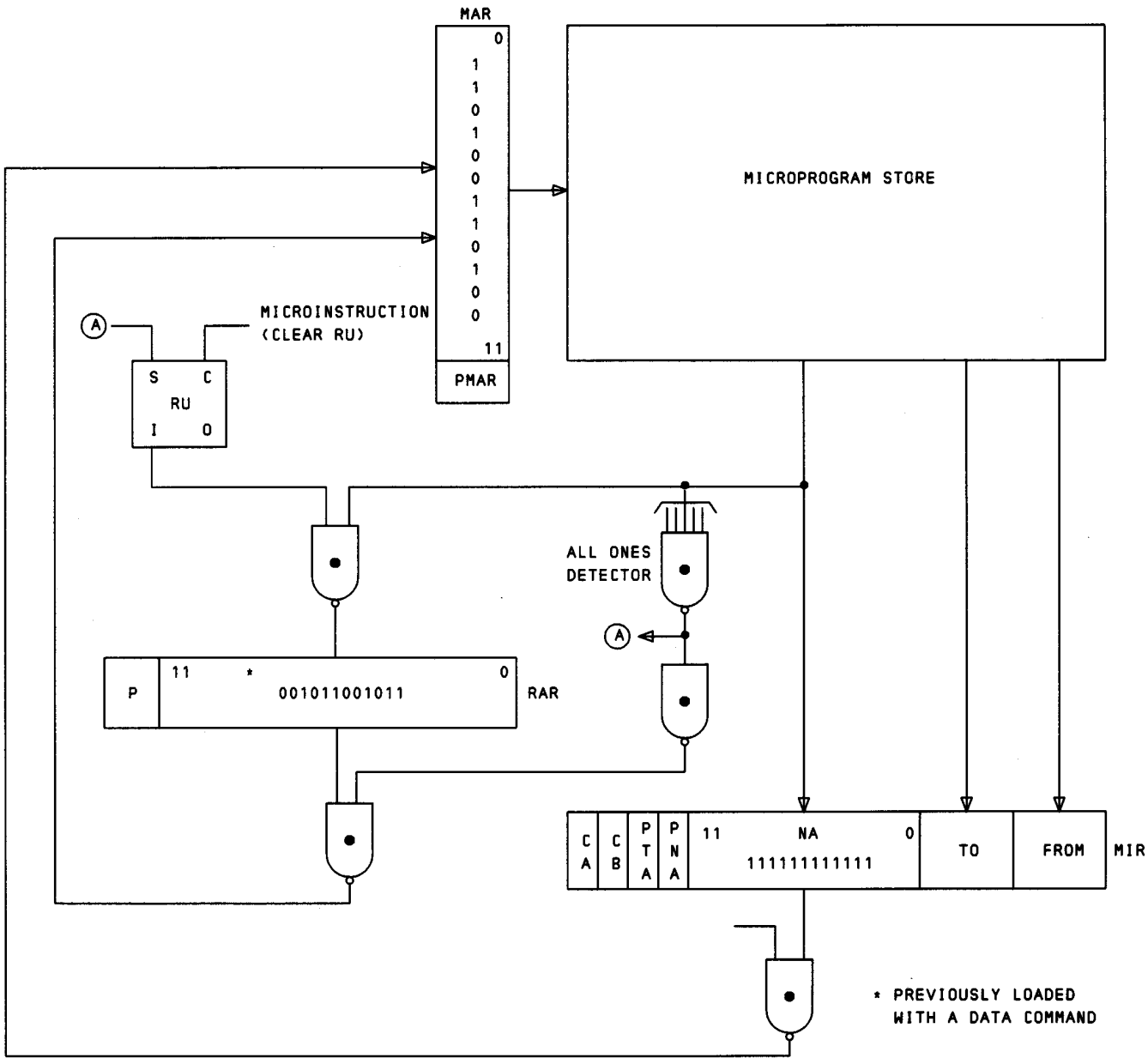


Fig. 31—Microsubroutine Return

DR - Indicates completion of main store cycles

TR1 - General purpose status bit intended for use by microprogram control for indicating past states

TR2 - Same as TR1

CF - Used to pass status information to main program sequence

I - Used to indicate which half word No. 2B ESS instruction is being processed.

5.43 On a conditional transfer instruction, a microinstruction (miscellaneous crosspoints) is used to select which of the MCS bits is to be tested. The state of this bit is then gated into the MAR bit 0 to achieve the transfer. Conditional transfers occur on even word boundaries in the microstore. Therefore, if the MCS bit is 0, the next microinstruction is located at the address read

from the NA field. If the MCS bit is 1, the next microinstruction is located at the address read from the NA field plus one.

5.44 The conditional transfer logic is checked by duplicating the MCS bits. The 0 copy of the MCS bits is gated to the low bit of the MAR. If a transfer is effected, the other copy, MCS 1, causes the parity bit of the MAR to be complemented. A failure in the conditional transfer logic will result in a parity error.

5.45 ***Indexing:*** Index commands are used to access tables contained within the microstore. Indexing (Fig. 33) is initiated by a microinstruction which ORs a 4-bit field (contained in the operand X or Y of the IB register) with the present address (NA) contained in the MIR. The starting address of the index tables must contain all zeros in the low four bits.

5.46 The indexing is checked by ORing the appropriate X or Y field into both the RAR and MAR. Since the starting address must fall on even word boundaries, the address parity bit P_A is the complement of P_{NA} when the parity of the X or Y field is odd. As a result, the same checking hardware used for normal microinstruction flow detects failures in the loading of the index field or the improper access of an index address.

5.47 ***MAR-RAR Matching:*** Normal instruction flow results in the MAR being loaded with the contents of the NA field from the previous microinstruction. This NA field is also gated to the RAR (except during subroutines) as well as the MAR. The loading of the duplicate information into the RAR allows the MAR and RAR contents (only bits 1-11; bit 0 is checked by the MAR parity function, see 5.38) to be matched on every microcycle. The match also checks the input and output circuitry associated with the loading of the MIR(NA). In addition the match checks all gating paths involved in loading the MAR and RAR, including the loading of the OP code (Fig. 34). At times the MAR and RAR knowingly mismatch (e.g., during subroutines). The matcher is inhibited at these times.

D. Main Memory Control

5.48 The main memory control provides the interface between the 3A CC and main store (MAS) bus for accessing data from main store and for storing data in MAS. The 3A CC is designed to use a direct-coupled bus in an asynchronous mode. The address portion of the bus is unidirectional while the data portion is bidirectional. The state of the read/write (R/W) flip-flop determines the direction of the data portion of the bus.

5.49 The main memory control (Fig. 35) is made up of a control and sequencing portion and register interface portion. The registers are the main memory status (MMS) register, program address (PA) register, store address register (SAR), store instruction registers (SIR0 and SIR1), store instruction buffer (SIB), store data registers (SDR0 and SDR1), instruction buffer (IB) and the necessary control circuitry.

5.50 The main memory control performs the following functions:

(a) Buffers the address, data, and control signals to be issued to the MAS bus

(b) Determines the state of the MAS bus after receiving a memory request from the microprogram control

(c) Activates a main store cycle if MAS bus is not busy

(d) On instruction fetches, gates the SAR contents to the PA register so that the PA+1 logic can compute the next store address

(e) Monitors the bus for a completion signal from MAS

(f) Buffers the instruction, data, and control signals from MAS bus

(g) On a read operation, gates the contents of the MAS bus to the SIRs or SDRs

(h) Sets the data ready (DR) bit to indicate the end of a memory cycle.

5.51 The following provides a description of the main memory control registers.

(a) ***Main memory status (MMS) register***—16-bit register used to store the present state of the main memory control and to formulate commands sent to main store. The bit designations and functions are shown in Table C.

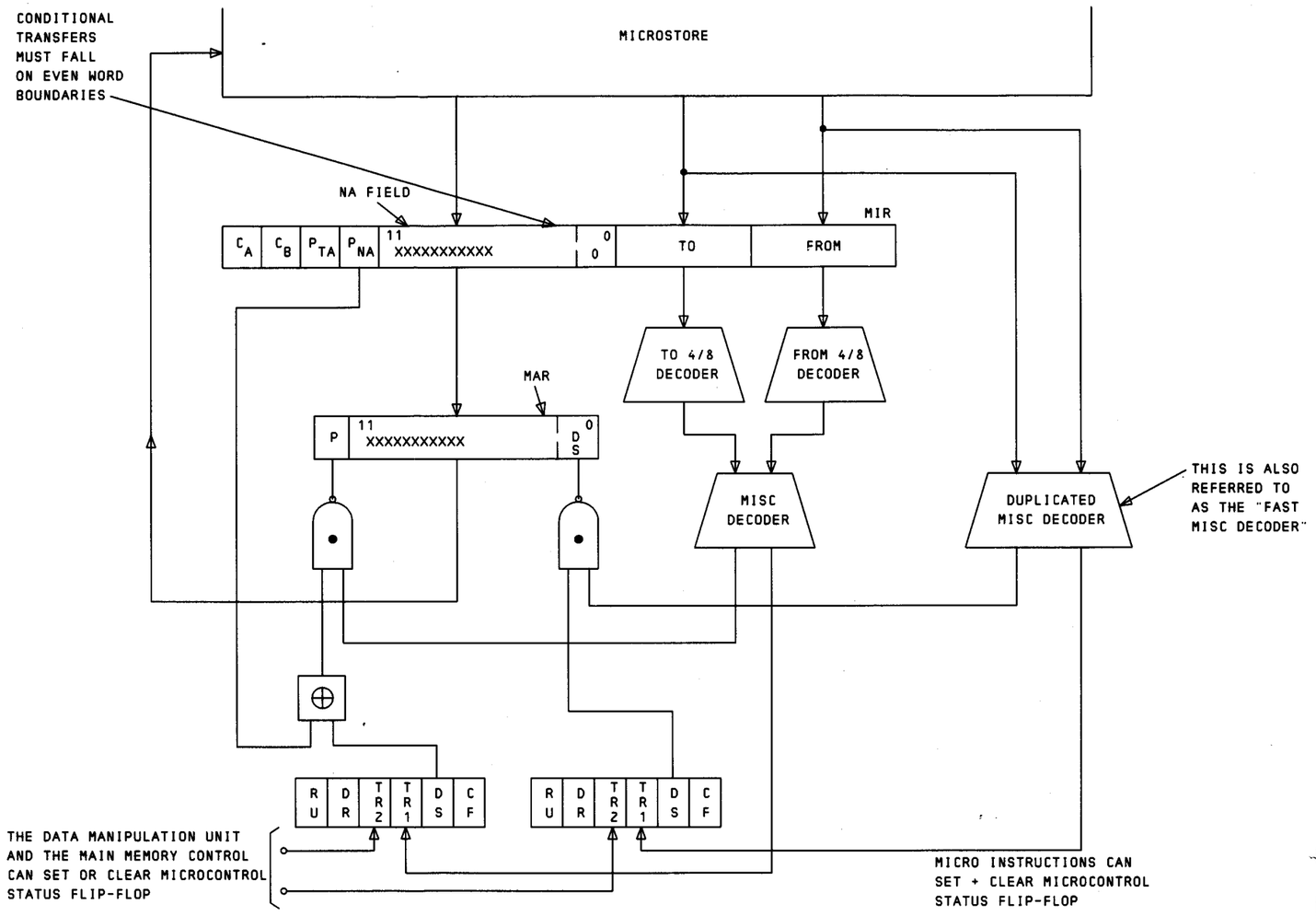


Fig. 32—Conditional Transfers

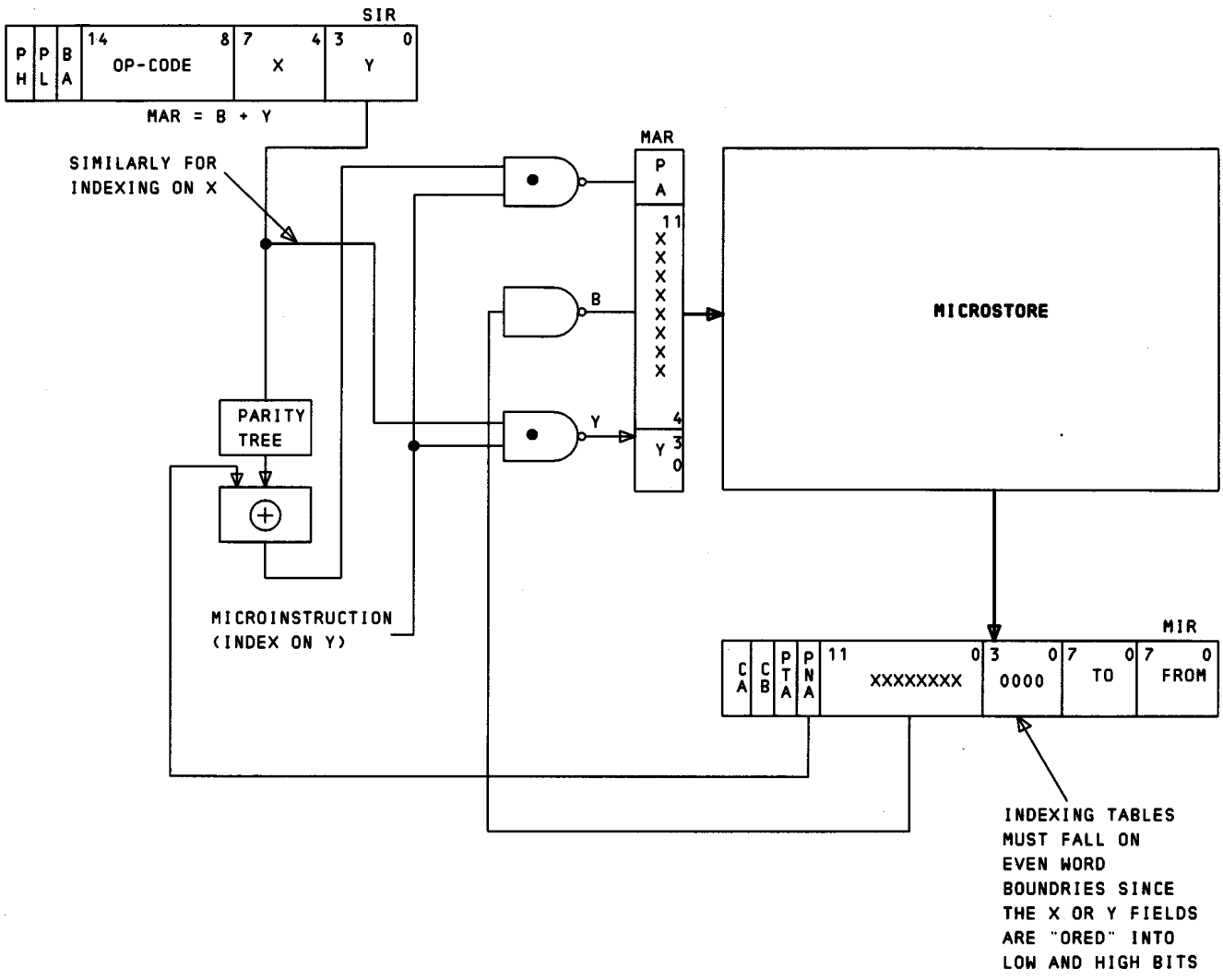


Fig. 33—Microstore Indexing

(b) ***Program address (PA) register*** — 20-bit register used to store the last program address which was accessed from main store.

(c) ***Store address register (SAR)*** — 20-bit register which stores the address of the memory location which is to be fetched from memory.

(d) ***Store instruction registers (SIR0 and SIR1)*** — 16-bit registers used to buffer program instructions. SIR0 and SIR1 are both required for buffering 24- bit 2B instructions while only SIR0 is required for 16- bit 3A instructions.

(e) ***Instruction Buffer (IB):*** The IB is a 16-bit register used to temporarily store the present OP code and operand fields obtained from main store so that the next instruction can be accessed (put in SIR0 and SIR1) concurrently with the execution of the present one (in the IB). The low eight bits of the IB connect to translators which may be used in conjunction with the microcode to control gating, and to define various options or data fields.

(f) ***Store instruction buffer (SIB)***—16-bit register used for buffering the second half of a 2B instruction before it is loaded into SIR1.

(g) ***Store data registers (SDR0 and SDR1)*** — 16-bit registers used to buffer

data. SDR0 and SDR1 are both required for buffering program and translation data while only SDR0 is required for buffering call store data.

Read Operation

5.52 ***Instruction Fetch:*** The microcode determines when a new instruction is to be fetched from main store (Fig. 36). The fetch is initiated by the control bits (CA and CB) of the microinstruction register (MIR) which is located in the microprogram control. The enable instruction fetch signal from the binary decoder loads the SAR with the instruction address by gating the contents of the PA+1 logic into the SAR. The request (REQ) and instruction or data (ID) flip-flops are set while the data ready (DR) flip-flop (located in MCS register) is cleared. The REQ flip-flop buffers the store request in case the bus is busy. Setting the ID flip-flop gates the instruction from MAS to the store instruction register(s).

5.53 If the MAS bus is idle, the seize flip-flop is set to inhibit the other 3A CC from accessing the bus. The command, address, and go signals are gated onto the MAS bus. These signals are maintained until the 3A CC receives the store complete signal and the instruction is buffered in the store instruction register(s). The store complete signal sets the DR flip-flop indicating to the microprogram control that the instruction fetch is completed. A single register (SIR0) is required for buffering 3A instructions in the main memory control, while three registers (SIR0, SIB, and SIR1) are required for buffering the No. 2 and 2B instructions (Fig. 37).

5.54 When the DR flip-flop is set the contents of SIR0 are loaded into the instruction buffer (IB) and the contents of SIB are loaded into SIR1. The purpose of the SIB is to improve the efficiency of MAS operations when using the extended MAS bus interface for emulation of the No. 2 ESS half-word command structure. For example, when an instruction is loaded into the SIR0 and the SIB, the first half of a double-word instruction is executed and this first half comes from the SIR0. By gating the SIB to SIR1 when this first half-word instruction is initiated the main memory control is free to issue the next MAS request for an instruction read since both the SIR0 and the SIB are available.

5.55 A 3A double word instruction requires an additional store fetch during the microsequence to obtain the address or data portion of the instruction. This half of the instruction is also read into SIR0. The address or data is gated from SIR0 to the proper registers for execution of the instruction.

5.56 The last instruction in a microsequence contains an all zeros NA field. This places all zeros in the NA field of the MIR. The microprogram control will loop on the all zeros location until the next instruction is fetched from main store.

5.57 If a No. 2B full word instruction is being executed, the microsequence will control the gating of the address or data portion of the instruction which is stored in SIR1. The address or data will be gated to the appropriate registers by the microsequence to complete the execution of the instruction.

5.58 At the end of the execution of the first half of a No. 2B half word instruction, the microcode transfers to the all zeros location. The contents of SIR1 are immediately gated to the IB and the OP code to the MAR and RAR. The second half word instruction which was buffered in SIR1 will be executed by the appropriate microsequence. During the execution of the instruction which was stored in SIR1, main store is usually addressed. When the microsequence for executing SIR1 is completed, an all zeros NA is reachd again. The microprogram control will loop on the all zeros address until the next instruction fetch is completed.

5.59 The all zeros loop is necessary since at the end of a short microsequence the microprogram control must wait for the next instruction to be fetched from main store (Fig. 38). Whenever the all zeros location is read out of the microstore into the NA field, the interrupt lead is checked (except between half-word instructions). If an interrupt is present, whether a main store fetch is completed or not, the starting address of a microsequence to service the interrupt is hardwire jammed into the MAR. If an interrupt is not present, the microprogram control loops on the all zeros location until the next instruction from main store is available and is loaded into the MAR by the load new OP code signal (LNOP). An output from the all zeros detector and an indication that the memory cycle

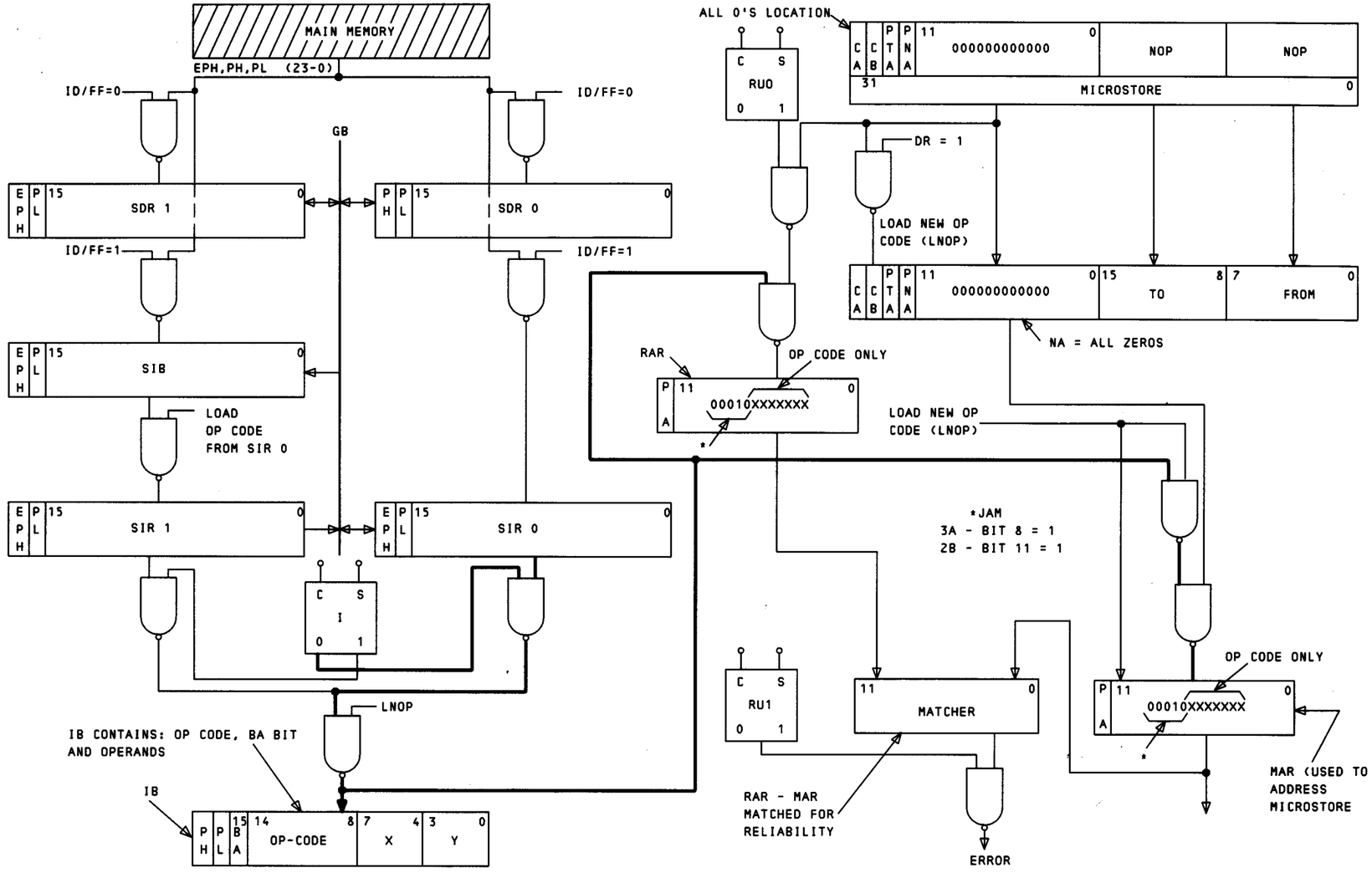


Fig. 34—Microaddress Register—Return Address Register Matching

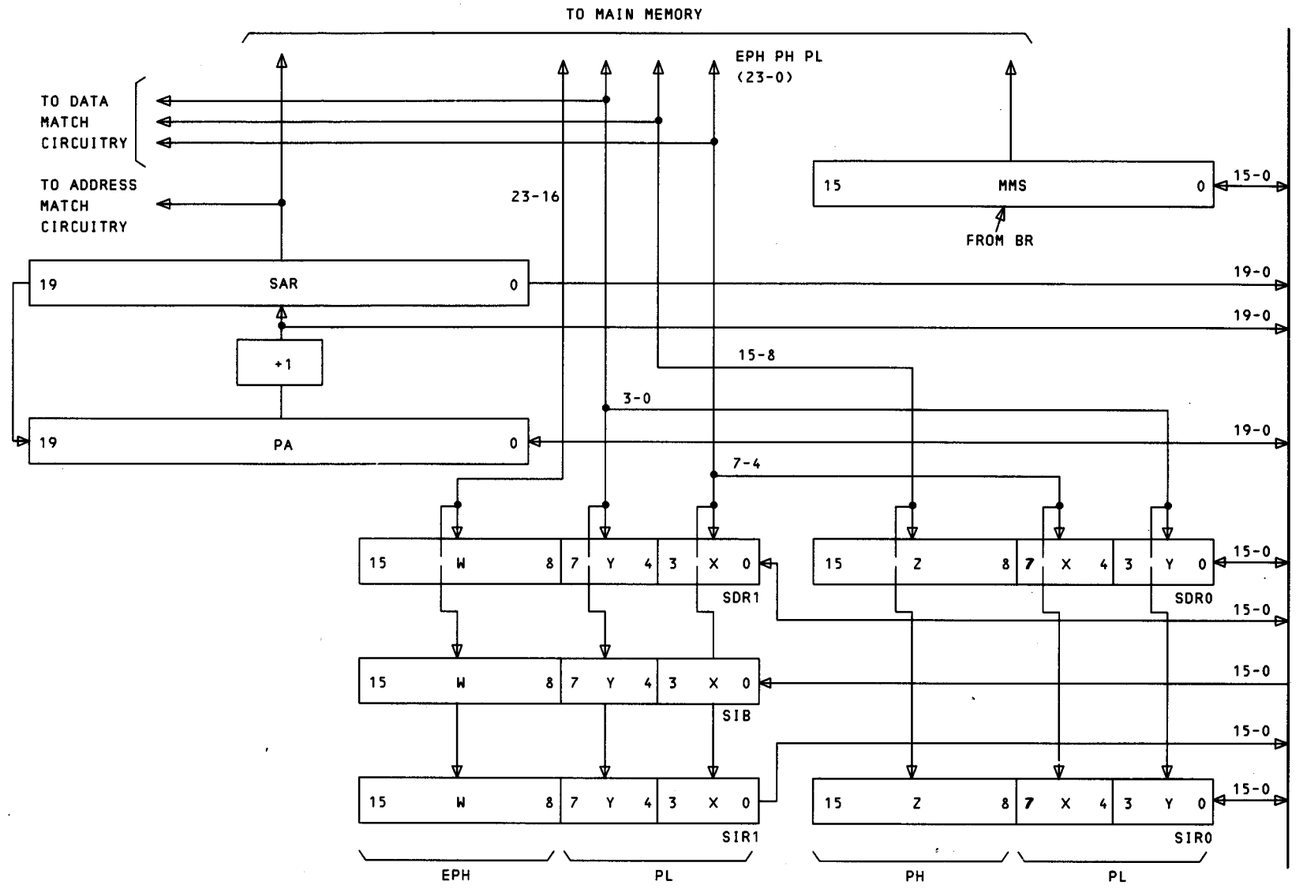


Fig. 35—Main Memory Control

TABLE C

DESIGNATIONS AND FUNCTIONS OF MAIN MEMORY STATUS REGISTER BITS

BIT	DESIGNATION	FUNCTION
0	Memory Maintenance (MM) 1	Used with MM2 and RW to formulate the command sent in a memory operation
1	MM1	Same as bit 0
2	MM2	Used with MM1 and RW to formulate the command sent in a memory operation
3	MM2	Same as bit 2
4	Read or Write (RW)	Indicates whether memory is to perform a read(1) or write(0) operation
5	RW	Same as bit 4
6	Idle (IDL)	If set, disables all communications to the other 3A CCs memory
7	IDL	Complement of bit 6. If set, disables all communications from other 3A CCs memory
8	Update (UPD)	Indicates whether or not to update the off-line memory
9	UPD	Same as bit 8
10	Isolate (ISO)	Prevents the other 3A CC from accessing this 3A CCs memory
11	ISO	Same as bit 10
12	Block Double Store Read (BDSR)	Inhibits double store read
13	BDSR	Same as bit 12
14	Complement Write (CW)	Activates complement write lead of main store bus. Main store controller (MASC) will have last word, write complement of last word, and store in the complement in the last read address
15	Block Error Recovery (BEC)	Inhibits all error recovery procedures within the 3A CC associated with incorrect store read data.

is complete (DR=1) results in a new instruction being loaded.

5.60 ***Data Fetch:*** When an instruction requires data to be fetched from memory, the microprogram loads the data address into the SAR. The data address may be loaded from another register or calculated within the data manipulation logic (DML). Once the data address is loaded in the SAR, a miscellaneous crosspoint initiates the store request by setting the REQ flip-flop. The ID and DR flip-flops are cleared. The ID flip-flop is cleared (see Fig. 36) to indicate that the information from memory will be gated into the SDR0 and SDR1 and to inhibit the SAR to PA gating. SDR0 and SDR1 are both required for buffering program and translation data while only SDR0 is required for buffering data pertaining to calls in progress.

5.61 When the memory bus becomes idle, an ISO flip-flop is set to inhibit the other 3A CC from accessing the bus. The command, address, and go signals are gated onto the bus. These signals are maintained until the 3A CC has buffered the memory response in SDR0 and SDR1 and set the DR flip-flop. The setting of the DR flip-flop indicates to microprogram control when the data fetch is completed.

Write Operation

5.62 The data to be stored in memory is loaded into SDR0 and SDR1 (Fig. 36) and the address is loaded into the SAR. Both SDR0 and SDR1 are required for writing program and translation data while only SDR0 is required for writing call processing data into memory. The loading of the SDR(s) clears the read/write (R/W) flip-flop to indicate a write operation. The REQ flip-flop is set and the DR flip-flop is cleared.

5.62 The update (UPD) flip-flop is set unless the off-line CU is not operational, being diagnosed, etc. The data is written into the on-line and off-line stores to keep the standby store up-to-date in case a switch of the 3A CCs become necessary.

5.64 When the necessary bus becomes idle, the ISO flip-flop is set to inhibit the other 3A CC from accessing the bus. The command, data, address, and go signals are gated onto the bus. These signals are maintained until the 3A CC has buffered the memory response and sets the DR and R/W flip-flops. The setting of the DR flip-flop indicates to microprogram control when the write operation is completed.

E. Gating Bus and Bus Parity Checker

5.65 The gating bus is the major communications path within the 3A CC. Most information is transferred between functional sections of the 3A CC via the gating bus. The gating bus can transmit up to 20 bits of data plus two parity bits during a data transfer operation.

5.66 The gating bus parity checker will fire when the contents of a register are gated onto the gating bus with incorrect parity (unless the checker is inhibited). The checker continually monitors the gating bus on every microcycle. The gating bus contents are stored in a register of the checker, then rippled through a parity tree. If a legitimate parity error is detected, the result of the parity check is gated (during the following microcycle) to the 3A CC error register (ER) bit 3 (bus parity error).

5.67 If bad parity is detected on an incoming message from an I/O channel, the error output is diverted to another bit in the ER. This diversion permits more graceful error recovery strategy then the "stop and switch" of the 3A CC that would otherwise occur if a gating bus parity error results. The diverted error is called an I/O parity error and will cause an error interrupt.

F. Data Manipulation Logic

5.68 The data manipulation logic provides the circuitry necessary to perform such functions as addition, rotation, logical combinations, and find low zero. The DML is duplicated (in each 3A CC) and matched to detect faults. When a mismatch occurs, a response is sent to the error register. A diagnostic uses the C register (which is used for other miscellaneous functions within the 3A CC as well) to localize the trouble. Information is gated into both DMLs (Fig. 39) via the gating bus (BUS x DML). After both DMLs perform the desired function, the results of DML0 are gated back onto the bus (DML0 x BUS) and to the matcher (results of DLM0). The results of DML1 are gated only into the matcher (DLM1 x matcher). The match enable lead is made active at the same time the information is gated onto the bus from DML0 and the information is matched. Parity is

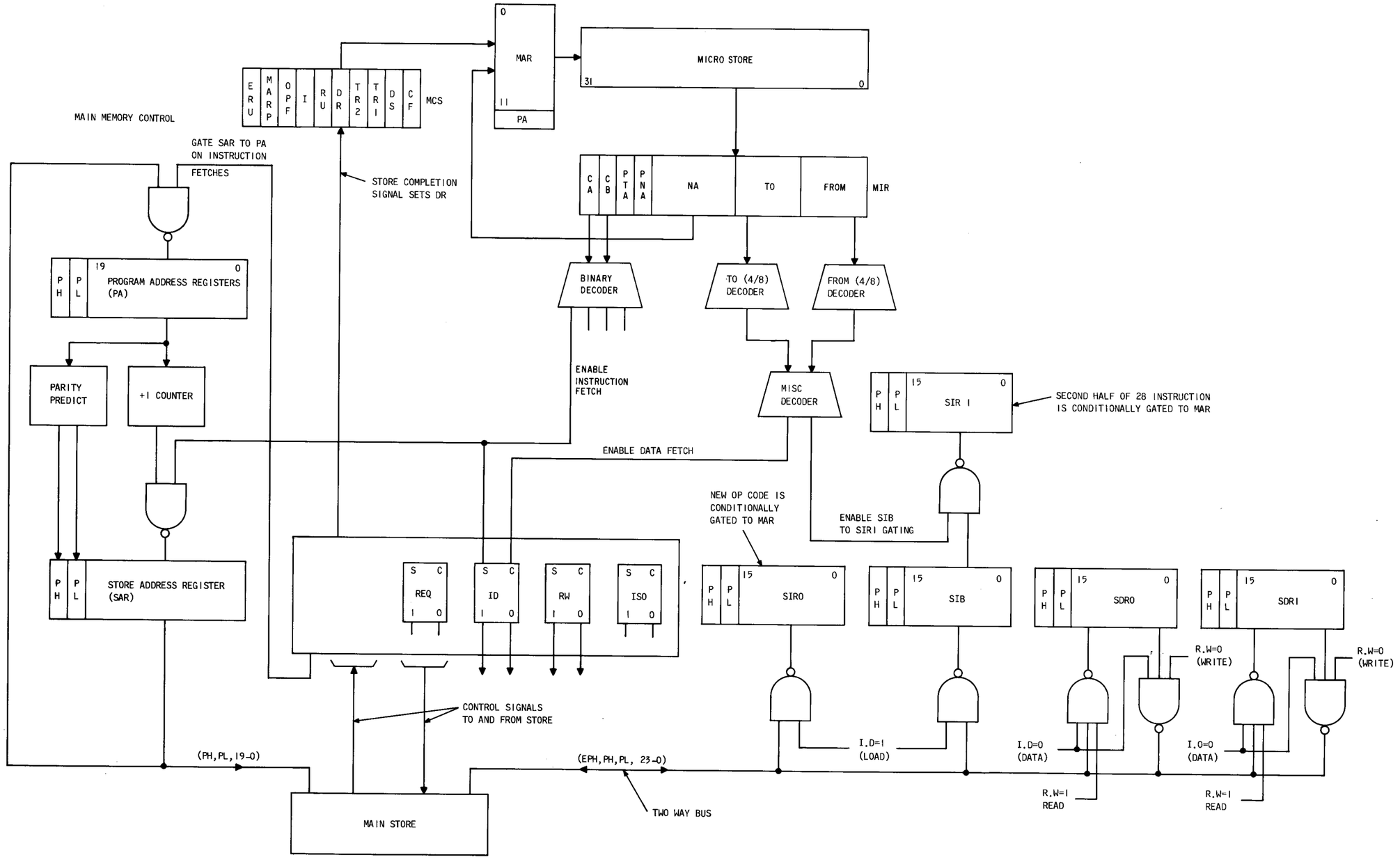


Fig. 36—Main Memory Control—Microprogram Control Interface

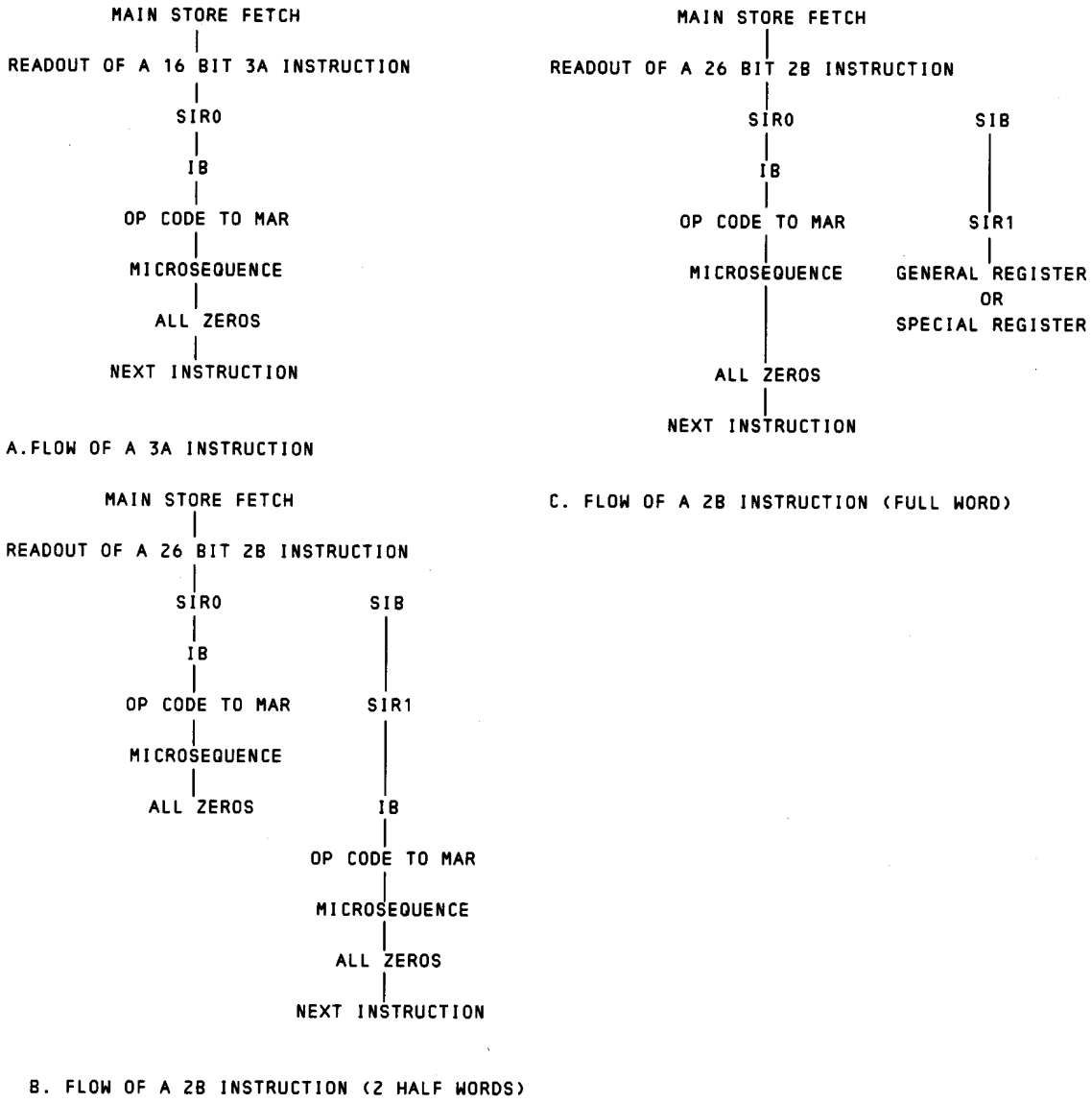


Fig. 37—Flow of 3A and 2B Instructions

generated on all functions by the parity generation circuit in order for the DML output to interface with the rest of the system. The parity is based on DML1 outputs.

5.69 Each DML consists of a function register (FR), two operand registers (AR and BR), and combinational logic circuits (Fig. 40). The FR is eight bits in length and contains the control used by the DML to perform the desired function. The AR and BR are each 22 bits in length. Two bits are parity bits and the remaining 20 bits contain the data to be manipulated. The combinational logic circuits are used to perform the functions on the data in AR and BR.

Loading Operation of DML

5.70 DML operations require the operand and/or function registers to be loaded. This is done under microprogram control (Fig. 40). The NA field is used for loading the function register. This is accomplished by subdividing the NA field into a 4-bit control field and an 8-bit data field. The 4-bit control in the form of a 2-out-of-4 code is decoded to generate the control for gating the

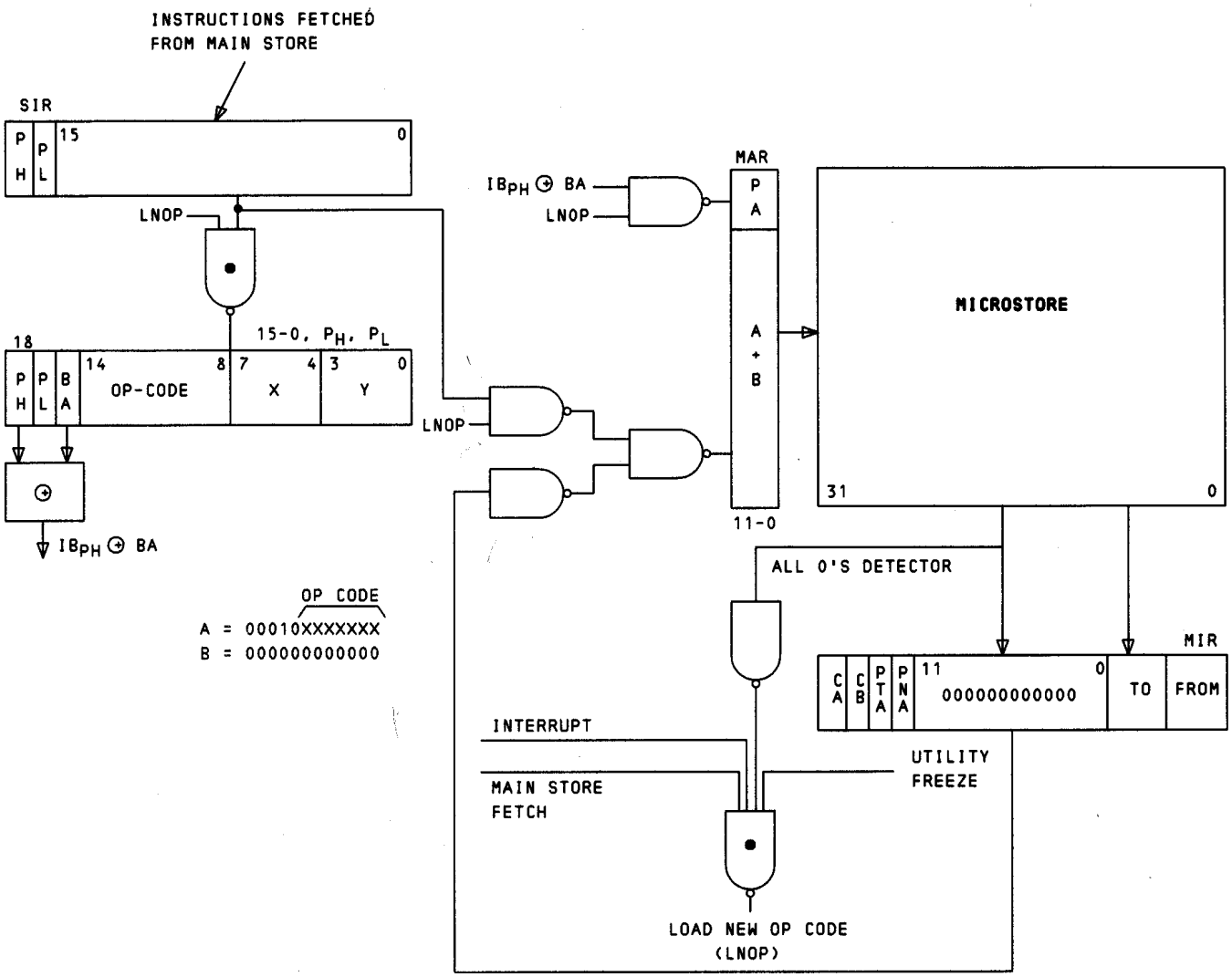


Fig. 38—Load New OP Code Operation

8-bit data field to the function register. Concurrently, the TO and FROM fields are used to load one of the operand registers in the DML. When both operands are loaded into the registers of the DML, the function is performed autonomously to the microprogram control. Since the NA field was used as a control field defined uniquely for this operation by the state of two control bits, C_A and C_B, the address of the next microinstruction is obtained by adding one to the last microstore address. At a maximum of two microcycles later, the results of the DML are gated out to a destination register.

5.71 In the loading of the function register, the gating operation, as well as the information gated out of the microstore, must be checked for correctness. A parity generator attached to the function register in the DML checks the register output against a parity bit read from microstore.

Boolean Functions

5.72 The microprogram control gates the required operands to the AR and BR and loads the FR with the appropriate Boolean (logic) function command. This command enables one of the 16 different Boolean functions to be implemented in the DML. The Boolean function is performed in two microcycles. The first microcycle is used to execute the function, and the second microcycle is used to gate the results to a destination register.

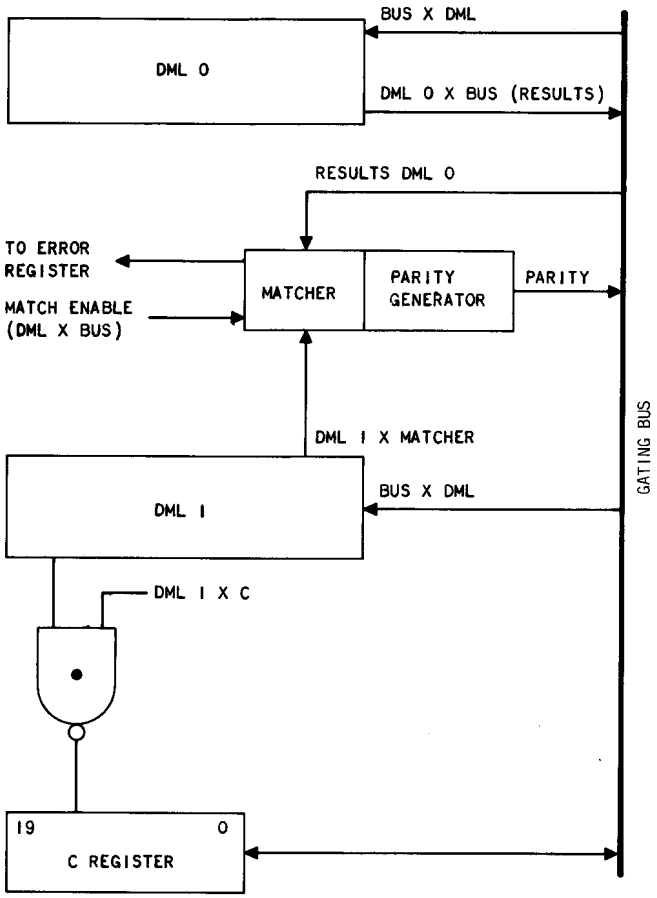


Fig. 39—Duplicated Data Manipulation Logic

Addition

5.73 The adder within the DML is used to add data constants and implements a +1 addition for various counter registers in the 3A CC. In order to perform relative addressing, both forward and backward, or to subtract data constants, the adder must perform the subtract function. The method of handling subtraction is to represent the subtrahend in 2s complement. The 2s complement is formed by complementing the subtrahend and adding a one to the result.

5.74 When a backward transfer is required, the subtract function must be performed. The complement of the operand field of the OP code or instruction is loaded into the B register (the remaining bits of B register are set to 1). The present memory address is gated to the A register. The DML adds B to A and then +1 to the result. Thus, the 2s complement has been used to form the new address.

Rotation

5.75 The rotate function in the DML is performed using combinational logic. This operation is a gating function of data from one register to another. The data is gated within the DML either into the B register from the A register or into the A register from the B register. When the data is gated from the AR to the BR (see Fig. 40) it may be rotated left or right, 0, 4, 8 or 12 bit positions. When the data is gated from the BR to the AR, it may be rotated left or right, 0, 1, 2, or 3 bit positions. Combinations of these rotate operations may be used to rotate 16 bits of data as many as 15-bit positions, left or right.

Find Low Zero Test

5.76 The 3A CC reads memory words to determine which equipment is idle and available for use. The find low zero (FLZ) instruction specifies detection of the first binary 0 (indicates idle equipment) in one of the general registers. In

this case, assume R1. The microprogram control loads the contents of R1 into the AR of the DML. The DML, by use of logic circuits, then detects the least significant 0 in the AR.

5.77 The find low zero logic converts the position of the low order zero found in the AR (Copy 0) and converts it into a binary number representing the position of this low order zero. In addition, if there are no zeros found, a signal is activated that indicates the AR is all ones. The result of this signal is gated into the DS bit of the MCS register when the find low zero test is enabled by a miscellaneous decoder crosspoint. A miscellaneous decoder crosspoint is used to activate the find low zero test and as such, gates the binary output to the low 4 bits of the BR0.

G. I/O Channel and Controller

5.78 The 3A CC utilizes serial I/O main channels to communicate with peripheral equipment designed to work with serial channels. Each I/O main channel contains a controller which can handle 20 I/O subchannels. The I/O channel controllers are the interface by which information is communicated between the rest of the 3A CC and the other 2B processor units. The initial system design uses the subchannels of one I/O main channel to communicate with the teletypewriter controllers, tape data controllers, and the system status panel.

5.79 A maximum of 18 serial I/O main channels can be used to communicate with peripheral equipment designed to work with serial channels (Fig. 41). The I/O structure allows growth in modules of one main channel (20 subchannels). This modularity and the fact that each communication link is a dedicated path enables the system to grow easily. Thus, in a working system, a new peripheral unit can be installed and then connected to a serial I/O channel without interfering with the operation of existing peripheral units.

I/O Controllers

5.80 An I/O controller contains an I/O status (IOS) register, an I/O control, an I/O data (IOD) register, decoders, bipolar drivers, and bipolar receivers (Fig. 42).

(a) ***IOS Register:*** The IOS register is a 10-bit register containing the channel status and subchannel selection code. The 3-out-of-6 subchannel selection code is contained in the upper six bits of the IOS with the control state code contained in the lower four bits.

(b) ***I/O Control:*** The I/O control shifts and directs the operations involved in the I/O order cycle. The main states of the I/O cntrol are the ***idle, transmit, receive,*** and ***message complete states.***

(c) ***IOD Register:*** The IOD register is a ***21-bit shift register*** which serves the following two functions:

(1) To contain the data, parity, and start code to be sent to the periphery.

(2) To contain the response from the periphery to the previously mentioned data.

(d) ***Decoders:*** The 1-out-of-20 select decoder circuit uses the upper six bits of the IOS register to address the desired subchannel.

(e) ***I/O Subchannels Drivers and Receivers:*** The drivers and receivers send and receive bipolar pulses over an ac transmission line. These pulses contain the message sent to or received from the periphery.

I/O Control Signals

5.81 There are nine miscellaneous decoder control signals associated with I/O operations. The nine miscellaneous control signals are used for the following functions:

(a) Gate the contents of R10 including the parity bits to the IOD register.

(b) Gate the contents of IOD to processor R11. The channel must be in the idle state or one of the message complete states for this operation to function correctly.

(c) Idle the main channel. It is necessary to have the channel in the idle state when loading all of the internal registers to prevent either sequence errors or check circuit errors. This idle may be used at any time to stop the channel without causing the check circuits to fire.

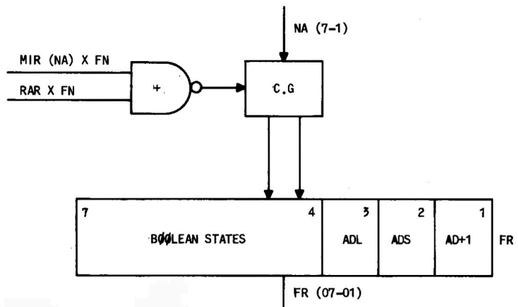
(d) Load the subchannel select field in the IOS from R9 bits (9-4) and load the control field with the transmit state.

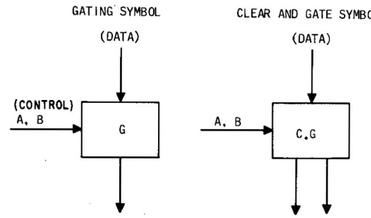
(e) Load the subchannel select field in the IOS from R9 bits (9-4) and load the control field with the transmit maintenance state.

(f) Load the subchannel select field in the IOS from R9 bits (9-4) and load the control field from R9 bits (3-0). In this case, 01 is added to

BØØLEAN STATES ØF THE FN REGISTERS

	FN REGISTER BIT				FUNCTION PERFØRMED
	7	6	5	4	(LE.F (AR, BR))
0	0	0	0	0	1
1	0	0	0	1	$\overline{AR} + \overline{BR}$
2	0	0	1	0	$\overline{AR} + BR$
3	0	0	1	1	$\overline{AR}$
4	0	1	0	0	$AR + \overline{BR}$
5	0	1	0	1	$\overline{BR}$
6	0	1	1	0	$\overline{AR}.\overline{BR} + AR.BR$
7	0	1	1	1	$\overline{AR}.\overline{BR}$
8	1	0	0	0	$\overline{AR + BR}$
9	1	0	0	1	$\overline{AR}.BR + AR.\overline{BR}$
10	1	0	1	0	$\overline{BR}$
11	1	0	1	1	$\overline{AR}.BR$
12	1	1	0	0	AR
13	1	1	0	1	$AR.\overline{BR}$
14	1	1	1	0	AR.BR
15	1	1	1	1	0





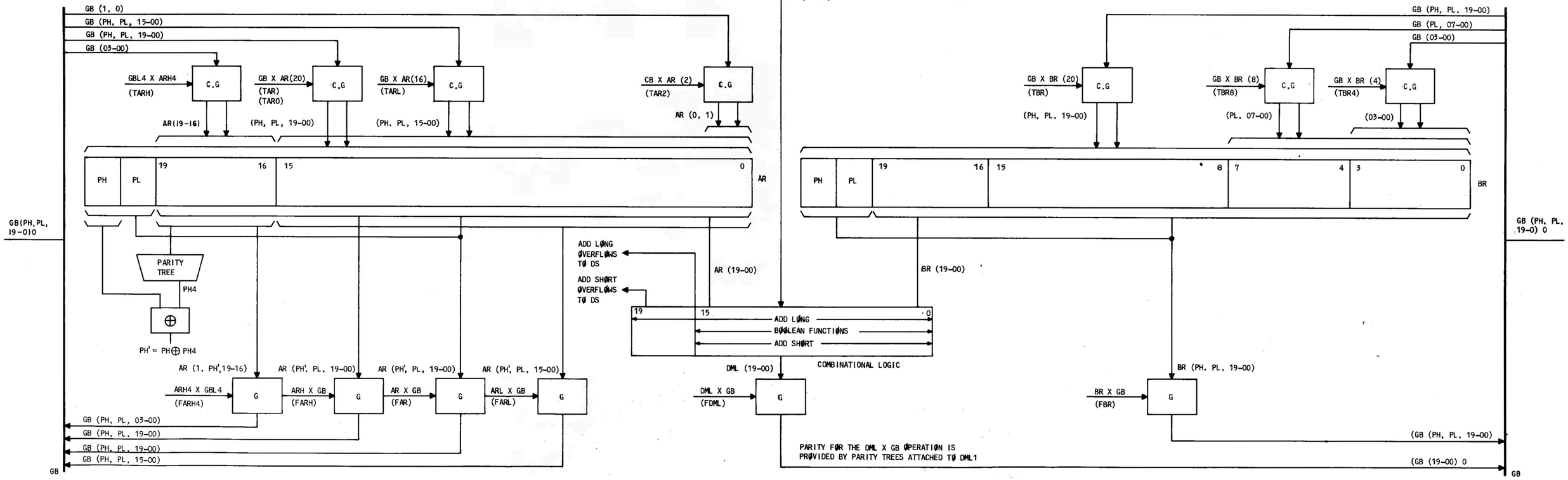


Fig. 40—Breakdown of Data Manipulation Logic

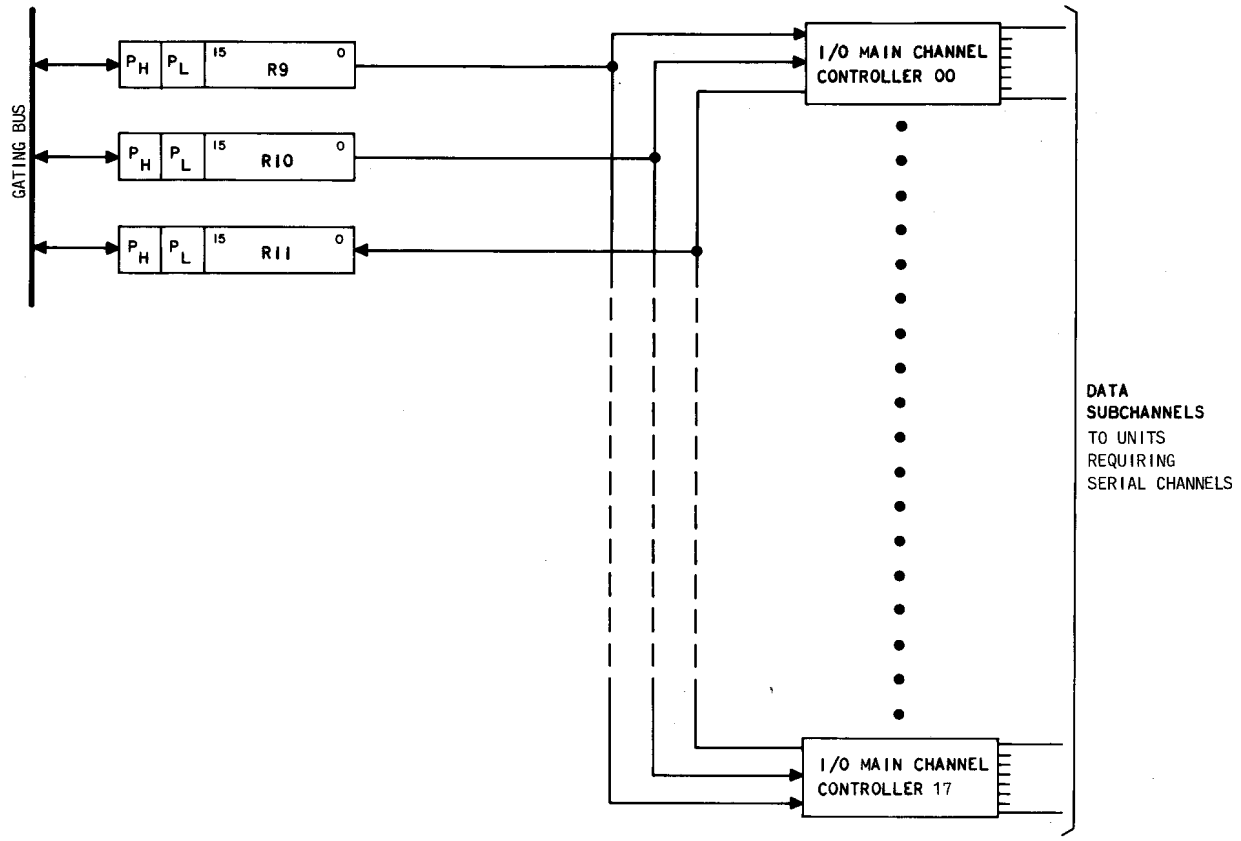


Fig. 41—Input/Output Facility

the front of the 4-bit control field to make up the proper 6-bit code.

(g) Test the channel to load processor status flip-flops with the I/O channel state.

(1) Idle state sets DS.

(2) Message state sets TR1.

(3) Maintenance message state sets TR2.

(h) Start the main channel.

(i) Enable the gating path from the IOS to the I/O return data path. The IOD must be cleared prior to this so that the two outputs will OR correctly.

I/O Order Execution

5.82 ***Data Sending:*** Before the I/O send operation begins, the microprogram control loads R9 and R10 with the control and data information necessary for the operation. The channel select (CS) field of R9 is decoded to select which one of the 20 main channels is to be enabled. This decoding results in the CS gate becoming enabled in the selected I/O main channel controller. The 16 bits of R10 are gated into the IOD. ***To check the gating to the IOD and the information to be sent out, a loop around test is performed.*** The contents of the IOD are loaded into R11. The contents of R10 and R11 are gated to the DML and matched. If an error is detected, a bit which will cause an interrupt is set in the error register.

5.83 By using the 6-bit code in the IOS, the decoder circuits select the desired subchannel. The microprogram control gives the start signal to the I/O controller. The contents of the IOD are then transmitted over the selected subchannel. After shifting out the complete contents of the IOD, the controller goes into a ***receive state*** and continues to provide timing pulses.

5.84 The message or data is sent out via bipolar pulses on an ac coupled transmission line at a rate of 6.67 megabits per second. The receiving end obtains its timing by using the bipolar bit stream. A 3-bit start code is contained within the message. In most cases, the reception of this code by the receiving register sets a bit to indicate to the peripheral device that the complete message has arrived. However, some devices may count bits or quit after receiving a number of bits. In any case, the shifting of the IOD is inhibited when it is emptied and the ***controller goes into the receive state awaiting the results of the peripheral device.*** The parity of the transmitter message may be checked in the device. Over the I/O channel, bipolar pulses (zeros) are continuously sent to the peripheral unit controller. These pulses provide the timing for the peripheral unit controller to execute the order and to send the response back to the I/O.

5.85 ***Data Receiving:*** Upon executon of the order by the peripheral device, a response is dumped into a shift register along with its correct parity at the peripheral device. The peripheral device then initiates a data transmission to the I/O by enabling the incoming bit stream to shift the message back to the IOD. The complete arrival of this message is indicated by the correct registration of the start code in the IOD. ***The sending of bipolar pulses to the periphery is then inhibited, and the I/O controller enters the message complete state.*** A bit in the MCS is set which indicates that the response is available in the IOD. When the response is read out by the microprogram control, the parity of the response is checked. If no error was found, the I/O controller is put in the idle state and the I/O order cycle is completed. If the parity check indicates an error, a bit is set in the error register.

5.86 ***After receiving the start signal from the 3A CC, the I/O controller operates independently of the 3A CC.*** This asynchronous operation within the I/O order cycle is carried out by buffering at both ends of the communications link. Therefore, the 3A CC may accomplish other tasks during this time. This is especially advantageous when a peripheral device is very slow or when various I/O channels are to operate simultaneously but accomplish different tasks.

H. Interrupt Facility

General Function

5.87 The interrupt facility provides for any desired input to the 3A CC to be recognized and serviced relative to its priority. Interrupts may be caused by such inputs as control panel operations, timing counter, and certain error conditions.

Description

5.88 The interrupt facility (Fig. 43) consists of an interrupt set register (IS) and an interrupt mask (IM) register. The IS buffers the input signal that requests the interrupt until the 3A CC is able to transfer its control and recognize the interrupt. The IM is used to mask out interrupts, such as stuck interrupts or interrupts of a lower priority.

5.89 The interrupt facility allows interrupts to occur at 16 different levels. Each of these levels can be individually masked by the 16 bits of the IM register. Only seven of the 16 levels are defined as shown in Table D. The highest level or priority of interrupt is the address or data match (bit 3 set); the lowest is a manual panel execute (bit 13 set).

Operation

5.90 Depending on the time of the interrupt relative to a microsequence, a period of up to one complete instruction sequence (two half word 2B instructions) may elapse before the interrupt is acknowledged by the microprogram control. When the all zeros location of microstore is accessed at the end of an OP code, the microprogram control tests for interrupts. If an unmasked interrupt is present and the block interrupt (BIN) flip-flop is not set, a hard-wired interrupt address is gated into the MAR regardless of whether or not a previous main memory fetch is completed (DR=1). This address initiates a routine in microstore that tests for the highest level of interrupt within the IS register and translates the bit position of that interrupt to a data constant. This constant points to the main memory program which handles that type of interrupt. During this process, the microprogram control sets the BIN flip-flop. The BIN flip-flop is used by the microprogram control to block any additional interrupts. This provides the interrupt program

NOTE:
THE TWO TYPES OF MESSAGES SENT TO THE PERIPHERAL DEVICES ARE THE NORMAL AND MAINTENANCE TYPE MESSAGES. BOTH ARE HANDLED BY THE I/O FACILITY IN THE SAME MANNER. THE ONLY DIFFERENCE IS THE START CODE OF THE IOD (101 MEANS MAINTENANCE MESSAGE; 011 MEANS NORMAL MESSAGE).

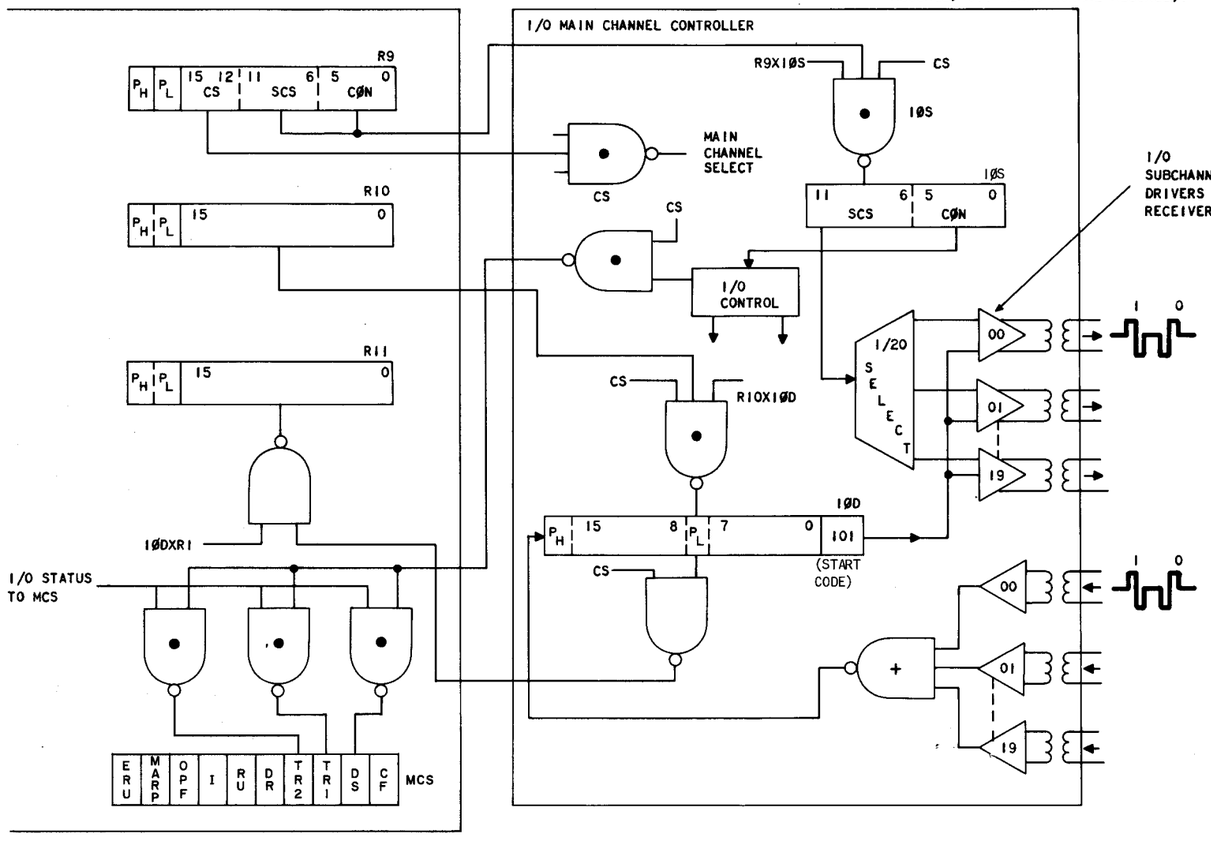


Fig. 42—Input/Output Main Channel Controller

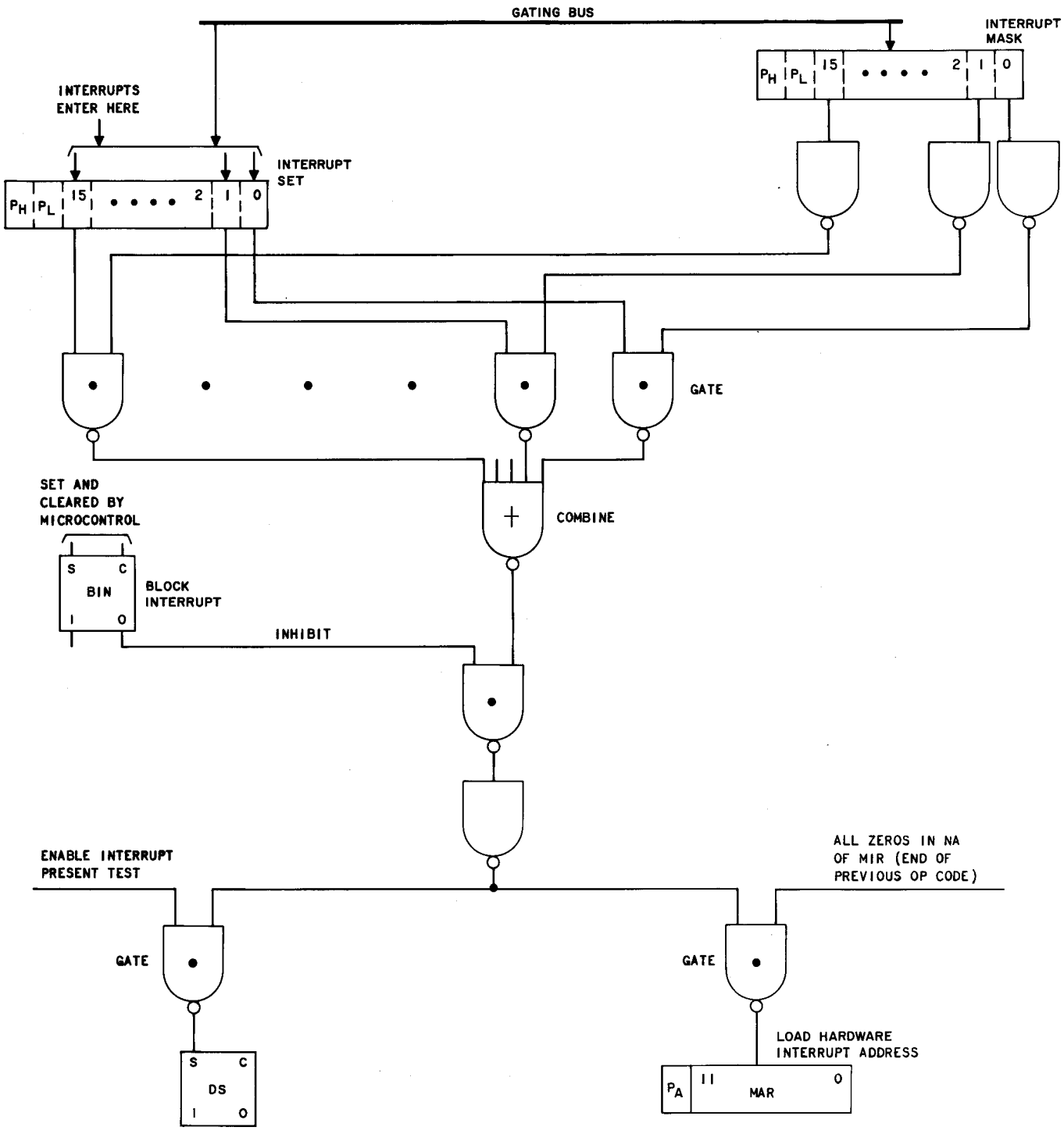


Fig. 43—Interrupt Facility

TABLE D

DEFINED LEVELS AND BIT LOCATIONS IN THE INTERRUPT SET REGISTER

IS BIT	FUNCTION
3	Address or Data Match
5	Error Register (Interrupt Class)
7	Other CC (External)
8	5 ms Interrupt
9	25 ms Interrupt
13	3A CC Control Panel Execute

time to set up and handle the present interrupt without others interfering.

Note: Interrupts are not serviced between the left and right half of 2B half-word instructions, therefore, two complete instructions can be processed before an interrupt is serviced.

5.91 Diagnostics are used to detect faults in the interrut facility.

I. Maintenance Channel and Maintenance Channel Controller

General Function

5.92 The maintenance channel (MCH) is an asynchronous, semiautonomous data transfer system capable of serial ac data transfers at a rate of 6.67 megabits per second. It provides a half-duplex mode (one-way transmission at a time) of communication between the duplicated 3A CCs. This communication is necessary for one 3A CC to determine the state of the other 3A CC and for the on-line 3A CC to exercise the other 3A CC as well.

5.93 The MCH controller of the on-line 3A CC is used to perform the following functions in relation to the other 3A CC:

(a) Arbitration of on-line/off-line status

(b) Periodic or diagnostic exercise

(c) Stopping

(d) Starting or initializing

(e) Updating the program timer

(f) Disabling the I/O

(g) Controlling the clock.

Description

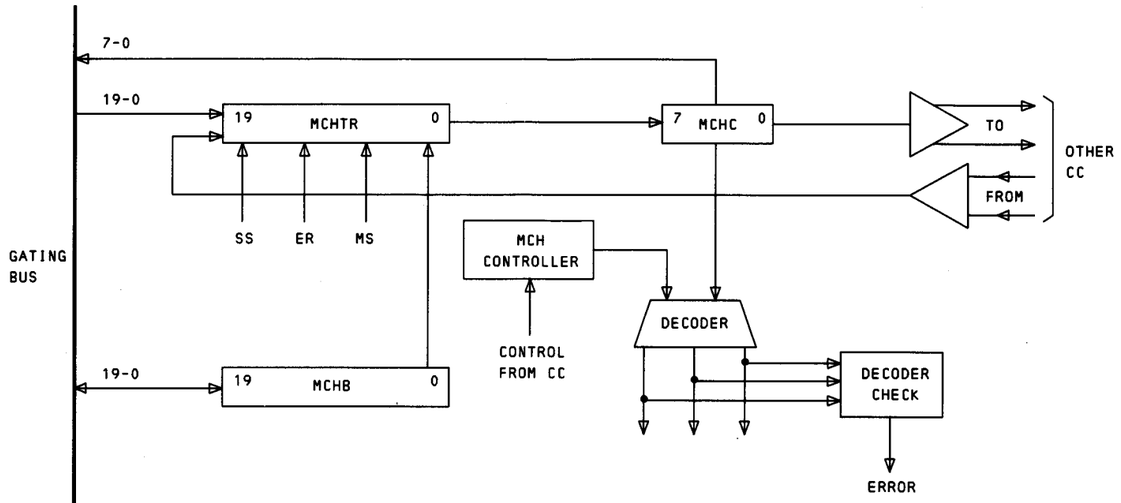
5.94 The MCH consists of two identical semi-independent controllers (one in each 3A CC) connected by a pair of coaxial cables. Each controller (Fig. 44) consists of an MCH transmit/receive register (MCHTR), an MCH command register (MCHC), an MCH buffer register (MCHB), sequencers and control logic, error check circuitry, and a command decoder. The MCHTR is a 22-bit shift register used to buffer data sent to or received from the other 3A CC. The MCHC is an 8-bit register used to buffer the start bit and 3-out-of-7 MCH order. The MCHB is a 22-bit register used to temporarily store the information received from the other 3A CC. The command decoder is used to decode the 3-out-of-7 MCH order located in the MCHC. The sequencers are responsible for the basic control of the MCH controller.

Operation

5.95 Transmissions are initiated by either of the 3A CCs (independent of its on-line/off-line status). The transmission link is seized and controlled by the 3A CC making the first request. In this description, the MCH controller that initiated the transmission will be called the master controller. The other controller will be called the slave controller.

5.96 The normal or rest state of the main sequencers (both on-line and off-line) is the monitor state. In this state each MCH controller is actively monitoring its own incoming serial data bus. Any incoming bit stream will be buffered in the appropriate register independent of the 3A CC status.

5.97 ***Normal Transmission:*** The microprogram control loads the MCHC and the MCHTR of the master controller. The microprogram control



LEGEND:
- CC - CENTRAL CONTROL
- ER - ERROR REGISTER
- MCHB - MAINTENANCE CHANNEL (MCH) BUFFER
- MCHC - MCH COMMAND REGISTER
- MCHTR - MCH TRANSMIT/RECEIVE REGISTER
- MS - MAINTENANCE STATE REGISTER
- SS - SYSTEM STATUS REGISTER

Fig. 44—Maintenance Channel Controller

also sets the start bit and puts the main sequencer into the master transmit state. The master controller then outpulses the 30-bit message onto the channel.

5.98 After sending out the message, the master controller advances to the master receive state while continuing to outpulse an all zero data stream. This all zero data stream is used by the receiving controller as a timing source.

5.99 The slave controller shifts the data and command into the MCHTR and MCHC until the start bit registration is obtained. When the start bit becomes 1, the slave sequencer goes into the slave execute state. In this state, the command is decoded and executed. The results are loaded into the MCHTR. The return code is loaded into the MCHC and the start bit is set to 1. The slave controller checks for errors and puts the slave sequencer into the slave transmit state. The return message is transmited via timing derived from the string of zeros from the master controller.

5.100 The master controller, which is in the master receive state and still outpulsing zeros, detects and shifts the return message into the MCHTR and MCHC. After the registration of the start bit, the data of MCHTR is gated into the MCHB. The master controller then ceases to outpulse all zeros and sets the return bit (it sets an error bit if an error code was returned). The master sequencer then enters the wait state.

5.101 When the slave controller detects no all zeros being outpulsed from the master controller, it initializes and goes into the monitor state. Thus, the master controller will detect the termination of the incoming bit stream from the slave controller. This causes the main sequencer of the master controller to initialize to its monitor state.

5.102 ***Switch Transmission:*** If a fatal error in the on-line 3A CC occurs, a system reconfiguration is necessary and a switch signal is sent to its MCH controller. The MCH must respond to this error signal automatically with a coded transmission to the standby 3A CC. This coded transmission tells the standby 3A CC to switch on-line and take control of the system. Since this switch message can occur at any time relative to MCH operation, it has the ability to abort any current MCH transmission, initialize the main sequencer, jam the MCHC to a 3-out-of-7 switch code, set the start bit, and initiate the switch transmission. All of this is accomplished without any external control other than the clock from the faulty 3A CC.

5.103 A separate sequencer called the switch sequencer is used in the MCH controller to initiate the switch transmission. As long as no error signal is present, the switch sequencer is not involved in the transmission and reception of MCH messages. It is continually monitoring the error lead in its normal or active state.

5.104 Upon receiving an error pulse, the switch sequencer goes into the initialization state. During this state, the switch sequencer initializes the main sequencer. After initializing the main sequencer, the switch sequencer goes into the load and go state. During this state, the switch sequencer jams the MCHC to the switch code, sets the start bit, and jams the main sequencer into the master transmit state. From this point, the main sequencer takes command and transmits the switch message as if it were a normal message. Meanwhile, the switch sequencer goes into the disable state and remains until initialized by the new on-line 3A CC.

5.105 After the main sequencer has transmitted the switch message, it goes into the master receive state expecting a reply from the other 3A CC. The switch message causes the standby 3A CC to intialize and go on-line. The new on-line 3A CC then sends a message to the 3A CC formerly on-line which forces its main sequencer to the monitor state. The new on-line 3A CC then recycles its own main sequencer to the monitor state. At this time, normal communications between the 3A CCs may be initiated. Before restoring the switch sequencer in the faulty 3A CC to the active state, the source of the error must be removed to prevent subsequent switch transmissons.

J. Control Panel and Control Panel Functions

General Function

5.106 The control panel provides a means of communication between maintenance personnel and the 3A CC. Numerous lamps and switches appear on the panel and provide the maintenance personnel manual control of the 3A CC for troubleshooting purposes.

5.107 With the control panel in the MANUAL mode and the EXECUTE switch activated, the panel causes an interrupt to notify the microprogram control of a requested panel function. Under microprogram control, the contents of the switches on the control panel can be selectively gated into the display buffer (Fig. 45). Once in the display buffer, the switches are interpreted and the appropriate action taken. The panel functions normally consist of load and display operations which are accomplished by gating information to the appropriate registers. In addition, the data match registers are provided to allow the panel to match on memory operations that use the address and data as indicated in the registers. The address input (AI) and address mask (AK) registers are used in address matching (Fig. 45). The data input (DI) and data mask (DK) registers are used in data matching.

5.108 The panel is divided into four major areas so that functionally related controls and displays are grouped together. These are STATUS, REGISTER SELECT, LOAD AND DISPLAY, and functional keys and lamps. Table E lists the keys, lamps, and switches of these areas and their purpose.

Load and Display

5.109 The LOAD AND DISPLAY portion of the control panel provides a means to display, load, and/or match inputs of up to 22 bits each.

Register Select

5.110 The REGISTER SELECT portion selects which register (general or special addressable) to load or display.

Functional Keys and Lamps

5.111 This portion of the control panel allows manual functions to be selected and performed. This portion is further broken down into REGISTER, COMPARE, MEMORY, MODE, and COMMAND sections.

- The REGISTER section of the panel is used to define either a load or display function for the selected register.
- The COMPARE section of the panel is used to match either the SAR with the data stored in the address input buffer or the store data bus with the data stored in the data input buffer.
- The MEMORY section of the panel allows maintenance personnel to read or write data into the main memory.
- The MODE section of the panel is used to execute one instruction or to put the 3A CC in a microprogram store loop which executes no program code but honors interrupts.
- The COMMAND section of the panel enables the user to execute the manual functions which are set up. If the manual function was incorrectly set up or for some other reason fails, a REJECT indication lights showing that the function was not executed.

Status

5.112 The STATUS portion of the control panel indicates and controls the state of the 3A CC and its power. For testing purposes, keys are available to place the 3A CC in the manual and test mode.

K. Miscellaneous

Subroutine Facility

5.113 A hardware assisted subroutine facility is used with the 3A CC. A 16-bit register called the hold-get (HG) register is used as a special push-down address arrangement. This arrangement automatically allocates a group of words in the main store to facilitate entry, nesting, return, and releasing of general registers for use by subroutines.

5.114 Each of the 32 levels of the HG area contains 16 words of storage. Words 0 and 1 of each level contain the subroutine return address, and words 2 through 15 contain the contents of general registers 2 through 15 at the time the subroutine was called.

5.115 The HG register is decremented by 16 on each subroutine call and incremented by 16 on each return. On each subroutine call, the complete (20-bit) address is saved in the first two words of the associated block.

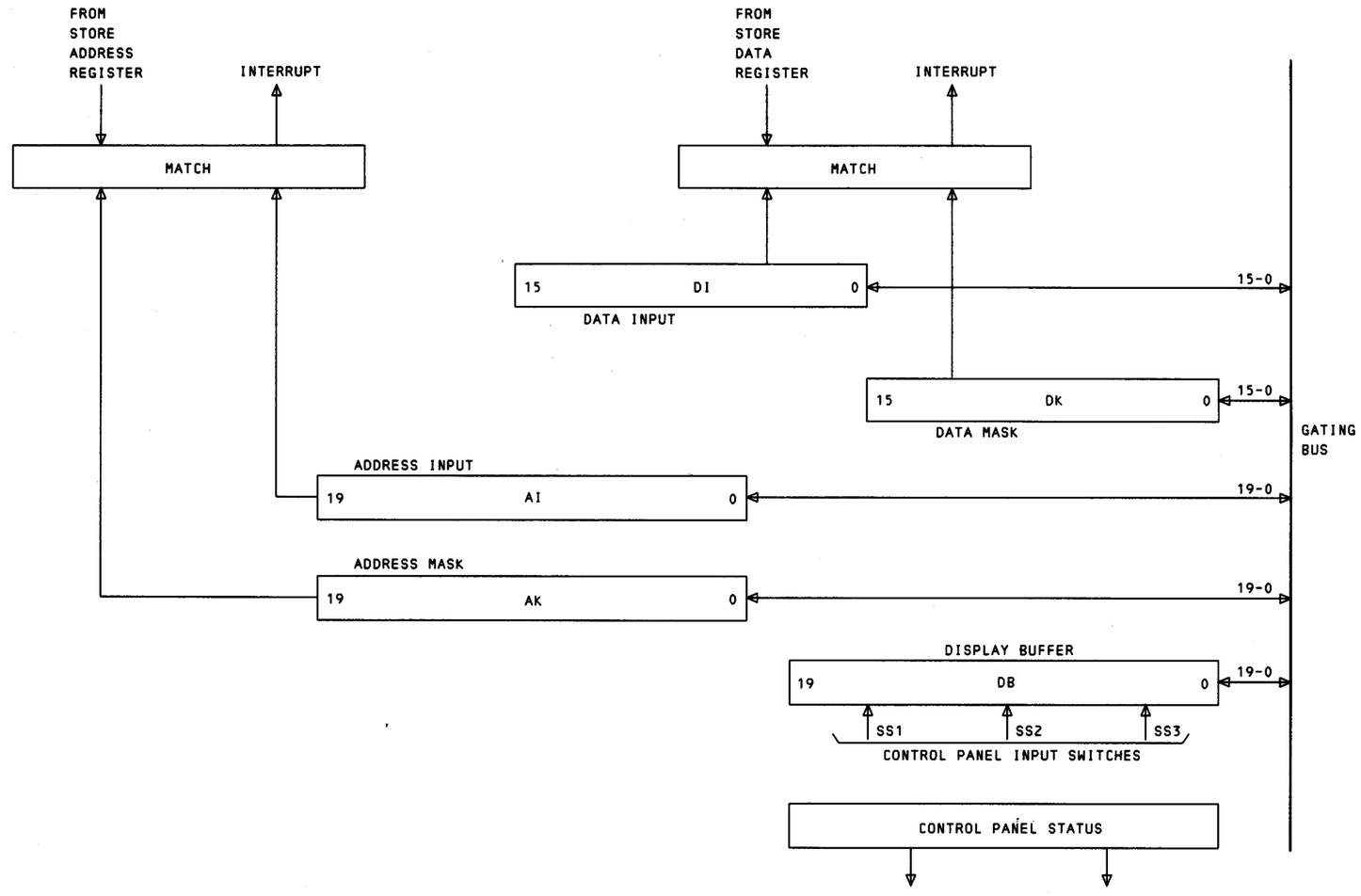


Fig. 45—Registers Associated With Control Panel Operations

TABLE E

3A CC CONTROL PANEL KEYS, LAMPS, AND SWITCHES

AREA	DESIGNATION	COLOR	INDICATION OR FUNCTION
LOAD AND DISPLAY	0-19, PH, PL (LEDs)	Green	Visual indication of the contents of the display buffer. The LEDs are divided into groups of three or four for easy conversion to either octal or hexadecimal.
	0-19, PH, PL (Switches)	Blue, White, and Gray	Manual input to the display buffer. Switches are divided into groups of three by blue and white colors for easy conversion to octal. Switches are also divided into groups of four for easy conversion to hexadecimal. PH (parity high) and PL (parity low) are used only when the ENABLE MANUAL PAR switch is operated.
	ENABLE MANUAL PAR (Switch)	Gray	Allows the manual setting of the parity switches (PH, PL). Unless this switch is operated, the 3A CC automatically generates parity for the input switches 0 through 19.
REGISTER SELECT	8, 4, 2, 1 (Switches)	White	Selects one of the 16 general or 16 special (addressable) registers depending upon the setting of the SPECIAL/ GENERAL EXTENDED switch.
		White Switch	This switch in the normal (down) position selects the general register group.
	SPECIAL/(Switch with LEDs) GENERAL EXTENDED	Green LEDs	The GENERAL LED indicates when this group has been selected. This switch in the operated (up) position selects the special register group. The SPECIAL LED indicates when this group has been selected. The special registers are: MCTL STAT — Microcontrol Status Register (read only) TIM — Timing Counter (read only) SYS STAT — System Status Register (read only) ST ADRS — Store Address Register PROG ADRS — Program Address Register MTCE STA — Maintenance State Register M MEM STAT — Main Memory Status Register MCH BUFR — Maintenance Channel Buffer INT SET — Interrupt Set Register INT MASK — Interrupt Mask Register HOLD GET — Hold Get Register ERR — Error Register DATA MASK — Data Mask Register DATA IN — Data Input Register ADRS MASK — Address Mask Register ADRS IN — Address Input Register

TABLE E (Cont)

3A CC CONTROL PANEL KEYS, LAMPS, AND SWITCHES

AREA	DESIGNATION	COLOR	INDICATION OR FUNCTION
REGISTER	LOAD (Switch)	White	Allows the contents of the LOAD AND DISPLAY switches to be loaded into the designated register of the REGISTER SELECT switches.
	DISPLAY (Switch)	White	Allows the appearance on the LOAD AND DISPLAY LEDs of the contents of the designated register in the REGISTER SELECT switches.
COMPARE	ADR (Switch)	White	Enables the match between the contents of the store address register (SAR) with the address input (AI) register.
	ENABLED (LED)	Green	Indicates when a COMPARE function is active.
	DATA (Switch)	White	Enables the match between the contents of the store data bus with the data input (DI) register.
MEMORY	INCR ADR (Switch)	White	Increments the contents of the SAR by one.
	STORE (Switch)	White	Writes the contents of the LOAD AND DISPLAY switches into the main store at address in SAR.
	DISPLAY (Switch)	White	Reads the main store at the address in the SAR and displays the contents of that location.
	HIGH BITS/ (Switch) LOW BITS	White	This switch in the normal (down) position allows the display or storing of bits 0 through 15 of data in a main store location. This switch in the operated (up) position allows the display or storing of bits 16 through 31 (possible) of data in a main store location. The operated position is not effective unless the BASIC EXTENDED switch is in the EXTENDED position.
MODE	HALT (Switch)	White	Puts the 3A CC in a microstore loop that executes no program code, but honors interrupts.
	HALTED (LED)	Green	Indicates that the 3A CC is not executing program codes.
	STEP (Switch)	White	Allows the execution of program instructions, one at a time.
	BASIC (Switch) EXTENDED		This switch in the up (EXTENDED) position allows the HIGH BITS/LOW BITS switch to operate properly in the up (HIGH BITS) position.

TABLE E (Cont)

3A CC CONTROL PANEL KEYS, LAMPS, AND SWITCHES

AREA	DESIGNATION	COLOR	INDICATION OR FUNCTION
COMMAND	REJECT (LED)	Green	Indicates that the last manual function attempted was not performed. This may be due to incorrect combination of panel keys.
	EXECUTE (Switch)	Gray	Initiates a microprogram interrupt that results in performance of the selected manual panel function.
STATUS	POWER (Key/Lamp)	Green	Depending on the power state, operation causes a sequential removal of power from the 3A CC (as allowed by system conditions) or a sequential restoral of power to the 3A CC.
	ACTIVE (Lamp)	Green	Indicates the status of the 3A CC. Follows the "one" side of the Central Control flip-flop.
	STANDBY (Lamp)	White	Indicates the status of the 3A CC. Follows the "zero" side of the Central Control flip-flop.
	MANUAL (Key/Lamp)	Amber	Operation enables panel functions in the standby 3A CC or on-line if in TEST MODE.
	ERROR STOPPED (Lamp)	Red	Indicates the status of the STOP flip-flop. This flip-flop is set by error detection circuits or the other 3A CC and is cleared by initialization hardware or the other 3A CC. When this lamp is lighted, the 3A CC is in the STOP state.
	RESET CIRCUITS (Key/Lamp)	Red	In standby, active only in the MANUAL mode. In on-line, active in MANUAL and TEST MODE. Initializes the critical flip-flops and puts the 3A CC in the HALT state. The HALTED LED lights.
	TEST MODE (Lamp)	Red	Lights only when the test mode switch (located on inside of panel) is active. Test mode switch enables panel functions and power removal in an on-line 3A CC.
	LAMP & POWER TEST (Key)		Utilized to ensure that all lamps within STATUS area will light and to perform a test of the power alarm circuits in the power converters. The power converters are located within other units of the frame. When the key is operated, the converter LEDs will light. When the key is released the converter LEDs extinguish.

5.116 Two instructions are provided to call a subroutine. The first is a double-word branch and save address (BSA) instruction. This instruction causes the HG register to be decremented by 16 (one level), the return address to be stored in words 0 and 1 of the HG area, and the transfer to an address contained in the BSA instruction. The second is a single-word branch and save address indirect (BSAI) instruction. This instruction causes the HG register to be decremented by 16 (one level), the return address to be stored in words 0 and 1 of the HG area, and a transfer to a new address. The address is determined by an 8-bit field (low eight bits) in the instruction and 12 bits (12 high bits) in the microstore.

5.117 After the subroutine is called, the contents of general registers 2 through 15 are saved and held in words 2 through 15 by a hold all instruction.

5.118 Four subroutine instructions are used to provide several options on return. These are listed and explained as follows:

(1) Return (BTSA)—Return to the saved address stored in words 0 and 1 of hold-get area.

(2) Get all and return (BTSAG)—Load registers 2 through 15 of the hold-get area and increment the hold-get register by 16, and then return to the saved address stored in words 0 and 1 of the hold-get area.

(3) Load return code and return (BTSAN)—Load the return code, N, into bits (3-0) of R0; zero the remaining bits (15-4) of R0; and then return to the saved address stored in words 0 and 1 of the hold-get area.

(4) Get all, load return code, and return (BTSAGN)—Load R2-R15 from words 2-15 of the hold-get area; load the return code, N, into bits (3-0) of R0 and zero bits (15-4) of R0; then return to the saved address stored in words 0 and 1 of the hold-get area and increment hold-get register.

Error Register (ER)

5.119 The ER buffers the error signals of the check circuits of the 3A CC to aid diagnostic programs in localizing hardware faults. Error signals in the ER are divided into the following three types:

(a) Error signals that result in a switch of 3A CCs

(b) Error signals that result in a hardware initialization

(c) Error signals that result in an interrupt.

(d) Error signals that result in an error correction.

Note: Bit 10 (store read parity error) does none of the preceding. It is examined by software.

Figure 46 shows the error register and its associated circuitry. The bit designations of the ER are shown in Table F.

5.120 The error signals that result in a switch of 3A CCs are located in bits 0 through 9 of the ER. If the block hardware check (BHC) flip-flop is in the zero state and one of these bits is set, one of two actions occurs, depending on the state of the lock on-line (LON) flip-flop. When LON=0, the stop flip-flop is set and a switch signal is sent to the MCH controller. When LON=1, a signal is sent to the microprogram control to cause a hardware initialization.

5.121 The error signals that result in an initialization are located in bits 11 through 13 of the ER. When one of these bits becomes a 1 and the BHC=0, a signal is sent to the microprogram control to cause a hardware initialization.

5.122 The error signals that may result in an interrupt are located in bits 14 through 21 of the ER. These bits are ORed together. When one of these bits becomes 1, a bit in the IS of the interrupt facility is set.

System Status (SS) Register

5.123 The SS is a 22-bit register that contains the bits which control the status, and some of the functions of the 3A CC. Table G shows the bit designations and functions of the SS.

TABLE F

INDICATIONS OF ERROR REGISTER

BITS	DESIGNATION	
0	TO Decoder Error	SWITCH ERRORS
1	FROM Decoder Error	
2	IB X, Y Field Error	
3	Bus Parity Error	
4	DML Mismatch	
5	MAR Parity Error	
6	Clock Error	
7	My Store Error	
8	MAR-RAR Mismatch	
9	Function Register Parity Error	
10	Store Read Parity Error	ERROR CORRECTION
11	*My* Store Write Protect	INITIALIZATION ERRORS
12	*My* Store Fast Time-out	
13	Branch Allowed Error	
14	*Other* Store Write Protect	INTERRUPT ERRORS
15	*Other* Store Error	
16	*Other* Store Fast Time-out	
17	I/O Multiple Channel Select	
18	PT Reset Received By On-line 3A CC	
19	Switch Received By On-line 3A CC	
20	I/O Channel Error	
21	I/O Bad Parity Received	

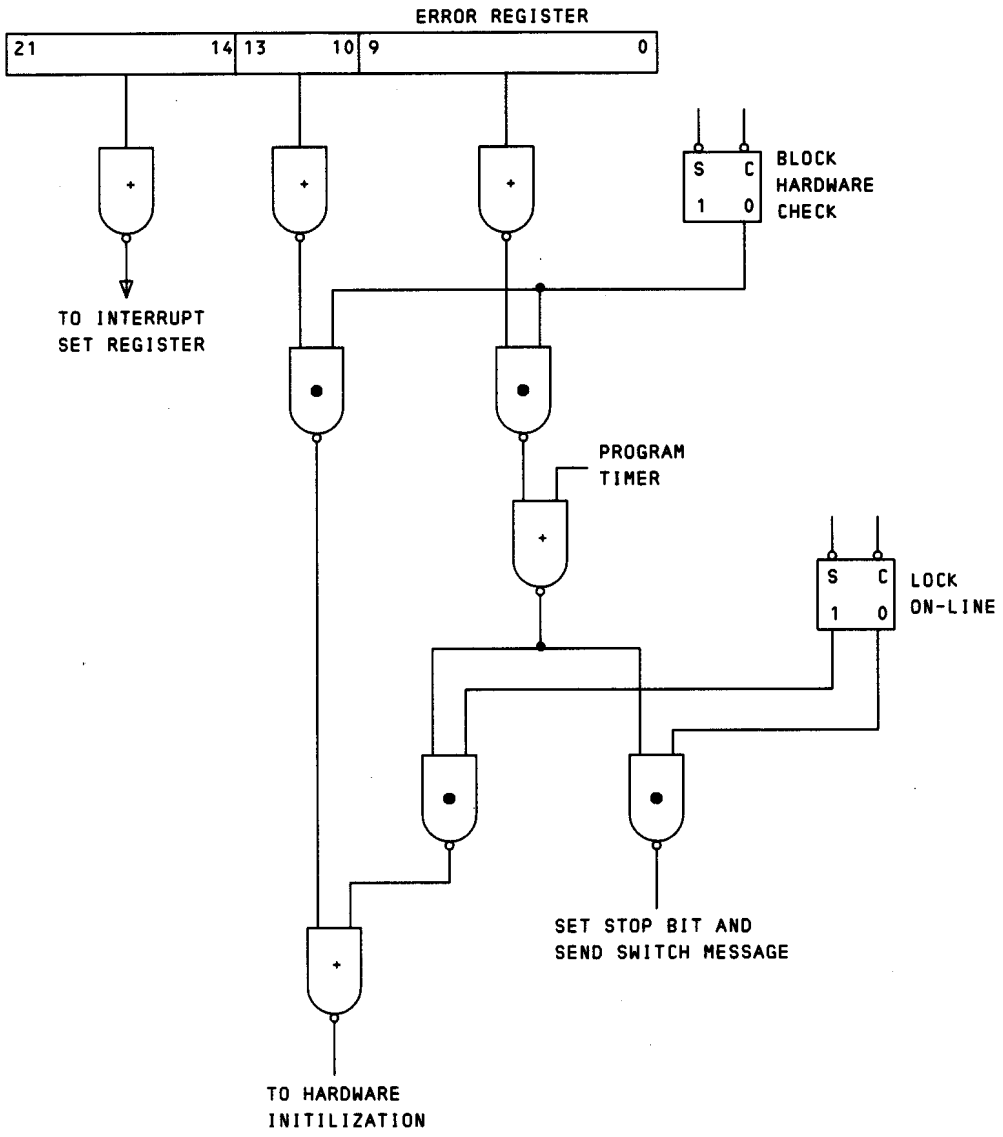


Fig. 46—Error Register and Associated Circuitry

TABLE G

DESIGNATIONS AND FUNCTIONS OF BITS IN SYSTEM STATUS REGISTER

BIT	DESIGNATION	FUNCTION
0	Address Match Enable (AME)	Enables the match between SAR and the address input register.
1	Block Hardware Check (BHC)	Disables the output of the error register (bits 0 through 13).
2	Block Interrupt (BIN)	Disables all interrupts.
3	Block Timer Check (BTC)	Blocks the input and output from the program timer.
4	Data Match Enable (DME)	Enables the match between store data bus and the data input register.
5	Halt (HLT)	Drives the control panel lamp which indicates the 3A CC is in a program halt condition.
6	Initialization Sanity Check 1 (ISC1)	Checks sanity of hardware initialization routine. If a failure is detected, a switch to the other 3A CC will occur.
7	Initialization Sanity Check 2 (ISC2)	Checks sanity of hardware initialization routine. If a failure is detected, the main memory will be reloaded.
8	Lock Off-Line (LOF)	Disables the I/O channels to prevent interference with on-line 3A CC. Allows processor power to be removed via the control panel power key.
9	Lock On-Line (LON)	Forces all hardware switch messages to initialize this 3A CC and keep it on-line.
10	Manual (MAN)	Drives the manual lamp on the control panel which indicates 3A CC (off-line) is in manual mode.
11	Microinterpret Mode (MINT)	Blocks the gating from the microstore into MIR. Enables the gating from main memory into the MIR.
12	Control (CC)	Indicates whether this 3A CC is on-line (CC=1) or off-line (CC=0) and controls various functions which protect on-line 3A CC from the off-line 3A CC.
13	Reject (REJ)	Drives the control panel lamp which indicates that a panel operation was not performed.
14	Stop (STP)	Jams the MAR to a maintenance address causing all zeros to be read out of microstore.
15	Disable	Disables the I/O channels to prevent interference with on-line 3A CC.

TABLE G (Cont)

DESIGNATIONS AND FUNCTIONS OF BITS IN SYSTEM STATUS REGISTER

BIT	DESIGNATION	FUNCTION
16	PRIVILEGE (PRI)	Used for certain applications of the 3A CC that require a privilege mode for instruction execution.
17	Display (DISP)	Disables the gating from the PA to the display buffer on all transfers.
18	Block Bus Parity Check (BPC)	Disables the data bus parity checker.
19	Initiate Program Reload Track (IPLTRK)	Used by the microcode within the sequence which initiates program reloads (IPL) from tape. The state of this bit determines which of the two identical tracks on the tape cartridge to read. This bit is also used to determine when to perform the more drastic initialization of the main store prior to reading data off the tape.
20	Central Control 0 (CC 0)	Always 1 in 3A CC 0 so that program knows which 3A CC it is running in.
21	CC 1	Always 1 in 3A CC 1 so that program knows which 3A CC it is running in.

Status Bits Stored in Memory

5.124 The out-of-service (OS) bit indicates that some abnormal condition is present and that this 3A CC should not be put on-line. This bit is set and reset by software.

5.125 The store-out-of-sync (SOS) bit indicates that the off-line store is out of date. This bit is set and reset by software.

Maintenance State Register

5.126 The maintenance state register (MSR) is a 16-bit register used to set up conditions in a 3A CC for testing purposes. The MSR should be clear during normal operation since most of the conditions set up by the MSR will cause errors. Table H shows the functions of the MSR bits.

6. MISCELLANEOUS CIRCUITS

PROTECTION CIRCUITS

6.01 The protection circuits are used to buffer the signals that cross-connect between the two control units. Those leads that cross connect are mainly the store bus signals that go from one 3A CC to the other 3A CC. The protection circuits prevent an over-voltage or over-current condition from affecting both 3A CCs.

3-VOLT REFERENCE AND FILTER CIRCUIT

6.02 The 3-volt reference and filter circuit provides the interface between the 3-volt power supply and the 1A logic used in the 3A CC.

6.03 This circuitry monitors a reference source and the incoming voltage source from the power supplies and feeds back a signal to the 3-volt supplies if more or less than a fixed amount is indicated. The power supply will respond with the appropriate voltage change. The filter circuit

TABLE H

FUNCTION OF MAINTENANCE STATE REGISTER BITS

BIT	FUNCTION
0	Override LOF
1, 2, 3	Test conditions for clock
4	Ground my store GO lead
5	Ground bus busy lead
6	Utility freeze
7	Ground *my* store complete lead
8	Ground *other* store complete lead
9	Enable IB (X, Y) to MIR when stopped and partially inhibit the FROM check
10	Enable RAR-MAR match independent of RU flip-flop
11	Hold MAR parity and disable miscellaneous decoders on board MC4
12	Disable miscellaneous decoders on board MC4
13	Code Merger Test 1
14	Code Merger Test 2
15	Jam store control lead 3=1

allows time for the power supply to respond to the signal from the reference source. If the reference indicates an out-of-limits range, a power alarm will result.

6.04 A +12-volt reference circuit is used to obtain a very accurate 12-volt reference voltage which is resistor divided down to 3-volts. The 3-volts, which is obtained, is used as the reference voltage in the 3-volt reference and filter circuit.

7. MAINTENANCE FEATURES

FAULT DETECTION

7.01 Fault detection is accomplished by hardware and/or software. The 3A CC is designed to be a self-checking unit utilizing the following error checking techniques.

7.02 ***Bit-Slicing:*** The 3A CC uses 2-bit partitioning or 2-bit slicing to aid in the detection of errors, especially in areas such as the general registers. Two-bit slicing means that two bits of each register are on a single circuit board (see Fig. 24). For example, the first circuit pack contains bits 0 and 8 of every general register. Partitioning is used so that a fault will affect at most only two bits of any register and therefore be detected by the two parity bits.

7.03 ***Parity Check:*** A parity bit is a bit associated with other bits in a field to make the total number of ones, including the parity bit odd. Parity checks are used throughout the 3A CC. Each time information is transferred from one location to another via the data bus, a parity check is performed in the gating bus parity checker.

Whenever incorrect parity is found, an error is indicated.

7.04 ***M-Out-of-N Codes:*** M-out-of-n codes are used in various areas of the 3A CC to provide maximum error detection capability such as the control signals required in the I/O channels and microprogram control. M-out-of-n means that "m" number of ones should be present in "n" number of bits. For example 4-out-of-8 means that no more than or no less than four ones will always be present. The associated decoder check circuits verify that the number of ones is correct. If an incorrect code is detected, an error is indicated.

7.05 ***Duplication:*** Some circuits of the 3A CC are duplicated to detect faults such as the data manipulation circuitry. Duplicated circuits of the 3A CC are given the same inputs and their outputs are then compared to ensure their correctness. Whenever the two outputs differ, an error is detected.

7.06 ***Periodic Detection Tests:*** Since the 3A CC uses self-checking circuits, its fault detection is adequate only as long as the check circuits work properly. Hardware is used to ensure that the check circuits provide an indication when a fault occurs. Hardware provides a means of simulating test conditions or circuit faults. By appropriately setting up the test conditions and applying a well-designed test sequence, the detection circuitry is checked on a periodic basis to verify its proper operation.

7.07 ***Program Timer:*** Although the 3A CC is designed to be as self-checking as possible, an overall system sanity check for both hardware and software is provided by the program timer. The use of the hardware timer is closely related to the system program. A reset is generated for the timer only if the program proceeds through the normal program loop correctly within the prescribed period. If the program deviates from the normal course, no reset is given. The timer automatically times out, stops processing, and starts the recovery process.

MAINTENANCE CHANNEL

7.08 The most powerful tool for maintenance access to the 3A CC is via the maintenance channel. Through the maintenance channel either 3A CC can control the other at the microinstruction level. The maintenance channel has the ability to inhibit the normal microsequencing, and under external control (other 3A CC), load the microstore instruction register (MIR). The loading of this register causes its contents to be executed using the 3A CCs hardware and timing. Since all microinstructions are executed from the MIR, this provides the other 3A CC with complete control in exercising the processor. This feature allows this exercise to be independent of the microstore contents.

7.09 Another important function of the MCH is its ability to verify the contents of microstore. This is accomplished by stopping the processor and loading the next address field of the MIR. This address is gated to the microstore address register (MAR) via processor timing, freezing the MAR at this new address and causing a read from this location. The contents of this word can then be returned to the other processor via subsequent MCH commands. Once the microstore is frozen, the MCH has the ability to begin the normal microsequencing at this new address by issuing a start order.

7.10 The MCH also provides more specialized functions to provide specific diagnostic data. For example, under MCH control, the system clock can be stopped and stepped along one clock phase at a time. In between steps, the state of all phases can be returned to the other processor for analysis. A list of the functions of the MCH are as follows:

(a) Gate the off-line error register into maintenance channel and transmit to on-line processor

(b) Gate off-line register SS into the maintenance channel and transmit to on-line processor

(c) Transmit a constant from the on-line to the off-line CU and leave it in the MCHB

(d) Return the off-line MCHB to the on-line MCHB

(e) Load off-line microinstruction register address field

(f) Load off-line microinstruction register "to" and "from" fields (execute a microinstruction)

(g) Load off-line microinstruction register address field, gate to the MAR and freeze

(h) Stop the off-line processor

(i) Start the off-line processor

(j) Initialize the off-line processor

(k) Disable off-line processor (Step A)

(l) Disable off-line processor (Step B)

(m) Program timer reset

(n) Return the 16-high bits of off-line microstore output to on-line MCHB

(o) Return the 16 low bits of the off-line microstore output to on-line MCHB

(p) Return off-line miscellaneous status bits to on-line MCHB

(q) Zero off-line ER

(r) Zero off-line MSR

(s) Control off-line clock.

INITIALIZATION

7.11 An initialization is a restart of the 3A CC in a particular state. Error signals that cause this action are usually caused by software errors.

7.12 The 3A CC initialization hardware consists of a series of inputs from which hardware initialization signals may be received. These signals are buffered and used to initialize the clock and some key flip-flops in the 3A CC, and to jam a starting address into the MAR. Control logic will then allow the 3A CC to start at this known address in the microstore and sequence through a series of steps that will result in initializing the 3A CC. Eventually the microprogram control will turn the initialization over to the program contained in main store.

DOUBLE STORE READ

7.13 The on-line 3A CC has the ability to read the off-line 3A CCs store when a parity error is detected on a read command of the on-line store. This is accomplished through the double store read function if the off-line store is in the update mode. A normal read command is issued by the on-line 3A CC to its associated store via the main store control. If a parity error is detected when a word is read out to the store, the 3A CC hardware can issue a read command to the off-line store. Double store read is a wired logic function and does not actually use these commands. The request changes the contents of the main memory status (MMS) register (part of main store control). This register contains two bits, called the IDLE bits. These bits define which main store control, 0 or 1, is to be active for a given memory cycle. The "read other store" command complements the IDLE bits so that a read can be issued to the off-line store instead of the on-line store. At the end of the "read other store" command the hardware returns the MMS to its previous state.

7.14 The double store read (DSR) is almost transparent to instruction processing. The only real disadvantage of the DSR mode of operation is the emphasis it puts on keeping the off-line store in the update mode.

COMPLEMENT CORRECTION

7.15 Complement correction is the ability of the 3A CC to correct a single bit error when a parity error is detected on a store read operation. Complement correction is utilized to correct single bit errors when the duplicated stores are not in the up-date mode. The double store read function, previously discussed, is blocked (BDSR=1) when the stores are not in the up-date mode.

7.16 If a single parity error is detected, the main store controller issues a store error C (SERC) signal instead of a store complete signal. The SERC signal together with BDSR=1 causes the 3A CC to invoke a microsubroutine to perform the complement correction. The starting address of the complement correction subroutine and its parity are hardware loaded into the MAR when the SERC fires ***and*** the microcontrol is testing for a store completion signal. When this occurs, the present address and its parity (PNA) that are being read out of the microstore are saved in the error return address register (ERAR). When the complement correction microroutine is complete, it gates the ERAR to the MAR, and the normal microprocessing resumes.

7.17 Complement correction is used if a memory location has a stuck bit in it that causes bad

parity when read. The microroutine complements the nonfailing bits of the word and then writes them back into the failing location. The data is therefore stored in a correct but complemented form. The complemented word has both parity bits bad. This is a flag which is used when the word is again read. Circuitry in the store detects this double bad parity and recomplements all bits before returning them to the 3A CC.

CONTROL PANEL

7.18 The 3A CC control panel provides the maintenance personnel with a means to accomplish the following functions:

- Monitor status of 3A CC
- Load and display registers
- Compare input data with main store data
- Compare input addresses with main store data
- Store and display data
- Halt and step machine through program segments
- Restart the machine using COMMAND-EXECUTE key.

The functions of the control panel keys, lamps, and switches are detailed in Table E.

8. DIAGNOSTICS

8.01 Diagnostic Programs are utilized by maintenance personnel in an operating office in conjunction with trouble locating manuals (TLMs) to locate a faulty circuit pack. Generally a single fault in the system is assumed. The basic principle which is employed in diagnosis of the 3A CC is the "start small" philosophy. This means that a small portion of the machine is first diagnosed, and if this portion of the machine operates properly, it can then be used for further diagnosis. As the diagnosis continues, the portion of the processor which has been checked increases until correct operation of the total processor is verified.

8.02 The 3A CC diagnostics are part of the control unit diagnostic programs which are divided into a series of tests and each test is subdivided into one or more segments. A test is the smallest unit of the diagnostic sequence which may be run independently. A list of tests in the diagnostic sequence is given in Table I. A description of the tests which pertain to the main store are given in Section 232-309-103, Functional Description of Main Store.

8.03 If the request for diagnostic tests is automatic, the complete sequence of CU diagnostics is assumed, starting with test 1, and proceeding sequentially. Running of diagnostics will terminate if a test fails and a failure message will be printed on the TTY containing the failing test number, the test segment number (logical subsection of test), relevant data about the failure and an index into the TLM. The referenced page in the TLM will describe the failing test in some detail and list possible bad circuit packs. If all tests pass, an appropriate TTY message will result.

8.04 If the request for diagnostic tests is by TTY, several options are provided. The sequence of tests may run in the same manner as for automatic requests. Alternately, all tests can be run, independent of failures, and a failure message for each failing test will result. Failure messages from this mode of operation should be interpreted carefully since the diagnostic test sequence utilizes the "start small" philosophy and a single hardware fault which causes an early test to fail may cause any number of later tests to fail as well. Another option for manual requests is for running a single test. A single test may be run once or repetitively, or in the step mode.

8.05 The on-line processor can diagnose the off-line in three ways, allowing the off-line varing degrees of autonomy. These will be referred to as test types A, B, and C.

8.06 ***Type A:*** The on-line diagnoses the off-line over the maintenance channel at the micro level. In this mode, the output of the off-line microstore is inhibited, but the main system clock is running. Microinstructions are loaded into the off-line microinstruction register by the on-line processor. In general, the sequence of instructions provided by the on-line processor will test a particular function of the off-line machine and then cause a result to be returned to the on-line where it can be checked.

TABLE I

DIAGNOSTIC TEST SEQUENCE OF 2B PROCESSOR

SYMBOLIC NAME	TEST NUMBER	DESCRIPTION
MCH	1	Maintenance Channel
BUS	2	Gating Bus
CLOCK	3	System Clock
INITIAL	4	Verification of Initialization by Monitor
TO	5	TO Decoder
FROM	6	FROM Decoder
MULFRXP	7	Test For Multiple Firing From Crosspoints
MULTOXP	8	Test For Multiple Firing To Crosspoints
GATEGEN	9	General Registers Gating
GATESP	10	Special Registers
MICMEM	11	Microstore Content (common system)
LOADFR	12	Function Register
ADDER	13	Add Function
DMLCMP	14	Matcher For Duplicate DML
BOOLEAN	15	Boolean Logic Functions
FLZ	16	Find Low Zero Function
ROTATE	17	Rotate Function
PACK	18	Pack and Unpack Gating Operations
MICSEQ	19	Micro Control Part 1
MICSEQ2	20	Micro Control Part 2
MISCEQ3	21	Micro Control Part 3
MICSEQ4	22	Micro Control Part 4
DSFLOP	23	DS Flip-Flop
PAPLUS1	24	Address Increment Adder
MAINSEQ	25	Store Bus Controller Part 1
MNSEQ2	26	Store Bus Controller Part 2
	27	
	thru	Main Store Diagnostic Tests Covered in Section
	64	232-309-103
MINTERP	65	Microinterpret Operation
MTMSCXP	66	Test For Multiple Firing Miscellaneous Crosspoints

TABLE I (Cont)

DIAGNOSTIC TEST SEQUENCE OF 2B PROCESSOR

SYMBOLIC NAME	TEST NUMBER	DESCRIPTION
DMLPAR	67	DML Parity Generator
BUSPAR	68	Gating Bus Parity Checker
POXYPAR	69	IB Register X and Y Field Parity Generator and Parity Check
TIMERS	70	Program Timer and Timing Counter
INTRUPT	71	Interrupts
ADMTCH	72	Panel Address Matcher
DTMTCH	73	Panel Data Matcher
IO	74	I/O Channels (serial)
STATUS	75	Status Bits and Initialization
STATUS2	76	Status Bits and Timeouts
DGSSP	77	System Status Panel (See Section 232-309-104)
DSRDG	78	Double Store Read
	79	Spare
	80	Application Microcode
	81	
	thru	2B Input/Output Control Circuit Diagnostic Test Covered in Section 232-309-108
	90	

8.07 ***Type B:*** The on-line permits the off-line processor to run short sequences of code and the on-line checks the results. In this mode, the on-line loads a main store address into the off-line store address register and enables the off-line microstore. The off-line runs sequences of code starting at the supplied address and terminates with a stop operation. The on-line then causes data to be returned over the maintenance channel for checking.

8.08 ***Type C:*** The off-line processor runs code and checks its own results as it proceeds. If a failure occurs, the off-line supplies data describing the failure to the on-line processor over the maintenance channel. In this mode, hardware check circuits in the off-line are prohibited from stopping the machine as they normally would during call processing. Failures are detected as incorrect results for an attempted operation.

8.09 The tests in the diagnostic sequence proceed from Type A to B to C. The reason being tht no portion of the processor can be relied on as a tool for diagnostics until it has itself been checked; therefore, a processor cannot do its own bit testing until the data manipulation logic has been diagnosed. A description of the diagnostic tests of the 3A CC will follow in order according to test sequence.

MAINTENANCE CHANNEL (PR-1C912-01)

8.10 The maintenance channel (MCH) test consists of four segments.

- ***Segment 1:*** MCH trouble may be on-line or off-line; therefore, some of the on-line circuitry is checked first. Gating to and from the MCH buffer (MCHB), MCH command (MCHC) register, and MCHTR are tested.

- ***Segment 2:*** All zeros and ones constants and a worse case of alternating ones and zeros are loaded into the MCH and transmitted to the off-line MCHB with the LDMCHB command and returned to the on-line with the RTNMCHB command. This is effectively a loop-around test of the channel which partially exercises the off-line and on-line MCH control logic. Checks are made for completion of transmission and for correct data.

- ***Segment 3:*** All MCH commands are fired except switch. In each case, there should be no errors and transmission should complete. An invalid code is then tried in the MCHC which should result in a multiple command error.

- ***Segment 4:*** Some of the MCH commands are tested functionally. Those commands which rely on little hardware in the off-line processor are tested here. Other commands are tested in the diagnostic test where they are first needed.

GATING BUS (PR-1C912-01)

8.11 The gating bus test consists of attempting to gate the off-line bus into the MCH. The result is returned to the on-line side. The data bits should be all zeros and the parity bits (PL and PH) should equal zero.

CLOCK (PR-1C912-01)

8.12 The clock test consists of five segments.

- ***Segment 1:*** A maintenance order to examine the clock phase outputs is issued. This order holds the clock to MCH gating path open for one full clock cycle such that if the clock is running, all phases should appear high.

- ***Segment 2:*** The clock reset function is tested. The clock is stopped by the MCH which blocks the oscillator input to the clock. A special MCH function is used which toggles the clock. The reset function should put the clock into phase, P3, independent of what phase the clock was in before the reset was issued.

- ***Segment 3:*** The error checking circuitry on the clock board checks for phases stuck high or low. The check circuit is verified with a series of maintenance states which force stuck clock phases.

- ***Segment 4:*** The clock is stopped with the MCH. The stop operation also resets the clock to P3. The clock is toggled by the MCH one half of a phase at a time and the phase outputs are checked.

- ***Segment 5:*** The maintenance stop clock mode has an analog time-out circuit which should start the clock several ms after it has been stopped. A gross check on this is made by stopping the clock, and returning to call processing for one base level scan. The clock is then checked to see that it is running.

VERIFICATION OF INITIALIZATION BY MONITOR (PR-1C912-01)

8.13 The off-line processor is initialized to a common state before each diagnostic test. This checks the state of the off-line to insure that initialization was successful. The verification consists of five segments.

- ***Segment 1:*** The B register (BR) is the source of all zeros and all ones constants used in initialization. This segment checks the operation of the ZBR (Zero BR) and SBR (Set BR) miscellaneous crosspoints

- ***Segment 2:*** Verifies that general registers were zeroed by initialization routine. Since the IS, IM, HG, CR, and SS are on the same boards as the general registers, they are checked here.

- ***Segment 3:*** Verifies that special registers, PA, SAR, IB, SIR, AK, AI, DK, DI, and SDR were zeroed by initialization routine.

- ***Segment 4:*** Two non-bit-sliced registers have been initialized. The AR should be zero and the MMSR should equal a particular constant.

- ***Segment 5:*** The dedicated gating path from the ER to the MCH and the ER "TO" and "FROM" gating paths are tested.

"TO" DECODER (PR-1C913-01)

8.14 The "TO" field decoder tests consists of three segments.

- ***Segment 1:*** Detects stuck row or column of "TO" field crosspoints.

- ***Segment 2:*** All "TO" field crosspoints which were not tested in segment 1 are tested.

- ***Segment 3:*** A group of incorrect 4/8 codes used in conjunction with two maintenance states will completely check the check circuitry for the "TO" field decoder.

"FROM" DECODER (PR-1C913-A1)

8.15 The "FROM" field decoder tests consists of three segments.

- ***Segment 1:*** Detects stuck row or column of "FROM" field crosspoints.

- ***Segment 2:*** All "FROM" field crosspoints which were not tested in segment 1 are tested.

- ***Segment 3:*** A group of incorrect 4/8 codes used in conjunction with two maintenance states will completely check the check circuitry for the "FROM" field decoder.

TEST FOR MULTIPLE FIRING "FROM" CROSSPOINTS (PR-1C914-01)

8.16 If an open input (stuck high) exists on a gate which does the final stage of decoding, a multiple crosspoint firing will occur, but the M/N checker will not detect it. Registers to be tested are set to all ones with PL and PH equal to one. A series of left "FROM" subfields are fired with the right subfield zero. Then a series of right "FROM" subfields are fired with the left subfield zero. The "TO" field is some register not presently being tested. This register is initially set to all zeros with PL and PH equal to zero. Gating onto the bus should not take place for any of the above cases but if it does, some bit or bits in the "TO" register will be set.

TEST FOR MULTIPLE FIRING "TO" CROSSPOINTS (PR-1C914-01)

8.17 The final stages of TO field crosspoints are tested similarly to the FROM fields in the previous test (8.16). The right TO subfields are fired with the left subfield zero and the left TO subfields are fired with the right subfield zero. The FROM field is the BR which has been set to all ones with PL and PH equal to one. The tested registers are initialized to zero with PL and PH equal to zero. If erroneous gating takes place, some data bit of a register is set or a parity bit is cleared.

GENERAL REGISTERS GATING (PR-1C915-01)

8.18 The general registers gating test consists of two segments.

- ***Segment 1:*** The ability to gate all zeros and all ones into and out of each general register is verified. The HG, IM, and low 16 bits of the CR are included in this test since they are on the same set of bit sliced boards as the general registers. The SS and MS are also on these boards but are not tested here due to the consequences of loading arbitrary constants into these registers. The IS will be tested in the interrupt diagnostic.

- ***Segment 2:*** A dedicated gating path from the duplicated DML to the CR is tested. The 20-bit data path plus parity is checked.

SPECIAL REGISTER GATING (PR-1C915-A1)

8.19 The special regrater gating test consists of three segments.

- ***Segment 1:*** The ability to gate all zeros and ones into and out of each of the following 16-bit special registers is tested:

  (a) SDR

  (b) SIR

  (c) IB

  (d) DK

  (e) DI

  Gating is also verified for the following 20-bit special registers:

  (a) AK

  (b) AI

  (c) SAR

  (d) PA

  (e) CR

  The low 16 bits of the CR were diagnosed previously since these bits are on boards with the general registers. The high bits are on boards with the corresponding bits of other special registers and have not been previously diagnosed. All of these registers are bit sliced.

- ***Segment 2:*** The dedicated path from the SAR to the PA is tested. This path is enabled by the store bus controller which has not been diagnosed yet. A failure here could result from a controller fault.

- ***Segment 3:*** Verifies gating of zeroes and ones from DB. This is a 20-bit register which is not bit sliced.

MICROSTORE CONTENT (PR-1C916-01)

8.20 The microstore content test reads and verifies the contents of the common system microcode.

FUNCTION REGISTER (PR-1C917-01)

8.21 Since the function registers (FR) content cannot be accessed for examination, the content must be verified by examination of the data manipulation logic (DML) output. Since the DML has not been diagnosed at this time, the DML as well as the decoding in the microcontrol which provides gating to the FR is suspect. A series of operations of the DML are required to verify that each bit of the FR can be set and reset.

- ***Segment 1:*** Five DML functions are tested. The first two are boolean operations. The left four bits in the FR control 16 boolean functions. The left four bits of the first two tested are all zeros and all ones. The other bits have a one function per bit significance. The long add, short add, and the add one functions are tested.

- ***Segment 2:*** The find low zero function is tested with two operands containing low zeros in bit 0 and bit 15.

- ***Segment 3:*** Segment 1 is repeated with the FR being loaded via the dedicated maintenance path from the return address register (RAR)

- ***Segment 4:*** Segment 2 is repeated with the FR being loaded via the dedicated maintenance path from the RAR.

ADD FUNCTION (PR-1C917-01)

8.22 The duplicated adders are tested over the MCH. The add function test consists of three segments.

- ***Segment 1:*** The duplicated adders are checked in the long add mode (20-bit add). Eight sets of operands are applied to DML0 and DML1 and the output of both adders is checked.

- ***Segment 2:*** Proper operation of carry overflow for a 16-bit add operation (short add) and a 20-bit add operation (long add) are verified.

- ***Segment 3:*** The add 1 function is tested.

MATCHER FOR DUPLICATE DML (PR-1C917-01)

8.23 The matcher is checked using the add operation. The test consists of four segments.

- ***Segment 1:*** A mismatch is tried on each of the 20 bits with a one in DML0 and a zero in DML1. The error bit should be set.

- ***Segment 2:*** A mismatch is tried on each of the 20 bits with a zero in DML0 and a one in DML1. These should set the error bit.

- ***Segment 3:*** Matches with all zeros and all ones are tired. These should not set the error bit.

- ***Segment 4:*** The gating paths from AR0 and BR0 to the bus and matcher and from AR1 and BR1 to the matcher are tested to insure that gating out of the AR or BR can take place independent of the content of the other and the DML output.

BOOLEAN LOGIC FUNCTIONS (PR-1C917-01)

8.24 All 16 boolean functions of two variables are checked. Four sets of operands are used to test each function.

FIND LOW ZERO FUNCTION (PR-1C917-01)

8.25 This test consists of three segments which verify the operation of the find low zero test.

- ***Segment 1:*** The operation of the find low zero test is checked with 16 operands such that the low zero appears in each bit position.

- ***Segment 2:*** The operation of the find low zero circuitry is checked to make sure that the all ones case is recognized properly. Test data consists of 16 operands each of which contains of all but one, plus an all ones operand.

- ***Segment 3:*** When the function register has the add long and add short bits set simultaneously, a special function is invoked. Firing the TFLZ crosspoint tests the AR for B(00000001XXXXXXXX) where X is a "don't" care. If such a constant is present, the DS is set. Otherwise it is cleared.

ROTATE FUNCTION (PR-1C917-01)

8.26 The rotate function is tested using two segments.

- ***Segment 1:*** A one is rotated through a field of zeros as follows. The one is placed in bit position 15 and a series of rotate functions are performed which exercise all rotate crosspoints (gate AR to BR by 0, 4, 8, 12 and gate BR to AR by 0, 1, 2, 3) with a net result of rotation right by 31-bit positions. The one should now be in bit position 0. The process is repeated leaving the one in bit position 1. This is continued sixteen times in total whereupon the one should again be in position 15. This exercises all rotate paths for each bit position.

- ***Segment 2:*** A zero is rotated through a field of ones in the same fashion as segment 1.

PACK AND UNPACK GATING OPERATIONS (PR-1C917-01)

8.27 Various commands which do packing and unpacking in the data manipulation use nine special crosspoints involving the A and B registers. These seven crosspoints are functionally tested using nine segments.

- ***Segment 1:*** The ARH4 "TO" crosspoint is tested. The function of this crosspoint is to gate PGB(3-0) to AR(19-16); no parity bits are changed.

- ***Segment 2:*** The ARL16 "TO" crosspoint is tested. The function of the crosspoint is to gate PGB(15-0) to AR(15-0); gate PL to PL; gate PH to PH; bits (19-16) are unchanged.

- ***Segment 3:*** The ARH4 "FROM" crosspoint is tested. The function of this crosspoint is to gate AR(16-19) to PGB(3-0); gate PH4 to PL; gate 1 to PH where PH4 is the parity over the high four bits.

- ***Segment 4:*** The ARL16 "FROM" crosspoint is tested. The function of this crosspoint is to gate AR(15-0) to PGB(15-0); gate PL to PL; gate (PH"+PH4)(PH+PH4") to PH.

- ***Segment 5:*** The ARL20 "FROM" crosspoint is tested. The function of this crosspoint is to gate AR(19-0) to PGB(19-0); gate PL to PL; gate (PH"+PH4)(PH+PH4") to PH.

- ***Segment 6:*** The BRL4 "TO" crosspoint is tested. The function of this crosspoint is to gate PGB(3-0) to BR(3-0)

- ***Segment 7:*** The BRL8 "TO" crosspoint is tested. The function of this crosspoint is to gate PGB(0-7) to BR(0-7); gate PL to PL.

- ***Segment 8:*** The ARL2 "TO" crosspoint is tested. The function of this crosspoint is to gate PGB(1-0) to BR(1-0).

- ***Segment 9:*** The crosspoint TBR15 causes BR(19-16) to be set if BR bit 15 is set.

- ***Segment 10:*** The DML parity generator is verified using eight operands. The operands are placed in the AR and a 20-bit add is performed with the BR containing all zeros.

MICROCONTROL PART 1 (PR-1C919-01)

8.28 The microcontrol part 1 diagnostic consists of four segments.

- ***Segment 1:*** A match circuit compares the content of the MAR and the RAR bits (11-1). This is done whenever a next address is gated from the MIR to MAR gating path. The matcher is checked by loading and freezing a constant in the MAR and loading a series of constants into the RAR which mismatch in only one bit position. Mismatches with the MAR equal to one and the RAR equal to zero as well as mismatches with the MAR equal to zero and the RAR equal to one are tried. Finally, match conditions with all zeros and all ones are tried. The MIR address field, the RAR, and the MAR are partially checked by this test.

- ***Segment 2:*** The ability to zero the RU flip-flop by a miscellaneous decoder crosspoint and set it with the all ones address is verified. The functions of the RU are checked. Specifically, the RU when zeroed should prevent the RAR from being cleared every microcycle and block the matcher which compares the RAR and MAR.

- ***Segment 3:*** The PGB all zeros checker is used in conjunction with the DML but it physically resides in the microcontrol. It is activated whenever the DML "FROM" field crosspoint is fired. Twenty operands are tried each containing a single one. These should result in the DS equal to zero. An all zeros operand is tried which should result in DS equal to one.

- ***Segment 4:*** The 1/16 decoders on the X and Y fields of the IB are tested. All 16 possible outputs are generated for each. The parity generator which creates parity high and parity low for the 1/16 code is also checked.

MICROCONTROL PART 2 (PR-1C919-A1)

8.29 The microcontrol part 2 diagnostic consists of three segments.

- ***Segment 1:*** The presence of all ones in the NA field indicates the return from a subroutine in the microprogram. The ones detector is duplicated. The duplicate RU bits are set by the duplicate detectors, one by each detector. One detector is also responsible for gating the RAR contents to the MAR. A test word of microcode is used to check the ones detectors. This word contains a known NA field. The address of this word is placed in the RAR. A NA containing all but one 1 is loaded into the MIR high bits with the MCH load MAR instruction. Since this address does not contain all ones, the RAR contents should not be gated to the MAR and the known fields should not appear at the microstore output. Also the RU should not be set. This test is repeated for all NAs containing a single zero. Finally the all ones address is tested.

- ***Segment 2:*** The gating path for RAR to MAR will be tested with an all zeros constant and an all ones constant. Since the MAR cannot be examined to verify the gating the test will be performed as follows.

  (a) Load RAR with the desired constant

(b) Put processor in micro loop on word zero

(c) Load all ones address into the MIR high bits, causing the RAR to MAR gating.

(d) Matcher on RAR and MAR will indicate a mismatch if proper gating was not accomplished.

- ***Segment 3:*** Six special crosspoints are used for register gating. These crosspoints have the following values:

NAME	TO	FROM
RX	01000000	11000000
RY	11000000	01000000
RSX	10000000	
RSY		10000000

All of these crosspoints cause the output of 3/6 decoders on the IB X and Y fields to be OR'd into the low six bits of the "TO" or "FROM" field of the MIR. In the case of the crosspoints whose values have B(11) in their upper two bits, the upper "1" is dropped for loading the MIR. The RX and RY crosspoints result in gating to and from the general registers. The RSX crosspoints are used to gate to or from special registers. Gating operations will be performed in order to test the crosspoints as well as the 3/6 decoders.

- ***Segment 4:*** The CA and CB bits in the MIR control several functions. These are:

CA	CB	FUNCTION
0	0	(Null)
0	1	Fetch a new OP code from memory
1	0	Use next address field as a data word
1	1	Control auxiliary function

The fetch and control functions are tested functionally. The data function will be tested in a later diagnostic. When the control function is specified the upper 4 bits of the next address field are used as a 2/4 code to specify the particular task. One of these is loading of the low eight bits of the MIR into the FR. This task will be tested here. Others will be tested in a later diagnostic.

- ***Segment 5:*** A parity generator verifies correct loading of the function register from the MIR. The parity should agree with the parity over the next address field specified in the MIR. Eight operands are used to exercise the parity generator. Two additional operands provide mismatches between the generated parity and the parity next address.

MICROCONTROL PART 3 (PR-1C919-B1)

8.30 The microcontrol part 3 diagnostic consists of four segments.

- ***Segment 1:*** A new OP code is gated from the SIR to the IB, MAR, and RAR when the following conditions exist:

(a) No interrupts are present

(b) The DR is set

(c) The all zeros next micro address is encountered.

The above are set up and gating is tried on two operands, all ones and all zeros. This checks the output gates on the SIR and the input gates of the IB.

- ***Segment 2:*** The presence of all zeros in the NA field indicates to the microcontrol that the next OP should be executed. The zeros detector is duplicated. One copy gates the new OP from the SIR to IB. The other gates the new OP to the MAR of the microcontrol. The copy which gates to the MAR is checked here. The LR OP code is used for this test. The low 16 bits of the first microinstruction contains octal 031700. The OP code is placed in the SIR. An address containing all but one zero is loaded

into the MIR high bits with the MCH "load MAR" instruction. Since this address does not contain all zeros, the new OP code should not be gated to the MAR, and the microstore low bits should not read octal 031700. This process is repeated for all next addresses containing a single one. Finally the all zeroes next address is tried.

- ***Segment 3:*** The gating path SIR to MAR is checked with two operands. Gating takes place when the next address is zero if the DR is set and there are no interrupts present. The RU is put in the update mode which prevents the RAR from being cleared every cycle. The RAR is then loaded with an operand and the RAR-MAR matcher is used to verify gating.

- ***Segment 4:*** Various gating paths have been exercised in the microcontrol. It is also necessary to check that no enables are stuck for these paths either on a single bit or on a whole register. Specifically, the gating paths involved are:

  SIR==>IB

  SIR==>MAR

  IB X or Y fields==>MIR

  Various states are set up which should not allow gating and the destination register is checked.

MICROCONTROL PART 4 (PR-1C919-C1)

8.31 The microcontrol part 4 diagnostic consists of two segments.

- ***Segment 1:*** The parity circuitry is tested. In general, this circuitry buffers the PNA bit of a microinstruction and compares it to the PTA bit of the next microinstruction. The buffering is done in the MARP flip-flop which represents the next parity bit for the MAR. Various factors can alter the above pattern. When a new OP code is loaded from the SIR, the parity of the OP code is buffered. When a return address is from the RAR is used, the parity of this address is buffered. Under certain conditions, the MARP flip-flop is toggled instead of gated into. This is done when the bits CA, CB equal 11, or when the " FROM" field crosspoints 'MIRL8' or 'MIRH12' are fired. When CA, CB equals 01 or CA, CB equals 10, the complement of PTA is compared with the buffered bit. The above functions are all tested. Use is made of a maintenance state with blocking into the MARP flip-flop and a crosspoint which toggles the flip-flop once. A test point on the D11 input to the flip-flop can be returned to the on-line processor with the MCS register.

- ***Segment 2:*** A series of operations of the microcontrol will be tested functionally. These are data high and low operations, tests on the TR1, TR2, DS, DR, and CF. Test microcode is used for this purpose.

DS FLIP-FLOP (PR-1C920-01)

8.32 The DS flip-flop is set and cleared by a number of functions in the processor. These are checked as follows.

- ***Segment 1:*** The DS is set by the "SDS" crosspoint. The DS is cleared by the "ZDS" crosspoint

- ***Segment 2:*** The DS is actuated in the long add mode by overflow from bit 19 of the adder (set if overflow, clear if no overflow).

- ***Segment 3:*** The DS is actuated in the short add mode by overflow from bit 15 of the adder (set if overflow, cleared if no overflow).

- ***Segment 4:*** The DS is actuated by the "TFLZ" crosspoint if the find low zero state is in the FR (cleared if a zero is found, set if no zero found).

- ***Segment 5:*** The DS is actuated by the "TBRO" crosspoint (set for BR bit 0 equal to 1, cleared for BR bit 0 equal to 0).

- ***Segment 6:*** The DS is actuated by gating the DML on the bus if neither the long add mode or short add mode are set (set for all zeros on bus, cleared for not all zeros).

- ***Segment 7:*** The DS is actuated by the "TINT" crosspoint (set for interrupt present, cleared for no interrupt present).

- ***Segment 8:*** The DS is actuated by the "TCH" crosspoint (set if the I/O channel specified in R9 is idle, cleared if it is busy).

- ***Segment 9:*** The DS is actuated by the "OPFXDS" crosspoint.

ADDRESS INCREMENT ADDER (PR-1C920-01)

8.33 The adder circuit for the PA+1 and gating path for PA+1 to SAR are tested using 22 operands which result in all combinations of PA bit and carry from the next lowest bit for each bit position. The parity predict circuit which generates the SAR PL and PH bits is also checked.

STORE BUS CONTROLLER PART 1 (PR-1C921-01)

8.34 The store bus controller part 1 diagnostic is a partial diagnostic of the 3A CCs main memory control. This diagnostic consists of five segments.

- ***Segment 1:*** The ability to load all bits of the MMS register with all ones and all zeroes is checked. This register contains seven duplicated bits plus two non duplicate bits for a total of 16 and is loaded via the BR.

- ***Segment 2:*** The REQ flip-flop should be set on both sequencer circuit packs whenever either store is to be accessed. The crosspoints, SARD, SARI and STSEQ, should cause the sequencer to operate. Also, the control bits in the MIR (CA,CB=01) should cycle the sequencer. The DR is cleared when the REQ flip-flop is set; one copy of the DR by each sequencer circuit pack. The DR can be used to monitor the setting of the REQ flip-flops.

- ***Segment 3:*** Once the REQ flip-flop is set, the appropriate store sequencer circuit pack(s) attempt to seize the store bus by setting the go and seize flip-flops. On the "my store" circuit pack, several conditions should block setting of these flip-flops. These are the presence of a "GO", "BUSY", or "SREQ" signal on the bus. These conditions are tested. The setting of the update and isolate bits yields a special maintenance state which diverts the "GO" flip-flop on each circuit pack to a store error bit in the ER.

- ***Segment 4:*** Several conditions affect the status of the read/write flip-flop on the main memory sequencer circuit packs. Loading the SDR clears the flip-flop on both sequencer boards (write state). The "SRW" miscellaneous crosspoint should put both read/write flip-flops into the read state.

- ***Segment 5:*** The instruction/data (ID) flip-flop is set and cleared by the various signals which initiate store sequences. Specifically, the "SARD" TO field crosspoint and "STSEQ" miscellaneous crosspoint reset the flip-flop (data state) and the "SARI" TO field crosspoint and MIR control bits (CA,CB=01) set the flip-flop (instruction state). The ID flip-flops on both sequencer boards are set simultaneously. The data state inhibits the SAR to PA gating which is enabled by the sequencer boards in the instruction state. This will be used to check the I/D flip-flop.

STORE BUS CONTROLLER PART 2 (PR-1C921-01)

8.35 The store bus controller part 2 diagnostic is the final portion of the diagnostic for the 3A CCs main memory control.

8.36 The setting of the DR bits by the store sequencer boards is tested. Each sequencer board sets one of the DR bits, depending on the state of the sequencer and on the conditions of both stores. Update, reverse, and read/write are tested. Store complete signals from one or both stores are simulated with maintenance states.

MICROINTERPRET OPERATION (PR-1C930-01)

8.37 The microinterpret operation diagnostic test consists of two segments.

- ***Segment 1:*** A microinterpret sequence consisting of one microinstruction is tried in the off-line 3A CC. This instruction gates the contents of R6 to AK. The result is checked. A failure could indicate a problem in the control circuitry or a problem with

the dedicated gating path from SIR to MIR. If a single bit gating error occurred, there should be an error indication in the ER for either a "TO" or "FROM" field but not both. If the control circuitry failed, both the "TO and "FROM" error bits will be set or neither will be set. The test is then repeated with a microinstruction to gate the contents of AK to R6. The microinstruction is the complement of the one used previously so that the SIR to MIR path is completely checked.

- ***Segment 2:*** A microinstruction sequence of several instructions is tried to check that the IB is not cleared while in the microinterpret mode. The microinterpret instruction should remain in this register throughout the sequence.

TEST FOR MULTIPLE FIRING MISCELLANEOUS CROSSPOINTS (PR-1C930-01)

8.38 This test consists of three segments.

- ***Segment 1:*** Multiple miscellaneous crosspoints can result from opened leads such that a miscellaneous crosspoint fires when only three of its four decoder subfields are present. A set of initial conditions is set up and then a series of crosspoints with one subfield equal to zero is tried. The initial conditions are then checked. These conditions should not have changed.

- ***Segment 2:*** A second set of initial conditions is used as in Segment 1 in order to handle contradictory cases (DS equals 0 in Segment 1 and DS equals 1 in Segment 2).

- ***Segment 3:*** Some crosspoints which were not tested in earlier segments are now tested.

GATING BUS PARITY CHECKER (PR-1C930-01)

8.39 The gating bus parity checker diagnostic test consists of four segments.

- ***Segment 1:*** The bus parity check circuit is tested using a series of operands with some containing good parity and some containing bad parity. Since the parity checker is inhibited when the processor in the stop mode, the off-line 3A CC must run its own code to exercise this circuit. The test constants will be supplied from the on-line 3A CC and the off-line will be made to execute an LR instruction which gates the test constant over the bus. The on-line 3A CC will check the error bit. The parity checker on the low eight bits of the bus is tested first. Eight constants are required and the high 12 bits contain all zeros and the PH equals one in each constant.

- ***Segment 2:*** The parity checker on the high 12 bits of the bus is tested in the same way as the low bits in Segment 1. Eight constants are required. The low eight bits contain zeros and PL equals one in each constant.

- ***Segment 3:*** This portion of the test verifies that the bus parity checker is inhibited under the following conditions.

  (a) A NOP in the "FROM" field

  (b) The processor is stopped

  (c) The block parity check bit in the SS register is set

  (d) A miscellaneous crosspoint is fired

  (e) The SDRC or SIRC "FROM" field crosspoints fire when DR equals zero

  (f) The processor is in the freeze state.

- ***Segment 4:*** The 2/4 code B(1001), which is used in conjunction with the control bits in a microinstruction (CA,CB equal 11), causes a bus parity error to be diverted to another error bit in the ER which does not result in switching of processors. This feature is used in conjunction with I/O instructions such that if a received message contains bad parity due to faulty peripheral equipment, switching will not occur.

IB REGISTER X AND Y FIELD PARITY GENERATOR AND PARITY CHECK (PR-1C930-01)

8.40 This diagnostic consists of three segments.

- ***Segment 1:*** The parity generator on the X field of the IB is used for table indexing in microstore and is tested as follows:

  (a) An address of a test micro word with PTA and PNA equal to zero is frozen in the MAR

  (b) A constant is put in the X field of the IB

  (c) The ER is cleared

  (d) The indexing crosspoint (TX) is fired

  (e) If the constant contains an odd number of ones, the complement of PNA is compared with PTA by the MAR parity circuit and a mismatch results.

- ***Segment 2:*** The parity generator on the IB Y field is tested as the X field was tested in Segment 1

- ***Segment 3:*** A check is made on the parity of the X and Y fields when the IB is loaded either from the bus or from the SIR. The output of the two parity generators which were tested in Segments 1 and 2 are combined and compared with PL. A mismatch sets error bit IBER. This function is checked using all eight combinations of X parity, Y parity, and PL.

PROGRAM TIMER AND TIMING COUNTER (PR-1C931-01)

8.41 The progam timer and timing counter diagnostic test consists of six segments.

- ***Segment 1:*** The TI (includes timing counter and program timer) is stopped with the set block timer check (SBTC) miscellaneous crosspoint for 256 microseconds. Three constants are gated into and back out of the TI (all zeros, all ones, and all zeros). This segment is run in the off-line 3A CC and a flag is left in the MCH buffer for the on-line to interrogate which is zero for success and one for failure.

- ***Segment 2:*** The four bit prescaler is zeroed by the gate BR to timer (BRXTI) crosspoint and then incremented 16 times. Bit zero of the TI should be set on the sixteenth count. This segment is run by the off-line processor in the manner of Segment 1.

- ***Segment 3:*** The combination of the TI equal to B(100) and the prescaler equal to B(0001) creates a 1.25 ms signal which should clear bits (0-3) of the TI, set bit 4 and clear the prescaler. The TI is set to B(100), the prescaler is zeroed and then incremented one time. The TI is examined after the prescaler is incremented. To prove that the prescaler is zeroed, the prescaler test of Segment 2 is run again. This segment is run in the off-line processor.

- ***Segment 4:*** The TI is tested to insure that it counts properly. Sixteen constants are used to show that each bit can be toggled by the next lowest bit. Note that the timer has several nonlinear points corresponding to 1.25 ms and 25 ms.

- ***Segment 5:*** The MCH order which clears the program timer (high eight bits of the TI) is tested. The zero program timer (ZPT) miscellaneous crosspoint is also tested.

- ***Segment 6:*** The SBTC miscellaneous crosspoint should set the block timer check (BTC) bit in the SS register which stops the TI and starts a 256-microsecond backup timer. When the backup timer times out, the BTC bit should be cleared and the TI should start counting.

INTERRUPTS (PR-1C931-01)

8.42 The interrupts diagnostic tests consists of three segments.

- ***Segment 1:*** Each level of interrupt is tested by setting its request bit in the interrupt set (IS) register and firing the TINT miscellaneous crosspoint. Firing of the TINT miscellaneous crosspoint should set both copies of the data manipulation logic status (DS) bit. The corresponding mask bit in the interrupt mask (IM) register is then set and the TINT crosspoint is fired. This should result in clearing both copies of the DS.

- ***Segment 2:*** The block interrupt (BIN) bit in the system status register is tested. This bit should block all interrupts.

- ***Segment 3:*** A functional test of interrupt processing is carried out. If an interrupt is present when the microcontrol is ready to load a new OP code, the address of the interrupt microcode is jammed into the MAR. This is tested with the load MAR(LDMAR) MCH order. If an attempt is made to load the MAR with address 0 using this order and DR bit equal to one and the I bit equal to zero, when an interrupt is present, the address of interrupt microcode should be jammed into the MAR.

PANEL ADDRESS MATCHER (PR-1C931-01)

8.43 The panel address matcher diagnostic test consists of four segments.

- ***Segment 1:*** The ability of the SAR and AI registers to match on all zeros and then on all ones is tested. The AK register is set to all zeros to allow a SAR-AI match test on all bits. The address match enable (AME) bit of the SS register is set to enable the address matcher circuit. The address or data match (ADMI) bit of the IS register is tested for an expected match indication.

- ***Segment 2:*** The ability of the address matcher to detect a SAR-AI register mismatch is tested. The SAR is set to all zeros and a one is stepped through the AI register. The AK register is set to all zeros to allow a SAR-AI match test on all bits. The ADMI bit of the IS register is tested for an expected mismatch indication.

- ***Segment 3:*** The ability of the address matcher to detect a SAR-AI register mismatch is tested with the AI register set to all zeros and by stepping a one through the SAR. The AK register is set to all zeros to allow a SAR-AI match test on all bits. The ADMI bit of the IS register is tested for an expected mismatch indication.

- ***Segment 4:*** The ability of the AK to mask on every bit is tested. The AK and SAR registers are set to all ones for this test. All bits of the AI register are set to ones except the bit which is being tested which is set to a zero. The zero is walked through every bit location of the AI register and a SAR-AI compare test is performed. Since the AK is set to all ones, the SAR-AI mismatches should not be detected. The ADMI bit of the IS register is tested for an expected match indication.

PANEL DATA MATCHER (PR-1C931-01)

8.44 The panel data matcher diagnostic test consists of four segments.

- ***Segment 1:*** The data matcher circuit which matches the DI register with the data on the store bus is tested for match conditions with all ones and all zeros. The ADMI interrupt bit in the IS register should be set in each case.

- ***Segment 2:*** A mismatch on a single bit is tried with a 0 on the store bus and a 1 in the DI. This is repeated for each bit. A mismatch on a single bit is tried with a one on the store bus and a zero in the DI. This is repeated for each bit.

- ***Segment 3:*** The ability to mask a match on data with the DK register is tested. The store bus and DK contain all ones and the DI contains a single zero. The one in the DK should mask the mismatch.

- ***Segment 4:*** The data match enable (DME) bit in the SS register is tested. When the DME is cleared, matching should be inhibited.

I/O CHANNELS (PR-1C931-01)

8.45 The I/O channels diagnostic test consists of three segments.

- ***Segment 1:*** Verifies proper main channel selection for each channel. Verifies that error bit is set when illegal code is used which selects multiple channels.

- ***Segment 2:*** The following operations are tried on all equipped I/O channels:

  (a) Select all subchannels and try illegal subchannel select codes.

(b) Load IOD register with test constants

(c) Return contents of IOD to R11

(d) Load IOS register via MD2CHTM, MD1CHTN, MD3CHC, and MD0IDCH crosspoints

(e) Return contents of IOS to R11

- ***Segment 3:*** The following operations are tested to ensure that one I/O channel can not affect another:

(a) Load IOD register from R10

(b) Load R11 from IOD register

(c) Load IOS register with various crosspoints

(d) Load R11 from IOS register

STATUS BITS AND INITIALIZATION (PR-1C932-01)

8.46 The status bits and initialization diagnostic tests consist of six segments.

- ***Segment 1:*** The status bits halt (HLT) and manual (MAN) control the lights on the 3A CC panel. The bits ISC1 and ISC2 are used as software switches by the microprogram control. The ability to set and clear these bits is tested.

- ***Segment 2:*** The condition flip-flop (CF) should be set by the SCF crosspoint and cleared by the ZCF crosspoint. Also the DSXCF miscellaneous crosspoint places the CF in the state of the DS. There are duplicated copies of the CF in the microcontrol status register and a copy in the system status register for display purposes. All copies are checked.

- ***Segment 3:*** Gating to the display buffer is tested which should take place when the DBC "TO" field crosspoint is fired and only when the display bit in the SS register is set.

- ***Segment 4:*** A switch message over the MCH should cause the operation of an initialization circuit which initializes some status bits and passes control to the initialization microcode. The microcode then transfers to the initialization program. Two test words (INITTST1 and INITTST2) are used which are loaded into call store to indicate to the initialization program to go down a test branch which zeros INITTST1 and halts the processor. This is an indication that the initialization was successful.

- ***Segment 5:*** The setting of error register bits 0-9 in the off-line 3A CC should result in a stop and switch message being sent to the on-line processor. In the on-line machine, this results in setting error bit, SWER. The error bit, SWER, sets interrupt bit ERR1. Setting error bits 10-13 in the off-line processor should result in the initialization of the off-line 3A CC. Setting of error bits 14-19, PL or PH should set bit ERR1 in the interrupt set (IS) register.

- ***Segment 6:*** The block hardware check (BHC) bit in the SS register should prevent error bits 0-9 from causing a stop and switch and prevent error bits 10-13 from causing an initialization.

STATUS BITS AND TIME-OUTS (PR-1C932-01)

8.47 The status bits and time-outs diagnostic test consists of two segments.

- ***Segment 1:*** The program timer has two time-outs. The first should cause a maintenance channel switch message if the CC flip-flop equals one (3A CC on-line) and should cause an initialization if the CC flip-flop equals zero (3A CC off-line). The second time-out should cause an initialization independent of the CC flip-flop. The block timer check (BTC) flip-flop in the SS register should inhibit all switch and initialization functions since the input and output of the program timer is blocked.

- ***Segment 2:*** When the CC flip-flop is set, MCH orders MSTART, MSTOP, and LDMAR should be blocked. Also the CLPT order should result in setting error bit PTRER (PT reset received by on-line CC) and the SWITCH order should result in setting error bit SWER (switch received by on-line CC).

- ***Segment 3:*** Test branch allowed (BA) bit.

- ***Segment 4:*** Off-line sanity test gross check of off-line machine

- ***Segment 5:*** Two auxillary control 2/4 checks are tested here. They are "turn off bus parity check" and "I/O DML match divert." These are tested functionally by executing an I/O command with a bad code.

DOUBLE STORE READ (PR-1C911-01)

8.48 The double store read test consists of the following diagnostic segments.

- ***Segment 1:*** This segment tests the gating paths, crosspoints and registers used for storing data on a double store read operation.

- ***Segment 2:*** A functional test of the double store read is performed using three different cases. With bad parity in the on-line store and good parity in the off-line store, and the block double store read bits of the main memory status register set, no read of the off-line should take place. If the block double store read bits are then cleared, a read of the off-line store should take place. When both stores contain good parity, no read of the off-line store should take place.

9. GLOSSARY

AI—address input (register)

AK—address mask (register)

AR—operand register A

BA—branch allow

BDSR—block double store read

BEC—block error recovery (flip-flop)

BHC—block hardware check (flip-flop)

BIN—block interrupt (flip-flop)

BR—operand register B

BSA—branch save address

BSAI—branch save address indirect

BTC—block timer check

C—clear

CA—control bit A

CB—control bit B

CF—condition flip-flop

CR—C register

CS—channel select

CW—complement write

DI—data input (register)

DK—data mask (register)

DML—data manipulation logic

DR—data ready (flip-flop)

DS—data manipulation logic status

DSR—double store read

ER—error register

ERAR—error return address register

ERU—error return address update

FR—function register

HG—hold get (register)

HIC—hybrid integrated circuit

IB—instruction buffer

ID—instruction or data (flip-flop)

IDL—idle (flip-flop)

IFF—I flip-flop

I/O—input/output

IOD—I/O data (register)

IOS—I/O status (register)

ISO—isolate (flip-flop)

LED—light-emitting diode

LOF—lock off-line

LON—lock on-line

MAR—microaddress register

MARP—MAR parity

MAS—main store

MCH—maintenance channel

MCHB—MCH buffer

MCHC—MCH command (register)

MCHTR—MCH transmit/receive (register)

MCS—microcontrol status (register)

MIR—microinstruction register

MMS—main memory status (register)

MRF—maintenance reset function

MSR—maintenance state register

NA—next address

NAND—not-and (gate)

ns—nanosecond

OP—operation (code)

OPF—OP code fil (bit)

PA—program address

PA+1—program address plus 1 (logic)

PH—parity high

PL—parity low

PNA—parity bit of next address

PROM—programmable read only memory

PT—program timer

PTA—parity bit for current address

RAR—return address register

REQ—request (flip-flop)

RI—register and immediate data

RN—immediate operand to register

RR—register to register

RU—RAR update (flip-flop)

R/W—read/write (flip-flop)

RXN—references memory by adding N to an address register pair

RXR—references memory by adding an index register to an address register

S—set

SAR—store address register

S-A-0—stuck-at-zero

S-A-1—stuck-at-one

SDR0—store data register zero

SDR1—store data register one

SERC—store error C

SIB—store instruction buffer

SIC—silicon integrated circuit

SIR0—store instruction register zero

SIR1—store instruction register one

SL—specifies 20-bits to reference memory and sometimes load a register pair

SOP—set OPF (bit)

SOS—store-out-of-sync (flip-flop)

SS—system status register

SSP—system status panel

TA—transfer allow

TC—timing counter

TDC—tape data controller

TR1—test register 1 (flip-flop)

TR2—test register 2 (flip-flop)

TTY—teletypewriter

T2L—transistor-transistor logic

UPD—update (flip-flop)

ZOP—zero OPF bit