PCP-175

GE-PAC* 30
CONTROL COMPUTER

MAINTENANCE MANUAL

VOLUME 1

GENERAL ELECTRIC

*Registered Trademark of General Electric Company

GE-PAC 30
CONTROL COMPUTER

MAINTENANCE MANUAL

VOLUME 1

GENERAL ELECTRIC

30-1 GE 29-026R01
30-2 GE 29-034R01

Figure 1-1. Typical GE-PAC 30 Digital System

GENERAL DESCRIPTION

CONTENTS

ILLUSTRATIONS

GENERAL DESCRIPTION

1. INTRODUCTION

Figure 1 shows a typical GE-PAC* 30 Digital System. The GE-PAC 30 is a general purpose, low cost system, versatile enough to perform a wide range of both industrial control and scientific computation. These third generation systems use integrated circuits for reliability, and feature a modular expansion system which permits an economical approach to the specific requirements of each installation.

A high speed, 16-bit halfword memory is addressable and alterable to the 8-bit byte level. Memory is field expandable to 65, 536 bytes. All memory is directly addressable with the primary instruction word, no paging or indirect addressing is required.

Sixteen General Registers, each 16-bits long, can be used as Index Registers or accumulators.

Register-to-register instructions permit operations between the 16 General Registers, eliminating redundant loads and stores.

A comprehensive instruction set includes efficient byte processing instructions, single instructions for loop control to increment-test-branch on index values, as well as instructions that test the condition code and branch directly to any location in memory.

Logical and arithmetic shift instructions can shift up to 15 positions with a single instructions.

Flexible, multiplexed input/output systems include an integrated priority interrupt facility and provide for direct addressing of up to 256 devices.

High speed memory access channels permit cycle stealing input/output to byte oriented peripherals or 16-bit halfword special purpose devices.

*Registered Trademark of General Electric Company

2. SCOPE OF MANUAL

This two volume maintenance manual is planned to enable an experienced digital technician to learn the theory of operation and maintenance of the GE-PAC 30 Digital System. This Volume, Volume 1, provides the theory and maintenance text. Volume 2 provides installation data and drawings. This Section of this Volume provides an introduction to both the system and to the documentation. The second Section describes the Processor. The Display System, including operating information is decribed next. The fourth Section describes the Input/Output (I/O) system. The core memory is described next. The Read-Only-Memory (ROM) is described next, followed by the Power System. General References are provided in the last Section.

Other GE-PAC 30 publications which may be of interest to readers of this manual are listed in Table 1.

3. SYSTEM ORGANIZATION

This Section describes the organization of a typical GE-PAC 30 Digital System from both the physical, and the functional standpoints.

3.1 Physical Organization

GE-PAC 30 Digital Systems employ a unique mechanical layout and wiring configuration which simplifies expansion of the system. Expansion consists of simply plugging in additional logic boards; no back plane wiring additions are normally required.

A GE-PAC 30 Digital System Consists of a basic card file which may be mounted in a standard 19" RETMA rack, and additional expansion card files mount up to 25 9.5" X 10.5" circuit boards designated mother-boards. Two types of mother-boards are available: the standard copper mother-board and a wire-wrap mother-board. The copper mother-boards require one slot in the card file. Copper mother-boards provide a system in which most components

TABLE 1. RELATED PUBLICATIONS

Title	Publication Number
GE-PAC 30-1 Digital System (C) Maintenance Manual-Volume 2-Drawings	29-027
GE-PAC 30-2 Digital System (C) Maintenance Manual-Volume 2-Drawings	29-035
System Interface Manual	29-003
Reference Manual	29-004
Logic Module Handbook	29-005
Programming Manual 30-1	29-013
Programming Manual 30-2	29-013R01
Micro-Instruction Reference Manual 30-1	29-017
Micro-Instruction Reference Manual 30-2	29-032

are soldered directly to the mother-board. Wire-wrap mother-boards are used for special applications, and in areas where the number of mother-boards used does not warrant generation of a copper mother-board. Because of the wire-wrap pins on the back of these mother-boards, two slots in the card file are required for the wire-wrap mother-board. Each wire-wrap mother-board may mount up to 40 smaller component boards designated daughter-boards. Daughter-boards plug into mother-boards via a set of 16 pins. The mother-board is divided into 40 fields as shown on Figure 2. Figure 3 illustrates a card file, mother-boards, and daughter-boards. A series of daughter-boards which provides a variety of standard logic functions is available from GE-PAC 30. Mother-boards with provisions for mounting potentiometers, relays, indicator lamps, capacitors, and resistors are also available. These general purpose components are described in the Logic Module Handbook, GE-PAC 30 Publication Number 29-005.

Note on Figure 3 that there are three sizes of daughter-boards. Figure 4 illustrates the size and pin designations for each of the three daughter-board sizes.

Each mother-board may have two 69-pin connectors. The back panel of a card file is shown on Figure 5. Note that the connectors are numbered from left to right on the wiring side. The lower row of connectors is designated Field 0; the upper row is designated Field 1. A strip power bus

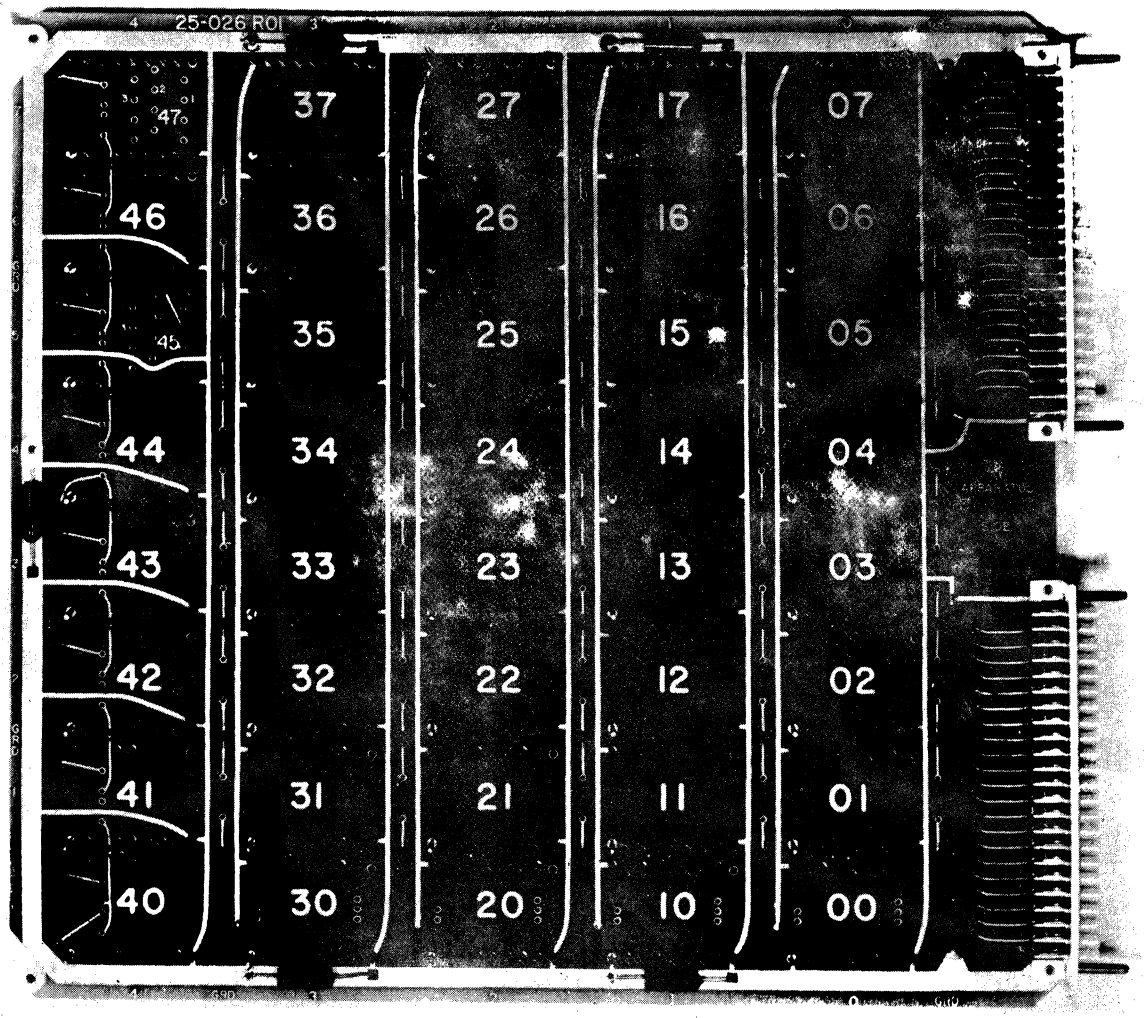


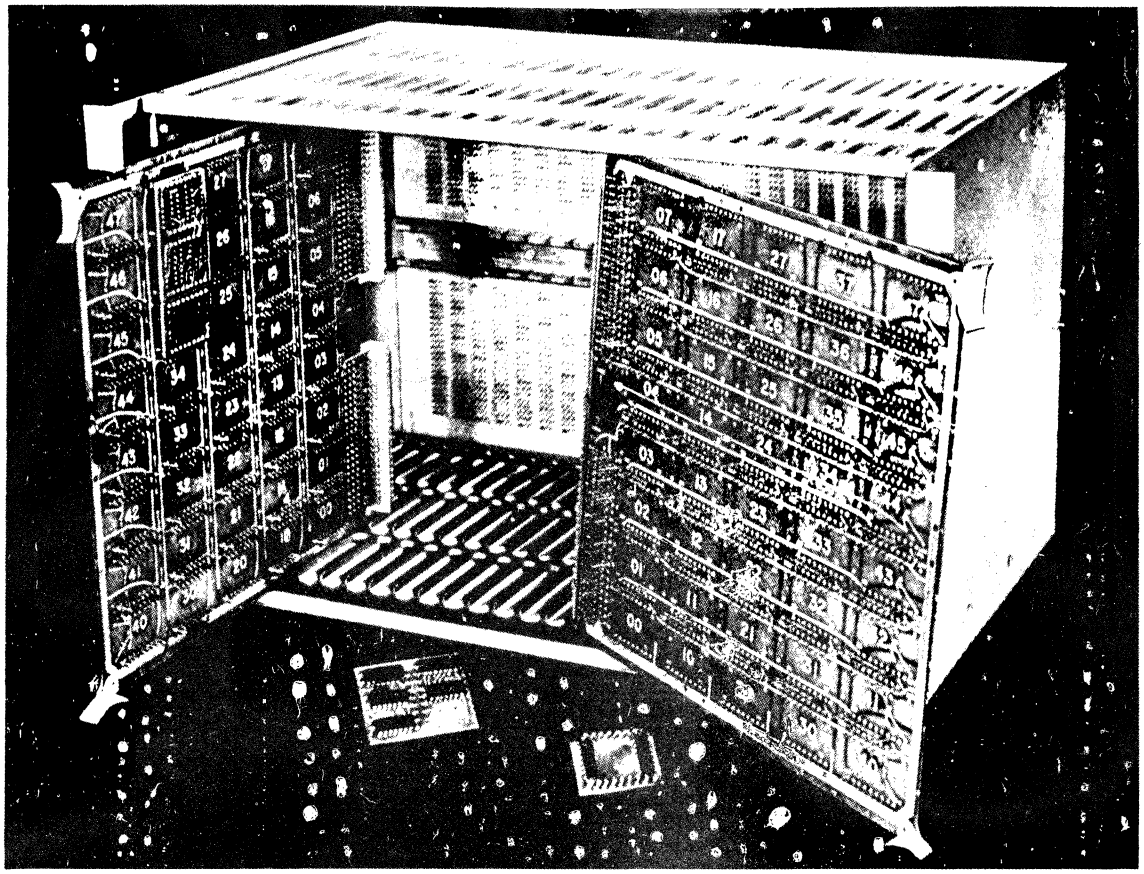
Figure 2 Mother-Board Layout

is provided between the two fields. Figure 6 shows the pin numbering system. The first digit of the pin number specifies the column number, from 0 to 2. The second two digits specify the row number, from 00 to 22. The dash number at the end specifies the connector field.

Wiring between card files is via plug in cables which mate with the wiring side of the mother-board connectors. Wiring between a card file and an external device is via a plug in cable which mates with one set of the daughter-board connectors on a mother-board. Figure 7 illustrates the interconnections between card files and external devices.

3.2 Functional Organization

GE-PAC 30 Digital Systems are modularly structured to provide a high degree of flexibility in configuring application oriented systems. The basic building blocks used in the organization of a system are the Processor, Memory Modules, a Read-Only-Memory (ROM) Control Unit, and the Input/Output (I/O) Controllers which interface to peripheral devices. The various elements of an GE-PAC 30 Digital System are organized around the primary controlling unit - the Processor. See Figure 8.



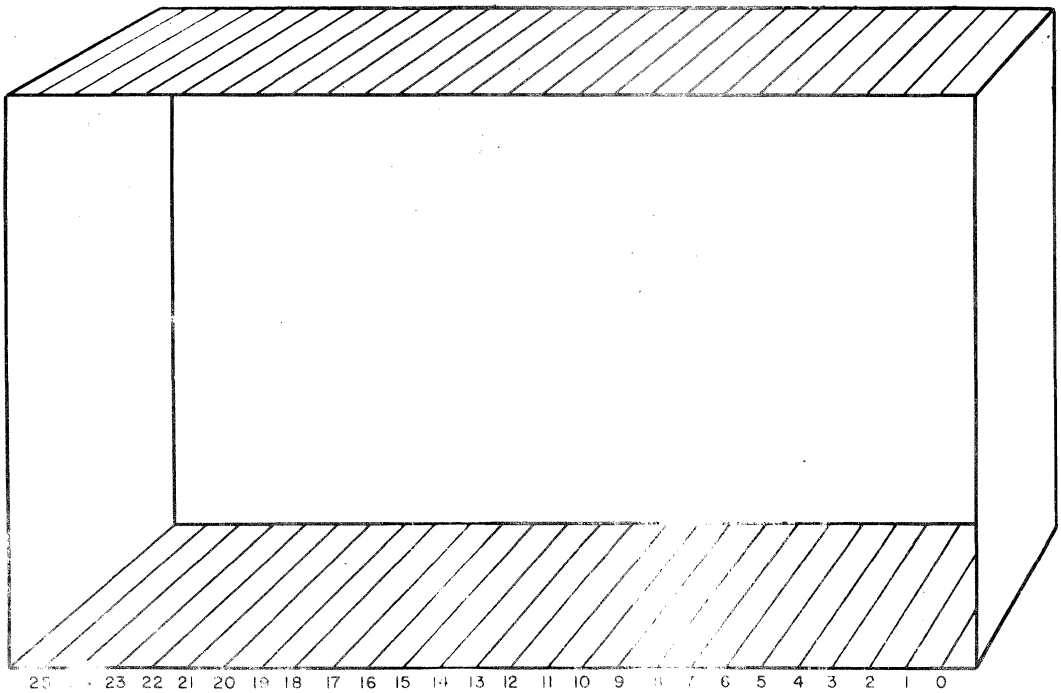


Figure 3 Typical GE-PAC 30 Rack, [illegible] Quarter Front View

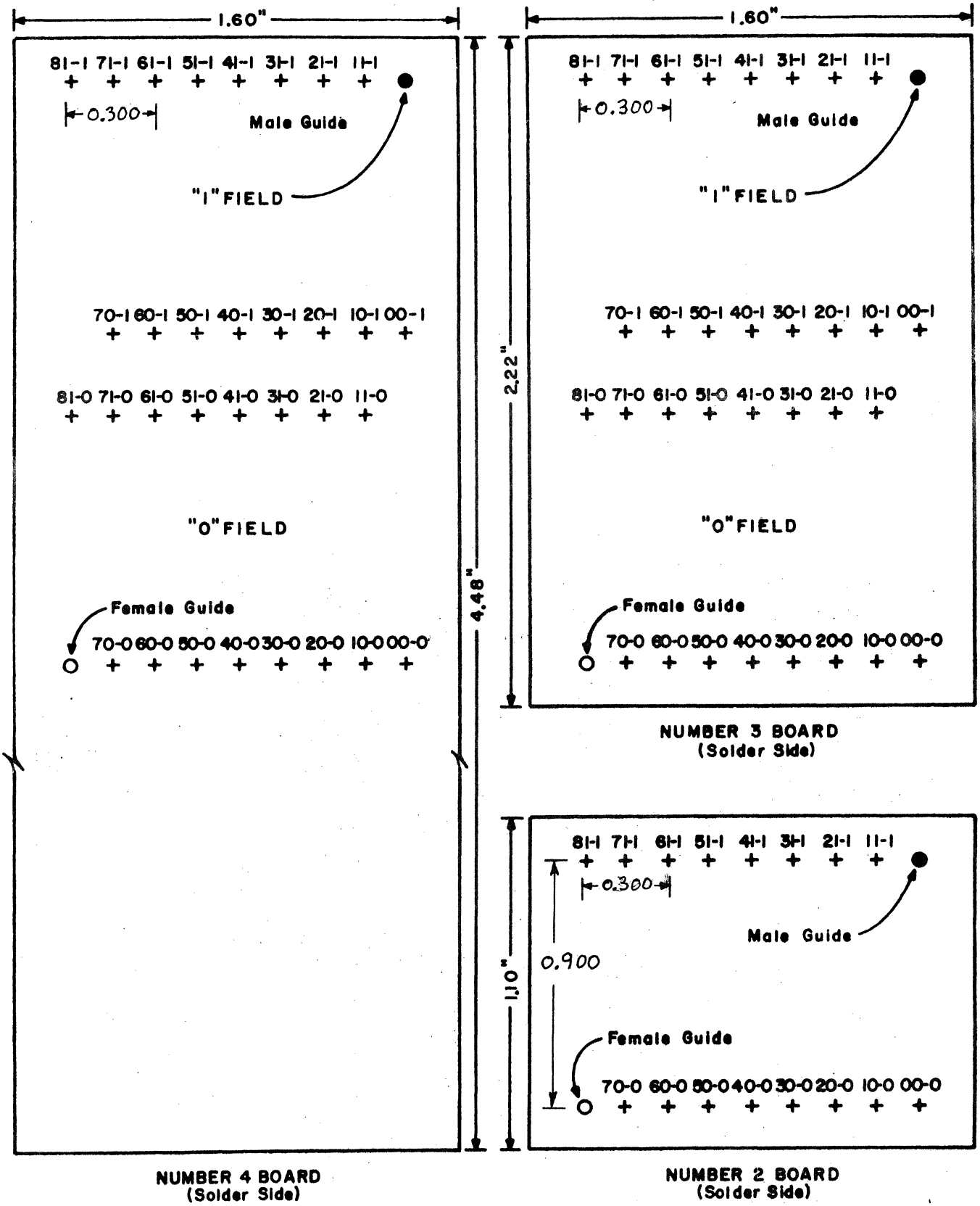


Figure 4 Daughter-Board Layout

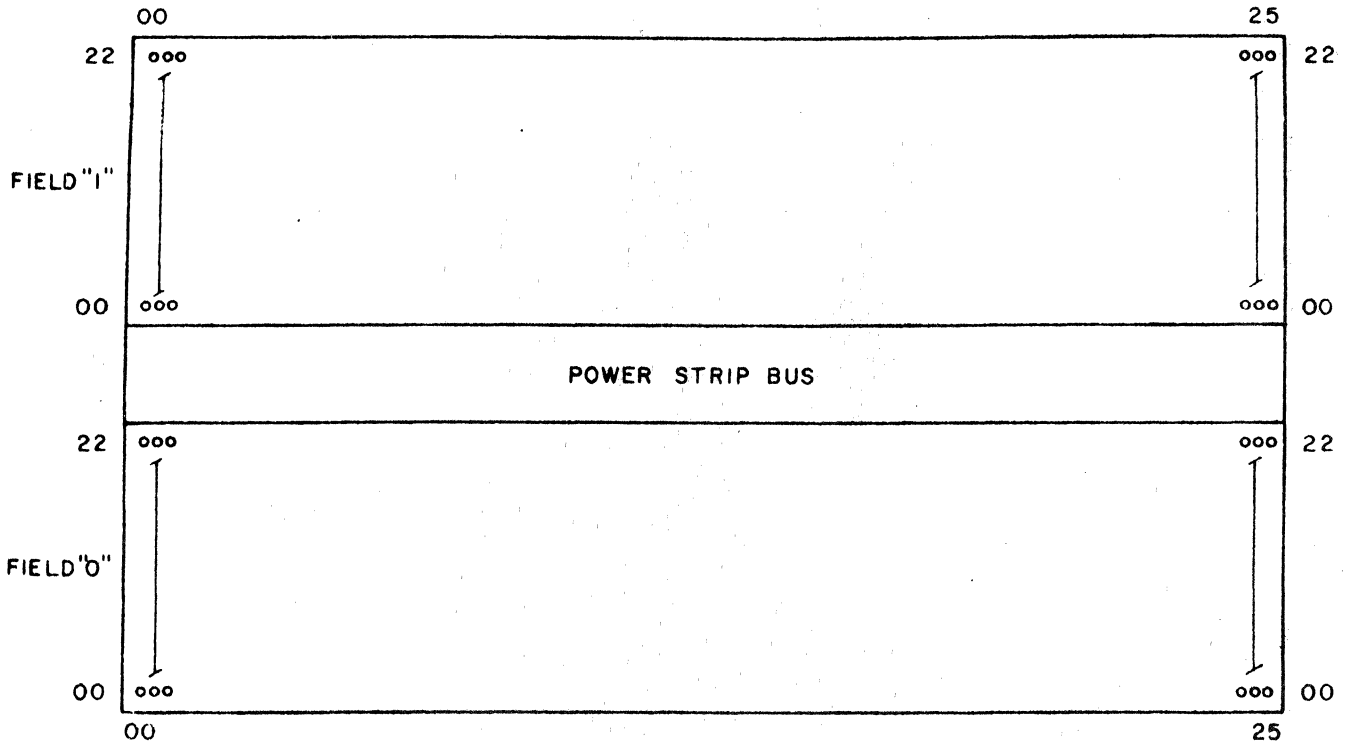


Figure 5 Typical GE-PAC 30 Rack, Back Panel Layout

The Processor contains facilities for:

1. Arithmetic and logic processing of data.
2. Sequencing instructions in the required order.
3. Addressing memory.
4. Fetching and storing information.
5. Initiating and controlling communications with external devices.
6. Taking action in response to interrupts.

These operations are performed as directed by instructions in the user's program.

The Processor and ROM provide a combination of hardware and firmware for decoding and executing user's instructions. GE-PAC 30 Processors are designed to execute ten basic micro-instructions. These micro-instructions are programmed into subroutines that are permanently wired into a Read-Only-Memory (ROM). Each subroutine (or firmware) micro-program in the ROM executes a simple operation. Combinations of these subroutines perform the more complex operations that make up each user's instruction. (See the Reference Manual GE-PAC 30 Publication Number 29-004 for detailed descriptions of user's instructions.)

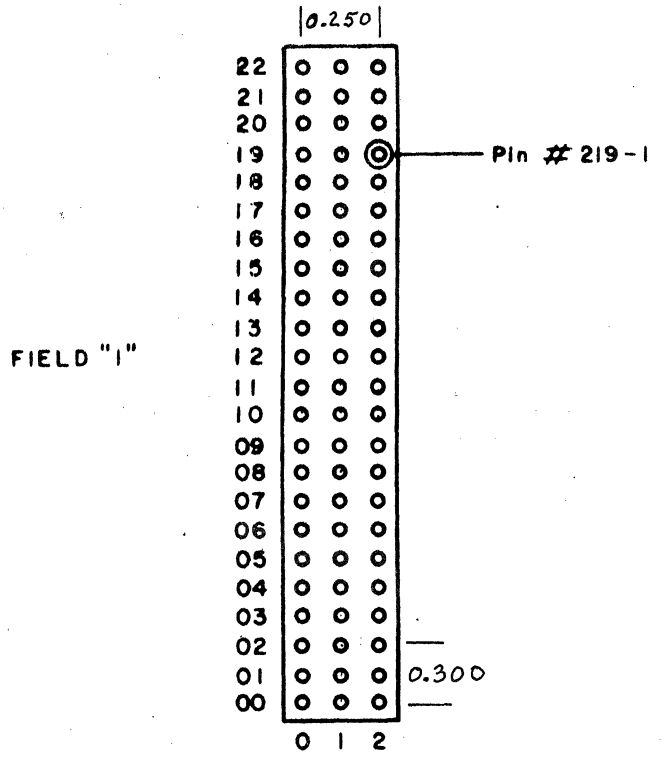


Figure 6 Mother-Board Connector Layout

The ten basic micro-instructions the GE-PAC 30 Processor is hardware wired to execute are:

1.	Add	(A)
2.	Subtract	(S)
3.	Exclusive OR	(X)
4.	AND	(N)
5.	Inclusive OR	(O)
6.	Load	(L)
7.	Command	(C)
8.	Test	(T)
9.	Branch On Condition	(B)
10.	Do	(D)

(See the Micro-Instruction Reference Manual for a detailed description of micr-instructions.)

The Processor has two-way communication channels with the Input/Output controllers and core memory system. These communication channels and the Processor internal registers employ an eight bit word length. Data is transferred within the Processor via eight bit B and S buses. The Processor performs micro-instructions by gating information from one location to another internally, or by sending signals to the units it controls.

Basically, the Processor executes successive micro-instructions read from the ROM. Together, these micro-instructions make up a micro-program which reads a user instruction out of core memory, decodes it, executes it, and then repeats the process for the next user instruction.

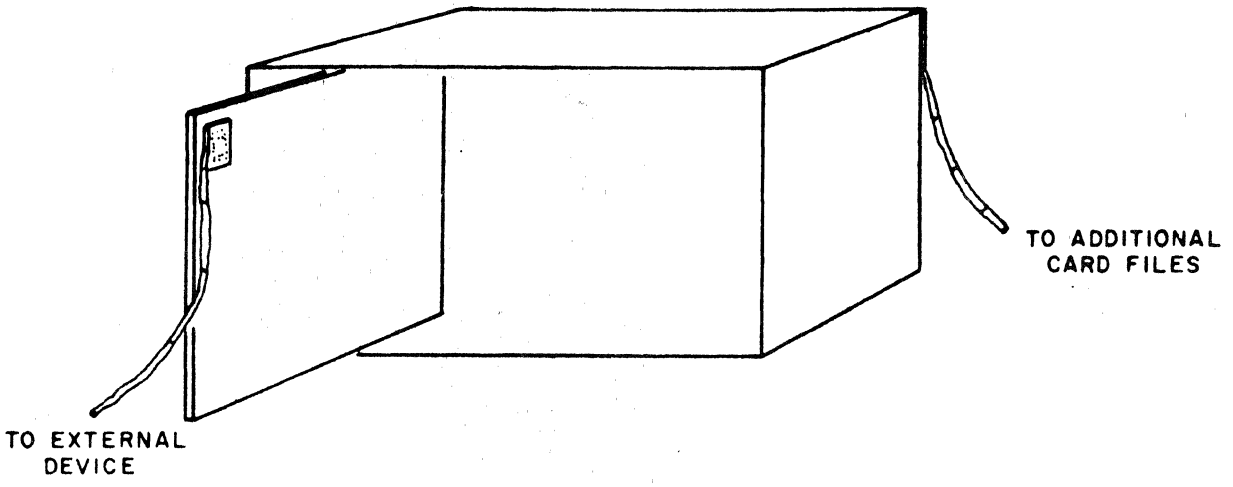


Figure 7 Typical GE-PAC 30 Rack, Interface Cable Layout

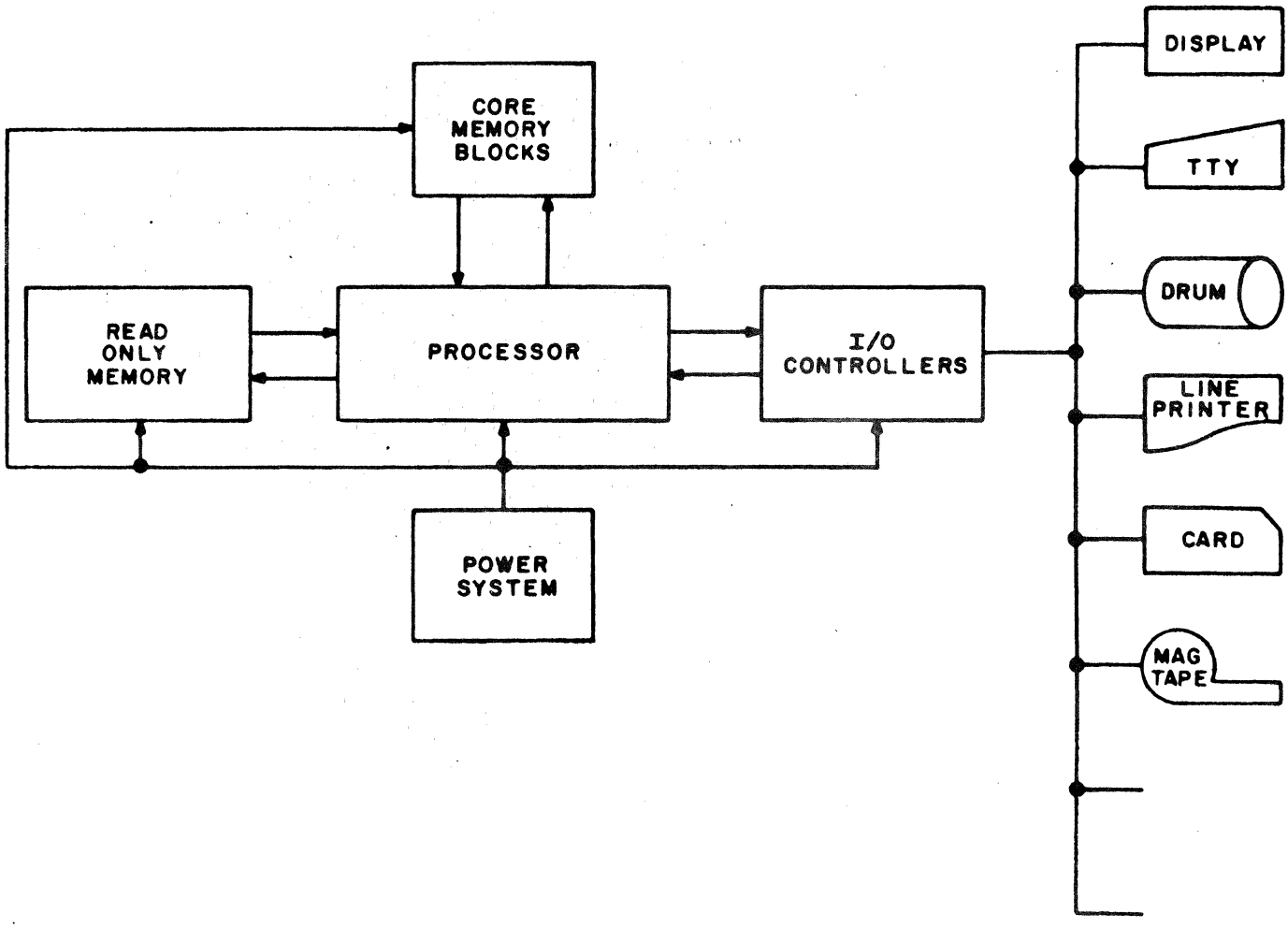


Figure 8 GE-PAC 30 Digital System Basic Block Diagram

4. DOCUMENTATION

This Section describes the style and conventions used with GE-PAC 30 documentation.

4.1 Number Notation

The most common form of number notation used in GE-PAC 30 documentation is hexadecimal notation. In this system, groups of four binary digits are represented by a single hexadecimal digit. Table 2 lists the hexadecimal characters employed.

To differentiate between decimal and hexadecimal numbers, hexadecimal numbers are preceded by the letter "X", and the number is enclosed in single quotation marks. Examples of hexadecimal numbers are: X'1234', X'5EC6', X'A340', X'EEFA', and X'10B9'.

4.2 Part Numbering System

GE-PAC 30 parts, drawings, and publications employ a common numbering system. The part number and drawing numbers for drawings which describe the part are related. The publication number is also often related to the part number of the device or program described. Figure 9 shows the format used for GE-PAC 30 part numbers. The fields are described in the following paragraphs.

4.2.1 Category Field. The 2-digit Category number indicates the broad class or category to which a part belongs. Typical examples of category number assignments are:

01 - Basic Hardware Systems

02 - Basic Hardware Expansions

03 - Basic Software Systems

04 - Basic Software Expansions

05 - Major Application Programs

06 - Self-contained Utility Programs

07 - Subroutines of General Utility

10 - Spare Parts Packages

12 - Card File Assemblies

13 - Panels

17 - Wire and Cables

19 - Integrated Circuits

20 - Transistors

27 - Peripheral Equipment

29 - Manuals

TABLE 2. HEXADECIMAL NOTATION DATA

Binary	Decimal	Hexadecimal	Binary	Decimal	Hexadecimal	Binary	Decimal	Hexadecimal
0000	0	0	0110	6	6	1100	12	C
0001	1	1	0111	7	7	1101	13	D
0010	2	2	1000	8	8	1110	14	E
0011	3	3	1001	9	9	1111	15	F
0100	4	4	1010	10	A			
0101	5	5	1011	11	B			

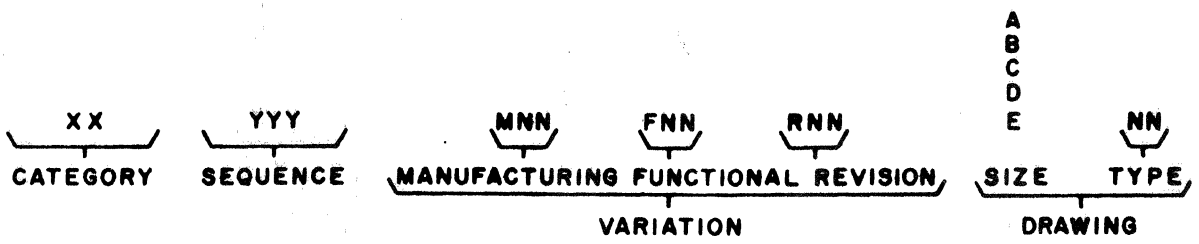


Figure 9 Part Number Format

34 - Power Supplies

35 - Assembled Printed Circuit Boards

36 - Electro-Mechanical Devices

1.2.2 Sequence Field. The Sequence number identifies a particular item within the category. Sequence numbers are assigned serially, and have no other significance.

NOTE

The Sequence Field, like all other part number fields, may be lengthened as required. The field lengths shown on Figure 9 are minimum lengths (insignificant zeros must be added to maintain these minimums).

1.2.3 Manufacturing Variation Field. The optional Manufacturing Variation Field consists of the letter "M" followed by two digits.

NOTE

A part number must contain a Category number and a Sequence number. All other fields are optional.

The M Field is used to distinguish between parts which are electrically and mechanically equivalent (interchangeable), but which vary in method of manufacture. For example, if leads are welded instead of soldered on an assembly, the M Field changes.

An important exception to the meaning of the M Field exists for categories related to software. Here the M Field number, when used, indicates the form in which a particular program is presented. For example, define a program as a set of machine instructions. These same identical instructions may be presented on punched cards, paper tape, or magnetic tape; and for any of these they could be in symbolic form. Thus, there are many ways to present the same identical program. These ways are identified by the M Field numbers as follows:

M01 - Symbolic Punched Cards

M02 - Relative Binary Punched Cards

M03 - Absolute Binary Punched Cards

M04 - Symbolic Magnetic Tape

M05 - Relative Binary Magnetic Tape

M06 - Absolute Binary Magnetic Tape

M07 - Symbolic Punched Paper Tape

M08 - Relative Binary Punched Paper Tape

M09 - Absolute Binary Punched Paper Tape

M10 - Bootstrap Binary Object Punched Paper Tape

M11 - Read-Only-Memory (ROM) Absolute Binary Object Punched Paper Tape

M12 - ROM Wiring and Test Set (ROMWATS) Wiring Punched Paper Tape

M13 - ROMWATS Check Punched Paper Tape

M14 - Binary 8-bit Paper Tape

4.2.4 Functional Variation Field. The optional Functional Variation Field consists of the letter "F" followed by two digits. The F Field is used to distinguish between parts which are not necessarily electrically or mechanically equivalent, but which are described by the same set of drawings. For example, a power supply may be strapped internally to operate on either 110 vac or 220 vac. Except for this strap, all power supplies of this type are identical. The strapping option is easily described by a note on the assembly and test specification drawings. Therefore, this is a functional variation.

4.2.5 Revision Field. The optional Revision Field consists of the letter "R" followed by two digits. The R Field is used to indicate minor electrical or mechanical changes to a part which do not change the part's original character. R Field changes often reflect improvements. Parts with different R Fields are not necessarily electrically or mechanically interchangeable.

4.2.6 Drawing Field. The optional Drawing Field consists of a letter from "A" to "E" followed by two digits.

The letter indicates the size of the original drawing. The sizes for each letter are:

A - $8\frac{1}{2}$" X 11"
B - 11" X 17"
C - 17" X 22"
D - 22" X 34"
E - 34" X 44"

The two digits indicate the drawing type as follows:

01 - Parts Lists

02 - Machining Details

03 - Assembly Details

04 - Lettering Details

05 - Art Details

06 - Wire Run Lists

07 - Revision Control

08 - Schematics

09 - Test Specifications

10 - Specifications

11 - Bill of Materials

12 - Information

13 - Program Listings

14 - Abstracts

15 - Program Descriptions

16 - Operating Instructions

17 - Design Specifications

18 - Flow Charts

4.2.7 Examples. The following list provides some examples of the part numbering system. The numbers were arbitrarily selected, and in most cases are fictitious.

35-060	The 60th printed-circuit board assigned a part number under this system.
35-060M01	A printed-circuit board electrically and mechanically interchangeable with the 35-060, but differing in method of manufacture.
35-060F01	A printed-circuit board not electrically and mechanically interchangeable with the 35-060, but described by the same set of drawings.
35-060R01	A revised 35-060 printed-circuit board, not necessarily electrically and mechanically interchangeable with the 35-060. Probably supercedes the 35-060.
35-060A01	The $8\frac{1}{2}$ by 11 inch parts list for a 35-060.
35-060R01B08	The 11 by 17 inch schematic for a 35-060R01.
06-072	The 72nd utility program assigned a part number.
06-072A13	An $8\frac{1}{2}$ by 11 inch listing of the 06-072 program.
06-072M03	An absolute binary deck of punched cards for the 06-072 program.
06-072A12	An $8\frac{1}{2}$ by 11 inch information drawing on the 06-072 program. Probably a manual describing the program.
29-060	The 60th manual assigned a number under this system. Note that this number is not referenced in any way to the part number of equipment described in the manual.

4.3 Drawing System

This Section describes the drawings provided with GE-PAC 30 equipment. Note that drawings provided with peripheral devices and other purchased items may vary from the system described in this Section.

A digital system may be divided into a collection of functionally independent cirucits such as core memory, processor, I/O device controllers, read-only-memory, etc. These circuits may or may not be saleable units in their own right, but in the electrical sense they are essentially self contained and capable of performing their function with minimum dependence on other functional circuits in the system. Hence a functional circuit is treated as a building block. Each functional circuit is described electrically by a detailed Functional Schematic (FS). Each FS contains a variety of information including type and location of daughter-boards or discrete intergated circuits (IC's), pin connections, all interconnections within the FS, mother-board connector pin numbers and connections to other FS's. Further, the FS's are drawn to reflect, in an orderly fashion, all logical operations performed by the circuits.

Generally, symbols used on FS's conform to MIL-STD-806B. All pins are numbered as shown earlier in this Chapter on Figures 4 and 6. In FS drawings, a daughter-board is treated as a disposable package in the

same sense that an IC or a transistor is considered as disposable. Thus, symbols for daughter-boards are presented on a daughter-board terminal basis. This does not necessarily reflect the actual physical layout of the board. For example, if only two diodes and the collector of a 4-input NAND gate are available at pins on the daughter-board, the symbol used on the FS drawing for that gate indicates a two input NAND.

Specific IC pin numbers within a daughter-board are considered inaccessible as far as FS drawings are concerned. The daughter-board pins on the other hand are accessible. Therefore, the daughter-board pins are shown on the FS. If any connection is contained completely within the daughter-board, no pin numbers are indicated. Where IC's, transistors, or other discrete components are mounted directly on the mother-board, symbolic notation and pin numbering conform with the standards established by the industry.

Registers are named according to the following rules:

1. The register mnemonic name has a maximum of three letters, excluding "I, O, Q, and Z".

2. Each bit in the register is numbered, usually starting at 00 on the left, and continuing to N-1 on the right, where N is the number of bits in the register.

3. The 00 bit is the Most Significant Bit and the N-1 is the Least Significant Bit.

The components on any given mother-board may be contained on plug-in daughter-boards, they may be connected directly to the mother-board, or there may be a combination of both. The technique for identifying and locating the components, and their pin numbers, is described in the following paragraphs.

First consider the case where all components are mounted on daughter-boards. In a few cases, one descriptive symbol can be assigned to a complete daughter-board. Generally though, it is more convenient to draw discrete logic elements (gates, flip-flops, etc.) rather than to develop specialized symbols for each and every daughter-board. Regardless of the symbol, all pin numbers and device identification numbers are relative to the daughter-board rather than to the specific IC's, transistors, etc. No specific information is shown on the FS that locates a device internally on a daughter-board. Four different pieces of information are indicated for a logic symbol when the device is contained on a daughter-board.

1. Daughter-Board Type

2. Mother-Board Location

3. Daughter-Board Pin Numbers

4. Daughter-Board Location

The daughter-board type is identified by its part number (category and sequence field). All daughter-boards have the number 35 in the category field. Following the daughter-board type number are three or four characters used to identify which mother-board contains the daughter-board. (If all daughter-boards on an FS sheet are on the same mother-board, the mother-board is identified in a footnote.) If possible, the mother-board is located absolutely by the back panel connector number that it plugs into. However, in many cases an absolute location cannot be specified for a mother-board, particularly in the case of I/O control cards and memory, where the same mother-board may be used many times. I/O mother-boards are identified mnemonically using three characters. Here the leading two characters are upper case letters, while

the last character may optionally be a number that specifies which mother-board in cases where more than one mother-board of that type is provided. For example, ME0 and ME1 for Memory boards 0 and 1 respectively. Figure 10 shows the symbol for a three-input NAND gate on the ME1 board. Note that 35-XXX is the daughter-board type, and ME1 is the mother-board containing the circuit. When possible, the designations are written inside the symbol as shown. The daughter-board type number must be shown for every symbol.

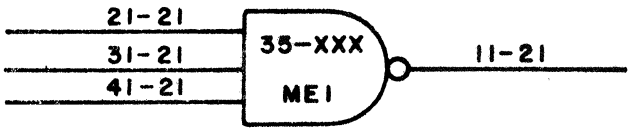


Figure 10 Three-Input NAND Gate Symbol, Wire Wrap Mother-Board

As mentioned previously, only daughter-board pin numbers are shown on FS drawings. These pin numbers correspond directly with the mother-board pin numbers within a mother-board field. Every connection available to a device via a daughter-board pin is shown. This is done by writing the pin number first, followed by the mother-board field number that contains the pin. Therefore, the pin numbering system for FS drawings indicates the position of the daughter-board in terms of the mother-board field (s) that it covers) as well as the pin number within that field. Thus, the output from the gate on Figure 10 appears at mother-(and daughter-)board pin number 11, in mother-board field 21 on the ME1 board. When IC's are mounted directly on the mother-board, they must be located in either the Left (L) or the Right (R) position within a mother-board field. Each symbol is labeled on the FS with the device part number (category and sequence). The category number for IC's is 19. Thus, a power gate mounted directly on a mother-board is indicated as shown on Figure 11. Note that the pin numbers correspond directly with the actual IC pin numbers. Note further that the mother-board field number is 10 and the the IC is in the Left position in that field. Figure 12 illustrates the method used to determine component location on a copper mother-board. Note that the same basic zone system employed on the wire-wrap boards is used on the copper mother-board as well. Use zone 26 as an example. Note that each zone is divided into two halves. The halves are designated Left and Right from the copper side. Therefore, the Left and Right zones are reversed, as indicated, if viewed from the apparatus side. Up to six other components (resistors, capacitors, and/or diodes) can be located in each half of the zone, in addition to the integrated circuit module. Three locations are provided above the integrated circuit module, and three locations are provided below the integrated circuit module. The locations are numbered one through six from bottom to top. As shown on Figure 12, the component designations take the form 26LR6, where 26 is the zone, L indicates the Left side of the zone as viewed from the copper side, R designates a resistor, and 6 is the component location. A capacitor in the same position would be

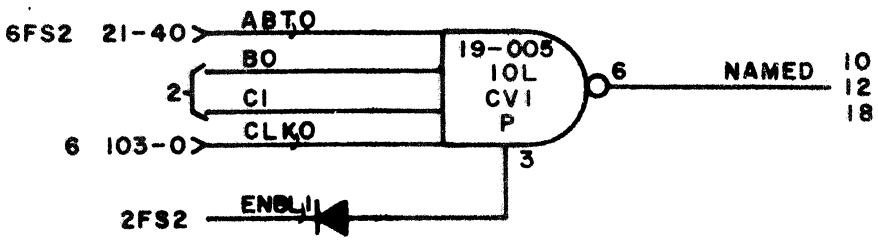


Figure 11 Power Gate Symbol, Copper Mother-Board

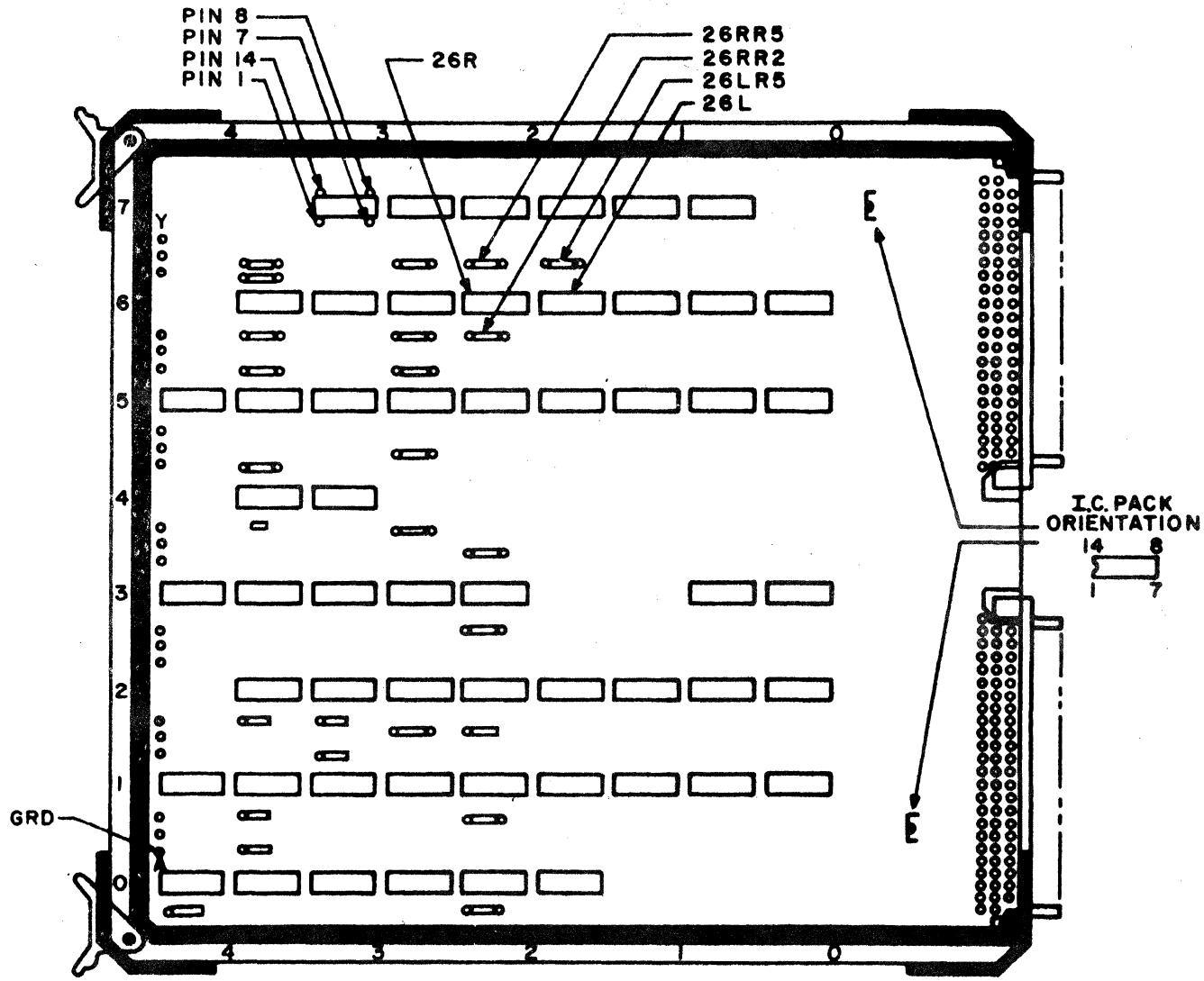


Figure 12 Typical Copper Mother-Board, Component Location Diagram

designated 26LC6; a diode would be designated 26LD6. Up to 21 test points are provided at the rear of the mother-board. The test points are lettered bottom to top from A through Y (omitting I, O, L and Q). The A test point is always ground.

A net is defined as an electrical connection between two or more points in a circuit. Ordinarily, a net has an originating end (usually a collector where the signal is generated) and one or more terminating ends. Often it is convenient to assign descriptive mnemonic names to nets as a way of identifying them on FS's. Whether a net is named or not is sometimes arbitrary. However, a net is always assigned a name if:

1. The net is contained on one drawing sheet but is not shown as a complete solid line on that sheet.

2. Part of the net appears on more than one sheet.

3. Part of the net connects with a different FS.

4. Part of the net leaves a mother-board.

If a net is named, the following rules are observed.

1. All mnemonic names are a maximum of five characters.

2. All decimal digits and upper case letters except the letters "I, O, Q, and Z" are permitted.

3. No other characters permitted.

4. Where possible, mnemonics are descriptive. However, it should be recognized that descriptive names are not always possible and a danger of misinterpreting a mnemonic exists.

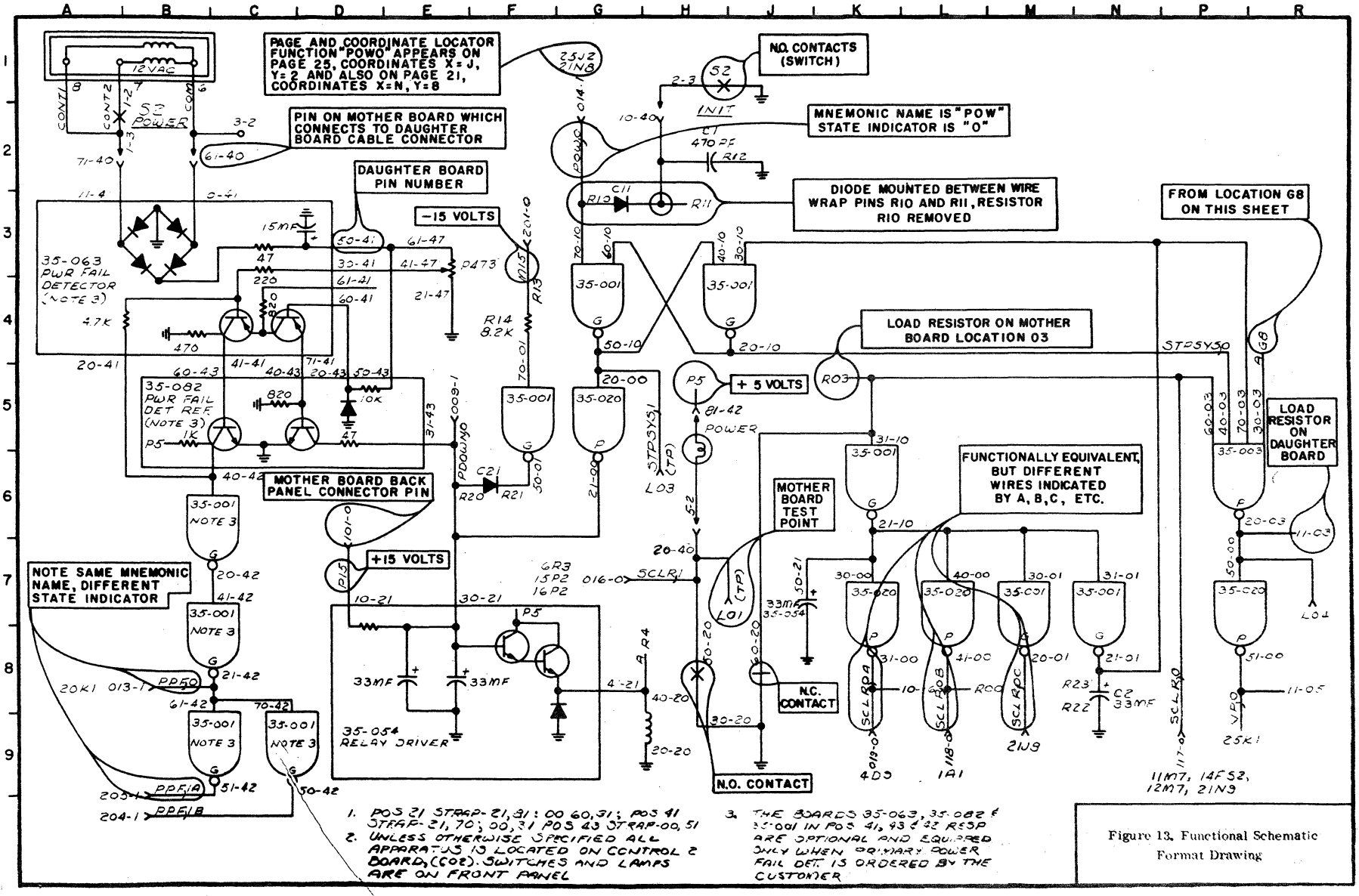
5. Mnemonic names are not repeated within an FS.

6. Every mnemonic is suffixed by a state indicator. This indicator consists of the digit "1" for the logically true state, or the digit "0" for the logically false state. For example, the set side of a flip-flop would have the "1" state indicator, while the reset side would have the "0" state indicator. The state indicator for a function changes each time that function is inverted. Thus, the state indicator permits assigning the same mnemonic to functions that are identical except for an inversion.

7. When a logical function is inverted, an inversion indicator is added after the state indicator. This allows for functionally equivalent, but electrically different nets to have the same mnemonic name. For example, assume a signal NAME,1. NAME,1 may be inverted to produce NAME,0. If NAME,0 is then inverted, NAME,1A is produced. NAME,1 and NAME,1A are functionally equivalent, but physically different nets.

Sometimes a net fans-out to many sheets in an FS. It is also possible for a net to fan-out to sheets in different FS's. In these situations, the net is assigned a mnemonic name. The net is also "zoned" from sheet to sheet to allow for properly identifying the originating and terminating ends of the net. The originating end of a net is defined as the collector at which a signal is generated. All other points to which the net connects are called terminating ends. When a lead leaves a sheet at the originating end, it is zoned to each and every sheet on which the net reappears, by indicating first the page number, followed by the FS number that contains that page. For example, assume that the gate shown on Figure 11 is on FS1, sheet 20. The output, NAME,0, appears on sheets 10, 12 and 18 of FS1. Note that the FS1 is implied. When a net enters a sheet from another sheet, it is labeled with the same mnemonic name, and is zoned back to the originating end of the net only. Thus, on Figure 11, the ENBL,1 may, however, have many other terminations in addition to the one shown. Generally then, when a net leaves the sheet where it originates, it is zoned to every other sheet where the net terminates, while the terminating end is zoned only to the orignating sheet.

When a lead leaves a mother-board, it usually does so through a mother-board back panel connector pin. These connector pins must be shown on the FS even if the complete net is shown on one drawing sheet. Only the connector pin number need be indicated under the pin symbol, since the connector number itself is implied by the



1. POS 21 STRAP-21,31 : 00 60,31; POS 41 STRAP-21,70; 00,31 POS 43 STRAP-00,51
2. UNLESS OTHERWISE SPECIFIED ALL APPARATUS IS LOCATED ON CONTROL BOARD, (C02). SWITCHES AND LAMPS ARE ON FRONT PANEL
3. THE BOARDS 35-063, 35-082 & 35-001 IN POS 41, 43 & 42 RESP ARE OPTIONAL AND EQUIPPED ONLY WHEN PRIMARY POWER FAIL DET. IS ORDERED BY THE CUSTOMER

Figure 13. Functional Schematic Format Drawing

mother-board location number in the logic symbol or in the footnote. Thus, on Figure 11, CLK0 enters the CU1 mother-board on pin 103 of the lower connector.

Sometimes, particularly in the I/O area, cables enter and leave via special daughter-boards plugged directly into the mother-board. Here the same symbol is used, but it is labeled as standard daughter-board pins. For example, on Figure 11, ABT0 enters CU1 via a cable connected to a daughter-board plugged into Field 40, to Pin 21. Figure 13 is a typical FS sheet with call-outs illustrating many of the conventions described in this Section.

The FS drawings for the basic Digital System and some of the more common expansions are included in Volume 2 of this manual. FS drawings for other expansions are included with the expansion or in publications which describe the expansions.

THE GE-PAC 30-1 PROCESSOR

TABLE OF CONTENTS

ILLUSTRATIONS

THE GE-PAC 30-1 PROCESSOR

1. INTRODUCTION

This Section describes the 30-1 Processor. Refer to the General Description Section earlier in this manual for a description of the functional organization of a GE-PAC 30-1 Digital System. This Section includes a block diagram analysis, functional descriptions which reference schematics provided in Volume 2 of this manual, maintenance information, and a mnemonic list.

2. BLOCK DIAGRAM ANALYSIS

Figure 1 is a block diagram of the 30-1 Digital System. The heart of the 30-1 is the Read-Only-Memory (ROM). The ROM permits micro-programming the 30-1 to appear as though it has the capabilities of a much larger computer. The operation of the 30-1 is directed by a program wired into the ROM. When executing the instructions of the emulated computer, the micro-program directs the hardware to read the next instruction to be executed from core memory. The micro-program then decodes the emulated (user's) instruction by performing logical and arithmetic operations on the data that was obtained from memory. Having decoded the instruction, the micro-program then enters a micro-subroutine that has been designed to perform the emulated instruction. The loop is then closed by incrementing an emulated instruction counter and returning to the point in the micro-program that will fetch the next instruction from memory. By adding the logic necessary to start, stop, and select a starting address for the micro-program, the machine is made to appear much larger.

The micro-program wired into the Read-Only-Memory consists of combinations of 16 micro-instructions. The 16 micro-instructions the 30-1 executes, and their operation codes, are listed below. For a detailed explanation of each instruction, refer to the Micro-Programming Reference Manual, Publication Number 29-017R01.

INSTRUCTION	OP-CODE
DO	0000
Branch	0001
Test	0010
Command	0011
Load	0100
Load Immediate	0101
OR	0110
OR Immediate	0111
AND	1000
AND Immediate	1001
Exclusive OR	1010
Exclusive OR Immediate	1011
Add	1100
Add Immediate	1101
Subtract	1110
Subtract Immediate	1111

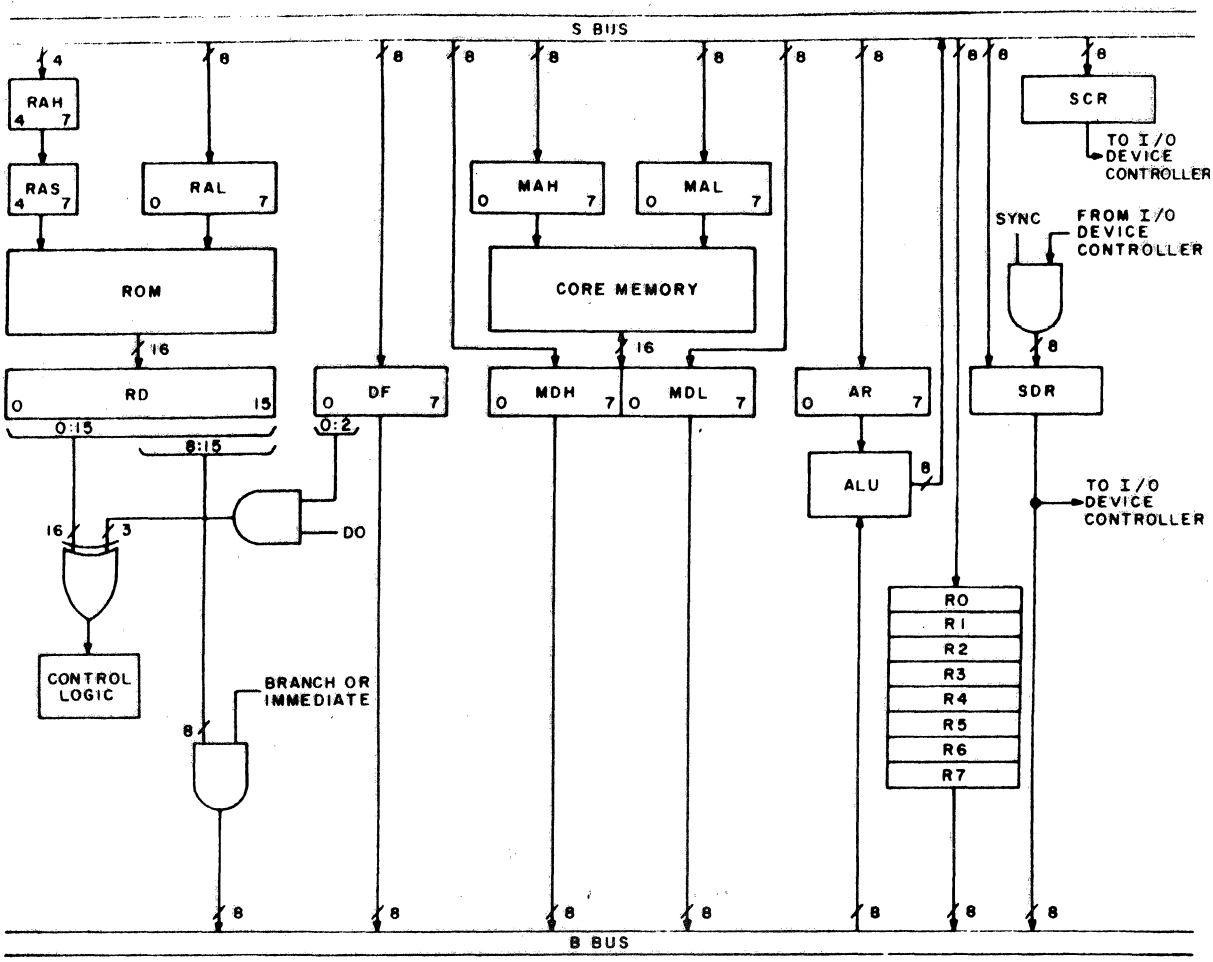


Figure 1. GE-PAC 30-1 Digital System, Block Diagram

The Read-Only-Memory (ROM) contains the micro-code program to fetch, decode and execute the user's instructions stored in the Core Memory. The ROM is a high speed, non-volatile, memory made up of pulse transformers. (See the ROM section of this manual.)

The Processor executes micro-instructions extracted from the Read-Only-Memory (ROM) by gating data through the Arithmetic Logical Unit (ALU). All data is transferred within the Processor on the eight bit B and S Buses shown on the top and bottom of Figure 1.

The location of the micro-instruction extracted from the ROM is designated by the twelve bit address contained in the ROM Address Lower register (RAL), and the ROM Address Slave register (RAS). RAL is an eight bit micro-instruction location counter. It is loaded with the contents of the S Bus during a Branch instruction or when specified as a Destination register, and incremented by one between execution of all other instructions. RAS is a four bit register that holds the ROM page address. It is loaded with the contents of the ROM Address Higher register (RAH) whenever RAL is loaded from the S Bus. Neither the RAS nor the RAL are involved in the incrementing of the RAL. Therefore, to transfer from one page of 256 words of ROM to another, the destination page number is first loaded into the RAH from the S Bus. When the location address is loaded into RAL, RAS is loaded with the contents of RAH. This insures that the new location address and page address are sensed in the ROM address decoders simultaneously.

Every micro-instruction extracted from the ROM is placed in the ROM Data Register (RD) where it remains while the instruction is executed. RD is a 16 bit micro-instruction register. It is reloaded from the ROM at the completion of each micro-instruction. The outputs from RD are DC inputs to the Control Logic. The Control Logic directs the Processor actions based on the contents of RD.

Within the Control Logic, the instructions are decoded to activate gating leads to the ALU. The Source and Destination address are decoded, and the signals to unload registers to the B Bus and load registers from the S Bus are generated. Core memory actions are initiated from Control and the signals directing the clock system and initialize circuits are also maintained. RD bits 0:3 are decoded as the operation code of the micro-instruction being executed, except during a DO instruction. During a DO instruction, bits 0:2 of the DO Flag Register are decoded as the operation code.

RD bit 3 determines Immediate instructions. An Immediate instruction is one that has the specific data to be manipulated appended to the instruction word itself. During execution of an Immediate instruction or a Branch instruction, RD bits 8:15 are gated onto the B Bus and treated as data. Since bit 3 of the DO Flag Register is not decoded during a DO instruction, Immediate instructions cannot be implemented.

The D field (4:7) of the RD register specifies the destination of the micro-operation performed in the ALU. The result of the micro-operation is placed on the S Bus by the ALU and then gated from the S Bus to the destination register specified by the D field.

The S field of the RD register (8:11) selects the register to be gated to the B Bus. This register is the source of one operand to be used in arithmetic and logical operations. The source of the other operand is always assumed to be the A Register (AR).

The E field of the RD register (12:15) is an extended micro-operation modifier. This field permits control of such things as flags and carries, or shifting and one's complementing operations.

The DO Flag Register (DFR) serves as an eight bit general purpose register, in addition to being the memory for flags indicating the results of the micro-instructions and the source of op-codes during execution of DO instructions. The DFR can be loaded with the contents of the S Bus or unloaded onto the B Bus. The DFR can be considered as two separate four bit registers. The flag register portion (bits 4:7) contains four flags: Carry (C), Overflow (V), Greater (G), and Less (L). These flags may be set or reset upon conclusion of arithmetic and logical operations and tested by the Branch micro-instruction. The DFR bits 0:3 form a four bit register. The most significant two bits are used as the Op-Code of the micro-operation when a DO micro-instruction is loaded into the RD register. The DO micro-instruction therefore directs the micro-processor to execute the micro-instruction located in the DFR. This provides a means of performing multiple functions with a single micro-subroutine.

The register stack consists of eight general purpose registers (R0 through R7). These registers are each eight bits long and may be considered as a high speed working memory for the micro-computer. Designating one of these registers in the Source field (S) of a micro-instruction causes its contents to be placed on the B Bus. Designating one of these registers in the Destination field (D) causes it to be loaded with the data present on the B Bus.

The Arithmetic Logic Unit (ALU) normally performs the micro-instruction in RD. The specific operation executed, is determined by the gating leads activated as the Control Logic decodes the micro-instruction.

The B Bus is always input as one operand to the ALU. The other operand is taken from the A Register (AR). The AR is an eight bit register that can be loaded from the S Bus and input to the ALU as an operand, if specified by the instruction being executed. Therefore, the ALU manipulates data received only from the B Bus, or data received from both the B Bus and the AR. The output from the ALU is gated onto the S Bus.

The Core Memory is the source of User instructions and data constants. It consists of ferro-magnetic cores and contains from 2K to 32K 16 bit locations. (See the Core Memory Section of this manual.) Memory operations are initiated by the Control Logic during execution of a Command micro-instruction. The location extracted from core memory is designated by the two eight bit memory address registers, MAH and MAL. Memory Address Higher (MAH) contains the most significant eight bits. MAH and MAL are loaded from the S Bus prior to the memory operation. Together, MAH and MAL form the 16 bit memory address. The core memory reads and writes 16 bit words. The two eight bit Memory Data Registers (MDH, MDL) hold the 16 bits read from, or written into, memory. Memory Data Higher (MDH) and Memory Data Lower (MDL) can be loaded from the S Bus in preparation for a memory write operation, or unloaded onto the B Bus following a memory read operation. Memory timing considerations are resolved by the Processor. If the micro-program tries to unload the Memory Data Registers immediately following a memory read, the Processor simply waits until the data is available. Non-memory directed micro-operations may be interleaved with the memory operations for maximum speed.

The System Control Register is an eight bit instruction register for the device controllers. It is loaded with the contents of the S Bus and output to the device controllers. Each bit of the SCR directs the device controllers to load data into, or accept data from, the System Data Register (SDR). The type of data to be loaded or unloaded is determined by the bits of the SCR. When the device has performed the function indicated by the SCR, it responds by raising the Sync line. The status of this line, and others, may be ascertained by the Test micro-instruction.

The System Data Register (SDR) serves as a buffer between the Processor and device controllers. During Write operations, SDR is loaded from the S Bus, and output to the device controllers. During Read operations, data is gated into SDR with the returning Sync pulse, and then unloaded onto the B Bus by the micro-code.

3. FUNCTIONAL DESCRIPTIONS

This Section provides descriptions of the major Processor functional groups. The descriptions reference both simplified drawings provided in this Section, and Functional Schematics provided in Volume 2 of this manual.

3.1 Clock Control

The clock generator and the clock skipping circuits are shown on FS22-28, and are located on the CO2 mother-board. The clock system for the 30-1 employs a crystal-controlled 2.7MC oscillator. The crystal itself is mounted with a voltage divider on the daughter-board in location 37. The output from the crystal circuit feeds the oscillator in location 36, which amplifies the crystal output. The amplified sinusoidal output, OSC$\frac{1}{2}$1 on pin 10 of the oscillator, varies between ground and +5V.

OSC$\frac{1}{2}$1 is connected to the T input of a JK flip-flop. When the flip-flop is reset, the high output CLKOFF$\frac{1}{2}$0 is ANDed with OSC$\frac{1}{2}$1 to develop the RAWCLK input to the pulse shapers. When the flip-flop is set, CLKOFF$\frac{1}{2}$0 goes low, preventing OSC$\frac{1}{2}$1 from developing RAWCLK, and inhibiting the clock outputs.

The J input of the CLKOFF flip-flop is tied to ground on the back panel. The K input is designated VP0. When the system is initialized, SCLR0 and VP0 go low. SCLR0 sets the CLKOFF flip-flop. As long as VP0 is low, the negative transition of OSC1 cannot reset CLKOFF, and clock pulses are inhibited until the initialize sequence is terminated. At this time VP0 and SCLR0 go high. CLKOFF resets on the next negative going transition of the oscillator output, enabling OSC1 to develop the RAWCLK input to the pulse shapers.

The low going result of ANDing OSC1 and RAWCLK is ORed with the normally high EXTCLK0 lead. This lead is provided so that an external clock may be used to run the system during trouble-shooting or marginal testing. This is accomplished by grounding the CLKOFF0 lead at pin 110-1 on the back panel to disable the internal oscillator, and then pulsing EXTCLK0 with a low-going output from an external oscillator. The external oscillator used should meet the following specifications:

1. square wave oscillator
2. logic levels - 0 to + 0.45 vdc for logic ZERO, 2.6 to 5 vdc for logic ONE
3. frequency - up to 2.7 MC
4. output drive - 6 milliamps

The output of the OR gate, RAWCLK0, is available at Test Point V on CO2. RAWCLK is next applied to 3 pulse shapers. The pulse shapers each contain two potentiometers. One potentiometer varies the pulse width, the other potentiometer varies the delay time. The procedure for adjusting these potentiometers is explained later is this Section. The outputs from the pulse shapers occur at the clock frequency, and are inhibited only when the system is initialized.

The output from the pulse shaper in position 46 is inverted to form the Current Drive pulse (CD0) for the ROM. CD0 is available at Test Point G on CO2. The output from the pulse shaper in position 35 is ANDed with LRAL0 and CLKSTP0 to develop STRB0, the strobe pulse for the ROM. STRB0 is available at Test Point D on CO2. The output from the pulse shaper in position 44 is ANDed with LRAL0 and CLKSTP0 to develop the system clocks, CLOD and CLOE. CLOD is available at Test Point E on CO2. CLOD is used primarily to gate data into the Processor registers. CLOE is used to gate information into the memory registers.

Two other inputs are used to develop CLOD, CLOE, and STRB0: LRAL0 and CLKSTP0. These are clock inhibits used to prevent generation of timing functions at specific times.

The normally high CLKSTP0 lead becomes active (low) when a memory operation is attempted and the memory is not ready. This inhibits the ROM strobe (STRB0) and the system clocks (CLOD and CLOE), allowing the Processor to wait for the memory operation to be completed. The CLKSTP0 signal is controlled by the two JK flip-flops shown at the top of FS22-28.

The flip-flop on the left inhibits CLOD, CLOE, and STRB0 if the micro-code program directs the Processor to unload the Memory Data Higher (MDH) or Memory Data Lower (MDL) registers after a memory cycle has been initiated and the data is not yet available. The low active, Unload Memory Data Higher (UMDH0) and Unload Memory Data Lower (UMDL0) signals are ORed and then ANDed with the set output of the JK flip-flop shown on the left. The state of this flip-flop is controlled by the Memory Data Available signal (MDAV0) coming from the core memory.

MDAV,0 is low if there is no memory operation in progress, or if a memory read is being executed and the readout is available in the Memory Data Register (MDR). MDAV,0 resets the flip-flop, preventing it from inhibiting the clock outputs. If memory is in the first half cycle of a memory read, MDAV,0 is high causing the JK to set at the trailing edge of the next system clock. If the micro-code program attempts to unload the Memory Data Register with this flip-flop set, the CLKSP,0 signal goes low, stopping the clock. When memory data becomes available, MDAV,0 goes low, the JK flip-flop resets, and CLKSTP,0 goes high, enabling clock pulses to be generated to unload the Memory Data Register.

Notice that the clock which triggers the flip-flop is not inhibited by CLKSTP,0. As soon as memory data becomes available, the flip-flop is reset on the trailing edge of the next pulse. Since STRB,0 was inhibited by the CLKSTP,0 signal, the next ROM instruction word was not read out and UMDH,0 or UMDL,0 remained low while the clock was stopped.

Since memory operations are asynchronous with the system clock, the flip-flop guarantees that a clock pulse will not be shortened when data becomes available.

The other JK flip-flop capable of developing CLKSTP,0 also has an input from memory. This lead, Memory Busy (MBY,1A), is normally low, holding the flip-flop reset. MBY,1A goes high and sets the flip-flop when any memory operation is initiated, and then remains high for the full memory cycle. If the micro-code program attempts to load any memory register or initiate another read or write cycle, the LMST,1 lead (FS22-28J2) goes high. LMST,1 is ANDed with the set output of the JK flip-flop to develop the CLKSTP,0 signal which prevents generation of CL,OD, CL,OE, and STRB,0. At the conclusion of the initial memory operation, the MBY,1A lead goes low, allowing the next clock pulse to reset the flip-flop. With the flip-flop reset, CLKSTP,0 goes high enabling CL,OE, CL,OD, and STRB,0. The pending memory operation is then executed.

The other function which inhibits CL,OD, CL,OE, and STRB,0 is the LRAL,0 function. LRAL,0 is normally high, and goes low to stop the clock when one of the following micro-ops is being executed.

1. Load RAL
2. Branch
3. Add
4. Subtract
5. Unload SDR

Stopping the clock for the first two micro-ops permits sufficient time for the new address to be decoded in the ROM decoding matrix. Stopping the clock for Add and Subtract permits sufficient time for the "worst case" carries to propagate through the arithmetic system. Stopping the clock for the last micro-op insures sufficient time for the data register to be loaded by the returning sync pulse, following a data request.

The flip-flop shown on FS22-28C7 provides the capability of skipping one clock pulse while performing Add, Subtract, Branch, Unload SDR, or Load RAL micro-ops. The J input to the flip-flop is normally low and becomes high if any one of the five micro-instructions listed previously is executed. The J input is fed by two NAND gates whose outputs are tied together. A low signal is required at the input of each gate to make the output go high. During a Branch instruction, BRA0 is low, forcing both NAND gate outputs high. During Add or Subtract, L,0 is low, forcing both outputs high. During a Load RAL instruction, LR14,0 is low, causing one NAND gate output to be high. BANK,0 is high forcing the other NAND gate output high. While unloading the System Data Register, USD,0 is low, and forces both outputs high. The high preset on the J input of the flip-flop is ANDed with the reset output from the flip-flop, which is high until the trailing edge of the next clock pulse. The output from this

gate, LRAL$\emptyset$, goes low and inhibits CLOE, CLOD, and STRB$\emptyset$. With a high on the J input to the flip-flop and the K input tied to +5 volts, the flip-flop complements on the next clock pulse. With the flip-flop set, the reset output goes low, forcing LRAL$\emptyset$ high and once again CLOE, CLOD, and STRB$\emptyset$ are enabled. Note, however, that one clock pulse has been skipped. LRAL$\emptyset$ and its complement LRAL1 are also wired to the RM1 mother-board to inhibit the incrementing of RAL and enable the gates from the S Bus to RAL.

During a Branch or a Load RAL micro-op, the information is gated into RAL from the S Bus with the RA clock (RCL$\emptyset$). During Add, Subtract, or Unload SDR, RCL$\emptyset$ is also skipped. This is accomplished by ORing L$\emptyset$ and USD$\emptyset$ at the gate shown in area A7.

If either L$\emptyset$ or USD$\emptyset$ is low, this gate produces a high output. This high is ANDed with the reset output from the flip-flop to produce a low into the gate that generates RCL$\emptyset$, thus inhibiting it. Notice that another input to the gate which generates RCL$\emptyset$ is CLKSTP$\emptyset$. This input is used to inhibit RCL$\emptyset$ from being generated to increment the address register during a memory clock stop sequence. During Branch or Load RAL, L$\emptyset$ and USD$\emptyset$ are both high inputs to the gate, forcing the output low. This low produces a high output from the gate shown in area D9, enabling RCL$\emptyset$ to be generated

In summary, all clocks are inhibited during an initialize sequence. If a memory operation is attempted when memory is busy, or if data is not yet available from memory; CL$\emptyset$OD, CLOE, RCL$\emptyset$, and STRB$\emptyset$ are inhibited. If a Branch or Load RAL is executed, CL$\emptyset$OD, CL$\emptyset$E, and STRB$\emptyset$ skip one pulse. If an Add, Subtract, or Unload SDR is executed; CL$\emptyset$OD, CL$\emptyset$E, STRB$\emptyset$, and RCL$\emptyset$ all skip one pulse.

3.2 Initialize Control

The Initialize Control logic is shown on FS22-27. This logic provides an orderly system shutdown when the system is initialized. When an initialize sequence occurs, the following actions take place:

1. Stop Clock
2. Reset RAL
3. Reset RAH
4. Reset DFR
5. Reset Bank flip-flop
6. Reset Utility flip-flop
7. Reset RD
8. Reset SCR
9. Provide Reset Signals for Memory and I/O
10. Set Wait Alarm

The system is initialized as a result of one of the following conditions:

1. INITIALIZE Switch depressed
2. Command Power Off
3. AC Input (optional), P15, N15, or P5 falls below minimum operating level

The master reset signal, SCLR$\emptyset$1, is active whenever the initialize relay is released by the PDOWN$\emptyset$ signal generated at FS22-27F5. During normal operating conditions, PDOWN$\emptyset$ is high, permitting the Darlington circuit controlling the initialize relay to conduct, and keeping the relay energized. PDOWN$\emptyset$ goes low to start an initialize sequence whenever one of the three conditions mentioned previously occurs.

When the normally open INITIALIZE Switch (H1) is depressed, a ground level sets the STPSYS$\emptyset$1 flip-flop (G4 - H4). The 470 Pf capacitor and 1K pull-up resistor filter the higher frequency components of the switch bounce.

The STPSYS1 flip-flop is also set if a Command Power Down is executed. The Processor tests for Primary Power Fail between execution of each user instruction. The test micro-instruction sets the Greater Than flag in the DO Flag register, if the latching POWER Switch is released or if the optional Primary Power Fail Detector determines that the AC voltage is lost. If the Greater Than flag is set upon conclusion of the Test micro-instruction, the micro-code program transfers to a Command Power Down. The Command instruction generates the low signal POW0 at FS22-24N4. POW0 causes the STPSYS1 flip-flop to set.

When the STPSYS flip-flop sets, PDOWN0 goes low. When PDOWN0 goes low, it removes the positive potential on the input of the Darlington circuit, causing it to turn off, and thus de-energizing the Initialize relay. The reset output from the STPSYS flip-flop generates the signal VP1, area R6. VP0 goes low and holds the K input to the CLKOFF flip-flop (FS22-28K4) low, preventing it from setting until the initialize sequence is terminated.

The third instance when PDOWN goes low to remove the positive potential from the input to the Darlington circuit and drop out the Initialize relay, is when one of the voltages, +5v, +15v, -15v, or the AC input is lost. If the N15 supply voltage is lost, the output from the gate in area F5 goes low and drives PDOWN0 low. If the +5v is lost, the collector supply for the Darlington circuit is removed, causing the Initialize relay to drop out.

If the AC input is lost, the PDOWN0 lead is also forced low, causing the Initialize relay to drop out. The AC input is sampled from the secondary of a 12VAC transformer and rectified. The pulsating DC voltage is applied to Potentiometer P473 (area E3). The potential picked off by the potentiometer is applied to the optional Power Fail Detector.

NOTE

The following description of the Primary Power Fail Detector assumes that the two transistors on the 35-063 daughter-board are Q1 and Q2 left to right, and that the two transistors on the 35-082 daughter-board are Q3 and Q4 left to right.

Initially, Transistor Q1, Q3 and Q4 are off and Transistor Q2 is conducting. With Transistor Q2 conducting, a high enough positive potential on the base of Q4 keeps Q4 cut off, and applies about +6v to the emitter of Q1. The potential picked off Potentiometer P473 (approximately 8v) is applied to the base of Q1. If the AC input is lost, or fluctuates enough, the potential on the base of Q1 becomes more negative and Q1 conducts.

When Q1 conducts, it places a positive potential on the base of Q3, turning Q3 on. This causes the potential on the emitter of Transistor Q2 to become less positive, turning Q2 off. When Q3 conducts, its collector goes to ground, generating the low signal PPF0. PPF0 is tested between execution of user instructions to determine if a Power Fail is in progress.

When Transistor Q2 turns off, the positive potential is removed from the base of Q4. Transistor Q4 then conducts, grounding the PDOWN0 lead, turning the Darlington circuit off, and dropping out the Initialize relay. The Initialize relay, which is a dry reed relay with SPDT contacts, is shown with the relay not energized.

When the relay releases, the SCLR1 contact opens, causing SCLR1 to go high. This removes the ground for the POWER lamp. The SCLR0 contact goes to ground. This metalic ground sets the Wait Alarm, resets, DFR, and is distributed to memory and I/O for initialize controls. SCLR0 also holds VP0 at ground to inhibit the clock.

SCLR$\downarrow$0 is inverted through the gate in area K6. This node will rise to +5 with a time constant determined by the 19-004 internal resistor and the 33uf capacitor. When SCLR$\downarrow$1 reaches the input threshold of the four gates it feeds, it turns them on, causing the outputs to go low. The output SCLR$\downarrow$0A resets RAS, RAH, and RAL. SCLR$\downarrow$0B resets RD, and SCLR$\downarrow$0C resets the Bank and Utility flip-flops. When the remaining gate (area N7) goes low, its output resets the STPSYS flip-flop and holds VB$\downarrow$0 at ground.

When the STPSYS flip-flop is reset, the STPSYS$\downarrow$0 input to stop the system clock, goes high. Resetting the flip-flop also removes the ground on the input to the Darlington circuit if the Initialize sequence resulted from depressing the INITIALIZE Switch. This allows the 66uf capacitor to slowly charge. When the threshold of the input transistor is reached, the Darlington circuit turns on, and energizes the relay.

When the SCLR$\downarrow$1 contact closes, it provides the ground return for the POWER lamp. The SCLR$\downarrow$0 contact opens and removes the reset functions for DFR, memory, and I/O controllers; and removes the set signal on the Wait Alarm. When this occurs, the output from the gate at K6 goes to ground immediately. This removes the reset signals SCLR$\downarrow$0A, SCLR$\downarrow$0B, and SCLR$\downarrow$0C. The output from the gate at N7 will have to charge up the 33uf capacitor. This delayed function removes the reset on the STPSYS flip-flop, and also removes the final clock inhibit.

The previous description applies for a Command Power Off, or manual Initialize, since these are the only sources to set the Initialize flip-flop.

For the case where Initialize is caused by a failure of P5, N15, P15, or Power is removed, the relay de-energizes and remains in that state until the fault is corrected.

When power is turned on, the system is initialized in an orderly fashion by the following mechanism. The SCLR$\downarrow$0 contact of the relay stays at ground enabling the reset signals SCLR$\downarrow$0A, SCLR$\downarrow$0B, SCLR$\downarrow$0C. This holds the clock off initially. As the P5 reaches normal operating levels, the Initialize flip-flop is reset after a delay. This insures that the PDOWN$\downarrow$0 node is not forced to ground and permits the 66 uf capacitor at the input to the Darlington to charge slowly. The relay then energizes.

3.3 Processor Registers

The Processor executes micro-instructions received from the ROM. This is accomplished, primarily, by moving data from one place to another and modifying it in the ALU. In order to manipulate data and execute the instructions received, the Processor uses a number of internal registers. Most of these are eight bit registers. Some registers perform special functions, but most are general purpose. Data is transferred between registers and other system elements via the eight bit B and S Buses in the Processor. Each register is described in the following paragraphs.

3.3.1 ROM Data Register. The ROM Data Register (RD) is a 16-bit register located on the ROM Interface board (RMI). The schematic drawings for this register are on FS44 - 1:3. The ROM Data Register can be thought of as an instruction register. Each bit of RD has double rail inputs from the ROM. Every micro-instruction extracted from the Read-Only-Memory is placed into the RD register, where it remains while the instruction is executed. A thorough knowledge of the ten basic micro-instructions and their instruction word formats is necessary to fully understand the RD register. (See the Micro-Instruction Reference Manual, Publication Number 29-017R01 for a detailed explanation.)

In most instances bits 0-3 of RD are decoded as the op-code of the micro-instruction being executed. RD3 = 1 defines Immediate instructions. An Immediate instruction is one that has the specific data to be manipulated appended to the instruction word itself. During the performance of an Immediate instruction, bits 8:15 of RD are gated onto the B Bus and treated as data, instead of being decoded as Source and Extended Fields, Test Code, or Command Code. The only instance when RD (0:3) is not used to determine the micro-instruction to be performed, is during execution of a DO instruction. During a DO instruction, bits 0:2 of the DO Flag Register are decoded as the op-code. Since bit 3 of the DO Flag Register is not decoded, it is impossible for the DO instruction to cause an Immediate instruction to be executed.

Bits 4:7 are normally decoded as the destination register address. This identifies the register designated to contain the result upon completion of the micro-instruction being performed. When a Branch instruction is executed, bits 4:7 define the condition that the Processor will branch on. If the Processor decodes a Test or Command instruction, bits 4:7 are part of the Command or Test Code.

Bits 8:11 are decoded as the source register address for micro-instructions in the Register to Register format. The Source register contains one of the operands to be used. If a Branch instruction is decoded, bits 8:11 are part of the branch address. If the Command or Test micro-op is decoded, bits 8:11 are decoded as part of the Command code. If an immediate instruction is decoded, bits 8:11 are part of the data.

Bits 12:15 normally make up the extended field for Register to Register instructions and are decoded to specify options for specific instructions. The extended field designates the function of carry, whether to shift or complement the operand manipulated, whether to set flags, or whether to enable the A Register input to the ALU. During other instructions, bits 12:15 are part of the Command code, or data.

Other than the double-rail inputs from the ROM, the only other input to the RD register is SCLRϕB. SCLRϕB goes low and clears the entire RD register when the system is initialized. The set and reset outputs from RD 0:2 are inputs to individual power gates. During all micro-instructions except DO, these gates reflect the contents of bits 0:2 in RD. When RD 0:3 contains 0000 (the op-code for a DO instruction), the output from the gate in area A5 of FS44-1, goes low. This low forces DRD0ϕ1, DRD01ϕ, and DRD02ϕ high. With a DO instruction in RD, the output from the four OR tied power gates shown on the right hand side of the page goes high, enabling the state of DO Flag Register bits 0:2 to be reflected in the DRD outputs. All of the outputs from the bits in the RD register appear directly in the Control logic so that the various op-codes and options can be decoded and the proper gating leads to the ALU can be activated. The set outputs of RD bits 8:15 feed individual power gates that generate the false B Bus outputs (low if a logic 1) if RD 03 is set, indicating an Immediate instruction.

3.3.2 ROM Address Lower Register. The ROM Address Lower Register (RAL) is an 8 bit register that contains the least significant portion of the address of the location to be extracted from the Read-Only-Memory. The schematics for RAL are located on FS44-4 and 5.

During performance of the major portion of the micro-program, micro-instructions are selected from successive locations in the Read-Only-Memory. The RAL register is wired as a standard up-counter to facilitate the selection of successive locations.

The RAL is incremented to read successive ROM locations except during a Branch micro-instruction when the specified condition is true, or if RAL is designated as a Destination Register. In either of these cases, the RAL increment signal is disabled and the new address is loaded into RAL from the S Bus.

The RAL register is cleared by the SCLR$_{0}$A signal when the system is initialized. The RCL$_{0}$ signal is used to increment the RAL register or to gate the information from the S Bus into RAL. The RCL$_{0}$ signal is inverted and applied to the T input of each flip-flop in RAL as RCL$_{1}$. If a Branch or a Load RAL instruction is performed, LRAL$_{0}$ is low and LRAL$_{1}$ is high. LRAL$_{1}$ is NANDed, bit for bit, with the S Bus to enable the J and K inputs to each of the flip-flops in the RAL register. If any inputs from the S Bus are high, the K input is forced low and the J input is forced high on the corresponding bit in the RAL register, causing it to set at clock time. If an S Bus input is low, the K input is forced high and the J input is forced low, causing the RAL flip-flop to reset.

When the Processor is not executing a Branch or Load RAL, LRAL$_{0}$ is high and LRAL$_{1}$ is low. With LRAL$_{1}$ low, the inputs from the S Bus are disabled and the RAL register is incremented by one on each negative transition of RCL$_{1}$. The RAL is incremented by complementing each bit of the register if all previous least significant bits are set. RAL07 complements every time. RAL06 complements if RAL07 is set. RAL05 complements if RAL07 and RAL06 are both set and so on. This is accomplished by NANDing the set output of each flip-flop with all previous bits to enable the J and K inputs for the next significant bit. If all previous bits are set, the J and K inputs to the next stage are high, allowing that bit to complement. If any of the previous bits are not set, the J and K inputs are low, preventing that flip-flop from changing state.

3.3.3 ROM Address Higher Register. The ROM Address Higher Register (RAH) is shown on FS44-6. RAH is a 4-bit extension of RAL which increases the ROM addressing capability to 4096 locations. This register is independent of the RAL register from both the programming and the hardware viewpoint. RAH holds the ROM page address and RAL holds the ROM location address. RAL is normally incremented to fetch micro-instructions from successive ROM locations. When RAL is equal to its highest address (hexadecimal FF), the next increment pulse causes it to recycle to an address of all zeros, but no carry is propagated into the least significant bit of RAH. For this reason RAH must be loaded independently each time a Branch or Load RAL selects a different page.

The RAH register is composed of RS flip-flops with single-to-double rail converters on the inputs. This register has inputs only from the S Bus. A new address is loaded in RAH at clock time from the S Bus when RAH is specified as the Destination register of a Load instruction. To address the RAH register, the Bank flip-flop must be reset. Signal LRA$_{1}$ gates the contents of the S Bus into RAH. LRA$_{1}$ is developed by the three inverters whose outputs are AND tied in the upper left hand corner of Sheet 6. The inputs to the three inverters are RCL$_{0}$, BANK$_{1}$ and LRAH$_{0}$. BANK$_{1}$ is low when the BANK flip-flop is reset. RCL$_{0}$ varies with the system clock. LRAH$_{0}$ is low when the destination address decoder decodes the address X'D'. When the three inputs are low, LRA$_{1}$ goes high, gating the S Bus into RAH. The only other input to RAH is SCLR$_{0}$A. SCLR$_{0}$A is normally high and becomes low to clear the RAH register when the INITIALIZE button on the Display Panel is depressed, a Power Down command micro-instruction is executed, or the power is turned off.

3.3.4 ROM Address Slave Register (RAS). The ROM Address Slave Register is a 4-bit register shown on FS44-6. RAS is slaved to the 4-bit RAH register. The single rail buffered outputs of RAS are decoded as the four most significant bits of the page of the ROM address to be selected.

The contents of RAH are loaded into RAS every time RAL is loaded from the S Bus. This allows the correct page address to be loaded into the RAH register prior to a Branch or Load RAL micro-instruction. When RAL is loaded with a new address as a result of executing a Branch or a Load RAL, the correct page is gated into RAS from RAH. In this way all 12 bits of the new ROM address arrive at the ROM address decoder simultaneously (allowing for slight variations in the circuit delays).

LRAS$\overline{1}$ gates RAH into RAS. LRAS$\overline{1}$ is developed by the two inverters collector AND tied on the left hand side of Sheet 6. The two inputs are RCL$\overline{0}$ and LRAL$\overline{0}$. LRAL$\overline{0}$ is the signal used to gate the S Bus into RAL. LRAL$\overline{0}$ is low during a Branch or a Load RAL micro-instruction. When LRAL$\overline{0}$ is low, LRAS$\overline{1}$ varies with RCL$\overline{0}$ and gates RAH into RAS on the negative transition.

The only other input to RAS is the normally high SCLR$\overline{0}$A signal which goes low when the system is initialized to clear RAS. SCLR$\overline{0}$A is the same signal that clears RAL.

3.3.5 DO Flag Register (DF). The DO Flag Register (DF) is an 8-bit register located on the BS1 and BS2 mother-boards. The schematic drawings are on FS22-10:12. The DF register is a general purpose register in addition to being the memory for flags indicating the results of instructions, and the source of op-codes during performance of DO micro-instructions.

The DF register is cleared by the SCLR$\overline{0}$ signal. SCLR$\overline{0}$ goes low when the system is initialized. SCLR$\overline{0}$ is connected to the C inputs of the DF register flip-flops and jams them all into the reset state when it goes low. When the DF register is used as a General Register, it is loaded with the contents of the S Bus or unloaded onto the B Bus. When the hexadecimal address F is decoded in the destination address coder, LDE$\overline{0}$ becomes low and LDE$\overline{1}$ becomes high. LDE$\overline{1}$ is NANDed with each bit of the S Bus. If any bit of the S Bus is high, the K input on the corresponding bit in the DF register goes low and the J input high, allowing the flip-flop to set on the next negative going clock transition. If the S Bus input is low, K goes high, and J goes low, allowing the flip-flop to reset.

Since the DF register is used to hold the flags indicating the result of micro-instructions, there are instances when individual bits will be set or reset independently. To facilitate this independence, different gating pulses are developed for individual bits. When the DF register is loaded from the S Bus, LDE$\overline{0}$ is used to generate all of the enables.

The DF register is unloaded onto the B Bus when the Source Register Decoder decodes the hexadecimal address F, causing the signal UFD$\overline{1}$ to go high. UFD$\overline{1}$ is NANDed with the set output of each bit of the DF register to generate the B Bus leads B0$\overline{0}$-B0$\overline{7}$. If any bit of the DF register is set, the corresponding bit on the B Bus goes low. Conversely, any bit of the DF register that is reset causes the corresponding bit on the B Bus to go high. The B Bus is a false bus and carries logic ONEs as low signals, and logic ZEROs are high signals.

The set and reset outputs of DF register bits (0:2) are direct inputs on the RM1 mother-board for use during execution of a DO

instruction. (Reference FS44-1.) During a DO instruction, the op-code is taken from DF bits 0:2 instead of from the ROM data register. DF03 is not gated to RMI; thus immediate instructions cannot be implemented through the use of the DO instruction.

In addition to being loaded from the S Bus, DF register bit 4 is used as the Carry Store flip-flop (C). DF04 is set or reset to reflect whether or not a carry or borrow resulted upon conclusion of an Add or Subtract micro-instruction, or to reflect the state of the bit shifted out following a Load micro-instruction designating the shift left or a shift right option. The gates which control setting or resetting DF04 are shown on FS22-10.

DF04 is set when the signal SDF4,0 is low. SDF4,0 is generated by three NAND gates collector OR tied. The inputs to the first NAND gate are SVSC,1, SR, and B07,1. SVSC,1 is developed on the Control motherboard (FS22-25). SVSC,1 will be high and SVSC,0 low if the Processor decodes a Load micro-instruction and ROM data register bit 15 is set, indicating that the carry out of the ALU is desired. SR,1 will be high when the Processor is doing a Load micro-instruction and the E field specifies a shift right option. B07,1 will be high if the least significant bit of the data present on the B Bus is a logic ONE.

The second NAND gate capable of driving SDF4,0 low has the inputs SVSC,1, SL,1, and B00,1. SVSC,1 was just discussed. SL,1 is high if the Processor is performing a Load micro-instruction whose E field designates the shift left option. B00,1 is high if the most significant bit of the information carried on the B Bus is a logic ONE. The third instance when SDF4,0 is low occurs when SVAC,1 and CSV,1 are both high. SVAC,1 is high during an Add or Subtract, if the E field designates the carry out of the ALU option. CSV,1 is high if a final carry or a borrow resulted during Add or Subtract.

SDF4,0 is applied to the K input of DF register bit 4, and inverted as an input to the J input. The gating pulse is developed by SVAC,0 and CLOD during Add or Subtract operations, and by SVSC,0 during Load instructions with shift options.

DF register bit 5 serves as the Overflow flag (V). The gating which controls setting or resetting DF05 is shown on Sheet 10. DF05 is set when SDF5,0 is low. SDF5,0 is generated by two NAND gates collector OR tied. One of the inputs to both NAND gates capable of generating SDF5,0. STF5,0 is generated on FS22-25. STF,0 is low when the Processor is performing any instruction that is not a Test, Load, or Immediate instruction, and bit 13 of the E field of the instruction word is set specifying the Set Flags option. STF,0 is inverted and NANDed with L,1 to generate the low signal STFL,0 during Add and Subtract.

The other two inputs needed to generate STF5,0 are C00,1 and CSV,0 for one gate, and LC00,0 and CSV,1 for the second gate. C00 is the carry into the most significant bit of the adder and CSV is the carry out of the most significant bit of the adder. Whenever these carries are different upon conclusion of an Add or Subtract (C00,1)(CSV,0) or (LC00,0)(CSV,1), an overflow condition exists and SDF5,0 goes low. SDF5,0 is applied to the K input of DF05 and inverted as an input to J. The gating signal for DF05 is generated by CL,OD and STFL,0.

DF05 can also be set by the SV,0 signal on the S input. SV,0 goes low whenever the False Sync flip-flop sets (reference FS22-29). The False Sync flip-flop is set if the Processor does not receive a return sync from a device controller within 42 microseconds from the time the System Control Register is loaded to initiate a control operation. Following an operation with a device controller, the overflow bit in DF is set if a sync was not returned from the device controller within the specified time.

DF06 and DF07 are the Greater Than and Less Than flags respectively. DF06 is set by the SDF6,0 signal if the data manipulated on the S Bus is greater than zero, or if the result of a Test micro-instruction was true. DF07 is set by the SDF7,0 if the data manipulated on the S Bus is less than zero, or if the result of a Test micro-instruction proved false.

SDF7,0 and SDF6,0 are generated at two different sources. The logic to set or reset DF06 and DF07 as a result of examining the data on the S Bus is shown on FS22-10. The inputs necessary to develop SDF7,0 and SDF6,0 are identical, with the exception of S Bus bit zero input. SDF7,0 goes low allowing DF07 to set if S Bus bit zero is set (S00,1) indicating a quantity less than zero. SDF6,0 goes low, allowing DF06 to set if S Bus bit zero is reset (S00,0) indicating a quantity greater than zero. The other two inputs required to develop either SDF7,0 or SDF6,0 are the complement of STE,0 and a signal generated by NANDing the eight reset outputs of the S Bus (S00,0-S07,0) and the reset outputs of DF06 and DF07. Using the complement of STE,0 was discussed in conjunction with setting DF05 (V) and is high enabling generation of SDF7,0 and SDF6,0 for the same conditions. The gate output is high enabling the generation of SDF7,0 and SDF6,0, the Less Than flag (DF07), or any bit on the S Bus is set. The reset outputs of DF07 and DF06 are necessary since Arithmetic, Logical, Shift, and Load instructions in the User Instruction Repertoire manipulate 16 bit operands. This is accomplished by double precision manipulation on the hardware level. The reset outputs of DF07 and DF06 allow the Greater Than flag to be set if the result of the least significant eight bits of an operation set either flag (G or L) and the most significant eight bits produced a zero result. This prevents leaving both flags reset to indicate a zero quantity or leaving the Less Than flag set in the event the least significant eight bits of a 16 bit operand indicated a quantity greater than or less than zero, and the most significant eight bits were all zero.

The logic to generate SDF7,0 and SDF6,0, as a result of executing a Test instruction, is shown on FS22-23. The test instruction examines the Processor to determine the state of specific functions. The testable functions are I/O Interrupt, I/O Sync, Console Interrupt, Console Single Instruction, Utility flip-flop, Memory Parity Fail flip-flop, and Primary Power Fail. The specific function to be tested is specified by the Test Code, bits 4:15 of the micro-instruction word. If the machine function specified by the Test Code is true, the Greater Than flag (G) is set and the Less Than flag (L) is reset. If the function specified by the Test Code is false, the Less Than flag (L) is set and the Greater Than flag (G) is reset.

On FS22-23, each of the testable functions is NANDed with the ROM data register bit with which it corresponds. The NAND gates are collector OR tied in groups of four. If one of the tested functions is true, the output at the collector OR tie goes low, driving the output of the next NAND gate high. This high is NANDed with the inverted output of the Test micro-instruction decoder, which will be high during a Test instruction, driving the output SDF6,0 low. When SDF6,0 is low, it forces SDF7,0 high. With SDF6,0 high and SDF7,0 low, the G flag sets and the L flag resets, indicating that the test was true. If the function tested for is low, the outputs from all the collector OR tied NAND gates are high, causing the output from the next gate to go low. This low causes SDF6,0 to be high. SDF6,0 is NANDed with the inverted output of the Test micro-instruction decoder, driving SDF7,0 low. With SDF7,0 low and SDF6,0 high, the L flag sets and the G flag resets, indicating that the Test was false. The same gating pulse is used to set or reset DF06 and DF07. It is developed on FS22-11. During performance of a micro-instruction specifying set test flags, the signal STE,0 and CLOD are used to develop the gating pulse. During a Test instruction, TEST,0 and CLOD are used.

3.3.6 Register Stack. The register stack is shown on Sheets 8 and 9 of FS22. The register stack consists of 8-bit General Registers that can be specified as Source (S) and Destination (D) registers for micro-instructions, during execution of the micro-code program.

These eight General Purpose Registers are loaded with the contents of the S Bus if specified as the Destination register, or unloaded onto the B Bus if specified as the Source register during performance of micro-instructions. The most significant four bits (0:3) of each of the eight registers are located on the Bit Slice 2 mother-board. The least significant four bits (4:7) of the eight registers are located on Bit Slice 1. Unlike all the previous registers of the JK or RS type, the register stack consists of two 35-014 daughter-boards, one on Bit Slice 1 and the other on Bit Slice 2. Each of the 35-014 daughter-boards is implemented to provide eight 4 bit words.

Referring to Sheets 8 and 9, if the Source Register Decoder selects one of the eight General Registers, the appropriate UR lead (UR00:UR07) goes low. This DC input is inverted and applied to a non-inverting 35-041. The 35-041 has an emitter follower output that will be at about 2.1 volts for the selected output. All other 35-041 outputs will be at ground through the emitter resistor. The outputs from the 35-041 (R0:R7) are input to the register stack. R0:R3 are developed on BS2 and applied to BS1. R4:R7 are developed on BS1 and applied as inputs on BS2.

The register stack is level active for reading, and does not require a clock pulse. As soon as one of the read leads (R0:R7) goes high, the output from the selected register appears at the output pins 10, 11, 70 and 71 on both BS1 and BS2. These outputs are about 3 volts for a logic ONE, and 0.6 volts for a logic ZERO. During execution of a micro-instruction that has not selected one of the registers, all of the outputs should be at 0.6 volts. The outputs from the 35-041 are inverted to develop the B Bus.

When one of the Destination Registers decoders selects one of the eight General Registers, the appropriate LR lead (LR00: LR07) goes low. This DC input is inverted and input to a non-inverting 35-041 where it is ANDed with a differentiated clock pulse to develop write signals W0:W7. The inverted system clock (CLOD) is input to the trailing edge differentiator. This develops a nominal 30 nanosecond positive going clock pulse as CLOD charges from 0 to +5 volts, preventing the write signal from going high to gate the S Bus into the Destination register until the end of the instruction. This pulse is normally 2 volts. The pulse width of 30 nanoseconds is fixed by the circuit constants and is not readily adjustable. The displacement, in time, of this pulse is a function of the CLOD pulse width.

When the 30 nanosecond pulse is developed, the appropriate write line goes high. W0: W3 are generated on BS1 and applied as inputs to BS2. W4:W7 are generated on BS2 and applied as inputs to BS1. When one of the write leads (W0:W7) goes high, the information on the S Bus is gated into the Destination register. The S Bus is an input to a voltage divider in the register stack which loads down the Bus. It is normal for the S Bus to be about 3.9 volts to represent a logic ONE.

The register stack is level active and does not require a gating pulse. The inverted LR lead (LR00:LR07) is ANDed with the differentiated clock pulse to prevent mutilation of the data on the S Bus. Mutilation could occur if the appropriate LR lead was not ANDed with the differentiated clock and a micro-instruction designating the same register as both the Source and Destination was executed. The Source Register decoder would activate the correct unload lead and the contents of the Source register would be placed on the B

Bus, modified in the ALU, placed on the S Bus, and gated back into the Destination register. Since the unload lead would still be high, the modified data would appear at the input to the ALU and be modified again before being placed into the Destination register.

3.3.7 Arithmetic Register (AR). The Arithmetic Register (AR) is shown on FS22-7. The Arithmetic Register is used to hold the first operand during execution of the Add, Subtract, OR, AND, and Exclusive OR micro-instructions. The AR consists of eight JK flip-flops with single rail inputs from the S Bus. The least significant four bits of the Arithmetic Register (AR04:AR07) are on Bit Slice 1 and the most significant four bits (AR00:AR03) are on Bit Slice 2. The AR can be loaded with the contents of the S Bus or the contents of the AR can be gated out as an input to the ALU.

The A Register is loaded with the contents of the S Bus, with gating signals LA$\overline{1}$A and LA$\overline{1}$B on the T inputs of the JK flip-flops. LA$\overline{1}$A loads AR bits (0:3) and LA$\overline{1}$B loads AR bits (4:7).

Both LA$\overline{1}$A and LA$\overline{1}$B are generated by similar gating configuration: two NAND gates with their outputs AND tied, as shown on FS22-22. The inputs to the two NAND gates on Bit Slice 2 that generate LA$\overline{1}$A are the system clock (CL$\overline{O}$D) and LA$\overline{0}$. LA$\overline{0}$ is low when the Destination Address Decoder decodes the Arithmetic Register's address (hexadecimal 8). When LA$\overline{0}$ and CLOD are both low, the output LA$\overline{1}$A goes high providing the negative going transition, bits (0:3), to gate in the S Bus. LA$\overline{1}$B is generated exactly the same on Bit Slice 1 to gate the S Bus in AR bits (4:7).

The set outputs from the Arithmetic Register are NANDed with UA$\overline{1}$ to develop the GA leads (GA00$\overline{0}$:GA07$\overline{0}$) as inputs to the ALU. UA$\overline{1}$ is generated on FS22-26 by three NAND gates whose outputs are collector AND tied. UA$\overline{1}$ is high enabling inputs from the AR to the ALU when each of the three NAND gates has at least one low input. UA$\overline{1}$ is low disabling inputs from the AR to ALU if one of the three NAND gates has all high inputs. Both inputs to the NAND gate on the right are high if the Processor decodes a Branch or Load instruction. All four inputs to the next NAND gate are high if the Processor decodes an OR instruction that is not immediate and RD bit 12 is set specifying the "No AR to the ALU" option. All three inputs to the last NAND gate are high when the Processor decodes an Add, Subtract, Exclusive OR, or AND micro-instruction that is not Immediate and RD 12 is set specifying the "No AR to the ALU" option.

In summary, UA1 goes high to gate the contents of the Arithmetic Register to the ALU during execution of all micro-instruction except non-immediate arithmetic or logical instructions specifying the "No AR to the ALU" option, or during execution of Branch or Load.

3.3.8 System Control Register (SCR). The System Control Register (SCR) is an eight bit register shown on FS22-17 and 18. The System Control Register is used to initiate control operations in device controllers, and to define the data present in the System Data Register (SDR). SCR can be considered an eight bit "instruction" register for the device controllers. The six most significant bits of SCR each have a specific meaning. System Control bit 6 is unassigned.

SCR bit zero (ADRS$\overline{0}$) is the device address bit. When SCR contains X'80', the SDR contains the address of a device.

SCR bit one (DA$\overline{0}$) is the data available bit. When SCR contains X'40', the SDR contains data for the device previously addressed.

SCR bit two (DR$\overline{0}$) is the data request bit. When SCR contains X'20', the Processor is requesting data from the device previously addressed. The data byte will be gated into SDR.

SCR bit three (SR$\downarrow$0) is the status request bit. When SCR contains X'10', the Processor is requesting the status of the device previously addressed. The Status Byte will be gated into SDR.

SCR bit four (CMD$\downarrow$0) is the command bit. When SCR contains X'08', SDR contains a command for the device previously addressed.

SCR bit five (RACK$\downarrow$0) is the return acknowledge bit. When SCR contains X'04', the existing external interrupt is recognized, and the address of the interrupting device is gated into SDR.

SCR bit seven (CL07$\downarrow$0) is the primary power fail bit. A ground indicates a primary power fail. This is only available if the system is equipped with the Primary Power Fail option.

The SCR register is composed of RS flip-flops with single-to-double rail converters on the inputs. SCR has inputs from the S Bus and the initialize control circuit.

SCR is cleared when the system is initialized by the SCLR$\downarrow$0 signal.

The S Bus is loaded into SCR with the signal LSC$\downarrow$1. LSC$\downarrow$1 is generated on FS 22-22 by three inverters whose outputs are AND tied. The three inputs are the System Clock (CLOD), BANK$\downarrow$0, and LR14$\downarrow$0. LR14$\downarrow$0 is low when the Destination Address Decoder decodes an address of X'E'. BANK$\downarrow$0 is low if the BANK flip-flop is set. The BANK flip-flop would be set to address SCR. When CLOD, BANK$\downarrow$0, and LR14$\downarrow$0 are all low, LSC$\downarrow$1 goes high and gates the S Bus into SCR.

The SCR outputs are wired on the back panel to the device controllers. The outputs are not gated or clocked. This is the 8 bit I/O control bus which is a one-way bus to the I/O controllers.

The buffered outputs SCxxx are OR tied to generate signals used in the I/O control circuit to generate I/O gating functions. If the Address bit, the Data Available bit or the Command bit of SCR is set, the signal SC14$\downarrow$0 will be low. If the Data Request bit or the Return Acknowledge bit of SCR is set, the signal SC25$\downarrow$0 goes low. If the Status Request bit of SCR is set, the signal SC03$\downarrow$0 goes low. SC25$\downarrow$0, SC03$\downarrow$0, and SC14$\downarrow$0 are used on FS 22-29. If any one of these signals is low, the time-out circuit is enabled. The time-out circuit is explained in detail under I/O control.

3.3.9 Systems Data Register (SDR). The Systems Data Register (SDR) is an eight bit register shown on FS22-13:16. The Systems Data Register serves as a buffer between the Processor and the device controller for all devices interfaced to the multiplexor channel. All data that is received from, or transmitted to, any device is held in the SDR.

The Systems Data Register is composed of RS flip-flops with single-to-double rail converters on the inputs. SDR has inputs from the S Bus, the I/O data bus, and the I/O control logic.

The contents of the S Bus are loaded into SDR when LSD$\downarrow$1 goes high. LSD$\downarrow$1 is generated on FS22-22 by three inverters with their outputs collector AND tied. The three inputs are the System Clock (CLOD), BANK$\downarrow$0, and LRAH$\downarrow$0. LRAH$\downarrow$0 is low when the Destination Address Decoder decodes an address of X'D'. BANK$\downarrow$0 is low when the BANK flip-flop is set. The BANK flip-flop is set when addressing SDR.

When CLOD, BANK$\downarrow$0 and LRAH$\downarrow$0 are all low, LSD$\downarrow$1 goes high and gates the S Bus into SDR. The SDR receives information from the device controllers via the DRL Bus. This is a one-way bus with all device controllers as transmitters, and the Processor as the only receiver. The DRL Bus is terminated with

a single-to-double rail converter. The information on the DRL Bus is gated into SDR when the signal LDL$_{\emptyset}$1 is high. LDL$_{\emptyset}$1 (FS22-29J8) is high when three conditions are met. First, the SCR must have either the Data Request bit (SCR02) or the Return Acknowledge bit (SCR05) set causing SC25$_{\emptyset}$0 to go low, or the Status Request bit (SCR03) set, causing SC03$_{\emptyset}$0 to go low. Second, the SYN0 signal must be low, indicating that a SYNC pulse was returned from the device controller addressed. Third, the False Sync flip-flop must be reset, indicating that the sync pulse from the device controller was received prior to generating a false sync. If these conditions are met, LDL$_{\emptyset}$1 goes high and gates the contents of the DRL Bus into SDR.

The other input to the System Data Register is WRSD$_{\emptyset}$1 from the I/O control logic. When WRSD$_{\emptyset}$1 is high, SDR05 is set and all other bits of SDR are reset.

WRSD$_{\emptyset}$1 is generated on FS22-29. The three inputs used to produce WRSD$_{\emptyset}$1 are SC03$_{\emptyset}$0, FSYN$_{\emptyset}$1, and CL$_{\emptyset}$1. SC03$_{\emptyset}$0 is low if the Status Request bit (SCR03) is set. FSYN$_{\emptyset}$1 is high if a False Sync was generated by the time-out logic, indicating that the device addressed did not respond with a sync within 42 microseconds from the time the Status Request was initiated. CL$_{\emptyset}$1 is the system clock. When SC03$_{\emptyset}$0 is low and CL$_{\emptyset}$1 and FSYN$_{\emptyset}$1 are high, WRSD$_{\emptyset}$1 goes high setting bit 5 of SDR and resetting all other bits.

The Systems Data Register can be output to the B Bus or to the device controllers. The contents of SDR are unloaded onto the B Bus when the signal USD$_{\emptyset}$1 is high. USD$_{\emptyset}$1 is high if the Source Register Decoder decodes the hex address D as the Source specified by a micro-instruction.

The contents of SDR are always present at the device controllers via the Data Available Lines (DAL00:DAL07). The DAL lines are generated by inverting the DC outputs of SDR with power gates. These DC outputs create a false bus which is wired on the back panel to all of the device controllers. The eight DAL lines make up a one-way bus with the Processor as the transmitter and all device controllers as receivers. Data on the DAL lines will only be received by the device controller addressed.

3.4 Arithmetic Logic Unit (ALU)

The Arithmetic Logic Unit (ALU) consists of logic to perform all required arithmetic, shift, and Boolean connect functions. The ALU is byte oriented, and may be divided into three functional groups: an A Register, summing logic, and carry logic. See Figure 2. The B Bus provides the operand for shift operations, and one of the operands for arithmetic and Boolean connect operations. The second operand for arithmetic and Boolean connect operations is the contents of the A Register. The A Register is loaded from the S Bus, prior to the arithmetic or Boolean connect operation.

The ALU receives seven control inputs from the op-code decoder. These inputs are briefly defined in the following list, and are summarized on the table associated with Figure 2.

Signal Designation	Definition
K1	Subtract
L1	Add or Subtract
M1	All Except Shifts, AND Immediate, and Load Complement
N1	Add, Subtract, and Load Complement
P1	All Except Exclusive OR
SL1	Shift Left
SR1	Shift Right

Each of the micro-instructions which uses any portion of the ALU is described in the following paragraphs. Unless otherwise noted, all gate references in the following paragraphs refer to the arbitrary designations on Figure 2.

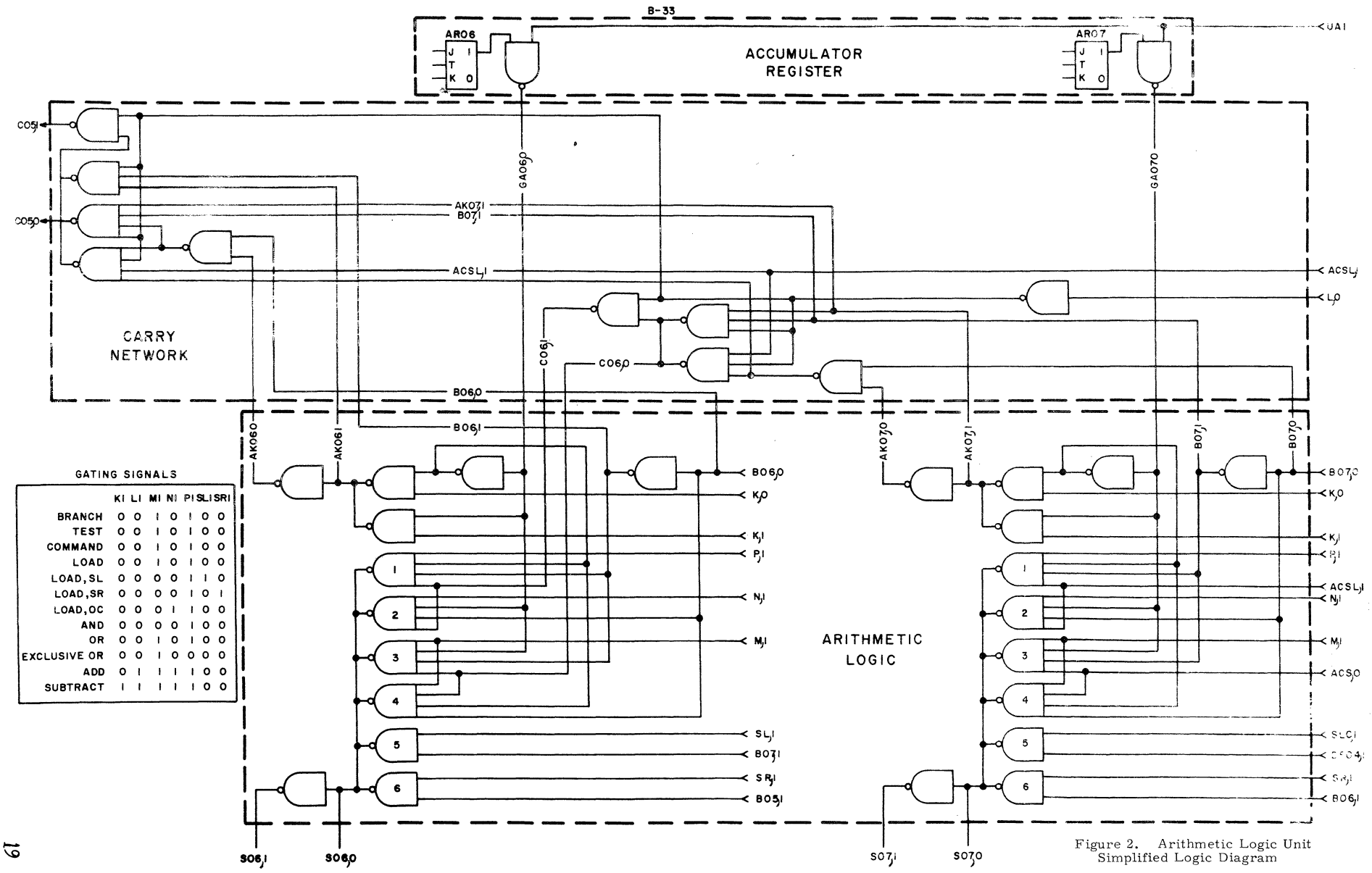


Figure 2. Arithmetic Logic Unit Simplified Logic Diagram

3.4.1 Load, Load Immediate, Branch, Command and Test. These micro-instructions cause the contents of the B Bus to pass unaltered through the ALU to the S Bus. This function is performed by Gate 3 of the summing circuit (Figure 2).

Since these are not arithmetic micro-instructions, the carry circuit is disabled ($L_J1 = 0$). This forces all carry circuit outputs high. Also, since these micro-instructions do not have a second operand, the A Register (AR) is not unloaded ($UA_J1 = 0$). This forces the GAn_J0 leads high, and the GAn_J1 leads low. Signal M_J1 is high at this time, and Signals N_J1, SL_J1, and SR_J1 are low as shown on the table on Figure 2.

Gate 3 then has high inputs from GAn_J0, Cn_J0, and M_J1. Therefore, if Bn1 is also high, all inputs are high to produce a low out. Gates 1, 2, 4, 5, and 6 are all disabled by Signals GAn_J1, N_J1, SL_J1, and SR_J1 which are low at this time.

3.4.2 Shift Left. This micro-instruction shifts the contents of the B Bus left one place, and places the shifted data onto the S Bus. The bit that is shifted out of bit 0 is losw unless the Carry Out option (CO) is specified. If CO is specified, the bit is stored in the Carry Store flip-flop DF04. Bit 7 picks up a zero unless the Carry In option (CI) is specified and DF04 is set. Gate 5 of the summing circuit is used to perform the Shift Left function.

Signals SL_J1 and P_J1 are high during a Shift Left, and Signals K_J1, L_J1, N_J1, M_J1, and SR_J1 are low (see the table on Figure 2).

Gate 5 uses SL_J1 to gate each bit of the B Bus onto the next most significant position of the S Bus. For example, B07 is gated onto S06, B06 is gated onto S05, etc. In the case of bit 7 of the ALU, SLC_J1 is high if the CI option is specified, to gate the Carry Store flip-flop ($DF04_J1$) onto bit 7 of the S Bus. If the CO option is specified, and $B0_J7_J1$ is high, SDF4,0 is low to set the Carry Store flip-flop DF04. Gates 1, 2, 3, 4, and 6 are all disabled by Signals N_J1, M_J1, and SR_J1 which are low during the Shift Left micro-instruction.

3.4.3 Shift Right. This micro-instruction shifts the contents of the B Bus right one place, and places the shifted data onto the S Bus. The bit that is shifted out of bit 7 is lost unless the Carry Out option (CO) is specified. If the CO option is specified, the bit shifted out of bit 7 is stored in the Carry Store flip-flop, DF04. Bit 0 is low unless the CI option is specified and DF04 is set. Gate 6 of the summing circuit performs the Shift Right function.

Signal SR_J1 is high at this time and Signals N_J1, M_J1, and SL_J1 are low. Signal SR_J1 is applied to Gate 6 to gate each bit of the B Bus onto the next least significant position of the S Bus. For example, B00 is gated onto S01, B01 is gated onto S02, etc. In the case of bit 0 of the ALU, SRC_J1 is high if the CI option is specified to gate the Carry Store flip-flop ($DF04_J1$) onto bit 0 of the S Bus. Gates 1, 2, 3, 4, and 5 are all disabled by Signals P_J1, N_J1, M_J1, and SL_J1 which are low during a Shift Right micro-instruction. If the CO option is specified, and $B0_J7_J1$ is high, SDF4,0 is low to set the Carry Store flip-flop, DF04.

3.4.4 Load Ones Complement. This micro-instruction complements the contents of the B Bus, and places the complemented data onto the S Bus. The Carry In (C_J1) option has no meaning for this micro-instruction. If the Carry Out (CO) option is specified, the Carry Store flip-flop, DF04, is reset. Gate 2 of the summing circuit is used to perform this function.

Since this is not an arithmetic micro-instruction, the carry circuit is disabled ($L_J1 = 0$). This forces all outputs of the carry circuit high. Also, since this micro-instruction does not have a second operand, the A Register (AR) is not unloaded ($UA_J1 = 0$). This

forces the GAn,0 leads high and the GAn,1 leads low. Signal N,1 is high at this time, and Signals M,1, SL,1, and SR,1 are low as shown on the table on Figure 2.

Gate 2 is enabled by Signals GAn,0, Cn, and N,1 which are all high. If Bn is a one, there is a low into the gate, a high out on Sn,0, and a low on Sn,1. If Bn is a zero, the gate has all high inputs, a low out on Sn,0, and a high on Sn,1. Gates 1, 3, 4, 5, and 6 are all disabled by Signals GAn,1, M,1, SL,1, and SR,1, which are all low. If the CO option is specified, Signal SVSC,0 is active. Signal SVSC,0 generates a signal which resets the Carry Store flip-flop, DF04. (See the schematics in Volume 2 of this manual.)

3.4.5 AND And AND Immediate. These micro-instructions logically AND each bit of the B Bus with the corresponding bit in the A Register (AR), placing the result onto the S Bus. Gate 1 of the summing circuit performs these functions.

During these micro-instructions, the carry circuit is disabled (L,1 = 0). This forces all outputs of the carry circuit high. Because these micro-instructions require a second operand, the A Register (AR) is unloaded (UA,1 = 1). The outputs from the AR are available to the summing circuit. Signal P,1 is high at this time, and Signals N,1, M,1, SL,1, and SR,1 are low.

3.4.6 OR And OR Immediate. These micro-instructions logically OR each bit of the B Bus with the corresponding bit in the A Register (AR), placing the result onto the S Bus. Gates 1, 3, and 4 of the summing circuit perform these functions.

During these micro-instructions, the carry circuit is disabled (L,1 = 0). This forces all outputs of the carry circuit high. Because these micro-instructions require a second operand, the A Register (AR) is unloaded (UA,1 = 1). The outputs from the AR are available to the summing circuit. Signals P,1 and M,1 are both high and Signals N,1, SL,1, and SR,1 are low.

Gate 1 has high inputs from P,1, and Cn. If Bn and GAn are also both high, the Gate 1 output is low. Gate 1 checks for the condition where Bn and the corresponding bit of the AR are both set. Gate 3 has high inputs from M,1 and Cn. If Bn is set and GAn is reset, all Gate 3 inputs are high, producing a low output. Gate 3, then checks for the condition where Bn is a one and the corresponding bit of the AR is reset. Gate 4 has high inputs from M,1 and Cn. If GAn is set and Bn is a zero, all inputs to Gate 4 are high, producing a low output. Gate 4, then, checks for the condition where Bn is a zero and the corresponding bit of the AR is set.

The output equation to the S Bus for these three gates during the OR and OR Immediate micro-instruction is:

$$\overline{(P,1)(M,1)\ (Bn,1)(GAn,1)+(Bn,1)(GAn,0)+(Bn,0)(GAn,1}$$

Gates 2, 5, and 6 are disabled by Signals N,1, SL,1, and SR,1 which are low.

3.4.7 Exclusive OR And Exclusive OR Immediate. These micro-instructions logically Exclusive OR each bit of the B Bus with the corresponding bit in the A Register (AR), placing the result onto the S Bus. Gates 3 and 4 of the summing circuit perform this function.

During these micro-instructions, the carry circuit is disabled (L,1 = 0). This forces all outputs of the carry circuit high. Because these micro-instructions require a second operand, the A Register (AR) is unloaded (UA,1 = 1). The outputs from the AR are available to the summing circuit. Signal M,1 is high, and Signals P,1, N,1, SL,1, and SR,1 are low.

Gate 3 has high inputs from M,1 and Cn. If Bn is a one and GAn is reset, all Gate 3 inputs are high, producing a low output. Gate 3, then, checks for the condition where Bn is a one and the corresponding bit of the AR is reset. Gate 4 checks for the condition where Bn is reset and the corresponding bit of the AR is set. Gates 1, 2, 5 and 6 are all disabled by Signals B,1, N,1, SL,1, and SR,1 which are low.

The output equation to the S Bus for these two gates during these micro-instructions is:

$$M,1\ (Bn,0\ \ GAn,1\ +\ Bn,1\ \ GAn,0)$$

3.4.8 Add And Subtract. When an addition or a subtraction is performed, each adder-subtracter stage looks at three inputs: one bit from each of the two operands plus the carry bit generated from the previous stage. For explanation purposes the inputs are designated: Cn (carry or borrow bit coming in from some previous stage), An (the first operand bit), and Bn (the second operand bit).

The adder-subtracter unit has two outputs: Sum and Carry. During Subtract, however, the sum output can be looked at as the difference output, while the carry output can be looked at as the borrow output.

Prior to the adder-subtracter description, the terminology and method of signal designations employed in this system must be understood. The following conventions apply to both the descriptions in this section, and to the functional schematics provided in Volume 2 of this manual.

The operands are designated B00 through B07, and GA00 through GA07 indicating the B Bus and Gated A Register respectively. The least significant bits are B07 and GA07. The Sum bits are designated S00 through S07, and correspond to the operand bits which produce them. For example, B07 and GA07 are combined to form S07, the least significant Sum bit. The Carry bits are designated CSV and C00 through C06. The CSV (Carry Save) bit is the carry from the most significant bits (B00 and GA00). Each remaining Carry bit is designated in accordance with the stage it carries into. For example, B07 and GA07 generate C06, which is summed with B06 and GA06.

To generate an output in a single stage adder-subtracter unit, the following conditions can exist:

Cn	0	0	0	0	1	1	1	1
An	0	0	1	1	0	0	1	1
Bn	0	1	0	1	0	1	0	1
Sum	0	1	1	0	1	0	0	1
Difference	0	1	1	0	1	0	0	1
Carry out	0	0	0	1	0	1	1	1
Borrow out	0	1	0	0	1	1	0	1

Sum-Difference: Note on the table that the sum and the difference are equal in a single stage adder-subtracter. The equation for either the sum or the difference is:

$$AnBnCn + An\overline{Bn}\overline{Cn} + \overline{An}Bn\overline{Cn} + \overline{An}\overline{Bn}Cn$$

Carry out: According to the table, the equation for the carry out is:

$$AnBn\overline{Cn} + An\overline{Bn}Cn + \overline{An}BnCn + AnBnCn$$

the equation can be reduced to:

$$AnCn + AnBn + BnCn$$

This means that any combination of two or more bits set will generate a carry out. The equation can be factored to read:

$$An(Bn+Cn) + BnCn$$

Borrow out: In subtraction, according to the table, the equation for borrow out can be written as:

$$A_nB_nC_{n-1} + \overline{A}_nB_nC_{n-1} + \overline{A}_nB_n\overline{C}_{n-1} + \overline{A}_n\overline{B}_nC_{n-1}$$

The equation can be reduced to:

$$\overline{A}_nB_n + \overline{A}_nC_n + A_nC_n$$

This equation can be factored to read:

$$\overline{A}_n(B_n+C_n) + B_nC_n$$

To simplify the equations, Add and Subtract are designated in the equations as:

$\overline{K}$ = Add

K = Subtract

The equations for carry and borrow can be combined to form the equation

$$C_n = \overline{K}A_{n+1}(B_{n+1} + C_{n+1}) + K\overline{A_{n+1}}(B_{n+1} + C_{n+1}) + B_{n+1}C_{n+1}$$

The previous equation can be expanded to produce:

$$C_n = \overline{K}A_{n+1}B_{n+1} + \overline{K}A_{n+1}C_{n+1} + KA_{n+1}B_{n+1} + KA_{n+1}C_{n+1} + B_{n+1}C_{n+1}$$

This equation can be factored to produce:

$$C_n = (\overline{K}A_{n+1} + K\overline{A_{n+1}})(B_{n+1} + C_{n+1}) + (B_{n+1}C_{n+1})$$

To further simplify this discussion, define:

$$\overline{K}A_{n+1} + K\overline{A_{n+1}} = AK_{n+1}$$

In order to make a one bit "look ahead" carry borrow out of the last Cn equation, Cn+1 must be expressed in terms of AKn+2, Bn+2, and Cn+2.

$$C_{n+1} = AK_{n+2}B_{n+2} + AK_{n+2}C_{n+2} + B_{n+2}C_{n+2}$$

Substitute this equation for Cn+1 in the Cn equation above to form the following one bit "look ahead" carry borrow equation:

$$C_n = AK_{n+1}B_{n+1} + AK_{n+1}AK_{n+2}B_{n+2} + AK_{n+1} + AK_{n+2}C_{n+2} + AK_{n+1}B_{n+2}C_{n+2} + B_{n+1}AK_{n+2} + B_{n+2} + B_{n+1}AK_{n+2}C_{n+2} + B_{n+1}B_{n+2}C_{n+2}$$

3.5 I/O Control

The Processor I/O Control logic is shown on FS22-29. Refer to the I/O Section of this manual for more information on the I/O System. The EXECUTE pushbutton is shown in the top left area of FS22-29. On a standard Digital System, the outputs from the EXECUTE Switch, ESNC0 and ESNO0, leave the top of the sheet and go to the Display Controller. The Display Controller, in turn, generates a Console Attention signal (CATN0) which is returned to the Processor as shown in FS22-29D9. The signal is inverted and sent to the Test Logic shown on FS22-23. The gate shown in FS22-29B8 provides the Attention signal from all other peripheral devices. The remainder of the logic on the left half of FS22-29 is not used in a standard system with a Display.

Although the Display Panel is standard on GE-PAC 30 Digital Systems, systems without the Display are available on special order. Such systems are designated Auto-Load Systems, and are provided with a smaller panel which contains the following controls and indicators:

1. POWER Switch and as associated indicator
2. EXECUTE pushbutton
3. INITIALIZE pushbutton
4. AUTO LOAD Switch
5. Wait indicator

All switches and indicators except AUTO LOAD perform the same functions as in a system with a Display. The AUTO LOAD Switch permits entry of the 50 Loader from tape (rather than manual entry as performed with a standard system). Refer again to FS22-29. On an Auto Load system, the EXECUTE Switch outputs are applied to cross-coupled gates which provide the T input to the flip-flop shown in area C5. The flip-flop generates Console Attention (CATN,1) in an Auto Load system. The Auto Load system tests CATN,1 to determine whether to halt or continue. System Clear (SCLR,0) sets the flip-flop initially via the SJK,0 input (area A5). The system therefore runs when power is applied or the INITIALIZE pushbutton is depressed. The first time EXECUTE is depressed, the flip-flop is toggled reset, and CATN,1 goes low. The system halts after the next sequence. The next time the EXECUTE pushbutton is depressed, the flip-flop is set, CATN,1 goes high, and the system runs. Note the gate in area F3. This gate produces a Single (SNGL,0) signal in an Auto Load System when the AUTO LOAD Switch is depressed. Note, again, that this paragraph applies only to Auto Load Systems.

The right half of FS22-29 is common to both standard and Auto Load systems. Note the three inputs in area H2 and K2. The inputs indicate that the following I/O instructions are to be performed.

SC03,0 - SR

SC25,0 - DR or RACK,0

SC014,0 - ADRS, DA, or CMD

If any of the signals are active, the gate at K3 provides an input to the False Sync circuit. The False Sync circuit assures that the Processor is not hung up waiting for a Sync response to an I/O command in case the peripheral device addressed is unable to execute the command, or is non-existent. The False Sync flip-flop (area M5 and N5) is set to generate FSYN,1 if SYN,0 is not received within approximately 42 microseconds after an Input/Output operation is initiated. The high output from the gate at K3 applies a low to the K input of the flip-flop. The J input is held low also by the output from the gate at P3. The capacitors shown in area M2 charge to the threshold level of the gate at N3 about 42 microseconds after the I/O command is sensed. Normally, by this time SYN,0 has been received from the addressed device controller, indicating that the specified operation has been completed. The micro-code program tests for TSYN,1 and resets the I/O Command bit in SCR when SYN,0 is received. The K input to the flip-flop is therefore high at this time, and the J input is held low. The high output from the gate at P3 therefore has no effect. If SYN,0 is not received, however, the J input goes high permitting the FSYN flip-flop to set. If FSYN,1 is generated, the Overflow bit in DFR is set by the gate at P7. This provides the micro-code program with an indication of the error. If this is an SR operation, and FSYN,1 is generated, X'04' is written in the SDR by WRSD,1 (area N9). This creates a status byte with the examine status bit set as an indication of the error for the user's program. The gate at M7 produces a TSYN,1 indication if either SYN,0 or FSYN,0 is active. The micro-code program tests for TSYN,1 and resets SCR as discussed previously. An LDL,1 signal is generated by the gates at H7 and H8 on an SR, DR, or RACK,0 condition. Note that LDL,1 is inhibited if False Sync is generated.

4. MAINTENANCE

This Section describes maintenance procedures which may be used to check and, if necessary, adjust the Processor

NOTE

Two optional maintenance aids available from GE-PAC 30 greatly facilitate maintenance on the Processor: the Systems Test Set, and the X-Ray ROM. GE-PAC 30 urges all users who intend to maintain their own system to purchase both a Systems Test Set and an X-Ray ROM. Maintenance procedures using these aids are provided with the aids.

4.1 Clock timing

Use the following procedure to check system clock timing.

1. Force the RD register to zero by grounding the SCLR,0B signal at pin 118-1900.

CAUTION

Exercise care in grounding points in the digital system as some components may be damaged if the wrong pins are grounded or pins are shorted to each other.

2. Use an oscilloscope to check that the following Y-Switch outputs are 110 to 150 nanoseconds wide.

YS00,1	017-RMS
YS01,1	117-RMS
YS02,1	217-RMS
YS03,1	016-RMS
YS04,1	115-RMS
YS05,1	015-RMS
YS06,1	216-RMS
YS07,1	116-RMS

If any of the switch outputs are not within the 110 to 150 nanosecond range, check the associated logic. If the logic is good, synchronize the oscilloscope on CO2 Test Point D and adjust the CD,0 pulse width to obtain the correct output. (See Figure 3.)

CAUTION

The adjustments described in this Section are carefully set at the factory using sophisticated test equipment not normally available in the field. The adjustments are very stable and should not require field adjustment. The adjustments should be changed only after the check provided indicates that they are out of tolerance and there are no faulty components in the system.

3. Force RAH07 to binary ONE by grounding pin 216-1 on RMS. Check that the following Y-Switch outputs are 110 to 150 nanoseconds wide.

YS08,1	215-RMS
YS09,1	014-RMS
YS10,1	114-RMS
YS11,1	214-RMS
YS12,1	012-RMS
YS13,1	213-RMS
YS14,1	113-RMS
YS15,1	013-RMS

If any of the switch outputs are not within tolerance, proceed as outlined previously in Step 2.

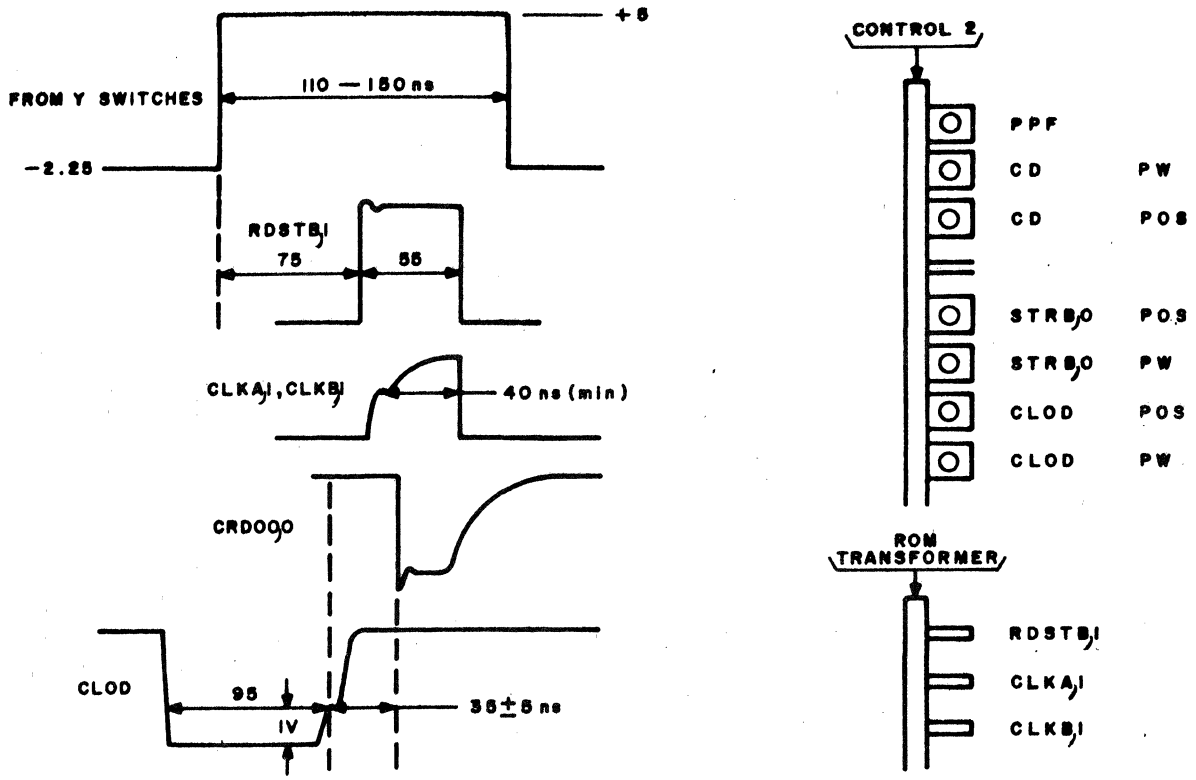


Figure 3. Processor Timing Data

4. Compare RDSTB,1 on the ROM transformer board (Figure 3) and YS15,1 (pin 013-0 on RMS). RDSTB,1 should start 75 ±10 nanoseconds after YS15,1. If out of tolerance, and no faulty components are found, adjust STRB,0 position to provide the 75 nanosecond difference.

5. Check that STRB,0 is 45 to 60 nanoseconds wide. If STRB,0 pulse width is out of tolorance, and no faulty components are found, adjust STRB,0 pulse width (Figure 3) to 50 nanoseconds.

6. Check the three strobes on the ROM transformer board (Figure 3) and verify that RDSTB,1 is coincindental with, or starts later than CLKA,1 and CLKB,1.

7. Observe all ROM readouts (listed on Table 1). False outputs on the CRDXX,0 leads should be less than 25 nanoseconds wide at the 2 volt level.

8. Compare the SRD03,0 ROM output (pin 105 of RMT) and CLOD (CO2 Test Point F). The rising edge of CLOD should occur 35 ±5 nanoseconds before the falling edge of SRD03,0. If not, adjust CLOD position (Figure 3).

9. Check that CLOD is 95 ±5 nanoseconds wide. If necessary, adjust CLOD pulse width (Figure 3).

10. Remove the grounds installed in Steps 1 and 3.

TABLE 1. ROM OUTPUTS

OUTPUT	PIN	OUTPUT	PIN
CRD00,0	020-RMT	SRD00,0	004-RMT
CRD01,0	120-RMT	SRD01,0	104-RMT
CRD02,0	019-RMT	SRD02,0	005-RMT
CRD03,0	119-RMT	SRD03,0	105-RMT
CRD04,0	018-RMT	SRD04,0	006-RMT
CRD05,0	118-RMT	SRD05,0	106-RMT
CRD06,0	017-RMT	SRD06,0	007-RMT
CRD07,0	117-RMT	SRD07,0	107-RMT
CRD08,0	016-RMT	SRD08,0	008-RMT
CRD09,0	116-RMT	SRD09,0	108-RMT
CRD10,0	015-RMT	SRD10,0	009-RMT
CRD11,0	115-RMT	SRD11,0	109-RMT
CRD12,0	014-RMT	SRD12,0	010-RMT
CRD13,0	114-RMT	SRD13,0	110-RMT
CRD14,0	013-RMT	SRD14,0	011-RMT
CRD15,0	113-RMT	SRD15,0	111-RMT

4.2 Overall Processor Test

Use the 30-1 Test Program, described in Publication Number 06-005R01A12, to perform a comprehensive test of the Processor. The program description and a listing are provided in the Programming Manual, Publication Number 29-013R02.

5. PROCESSOR MNEMONICS

The following list provides a brief description of each mnemonic found in the Processor. The FS22 or FS44 source of each signal is also provided

MNEMONIC	MEANING	LOCATION
ACS	Carry False Input to the Least Significant Bit of the Adder	FS22-25E8
ACSL	Carry True Input to the Least Significant Bit of the Adder	FS22-25D8
ADRS	Address Bit of System Control Register	FS22-17C7
AK00-AK07	A Register Inputs to Carry Circuits	FS22-32B9

MNEMONIC	MEANING	LOCATION
AR00-AR07	A Register Output Signals	FS22-7B6
ATN	Attention - Request for Interrupt by Any External Device	FS22-29B9
AUTOLD	Auto Load Command for Non Display Machines	FS22-29F2
B00-B07	B Bus	FS22-8, 9, 11, 12, and 13 FS44-2 and 3
BANK	Bank Flip-Flop - High for Addressing SDR and SCR and Low for Addressing RAL and RAH	FS22-24F6
BRA	Branch Micro-Instruction	FS22-26B8
CATN	Console Attention - the Interrupt from Display	FS22-29C9
CD	Current Drive Pulse for ROM	FS22-28K8
CL	Clock	FS22-28C2
CLKOFF	Clock Off	FS22-28P4
CLKSTP	Clock Stop	FS22-28D5
CLOD	(System)Clock D	FS22-28G8
CLOE	(System)Clock E	FS22-28H8
CMD1A	Command Micro-Instruction (Delayed)	FS22-23C8
CMD1B	Command Micro-Instruction	FS22-23D8
CMD	Command Bit of System Control Register	FS22-18C7
C00-C07	Carry Outputs	FS22-30F9
CRD00-CRD15	Clear ROM Data Register Inputs from the ROM	FS44-1C2
CSV	Carry Save from Most Significant Bit of the Adder	FS22-30C8
DA	Data Available Bit of System Control Register	FS22-17G7
DAL00-DAL07	Data Available Lines from System Data Register	FS22-13E8

MNEMONIC	MEANING	LOCATION
DF00-DF07	DO Flag Register Outputs	FS22-11D7
DRD00-DRD02	DO ROM Data (0:2) - Reflect Op-Codes from ROM Data Register or DO Flag Register	FS44-1C8
DRL00-DRL07	Data Request Lines to System Data Register from Devices.	FS22-13F9
DR	Data Request Bit of System Control Register	FS22-17L7
ENBC	Enable Console Attention	FS22-29E2
ENDL	Enable Data Lines	FS22-20P3
ESNC	Execute Switch Contact Normally Closed	FS22-29B2
ESNO	Execute Switch Contact Normally Open	FS22-29D2
EXTCLK	External Clock Input	FS22-28R5
FSYN	False Sync	FS22-29R6
GA000-GA070	Gated A Register Outputs to the Summing Circuit	FS22-7B9
HLFCLK	Half Clock	FS22-28R3
INHINC	Inhibit Increment of ROM Address	FS44-4B9
K	Gating Lead High for Subtract	FS22-26E8
L	Gating Lead High for Add or Subtract	FS22-26E8
LA1A	Load A Register (0:3) with the Contents of the S Bus (0:3)	FS22-22G7
LA1B	Load A Register (4:7) with the Contents of the S Bus (4:7)	FS22-22N7
LD	Load Micro-Instruction	FS22-25G8
LDF	Load DO Flag Register (0:7) with the Contents of the S Bus (0:7)	FS22-21B8
LDL	Load Data Lines Signal	FS22-29H9

MNEMONIC	MEANING	LOCATION
LMAH	Load Memory Address Higher with the S Bus	FS22-21H8
LMAL	Load Memory Address Lower with the S Bus	FS22-21F8
LMDH	Load Memory Data Higher with the S Bus	FS22-21K8
LMDL	Load Memory Data Lower with the S Bus	FS22-21J8
LMST	Load Memory Start Clock Inhibit - Stops System Clock if a Memory Cycle is in Progress and the Micro-Code Program Attempts to Alter MD or MA or start another memory operation.	FS22-28J2
LRAH	Load ROM Address Higher with the S Bus	FS22-21N8
LRAL	Load ROM Address Lower with the S Bus	FS22-28G8
LR140	Load Register 14 (SCR or RAL) with the S Bus	FS22-21E8
LR0-LR7	Load General Register 0-7 with the S Bus	FS22-20D8
LSC	Load System Control Register with the S Bus	FS22-22C7
LSD	Load System Data Register with the S Bus	FS22-22J7
M	Gating Lead High for all Micro-Instructions Except AND or LOAD if Shift Left, Shift Right, or Load One's Complement is Specified.	FS22-26L8
MBY	Memory Busy	FS22-28F2
MDAV	Memory Data Available	FS22-28D2
N15	Negative 15 Volts	FS22-27F3
N	Gating Lead High for Add, Subtract, or Load One's Complement	FS22-26F8
OSC	Crystal Oscillator Output	FS22-28K2
P5	Plus 5 Volts	FS22-27J5
P15	Plus 15 Volts	FS22-27C7
P	Gating Lead High for all Micro-Instructions Except Exclusive OR	FS22-26H8

MNEMONIC	MEANING	LOCATION
PDOWN	Power Down from Power Fail Detector, or Initialization, or N15 Failure	FS22-27F6
PFF	Clear Parity Fail Flip-Flop Command	FS22-24K8
POFF	Power Off - Power Switch Contact	FS22-29A2
POW	Command Power Down	FS22-24N9
PPF	Primary Power Fail Indicator to Test Logic	FS22-27B9
R0-R7	Read General Register (0-7)	FS22-8J1
RACK	Return Acknowledge Bit of System Control Register	FS22-18G7
RAH04-RAH07	ROM Address Higher Register Outputs	FS44-6E8
RAL00-RAL07	ROM Address Lower Register Outputs	FS44-4E8
RAWCLK	Raw Clock Output	FS22-28N5
RCL	ROM Clock Used to Increment or Gate S Bus into RAL	FS22-28F8
RD00-RD15	ROM Data Register Outputs	FS44-1C8
RJK	Reset JK Flip-Flop - Tied to System Clear in a Display Machine	FS22-29E2
S00-S07	S Bus	FS22-7B2
SCLR	System Clear	FS22-27H7
SC00-SC07	System Control Register Outputs	FS22-17B7
SC140	System Control Register Specifies Data Request or Return Acknowledge	FS22-17B7
SC250	System Control Register Specifies Data Request or Return Acknowledge	FS22-17J7
SC030	System Control Register Specifies Status Request	FS22-17P7
SDF4	Set DO Flag Register Bit 4	FS22-10N8
SDF5	Set DO Flag Register Bit 5	FS22-10J8
SDF6	Set DO Flag Register Bit 6	FS22-10F8, 23R7
SDF7	Set DO Flag Register Bit 7	FS22-10C8, 23R8

MNEMONIC	MEANING	LOCATION
SJK	Set JK Flip-Flop - Tied to System Clear in a Non Display Machine	FS22-29B5
SL	Shift Left	FS22-25M8
SLC	Shift Left with Carry In	FS22-25N8
SNGL	Single Users Instruction Cycle	FS22-29F9
SR1	Shift Right	FS22-25H8
SR0	Status Request Bit of System Control Register	FS22-17R7
SRC	Shift Right with Carry In	FS22-25K8
SRD00-SRD15	Set ROM Data Register (0:15) - Inputs from ROM	FS44-1C2
STF	Set Flags	FS22-25B8
STFL	Set Flags for Add or Subtract	FS22-25C8
STPFS	Start Test for False Sync	FS22-29K4
STPSYS	Stop System Clocks	FS22-27H4
STRB	Strobe for ROM	FS22-28J8
STRT	Start Memory Cycle	FS22-24E8
SV	Set Overflow Flag on False Sync	FS22-29P9
SVAC	Save Add Carry	FS22-25A8
SVSC	Save Shift Carry	FS22-25F8
SWTAL	Set Wait Alarm Command	FS22-24N9
SYN	Sync from Device Controller	FS22-29H2
TST	Test Micro-Instruction	FS22-23D8
TSYN	Testable Sync (OR of SYN and False SYN)	FS22-29M9
UA	Unload A Register	FS22-26N9

MNEMONIC	MEANING	LOCATION
UDF	Unload DO Flag Register onto the B Bus	FS22-19K8
UMDL	Unload Memory Data Register Lower onto the B Bus	FS22-19N8
UMDH	Unload Memory Data Register Higher onto the B Bus	FS22-19P8
UR0-UR7	Unload General Register 0-7 onto the B Bus	FS22-19B8
USD	Unload System Data Register onto the B Bus	FS22-19L8
UT	Utility Flip-Flop	FS22-24H6
VP	Stop System Clock upon Initialization	FS22-27R6
W0-W7	Write the S Bus into General Register 0-7	FS22-9G2
WA	Wait Alarm	FS22-24M8
WR	Write to Memory	FS22-24D8
WRSD	Write System Data Register on False Sync	FS22-29N9
WUN	ONE - High if RAL 5, 6, 7 are All Set	FS44-4R3

THE GE-PAC 30-2 PROCESSOR
TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

ILLUSTRATIONS

THE GE-PAC 30-02 PROCESSOR

1. INTRODUCTION

This Section describes the 30-2 Processor. Refer to the General Description Section earlier in this manual for a description of the functional organization of a GE-PAC 30-2 Digital System. This Section includes a block diagram analysis, functional descriptions which reference schematics provided in Volume 2 of this manual, maintenance information, and a mnemonic list.

2. BLOCK DIAGRAM ANALYSIS

Figure 1 is a block diagram of a 30-2 Digital System. Processor operation is controlled by the Read-Only-Memory (ROM). Micro-programming, hard-wired into the ROM, makes the 30-2 appear to have the capabilities of a much larger machine. When executing the instructions of the emulated computer, the micro-program directs the hardware to read the next instruction to be executed from core memory. The hardware is then allowed to decode the user's instruction and point the micro-program to the micro-subroutine that has been designed to perform the emulated instruction. The loop is closed by incrementing the program Location Counter and returning to the point in the micro-program that will fetch the next instruction from core memory.

The micro-program wired in the ROM consists of combinations of micro-instructions. Each micro-instruction performs a basic operation. The sixteen different micro-instructions the 30-2 executes, and their operation codes, are listed in Table 1. For a detailed explanation of each instruction, refer to the 30-2 Micro-Instruction Reference Manual, Publication Number 29-032.

TABLE 1. MICRO-INSTRUCTIONS

INSTRUCTION	OP-CODE
Decode	0000
Branch	0001
Test	0010
Command	0011
Load	0100
Load Immediate	0101
OR	0110
OR Immediate	0111
AND	1000
AND Immediate	1001
Exclusive OR	1010
Exclusive OR Immediate	1011
Add	1100
Add Immediate	1101
Subtract	1110
Subtract Immediate	1111

The Read-Only-Memory (ROM) contains the micro-code program to fetch and execute the user's instructions stored in Core Memory. The ROM is a high speed, non-volatile memory made up of pulse transformers. (See the ROM Section of this manual.)

The Processor executes micro-instructions extracted from the ROM by gating data through the Arithmetic Logic Unit (ALU). All data transfers within the Processor are made on the sixteen bit B and S Buses, shown at the top and bottom of Figure 1.

The ROM location of the micro-instruction is defined by the twelve bit address contained in the ROM Address Lower Register (RAL) and the ROM Address Slave Register (RAS). RAL is an eight bit micro-instruction location counter. It is loaded from the S Bus during a Branch instruction or when specified as a Destination Register, and incremented by one between the execution of all other micro-instructions. RAS is a four bit register that holds the ROM page address. It is loaded with the contents of the ROM Address Higher Register (RAH) whenever RAL is loaded from the S Bus. Neither the RAS nor the RAH is involved when the RAL is incremented. To transfer from one page (256 ROM words) to another, the destination page number is first loaded into RAH from the S Bus. When the location address is loaded into RAL, RAS is loaded from RAH. This insures that the new location and page addresses are sensed in the ROM address decoders simultaneously.

Every micro-instruction extracted from ROM is placed in the ROM Data Register (RD) where it remains while the instruction is being executed. RD is a sixteen bit micro-instruction register. It is reloaded from ROM at the completion of each micro-instruction. The outputs from RD are inputs to the Processor Control and Decoder Logic.

Within Control, the instructions are decoded to activate gating leads to the ALU. The source and destination addresses are decoded and the signals to unload registers to the B Bus and load registers from the S Bus are generated. Core memory actions are initiated from Control, and the signals directing the clock system and initialize circuits are also maintained. RD bits 0:3 are decoded as the operation code of the micro-instruction being executed.

RD bit 3 defines Immediate instructions. An Immediate instruction is one that has the data to be manipulated appended to the micro-instruction word itself. During execution of an Immediate instruction or a Branch instruction, RD bits 8:15 are gated onto B Bus bits 8:15, and treated as data.

The D field (RD4:7) specifies the Destination of the micro-instruction result formed in the ALU. The result is placed on the S Bus by the ALU and then gated from the S Bus to the Destination register specified by the D field.

The S field (RD8:11) selects the register to be gated to the B Bus. The selected register is the source of one operand to be used in arithmetic and logical operations. The other operand is always assumed to be in the A Register (AR).

The E field of the RD register (RD12:15) is an extended micro-operation modifier. This field permits control of such things as flag actions, shifts, I/O operations or phase actions. The D, S, and E fields are interpreted for Branch, Test, or Command micro-instructions. The S and E fields are interpreted differently for Immediate micro-instructions.

The instruction register (IR) is a sixteen bit register used to hold the user's instruction. The OP field or IR (0:7) holds the user's operation code. The meaning of the remaining bits depends on the type of user's instructions, but generally the following applies. The R1 field (8:11) specifies the Destination register of the user's result. The S Bus is gated into the General Register specified by the R1 field. The R2 field (12:15) selects the General Register to be gated to the B Bus as an operand.

The outputs from the Instruction Register are inputs to the Decoder. The Decoder consists of logic to interpret user instructions. Also, through the Decoder Read-Only-Memory (DROM), the micro-program is guided to the subroutine designed to execute the emulated instruction.

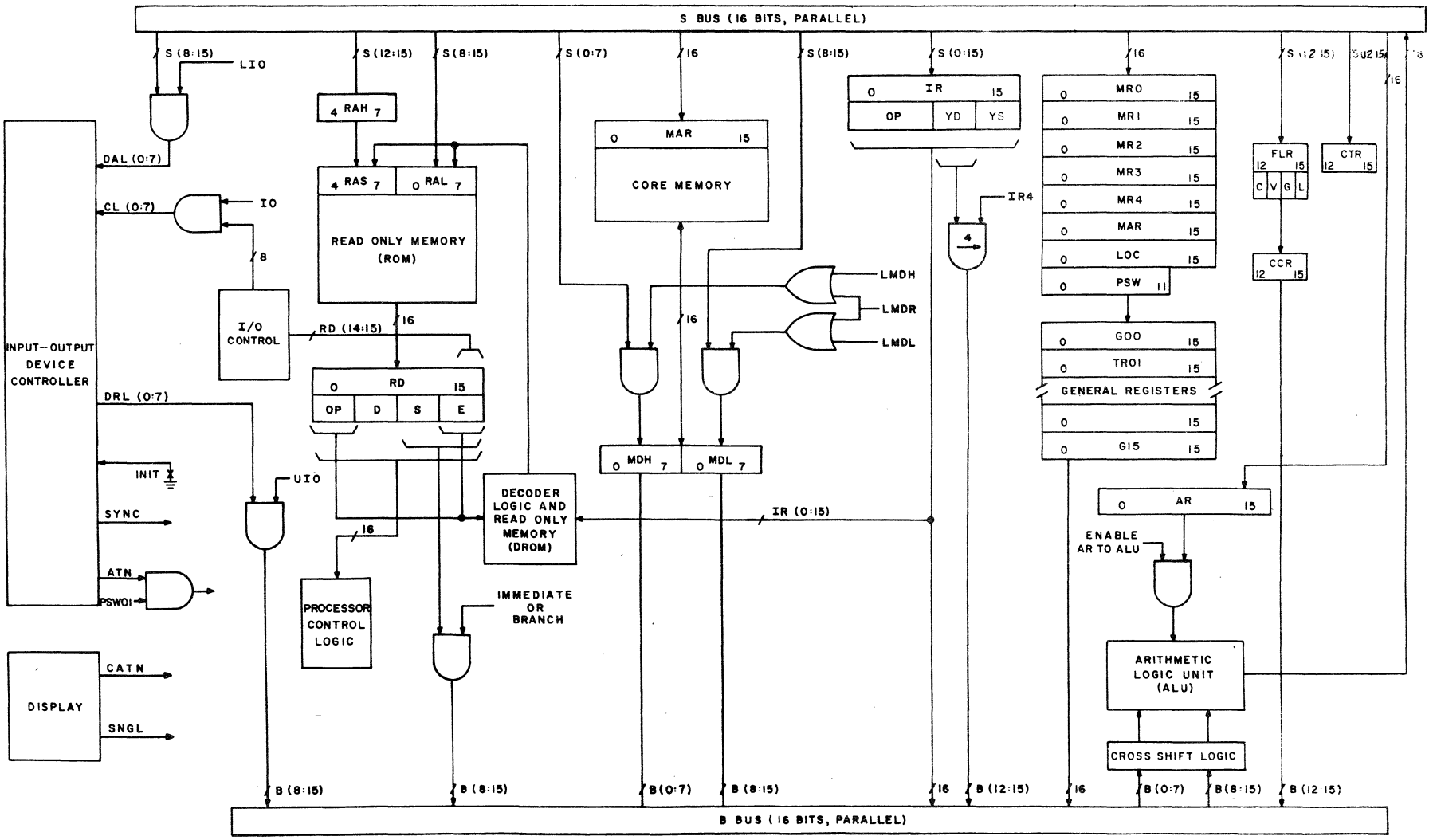


Figure 1. GE-PAC 30-2 Digital System, Block Diagram

The micro-register stack consists of five general purpose registers (MR0 through MR4) and three special purpose registers (MAR, LOC, and PSW). These registers are each sixteen bits long and may be considered as a high speed working memory for the micro-computer. Designating one of these registers in the Destination field (D) of RD causes it to be loaded with the data present on the S Bus. Designating one of these registers in the Source field (S) causes its contents to be placed on the B Bus.

The user's register stack consists of sixteen general purpose registers (G00 through G15). These registers are each sixteen bits long. As mentioned before, the General Registers are addressed by the R1 and R2 fields of the Instruction Register. The micro-program has access to the General Registers only by addressing the appropriate IR field. Thus, the General Registers are indirectly addressed.

When a General Register is specified as the Source for a micro-instruction, that particular register is selected by IR and its contents are placed on the B Bus. When a General Register is specified as the Destination, the particular register is selected by IR and loaded from the S Bus.

The Arithmetic Logic Unit (ALU) performs the micro-instruction specified by RD. The operation executed is determined by the control lines activated when the Processor Control Logic decodes the micro-instruction.

The B Bus always provides one operand to the ALU. The other operand is taken from the A Register (AR). The AR is a sixteen bit register that can be loaded from the S Bus and input to the ALU as an operand if needed by the instruction being executed. The output from the ALU is the S Bus.

The Flag Register (FLR) is a four bit register containing the following flags: Carry (C), Overflow (V), Greater than zero (G), and Less than zero (L). These flags are modified at the conclusion of arithmetic and logical operations to reflect the result of the operation. The FLR is tested by the Branch on Condition micro-instruction. The FLR can be loaded from bits 12 through 15 of the S Bus and unloaded to the Condition Code Register.

The CCR is a four bit flag register for the emulated computer, reflecting the results of user's instructions. It may be loaded only from FLR. The Condition Code Register supplies bits 12 through 15 of the B Bus when PSW is specified in the Source field of RD.

The Counter (CNTR) is a four bit decrementing register used on commands - Repeat, Multiply, and Divide - and the Branch on Counter instruction. The Counter may be preloaded from S Bus bits 12:15.

The Core Memory is the source of User instructions and data constants. It consists of ferro-magnetic cores and contains from 2K to 32K sixteen bit locations. (See the Core Memory Section of this manual.) Memory operations are initiated by Processor Control during a Command or a Decode micro-instruction. The location addressed in core memory is designated by the sixteen bit Memory Address Register (MAR). MAR is loaded from the S Bus prior to the memory operation. The two Memory Data Registers (MDH and MDL) hold the sixteen bits to be read from, or written into, memory. MDH and MDL may be loaded simultaneously or individually from the S Bus in preparation for a memory write operation. MDH and MDL may be unloaded simultaneously to the B Bus following a memory read. Memory timing considerations are resolved by the Processor. If the micro-program tries to unload the Memory Data Register immediately following a memory read, the Processor simply waits until the data is available.

Non-memory directed operations may be interleaved with the memory operations for maximum efficiency.

The I/O control directs the Device Controllers to load data into the B Bus or accept data from the S Bus. The type of data to be loaded or unloaded is determined by RD bits 14 and 15 and the direction of transfer desired (Input or Output). When the device has performed the function indicated by the active Control Line, it responds by raising the Sync Line, terminating the transfer.

3. FUNCTIONAL DESCRIPTIONS

This Section provides descriptions of the major Processor functional areas. The descriptions reference both simplified drawings provided in this Section, and Functional Schematics provided in Volume 2 of this manual.

3.1 Clock Control

The clock generator is shown on Sheet 23 of FS45, and is located on the C01 motherboard.

The clock system for the 30-2 employs a crystal-controlled 2.5 MHz oscillator. The crystal itself is mounted with a voltage divider on the daughter-board in location 26. The output from the crystal circuit feeds the oscillator in location 36, which amplifies the crystal output. The amplified sinusoidal output, OSC1 on pin 10 of the oscillator, varies between ground and +5V.

OSC,1 is connected to the T input of a JK flip-flop. When the flip-flop is reset, the high output CLKOFF,0 is ANDed with OSC,1 to develop the RAWCLK input to the pulse shapers. When the flip-flop is set, CLKOFF,0 goes low, preventing OSC,1 from developing RAWCLK, and inhibiting the clock outputs.

The J input of the CLKOFF flip-flop is tied to ground on the back panel. The K input is designated VP,0. When the system is initialized, STPSYS,1 goes high and VP,0 goes low. STPSYS,1 sets the CLKOFF flip-flop. As long as VP,0 is low, the negative transition of the OSC,1 cannot reset CLKOFF, and clock pulses are inhibited until the initialize sequence is terminated. At this time, STPSYS,1 goes low and VP,0 goes high. CLKOFF resets on the next negative going transition of the oscillator output, enabling OSC,1 to develop the RAWCLK input to the pulse shapers.

The low going result of ANDing OSC,1 and CLKOFF,0 is ORed with the normally high EXTCLK,0 lead. (This lead is provided so that an external clock may be used to run the system during trouble-shooting or marginal testing. This is accomplished by grounding the CLKOFF,0 lead at pin 110-1 on the back panel to disable the internal oscillator, and then pulsing EXTCLK,0 with a low going output from an external oscillator.) The external oscillator used should meet the following specifications:

1. square wave oscillator
2. logic levels - 0 to + 0.45 vdc for logic ZERO, 2.6 to 5 vdc for logic ONE
3. frequency - up to 2.7 MHz
4. output drive - 6 milliamps

The output from the OR gate, RAWCLK1, is available at Test Point L06 on C01. RAWCLK is next applied to 5 pulse shapers. The pulse shapers each contain two potentiometers. One potentiometer varies the pulse width, and the other varies the delay time. The proper method for adjusting these potentiometers is explained later in this Section. The outputs from the pulse shapers occur at the clock frequency, and are inhibited only when the system is initialized.

The output from pin 31 of the pulse shaper in position 37 is inverted and forms the current drive pulse (CD$\not$0) for the ROM. CD$\not$0 is not variable; all other clocks are adjusted relative to CD$\not$0. CD$\not$0 is available at Test Point L08 on C01. Output pin 10 of the pulse shaper in position 37 is ANDed with SKPSTB$\not$0 and SKDMST$\not$0 to develop ROMSTB$\not$0, the strobe pulse for the ROM. ROMSTB$\not$0 is available at Test Point L10 on C01. The output from pin 10 of the pulse shaper in position 35 (SYSCL1) is ANDed with SKDMST$\not$0 to develop the system clocks, CL0D and CL0E. CL0D is available at Test Point L12 on C01. CL0D is used primarily to gate data into the Processor registers. CL0E is used to gate information in and out of the memory registers. CL0E is available at Test Point L13 on C01.

SYSCL$\not$1 is ANDed with DMST$\not$0 and NORCL$\not$0 to produce RCL0, and with MST$\not$0 to produce SCL$\not$0. RCL$\not$0 is available on Test Point L15 on C01. RCL$\not$0 is used to gate data into the ROM Address Register and to increment the ROM Address Lower Register, RAL. SCL$\not$0 is a system clock which is inhibited only when memory speed conflicts with the Processor's demands.

The output from pin 31 of the pulse shaper in position 35 is ANDed with SKDMST0 to produce CCL$\not$0. CCL$\not$0 is available on Test Point L11 on C01. CCL$\not$0 is the gating clock for the Command micro-instruction.

The output from pin 31 of the pulse shaper in position 35 is ANDed with SKDMST$\not$0 to produce STCLK$\not$0. STCLK$\not$0 is used to strobe data into the Processor register stacks. STCLK$\not$0 is available on Test Point L14 on C01.

SYSCL$\not$1 feeds the lower pulse shaper in position 41. Output pin 31 of this pluse shaper is named DCL$\not$1. DCL$\not$1 is available on Test Point L07 on C01. DCL$\not$1 is positioned to appear between system clocks, and is not variable. It is used when multiple operations have to be completed within one clock period.

It is necessary to prevent generation of certain timing functions at specific times. To achieve this, a hierarchy of clock inhibits is implemented. This network of clock stops can be divided into three categories:

1. Initialize. All clocks are stopped.

2. Asynchronous to synchronous timing conflicts. This category involves memory directed operations and Input/Output operations.

3. Synchronous extended cycle operations.

The Add and Subtract micro-instructions require an extra system cycle to allow the "worst case" carries to propagate through the arithmetic system. Branches and Loads to RAL require an extra system cycle to allow sufficient time for the new address to be decoded in the ROM decoding matrix. RCL$\not$0 and ROMSTB$\not$0 may be individually stopped to allow repitition of a single micro-instruction.

The clock inhibit circuits are shown on Sheet 22 of FS45. When a memory directed operation is attempted and memory is not ready, the normally high MST$\not$0 lead becomes active (low). MST$\not$0 is available on Test Point L03 on C01. MST$\not$0 inhibits all clocks except SYSCL$\not$1 and DCL$\not$0, forcing the Processor to wait for the memory operation to be completed. Note that SCL$\not$0 is not inhibited if the micro-operation is Decode. The MST$\not$0 signal is controlled by two JK flip-flops, shown at the bottom of Sheet 22.

The lower flip-flop inhibits clocks if the micro-code program directs the Processor to unload the Memory Data Register (UMDR$\not$0) after a memory cycle has been initiated and the data is not yet available. The low active Unload Memory Data Register (UMDR$\not$0) signal is inverted and ANDed with the set output from the lower flip-flop. The state of this flip-flop is controlled by the Memory Data Available signal (MDAV$\not$0A) coming from the core memory. MDAV$\not$0A is low if there is

no memory operation in progress, or if a memory read is being executed and the readout is available in the Memory Data Register (MDR). MDAV,0A resets the flip-flop, preventing it from inhibiting the clock outputs. If memory is in the first half cycle of a memory operation (read), MDAV,0A is high causing the flip-flop to set at the trailing edge of the next system clock. If the micro-code program attempts to unload the Memory Data Register while this flip-flop set, MST,0 goes low stopping the clock. When memory data becomes available, MDAV,0 goes low, the JK flip-flop resets, and MST,0 goes high, which allows clock pulses to be generated to unload the Memory Data Register.

Notice that the clock which triggers the flip-flop (SYSCL,1) is not inhibited by MST,0. As soon as memory data becomes available, the flip-flop is reset on the trailing edge of the next SYSCL,1 pulse. Since ROMSTB,0 was inhibited by the MST,0 signal, the next ROM instruction word was not read out and UMDR,0 remained low while the clock was stopped.

Since memory timing is a synchronous with the system clock, the flip-flop guarantees that a clock pulse will not be shortened when data becomes available.

The other JK flip-flop capable of developing MST,0 also has an input from memory. This lead, Memory Busy (MBY,1A), is normally low, holding the flip-flop reset. MBY,1A goes high and sets the JK flip-flop when any memory operation is initiated, and remains high for the full memory cycle. If the micro-code program attempts to load any memory register or initiate another read or write cycle, the output from the OR gate goes high. This is ANDed with the set output from the JK flip-flop to develop the MST,0 signal which prevents generation of all System clocks. At the conclusion of the initial memory operation, the MBY,1A lead goes low, allowing the next SYSCL,1 pulse to reset the flip-flop.

With the flip-flop reset, MST0 goes high enabling all System Clocks. The pending memory operation is then executed.

When an I/O Load micro-instruction is performed, the DST,0 signal becomes active (low). DST,0 is ORed with MST,0 to produce DMST,0. DMST,0 inhibits RCL,0. DST,0 and MST,0 are ORed with LRAL,0 to produce SKDMST,0. SKDMST,0 inhibits CL0D, CL0E, STCLK,0, and CCL,0.

At least 2 clocks are skipped (inhibited) on I/O Loads. This allows sufficient time for the I/O control and data lines to settle and an I/O Sync (SYN) to be returned. When a Sync returns from the device, DST,0 goes high, releasing the clocks. If the device controller is slow returning sync, more than 2 clock periods will be skipped.

The ENSKP flip-flop shown on FS45-22G3 provides the capability of skipping one clock period while performing Add, Subtract, Branch, Decode, or Load RAL micro-ops. The K input to the ENSKP flip-flop is normally low. It becomes high if any one of five micro-instructions mentioned above is executed. The K input is fed by two NAND gates whose outputs are tied together. During a Branch instruction, BRA,0 is low, forcing both NAND gate outputs high. During Add or Subtract, L,0 is low, forcing both outputs high. During a Load RAL instruction, UO1,0 is low, causing one NAND gate output to be high. BANK,1 is low forcing the other NAND gate output high. During a Decode instruction, D0 is low forcing both outputs high. The high present on the K input of ENSKP is ANDed with the set output from the ENSKP flip-flop which is high until the trailing edge of the next clock pulse. The output from this gate, LRAL,0, goes low and inhibits CL0E, CL0D, STCLK,0, CCL,0, and ROMSTB,0. With a high on the K input of ENSKP and the J input tied to +5 volts, the ENSKP flip-flop complements on the next clock pulse. With ENSKP reset, the set output goes low, forcing LRAL,0 high and

once again CL0E, CL0D, STCLK$\not$0, CCL$\not$0, and ROMSTB$\not$0 are enabled. Note, however, that one clock period has been skipped. LRAL$\not$0 and its complement LRAL1 are also wired to the RMI mother-board to inhibit the incrementing of RAL and enable the gates from the S Bus to RAL.

During a Branch or a Load RAL micro-op, the information is gated into RAL from the S Bus with the RAL clock (RCL$\not$0). During Add, Subtract, or Decode, RCL$\not$0 is also skipped. This is accomplished by ORing L$\not$0 and D$\not$0 at Gate 51-15.

If either L$\not$0 or D$\not$0 is low, this gate produces a high output. This high is ANDed with the set output of the ENSKP flip-flop to produce a low into the gate that generates NORCL$\not$0, which inhibits RCL$\not$0. Notice that another input to the gate which generates RCL$\not$0 is DMST$\not$0. This input is used to inhibit RCL$\not$0 from being generated to increment the address register during a memory or device clock stop sequence. During Branch or Load RAL, L$\not$0 and D$\not$0 are both high inputs to Gate 51-15, forcing the output low. This low produces a high output from Gate 21-15, NORCL$\not$0, which allows RCL$\not$0 to be generated. Special conditions arise during multiply, divide, or repeat which cause NORCL$\not$0 to become active. NORCL$\not$0 inhibits only RCL$\not$0. Special conditions in the Decode micro-instruction cause ROM Strobe (ROMSTB$\not$0) to be skipped. This is done by making the lead SKPSTB$\not$0 active (low). SKPSTB$\not$0 also generates NORCL$\not$0, thus inhibiting RCL$\not$0 and ROMSTB$\not$0. SKPSTB$\not$0 disables LRAL$\not$0 at Gate 20-16. When SKPSTB$\not$0 is active, only RCL$\not$0 and ROMSTB$\not$0 are skipped. Table 2 provides a summary of clock stop conditions.

TABLE 2. GE-PAC 30-2 CLOCK STOP SUMMARY

	SCLR$\not$0	MST$\not$0	DST$\not$0	D$\not$0* L$\not$0	LRAL$\not$0	D$\not$0* SKPSTB$\not$0	NORCL$\not$0
RAWCLK$\not$1	0	1	1	1	1	1	1
SYSCL$\not$1	0	1	1	1	1	1	1
CD$\not$0	0	1	1	1	1	1	1
DCL$\not$0	0	1	1	1	1	1	1
SCL$\not$0	0	0	1	1	1	1	1
RCL$\not$0	0	0	0	0	1	0	0
CL0D	0	0	0	0	0	1	1
CL0E	0	0	0	0	0	1	1
CCL$\not$0	0	0	0	0	0	1	1
STCLK$\not$0	0	0	0	0	0	1	1
ROMSTB$\not$0	0	0	0	0	0	0	1

1 = Active
0 = Inactive

* D$\not$0 acts like L$\not$0 unless Phase Zero RR. Then SKPSTB$\not$0 overrides.

3.2 Initialize Control

The Initialize Control logic is shown on Sheet 21 of FS45. This logic provides an orderly system shutdown when the system is initialized. When an initialize sequence occurs, the following actions take place:

1. Stop Clock
2. Reset RAL
3. Reset RAH
4. Reset FLR
5. Reset Bank flip-flop
6. Reset Utility flip-flop
7. Reset RD
8. Reset RUN flip-flop
9. Reset PTYS flip-flop
10. Force Phase 3
11. Set PSW01 flip-flop
12. Reset FNY0 and FNY1 flip-flops
13. Reset NORCL flip-flop
14. Reset FRCGP1 flip-flop
15. Set Wait Alarm
16. Provide reset signals for Memory and I/O

The system is initialized as a result of one of the following conditions:

1. Command Power Down or Power Up.
2. AC input (with optional Power Fail Detector), P15, N15, or P5 falls below minimum operating level.

The master reset signal, SCLR$\not$0, is active when the initialize relay is de-energized. This happens when the POWDN$\not$0 signal generated at G6 on FS45-21 is active. During normal operating conditions, POWDN$\not$0 is high, permitting the Darlington circuit, controlling the Initialize Relay, to conduct. As long as the Darlington circuit conducts, the relay remains energized. POWDN$\not$0 goes low to start an initialize sequence when one of the two conditions mentioned occurs.

The STPSYS$\not$1 flip-flop (J3) is set if a Command Power Down (POW) is executed. The Processor tests for Primary Power Fail (PPF$\not$0) between execution of each user instruction. PPF$\not$0 is true if the latching POWER Switch is released; if the optional Primary Power Fail Detector on Sheet 21 determines a voltage is lost, or if the Console INITIALIZE Switch is depressed. When Primary Power Fail is detected, the microcode program transfers to a Command Power Down. The Command instruction generates the low signal, POW$\not$0, at R8 on Sheet 13. POW$\not$0 sets the STPSYS flip-flop.

When STPSYS$\not$1 goes high, a low is placed on the DC set input to the CLKOFF flip-flop (FS45-23C4). STPSYS$\not$1 is high causing POWDN0 to go low. When POWDN$\not$0 goes low, it removes the positive potential on the input to the Darlington circuit, turning the circuit off and de-energizing the relay. The reset output of the STPSYS flip-flop generates the signal VP$\not$0 (FS45-21R6). VP$\not$0, low, holds the K input of the CLKOFF flip-flop low, preventing it from setting until the initialize sequence is terminated.

The second instance when POWDN$\not$0 goes low to drop out the Initialize Relay, occurs when one of the voltages - +5v, +15v, -15v, or the AC input - is lost. If the N15 supply voltage is lost, the output of the NAND gate on Sheet 21-G3 goes low. If the +5v is lost, the collector supply for the Darlington circuit is removed, causing the Initialize Relay to drop out. If the AC input is lost, the POWDN$\not$0 lead is also forced low. The AC input is sampled from the secondary of a 12 VAC transformer and rectified. The pulsating DC voltage is applied to potentiometer P451 (FS45-21E3). The potential selected by the potentiometer is applied to the optional Power Fail Detector.

NOTE

The following description of the Primary Power Fail Detector assumes that the two transistors in the 35-063 are Q1 and Q2, left to right, and that the two transistors in the 35-082 are Q3 and Q4, left to right.

Initially, transistors Q1, Q3 and Q4 are off and transistor Q2 is conducting. With transistor Q2 conducting, a high enough positive potential on the base of Q4 keeps Q4 cut off, and applies about +6v to the emitter of Q1. The potential from potentiometer P451 (approximately 8v) is felt on the base of Q1. If the AC input is lost, or fluctuates enough, the potential on the base of Q1 becomes more negative and Q1 conducts.

When Q1 conducts, it places a positive potential on the base of Q3, turning Q3 on. This causes the potential on the emitter of transistor Q2 to become less positive, turning Q2 off. When Q3 conducts, its collector goes to ground, generating the low signal PPF$_L$0. PPF$_L$0 is tested between execution of User instructions to determine if a Power Fail is in progress.

When transistor Q2 turns off, the positive potential is removed from the base of Q4. Transistor Q4 then conducts, grounding the POWDN$_L$0 lead, turning the Darlington circuit off, and dropping out the Initialize relay. The Initialize relay, which is a dry reed relay with SPDT contacts, is shown with the relay not energized.

When the relay releases, the SCLR$_L$1 contact opens, causing SCLR$_L$1 to go high. This removes the ground for the POWER lamp. The SCLR$_L$0 contact goes to ground. This metalic ground is distributed to memory and I/O for initialize control. SCLR$_L$0 holds VP$_L$0 at ground to inhibit the CLKOFF flip-flop resetting and holds the DC set input of CLKOFF at ground.

SCLR$_L$0 is inverted through Gate 21-24 (M6). The output of this gate rises to the +5 volts and turns on the four gates it feeds. The output SCLR$_L$0A, resets RAL, RAS, and RAH. SCLR$_L$0B clears RD. SCLR$_L$0C clears the FLR and resets the following flip-flops: BANK, UT, RUN, PTYS, FNY0, FNY1, NORCL, FRCTP1, and SOPC. SCLR$_L$0C sets the following flip-flops: WA0, PSWO1, DST, FA$_L$0, and FA$_L$1. When the remaining Gate, 20-00 (L3) goes low, its output resets the STPSYS flip-flop and holds VP$_L$0 at ground.

Resetting the STPSYS flip-flop removes the ground on the input to the Darlington circuit. This allows the 66uf capacitor to slowly charge. When the threshold of the input transistor is reached, the Darlington circuit turns on, and is energized.

When the SCLR$_L$1 contact closes, it provides the ground return for the POWER lamp. The SCLR$_L$0 contact opens and removes the reset functions from memory. When this occurs, the output from the NAND gate at M6 goes to ground immediately. This removes the reset signals SCLR$_L$0A, SCLR$_L$0B, and SCLR$_L$0C. The output from the NAND gate at L3 clears the STPSYS flip-flop and causes the line VP$_L$0 to begin to rise. VP$_L$0 is delayed by the 33 uf capacitor to insure that the clock will not start until all initialize activity is complete.

The previous description applies for a Command Power Off, since this is the only way to set the Initialize flip-flop.

For the case where Initialize is caused by a failure of P5, N15, P15, or Power is removed, the relay de-energizes and remains in that state until the fault is corrected.

When power is turned on, the system is initialized in an orderly fashion by the following mechanism. The SCLR$_L$0 contact of the relay stays at ground enabling the reset signals SCLR$_L$0A, SCLR$_L$0B, SCLR$_L$0C. This holds the clock off initially. As the P5 reaches normal operating levels, the Initialize flip-flop is reset after a delay. This insures that the PDOWN$_L$0 node is not forced to ground and permits the 66uf capacitor at the input to the Darlington to charge slowly. The relay then energizes the system resets.

3.3 Processor Registers

The Processor executes micro-instructions received from the ROM. This is accomplished, primarily, by moving data from one register to another and modifying it in the ALU. In order to manipulate data and execute the instructions received, the Processor uses a number of internal registers. Most of these are 16-bit registers. Some registers perform special functions, but most are general purpose. Data is transferred between registers and other system elements via the 16-bit B and S Buses in the Processor. Each register is described in the following paragraphs.

3.3.1 ROM Data Register. The ROM Data Register (RD) is a 16-bit register located on the ROM Interface board (RMI). The schematic drawings for this register are on FS44 - 1:3. The ROM Data Register can be thought of as the micro-instruction register. Each bit of RD has double rail inputs from the ROM. Each micro-instruction extracted from the Read-Only-Memory is placed into the RD register, where it remains while the instruction is executed. A thorough knowledge of the ten basic micro-instructions and their instruction word formats is necessary to fully understand the RD register. (See the Micro-Instruction Reference Manual, Publication Number 29-032 for a detailed explanation.)

Bits 0:3 of RD are decoded as the op-code of the micro-instruction being executed. RD bit 3 determines Immediate instructions. An Immediate instruction is one that has the specific data to be manipulated appended to the instruction word itself. During the performance of an Immediate instruction, bits 8:15 of RD are gated onto the B Bus bits 8:15 and treated as data, instead of being decoded as Source and Extended Fields, Test Code, or Command Code.

Bits 4:7 are normally decoded as the Destination Register address. This identifies the register designated to contain the result upon completion of the micro-instruction being performed. When a Branch instruction is executed, bits 4:7 define the condition that the Processor will branch on. If the Processor decodes a Test or Command instruction, bits 4:7 are part of the Command or Test Code.

Bits 8:11 are decoded as the Source register address for micro-instructions in the Register to Register format. The Source register contains one of the operands to be used. If a Branch instruction is decoded, bits 8:11 are part of the Branch address. If the Command or Test micro-op is decoded, bits 8:11 are decoded as part of the Command Code. If an Immediate instruction is decoded, bits 8:11 are part of the data.

Bits 12:15 normally make up the Extended field for Register to Register instructions and are decoded to specify options for specific instructions. The Extended field designates the function of Carry, whether to shift or cross shift the operand manipulated, whether to set flags, whether to enable the A Register input to the ALU, etc. During other instructions, bits 12:15 are part of the Command Code, or data.

Other than the double-rail inputs from the ROM, the only other input to the RD register is SCLR0B. SCLR0B goes low and clears the entire RD register when the system is initialized, or when a non-existent page of ROM is addressed (FS45-21P3). All of the outputs from the bits in the RD register appear directly in the Control Logic so that the various op-codes and options can be decoded and the proper gating leads to the ALU can be activated. The set outputs of RD bits 8:15 feed individual power gates that generate the false B Bus outputs (low if a Logic ONE) if RD 03 is set, indicating an Immediate instruction.

3.3.2 ROM Address Lower Register. The ROM Address Lower Register (RAL) is an eight bit register that contains the least significant portion of the address of the location to be extracted from the Read-Only-Memory. The schematics for RAL are located on Sheets 4 and 5 of FS44.

During performance of the major portion of the micro-program, micro-instructions are selected from successive locations in the Read-Only-Memory. The RAL register is wired as a standard up-counter to facilitate the selection of successive locations.

The RAL is incremented to read successive ROM locations except during a Branch micro-instruction when the specified condition is true, or if RAL is designated as a Destination register. In either of these cases, the RAL increment signal is disabled and the new address is loaded into RAL from the S Bus.

The RAL register is cleared by the SCLR/0A signal when the system is initialized, or conditionally on a Decode. The RCL/0 signal is used to increment the RAL register or to gate the information from the S Bus into RAL. The RCL/0 signal is inverted and applied to the T input of each flip-flop in RAL as RCL/1. If a Branch or a Load RAL instruction is performed, LRAL/0 is low and LRAL/1 is high. LRAL/1 is NANDed, bit for bit, with the S Bus to enable the J and K inputs to each of the flip-flops in the RAL register. If any inputs from the S Bus are high, the K input is forced low and the J input is forced high on the corresponding bit in the RAL register, causing it to set. If an S Bus input is low, the K input is forced high and the J input is forced low, causing the RAL flip-flop to reset.

When the Decode micro-instruction generates a new ROM address, the ROM address registers are first cleared, then the new address is loaded. The Decode instruction generates SCLR/0A on the Delayed Clock (DCL1). SCLR/0A clears RAL, RAH, and RAS. Then on the next system clock, the appropriate bits of RAL and RAS are set by placing a low on the DC Set inputs (SRAxxx/0). Note that the RAH is always left in the cleared state.

When the Processor is not executing a Branch or a Load RAL, LRAL/0 is high and LRAL/1 is low. With LRAL/1 low, the inputs from the S Bus are disabled and the RAL register is incremented by one on each negative transition of RCL/1. The RAL is incremented by complementing each bit of the register if all previous least significant bits are set. RAL07 complements every time. RAL06 complements if RAL07 is set. RAL05 complements if RAL07 and RAL06 are both set and so on. This is accomplished by NANDing the set output of each flip-flop with all previous bits to enable the J and K inputs for the next significant bit. If all previous bits are set, the output of the particular 19-003 is low. The signal is inverted, placing a high on the J and K inputs, allowing that bit to complement. If any of the previous bits are not set, the output from the 19-003 is high, forcing the J and K inputs low and preventing that flip-flop from changing state.

3.3.3 ROM Address Higher Register. The ROM Address Higher Register (RAH) is shown on Sheet 6 of FS44. RAH is a four bit extension of RAL which increases the ROM addressing capability to 4096 locations. This register is independent of the RAL register from both the programming and the hardware viewpoint. RAL is normally incremented to fetch micro-instructions from successive ROM locations. When RAL is equal to its highest address (hexadecimal FF), the next increment pulse causes it to recycle to an address of all zeros, but no carry is propagated into the least significant bit of RAH. For this reason RAH must be loaded independently each time a Branch or Load RAL selects a different page.

The RAH register is composed of RS flip-flops with single-to-double rail converters on the inputs. This register has inputs only from the S Bus. A new address is loaded into RAH at clock time from the S Bus when RAH is specified as the Destination register of a Load instruction. To address the RAH register, the BANK flip-flop must be reset. Signal LRA/1 gates the contents of the S Bus into RAH. LRA/1 is developed by the three inverters whose outputs are AND tied in the upper left hand corner of Sheet 6. The inputs to the three inverters are RCL/0, BANK/1 and LRAH/0. BANK/1 is low when the Destination address decoder decodes the address X'0'. When the three inputs are low, LRA/1 goes high, gating the S Bus into RAH. The only other input to RAH is the SCLR/0A lead. SCLR/0A is normally high and becomes low to clear the RAH register when a Power Down Command micro-instruction is executed, the power is turned on, or conditionally on a Decode micro-instruction.

3.3.4 ROM Address Slave Register (RAS). The ROM Address Slave Register is a four bit register shown on Sheet 6 of FS44. RAS is slaved to the four bit RAH register. The single rail buffered outputs of RAS are decoded as the page of the ROM address to be selected.

The contents of RAH are loaded into RAS every time RAL is loaded from the S Bus. This allows the correct page address to be loaded into the RAH register prior to a Branch or Load RAL micro-instruction. When RAL is loaded with a new address as a result of executing a Branch or Load RAL, the correct page is gated into RAS from RAH. In this way all twelve bits of the new ROM address arrive at the ROM address decoder simultaneously (allowing slight variations in the circuit delays).

LRAS/1 gates RAH into RAS. LRAS/1 is developed by the two inverters collector AND tied on the left hand side of Sheet 6. The two inputs are RCL0 and LARL/0. RCL/0 varies with system clock. LRAL/0 is the signal used to gate the S Bus into RAL. LRAL/0 is low during a Branch or a Load RAL micro-instruction. When LRAL/0 is low, LRAS/1 varies with RCL/0 and gates RAH into RAS on the negative transition.

The normally high SCLR/0A input to RAS goes low conditionally on a Decode or when the system is initialized. SCLR/0A clears all of the ROM address registers. When the Decode micro-instruction generates a new ROM address, SCLR/0A is generated on DCL/0. The appropriate SRAn/0 leads go low on the next system clock to set the desired address.

3.3.5 Flag Register (FLR). The Flag Register (FLR) is a four bit register located on the C02 mother-board. The schematic drawings are on Sheets 4 and 5 of FS45.

The FLR is cleared by the SCLR/0C signal. SCLR/0C goes low when the system is initialized. SCLR/0C is connected to the DC Clear inputs of the FLR flip-flops.

When the hex address 'B' is decoded in the Destination address decoder, LDF/0 becomes low and LDF/1 becomes high. LDF/1 is NANDed with each bit of the S Bus. If any bit of the S Bus is high, the K input on the corresponding bit in the FLR goes low and the J input high, allowing the flip-flop to set on the next negative going clock transition. If the S Bus input is low, the K input goes high, and the J input goes low, allowing the flip-flop to reset.

Since the FLR is used to hold the flags indicating the result of micro-instructions, there are instances when individual bits will be set or reset independently. To facilitate this independence, different gating pulses are developed for individual bits.

When the FLR is loaded from the S Bus, LDF,0 is used to generate all of the gating pulses. The gating pulse for bits 14 and 15 of the FLR is developed by two inverters collector AND tied on the upper right hand side of Sheet 5. The input to one is the system clock CL0D and one input to the other is the low signal LDF,0, enabling the necessary negative transition to the T input on FLR bits 14 and 15 as the system clock changes from 0 to +5 volts.

The gating pulse for bit 13 is developed similarly. The same is true for bit 12.

The FLR register is unloaded to the Condition Code Register only, as an option on the Decode micro-instruction.

In addition to being loaded from the S Bus, FLR bit 12 is used as the Carry Store flip-flop (C). FLR12 is set or reset to reflect whether or not a Carry or Borrow resulted upon conclusion of an Add or Subtract micro-instruction, or to reflect the state of the bit shifted out following a Load micro-instruction designating the shift left or a shift right option. The gates which control setting or resetting FLR12 are shown on Sheet 4 of FS45.

FLR12 is set when the signal SFL12,0 is low. SFL12,0 is generated by three NAND gates collector OR tied. The inputs to the first NAND gate are SHR,1 and B15,1. SHR,1 is developed on the CO0 mother-board (Sheet 14 of FS45). SHR1 will be high if the Processor decodes a Load micro-instruction and ROM Data Register bit 13 is set, indicating the Shift Right option. B15,1 will be high if the least significant bit of the data present on the B Bus is a Logic ONE.

The second NAND gate capable of driving SFL12,0 low has the inputs SHL,1, and B00,1. SHL,1 is high if the Processor is performing a Load micro-instruction whose E field designates the Shift Left option. B00,1 is high if the most significant bit of the information carried on the B Bus is a Logic ONE. The third instance when SFL120 is low occurs when SVAC,1 and CSV,1 are both high. SVAC,1 is high during an Add or Subtract, if the E field designates the carry out of the ALU option. CSV,1 is high if a final carry or a borrow resulted during Add or Subtract.

SFL12,0 is applied to the K input of FLR bit 12, and inverted as an input to the J input. The gating pulse is developed by SVAC,0 and CL0D during Add or Subtract operations, and by SVSC,0 and CL0D during Load instructions with shift options.

FLR bit 13 serves as the Overflow flag (V). The gating which controls setting or resetting FLR13 is shown on Sheet 4 of FS45. FLR13 is set when SFL13,0 is low. SFL13,0 is generated by two NAND gates collector OR tied. One of the inputs to both NAND gates is STF,1. STF,1 provides a high input to both NAND gates capable of generating SFL13,0. STF,0 is generated on Sheet 14 of FS45. STF,0 is low when the Processor is performing any instruction that is not a Test, Load, or Immediate instruction, and bit 13 of the E field of the instruction word is set specifying the Set Flags option. STF,0 is inverted and NANDed with L1 to generate the low signal STFL,0 during Add and Subtract.

The other two inputs needed to generate SFL13,0 are C00,1 and CSV,0 for one gate and C00,0 and CSV,1 for the second gate. C00 is the carry into the most significant bit of the adder and CSV is the carry out of the most significant bit of the adder. Whenever these carries are different upon conclusion of an Add or Subtract (C00,1)(CSV,0) or (C00,0)(CSV,1), an overflow condition exists and SFL13,0 goes low. SFL13,0 is applied to the K input of FLR13 and inverted as an input to J. The gating signal for FLR13 is generated by CL0D and STFL,0.

FLR13 can also be set by the SV,0 signal on the S input. SV,0 goes low when the False Sync flip-flop is set (reference Sheet 25 of FS45). The False Sync flip-flop is set if a power fail condition is detected or if the Processor does not receive a return sync from a device controller within 35-50 microseconds from the time the I/O control line is activated to initiate an I/O operation.

FLR14 and FLR15 serve as the Greater than and Less than flags respectively. FLR14 is set by the SFL14,0 signal if the data manipulated on the S Bus is greater than zero or the result of a Test micro-instruction was true. FLR15 is set by SFL15,0 if the data manipulated on the S Bus is less than zero or if the result of a Test micro-instruction proved false.

SFL14 and SFL14 are generated at two different sources. The logic to set or reset FLR14 and FLR15 as a result of examining the data on the S Bus is shown on Sheet 4 of FS45. The inputs necessary to develop SFL14 and SFL15 are identical with the exception of S Bus bit zero. SFL14,0 goes low allowing FLR14 to set if S Bus bit zero is reset (S00,0) indicating a quantity greater than zero. SFL15,0 goes low allowing FLR15 to set if S Bus bit zero is set (S00,1) indicating a quantity less than zero. The other input required to develop SFL14,0 or SFL15,0 is the complement of STF,0. Using the complement of STF,0 was discussed in conjunction with setting FLR13 (the V flag) and is high for SFL14,0 and SFL15,0 on the same conditions. GF145,1 is only low if the S Bus contains all zero data and the G (FLR14) and L (FLR15) flags are reset. There are AND ties on the lead which may also keep it low, but these will be discussed later. The S Bus equal to zero indication comes into the 35-022 in position 60-04 as 4 separate signals. Each signal comes from a different Bit Slice board. Each signal will be high if the four S Bus bits on that Bit Slice are zero. S003,1 comes from Bit Slice 0 and is high if S Bus 0:3 is zero. S047,1 comes from Bit Slice 1 and is high if S Bus 4:7 is zero. S811,1 comes from Bit Slice 2 and is high if S Bus 8:11 is zero. S125,1 comes from Bit Slice 3 and is high if S Bus 12:15 is zero. GF1415 is high to allow FLR12 and FLR13 to be triggered. GF145,1 will be high if any bit on the S Bus is set or if FLR14 or FLR15 is set. The G and L flags facilitate microcode handling of multiple precision operands. The reset outputs from FLR14 and FLR15 allow the Greater than flag to be set if the result of the least significant sixteen bits produces a zero result.

The GF145,1 lead is AND tied to the two NAND gates at the top of Sheet 5. The combined signal is then fed to the Trigger inputs of FLR14 and FLR15.

The logic to generate SFL14,0 and SFL15,0 as a result of a Test micro-instruction is shown on Sheet 4 of FS45. The Test instruction examines specified conditions in the Processor. The testable functions are I/O Interrupt, Auto Restart, Console Interrupt, Console Single Instruction, the Utility flip-flop, the Memory Parity Fail flip-flop, Fast I/O Interrupt, and Primary Power Fail. The function to be tested is specified by the Test Code, bits 4:15 of the micro-instruction word. If the machine function specified by the Test Code is true, the Greater than flag (G) is set and the Less than flag (L) is reset. If the function specified by the Test Code is false, the Less than flag (L) is set and the Greater than flag (G) is reset.

On Sheet 12, each of the testable functions is NANDed with the ROM data register bit with which it corresponds. The NAND gates are collector OR tied in groups of four. If one of the tested functions is true, the output at the collector OR tie goes low, driving the output of the next NAND gate high. This high (TTEST,1) is inverted to produce TTEST,0. Both leads (TTEST,1 and TTEST,0) are shown on Sheet 4. TTEST,1 is NANDed with CTST,1 which will be high during a Test instruction, driving the output SFL14,0 low.

When SFL14,0 is low, SFL15,0 is high. With SFL14,0 low and SFL15,0 high, the G flag sets and the L flag resets, indicating that the test was true. If the function tested for is false, the outputs from all collector OR tied NAND gates are high causing the output TTEST,0 to go high. This high causes SFL14,0 to be high. When SFL14,0 is high, SFL15,0 is low. With SFL15,0 low and SFL14,0 high, the L flag sets and the G flag resets, indicating that the Test was false. The same gating pulse is used to set or reset FLR14 and FLR15. It is developed on Sheet 5 of FS45. During performance of a micro-instruction specifying set test flags, the STE,0 and CL0D signals are used to develop the gating pulse. During a Test instruction, CTST,0 and CL0D are used.

3.3.6 Register Stack. The register stack is shown on Sheet 3 of FS45. The register stack consists of sixteen bit registers that can be specified as Source (S) and Destination (D) registers for micro-instructions, during execution of the micro-code program.

These 24 General Purpose Registers are loaded with the contents of the S Bus if specified as the Destination register, or unloaded onto the B Bus if specified as the Source register during performance of micro-instructions. Bits 0:3 of each of the 24 registers are located on the Bit Slice 0 mother-board. Bits 4:7 of each register are located on Bit Slice 1. Bits 8:11 of each register are located on Bit Slice 2. Bits 12:15 of each register are located on Bit Slice 3.

Unlike all the previous registers of the JK or RS type, the register stack consists of forty-eight 19-009 memory cell integrated circuits, twelve on each Bit Slice. The 19-009 IC's are paired to provide four 4-bit words.

Referring to Sheet 9, if the Source Register Decoders select one of the 24 Registers, the appropriate U lead goes low. This DC input is inverted and applied to a non-inverting 35-041. The 35-041 has an emitter follower output that will be at about 2.1 volts for the selected output. All other 35-041 outputs will be at ground through the emitter resistor. The outputs from the 35-041 are input to the register stacks. See Sheets 2 and 3.

The register stack is level active, and does not require a gating pulse. As soon as one of the read leads Rnnn,1 goes high, the output from the selected register appears at the output pins on all Bit Slices. These outputs are about 3 volts for a logic ONE, and 0.6 volts for a logic ZERO. During execution of a micro-instruction that has not selected one of the registers, all of the outputs should be at 0.6 volts. The outputs from the 19-009's are inverted to develop the B Bus.

When one of the Destination Register decoders selects one of the 24 registers, (reference Sheet 11) the appropriate L lead Lnnn0 goes low. This DC input is inverted and input to a non-inverting 35-041 where it is ANDed with a differentiated clock pulse to develop write signals Wnnn,1. The inverted stack clock (STCLK,0) is input to the trailing edge differentiators shown on Sheet 11. This develops a 30 nanosecond positive going clock pulse as STCLK,0 charges from 0 to +5 volts, preventing the write signals from going high to gate the S Bus into the Destination register until the end of the instruction. This pulse is normally 2 volts and must be greater than 1.8 volts. The pulse width of 30 nanoseconds is fixed by the circuit constants and is not readily adjustable. The displacement, in time, of this pulse is a function of the STCLK,0 pulse width and position.

When this 30 nanosecond pulse is developed, the appropriate write line goes high. When one of the write leads, Wnnn$_{/}$1, goes high, the information on the S Bus is gated into the designated register. See Sheets 2 and 3. The S Bus is input to a voltage divider in the register stack which loads down the Bus. It is normal for the S Bus to be about 3.9 volts to represent a logic ONE.

The register stack is level active and does not require a gating pulse. The inverted L lead (Lnnn$_{/}$1) is ANDed with the differentiated clock pulse to prevent multilation of the data on the S Bus. Multilation could occur if the appropriate L lead was not ANDed with the differentiated clock and a micro-instruction designating the same register as both the Source and Destination was executed. The source register decoder would activate the correct unload lead and the contents of the Source register would be placed on the B Bus, modified in the ALU, placed on the S Bus, and gated back into the Destination register. Since the unload lead would still be high, the modified data would appear at the input to the ALU and be modified again before being placed into the Destination register.

3.3.7 Arithmetic Register (AR). The Arithmetic Register (AR) is shown on Sheet 1 of FS45. The Arithmetic Register is used to hold the first operand during execution of the Add, Subtract, OR, AND, and Exclusive OR micro-instruction. The AR consists of sixteen RST flip-flops with double-rail inputs from the S Bus. AR 0:3 are on BS0, AR 4:7 are on BS1, AR 8:11 are on BS2, and AR 12:15 are on BS3. The AR can be loaded with the contents of the S Bus. The contents of the AR can be gated out as an input to the ALU.

The A Register is loaded with the contents of the S Bus by gating signal LAR$_{/}$0, which is inverted and input to the RST flip-flop. LAR$_{/}$0 is generated on Sheet 10 of FS45. LAR$_{/}$0 is low when the Destination Address Decoder decodes the Arithmetic Register's address (hexadecimal 8). The inverted CL0D goes low providing the negative going transition to the T inputs of the AR to gate in the S Bus.

The set outputs from the Arithmetic Register are NANDed with UA$_{/}$1 to develop the GA leads (GA000:GA150) as inputs to the ALU. UA1 is generated on Sheet 8 of FS45 by two NAND gates whose outputs are collector AND tied. UA$_{/}$1 is high enabling inputs from the AR to the ALU when each of the two NAND gates has at least one low input. UA$_{/}$1 is low disabling inputs from the AR to the ALU if one of the two NAND gates has all high inputs. Both inputs to the NAND gate on the left are high if the Processor decodes a Decode, Branch, Command, or Load micro-instruction.

Both inputs to the right NAND gate are high if the Processor decodes a non-immediate micro-instruction and RD bit 12 is set specifying the "No AR to the ALU" option.

In summary, UA1 is high to gate the contents of the Arithmetic Register to the ALU during execution of all micro-instructions except non-immediate arithmetic or logical instructions that specify the "No AR to the ALU", or during Branches or Loads.

3.3.8 Instruction Register (IR). The Instruction Register is shown on Sheet 1 of FS45. The Instruction Register holds the user's instruction word during emulated execution. The IR consists of sixteen RST flip-flops with double-rail inputs from the S Bus. IR 0:3 are on BS0, IR 4:7 are on BS1, IR 8:11 are on BS2 and IR 12:15 are on BS3. The IR can be loaded from the S Bus. The contents of the IR can be unloaded to the B Bus.

The IR is loaded with the contents of the S Bus by gating signal LIR$_{/}$0, which is inverted and input to the gated RST flip-flops. LIR$_{/}$0 is generated on Sheet 10 of FS45. LIR$_{/}$0 is low when the Destination Address Decoder decodes the Instruction Register's address (hexadecimal 9). The inverted CL$_{/}$0D goes low providing the negative transition to the T inputs of IR to gate in the S Bus.

The set outputs of IR are constantly available in the hardware decoder section. The set outputs of IR are NANDed with the inverted UIR$_{/}$0 to develop the B Bus. UIR$_{/}$0 is generated on Sheet 9 of FS45. UIR$_{/}$0 is low when the Source decoders detect the Instruction Register's address (hexadecimal 9).

3.3.9 Condition Code Register (CCR). The Condition Code Register (CCR) is shown on Sheet 6 of FS45. It is used instead of the least significant four bits of the Program Status Word (PSW 12:15). Loading and unloading of PSW is disabled on Bit Slice 3 which contains the least significant four bits. See Sheet 3 of FS45. When PSW is loaded, S Bus bits 0:11 are loaded to PSW bits 0:11. The load PSW signal (LPSW$_{/}$0) produces LDF$_{/}$0 which will gate S Bus bits 12:15 into the Flag Register (FLR). See Sheet 10.

The CCR may only be loaded from the Flag Register. The Condition Code Register is loaded with the contents of the Flag Register by gating signal LCC$_{/}$0. LCC$_{/}$0 is inverted and NANDed with FLR 12:15 to set the RS flip-flops in CCR.

LCC$_{/}$0 is generated on Sheet 14 of FS45. LCC$_{/}$0 is low on the skipped clock of a Decode micro-instruction if the "Jam FLR to CCR' option (RD 13 = 1) is specified.

The set outputs from the Condition Code Register are NANDed with the complemented UPSW$_{/}$0 signal to form B Bus bits 12:15. UPSW$_{/}$0 is generated on Sheet 9 of FS45. UPSW$_{/}$0 is low when the Source Decoders detect the Program Status Word's address (hexadecimal 7).

3.3.10 Counter Register (CNTR). The Counter Register (CNTR) is shown on Sheet 7 of FS45. It is used to count the number of executions of a single micro-instruction when the Processor is in the Repeat mode. The Counter holds the Processor in the Multiply or Divide mode for sixteen passes. The Counter is also used as a testable item for Conditional Branches.

The four bit decrementing counter is loaded from the S Bus by the signal LCTR$_{/}$0. LCTR$_{/}$0 is generated on Sheet 10 of FS45. LCTR$_{/}$0 is low when the Destination Decoders detect the counter's address (hexadecimal C).

LCTR$_{/}$0 is one input to the OR gate at the bottom of Sheet 7. The other input is CLR$_{/}$0 which may be low during the first half of a Decode micro-instruction. LCTR$_{/}$0 or CLR$_{/}$0 low produces the high active signal RSCTR$_{/}$1 which is available at Test Point L06 on C02. RSCTR$_{/}$1 is NANDed with DCL$_{/}$1 and applied to the DC Clear inputs of the CNTR flip-flops.

LCTR$_{/}$0 is input to one of two inverters at the top of Sheet 7. The input to the other inverter is CL$_{/}$0D. The AND tied output is NANDed with bits 12:15 of the S Bus and input to the CNTR flip-flops on the DC set inputs. The Counter is cleared before the S Bus is gated in. The Counter is decremented by the DC signal DECTR$_{/}$0 which is available at Test Point L10 of C02. DECTR$_{/}$0 is generated on Sheet 20 of FS45. DECTR$_{/}$0 is low during the first instruction in the Multiply and Divide loops - on the Branch on Counter micro-instructions, or in Repeat Mode as long as the Counter is not all zeros.

DECTR$_{j}$0 is one input to two inverter tied AND gates on Sheet 7. These AND gate outputs are applied to the Trigger inputs of CTR12 and 13 and CTR14 and 15 respectively. The CNTR is decremented by complementing each bit of the register if all previous least significant bits are reset. CTR15 complements every time. CTR14 complements if CTR15 is reset. CTR13 complements if CTR14 and CTR15 are both reset. CTR12 complements if CTR13, CTR14, and CTR15 are all reset.

3.4 Arithmetic Logic Unit (ALU)

The Arithmetic Logic Unit (ALU) consists of logic to perform all required arithmetic, shift and Boolean connect functions. The ALU is halfword oriented and may be divided into three functional groups: an A Register, summing logic, and carry logic. See Figure 2. The B Bus provides the operand for shift operations, and one of the operands for arithmetic and Boolean connect operations. The second operand for arithmetic and Boolean connect operations is in the A Register. The A register is loaded from the S Bus prior to the arithmetic or Boolean connect function.

The ALU receives seven control inputs from the op-code decoder. These inputs are briefly defined in the following list and summarized on the table associated with Figure 2.

Signal Designation	Definition
K$_{j}$1	Subtract
L$_{j}$1	Add or Subtract
M$_{j}$1	All except shifts, AND and AND Immediate
P$_{j}$1	All except Exclusive OR
SHL$_{j}$1	Shift Left
SHR$_{j}$1	Shift Right
X$_{j}$1	Cross Shift

Each micro-instruction which uses any part of the ALU, is described in the following paragraphs. Unless otherwise noted, all gate references in the following paragraphs refer to the arbitrary designations on Figure 2.

3.4.1 Load, Load Immediate, Branch, Test and Command. These micro-instructions cause the contents of the B Bus to pass unaltered through the ALU to the S Bus. This function is performed by Gate 3 of the summing network (Figure 2).

Since these are not arithmetic micro-instructions, the carry circuit is disabled (L$_{j}$1 = 0). This forces all carry circuit outputs high. Also, since these micro-instructions do not have a second operand, the A Register (AR) is not unloaded (UA$_{j}$1 = 0). This forces the GAn$_{j}$0 leads high and the GAn$_{j}$1 leads low. Signals M$_{j}$1 and P$_{j}$1 are high and X$_{j}$1, SHL$_{j}$1, and SHR$_{j}$1 are low as shown on the table.

Gate 3 then has high inputs from GAn$_{j}$0, Cn$_{j}$1, M$_{j}$1, and X$_{j}$0. If Bn$_{j}$1 is also high at this time, then all inputs are high to produce a low out. Gates 1, 2, 4, 5, and 6 are all disabled by signals GAn$_{j}$1, I$_{j}$1, SHL$_{j}$1, and SHR$_{j}$1 which are all low.

3.4.2 Shift Left. This micro-instruction shifts the contents of the B Bus left one place, and places the shifted data onto the S Bus. The bit that is shifted out of bit 0 is lost unless the Carry Out option (CO) is specified. If CO is specified, the bit is stored in the Carry Store flip-flop, FLR12. Bit 15 picks up a zero unless the Carry In option (CI) is specified and FLR12 is set. Gate 5 of the summing circuit is used to perform the Shift Left function.

Signal SHL$_{j}$1 is high during a Shift Left, and signals M$_{j}$1 and SHR$_{j}$1 are low. (See the Table on Figure 2.)

Gate 5 uses SHL1 to gate each bit of the B Bus onto the next most significant position of the S Bus. For example, B15 is gated onto S14, B14 is gated onto S13, etc. In the case of bit 15 of the ALU, CTSL,1 comes high if the CI option is specified, to gate the carry store flip-flop (FLR12,1) into bit 15 of the S Bus. If the CO option is specified and B00,1 is high, SFL12,0 is low to set the Carry Store flip-flop, FLR12. Gates 1, 2, 3, and 4 are all disabled by GAn,1, L,1, and M,1 which are low during the Shift Left micro-instruction.

3.4.3 Shift Right. This micro-instruction shifts the contents of the B Bus right one place, and places the shifted data onto the S Bus. The bit that is shifted out of bit 15 is lost unless the Carry Out option (CO) is specified. If the CO option is specified, the bit shifted out of bit 15 is stored in the Carry Store flip-flop, FLR12. Bit 0 receives a zero unless the CI option is specified and FLR12 is set. Gate 6 of the summing circuit performs the Shift Right Function.

Signal SHR,1 is high at this time and signals M,1, SHL,1, and X,1 are low. Signal SHR,1 is applied to Gate 6, gating each bit of the B Bus onto the next least significant position of the S Bus. For example, B00 is gated onto S01, B01 is gated onto S02, etc. In the case of bit 0 of the ALU, CTSR,1 comes in high if the CI option is specified and FLR12 is set putting a ONE onto bit 0 of the S Bus. Gates 1, 2, 3, 4, and 5 are all disabled by signals GAnn,1, L,1, M,1, and SHL,1 which are all low during a Shift Right micro-instruction. If the CO option is specified, and B15,1 is high. SFL12,0 is low to set the Carry Store flip-flop, FLR12.

3.4.4 Cross Shift. This micro-instruction swaps the upper and lower halves of the B Bus and places the data on the S Bus. The Carry In option (CI) has no meaning for this micro-instruction. If the Carry Out option (CO) is specified, the Carry Store flip-flop (FLR12) is reset. Gate 3 of the summing circuit and Gates 7 and 8 are used to perform this function. Since this is not an Arithmetic micro-instruction, the carry circuits are disabled (L,1 = 0). This forces all carry outputs high. Also, since this micro-instruction does not have a second operand, the A Register (AR) is not unloaded (UA,1 = 0). This forces the GAn,0 leads high, and the GAn1 leads low. Signals M,1, P,1, and X,1 are high at this time, and signals SHL,1 and SHR,1 are low as shown on the table associated with Figure 2.

The B Bus halves are swapped prior to insertion to the ALU. Signal X,0 is low disabling Gate 8 and the cross shifted B Bus input. For example, B07 is applied to ALU bit 15, B06 is applied to ALU bit 14, B08 is applied to ALU bit 0, B09 is applied to ALU bit 1, etc.

Gate 3 then has high inputs from GAn,0, Cn,1, M,1, and the ACSL,1 function. Therefore, if the output of Gate 8 is also high at this time, all inputs are high to produce a low out. Gate 1, 2, 4, 5, and 6 are all disabled by signals GAn,1, L,1, SHL,1, and SHR,1 which are all low. If the CO option is specified, signal SVSC,0 is active. Signal SVSC,0 generates a triggering signal which resets the Carry Store flip-flop, FLR12. (See the schematics in Volume 2 of this manual.)

3.4.5 AND And AND Immediate. These micro-instructions logically AND each bit of the B Bus with the corresponding bit in the A Register (AR), placing the result onto the S Bus. Gate 1 of the summing circuit performs this function. During these micro-instructions, the carry circuit is disabled (L,1 = 0). This forces all outputs of the carry circuit high. Because these micro-instrcutions require a second operand, the A Register (AR) is unloaded (UA,1 = 1). The outputs from the AR are available to the summing circuit. Signal P,1 is high at this time and signals M,1, SHL,1, SHR,1, and X,1 are low.

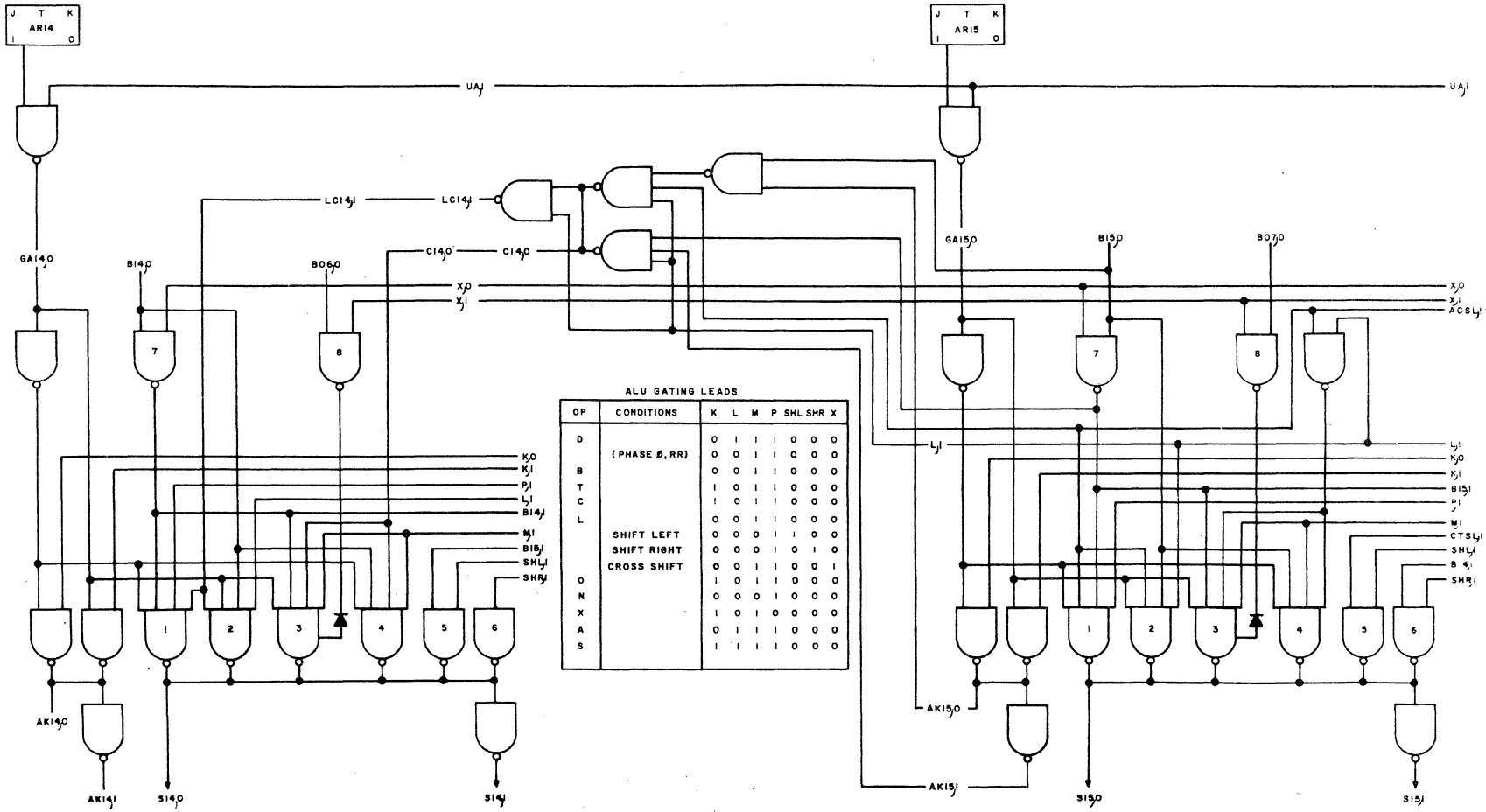


Figure 2. Functional ALU Block Diagram

Gate 3 then, has high inputs from M2 and Cn0. If Bn is set and GAn is reset, all in- from the B Bus and the AR. Therefore, if $Bn_{J}1$ and $GAn_{J}1$ are both high, Sn0 is low and $Sn_{J}1$ is high. Gates 2, 3, 4, 5, and 6 are disabled by singals $L_{J}1$, $M_{J}1$, $SHL_{J}1$, and $SHR_{J}1$ which are all low.

3.4.6 OR And OR Immediate. These micro-instructions logically OR each bit of the B Bus with the corresponding bit in the A Register (AR), placing the result onto the S Bus. Gates 1, 3, and 4 of the summing circuit perform these functions.

During these micro-instructions, the carry circuit is disabled ($L_{J}1 = 0$). This forces all outputs of the carry circuit high. Because these micro-instructions require a second operand, the A Register (AR) is unloaded ($UA_{J}1 = 1$). The outputs from the AR are available to the summing circuit. Signals $M_{J}1$ and $P_{J}1$ are both high and signals $SHL_{J}1$, $SHR_{J}1$, and $X_{J}1$ are low.

Gate 1 has high inputs from $P_{J}1$ and $Cn_{J}1$. If Bn and GAn are also both high, the Gate 1 output is low. Gate 1 checks for the condition where Bn and the corresponding bit of AR are both set. Gate 3 has high inputs from $M_{J}1$, $Cn_{J}0$ and the output of Gate 8. If Bn is set and GAn is reset, all Gate 3 inputs are high, producing a low output. Gate 3, then, checks for the condition where Bn is set and GAn is reset. Gate 4 has high inputs from $M_{J}1$ and $Cn_{J}0$. If GAn is set and Bn is reset, all inputs to Gate 4 are high, producing a condition where Bn is reset and the corresponding bit of AR is set.

The output equation to the S Bus for these three gates during the OR and OR Immediate micro-instructions is:

$$(P_{J}1)(M_{J}1)\boxed{(\overline{BN_{J}1})(GAn_{J}1)+(Bn_{J}1)(GAn)+(Bn_{J}0)(GAn_{J}\overline{1})}$$

Gates 2, 5, and 6 are all disabled by signals $L_{J}1$, $SHL_{J}1$, and $SHR_{J}1$ which are low.

3.4.7 Exclusive OR And Exclusive OR Immediate. These micro-instructions logically Exclusive OR each bit of the B Bus with the corresponding bit in the A Register (AR), placing the result onto the S Bus. Gates 3 and 4 of the summing circuit perform this function.

During these micro-instructions, the carry circuit is disabled ($L_{J}1 = 0$). This forces all outputs of the carry circuit high. Because these micro-instructions require a second operand, the A Register (AR is unloaded ($UA_{J}1 = 1$). The outputs from the AR are available to the summing circuit. Signal $M_{J}1$ is high and signals $P_{J}1$, $SHL_{J}1$, $SHR_{J}1$, and $X_{J}1$ are low.

Gate 3 then has high inputs from M2 and Cn0. If Bn is set and GAn is reset, all inputs to Gate 3 are high, producing a low output. Gate 3, then, checks for the condition where Bn is set and the corresponding bit of the AR is reset. Gate 4 checks for the condition where Bn is reset and the corresponding bit of the AR is set. Gates 1, 2, 5, and 6 are all disabled by signals $P_{J}1$, $N_{J}1$, $SL_{J}1$, and $SR_{J}1$ which are low.

The output equation to the S Bus for these two gates during these micro-instruction is:

$$M_{J}1(Bn_{J}0\ GAn_{J}1 + Gn_{J}1\ GAn_{J}0)$$

3.4.8 Add And Subtract. When an addition or subtraction is performed, each adder-subtracter stage looks at three inputs: one bit from each of the two operands plus the carry bit generated from the previous stage. For explanation purposes, the inputs are designated: Cn (carry or borrow bit coming in from some previous stage), An (the first operand bit), and Bn (the second operand bit).

The adder-subtracter unit has two outputs: Sum and Carry. During Subtract however, the sum output can be looked at as the difference output, while the carry output can be looked at as the borrow output.

Prior to the adder-subtracter description, the terminology and method of signal designations employed in this system must be understood. The following conventions apply to both the descriptions in this Section, and to the functional schematics provided in Volume 1 of this manual.

The operands are designated B00 through B15, and GA00 through GA15 indicating the B Bus and Gated A Register resepctively. The least significant bits are B15 and GA15. The Sum bits are designated S00 through S15, and correspond to the operand bits which produce them. For example, B15 and GA15 are combined to form S15, the least significant Sum bit. The Carry bits are designated CSV and C00 through C14. The CSV (Carry Save) bit is the carry from the most significant bits (B00 and GA00). Each remaining carry bit is designated in accordance with the stage it carries into. For example, B15 and GA15 generate C15, which is summed with B14 and GA14.

To generate an output in a single stage adder-subtracter unit, the following conditions can exist:

Cn	0	0	0	0	1	1	1	1
An	0	0	1	1	0	0	1	1
Bn	0	1	0	1	0	1	0	1
Sum	0	1	1	0	1	0	0	1
Difference	0	1	1	0	1	0	0	1
Carry Out	0	0	0	1	0	1	1	1
Borrow Out	0	1	0	0	1	1	0	1

Sum-Difference: Note on the table that the sum and the difference are equal in a single stage adder-subtracter. The equation for either the sum or the difference is:

$$AnBnCn + An\overline{Bn}\overline{Cn} + \overline{An}Bn\overline{Cn} + \overline{An}\overline{Bn}Cn$$

Carry Out: According to the table, the equation for the Carry Out is:

$$AnBn\overline{Cn} + An\overline{Bn}Cn + \overline{An}BnCn + AnBnCn$$

the equation can be reduced to:

$$AnCn + AnBn + BnCn$$

This means that any combination of two or more bits set will generate a Carry Out. The equation can be factored to read:

$$An(Bn+Cn) + BnCn$$

Borrow Out: In subtraction, according to the table, the equation for Borrow Out can be written as:

$$An\overline{Bn}Cn\text{-}1 + \overline{An}BnCn\text{-}1 + \overline{An}Bn\overline{Cn}\text{-}1 + \overline{An}\overline{Bn}Cn\text{-}1$$

The equation can be reduced to:

$$\overline{An}Bn + \overline{An}Cn + BnCn$$

This equation can be factored to read:

$$\overline{An}(Bn+Cn) + Bn\,Cn$$

To simplify the equations, Add and Subtract are designated in the equations as:

$\overline{K}$ = Add
K = Subtract

The equations for carry and borrow can be combined to form the equation:

$$Cn = \overline{K}An+1\ (Bn+1 + Cn+1) + K\overline{An+1}\ (Bn+1 + Cn+1) + Bn+1\ Cn+1$$

The previous equation can be expanded to produce:

$$Cn = \overline{K}An+1\ \overline{Bn+1} + \overline{K}An+1\ Cn+1 + K\overline{An+1}\ Bn+1\ K\overline{An+1}\ Cn+1 + Bn+1\ Cn+1$$

This equation can be factored to produce:

$Cn = (\overline{K}An+1 = \overline{KAn+1})(Bn+1 + Cn+1) + (Bn+1\ Cn+1)$

To further simplify this discussion, define:

$\overline{K}An+1 + \overline{KAn+1} = AK\ n+1$

In order to make a one bit "look ahead" carry/borrow out of the last Cn equation, Cn+1 must be expressed in terms of AKn+2, Bn+2, and Cn+2.

$Cn+1 = AKn+2\ Bn+2 + AKn+2\ Cn+2 + Bn+2\ Cn+2$

Substitute this equation for Cn+1 in the previous Cn equation to form the following one bit "look ahead" carry/borrow equation

$Cn = AKn+1\ Bn+1 + AKn+1\ AKn+2\ Bn+2 + AKn+1 + AKn+2\ Cn+2 + AKn+1\ Bn+2\ Cn+2 + Bn+1\ AKn+2 + Bn+2 + Bn+1\ AKn+2\ Cn+2 + Bn+1\ Bn+2\ Cn+2$

3.5 Counter Dependent Operations

The special commands - Multiply, Divide, Repeat - force the Processor into a counter dependent mode of operation designed to execute the specified function. The particular mode the Processor assumes is defined by bits 4 and 5 of the Command micro-instruction.

Command Divide mode is defined as bit 4 set and bit 5 reset. Command Multiply mode is defined as bit 4 reset and bit 5 set. Command Repeat mode is defined as bits 4 and 5 both set.

The circuitry which implements these modes is shown on Sheets 19 and 20 of FS45. The two flip-flops at the left of Sheet 19 (FNY,0 and FNY,1) serve as a two bit instruction register used to hold the Processor in the desired mode. The mode register is DC cleared by SCLR,0C. SCLR,0C goes low on initialize. The register is cleared at the conclusion of the iterative mode; on counter empty for Multiply or Divide, or when the counter equals one or zero on Repeat mode.

FNY,0 is set when signal SFNY,1 is high. SFNY,1 is high on a Command micro-instruction if RD bit 4 is set. When a Command instruction is decoded, signal CMND,0 goes low (see Sheet 19). CMND,0 is inverted and ANDed with RD bit 4 to produce SFNY,1. FNY,1 is set when signal SFNY,1 is high. SFNY,1 is high on a Command instruction if RD bit 5 is set.

Signals SFNY,1 and SFNY,1, and their complements, are decoded to produce signals that precondition the other two flip-flops on Sheet 19. Flip-flop NORCL (F8), when set, stops the ROM clock (RCL,0). RAL will not increment further until NORCL resets. NORCL is clocked set on the Command Multiply instruction (CMPY,0), the Command Divide instruction (CDVD,0), or on the Command Repeat instruction (CRPT,0) provided the counter does not equal one or zero.

Flip-flop NORCL remains set until the iterative mode is terminated, at which time RAL resumes normal counting. NORCL is clocked reset on Multiply or Divide when the counter decrements from hexadecimal 1 to 0 or on Repeat mode when the counter equals hexadecimal 2, 1, or 0.

Flip-flop FRCGP,1 (M8), when reset, allows the user's indirect registers to be selected from the addresses specified by the R1 and R2 fields of the Instruction Register (IR bits 8:11 and IR bits 12:15 respectively). If

FRCGP,1 is set and the IR Source or Destination address is even, the next sequential address is selected. This function is shown on Sheet 8 for Source addressing and on Sheet 10 for Destination addressing. FRCGP,1 literally means "force General Register Plus 1".

FRCGP,1 is clocked-set on the Command Divide instruction (CDVD,0) and within the Divide loop on the Add instruction. FRCGP,1 is clocked set within the Multiply loop when the Add instruction is not being performed. FRCGP,1 toggles reset on the next system clock (CL0D).

The state of FNY,0 and FNY,1 is decoded to establish the mode of operation. The mode decoders are shown on Sheet 20 of FS45. Processor mode is used to generate the signals DECTR,0 which decrements the counter, and FRAL07 which is applied to the ROM address buffers, forcing the even address in ROM to be strobed.

When the Processor enters a counter dependent mode, the clock that increments RAL leaving the Command micro-instruction also sets NORCL which keeps RAL from incrementing further. Since the special Command instruction must be wired into an odd address when NORCL sets, RAL contains an odd address. By using the FRAL07 signal, two different words can be selectively strobed from ROM without changing the address.

3.5.1 Multiply. The 30-2 hardware multiply is initiated with a Command micro-instruction. The following preliminary conditions are assumed; the Instruction Register Destination field (IR 8:11) contains an even register address; the next sequential address (YDP,1) points to the register containing the multiplier. The multiplicand has been placed in the A Register, and the counter is cleared. The Multiply command must be wired into an odd address. The next two ROM locations must be wired as follows:

odd	C	MPY
even	L	YD, YD, SR
odd	A	YD, YD, CO

The C MPY instruction is wired into an odd address. The clock that strobes that instruction into RD increments RAL to the next address; that of the Load instruction. The clock that strobes the Load into RD increments RAL to the next sequential address: that of the Add instruction. The same clock sets the FNY flip-flops to the 01 condition which defines the Multiply mode for the remainder of the sequence, and sets the NORCL flip-flop. RAL contains the address of the Add micro-instruction and will not increment further until NORCL resets.

RD contains L YD, YD, SR. As this instruction is performed, signal FRAL07 is active, so the ROM address decoders see the address of the Load instruction rather than the address of the Add. The next clock strobes the Load instruction a second time. The same clock sets the FRCGP,1 flip-flop, making the YD field of IR appear odd, which causes the Load instruction to be executed as L YDP,1, YDP,1, SR. A 32-bit Shift Right is performed with YD containing the most significant 16 bits and YDP,1 the least significant 16 bits. If the 'implied' instruction - L YDP,1, YDP,1, SR - produces a carry (B15 = 1), the FRAL07 signal is false, letting the ROM address decoders see the address of the Add instruction. If the implied instruction does not produce a carry (B15 = 0), FRAL07 remains active, causing the Load instruction to be read again. FRCTP,1 then resets, and the Add or the first Load is performed. If the hardware gets to the Add instruction, the multiplicand (AR) is added to the product (YD); FRAL07 is active so the sequence returns to the first Load.

The counter is decremented on the first Load instruction. The first Decrement wrapped the counter from X'0' to X'F'. When the counter is decremented from X'1' to X'0', the NORCL flip-flop is reset.

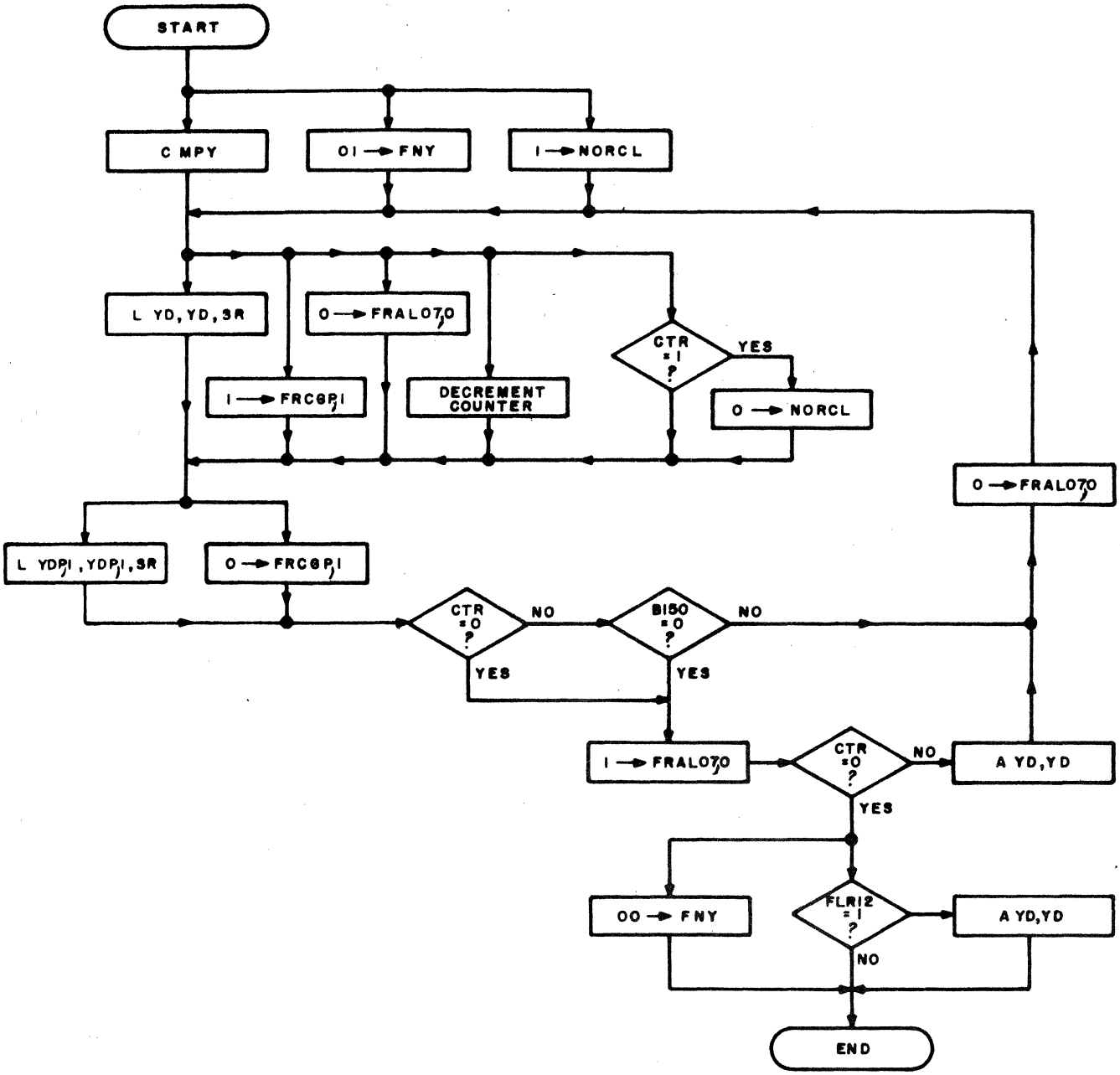


Figure 3. Hardware Multiply Flow Chart

RAL increments at the conclusion of the implied instruction. With the counter equal to zero, the Add will be performed regardless of B15. If the Add should not be performed (FLR12 = 0), the Destination is disabled (KILDST0). At the conclusion of the Add, the FNY flip-flops are reset, the sequence is terminated and the next sequential micro-instruction is performed. See Figure 3.

3.5.2 Divide. Hardware divide is initiated with a Command micro-instruction. The following preliminary conditions are assumed; the Instruction Register Destination field (IR 8:11) contains an even address, the address of the most significant 16 bits of the dividend. The next sequential address (YDP1) points to the register containing the least significant 16 bits of the dividend. The divisor is negative and resides in the A Register. The counter is cleared. The Divide command must be wired into an odd address. The next two ROM locations must be wired as follows:

odd	C	DIV
even	L	YD, YD, SL
odd	A	YD, YD, CO

The C DIV instruction is wired into an odd address. The clock that strobes that instruction into RD increments RAL to the next address; that of the Load instruction. The clock that strobes the Load instruction into RD increments RAL to the next sequential address; that of the Add instruction. The same clock sets the FNY flip-flops to the 10 condition which defines the Divide mode for the remainder of the sequence, and sets both the NORCL and FRCTP1 flip-flops. RAL contains the address of the Add instruction and will not increment further until NORCL resets.

RD contains L YD, YD, SL; but because FRCTP1 is set, the Register Address Decoders interpret this instruction as L YDP1, YDP1, SL. As this instruction is performed, the FRAL07 signal is active so the ROM address decoders see the address of the Load instruction, rather than the Add. The next clock strobes the Load instruction a second time and resets FRCTP1. The register address decoders interpret the instruction in RD as it acutally appears - L YD, YD, SL. A 32-bit Shift Left is performed. As the second Load is performed, FRAL07 is false. The next clock strobes the Add instruction into RD. If the Addition does not result in a carry (CSV = 0), the Destination is disabled and the partial remainder (YD) remains unchanged. If the carry does result (CSV = 1), the sum produced replaces the partial remainder (YD). During the Add instruction, FRAL07 is active. The sequence returns to the first Load upon conclusion of the Add, setting FRCTP1 to access the least significant 16 bits of the dividend first. The carry produced by the Addition is shifted to the least significant bit of YDP1, forming the quotient.

The counter is decremented on the first Load instruction (L YDP1, YDP1, SL). The first decrement wrapped the counter from X'0' to X'F'. When the counter is decremented from X'1' to X'0', the NORCL flip-flop is reset. RAL increments at the conclusion of the second Load instruction (L YD, YD, SL). With the counter equal to zero, the same Destination inhibits pertain to the final Add. Signal FRAL07 is false so the sequence will not return to the first Load. The FNY flip-flops are reset, and the next sequential micro-instruction is performed. See Figure 4.

3.6 Phase Control

The Phase Control hardware is shown on Sheets 16, 17, and 18 of FS45 and in block diagram form on Figure 5. Phase is a hardware condition that is affected by the Decode micro-instruction and user activity. Phase, in-turn, affects actions resulting from the Decode micro-instruction. (See 30-2 Micro-Instruction Reference Manual, Publication Number 29-032.)

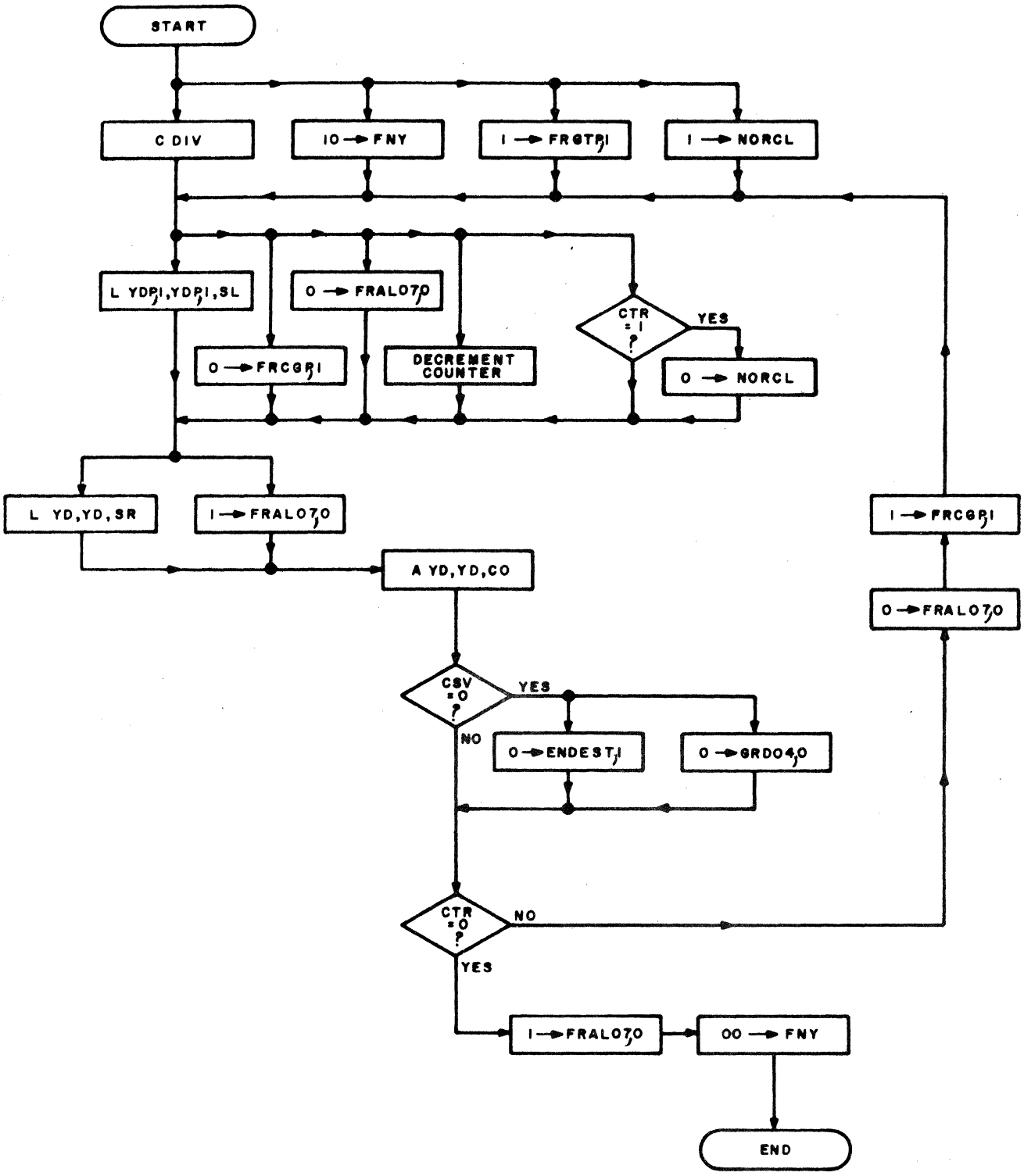
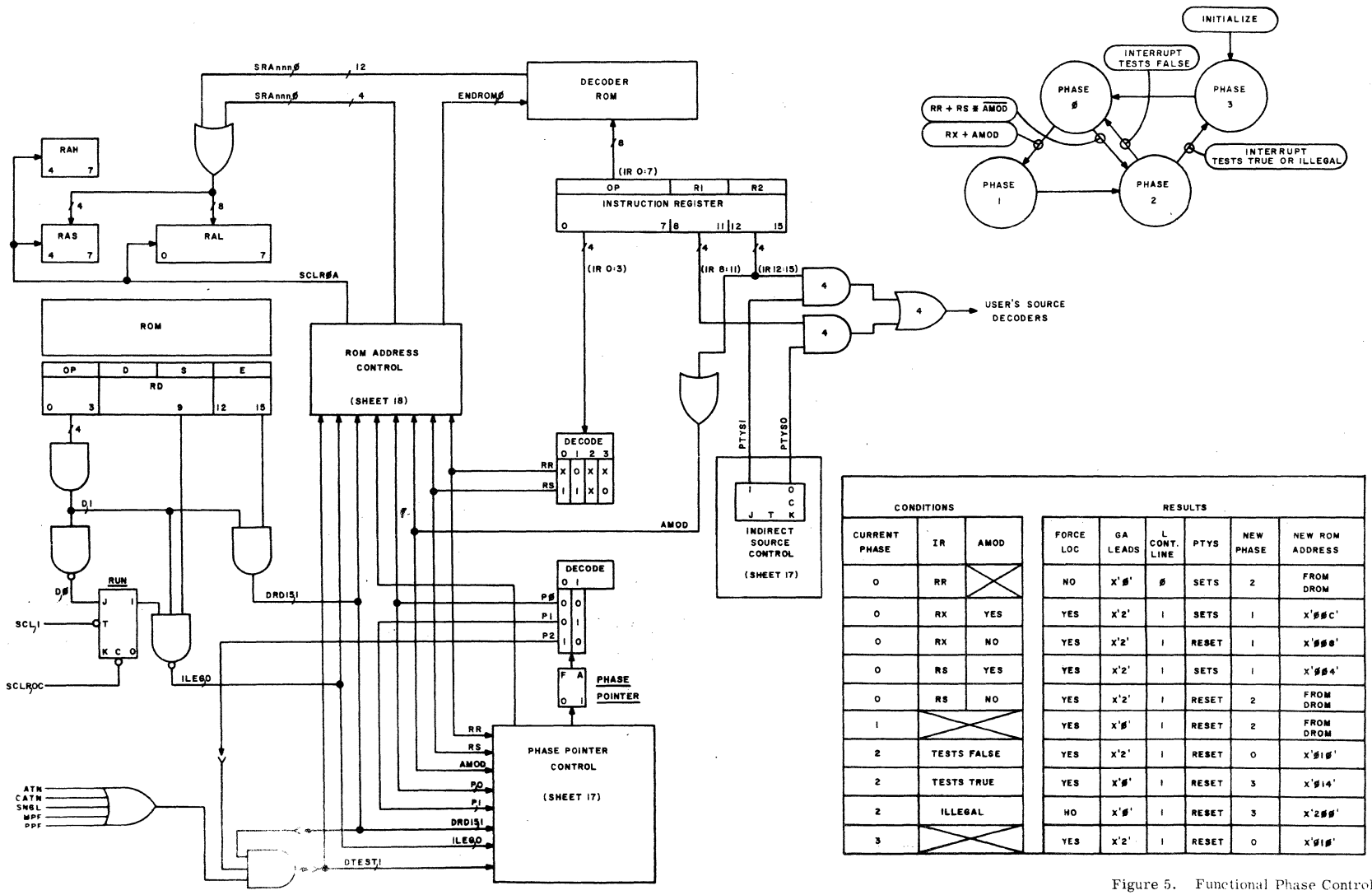


Figure 4. Hardware Divide Flow Chart



CONDITIONS			RESULTS					
CURRENT PHASE	IR	AMOD	FORCE LOC	GA LEADS	L CONT. LINE	PTYS	NEW PHASE	NEW ROM ADDRESS
0	RR	X	NO	X'Ø'	Ø	SETS	2	FROM DROM
0	RX	YES	YES	X'2'	1	SETS	1	X'ØØC'
0	RX	NO	YES	X'2'	1	RESET	1	X'ØØ8'
0	RS	YES	YES	X'2'	1	SETS	1	X'ØØ4'
0	RS	NO	YES	X'2'	1	RESET	2	FROM DROM
1	X	X	YES	X'Ø'	1	RESET	2	FROM DROM
2	TESTS FALSE		YES	X'2'	1	RESET	0	X'Ø1Ø'
2	TESTS TRUE		YES	X'Ø'	1	RESET	3	X'Ø14'
2	ILLEGAL		NO	X'Ø'	1	RESET	3	X'2ØØ'
3	X	X	YES	X'2'	1	RESET	0	X'Ø1Ø'

Figure 5. Functional Phase Control Diagram

The two stage Phase Pointer defines the four hardware Phases. The Phase Pointer is a modified up counter able to skip counts. Reference Figure 5.

From count 00 (Phase Zero) the pointer can go to a count of 01 (Phase One) or 10 (Phase Two). From count 01 (Phase One) the pointer will go to a count of 10 (Phase Two). From count 10 (Phase Two) the pointer can go to a count of 00 (Phase Zero) or 11 (Phase Three). From count 11 (Phase Three) the pointer will go to a count of 00 (Phase Zero). A Phase change can only occur on a Decode micro-instruction. Initialize unconditionally forces the Phase Pointer to Phase Three.

Each hardware phase has corresponding set(s) of micro-code routines. The starting address of these routines is produced by the phase control hardware.

When Phase Zero is entered, the ROM address is forced to hexadecimal 010, the starting address (entry point) of the Phase Zero firmware set. Phase Zero is dedicated to User's instruction class decoding. Depending on the User's instruction which is resident in the Instruction Register (IR), either Phase One or Phase Two will follow the Decode micro-instruction terminating Phase Zero. The ROM address is forced to hexadecimal 00C if the User's instruction is of the RX format (Operation Code 4x, 6x, or Dx) and if the R1 field of IR does not equal zero. The ROM address is forced to hexadecimal 008 if the User's instruction is of the RX format and if the R1 field of IR is equal to zero. The Phase One firmware sets are dedicated to second operand address arithmetic.

The Phase Two entry point is derived from the Decoder ROM (DROM). The DROM may have up to 128 twelve bit words wired into it. Words in DROM are addressed by the Operation Code field of IR (IR 0:7). The DROM has a word line for each instruction in the user's repertoire. Enabling the DROM causes the selected word line to be strobed during the Phase Zero or Phase One Decode instruction. The readout providing the starting address of a Phase Two firmware set, is jammed into RAS and RAL.

Instructions not in the user's repertoire are illegal and will not have a corresponding word line in DROM. When Phase Two is entered and a non-existent DROM word line is strobed, the readout, all zeros, is jammed into RAS and RAL. Location zero (000) in ROM must be wired with all zeros (0000). When ROM address hexadecimal 000 is read, the contents are placed in RD. All zeros in RD are defined as 'illegal' and will result in an unconditional Phase Three. ROM Address is forced to hexadecimal 200, the entry point of the illegal instruction trap.

When Phase Two is exited, the Decode micro-instruction will test for interrupts. The interrupts tested are I/O Interrupt (ATN), Console Interrupt (CATN), Console Single mode (SNGL), Memory Parity Fail (MPF), Primary Power Fail (PPF), and FAST. If any interrupt is true, Phase Three will be entered and the ROM addressing will be forced to hexadecimal 014. If no interrupts are pending, Phase Zero will be entered and the ROM address will be forced to hexadecimal 010.

The Phase Three firmware sets - entry points 000, 014, and 200 - are dedicated to display and interrupt support. When Phase Three is exited, Phase Zero will be entered and ROM address will be forced to hexadecimal 010, the Phase Zero entry point.

The Decode micro-instruction (op-code hexadecimal 0) is detected by Gate 21-30 on Sheet 16 of FS45. The signal D0 will be low on a Decode instruction or on Illegal (all zeros in RD). Signal D0 appears at the clock skipping circuits on Sheet 22. Decodes or Illegals are always two clocks long. The Decode instruction is distinguished from

the Illegal condition by at least one high bit in RD bits 4:15. It turns out that the Source registers the micro-programmer is allowed to use on a Decode instruction - namely LOC and YD - both have bit 9 set. If the op-code is hexadecimal 0 and bit 9 is set, RD contains a Decode instruction. If the op-code is hexadecimal 0 and bit 9 is not set, RD contains all zeros. Unless RD was cleared because of system initialize, signal ILEG,0 will go low forcing Phase Three, and the ROM address to hexadecimal 200.

When the Processor is initialized, the signals SCLR,0A, SCLR,0B, and SCLR,0C are low. SCLR,0A clears the ROM Address registers. SCLR,0B clears the ROM Data Register (RD). SCLR,0C clears, among other things, the RUN flip-flop on Sheet 16 of FS45.

The zero op-code(RD 0:3) is detected by Gate 21-30 (A3). Signal D0 goes low and D1 goes high. D1 is ANDed with RD0,0 and RUN1 at Gate 61-07 (D7). If RUN were not reset at this time, ILEG,0 would go low and ILEG,1 would go high. The RUN flip-flop is set when D,0 (RD 0:3 = 0) goes false.

When a Decode micro-instruction is performed, the extended op-code field (RD 12:15) takes on new meaning. Bit 12 set generates a memory read cycle. Gate 21-03 on Sheet 16 generates a memory start (STRT,0) on the first half (ENSKP,1) of a Decode instruction (RD 0:3 = 0) if RD bit 12 and the Processor is not in Phase Zero of an RR User's instruction (P01*IR01,0). If bit 13 is set, the contents of the FLR are copied in the CCR. Gate 21-25 on Sheet 14 generates the load CCR pulse (LCC,0) on the first half (ENSKP,1A) of a Decode instruction (D1) if RD bit 13 is set. Signal LCC,0 jams the FLR to the CCR on Sheet 6. If bit 14 is set, the Flag Register, Counter Register, BANK flip-flop and UTility flip-flop are cleared. Gate 61-22 on Sheet 14 generates the clear lead (CLR,0) on the second half (ENSKP,0) of a Decode instruction (D,1) if RD bit 14 is set. Signal CLR,0 allows FLR, CNTR, BANK, and UT to clock clear. If RD bit 15 is set, the Phase Pointer and ROM Address will change. Gate 50-34 on Sheet 16 generates the phase change signal (DRD15,0) if RD bit 15 is set on a Decode instruction (D1).

If a phase change is called for (DRD15,0 = 0), the ROM Address registers are cleared (SCLR,0A = 0) then a new address is loaded to RAS and RAL. (See Sheet 18 of FS45.) The new address to ROM is generated by the Decoder ROM (DROM) if the Processor is going into Phase Two. Otherwise, the address is generated by the hardware shown on Sheet 18.

The DROM will strobe an address defined by the user's operation code (IR 0:7). The DROM readout is jammed into RAS and RAL on the first half of the Decode instruction.

Gate 20-03 on Sheet 18 generates the SCLR,0A signal to clear the ROM address registers. SCLR,0A goes low on Illegal (ILEG,0) or on the Phase Change option of the Decode instruction (DRD15,0) if ENSKP is set (ENSKP,1A). ENSKP,1A limits the SCLR,0A signal to the first half of the instruction.

The Processor enters Phase Two if it is currently in Phase One (P11), if it is in Phase Zero and IR holds an RR format instruction (P0,1 IR01,0) or if it is in Phase Zero and IR holds a non-indexed RS format instruction (P0,1 RS,1 AMOD,0). Signal GTP2,1 on Sheet 17 goes high to sequence the Phase Pointer to a count of 2 and gate the DROM Enable lead (ENDROM,0) on Sheet 18.

The Decode micro-instruction terminating the Phase Zero firmware set must be written with YD the Source and AR the Destination. All other Decode instructions must have LOC the Source and Destination. The Phase Zero decode will load AR from YD if the IR holds an RR format instruction. The conditions P01 IR01,0 D1 produce the signal P0RR,0 at Gate 61-11 on Sheet 16.

PORB,0 disables the L control line on Sheet 15. The B Bus will pass unaltered through the ALU to the S Bus and be loaded to the AR on the 'skipped' clock. For the second half of the Decode instruction, all Destinations are disabled (KILDST,0 = 0 on Sheet 16). This is done because the PTYS flip-flop sets on the skipped clock, changing the Source information on the S Bus. On the Phase Zero RR Decode, only RCL,0 and ROMSTB,0 are skipped. Signal SKPSTB,0 on Sheet 16 disables the normal clock skipping function of ENSKP (see Sheet 22) and disables ROMSTB,0 on Sheet 23. SKPSTB,0 also produces NORCL,0 (see Sheet 16) which disables RCL,0. If the IR does not hold an RR format instruction, the lead FRCLOC,0 goes low (Gate 21-12 on Sheet 16) forcing the Location Counter to be both Source and Destination.

The Decode instruction does not unload the A Register (UA,1 0 on Sheet 8). The GA leads are false. When LOC is forced as Source and Destination, zeros would be added to the Location Counter. It is possible to force GA14 active (GA leads equal hexadecimal 0002) and increment the Location Counter by two on a Decode instruction. This function is realized on Sheet 18 of FS45.

3.7 I/O Control

The Processor I/O control logic is shown on Sheet 24 and 25 of FS45. Refer to the I/O Section of this manual for more information on the I/O System.

The EXECUTE pushbutton is shown in the top left area of FS45-24. On a standard Digital System, the outputs from the EXECUTE switch, ESNC,0 and ESNO,0, leave the side of the Sheet and go to the Display Controller (FS31). The Display Controller, in turn, generates a Console Attention signal (CANT,0) which is returned to the Processor at D9 of FS45-24. The signal is inverted and sent to the Test Logic shown on FS45-12. Gate 50-25 disables CATN,0 from the EXECUTE flip-flop (D6). When the Display Controller mother-board is plugged in, signal ENBC,0 is interlocked to ground, disabling Gate 50-25 and 50-46. Gate 50-46 is used in conjunction with the Autoload option.

Although the Display Panel is standard on GE-PAC 30 Digital Systems, systems without the Display are available on special order. Such systems are designated Auto-Load Systems and are provided with a smaller panel which contains the following control and indicators:

1. POWER Switch and as associated indicator
2. EXECUTE pushbutton
3. INITIALIZE pushbutton
4. AUTO LOAD Switch
5. Wait indicator

All switches and indicators except AUTO LOAD perform the same functions as in a system with a Display. The AUTO LOAD Switch permits entry of the 50 Loader from tape (rather than manual entry as performed with a standard system). Refer again to FS45-24. On an AUTO LOAD system, the EXECUTE Switch outputs are applied to cross-coupled gates which provide the T input to the flip-flop in areas D5. This flip-flop generates Console Attention (CATN,1) in an Auto Load system. The Auto Load system tests CATN,1 to determine whether to halt or continue. System Clear (SCLR,0) sets the flip-flop initially via the SJK,0 input (Area E2). The system therefore runs when power is applied. The first time EXECUTE is depressed, the flip-flop is toggled reset, and CATN goes low. The system halts after the next sequence. The next time the EXECUTE pushbutton is depressed, the flip-flop is set. CATN,1 goes high, and the system runs. Note the gate in area F8. This gate produces a Single (SNGL,0) signal in an Auto Load System when the AUTO LOAD Switch is depressed. Note again, that this paragraph applies only to Auto Load systems.

All I/O support is done with the Load micro-instruction. I/O operations can be classified as either input oriented or output oriented. If I/O appears in the Source field of a Load instruction, an input operation is desired. When I/O is the Source (RD bits 8:11 equal hexadecimal D), signal UIO$_{,}$0 will be low (FS45-9N8). If I/O appears in the Destination field of a Load instruction, an output operation is desired. When I/O is the Destination (RD bits 4:7 equal hexadecimal D), signal LIO$_{,}$0 will be low (FS45-10D8).

An I/O operation is initiated if I/O is the Source or Destination (UIO$_{,}$0 or LIO$_{,}$0 = 0) of a Load micro-instruction (LOAD$_{,}$0 = 0). LOAD$_{,}$0 will be low if RD bits 0:3 equal hexadecimal 4, (FS45-14K9). If an I/O operation is qualified by the above conditions, signal GIO$_{,}$1 goes high (FS45-24P8), enabling the timing on Sheet 25 to sequence. The timer sequences will be discussed separately for input operations and output operations.

3.7.1 Input I/O Operations. The initialized state of the I/O Timer is DST set and SOPC reset. GIO$_{,}$1 and SOPC$_{,}$0 are two inputs to the NAND gate at 25E5. The third input is the delayed CCL$_{,}$0. When CCL$_{,}$1 goes false, the output of Gate 41-05 rises to logic ONE at the time constant defined by the 100pf capacitor and the internal 2K resistor (about 40 ns); at which time the output of Gate 20-07 goes low forcing flip-flop DST reset.

With DST reset, signal DST$_{,}$0 (25K6) is low to stop the system clocks (see Sheet 22).

Signal UIO$_{,}$0 is low. UIO$_{,}$0 is one input to an AND tied inverter pair. The other input is Gated False Sync (GFS$_{,}$1). The output, UDIO$_{,}$1, (FS45-24P7) is ANDed with the Data Request Lines (DRL 0:7) forming the B Bus bits 8:15 (see Sheet 3P7). Data on the DRL's are gated to the B Bus unless a false Sync condition exists. The B Bus data are duplicated to the S Bus.

Signal DST$_{,}$0 and the inverted SYN$_{,}$0 are inputs to an AND tied inverter pair at 25E7. SYN$_{,}$0 will be false at this time. The high output from this inverter pair allows flip-flop SOPC to toggle set. When SOPC is set, a control line to I/O is enabled. The control line is selected by bits 14 and 15 of RD. If RD 14:15 equal 01, Data Request (DR$_{,}$0) is active. If RD 14:15 equal 10, Acknowledge (ACKA00$_{,}$0) is active. If RD 14:15 equal 11, Status Request (SR$_{,}$0) is active. The addressed Device Controller should respond to the control line with data on the DRL's and a Sync.

The positive transition of SOPC setting starts the False Sync Delay (25F2). The False Sync circuit assures that the Processor does not hang up waiting for a Sync response that will not return - e.g. the device addressed is unable to execute the command, or is non-existent. The False Sync flip-flop (E3 and E4) is set to generating GFS$_{,}$1 (25K3) if SYN$_{,}$0 is not received with approximately 45 microseconds after the control line is activated.

The setting of the False Sync flip-flop is enabled by signal SOPC$_{,}$1A on Gate 20-06. The quiescent state of the False Sync flip-flop is both outputs high. Until the False Sync delay times out, signal FSYN$_{,}$1 equals a zero, keeping output 20-06 high. Until SYN$_{,}$0 returns, signal SYN$_{,}$1 is a zero, keeping output 21-06 high. The capacitors shown at 25 C1 charge to the threshold level of the output circuits at G2 after about 45 microseconds. Normally, by this time, SYN$_{,}$0 has been received from the addressed Device Controller, indicating that the specified operation has been completed. SYN$_{,}$0 is inverted at Gate 21-00. SYN$_{,}$1 resets the False Sync flip-flop.

Meanwhile, the timer has remained with SOPC set and DST reset. DST will not set until a Sync is received or False Sync is generated. When either SYN$_{,}$1 or FSYN$_{,}$1 goes high, the appropriate False Sync flip-flop output goes low, and Gate 61-06 (H4) goes high allowing the timer to resume.

DST sets, placing a low on the J input to SOPC and a high on the K input. The clock inhibit signal, DST,0 (FS45-25K6) is removed. The next system clock gates the Device Controller response (on the S Bus) to the Destination Register specified. The same clock pulls a new instruction from ROM and resets SOPC.

If the next micro-instruction is also an I/O Load, GIO,1 remains true and DST,0 goes back true stopping clocks until this I/O operation is finished and the next one begins. If the next micro-instruction is not an I/O Load, signal GIO,1 will go low and the timer assumes its initialized state.

When SOPC resets, the control line (DR,0, ACKA0,0 or SR,0) is dropped. When the control line drops, the device controller drops its sync. If a false sync is generated, Gate 51-36 (25N3) goes low to set the Overflow flag in FLR, indicating that a False Sync was received. In addition, if False Sync is generated on a Status Request, B Bus bit 13 is forced active (FS45-24P9). This gates a hexadecimal '0004' to the Destination register, indicating the Examine Status response.

3.7.2 Output I/O Operations.

The initialize state of the I/O Timer is DST set and SOPC reset. Signal LIO,0 is inverted at Gate 21-44 (FS45-24E6) and NANDed with S Bus bits 8:15 producing the Data Available Lines, DAL 0:7 (see Sheet 3M7).

GIO,1 and SOPC,0 are two inputs to the NAND gate at 25E5. The third input is the delayed CCL0. When CCL,1 goes false, the output of Gate 41-05 rises to logic ONE at the time constant defined by the 100pf capacitor and the internal 2K resistor (about 40 ns); at which time the output of Gate 20-07 goes low forcing flip-flop DST reset.

With DST reset, signal DST,0 (25K6) is low to stop the system clocks (see Sheet 22). SOPC set gates the control line to I/O. Signal DST,0 and the inverted SYN,0 are inputs to an AND tied inverter pair at 25E7. SYN,0 will be false at this time. The high output from this inverter pair allows flip-flop SOPC to toggle set. The control line is selected by RD bits 14 and 15. If RD 14:15 = 01, Data Available (DA,0) is active. If RD 14:15 = 10, Address (ADRS,0) is active. If RD 14:15 = 11, Command (CMD,0) is active. The address Device Controller should respond to the control line with a Sync.

The False Sync Delay starts timing out. If SYN,0 is received or False Sync is generated, (FSYN1 = 1), the Timer is allowed to resume sequencing. DST clocks set dropping the control line and the clock inhibit, DST,0. The next system clock pulls a new instruction from ROM and resets SOPC. SOPC disengages the False Sync flip-flop. If a False Sync is generated, the Overflow flag in FLR is set.

4. MAINTENANCE

This Section describes maintenance procedures which may be used to check and, if necessary, adjust the Processor.

NOTE

Two optional maintenance aids available from GE-PAC 30 greatly facilitate maintenance on the Processor: The System Test Set, and the X-Ray ROM. GE-PAC 30 urges all users who intend to maintain their own systems to purchase both a System Test Set and an X-Ray ROM. Maintenance procedures using these aids are provided with the aids.

4.1 Clock Timing

Use the following procedure to check system clock timing.

1. Force the RD Register to Zero by grounding the SCLR,0B signal at pin 203-2100.

CAUTION

Exercise care in grounding points in the digital system as some components may be damaged if the wrong pins are grounded.

2. Disable the Decoder hardware by grounding the D1 signal at pin 207-1300.

CAUTION

The adjustments described in this Section are carefully set at the factory using sophisticated test equipment not normally available in the field. The adjustments are very stable and should not require field adjustment. The adjustments should be changed only after the check provided indicates that they are out of tolerance and there are no faulty components in the system.

3. Use an oscilloscope to check that the following Y Switch outputs are 120 to 150 nanoseconds wide.

YS00,1	017-0700
YS01,1	117-0700
YS02,1	217-0700
YS03,1	016-0700
YS04,1	115-0700
YS05,1	015-0700
YS06,1	216-0700
YS07,1	116-0700

If any of the switch outputs are not within the 120 to 150 nanosecond range, check the associated logic.

4. Force RAH07 to binary ONE by grounding pin 018-1001. Check that the following Y Switch outputs are 120 to 150 nanoseconds wide.

YS08,1	215-0700
YS09,1	014-0700
YS10,1	114-0700
YS11,1	214-0700
YS12,1	012-0700
YS13,1	213-0700
YS14,1	113-0700
YS15,1	013-0700

If any of the switch outputs are not within tolerance, check the associated logic.

5. Compare STRB,1 on the DROM Transformer board (Figure 6) and YS15,1 (013-0700). STRB,1 should start 75 ±10 nanoseconds after YS15,1. If out of tolerance, and no faulty components are found, adjust ROMSTB,0 position (C01-P472) to provide the 75 nanosecond difference.

6. Check that ROMSTB,0 is 45 to 60 nanoseconds wide. If ROMSTB,0 pulse width is out of tolerance, and no faulty components are found, adjust ROMSTB,0 pulse width (Figure 6) to 50 nanoseconds.

7. Check the three strobes on the DROM Transformer board (Figure 6) and verify that STRB,1 is coincendental with, or starts later than, CLKA,1 and CLKB,1.

8. Observe all ROM readouts (listed on Table 3). False outputs on the CRDXX,0 leads should be less than 25 nanoseconds wide.

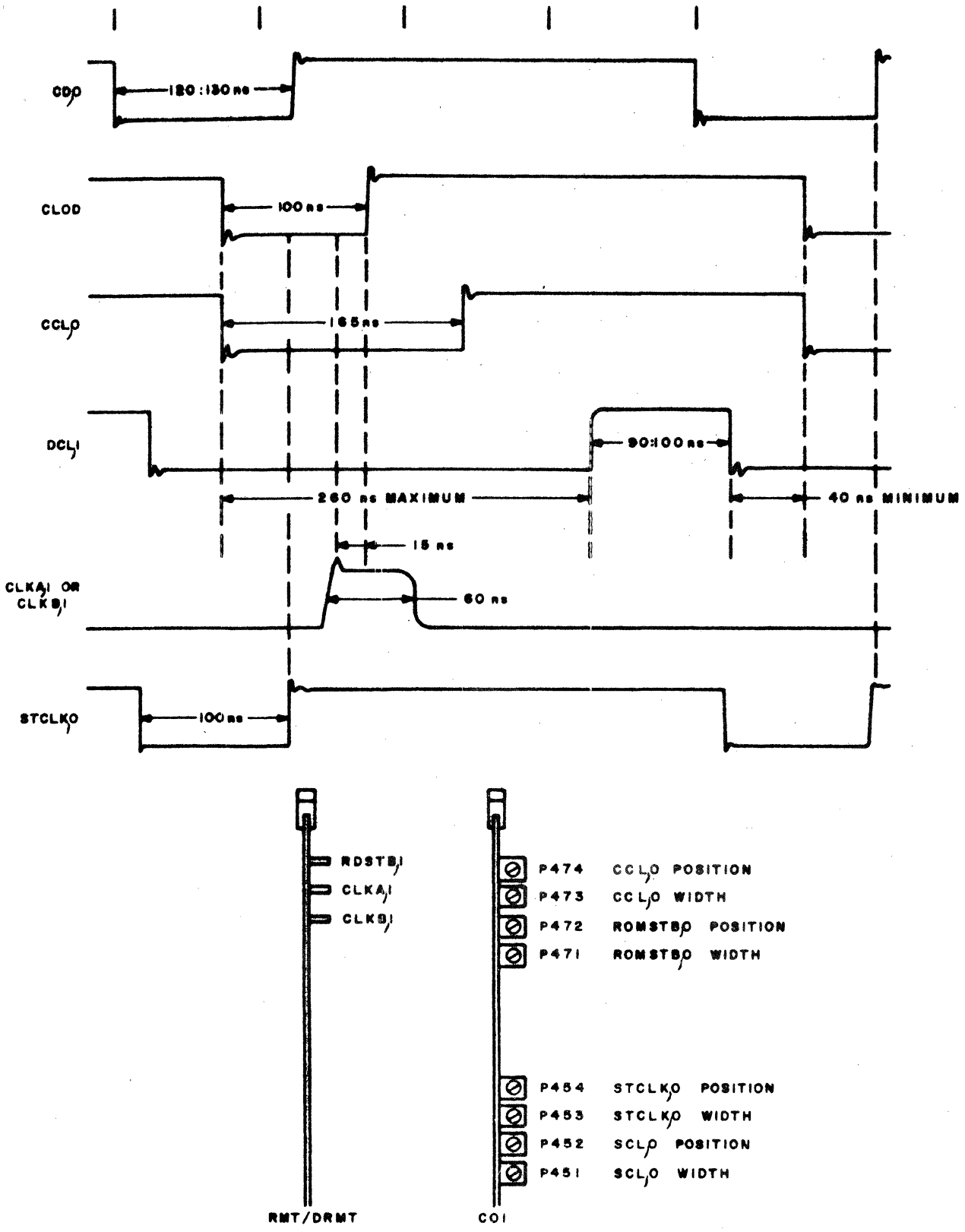


Figure 6. Processor Timing Data

TABLE 3. ROM OUTPUTS

OUTPUT	PIN	OUTPUT	PIN
CRD00,0	021-0901	SRD00,0	006-0901
CRD01,0	021-0901	SRD01,0	106-0901
CRD02,0	020-0901	SRD02,0	007-0901
CRD03,0	120-0901	SRD03,0	107-0901
CRD04,0	019-0901	SRD04,0	008-0901
CRD05,0	119-0901	SRD05,0	108-0901
CRD06,0	018-0901	SRD06,0	009-0901
CRD07,0	118-0901	SRD07,0	109-0901
CRD08,0	017-0901	SRD08,0	010-0901
SRD09,0	117-0901	SRD09,0	110-0901
CRD10,0	016-0901	SRD10,0	011-0901
CRD11,0	116-0901	SRD11,0	111-0901
CRD12,0	015-0901	SRD12,0	012-0901
CRD13,0	115-0901	SRD13,0	112-0901
CRD14,0	014-0901	SRD14,0	013-0901
CRD15,0	114-0901	SRD15,0	113-0901

9. Compare the SRD03,0 ROM output (pin 107-0901) and CL,0D (C01-L12). The rising edge of CL,0D should occur 20 ±5 nanoseconds before the falling edge of SRD03,0. If not, adjust CL0D position (Figure 6).

10. Check that CL0D is 100 ±16 nanoseconds wide. If necessary, adjust CL,0D pulse width (Figure 6).

11. Compare the CCL,0 output (C01-L11) with CL0D. The falling edge of CCL,0 should be coincident with the falling edge of CL0D. If not, adjust CCL,0 position (Figure 6).

12. Check that the rising edge of CCL,0 occurs 50 to 60 nanoseconds after the rising edge of CL,0D. If not, adjust CCL,0 pulse width (Figure 6).

13. Check the STCLK,0 output (C01-L14). STCLK,0 should be 100 ±10 nanoseconds wide and the rising edge should occur 40 to 50 nanoseconds before the rising edge of CL0D. If necessary adjust STCLK,0 pulse width and position.

14. Check that DCL,1 (C01-L07) is 90 to 100 nanoseconds wide and the falling edge of DCL,1 preceeds the falling edge of CL0D by 40 to 60 nanoseconds (Figure 6). If out of tolerance, check the associated logic.

15. Remove the grounds installed in steps 1, 2, and 4.

4.2 Overall Processor Test

Use the 30-2 Test Program described in Publication Number 06-036A12 to perform a comprehensive test of the Processor.

5. MNEMONICS

The following list provides a brief description of each mnemonic found in the Processor. The FS45 source of each signal is also provided.

MNEMONIC	MEANING	FS45 LOCATION
ACKA00	Interrupt Acknowledge	25R5
ACS0	Add or Subtract with Carry In	14E9
ACSL1	Add or Subtract and Carry in or L Control Line False	14F9
ADRS0	Device Address	25R6
AMOD1	Address Modification	16R9
AR001 through AR151	A Register	1F8
ARST1	Auto Restart	12D1
AUTOLD1	Auto Load	24B5
ATN0	Input/Output Attention	12C1
B000 through B150	B Bus	3J8
B130	B Bus bit 13 forced active on False Sync STAT	24R9
BANK1	Register BANK Flip-Flop	13G8
BRA0	Branch Micro-Op detected	15L8
C001 through C121	Carry Outputs	26P6
C010 through C140	Carry Outputs	26P6
CATN0	Console Attention, Console Execute	24P4

MNEMONIC	MEANING	FS45 LOCATION
CCL/0	Command Clock	23R4
CCMD/1	Clocked Command	13E3
CD/0	ROM Current Drive	23R6
CDVD/0	Command Divide	19E5
CEMT/1	Counter is Empty	7L9
CL07/0	I/O Control Line 7	21C9
CL/0D	Destination Clock	23R3
CL/0E	Externally Used Clock	23R4
CLKOFF/0	Clock Off Flip-Flop	23A3
CLR/0	Clear Option of Decode	14F9
CMD/0	Device Command	25R6
CMND/1	Command Micro-Op Decoded	15K8
CMPY/0	Command Multiply Mode	19L5
CLRCTR/0	Clear the Counter	7D9
CNRCL/1	Clear NORCL Flip-Flop	19K7
CRPT/0	Command Repeat	19F5
CS/1	Cross Shift	15E6
CSV/1	Carry to Save	26N3
CTLT2/1	Counter is Less Than Hexadecimal 2	19G2
CTONE/1	Counter is Hexadecimal ONE	19J1
CTSL/1	Carry True, Shift Left	14P9
CTSR/1	Carry True, Shift Right	14N9
CTST/1	Test Micro-Op Decoded	15J8

MNEMONIC	MEANING	FS45 LOCATION
CTR12,1 through CTR15,1	Counter Register	7K1
D,1	Decode Micro-Op	16B9
DA,0	Output Data Available	25R5
DAL00,0 through DAL07,0	Data Available Lines	3M8
DB0,1	Destination Decode Bit 0	10L9
DB1,1	Destination Decode Bit 1	10M9
DB2,0	Destination Decode Bit 2	10N9
DBD,1	Take Decoding Bits from Destination field of IR	8G9
DBS,1	Take Decoding Bits from Source field of IR	10K3
DCL,1	Delayed Clock	23R7
DECTR,0	Decrement the Counter	20R1
DEVN,1	Destination Address is Even	10R9
DR,0	Input Data Request	25R4
DR00,0 through DR07,0	Data Request Lines	3P3
DRD15,1	Decode Micro-Op with Phase Change	16F9
DST,0	I/O Device Clock Stop Flip-Flop	25K6
DTEST,1	Decode and Testable Function is True	12P8
DMST,0	Device or Memory Clock Stop	22R6
DVD,1	Divide Mode	20F5

MNEMONIC	MEANING	FS45 LOCATION
ENBC,0	Display Controller Enable	24B5
ENDEST,1	Enable Destinations	20R5
ENCLR,1	Enable ROM Address Clear	18D4
ENDROM,0	Enable Decoder ROM	18B9
ENSKP,1	Enable Clock Skip Flip-Flop	22G2
ENT,1	Enable General Registers	8F9
ESNO,0	Execute Switch Normally Open Contact	24C4
ESNC,0	Execute Switch Normally Closed Contact	24C3
EXTCLK,0	External Clock Input	23A2
FA0,1	Phase Pointer Flip-Flop 0	17E9
FA1,1	Phase Pointer Flip-Flop 1	17G9
FLR12,1 through FLR15,1	Flag Register	5K8
FNY0,1	Counter Dependent Mode Flip-Flop 0	19A9
FNY1,1	Counter Dependent Mode Flip-Flop 1	19C9
FRAL07,0	Force RAL07 Active ONE	20R7
FST,1	Fast I/O Interrupt Flip-Flop	25R2
FAST,1	Fast I/O Interrupt Input Line	12N2
FRCLOC,0	Force Location Counter to be Source and Destination	16J9
FRGP1,1	Force General Register Selection Plus 1 Flip-Flop	19N9
FSYN,1	False Sync Delay Output	25J2

MNEMONIC	MEANING	FS45 LOCATION
GA00/0 through GA15/0	Gated A Register Outputs	1E8
GA14/0	Gated A Register Bit 14 forced active	18N8
GCMND/1	Gated Command Micro-Op	13E2
GCTR/1	Gate into the Counter	7C2
GF12/01	Gate into FLR12	5D6
GF13/01	Gate into FLR13	5D4
GF14/51	Gate into FLR14 and 15	5D2
GFS/1	Gated False Sync	25J3
GIO/1	Gated Input/Output	24F8
GOBRA/1	Go Branch (enable)	15P8
GRD03/0	Gated RD Bit 3	8C9
GRD04/0	Gated RD Bit 4	20R3
GTP2/1	Go To Phase 2	17F6
HJ/0	Hardware Jam Enable	18D7
HLFCLK/1	Half Clock Maintenance Function	23A5
ILEG/1	Illegal Condition Detected	16C9
IR00/1 through IR15/1	Instruction Register	1M8
KILDST/0	Kill Destinations	20R6
L/1	L (Add or Subtract) ALU Control Line	15C8
LAR/0	Load the A Register	10F8

MNEMONIC	MEANING	FS45 LOCATION
LCO1,1 through LC14,1	Gated Carry Outputs	26N8
LCC,0	Load the Condition Code Register	14G8
LCTR,0	Load the Counter	10E9
LDF,0	Load the Flag Register	10G9
LDSTK,1	Load Pulse to the Register Stacks	11J5
LFLR,0	Load the Flag Register	10E9
LIO,0	Load to Input/Output	10D9
LIR,0	Load the Instruction Register	10E9
LMAR,0	Load Memory Address Register	10F9
LMDH,0	Load Memory Data Higher	10J9
LMDL,0	Load Memory Data Lower	10J9
LOAD,0	Load Micro-Op Decoded	14K8
LPSW,0	Load the Program Status Word	10H9
LRAH,0	Load ROM Address Higher	4R9
LRAL,0	Load ROM Address Lower	22R4
LYD,0	Load User's Destination Register	10D8
M1A	M Control Line to ALU	15D9
M1B	Alternate Generation of M1A	15E9
MBY,1A	Memory Busy	22A7
MDAV,0A	Memory Data Available	22A9
MPDV,1	Multiply or Divide	20E7
MPF,0	Memory Parity Fail	12J1

MNEMONIC	MEANING	FS45 LOCATION
MST/0	Memory Caused Clock Stop	22R8
MPY/1	Multiply Mode	20E2
MOP1/0	No Phase 1	17F3
NORCL/O	No ROM Clock	22R3
OSC/1	Oscillator Output	23D7
P0/1	Phase 0	17J8
P0RR/0	Phase 0, RR User's Format	16K9
P1	P Control Line to ALU	15J9
P1/0	Phase 1	17K8
P2/1	Phase 2	17M8
PFF/0	Clear the Parity Fail Flip-Flops	13P8
PPF/0	Primary Power Fail	21D9
POFF/0	Power Off (Switch Contact)	12M1
POW/0	Command Power Down	13R8
POWDN/0	Power Down	21F5
PSW01	Hardware Copy of PSW Bit 1	12G7
PTYS/1	Pointer for User's Source	17P8
RAWCLK/1	Raw Clock	23E2
RCL/O	ROM Clock	23R3
RJK/0	Reset JK Flip-Flop (hardwire)	24B1
RLOC/1	Read Out Location Counter	9N8
RMAR/1	Read Out Memory Address Register	9N8
RMR0/1 through RMR4/1	Read Out Micro-Registers	9N6

MNEMONIC	MEANING	FS45 LOCATION
ROMSTB/0	ROM Strobe	23R5
RPT/1	Repeat Mode	20E4
RSCTR/1	Reset the Counter	7B8
RPSW/1	Read Out PSW	9N8
RS/1	User's RS Format Instruction	16L8
RG00/1 through RG15/1	Read Out the General Registers	9N2
RUN/1	RUN Flip-Flop	16D4
S00/0 through S15/0	S Bus	27, 28
S003/1	S Bus (0:3) Equals Zero	1A8
S047/1	S Bus (4:7) Equals Zero	1A8
S125/1	S Bus (12:15) Equals Zero	1A8
S811/1	S Bus (8:11) Equals Zero	1A8
SB0/1	Source Decoder Bit 0	8K9
SB1/1	Source Decoder Bit 1	8M9
SB2/0	Source Decoder Bit 2	8N9
SCL/0	System Clock	23R1
SCLR/0	System Clear	21R8
SCLR/0A	System Clear to RAH and RAL	21R5
SCLR/0B	System Clear to RD	21R4
SCLR/0C	System Clear for General Use	21R5
SCLR/1	System Clear Relay Contact	21P9

MNEMONIC	MEANING	FS45 LOCATION
SEVN/1	Source Address is Even	8P9
SFL12/0	Set FLR12	4C8
SFL13/0	Set FLR13	4F8
SFL14/0	Set FLR14	4L8
SFL15/0	Set FLR15	4M8
SFNY0/1	Set Counter Command Store Flip-Flop 0	19A7
SFNY1/1	Set Counter Command Store Flip-Flop 1	19C7
SHL/1	Shift Left	14L8
SHR/1	Shift Right	14K8
SJK/0	Set JK (hard wire)	24B4
SKDMST/O	Clock Skip or Device or Memory Stop	22R4
SKPSTB/O	Skip ROM Strobe	16N8
SNGL/1	Console Single Mode	24P4
SOPC/1	I/O Control Line Output Flip-Flop	25J7
SPBRA/1	Special Branch	15M9
SR/0	I/O Status Request	26R5
SRAH06/0	Set RAH Bit 6	18L8
SRAL03/0	Set RAL Bit 3	18H8
SRAL04/0	Set RAL Bit 4	18J8
SRAL05/0	Set RAL Bit 5	18K8
SRLCT/0	Shift Right or Left with Carry True	14N5
STF/0	Set Test Flags	14C9
STFL/0	Set Test Flags on Add or Subtract	14E9

MNEMONIC	MEANING	FS45 LOCATION
STPSYS,1	Stop System Flip-Flop	21J3
STRT,0	Start Memory Cycle	13F8
SV,0	Set the Overflow Flag on False Sync	25R3
SVAC,1	Save Adder Carry	14B9
SVSC,0	Save Shift Carry	14H9
SWTAL,0	Set Wait Alarm	13L7
STCLK,0	Register Stack Clock	23R6
SYN,0	Device Sync	25A4
SYSCL,1	System Clock	23P1
TEST,1	Testable Function Sync Flip-Flop	12K8
TTEST,1	True Testable Function	12R5
U00,0	Destination Address is X'0'	11N6
U01,0	Destination Address is X'1'	11N6
U02,0	Destination Address is X'2'	11N6
U03,0	Destination Address is X'3'	11N6
UA,1	Unload the A Register	8D9
UD10,1	Unload I/O	24R7
UIO,0	Unload I/O	9N8
UIR,0	Unload the Instruction Register	9N8
UIR4,0	Unload IR's YD Field Only	9N8
UPSW,0	Unload Program Status Word	9N7
UT,1	Utility Flip-Flop	13J8
UMDR,0	Unload Memory Data Register	8A9

MNEMONIC	MEANING	FS45 LOCATION
UYD,0	Unload User's Destination	9N8
VP,0	Stot System Clocks	21R6
WA,0	Wait Alarm Flip-Flop	13L8
WMAR,1	Write Pulse to MAR	11N9
WMR0,1 through WMR4,1	Write Pulse to Micro-Registers	11N7
WLOC,1	Write Pulse to Location Counter	11N9
WPSW,1	Write Pulse to PSW	11N9
WR,0	Memory Write	13E8
WG00,1 through WG15,1	Write Pulse to G Register	11N3
X1A	Cross Shift	15G9
X1B	Cross Shift	15H9

THE DISPLAY SYSTEM

ILLUSTRATIONS

THE DISPLAY SYSTEM

1. INTRODUCTION

The Display System provides a means for reading the contents of all system registers and any core memory location, together with the capability of manually entering data and programs. Figure 1 shows the layout of the Display Panel. Two register displays are provided. The information displayed in these two registers is selected by the REGISTER DISPLAY and the MODE CONTROL Switches. Sixteen pushbutton switches provide a means for entering Memory Data and Memory Address into the machine. A SPEED CONTROL is provided to control the rate of instruction execution for use as a demonstration aid, or as a program debug aid. Other facilities provided are an EXECUTE pushbutton for requesting the operation selected on the switches to be executed; an INITIALIZE pushbutton for resetting the system and its peripherals; and a primary POWER pushbutton.

The Display is an Input/Output device and is interfaced over the Multiplexor Bus. The Display is controlled by a Display Controller board, and is supported by a special microprogram sequence which is wired into the Read-Only-Memory. The Display microprogram is only executed when a special interrupt (Console Attention - CATN) is generated at the Display Panel.

This Section describes the operation of a GE-PAC 30 Digital System using the Display Panel, describes the theory of operation of the Display Controller, and provides maintenance information for the Display System.

2. OPERATION

Refer to Figure 1 during the following description. The control console is comprised of six distinct elements:

1. Control Switches: POWER, INITIALIZE, and EXECUTE.
2. MODE CONTROL Rotary Switch.
3. SPEED CONTROL Rotary Switch.
4. REGISTER DISPLAY Rotary Switch.
5. Sixteen Data/Address switches.
6. Display of two 16-bit halfword registers.

Each of the elements is described in the following paragraphs. Console operating procedures are provided following the descriptions.

2.1 Control Switches

The latching POWER switch applies power to the Processor and device controllers. An indicator lamp is associated with the POWER switch.

The momentary INITIALIZE switch resets peripheral device interrupts and certain other functions in the Processor. After initialization, the Processor is left in the Halt mode.

The momentary EXECUTE switch causes the Processor to perform the function selected by the MODE CONTROL switch. The associated indicator lamp is on when the Processor is in the interruptable Wait state or Halt mode; the lamp is off when the Processor is in the Run mode.

2.2 MODE CONTROL Switch

The rotary MODE CONTROL switch selects the following modes of operation, which become effective when the EXECUTE switch is depressed:

RUN: the Processor continuously executes instructions at rated speed.

HALT (FIX): instruction execution is stopped at the moment the EXECUTE switch is depressed and the Processor is placed in the Wait state. The register displays are operative in this mode.

HALT FLP: 30-2 only. The HALT FLP position is similar to the HALT (or HALT FIX) position except that in 30-2 Processors equipped with the optional floating-point hardware, the selected registers are displayed in the floating-point format, on the top row of the Display Panel.

VARI (FIX): the Processor executes instructions at the rate selected by the variable SPEED CONTROL. The register displays are operative in this mode.

VARI FLP: 30-2 only. The VARI FLP position is similar to the VARI (or VARI FIX) position except that in 30-2 Processors equipped with the optional floating-point hardware, the selected registers are displayed in the floating-point format.

ADRS: selects the instruction location address portion of the Program Status Word (PSW(16:31)). The new address is entered in the sixteen Address Switches below the register display.

MEMR: the Memory Read mode permits display of memory data in the register display.

MEMW: the Memory Write mode permits entry of data into memory from the sixteen Data Switches below the register display.

2.3 SPEED CONTROL Switch

The variable SPEED CONTROL switch provides a dynamically changing display when in the Variable mode. The instruction execution rate can vary from 1 to 100 cps by rotating the control clockwise from SLOW to FAST. When in the SNGL position, a single instruction is executed and displayed each time the EXECUTE switch is depressed.

2.4 REGISTER DISPLAY Switch

The REGISTER DISPLAY switch selects pairs of 16-bit registers for display in the lighted panel positions labeled DISPLAY 1 and DISPLAY 2. Beginning at the one o'clock position and moving clockwise, the registers displayed are:

INST: (D1) The current instruction.
(D2) The Address field of the current instruction if RX or RS format.

PSW: (D1) The Program Status and Condition Code.
(D2) The location address of the current instruction.

R0/1: (D1) General Register 0.
(D2) General Register 1.

(Note: the seven succeeding pairs of General Registers are selected similarly.)

OFF: (D1) and (D2) are blank.

REGISTER DISPLAY: The position at 12 o'clock displays the second operand in (D1) and the result in (D2).

2.5 Data/Address Switches

The 16 Input Register latching pushbutton switches provide a means of entering information manually. An address set in the switches is entered into the instruction location address portion of the Program Status Word (PSW (16:31)) when the ADRS mode is selected and the EXECUTE switch is depressed.

Data set in the switches is written into memory when the MEMW mode is selected and the EXECUTE switch is depressed. The halfword location written into is specified by the instruction address portion of the PSW.

2.6 Register Display

The two 16-bit halfword register displays are operative when the VARIable Mode or when MEMR or MEMW have been selected. The display registers remain static when in the RUN mode.

The diagrams above the register display indicate the data fromat in (D1) and (D2) when the PSW, Instruction Register, a General Register pair, or MEMR/MEMW are selected for display.

2.7 Console Operating Procedures

To bring up power and initialize the system:

1. Depress the latching POWER switch.
2. Depress the momentary INITIALIZE switch.

To shut down power to the system:

1. Set the MODE CONTROL switch to HALT.
2. Depress the momentary EXECUTE switch.
3. Release the latching POWER switch.

To begin execution of a program:

The system must be in the Halt mode.

1. Set the MODE CONTROL switch to ADRS.
2. Enter the program starting address in the 16 address switches.
3. Depress the momentary EXECUTE switch.
4. Set the MODE CONTROL switch to RUN.
5. Depress the EXECUTE switch.

To halt execution of a program:

1. Set the MODE CONTROL switch to HALT.
2. Depress the EXECUTE switch.

To read memory from Display Registers:

The system must be in the Halt mode.

1. Set the MODE CONTROL switch to ADRS.
2. Enter the memory read starting address in the 16 address switches.
3. Depress the EXECUTE switch.
4. Set the MODE CONTROL switch to MEMR.

5. Depress the EXECUTE switch.

6. The memory data is read from Display Register 2 (D2). The memory address of the data being displayed is in Display Register 1 (D1).

7. Depress the EXECUTE switch to display memory data from successive memory locations. The memory address is automatically incremented each time the EXECUTE switch is depressed.

To write into memory:

The system must be in the Halt mode.

1. Set the MODE CONTROL switch to ADRS.

2. Enter the memory write starting address in the 16 address switches.

3. Depress the EXECUTE switch.

4. Set the MODE CONTROL switch to MEMW.

5. Enter the data to be written into memory in the 16 data switches.

6. Depress the EXECUTE switch.

7. The memory data entered is displayed in Display Register 2 (D2). The memory address which was written into is displayed in Display Register 1 (D1). To write into successive memory locations repeat from Step 5. The memory address is automatically incremented with each depression of the EXECUTE switch.

To display the Instruction Register, Program Status Word or General Registers:

The system must be in the Halt mode.

1. Set the REGISTER DISPLAY switch to select the registers desired for display.

2. Depress the EXECUTE switch. The registers selected for display will appear in D1 and D2 in the format indicated by the four diagrams above the lighted display.

To display registers in the VARIable speed mode:

The system must be in the Halt mode.

1. Set the MODE CONTROL switch to ADRS.

2. Enter the starting memory location address in the 16 address switches.

3. Depress the EXECUTE switch.

4. Set the MODE CONTROL switch to VARI.

5. Set the SPEED CONTROL switch to SNGL or to a SLOW-FAST setting.

6. Set the REGISTER DISPLAY switch to select the registers desired for display.

7. Depress the EXECUTE switch to begin operation of the program with display of the selected registers. If SNGL step was selected, the EXECUTE switch is depressed to cause single step execution of successive instructions.

8. The REGISTER DISPLAY switch setting can be changed during operation in the variable speed mode. The SPEED CONTROL switch can also be changed from SNGL to a SLOW-FAST setting without halting operations.

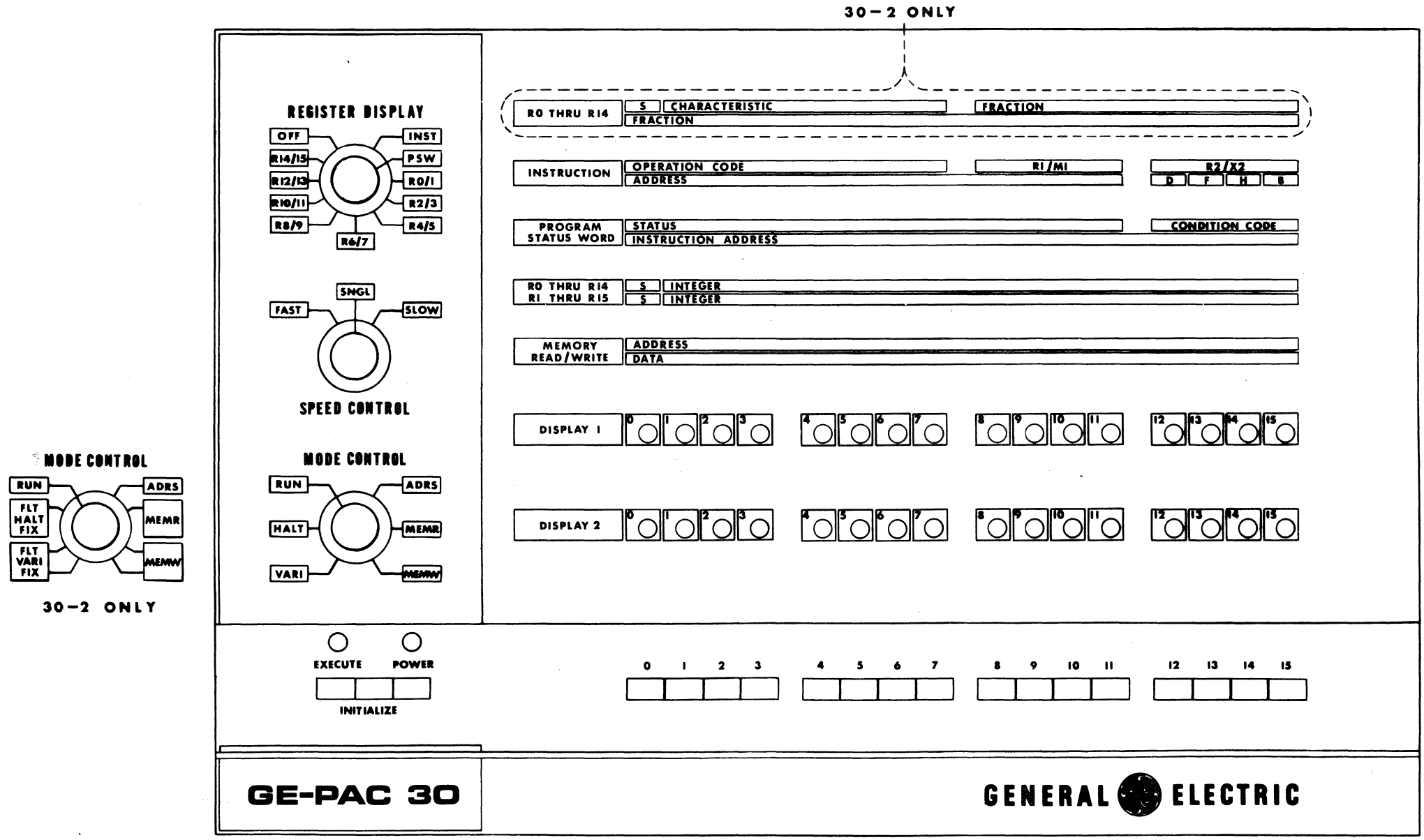


FIGURE I Typical GE-PAC 30 Computer

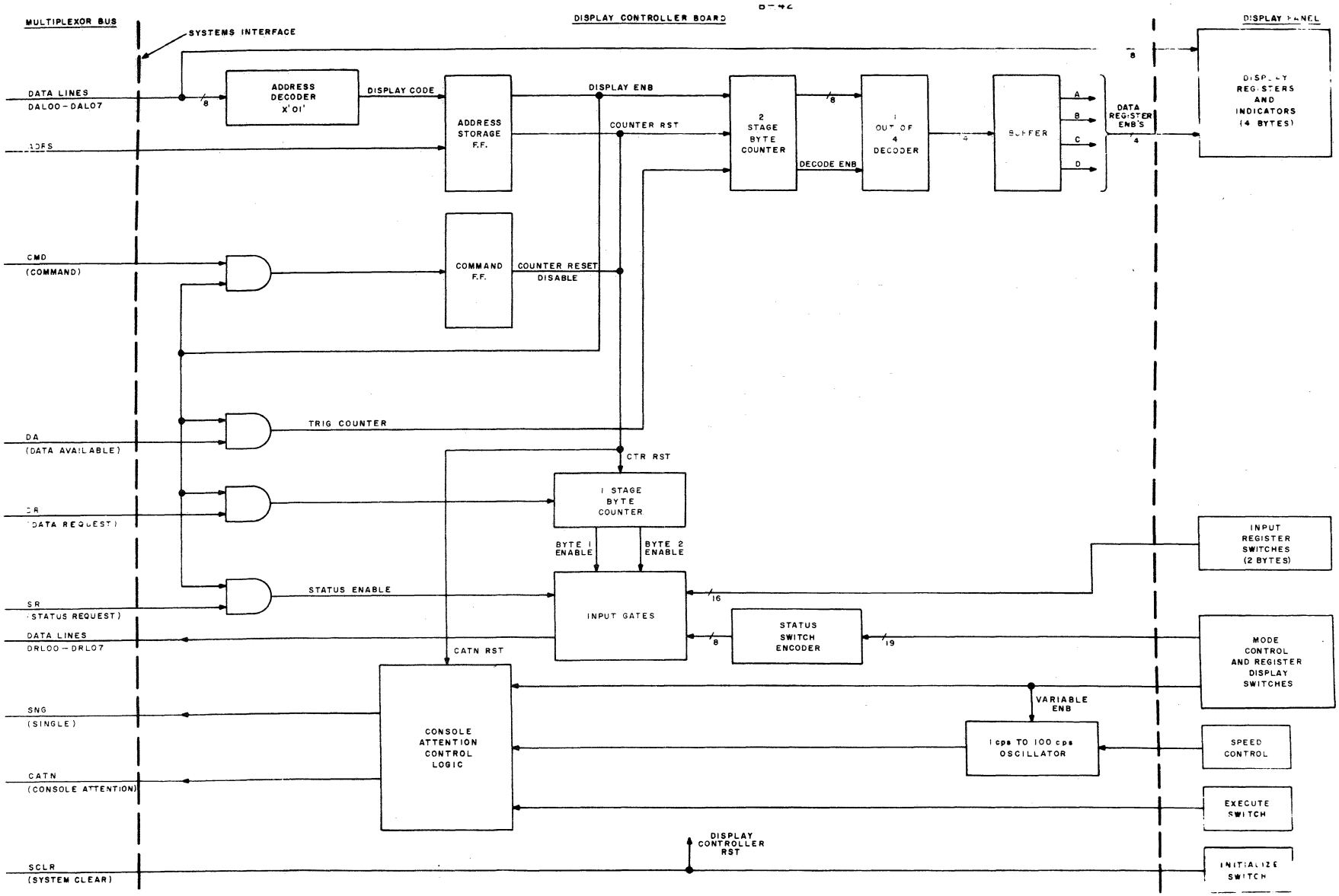


Figure 2. Display System Block Diagram

3. BLOCK DIAGRAM ANALYSIS

Refer to the block diagram on Figure 2. The Processor Multiplexor Bus is shown on the left, the Display Device Controller is shown in the center, and the Display Panel is shown on the right side of Figure 2.

The address logic is shown in the upper left area of the block diagram. On receiving a request for service via the EXECUTE switch and the CATN signal, the Processor outputs the 8-bit display address (X'01') on the Data Available Lines (DAL00:07). This is followed by an ADRS control pulse which causes the decoded address to be strobed into the Address Storage flip-flop. The set output from this flip-flop enables all other I/O commands into the Display Controller logic.

Data to the Display Panel is transmitted byte by byte, serially. The first byte is output on the DAL00:07 lines, and the Data Available command (DA) is raised. This command causes the first byte of data to be strobed into bits 8 through 15 of Register 1. The DA line is then dropped, and the 2-stage byte counter is incremented by one. The second byte of data is then placed of DAL00:07, and DA raised again. This process is repeated 4 times. Note that each time the display is addressed, the byte counter is automatically reset to zero in preparation for the first byte of data.

Data from the Input Register data switches is read into the system byte serially via the Data Lines (DRL00:07). On raising the Data Request (DR) command, bits 8 through 15 of the data switches are read into the system. The DR command is then dropped and the 1 stage byte counter is toggled, enabling the second byte of data. The DR command is raised again and bits 0 through 7 of the data switches are read. The 1 stage byte counter is automatically reset when the Display is addressed.

The Status of the mode and register switches is read into the system as a single byte of encoded data by raising the Status Request (SR) command. The 12 positions of the REGISTER DISPLAY switch are encoded onto bits 4 through 7; the MODE CONTROL switch positions are encoded onto bits 0 through 3 of the DRL Lines. The 32 display lamps and the 16 display switches may be I/O programmed for use as 32 indicators and 16 status switches. The main use of this feature is in I/O device test procedures, e.g. to display the binary result of an Analog-to-Digital Converter. The device is made programmable by executing an OC instruction. This causes the CMD line to set a flip-flop which disables the byte counters from being reset when the device is addressed.

The control logic is shown on the bottom of the block diagram. Depressing the EXECUTE switch sets a flip-flop in the Console Attention Control Logic. This generates a CATN signal to the Processor. On receiving the CATN signal, the Processor addresses the Display, which resets the flip-flop.

When the MODE CONTROL switch is in the VARI position, a variable astable multivibrator is enabled. The frequency of the oscillator may be varied from 1 H_Z to 100 H_Z by means of the SPEED CONTROL potentiometer on the Display Panel. The oscillator output sets the CATN flip-flop after depression of the EXECUTE button. Thus, automatic variable EXECUTES are achieved causing the execution of one instruction. The VARI position of the MODE CONTROL switch also causes a Single control line (SNG) to go high to signify this mode to the computer.

With the SPEED CONTROL potentiometer in the fully counter clockwise position, a switch is operated which disables the oscillator, and allows instructions to be executed in the manual step mode, each time the EXECUTE button is depressed.

The INITIALIZE Switch on the panel, when depressed, causes the resetting of the Processor, Display and all input/output devices.

4. FUNCTIONAL DESCRIPTIONS

This section provides technical descriptions of the functional groups outlined in earlier sections. The descriptions reference both simplified drawings included in this section, and the functional schematics provided in Volume 2 of this manual.

4.1 Addressing Logic

Refer to Figure 3 and FS31-1. The numbers associated with gates on Figure 3 are for reference in this description only, they have no other significance. Eight-input AND Gate 1 decodes address X'01' on the Data Available Lines (DAL00:07). The decoded address is strobed into the Address flip-flop Gates 2, 3, 4 and 5 by the ADRS command, causing the Display Enable line to set. The SR, DA, DR, or CMD commands are now enabled by Gates 11, 12 and 13. The ADSY,0 signal resets the display logic during the ADRS,I Strobe.

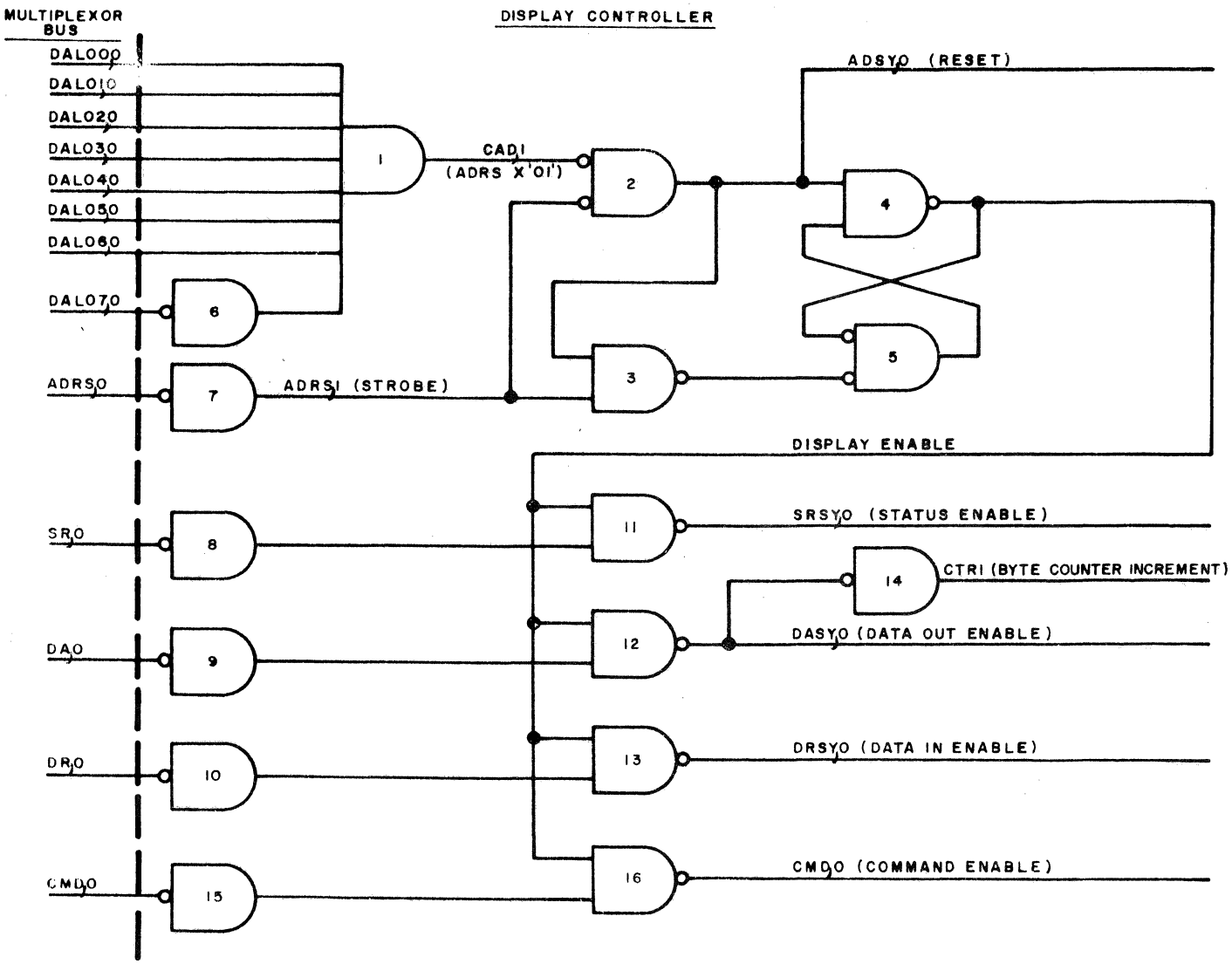


Figure 3. Display Addressing, Logic Diagram

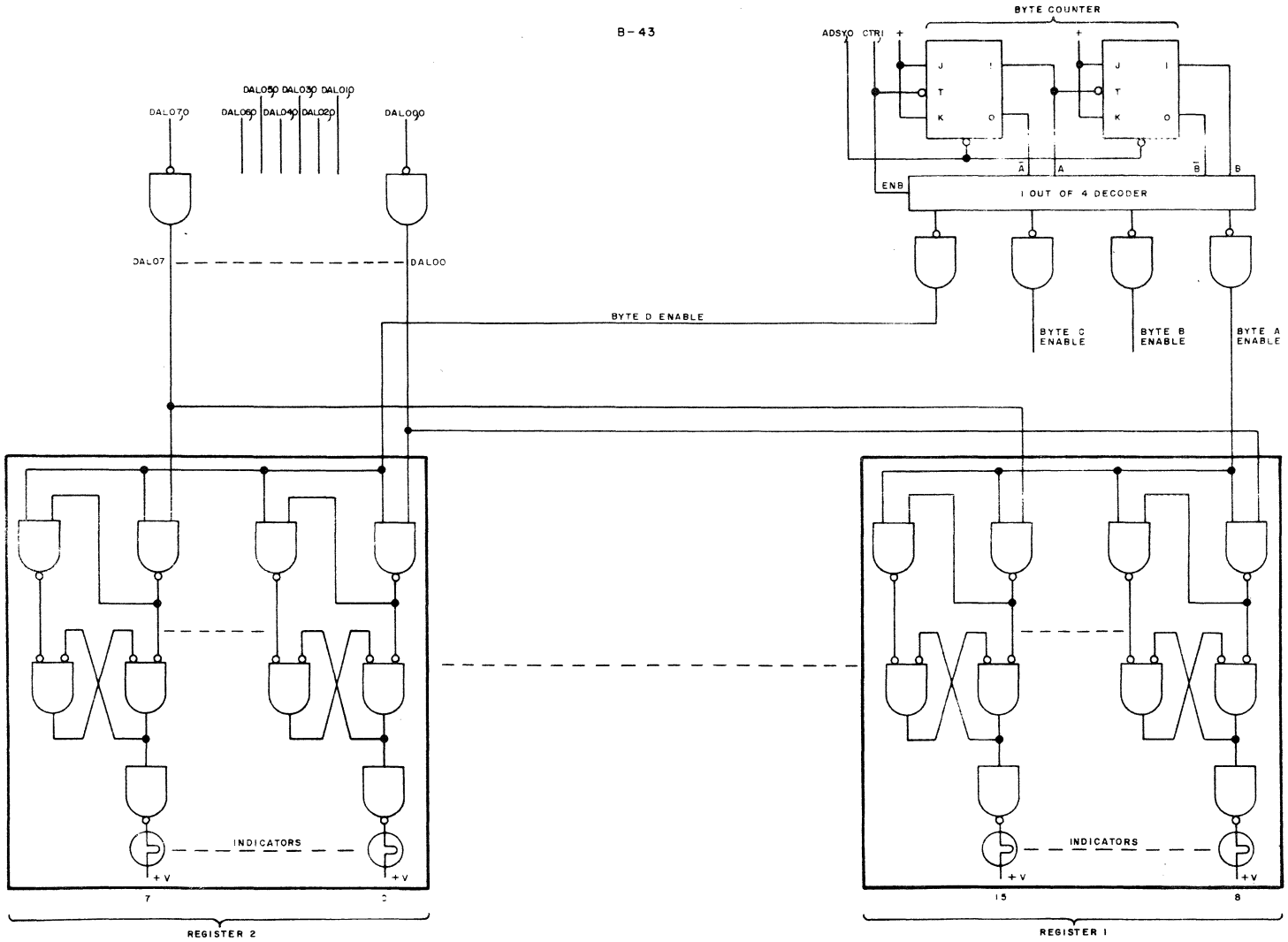


Figure 4. Data Output Logic

4.2 Data Output

Refer to Figures 4 and 5, and to Functional Schematic FS31-2. The ADYS,0 line, which goes high during the Display ADRS command, resets the Byte Counter to X'0'. When the Data Out Enable line (DASY,0) goes high, Byte A Enable strobes the data on DAL00:07 to bits 8 through 15 of Register 1 on the Display Panel. The data is stored in gate flip-flops, and displayed on the appropriate indicators. The Data Out Enable (DASY,0) is then lowered, and CTR,1 increments the Byte Counter. On raising (DASY,0) again, the decoder causes Byte B Enable to strobe the data on DAL00:07 to bits 0 through 7 of Register 1. This process is repeated until both registers are updated with the appropriate data. Figure 5 shows the timing diagram for the Display. The Display Registers are shown on FS31-7 and FS73-8.

4.3 Data Input

Refer to Figures 6 and 7, and to Functional Schematics FS31-4, FS31-5, and FS31-6. The SCLR,0 line resets the Data Switch flip-flop as for the Data Output. (The Data Switch flip-flop is shown in the lower left corner of FS31-2.) When the Data Request (DRG,1) line goes high, the least significant byte enable (LOW,1) goes high and reads in the condition of bits 8 through 15 of the Data/Address Switches. (The switches are shown on the Bottom of FS31-6.) When the DRG,1 line goes low, the Data Switch flip-flop is toggled. As the DRG,1 line goes high again, the most significant byte enable (HIGH,1) goes high to read in the condition of bits 0 through 7 of the Data/Address Switches. Figure 7 is a timing diagram for data input.

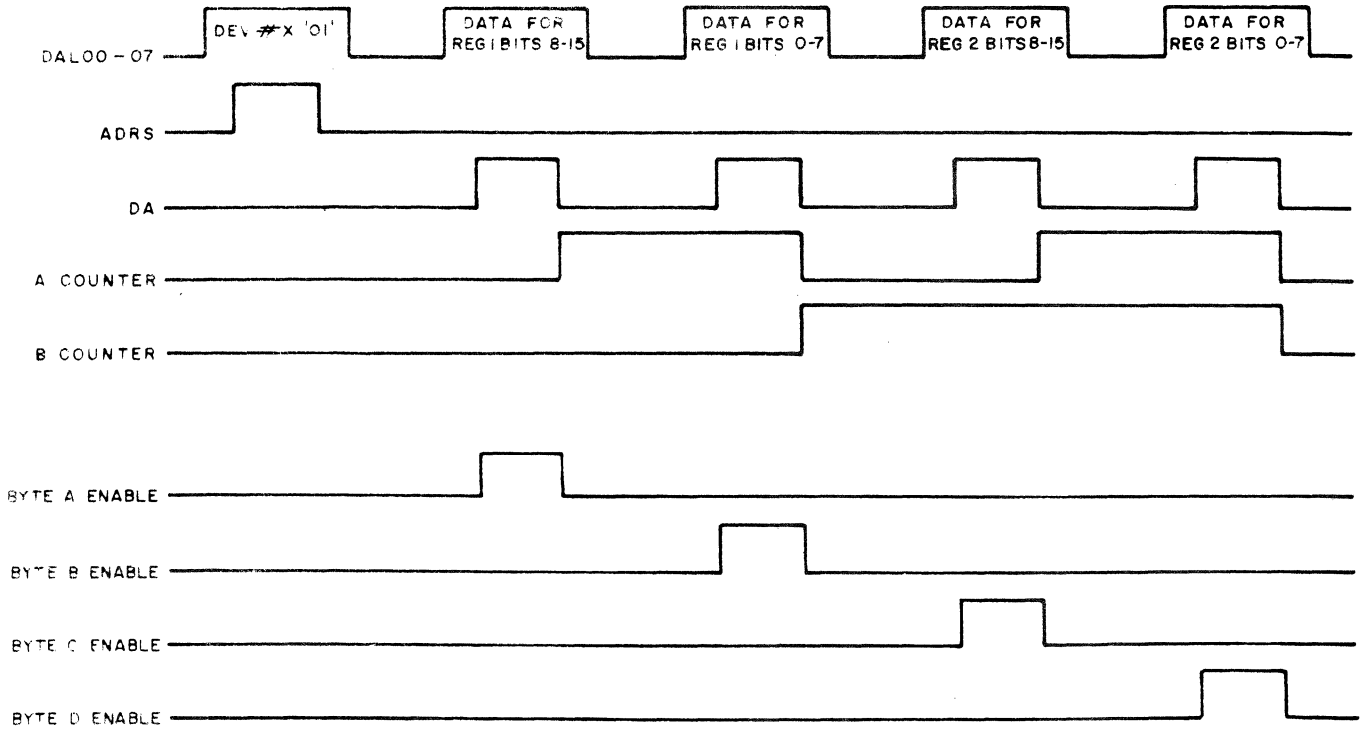


Figure 5. Data Output Timing

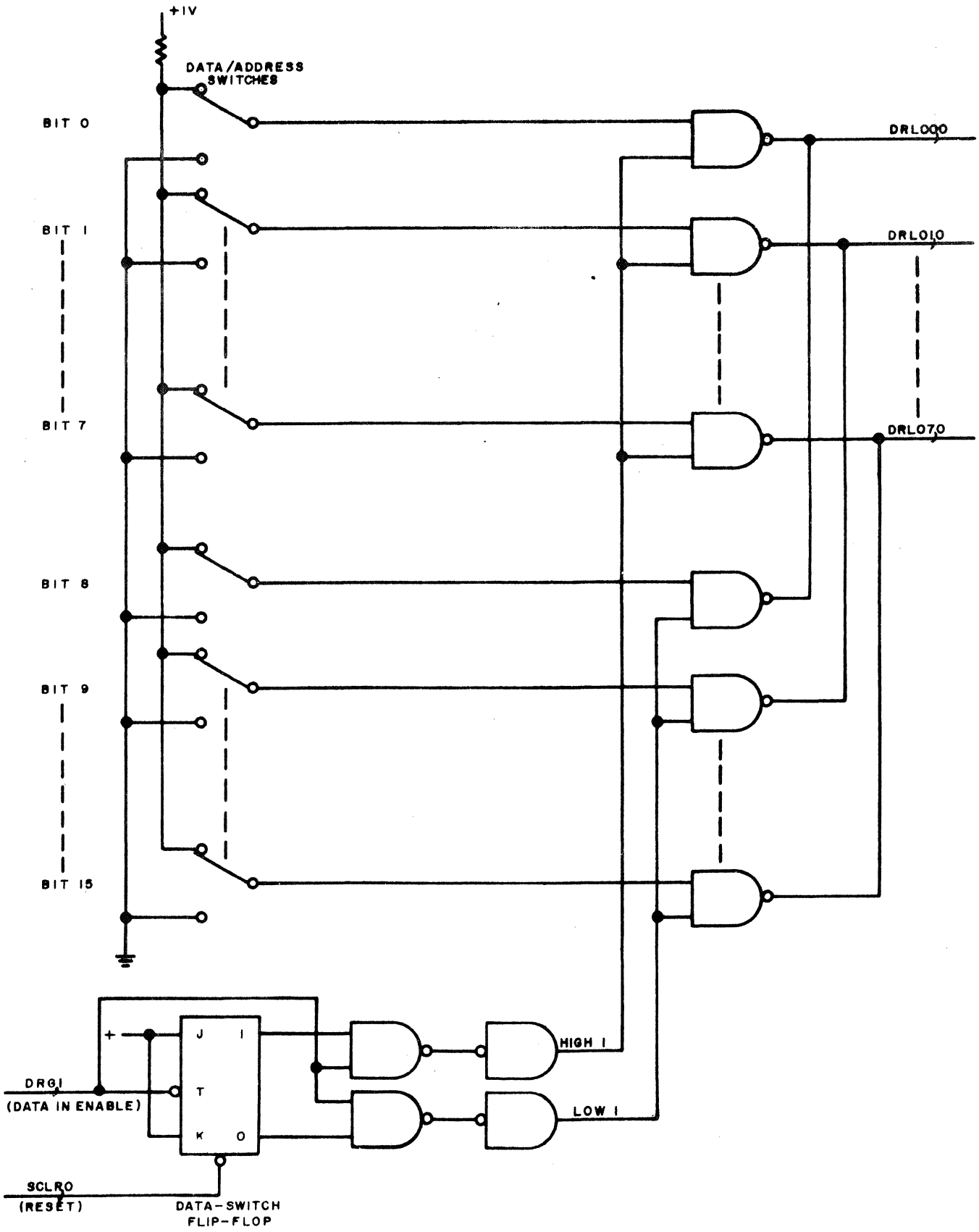


Figure 6. Input Register Switches

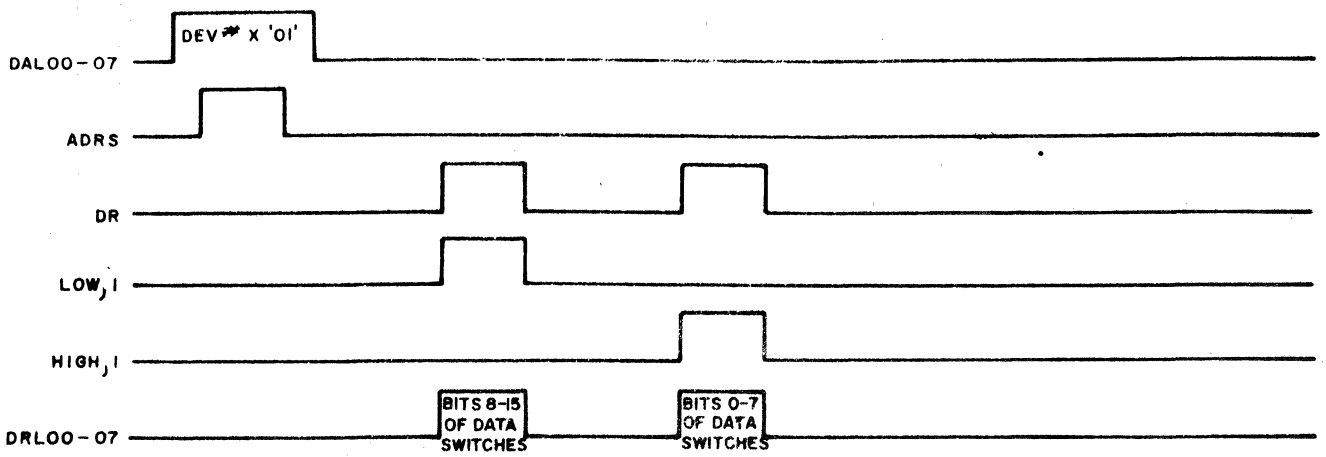


Figure 7. Data Input Timing

TABLE 1. DISPLAY STATUS BYTE CODING

STATUS BYTE	0	1	2	3	4	5	6	7
	MODE				REG			

Mode Control Switch

VARI	0	1	0	0					
HALT	1	1	0	0					
RUN	1	0	0	0					
ADRS	0	0	1	1					
MEMR	0	0	1	0					
MEMW	0	0	0	1					
HTFT	1	1	1	0					
VRFT	0	1	1	0					

Register Select Switch

				0	0	0	0	OFF
				0	0	0	1	Register Display
				0	0	1	0	INST
				0	1	0	0	PSW
				1	0	0	0	R0/1
				1	0	0	1	R2/3
				1	0	1	0	R4/5
				1	0	1	1	R6/7
				1	1	0	0	R8/9
				1	1	0	1	R10/11
				1	1	1	0	R12/13
				1	1	1	1	R14/15

4.4 Status Input

Refer to Figures 8 and 9, and to Functional Schematics FS31-4, FS31-5, and FS31-6. The numbers within gates on Figure 8 are for this explanation only. They have no other significance. The REGISTER DISPLAY Switch is encoded onto the least significant 4 bits of the status byte by OR Gates 1 through 4. The MODE CONTROL Switch is encoded onto the most significant 4 bits of the status byte by OR Gates 5 through 8. The resulting status byte is gated onto the DRL00:07 lines by the SRSY$\overline{0}$ (Status Enable) line. Table 1 lists the status codes for the switch positions. Figure 9 shows the status timing.

4.5 Control Logic

Refer to FS22-29 if a 30-1 Processor or FS45-28 if a 30-2 Processor, FS31-3, FS31-6 and Figure 10. The arbitrary reference designations used on Figure 10 are referenced in the following description. The EXECUTE switch outputs are applied to Flip-Flop 1 which is normally set. When the EXECUTE Switch is depressed, Flip-Flop 1 is cleared. The rising edge of the flip-flop ZERO side output triggers the monostable circuit formed by Gates 3, 4, and 5. C1 and R1 are selected to give approximately a 1 microsecond negative going pulse on the output from Gate 5. This pulse sets the flip-flop formed by Gates 19 and 20, which in turn generates a Console Attention (CATN) to the Processor. This flip-flop is reset by ADSY$\overline{0}$ when the Processor addresses the device.

When the MODE CONTROL switch is set to the VARI position, the VAR$\overline{0}$ line from the switch to Gate 15 goes low. After the EXECUTE button is depressed, this condition is strobed into the flip-flop formed by Gates 17 and 18 by the 1 microsecond output pulse from Gate 1. This, in turn, generates a single (SNG$\overline{0}$) level to the Processor to request display support service. The output from Gate 18 also enables the astable multivibrator, (made up of Gates 6, 7, 8, 9, 10 and 11) to oscillate.

The SPEED CONTROL Potentiometer on the Display panel, determines the frequency of oscillation. In the counter clockwise (CCW) position, a 100K ohm resistance is placed in series with R4, causing the oscillator to run at approximately 1 H_z. In the clockwise (CW) position the oscillator runs at approximately 100 H_z. The output from Gate 8 produces approximately a 5 millisecond positive-going pulse (determined by R3, C3) which triggers the monostable formed by Gates 12, 13 and 14.

5. PHYSICAL DESCRIPTION

Refer to Figure 11. The Display Controller logic is contained on one 35-109 mother-board. The mother-board normally plugs into the first I/O position of the Multiplexor Bus. If the system does not contain the Display Option, the position may be used for another device controller.

The logic on the Display Controller is implemented using DTL logic packs except for the monostable and astable multivibrator timing networks which are mounted on a #2 daughter-board 35-027.

The Display switches and Display Registers and Indicators are mounted on the Display Panel. The Display Registers and Indicators are mounted on a printed board type 35-034 which mounts behind the Display Panel. The board contains 32 register stages and indicators.

The Display Panel is plugged into the Display Controller via a cable harness containing four cable cards. These cables plug into locations 41 through 44. Another cable card associated with the EXECUTE, INITIALIZE and POWER switch assembly, plugs into position 40 of the Control #2 board in a 30-1 or the Control #3 board in a 30-2.

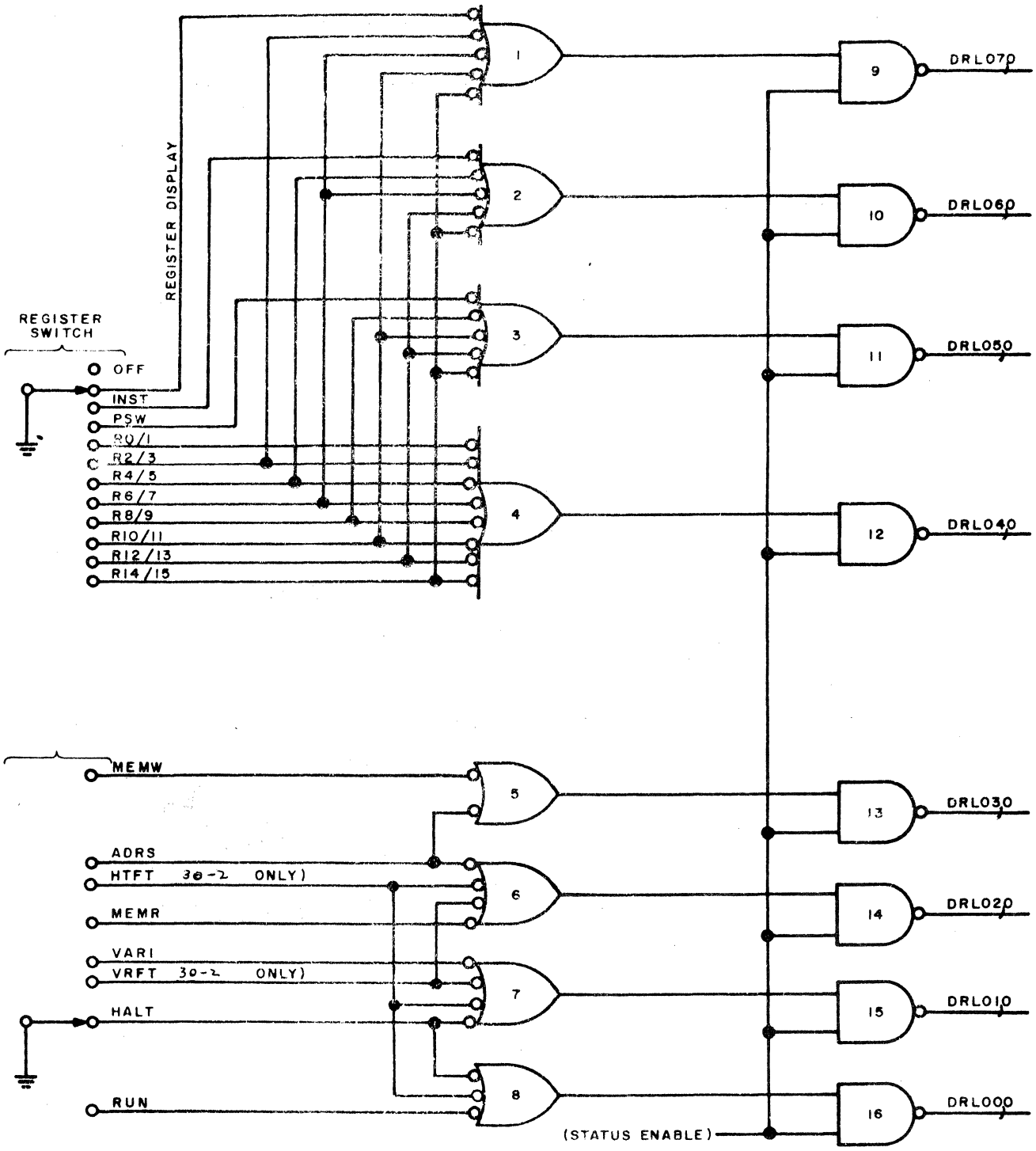


Figure 8. Status Switch Encoding and Input

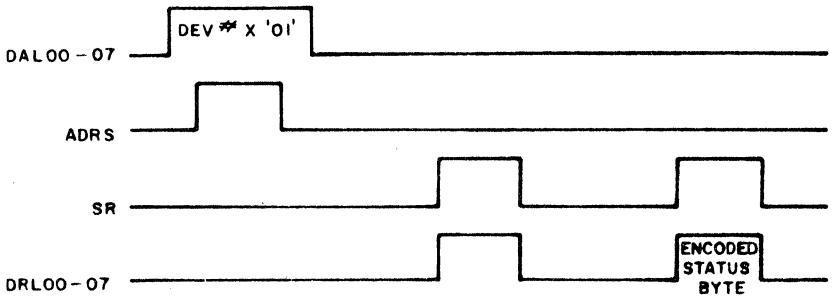


Figure 9. Status Timing

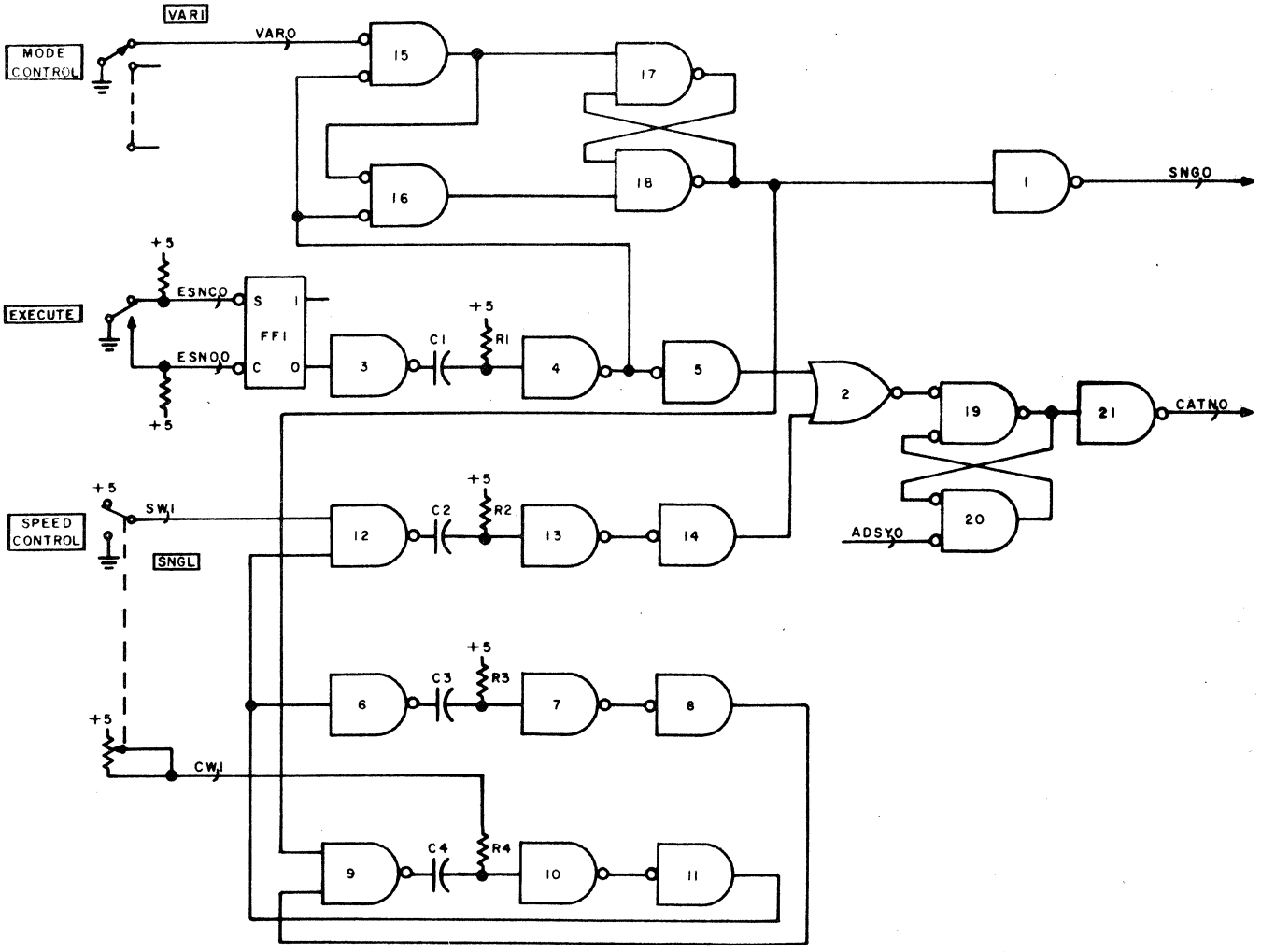


Figure 10. Control Logic

6. DISPLAY SYSTEM MNEMONICS

The following list provides a brief description of each mnemonic in the Display System. The FS31 source of each signal is also provided.

MNEMONIC	MEANING	FS31 LOCATION
ADRS,0	Address command from the Processor	1A7
ADRS,1	Output from Address flip-flop	1N2
ADSY,0	Address Sync signal to set preferred states in the controller	1P4
AENB,1	Enable DAL (00:07) to Display Register 1 bits 8:15	2K8
BENB,1	Enable DAL (00:07) to Display Register 1 bits 0:7	2L8
CATN,0	Console Attention signal to the Processor	3N2
CENB,1	Enable DAL (00:07) to Display Register 2 bits 8:15	2M8
CMD,0	Command signal from the Processor	1A9
CMDG,1	Command gated signal	1P9
CTR,1	Byte counter increment signal	1L6
CTRST,1	Byte counter reset signal	1P5
CW,1	Common output from SPEED CONTROL	6K4
DA,0	Data Available signal from the Processor	1A8
DAG,1	Data Available gated signal	1P7
DAL00,0 through DAL07,0	Data Available Lines from the Processor	1A1
DENB,1	Enable DAL (00:07) to Display Register 2 bits 0:7	2P8
DR,0	Data Request signal from the Processor	1A7

B-38-I

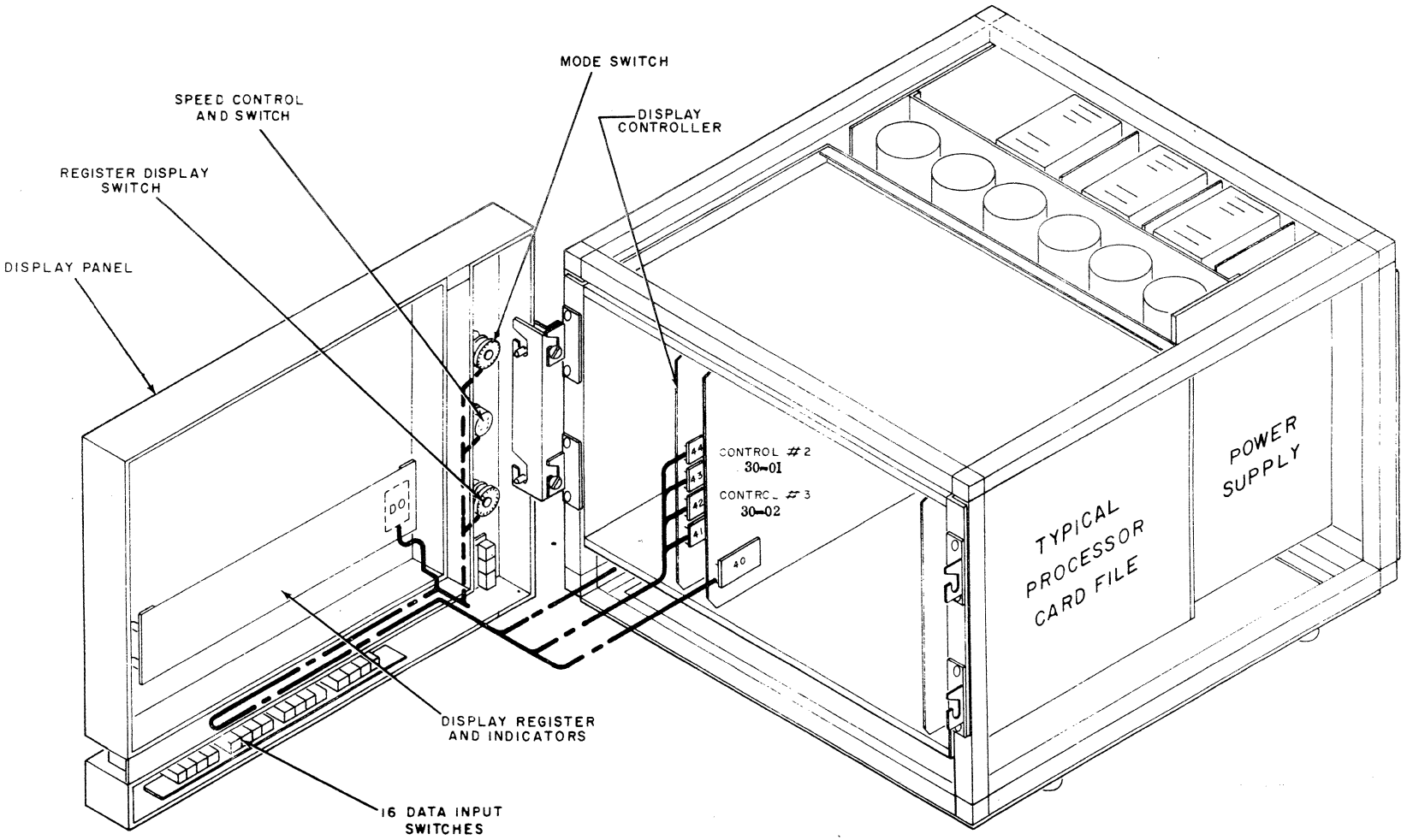


Figure 11. Display System Physical Layout

MNEMONIC	MEANING	FS31 LOCATION
DRG, 1	Data Request gated signal	1P7
DRL00, 0 through DRL07, 0	Data Request Lines from the Processor	4D9
DS00, 1 through DS15, 1	Data/Address switch outputs	6B6
ESNC, 0	EXECUTE switch normally closed signal from the Processor	3A2
ESN0, 0	EXECUTE switch normally open signal from the Processor	3A3
HIGH, 1	Signal which enables Data/Address switches 0:7 to DRL 0:7	1H8
HLT, 0	HLT signal from MODE CONTROL switch	6C4
HTFT, 0	Halt - Floating Point signal from 30-2 MODE CONTROL switch	6C4
LOW, 1	Signal which enables Data/Address switches 8:15 to DRL 0:7	1H7
MR, 0	MR signal from MODE CONTROL switch	6E4
MW, 0	MW signal from MODE CONTROL switch	6E4
NSTR, 0	Output from REGISTER DISPLAY switch when it is in the INST position	6M4
P5	Positive 5 volt supply voltage	
PSW, 0	PSW signal from REGISTER DISPLAY switch	6M4
R00, 0 through R14, 0	R0/1 through R14/15 signals from REGISTER DISPLAY switch	6M4
RDGDSP, 0	Output from REGISTER DISPLAY switch when it is in the unlabeled 12 o'clock position	6L4
RUN, 0	RUN signal from MODE CONTROL switch	6C4

MNEMONIC	MEANING	FS31 LOCATION
SADRS,0	ADRS signal from MODE CONTROL switch	6D4
SCLR,0	System Clear signal from the Processor	2A1
SNGL,0	Single step signal to the Processor	3M6
SR,0	Status Request signal from the Processor	1A8
SRG,1	Status Request gated signal	1P8
SW,1	Switch output from SPEED CONTROL	6G4
SYN,0	Synchronized pulse to the Processor	1P6
VAR,0	VAR signal from MODE CONTROL switch	6B4
VARCLK,1	Variable clock signal	3N8
VIND	Positive 5 volt supply to the Display indicators	7A9
VRFT,0	Variable - Floating Point signal from 30-2 MODE CONTROL switch	6B4

THE MEMORY SYSTEM

CONTENTS

ILLUSTRATIONS

THE MEMORY SYSTEM

1. INTRODUCTION

This Section describes the GE-PAC 30 2 1/2D Core Memory System. A brief review of core memory theory precedes the detailed system description. The descriptions referenced both simplified drawings provided in this Section, and the functional schematics provided in Volume 2 of this manual. Maintenance data and a mnemonic list are included later in the Section.

2. BASIC CORE MEMORY THEORY

Until recently, most core memories were either coincident current (3D) or linear select (2D) configurations. Figure 1 illustrates the layout of a typical 3D core memory stack. Note that each core has 4 wires passing through it, and that the X and Y drive lines pass through n cores in each bit plane. A core is switched by applying 1/2 the current necessary to switch the core (1/2 H) to the appropriate X and Y drive lines. One core in each bit plane is thereby addressed. During a read operation, current flow in the drive lines is such as to force the cores to the 0 state. Nothing occurs in any core which is already in the 0 state. If a core is in the 1 state, however, the core switches. When the core changes state, a signal is induced into the sense winding for that bit plane. The induced signal is used to generate a 1 indication for that bit position. During a write operation, current flow is in the opposite direction in the drive lines, and tends to force the cores to the 1 state. An opposing current is applied to the inhibit winding for each bit plane which is to remain at 0.

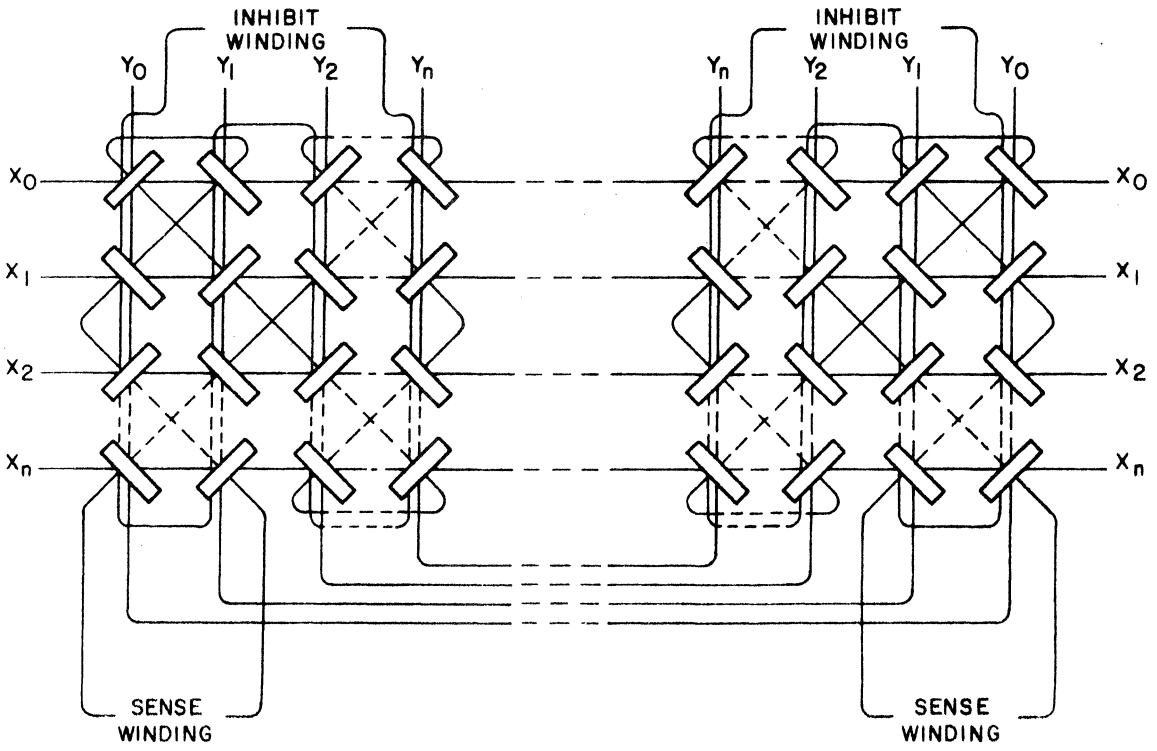


Figure 1. Typical 3D Core Memory Configuration

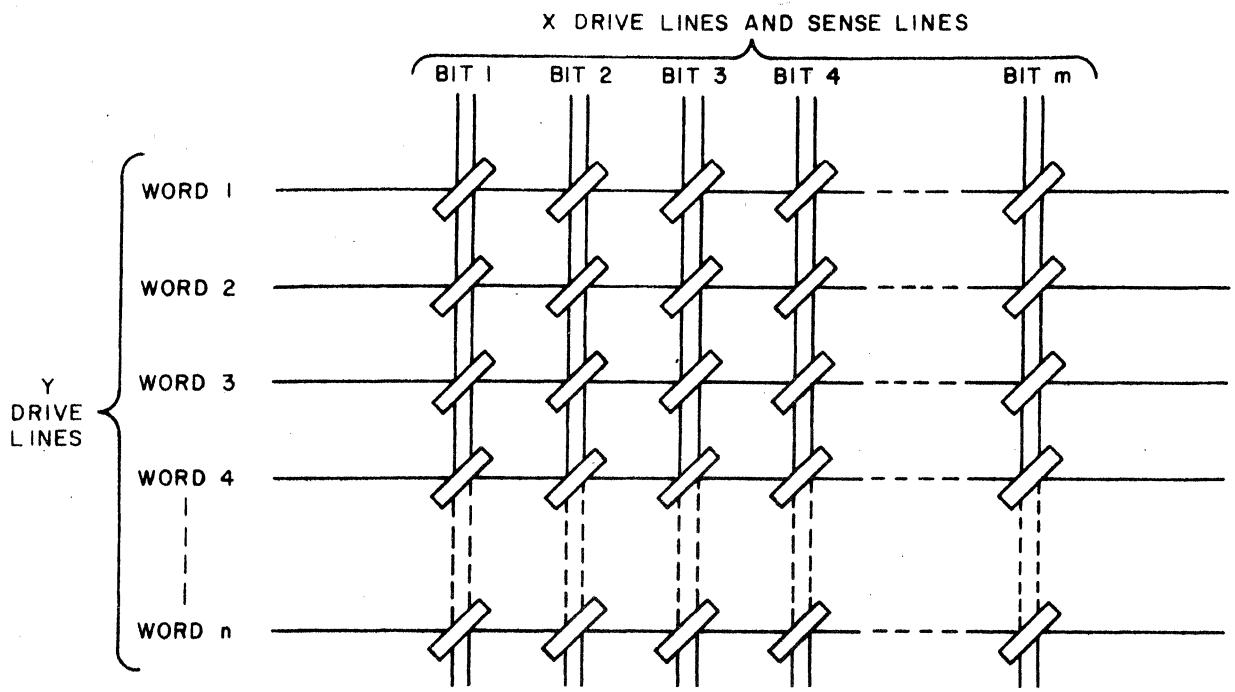


Figure 2. Typical 2D Core Memory Configuration

Figure 2 shows a typical 2D core memory stack configuration. Each memory word has an associated Y drive line in the 2D configuration. During a read operation, the Y drive line provides current sufficient to switch all cores in the addressed word to the 0 state. As the Y drive line only passes through the cores in the addressed word in a 2D system, drive currents greater than full drive may be used (typically 1.5 H). Cores which were in the 1 state prior to the read operation induce a voltage in the associated sense line as described for the 3D configuration. For a write operation, a current coincidence between the X and Y drive lines is used. No current is applied to the X drive lines for bits which are to be left in the 0 state.

A comparison of the 3D and 2D configuration shows that:

1. The 2D configuration is much faster due to the shorter drive lines and the fact that larger read currents may be used.
2. Although the 2D stack (with at least 1, and sometimes 2, fewer wires through each core) is much cheaper, the additional electronics required by the separate word drive lines normally makes the total memory system cost higher than the 3D.
3. The 3D configuration is therefore normally slower, but the system cost is lower.

The 2 1/2D configuration is a compromise between the 2D and 3D configurations. The 2 1/2D memory uses a 3 wire core arrangement similar to the 2D configuration, but a level of decoding is provided within the stack, similar to the 3D configuration. Figure 3 is a simplified drawing of a basic 2 1/2D configuration. Note that the X lines go through all bits as in the 3D layout. There is a separate set of Y lines for

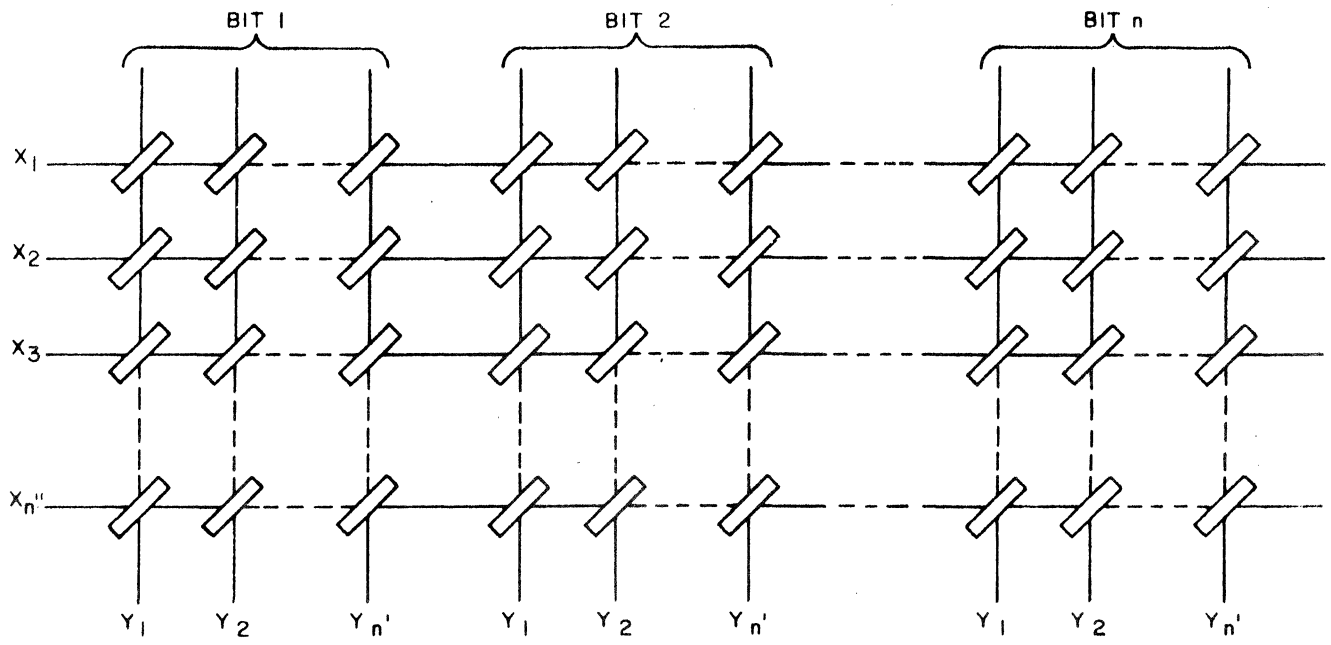


Figure 3. Basic 2 1/2D Configuration, Simplified Diagram

each bit, however. The sense lines for each bit (not shown on Figure 3) are arranged similar to the 3D configuration. For a read operation, the appropriate X drive line, and the appropriate Y drive line for each bit (n Y drive lines for the example in Figure 3) drive current in the direction which forces the cores to the 0 state. Cores which were in the 1 state induce a signal into the sense winding as described previously. For a write operation, half currents are applied to the X drive line and to all Y drive lines associated with positions which are to be set to the 1 state. Thus, cores which are to remain in the 0 state receive only a single half current (on the X drive line).

Figure 4 illustrates a wiring method which is commonly used to cut down on the number of drive circuits required. One set of drive lines (the Y drive lines in the example) is folded and run back through another set of cores in the same bit position. The number of Y drive lines is thereby halved. Only one of the two cores at the intersection of each pair of X and Y drive lines is selected at any time. The selection depends on which way current is driven through the two lines. For example, assume current flow in the X2 and Y1 lines in the direction shown by the arrows on Figure 4. The bit 1 core which is shaded solid is selected in this case. If the current in the X2 drive line is reversed, the dashed shaded core is selected. Note, however, that if the first case writes a 1 into the core, the latter case reads, or forces the addressed core to the 0 state. The four conditions of current flow direction correspond to read and write operations in each of the two cores.

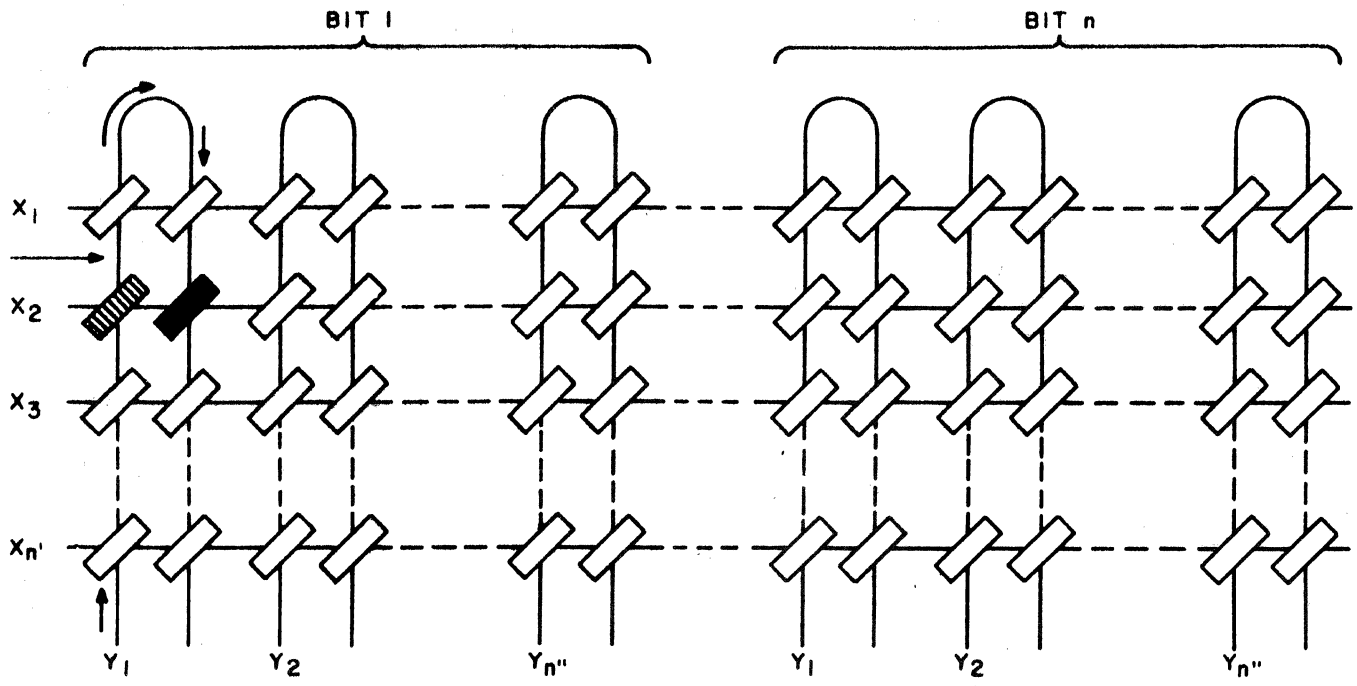


Figure 4 . Basic 2 1/2D Configuration, Simplified Diagram Showing Folded Y Drive Lines

3. PHYSICAL DESCRIPTION

The GE-PAC 30 2 1/2D Memory System consists of up to eight memory blocks. Each memory block provides storage for either 4,096 or 8,192 bytes. Note that while the Model 3 addresses 8-bit bytes, the memory system operates on 16-bit halfwords. Thus, the 8KB block actually consists of 4K halfwords. If a block contains the optional parity feature, two additional bits are provided, for a total of 18 bits per memory location.

The memory system consists of up to six types of mother-boards. Each 4KB or 8KB block has a dual mother-board unit consisting of an ME0 mother-board and an ME1 mother-board. The ME0-ME1 combination contains the cores, the decoding matrix, the bit switches, and the sense amplifiers for the particular 4KB or 8KB provided on that memory block. A third type of mother-board is the Memory Switch (MSW) board. An MSW board provides the X switches for up to 8KB of memory. The Memory Current Drive (MCD) mother-board provides current drive to the bit lines of up to 8KB of memory. A single Memory Interface (MSF) mother-board provides the timing. Memory Address Register (MAR), and Memory Data Register (MDR) for an entire memory system (up to 64KB). The sixth type of board is the optional Memory Parity and/or Memory Protect mother-board (MPP). The parity section provides logic and registers for 2 byte parity bits. The protect section converts Write orders to Read orders in protected areas of memory. A protection system involves additional controls and program and will not be treated in this chapter.

4. BLOCK DIAGRAM ANALYSIS

Figure 5 is a simplified block diagram of the memory system. The dashed lines which enclose functions on Figure 5 indicate the mother-board which provides that function. The Memory Address Register (MAR) is shown in the upper left area of Figure 5 The MAR holds the 16-bit byte address that the program is currently concerned with. As explained previously, the memory system operates on halfwords, so the 16th MAR bit (MA15) is not decoded in the memory system. Memory Addressing is described in detail later in this Chapter. Basically, however, bits MA0 through MA2 select a memory block, MA3 through MA10 select an X line, and MA12 through MA14 select a bit line. Bit MA11 selects the direction of word current as described in connection with Figure 4. The MAR is loaded from the S Bus.

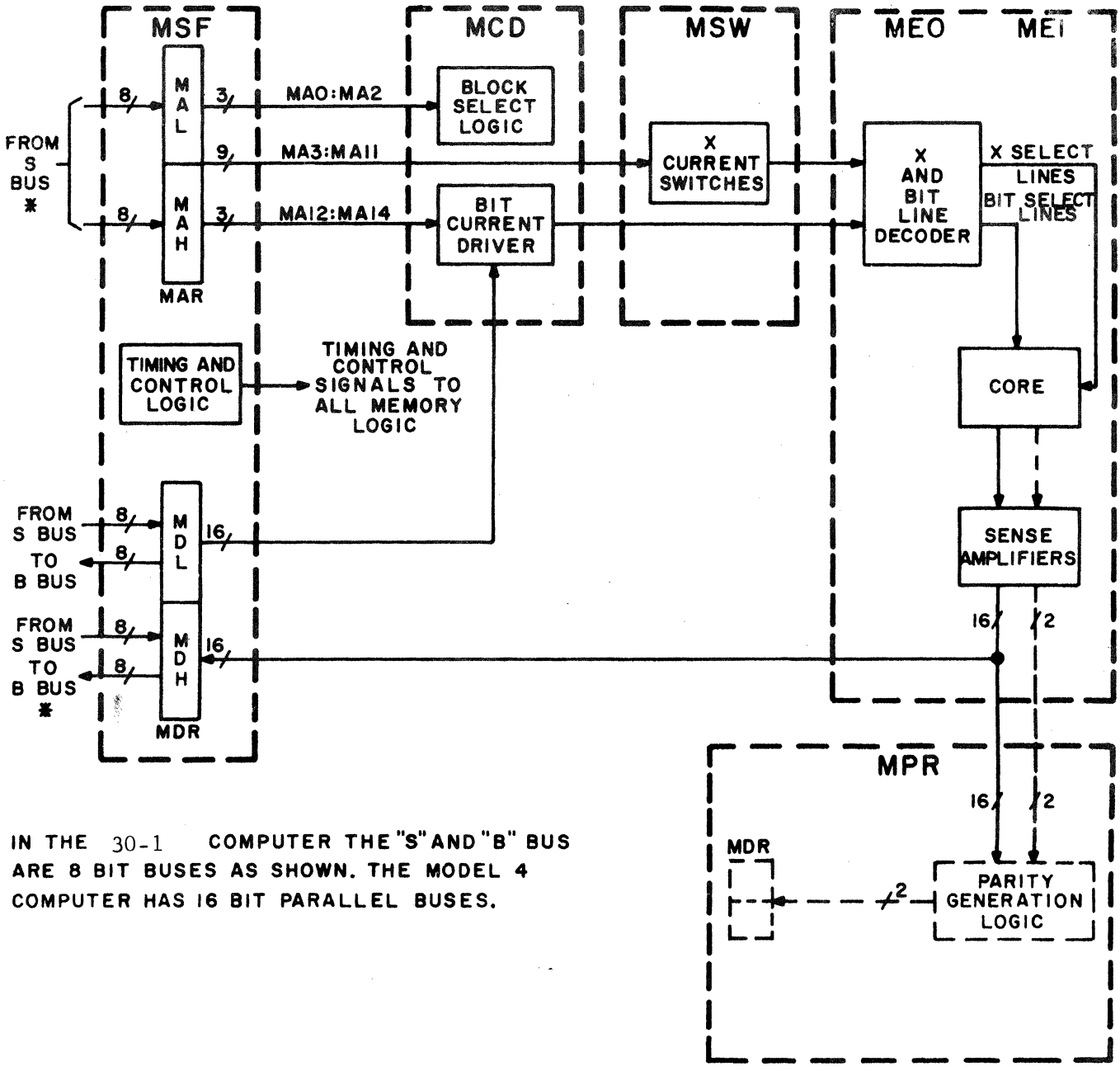


Figure 5 Memory System Block Diagram

Timing and control logic directs the sequence of operations within the memory system, depending upon the operation (read-restore or clear-write) to be performed. The Memory Data Register (MDR) stores the 16-bit halfword being read from or written to memory. The MDR is loaded from the S Bus or memory, and unloaded to the B Bus or memory. Note again that the bus transfers may be in 8-bit bytes, while the transfers to or from memory are in 16-bit halfwords.

The optional parity logic is shown on the bottom of Figure 5. Odd parity is generated for each byte if this option is installed. Note that the **MPP** mother-board also provides two additional MDR stages which store the parity bits.

5. ADDRESSING

Addressing in a 2 1/2D memory requires the selection of the two wires, plus selection of the direction of current flow in each wire. Figure 6 is a simplified schematic of the memory stack. Four cores (designated A, B, C, and D) are shown on Figure 6. The four cores are all the same bit position, i.e. the same bit position in each of four words. The transistor switches (designated S1 through S8) select the appropriate wires, and determine the direction of current flow through the wires. Thus, S1 through S4 select the bit wire and bit current direction, while S5 through S8 select the word wire and word current direction. The MAR decoding used to select the appropriate transistor switches is shown on Figure 7. For more detailed

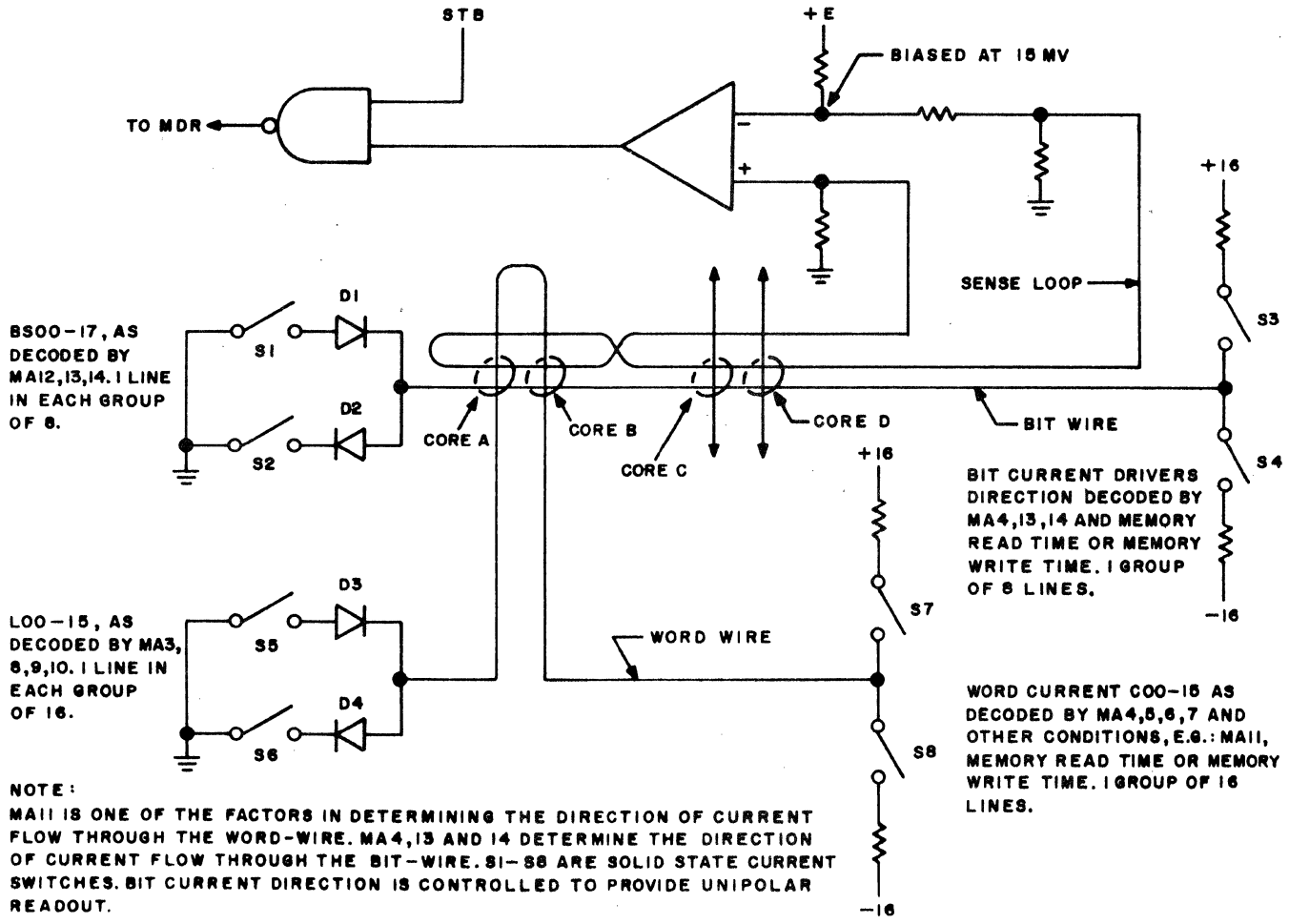


Figure 6. Memory Addressing, Simplified Schematic Diagram

information on the decoding, refer to the Functional Schematic FS50 sheets referenced on Figure 7.

Returning to Figure 6, assume that to write into Core A, Switches S2, S3, S5, and S8 are closed. Once that assumption is made, the following statements are true:

1. To write into Core B, Switches S2, S3, S6, and S7 are closed.

2. To read Core A, Switches S1, S4, S6, and S7 are closed.

3. To read Core B, Switches S1, S4, S5, and S8 are closed.

Figure 8 shows the layout of the memory stack (Mother-boards ME0 and ME1). The bit lines are functionally grouped into groups of 8 lines each.

NOTE

Actually, the bit lines are physically interspersed such that BS00 in one group is adjacent to the corresponding line (BS10) in a group wired from the other direction, BS01 is adjacent to a BS11, etc. Thus, noise signals generated by the select lines tend to cancel.

The BWnn signals select a group of 8 bit lines, for example, BW17 shown on the upper right of Figure 8. The BWnn lines correspond to the bit position in the memory word. The BSnn signals select 1 line in each group of 8 bit lines. Thus, a single bit line in each group of 8 lines which has been enabled by the appropriate BWnn signal, is selected. For example, during a Read operation one bit line in each group is selected. During a Write Operation, only bit lines corresponding to positions which are to store a binary ONE are selected.

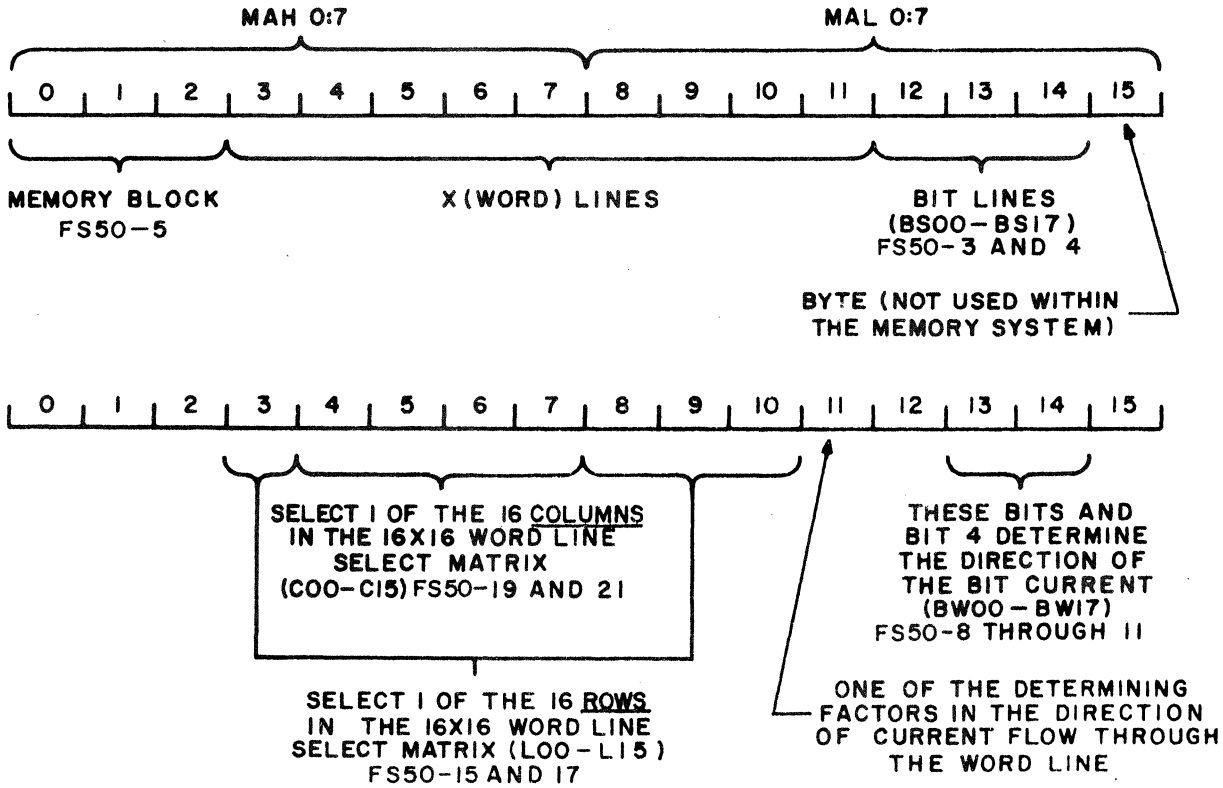


Figure 7. MAR Decoding

NOTE

Figure 8 shows an 18-bit stack. Stacks which do not contain the optional parity feature do not contain the BW16 and BW17 groups of lines.

The folded X or word lines are shown vertically on Figure 8. A Cnn signal selects 1 of 16 groups of 16 word lines (8 word lines on ME0 and 8 on ME1). The Lnn signal then selects 1 of the 16 lines in each of the 16 groups. Note that the method of Word line selection is similar to the method used with the Bit lines. Only the space limitation on Figure 8 makes the Word line arrangement more difficult to visualize.

6. TIMING

Normally, the sequence of memory operations is:

1. The Processor loads MAL.

2. The Processor loads MAH.

NOTE

The 30-2 Processor loads MAL and MAH simultaneously.

3. The Processor loads MDL.

4. The Processor loads MDH.

NOTE

The 30-2 Processor loads MDL and MDH simultaneously.

5. The Processor sends a start signal (STRT,O) and a Read or Write indication (WR,O) to memory.

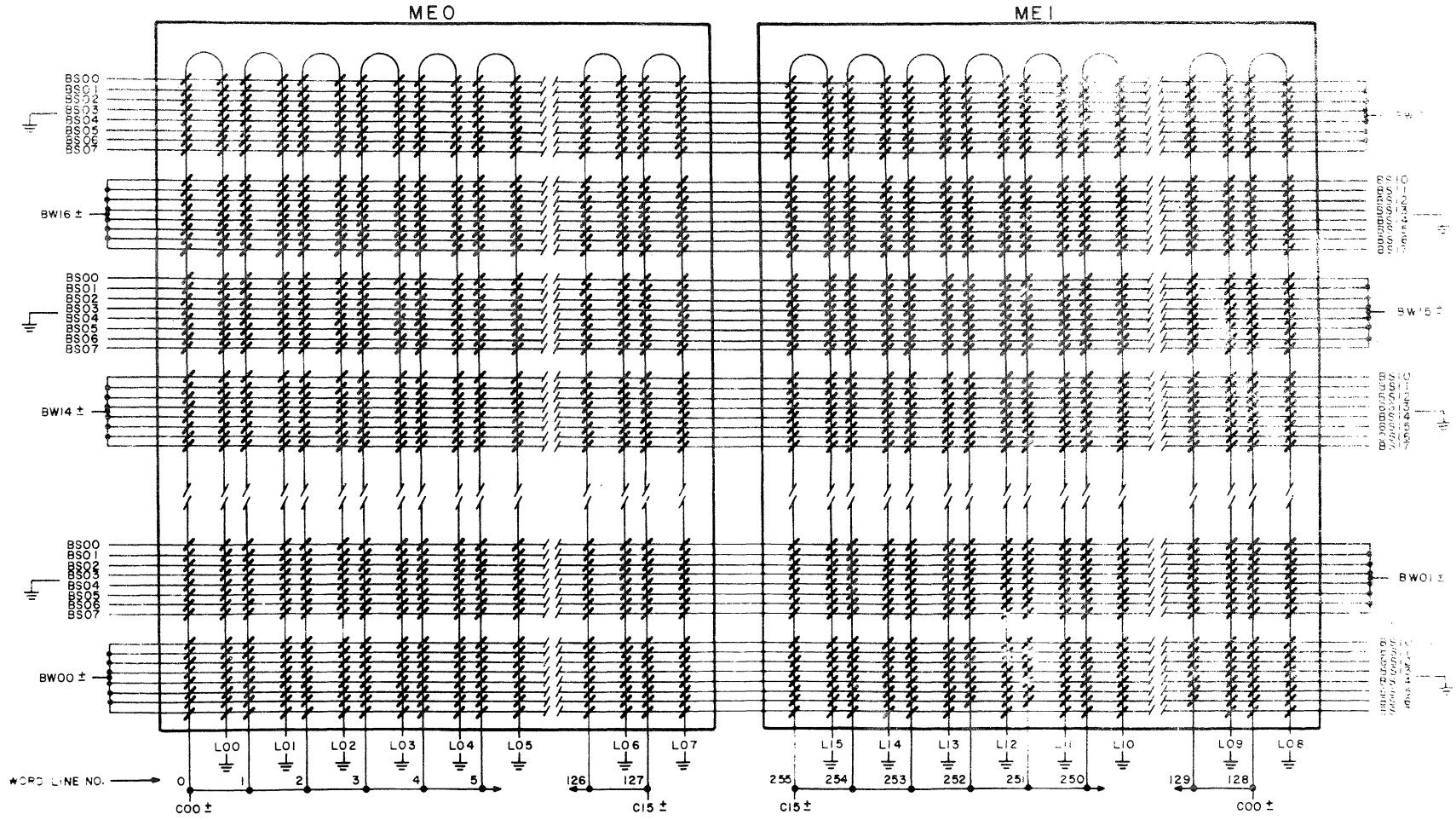
6. The memory raises a Busy line (MBY,I) to the Processor.

7. The memory performs the requested operation and then lowers MBY,I.

Figure 9 shows the timing at the initiation of a memory operation. A Start (STRT,O) signal from the Processor initiates each memory cycle. As noted on Figure 9, the STRT,O line must be high between memory cycles. The STRT,O must be high when the memory comes out of busy, or it will restart immediately. The STRT,O signal is normally reset on the next system clock, when the next ROM instruction is fetched by the Processor. The STRT,O signal sets the Memory Busy (MBY) flip-flop and starts a delay which produces the 50 nanosecond Command Sense Pulse (CSP). The CSP signal is used in the memory to sense the Write (WR,O) line from the Processor. Thus, the memory has been addressed, has responded with a MBY signal to indicate that a cycle is in progress, and has determined which of the four memory operations it is to perform.

Figure 10 shows the timing for each of the four memory operations in the 30-1 Processor. Figure 11 shows the full cycle timing for the 30-2 Processor.

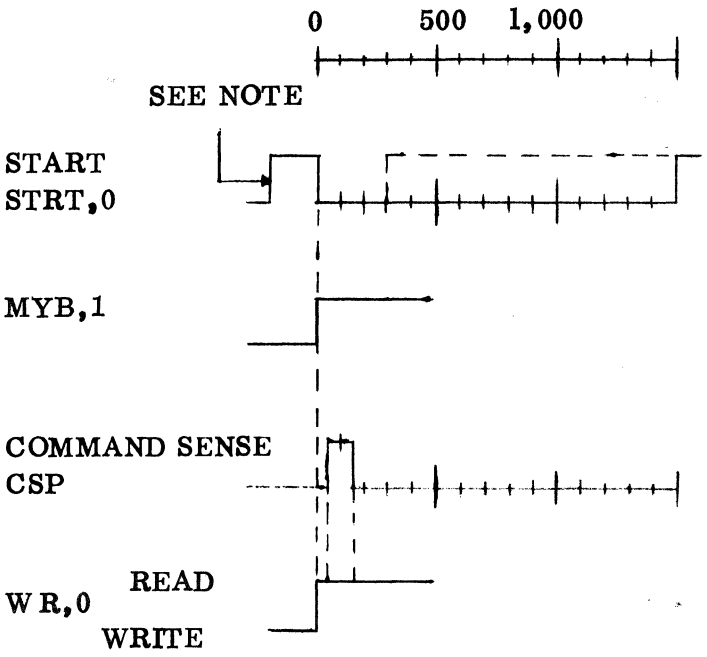
The two memory systems are identical except for the timing and the size of memory cores used. Model 30-2 memory requires 23 mil cores. Model 30-1 memory may use 30 mil or 23 mil cores. Current drive levels are identical and all circuit packs are interchangeable. Starting with the Write or Erase and Write (EAW) operation, note that the Bit Read Signal (BR,O) is developed as soon as the EAW operation is sensed. Approximately 300 nanoseconds later, the Word Read signal (XR,O) is developed. The addressed word is then read, but is not gated to the MDR. The net effect is an erase of the addressed location. The Write Switch signal (WS,O) is generated next, followed by the Write Current signal (WC,O). The new data



NOTE:
X (WORD) LINES – VERTICAL
BIT LINES – HORIZONTAL

Figure 8. Memory Stack Layout

from the MDR is now written into the addressed location. The blip on the Memory Data Available (MDAV,O) line is not significant during the EAW operation. The MDR is not changed by this operation. If the optional parity feature is included, parity is set at the start of the cycle by SPAR,O and checked just prior to the Write operation by CPAR,O.



NOTE:
MINIMUM 50 NANOSECONDS.MUST GO HIGH 50 NANOSECONDS BEFORE NEXT CYCLE.

Figure 9. Memory Cycle Initiation Timing Diagram

The Read or Read-Regenerate (RRG) operation is shown next. Note that the BR,O, XR,O, WS,O, and WC,O signals are exactly as described for the EAW operation. The MDAV,O line is inhibited until the new data has been loaded into the MDR. The MDHR,O and MDLR,O signals clear the MDR prior to the read. The parity is checked as described previously.

7. CIRCUIT DESCRIPTIONS

This section provides circuit descriptions which reference the functional schematics included in Volume 2 of this manual.

7.1 Bit Access

Sheets 1 and 2 of FS50 shows the memory bit wires and the bit access diodes. Wires BW0-0 through BW16-7 are wires in the core plane. The memory address causes two pairs of bit switch leads (for example, BSION and BSIOP, and BSOON and BSOOP to be grounded. The 14 other pairs (16 pairs if the parity option is installed) are like an open circuit as the associated diodes are back biased by $\pm$ 15 volts through the 1K resistors shown. During B-read time (see Sheet 28), the B-current drivers switch the common end of the B-wires (BW00 through BW17) to $\pm$ 16 volts through a current limiting resistor. Current flows through one of the 8 B-wires in each bit plane, depending on which pair of B-access switches is closed to ground. The voltage on the common end of the B-wires rises to approximately 10 volts as the increasing current changes the B-wire inductance, decaying in about 100 nanoseconds to a steady state value of 2.5 volts during the flat top of the B-current. The direction of the B-read current depends on address. At any given address every other B-current will be in the opposite direction. For example, assume that BSION and BSIOP are grounded. BW00, BW04, BW08, BW12 and BW16 are then switched to a positive current supply, and BW02, BW06, BW10, and BW14 are switched to a negative supply. The first B-wire in each bit plane carries a half drive current of approximately 300ma. Five B-drivers supply current into the BSIOP lead to its access switch (1.5 amp). Four drivers draw current from the BSION lead and its access switch (1.2 amp).

NOTE

If the memory does not include parity, BW16 and BW17 do not exist and both access switch leads carry 1.2 amps.

The driven (common) end of even numbered bit wires is terminated on the ME0 mother-board. These B-wires are continuous through the folded core plane and connect to diodes on the ME1 mother-board.

The odd numbered B-wires and their access diodes are shown on Sheet 2. The driven end of odd B-wires is on ME1 with the diodes on ME0. Otherwise, Sheet 2 is the same as Sheet 1.

NOTE

If bit 0 current is positive (i.e. toward the diodes), bit 1 current will also be positive and bits 2 and 3 current will be negative, etc.

Sheets 3 and 4 show the 1.5 amp B-access switches and address decoding. The switches are closed to ground in pairs. Control comes from a 1 out of 8 decoder whose input is Memory Address bits MA14, MA13, and MA12 (MA15 is not used by the core memory). The decoder is gated by the positive EN$\overline{1}$ (enable) pulse. The bit access switches are dc coupled switches that close at the beginning of the cycle and remain closed to the end of the cycle. The even numbered switches shown on Sheet 3, and the odd numbered switches shown on Sheet 4, operate in parallel. Each group has its own decoder because of the fanout power required.

Sheet 5 is an address field recognition circuit that generates enable pulses if the address is in the range of a particular memory block. As described previously in Section 5.5, the core address is a 16 bit number. The least significant bit is not used by core. A 4,096 word core uses 12 address bits. This leaves the 3 most significant bits unused. A Processor can contain up to 8 4096-word memory blocks. Bits 0, 1 and 2 select the block. On Sheet 5, the four most significant bits are converted to double rail, and then connected to an AND gate through a hard-wired jumper board. This allows selection of memory in 2,048 or 4,096 word blocks. The negative enable ENOA is interlocked through the ME0 mother-board. This disables ENOB, which in turn disables all current drivers, if the ME0 mother-board is not plugged in. The current drivers should not be operated into an open circuit because of the breakdown voltage on the transistors.

Sheet 6 is the control logic for B-current direction and timing. The direction of the B-read current must be changed depending on address in order to obtain a unipolar readout signal. Address bits 4, 13 and 14 determine B-current direction, as shown on the 1 out of 8 decoder (35-019) on Sheet 6. The current driver gate leads R1W0$\overline{1}$ and R0W1$\overline{1}$ are normally low. If the address is all 0's, R1W0$\overline{1}$ goes high for Read and R0W1$\overline{1}$ goes high during Write. The B-read pulse (BR$\overline{0}$) is a negative going pulse which defines Read time. The Write Current Pulse (WC$\overline{0}$) is a negative pulse (ENOB) is negative during both Read and Write. If the ENOB is not present, both R0W1$\overline{1}$ and R1W0$\overline{1}$ are shunted to ground.

WS$\overline{0}$ is the write switch pulse (negative pulse). It is simply inverted on Sheet 2. See Sheet 28 or Figures 10 or 11 for pulse timing.

Sheets 7, 8, 9 and 10 show the 16-bit current drivers. Refer to Sheet 7. At t 0, R1W0$\overline{1}$, R0W1$\overline{1}$ and WSIA are low. Assume that the address is all 0's. R1W0$\overline{1}$ goes high for Read, turning on the four associated ac switches (35-005) or (35-039). Note that B-read currents BW00 and BW04 are positive, while BW02 and BW06 are negative. During Write time, WSIA goes

high so that if a memory data lead (MD00:MD15) is high, it disables the respective ac switch drive gate for either polarity. Next, the current gate lead (e.g. R0W1,1) goes high and fires the drivers associated with a data input of 1.

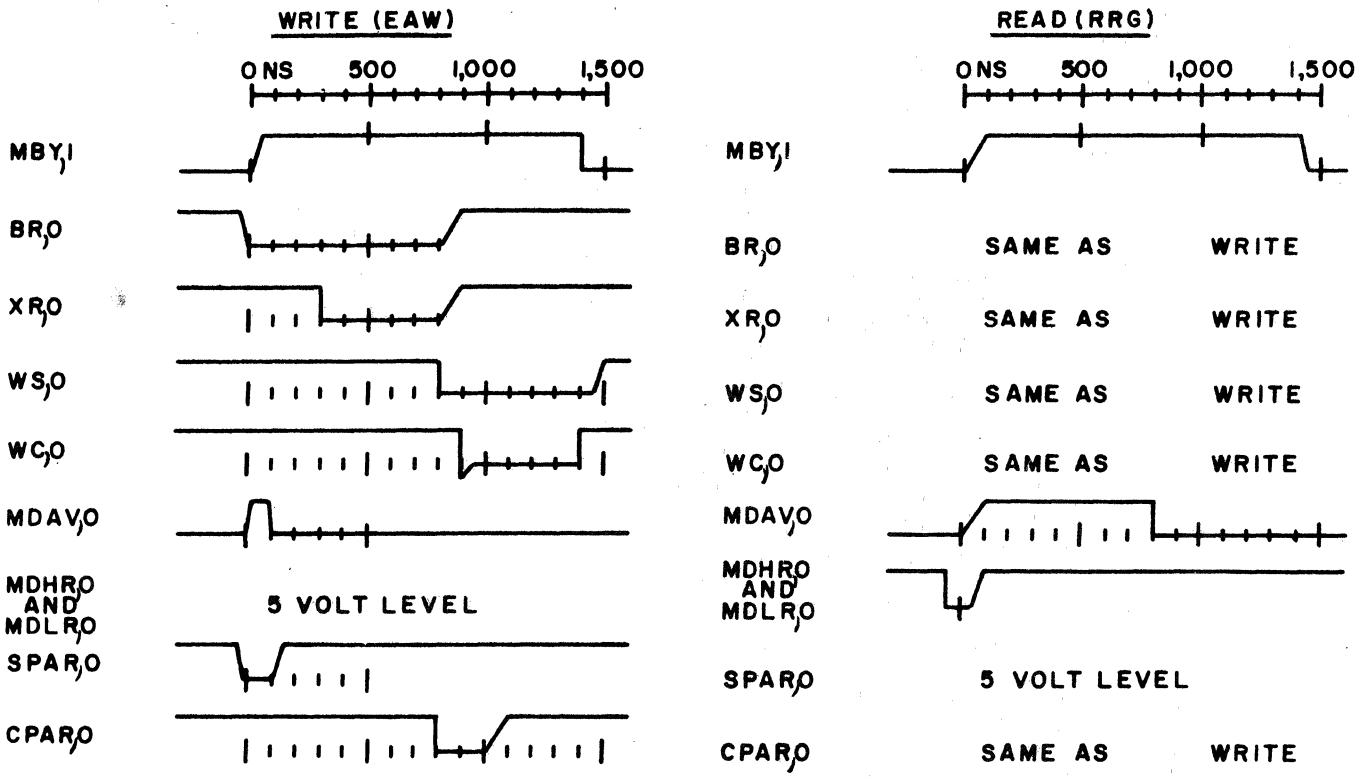


Figure 10. Model 30-1 Memory Timing Diagram

The current drive resistor (e.g. R25 and R26, Sheet 7) and the supply voltage are the major current determining parameters. Nominally R25 and R26 are 75 ohm 3 watt, non-inductive resistors. Resistor R30 is a 68 ohm 1/4 watt damping resistor to prevent turn-off ringing.

Sheet 11 shows the current drivers for the two parity bits. It is the same as shown on Sheets 6:10 except that the Write control gate is on the parity logic circuit shown on FS51, Sheet 3.

7.2 X(Word) Access

Sheets 12 and 13 show the 1 out of 256, word selection matrix. It is a 16x16 bipolar diode matrix. Each sheet shows an 8x16 matrix, the Column leads C00 through C15 are common to both sheets. The matrix "Line" leads L00A through L15C are connected to ground through "line switches". The column leads C00 through C15 are connected through "column switches" to current regulating resistors (Sheet 14). In the idle state, all switches are open. The column lines are biased at ground through a resistor and all diodes are back biased by 16 volts through the resistors shown on Sheets 12 and 13. To drive positive current in X-line 0, L00C is switched to ground and C00 is switched through a resistor (approximately 33 ohms) to +16 volts. For negative current in X-line 0, L00A is grounded and C00 is switched through a resistor to -16 volts.

Sheet 14 is the X-current control logic. It is similar to the B-logic shown on Sheet 6. The LP1 or LN1 lead is raised to close the X-line switch. Address bit 11 determines X current direction. If MA11 is 0, LP1 goes high when BR0 goes low. Slightly later, at X-read time, XR0 goes low and allows CP1 to go high. CP1 or CN1 turns on the selected column switch. During write time, WS0 and WC0 reverse the states of the LP1-LN1 and CP1-CN1 leads.

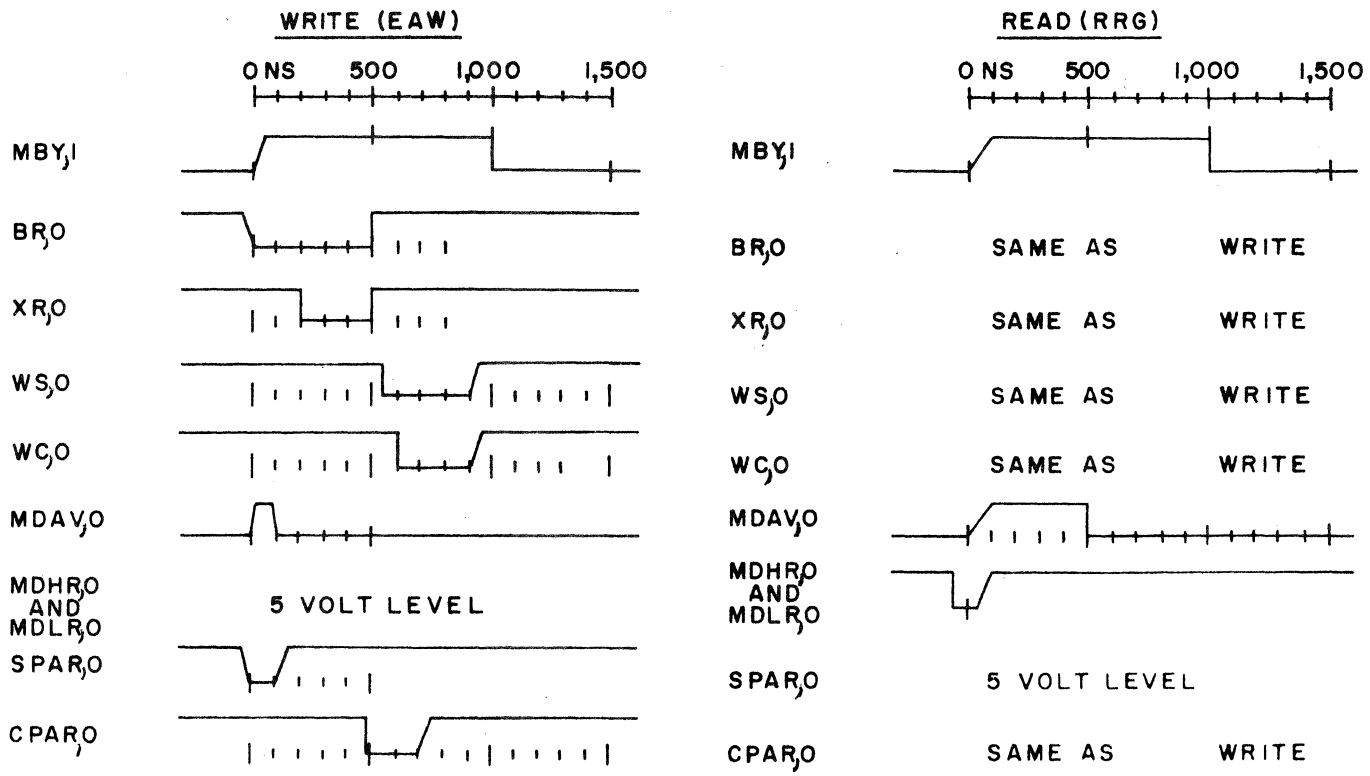


Figure 11. Model 30-2 Memory Timing Diagram

Again, ENOB disables everything if the memory is not enabled. See Sheet 28 and Figure 10 or 11 for timing details.

Sheets 15 through 22 show the address decoders and switches for the X Access.

7.3 Readout

Sheet 23 shows the readout system. The readout amplifier is an integrated differential amplifier whose output ranges between +3 volts or -0.5 volts. The gain is nominally 1500, so than an input difference signal of 2 millivolts swings the output from one limit to the other. The input leads SxN and SxP connect to the sense lines in the core plane. Core readout signals are unipolar and cause SxP to go positive with respect to SxN. A "1" output signal is greater than 30 millivolts and a "0" output is less than 10 millivolts.

The input resistance network puts approximately 15 millivolts bias on the differential amplifier, forcing the output to a logic zero. If the input signal exceeds this 15 millivolts bias, the output goes high. During readout time, a positive strobe pulse STB gates the readout onto the MS lead. Strobe width is approximately 100 nanoseconds. See Sheet 28 or Figure 10 or 11 for timing details.

Sheet 24 shows the strobe timing delay. Strobe (STB) time is referenced to the X-Read gate (XR,0). The delay is nominally 210 nanoseconds. See Section 8.2.

Sheets 25 and 26 show the single-rail to double-rail address buffers. Input is from the basic Memory Bus (Memory Address Registers).

7.4 Memory Interface

FS28, Sheets 1 and 2, show the Memory Address Register (MAR). The MAR is divided into a high (MAH, Bits 0:7) and a low (MAL, Bits 8:15) section. Input is from the S Bus. The content of the S Bus is gated into the register by the positive pulses LMAH,1 or LMAL,1. The buffered output feeds the Memory Address bus. Note that the Memory Address bus is a false bus (1's low). The output of the MAR is dc, with no provision for gating.

Sheet 3 and 4 show the Memory Data Register (MDR). The MDR is also in two 8-bit sections MDH (Bits 0:7) and MDL (Bits 8:15). Input is from the S Bus gated by LMDH,1 or LMDL,1. The MDR may be reset by negative pulses on the MDRH,0 and MDRL,0 lines. Each flip-flop is set directly by the readout on the Memory Sense (MS) bus. The MDR is "unloaded" onto the B Bus by the UMDH,1 or UMDL,1 lead, through buffers. Unload signals do not change the register. The 0-sides of the MDR flip-flops drive the Memory Data bus to control the Bit-current drivers. When writing into memory, the MDR may be set anytime before the WS0 signal gates the MD bus into the B-current drivers. (See Sheets 7-11 of FS50.)

NOTE

In the 30-2, the S and B Buses are 16-bit buses. The high and low load or unload signals are therefore generated simultaneously in the 30-2. For example, LMAH,1 and LMAL,1 are generated at the same time to load the address.

Sheet 5 (FS28) shows the memory timing circuits. Timing is started by a negative going edge on the STRT,0 lead.

A negative edge on STRT$_{,}$0 sets the JK flip-flop. If the memory is not busy, both inputs to the delay in location 41 are high and the delay begins timing. At t=50 nanoseconds, the delay output sets the MBY flip-flop, interrogates decoding gates, and sets parity.

Consider first the action on a RRG order. Decoding gate output 31R-06 goes low which: (1) sets the B-read flip-flop, 34L-06 goes high, and buffer output BR$_{,}$0 goes low (2) sets the full cycle flip-flop, 31R-08 goes high so that the output of the read timing chain will continue through the write timing chain (3) starts the delay in location 46. Delay 46 times out at t=350 (or 250) nanoseconds and the positive output pulse starts Delay 45. Pin 71-45 goes high during Delay 45. Pin 71-45 goes high during Delay 45 forming the XR$_{,}$0 timing pulse. XR$_{,}$0 triggers the delay on the ME0 board, developing STB. The end of Delay 45; (1) resets the B-read flip-flop with a negative pulse on 34L-13 ending the BR$_{,}$0 pulse. (2) Sets the MDAV flip-flop that had been reset at the beginning of the cycle by decoding output 30R-11. (3) Generates CPAR$_{,}$0 to check parity. (4) Checks the state of the full cycle flip-flop 31R-08. In this case, 31R-08 is high, so 33R-03 goes low setting the Write Switch flip-flop 33R-04, and starting Delay 44. The Write Switch flip-flop forms the WS$_{,}$0 pulse. Delay 44 provides a time for Read current to decay, Write Switches to close, and Write logic to stabilize. Delay 43, pin 71-43 forms the WC$_{,}$0 pulse. The output from Delay 43 initializes the timing and resets the MBY flip-flop.

Timing for the other memory operations should be easily traceable. Following are some general comments on timing.

1. If memory is busy, the STRT$_{,}$0 pulse sets the JK flip-flop so that as soon as MBY is reset, the next cycle begins.
2. If parity is used, the parity bit is computed and set, by SPAR$_{,}$0 at t=50 nanoseconds. Therefore, the data register must have been set up at t=0.
3. Parity is checked at the end of read time on <u>both</u> Read and Write orders. Thus, it is possible to get parity fail on a Write operation if a logic error exists.

7.5 Parity and Memory Protect

The parity and memory protect circuits are shown on FS51. These are optional circuits, either or both may be provided, and they are not related except that they share the same mother-board.

The Parity circuit is shown on FS51, Sheets 3, 4, 5, and 6. There are two separate 9-bit parity generators and check circuits. Bit 16 is the parity for data bits 0-7. Bit 17 is parity for data bits 8-15. Sheet 6 and the upper half of sheet 5 show a conventional 9-bit parity using two stages of 3-bit parity circuits. The data input (e.g., MD00$_{,}$0) is high (+5v) for ZEROs. The output from the check circuit BYP16$_{,}$0 (Sheet 5) is high for even parity. Referring to Sheet 4, if BYP16$_{,}$0 is high, the set parity pulse SPAR$_{,}$0 will change the state of the Bit 16 flip-flop, thus making parity odd and making BYP16$_{,}$0 low. Note that the circuit does not set the parity bit depending on the 8 data bits, but it rather changes the parity bit if it is wrong.

To check parity on a Read, the Parity Register flip-flops (Sheet 3) are reset by MDHR$_{,}$0. Bit 16 readout, MS16$_{,}$0, would set its flip-flop to a 1 and data bit 16 MD16$_{,}$0 along with MD00$_{,}$0 - MD07$_{,}$0 are decoded in the 9-bit parity check circuit. If parity is even (error condition), BYP16$_{,}$0 will be high and the parity check pulse, CPAR$_{,}$0 (Sheet 4), will put a 100 nanosecond negative pulse on PPH$_{,}$0 and set the Parity Fail flip-flop. The 0-side of the Parity Fail flip-flop, PFH$_{,}$0, flags the Processor via the MPF$_{,}$0 signal.

The low byte parity, bit 17, operates the same as bit 16.

In the standard system, the outputs from the byte parity circuits are tied together so the system recognizes a parity failure on either byte.

The Memory Protect Circuits provide a means of allocating blocks of core to be protected so that a Write order for an address in the block will be converted to a Read-Restore order, and a flag will be set to indicate the action. The core memory may be partitioned into 16 blocks with an individual protect control lead for each block. An overall protection override provides for loading any area in core.

Allocation of protected areas can be controlled by one of the following methods:

1. Hard-wired protection pattern with no external control, wherein it would be necessary to physically remove the Memory Protect Circuit Board from the card file to initially load the memory.

2. Hard-wired protection pattern with the override lead brought out to a remote switch to allow loading protected areas.

3. Selectable protection pattern with the 16 block control leads and the override brought out to a remote control system, for example toggle switches.

4. Programmable protection pattern with the block control leads cabled to a special I/O Controller to provide dynamic changes in the protection pattern under computer control. Trying to write into protected areas causes an I/O interrupt.

The memory protect circuit monitors the internal memory address bus and may, therefore, be used directly on any 30-1 or 30-2 computer with Standard Memory Interface, Direct Memory Access, or High Speed Direct Memory Access, without sacrificing execution time.

The Memory Protect Circuit is illustrated in Figure 12. A wire-wrap address jumper board is provided to allow different block sizes as follows:

Address Bits	Bytes/Block	Bytes/16 Blocks
0 1 2 3	4096	64K
1 2 3 4	2048	32K
2 3 4 5	1024	16K
3 4 5 6	512	8K
4 5 6 7	256	4K

The jumper board is normally wired for 1024 Byte Blocks and may be changed in the field.

The individual block control leads, as well as the override and flag leads, terminate on a daughter-board location which may contain a wired jumper board for a fixed protection pattern, or cable connection to a remote controller.

FS51, Sheet 2, shows the detail wiring for the 1 out of 16 decoder of Figure 12. Sheet 1 shows the OR gate, the Flag flip-flop, and the control circuits. Sheet 1 shows a range decoder (not illustrated on Figure 12) that may be used if more than 16 protection blocks are required. This, range decoder, will override protection if the address is not in range. It is used for paralleling memory protect circuits or it may be used where a portion of core is partitioned into protection blocks and the rest of core is open.

Figure 13 shows the detailed pin configuration of the Address jumper board and the control cable connectors.

8. TESTS AND ADJUSTMENTS

This Section describes tests and adjustments which should be used periodically as a check of memory system operation, and as an aid in trouble-shooting when a memory problem exists. Perform the checks in the order that they are listed in this Section.

NOTE

The Power Supply adjustments listed in the Power System Section, and the System Clock adjustments listed in the Processor Section, should be performed prior to the memory adjustments.

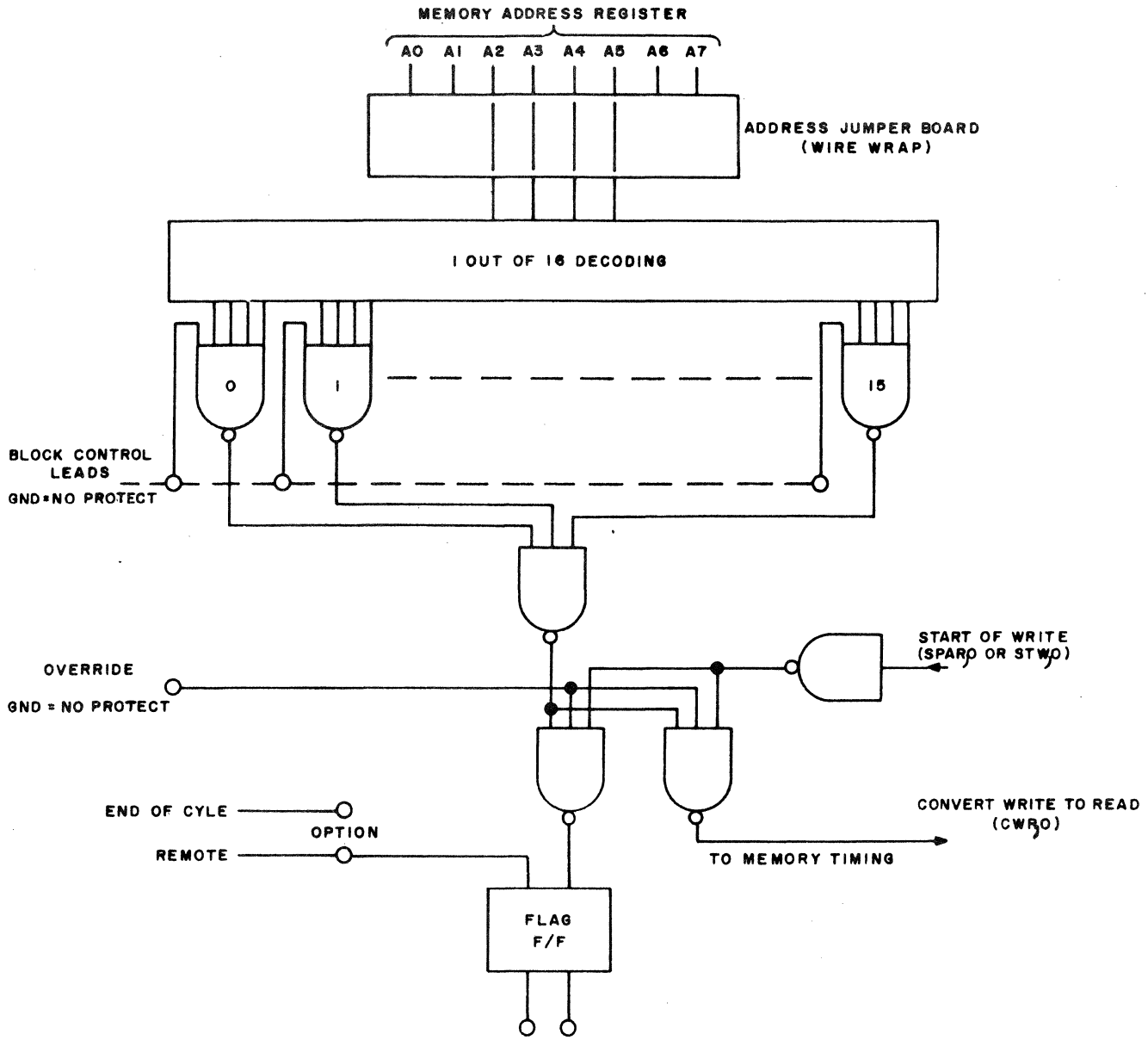
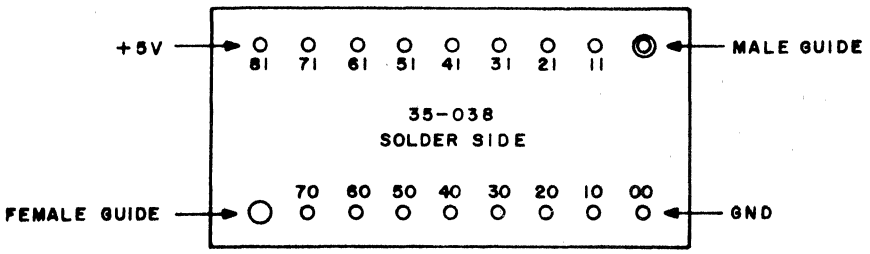


Figure 12. Memory Protect Block Diagram



DAUGHTER-BOARD PIN No.	PROTECT ADDRESS JUMPER LOC. 07	RANGE ADDRESS JUMPER LOC. 44	CABLE CONN. LOC 46.	CABLE CONN. LOC. 47
00	GND	GND	GND	GND
10	CPR,0 (Clear Flag)	MA03,0	BC2	BC12
20	MBY,1A (BUSY)	MA03,1	BC4	BC8
30	PA0	INRG,1	BC6	BC10
40	PA3	MA01,0	BC0	BC14
50	PA2	MA01,1	PROL,0 (D.C.)	
60	PA1	MA0,1	CWR,0 (PULSE)	
70	MA01,0	OVRB,0	PROS,1 (HI FLAG)	
11	MA07,0	MA02,0	BC1	BC15
21	MA06,0	MA02,1	BC5	BC13
31	MA05,0	A3	BC7	BC11
41	MA04,0	A2	BC3	BC9
51	MA03,0	A1	OVRB,1 (OVERRIDE)	
61	MA02,0	A0	CPR,0 (CLEAR FLAG)	
71	MA00,0	MA0,0	PROSO (LO FLAG)	
81	+5V	+5		

Use CPR,0, Neg. Pulse to clear flag or connect CPR,0 to MBY,1A to clear end of Memory Cycle.

Figure 13. Memory Protect Cable Connections

8.1 Voltage Adjustment

The nominal ±16.5 volt input to the memory system should be checked prior to any other memory checks. The 16.5 volt regulator on the power supply has a control input from a thermister mounted on the ME0 motherboard. It is therefore important to consider the ambient temperature when adjusting the 16.5 volt regulator. The following test equipment is required for this adjustment:

1. A laboratory thermometer accurate to ±1%.

2. A digital voltmeter capable of reading 15 to 18 volts ±1%.

Use the following procedure to check, and adjust if necessary, the 16.5 volt regulator.

1. Measure the ambient temperature. Refer to Figure 14 and determine the voltage setting for the temperature measured.

2. Compare the voltage at the -16.5 bus (the lower outer bus) and +16.5 bus (the top outer bus) to assure that the voltages are equal. If the voltages are not equal, adjust the VOLTAGE trimpot on the power supply A2 module to obtain equal readings.

3. Measure the voltage at the +16.5 bus (the top outer bus).

4. Adjust the VOLTAGE trimpot on the power supply A1 module to obtain the reading calculated in Step 1.

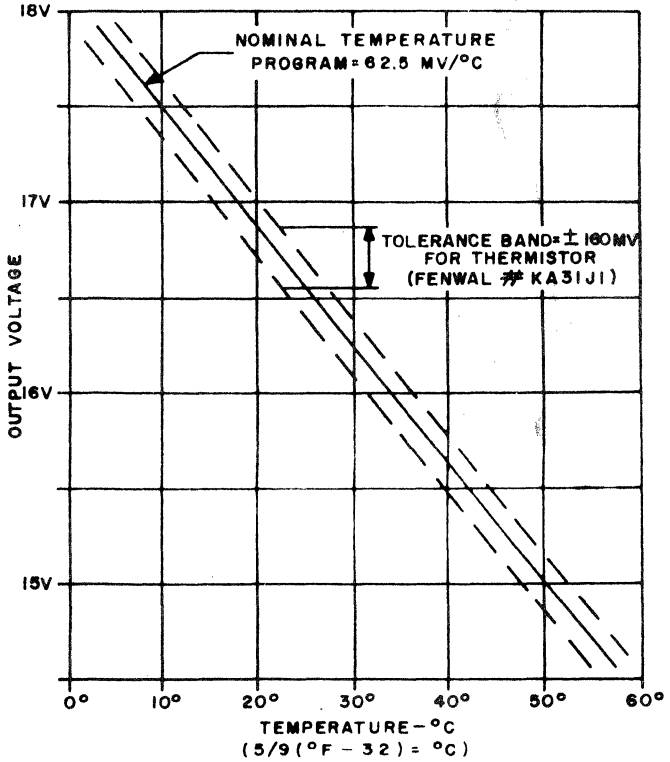


Figure 14. Thermistor Temperature Program

8.2 Strobe Timing Adjustment

There is a single timing adjustment on the GE-PAC 30 2 1/2D Memory System.

CAUTION

The adjustment described in this Section is carefully set at the factory using sophisticated test equipment not normally available in the field. The adjustment is very stable and should not require field adjustment. The adjustment should be changed only after the check provided indicates that it is out of tolerance and there are no faulty components in the system.

A dual trace oscilloscope with a calibrated time base is required for this check. Use the following procedure to check Strobe timing.

1. Load and run the Memory Test as described in Publication Number 06-003R01A12.

2. Synchronize the oscilloscope on the negative-going edge of XR0, MSF Test Point L12. See Figure 15.

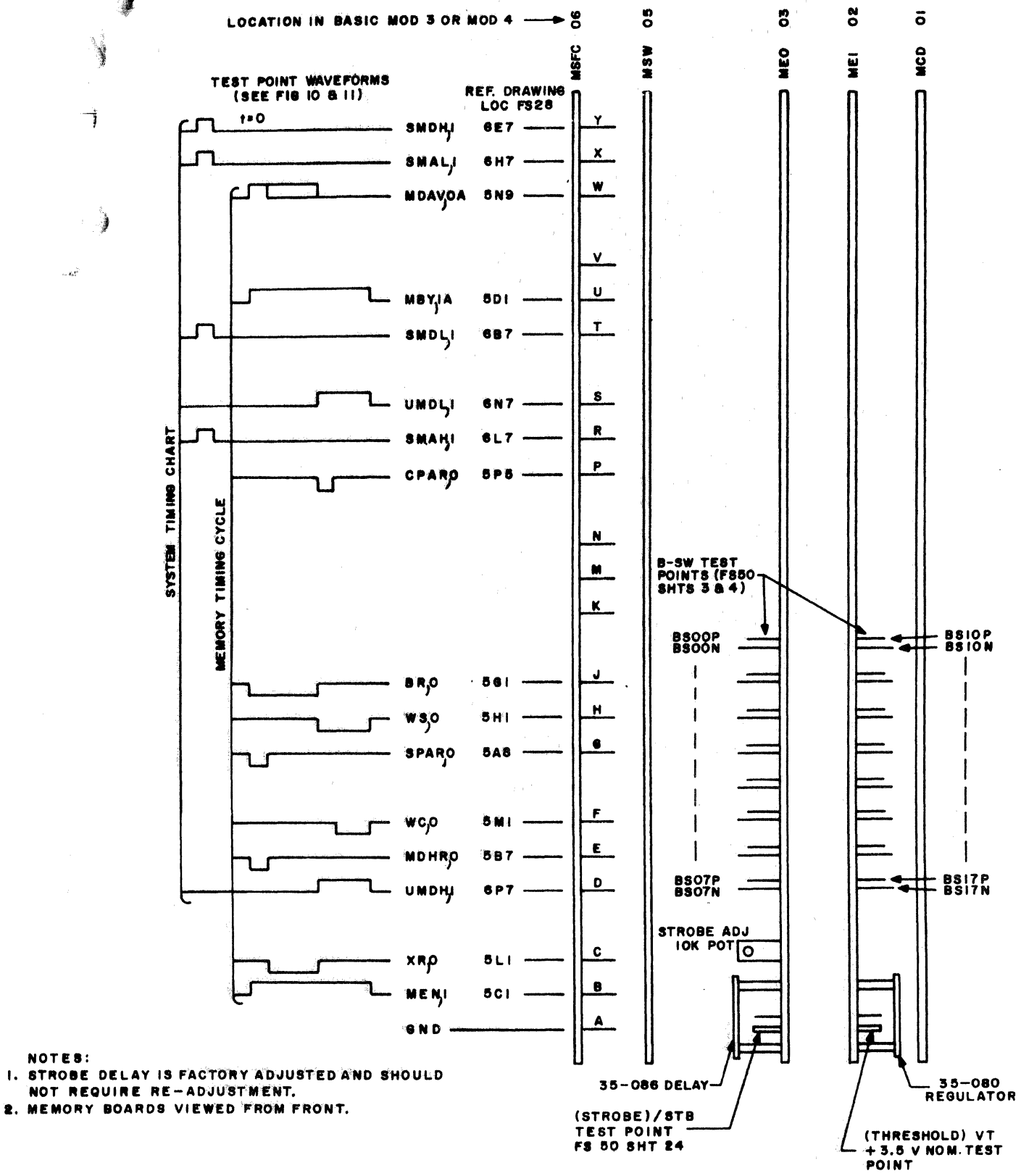


Figure 15. Memory Timing Data

3. Observe the output from any sense amplifier T0:T15 (T0: T17 if the optional parity feature is provided). The outputs are available on ME0 and ME1 back panel pins 113-1 through 120-1.

4. On the second trace, observe the Strobe output, on the bottom ME0 test point (Figure 15). The relationship should be as shown on Figure 16.

5. If the relationship is not as shown on Figure 16, and all components are found to be operating correctly, adjust the trimpot shown on Figure 15.

8.3 Memory Test

Run the Memory Test as described in GE-PAC 30 Publication Number 06-003R01A12 (part of the Programming Manual, Publication Number 29-013R02).

8.4 Marginal Test

The following marginal test may be performed periodically to locate memory areas which may cause future problems. Test equipment required for this test is:

1. The Test Circuit shown on Figure 17 or the GE-PAC 30 Systems Test Set which contains such a circuit.

2. A voltmeter capable of measuring 3 to 5 volts ±2%.

Use the following procedure:

1. Connect the test circuit as shown on Figure 17.

2. Connect the voltmeter to back-panel pin 211-1 on ME0 or ME1, or the bottom test point on ME1 (Figure 15). Set the MARGINAL TEST - NORMAL Switch to MARGINAL TEST. The voltmeter now indicates the test threshold voltage (VT) applied.

3. Run the Memory Test referenced in Section 8.3.

4. Vary VT between 2.5 volts and 4.5 volts via the potentiometer on the test setup. The test program should continue to run normally.

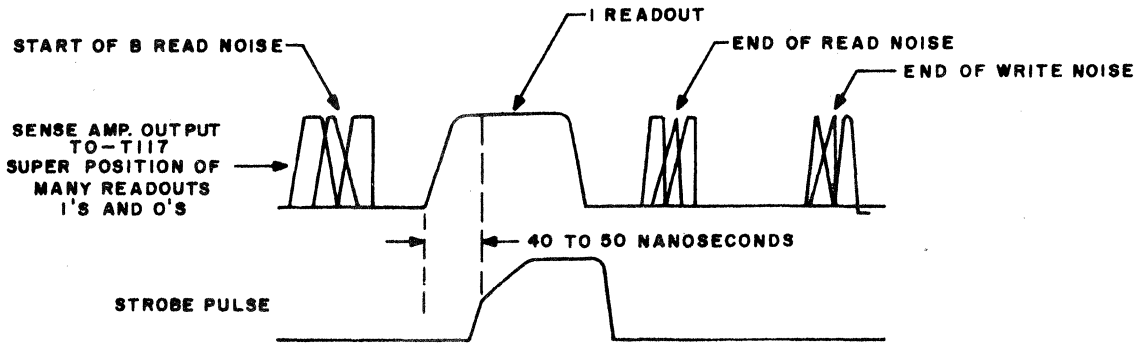
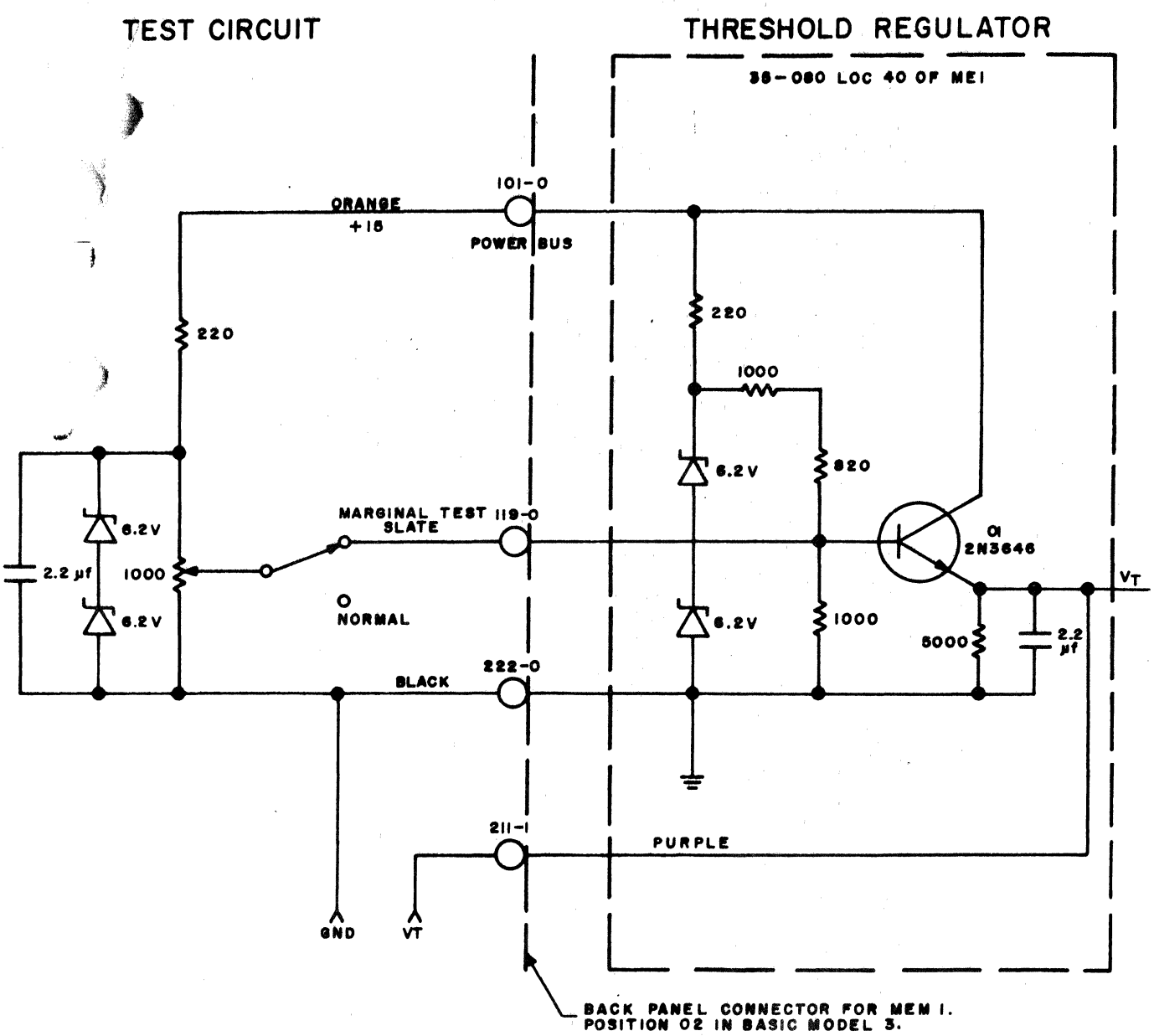


Figure 16. Strobe Timing Data



With the switch in the normal position, the Zener Regulated Divider puts the base of Q1 at 4.2 V, and VT is 3.5 V ± .1.

With the switch in the marginal test position, the pot over-rides the divider and VT can be varied from near 0 V to + 12 or 13 volts.

Core memory should operate for VT in the range of 2.5 to 4.5.

Figure 17. Memory Marginal Test Setup

9. MEMORY MNEMONICS

The following list provides a brief definition of the mnemonics used in the memory system documentation. The functional schematic and sheet on which each mnemonic is generated are also provided.

MNEMONIC	MEANING	LOCATION
BR/0	Bit Read timing pulse	FS28-5G2
BS00N:BS07N	Bit switches-negative	FS50-4R5
BS00P:BS07P	Bit switches-positive	FS50-4P5
BW00:BS16	Bit wires (common)	FS50-7P2
BW0-0:BW16-7	Bit wire, bit plane 0 wire 0 through bit plane 16 wire 7.	FS50-1P4
B00/0 - B07/0	B-Bus (bit 0-7)	FS28-3A7
BYP16/0	Byte parity (high, data 0-7)	FS51-4R3
BYP17/0	Byte parity (low, data 8-15)	FS51-6R3
CL0E	(System) Clock E	FS28-6A3
CN/1	Column negative X current gate	FS50-14P5
CP/1	Column positive X current gate	FS50-14P4
CPAR/0	Check parity timing pulse	FS28-5P5
CPFH/0	Clear parity flip-flop high	FS50-19G3
CPFL/0	Clear parity flip-flop low	FS50-19P3
C00 - C15	(X matrix) Column lead (0 - 15)	FS50-20F2
C00N/0 - C15N/0	Column negative (switch drive)	FS50-21G3
C00P/0 - C15P/0	Column positive (switch drive)	FS50-21P3
EN/0A	Enable (to interlock)	FS50-5L8
EN/0B	Enable (to interlock output)	FS50-5E9

MNEMONIC	MEANING	LOCATION
EN,1	Enable	FS50-5K8
LMAH,0, LMAH,1	Load memory address high (0:7)	FS28-6K2
LMAL,0, LMAL,1	Load memory address low (8:15)	FS28-6H2
LMDH,0, LMDH,1	Load memory data high (0 - 7)	FS28-6E2
LMDL,0, LMDL,1	Load memory data low (8 - 15)	FS28-6B2
LN,1	Line negative X-Switch gate	FS50-14P3
LP,1	Line positive X-Switch gate	FS50-14P2
L00N,0 - L15N,0	Line negative (switch drive)	FS50-15F3
L00P,0 - L15P,0	Line positive (switch drive)	FS50-15N3
L00A - L15A	Line anode (X-diode matrix)	FS50-16F3
L00C - L15C	Line cathode (X-diode matrix)	FS50-16F4
MA00:MA15	Memory Address Bus	FS28-1B8
MD00,0:MD17,0	Memory Data Bus	FS51-4A1
MDAV,0	Memory data available signal from memory to the Processor	FS28-5P8
MDHR,0	Memory Data Register high reset	FS28-5D8
MDLR,0	Memory Data Register low reset	FS28-5E7
MS00,0 - MS17,0	Memory Sense Bus	FS28-3B1
NXC	Negative X-Current	FS50-14N8
PFH,0, PFH,1	Parity flip-flop high (0 - 7)	FS51-3P1
PFL,0, PFL,1	Parity flip-flop low (8 - 15)	FS51-3P8
PPH,0	Parity fail pulse high (0 - 7)	FS51-3P4
PPL,0	Parity fail pulse low (8 - 15)	FS51-3P5

MNEMONIC	MEANING	LOCATION
PXC	Positive X-current	FS50-14N7
ROW1,1	Read reverse Write normal (B-current gate)	FS50-6N2
R1W0,1	Read normal Write reversed (B-current gate)	FS50-6N3
SCLR,0	System Clear	FS28-5C3
SPAR,0	Set parity signal	FS28-5A8
STB	Strobe	FS50-24K5
STRT,0	Start signal from Processor to memory	FS28-5A2
S0N - S16N	Sense negative side of sense wire	FS50-23A2
S0P - S16P	Sense positive side of sense wire	FS50-23A4
S00,1 - S07,1	S-Bus	FS28-1B1
T00 - T17	Test point (readout amplifier)	FS50-23P3
UMDH,0, UMDH,1	Unload memory data high (0 - 7)	FS28-6P8
UMDL,0, UMDL,1	Unload memory data low (8 - 15)	FS28-6N8
WD16,0	Write data 16 (parity high)	FS51-3L4
WD17,0	Write data 17 (parity low)	FS51-3L6
WC,0	Write current timing pulse	FS28-5M2
WR,0	Write order	FS28-5A6
WS,0, WS,1A	Write switch timing pulse	FS28-5J2
XR,0, XR,1A	X-read timing pulse	FS28-5L1

CONTENTS

ILLUSTRATIONS

THE READ ONLY MEMORY

1. INTRODUCTION

The Read Only Memory (ROM) is a high speed non-destructive memory used to store the micro-program. As described in the General Description, the micro-program interprets and executes the user program. Each ROM module provides up to 1024 16-bit words. A basic digital system can employ two ROM modules. The cycle time of the GE-PAC 30 ROM is approximately 300 nanoseconds. The ROM functional schematics (FS3) referenced in this Section are provided in Volume 2 of this manual.

2. BASIC THEORY OF OPERATION

The basic memory cell in the ROM is a pulse transformer with a U shaped core made of a linear ferrite material (see Figure 1). A multi-turn secondary, called the sense winding, is wound on the U core. The primary circuit is made up of many singularly-selectable, one-turn windings which either pass through, or by-pass the U core. Each primary winding is called a word line.

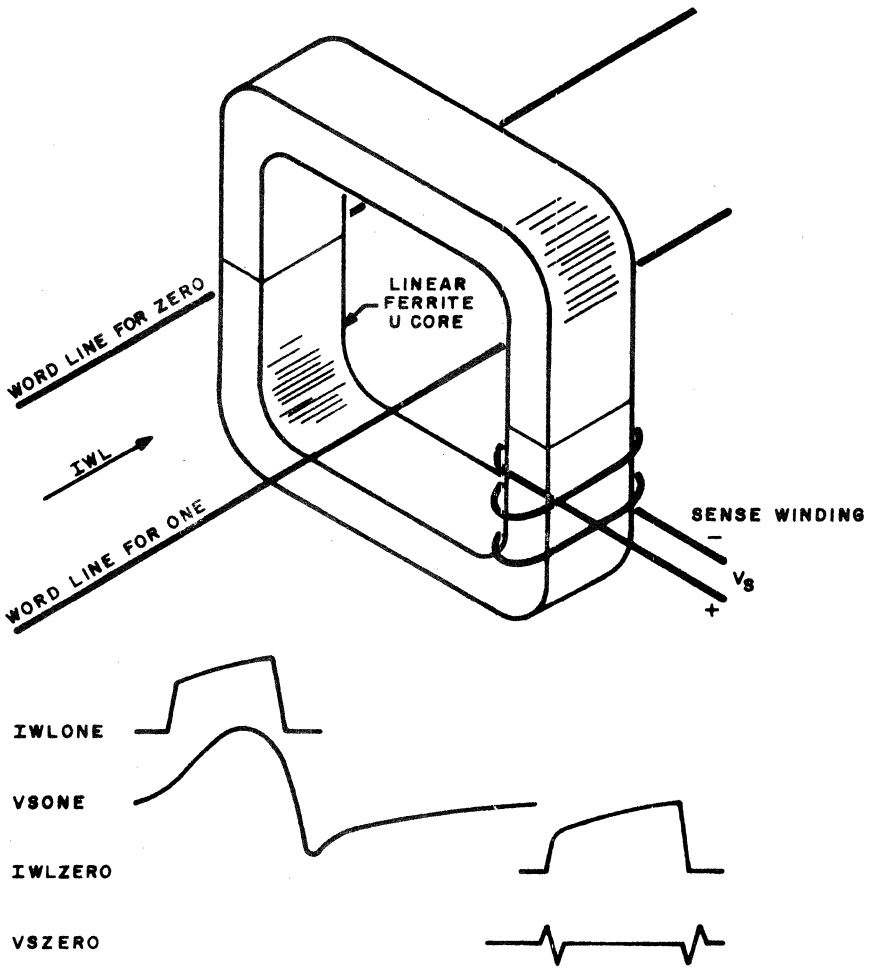


Figure 1. Basic U Core Layout

When a current pulse is developed on a word line which passes through the U core, a voltage appears across the sense winding. This word line is said to store a ONE in that transformer. Similarly, if a current pulse is developed on a word line that does not pass through the U core, no voltage appears across the sense winding. The word line is said to store a ZERO in that transformer.

Figure 2 is a simplified diagram of the ROM showing one word line and six U cores. The bit outputs are indicated on Figure 2 for the word line shown. In the GE-PAC 30 ROM, each word line is strung through 32 transformers. Thus, each word line stores 32 binary bits. A strobe selection system selects one 16-bit word from the 32 bits addressed.

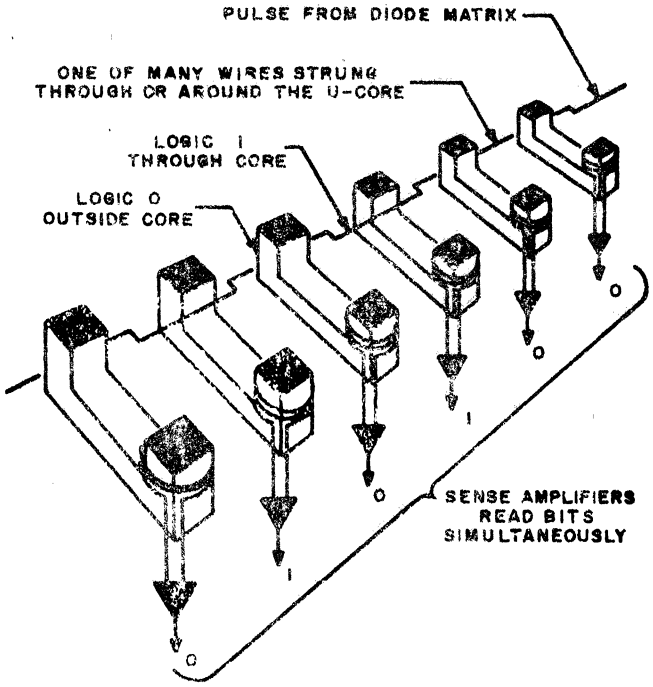


Figure 2. Basic ROM Layout

3. BLOCK DIAGRAM ANALYSIS

The basic GE-PAC 30 1024 word ROM is contained on two mother-boards; the ROM Switch (RMS) and the ROM Transformer (RMT) boards. Refer to the block diagram on FS3-8. Address decoding is performed on the switch board. A 16 X 32 diode matrix on the transformer board is driven by the switch board. With one word line per diode, there are a total of 512 word lines. Each word line is strung through 32 U core transformers. Thus, each word line stores 32 data bits, or two 16 bit ROM data words. One 16-bit data word is selected by strobing one of the two sets of 16 readout amplifiers. The selected 16 outputs are sent to the Processor double-rail, hence the 32 output lines. With this arrangement, each ROM transformer board may contain up to 1024 hard wired 16-bit words. Two transformer boards may be driven in parallel from the same switch board if 2048 ROM words are needed.

4. FUNCTIONAL DESCRIPTIONS

The following paragraphs provide functional descriptions which reference the FS3 schematics included in Volume 2 of this manual.

Figure 3 shows the significance of the ROM address (RAH and RAL) bits. For example, RAH 4 and 5 specify the RMT board if the ROM contains more than 1024 words, the RAH 6 and RAL 0:3 bits specify the X Switch, etc.

Refer to FS3-1. The ROM address buffers and single-to-double rail converters are located on the ROM switch board. Three 35-019 daughter-boards in locations 14-15, 16-17, and 46-47 provide these functions. The 35-023 daughter-board in location 05 simply inverts the 3 inputs as shown.

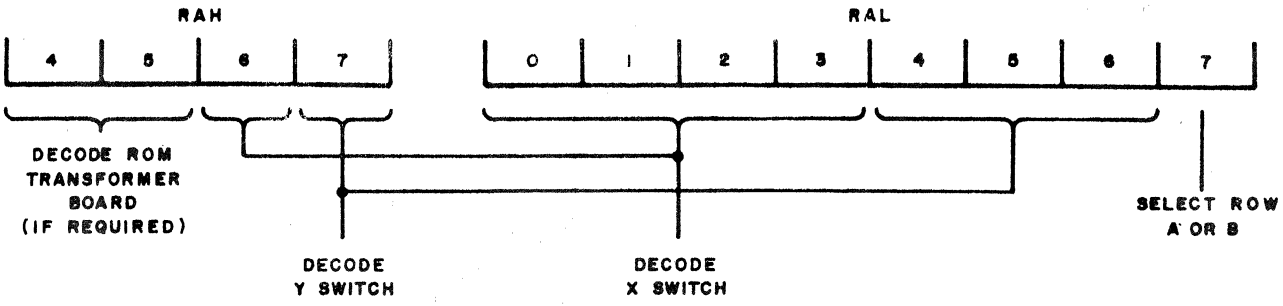


Figure 3. ROM Addressing

The 32 X Current Switches shown on FS3-2, are 35-064 daughter-boards. The four boards are positioned in locations 10-11, 12-13, 20-21, and 22-23. The switches are selected according to the truth table provided on Table 1.

TABLE 1. X SWITCH SELECTION DATA

RAH	RAL				MNEMONIC	SWITCH	
06	00	01	02	03			
0	0	0	0	0	XS00,0	X	Switch 00
0	0	0	0	1	XS01,0	X	Switch 01
0	0	0	1	0	XS02,0	X	Switch 02
0	0	0	1	1	XS03,0	X	Switch 03
0	0	1	0	0	XS04,0	X	Switch 04
0	0	1	0	1	XS05,0	X	Switch 05
0	0	1	1	0	XS06,0	X	Switch 06
0	0	1	1	1	XS07,0	X	Switch 07
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	0	0	XS28,0	X	Switch 28
1	1	1	0	1	XS29,0	X	Switch 29
1	1	1	1	0	XS30,0	X	Switch 30
1	1	1	1	1	XS31,0	X	Switch 31

The 16 Y Current Switches, shown on FS3-3, are on 35-039 daughter-boards in locations 24, 25, 34, and 35. The switches are selected, one at a time, by the gated output from two 35-019 decoders in locations 26-27 and 36-37. The CDJ1 input to the decoders determines the pulse width and position of the Y Current pulses. (The output from each Y Current Switch is returned to -3 volts to insure back bias on the matrix diodes when the Y switches are off.) The truth table on Table 2 specifies the selection of the Y Current Switches. Note that CDJ1 must be high in each case.

TABLE 2. Y SWITCH SELECTION DATA

RAH	RAL			MNEMONIC	SWITCH	
07	04	05	06			
0	0	0	0	YS00J1	Y	Switch 00
0	0	0	1	YS01J1	Y	Switch 01
0	0	1	0	YS02J1	Y	Switch 02
0	0	1	1	YS03J1	Y	Switch 03
0	1	0	0	YS04J1	Y	Switch 04
0	1	0	1	YS05J1	Y	Switch 05
0	1	1	0	YS06J1	Y	Switch 06
0	1	1	1	YS07J1	Y	Switch 07
1	0	0	0	YS08J1	Y	Switch 08
1	0	0	1	YS09J1	Y	Switch 09
1	0	1	0	YS10J1	Y	Switch 10
1	0	1	1	YS11J1	Y	Switch 11
1	1	0	0	YS12J1	Y	Switch 12
1	1	0	1	YS13J1	Y	Switch 13
1	1	1	0	YS14J1	Y	Switch 14
1	1	1	1	YS15J1	Y	Switch 15

The strobe select logic, shown on FS3-4, is performed on the Switch board by a 35-064 daughter-board in location 44-45. The outputs from this decoder define which transformer board is to be interrogated, and which set of 16 read-out amplifiers on that board is to be strobed. RAL07 determines which set of amplifiers, either the "A" set or the "B" set, on any transformer board is selected. RAH 4 and 5 determine which transformer board is selected as shown on Table 3.

The 16 X 32 diode matrix, shown on FS3-6 and located on each transformer board, is driven by the Y Current Switches and the X Current Switches. A current limiting resistor is in series with each Y Current load. One word line in series with each of the 512 diodes is strung through the transformer array as indicated in Figure 4. Note that with one Y Switch and one X Switch turned on, only one diode is forward biased, allowing current to flow in one unique word line.

The Strobe Amplifier on the ROM Transformer board is a 35-020 daughter-board in location 06 (FS3-4). STRB$\not0$ is the signal that determines the strobe width and position for either the "A" set of amplifiers, or the "B" set of amplifiers, FIRD$\not0$A or FIRD$\not0$B, respectively. STRB$\not0$ is active every time either CLKA$\not1$ or CLKB$\not1$ is active.

The Read Out Amplifiers on the ROM transformer board are contained on four 35-062 daughter-boards located in positions 46-47, 36-37, 26-27, and 16-17. Each amplifier has an input from both the A and B set of sense windings. The active set is selected by CLKA$\not1$ or CLKB$\not1$. Each readout amplifier provides a double rail output to the RD register in the Processor. See Figure 5.

5. MAINTENANCE

Clock timing adjustments which may affect the ROM are provided in the Processor section of this manual. The following list provides suggested solutions to possible ROM problems. In each case a fault indication is followed by possible causes of that indication.

5.1 All Zeros Read Out For All Addresses

1. No current in word lines. Caused by missing CD$\not1$ pulse.

2. Timing misadjusted. (See Processor Section for timing procedure.)

TABLE 3. ROW SELECTION DATA

RAH		RAL	MNEMONIC	ACTION
04	05	07		
0	0	0	F1RD$\not0$A	Select A side first 1K ROM
0	0	1	F1RD$\not0$B	Select B side first 1K ROM
0	1	0	F2RD$\not0$A	Select A side second 1K ROM
0	1	1	F2RD$\not0$B	Select B side second 1K ROM

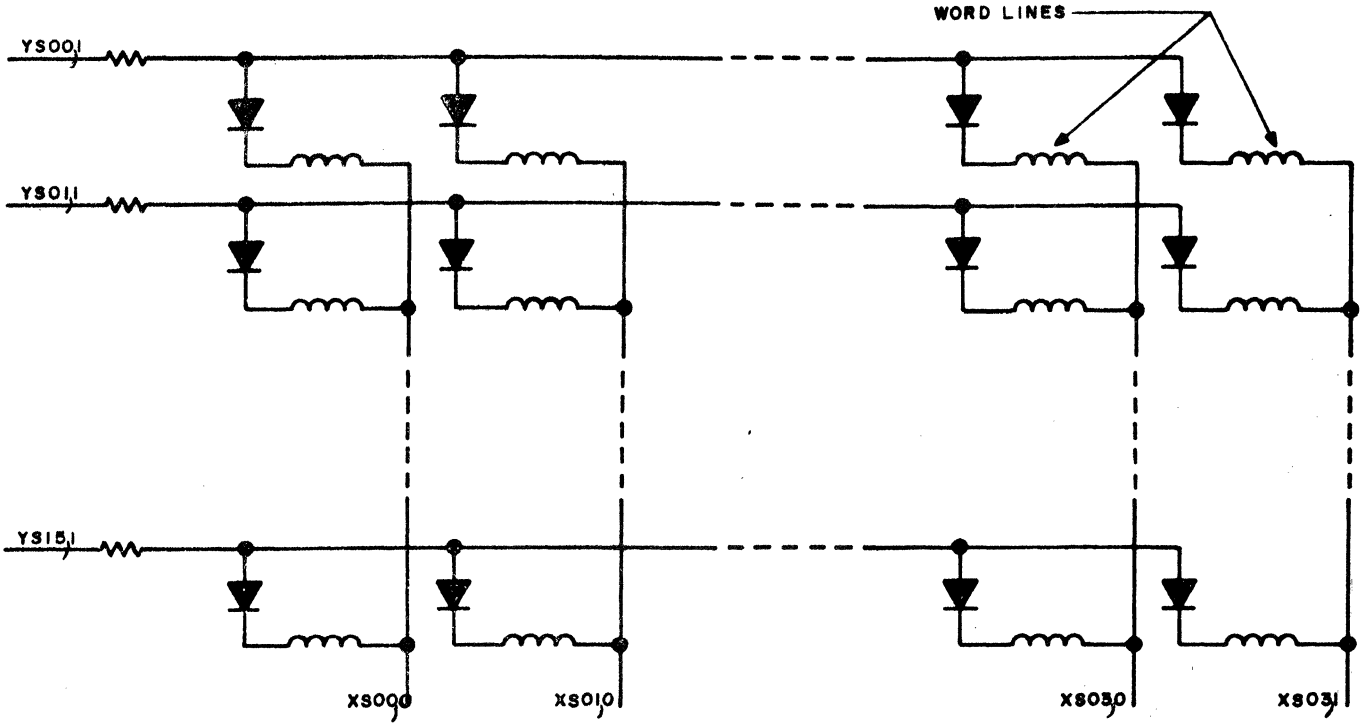


Figure 4. 16 X 32 Diode Matrix Layout

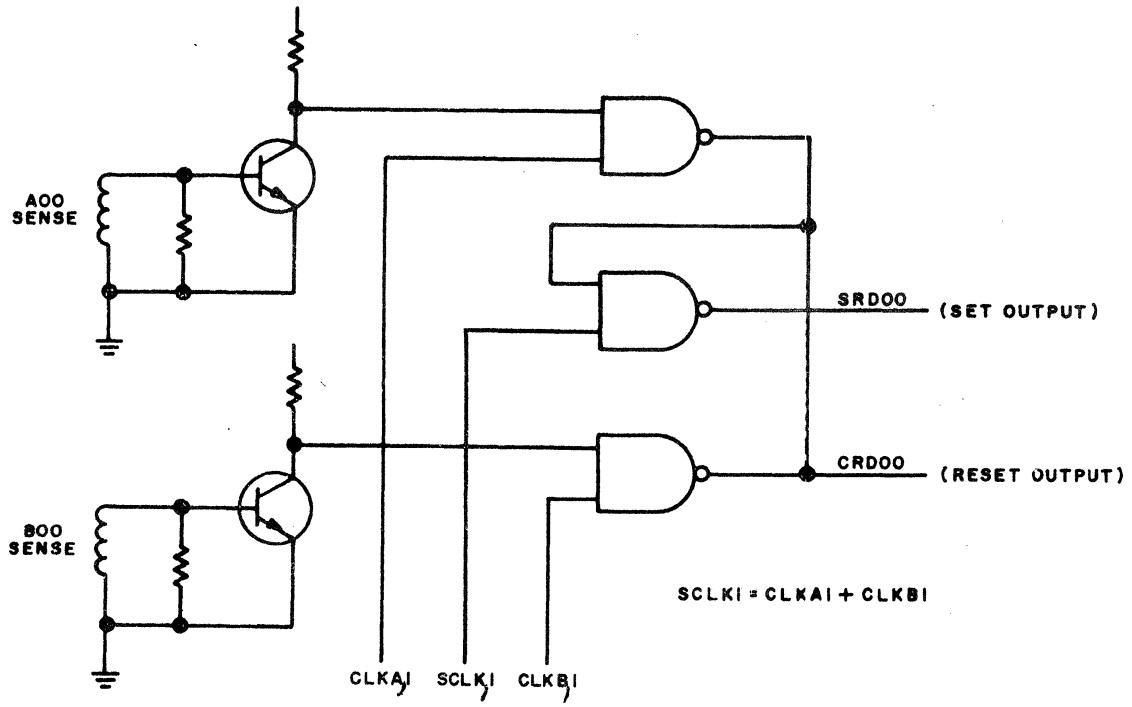


Figure 5. Read Amplifier Logic Diagram

5.2 All Ones Read Out For All Addresses

1. Grossly misadjusted timing. Readjust STRBØ and CDJ1.
2. Missing FIRDØA and FIRDØB control leads.

5.3 All Zeros Read Out For Two Adjacent Addresses (For Example - Hex Address 000 and 001, or, 236 and 237), But Otherwise Readout Correct

1. Broken word line.
2. Open diode in matrix.

5.4 All Zeros Read Out For All Addresses Having RAH06 And RAL4:6 In One Common State

1. Bad Y Current Switch.

5.5 All Zeros Read Out For All Addresses Having RAH07 And RAL0:63 In One Common State

1. Bad X Current Switch.

5.6 Consistent Failure Of One Bit For All Even Addresses

1. Bad Read Out Amplifier, A side.
2. Open Sense Winding, A side.

5.7 Consistent Failure Of One Bit For All Odd Addresses

1. Bad Read Out Amplifier, B side.
2. Open Sense Winding, B side.

5.8 Nonsensical Readout

1. Shorted diode in matrix.
2. More than one X Current Switch closed at a time.
3. More than one Y Current Switch closed at a time.
4. Timing incorrectly adjusted.

6. ROM MNEMONICS

The following list provides a brief description of each signal mnemonic found in the ROM. The source on FS3 of each signal is also provided.

MNEMONIC	MEANING	FS3 LOCATION
A00N through A15N and A00P through A15P	A Row sense winding outputs	5A1
B00N through B15N and B00P through B15P	B Row sense winding outputs	5A2

MNEMONIC	MEANING	FS3 LOCATION
CD,1	Current Drive Pulse. Derived from CD0 from the Processor.	1M6
CLKA,1	Clock to Row A Read Out Amplifiers	4P6
CLKB,1	Clock to Row B Read Out Amplifiers	4P8
CRD00 through CRD15	Clear ROM Data Register signals to the Processor	5F2
EN,1	Enable signal for Strobe singals	1M7
F1RD,0A	Decoded Strobe Enable RMT1-Row A	4P2
F1RD,0B	Decoded Strobe Enable RMT1-Row B	4P2
F2RD,0A	Decoded Strobe Enable RMT2-Row A	4P3
F2RD,0B	Decoded Strobe Enable RMT2-Row B	4P3
N15	Negative 15 volt supply voltage	
P5	Positive 5 volt supply voltage	
RAH04,0 through RAH07,0	ROM Address High Register inputs from the Processor. RAH4 and 5 select the RMT board if over 1024 ROM words are required. RAH6 and 7 help select the X and Y Switch respectively.	1J1
RAL00,0 through RAL07,0	ROM Address Low Register inputs from the Processor. RAL0:3 with RAH6 determine the X Switch. RAL4:6 with RAH7 determine the Y Switch. RAL7 selects Row A or Row B.	1A1
SRD00 through SRD15	Set ROM Data Register signals to the Processor.	5F1

MNEMONIC	MEANING	FS3 LOCATION
STRB1	Strobe signal. Derived from STRB0 from the Processor.	1M5
XS00 through XS31	Decoded X Switch outputs to 16 X 32 matrix.	2E1
YS00 through YS15	Decoded Y Switch outputs to 16 X 32 matrix.	3N1

MODEL 30-2
READ-ONLY-MEMORY

CONTENTS

ILLUSTRATIONS

THE READ-ONLY-MEMORY

1. INTRODUCTION

The Read-Only-Memory (ROM) is a high speed, non-destructive memory used to store the micro-program. As described in the General Description, the micro-program interprets and executes the user program. A basic 30-2 Digital System must use a Decoder ROM (DROM) module and may have an additional ROM module. The DROM module provides up to 512 16-bit words of micro-programming and up to 128 12-bit words of DROM data. The ROM module provides up to 1024 additional 16-bit words for expanded micro-programs. The cycle time of the GE-PAC 30 DROM and expansion ROM is approximately 400 nanoseconds. Functional schematics (FS 38, 39, and 40) referenced in this Section are provided in Volume 2 of this manual. In this Section, 'ROM' is used to refer to the ROM section of the DROM Module and the Expansion ROM, and 'DROM' is used to refer to the 12-bit data section of the DROM Module.

2. BASIC THEORY OF OPERATION

The basic memory cell in the DROM/ROM is a pulse transformer with a U shaped core made of a linear ferrite material (see Figure 1). A multi-turn secondary, called the sense winding, is wound on the U core. The primary circuit is made up of many singularly-selectable, one-turn windings which either pass through, or by-pass the U core. Each primary winding is called a word line.

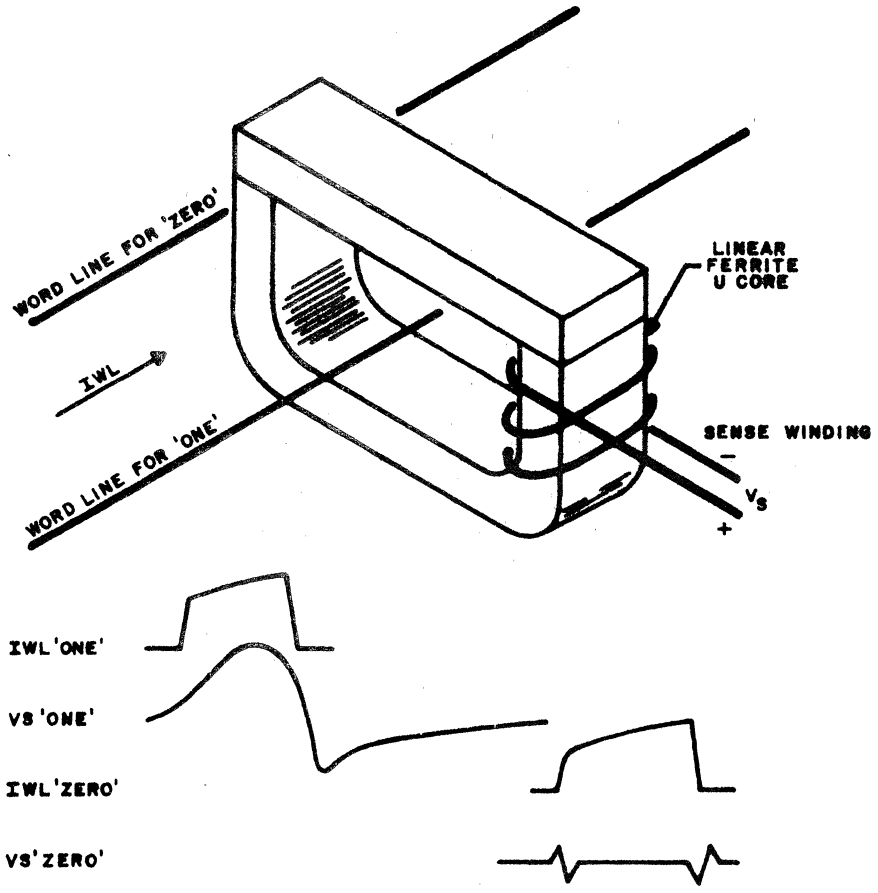


Figure 1. Basic U Core Layout

When a current pulse is developed on a word line which passes through the U core, a voltage appears across the sense winding. The word line is said to store a ONE in that transformer. Similarly, if a current pulse is developed on a word line that does not pass through the U core, no voltage appears across the sense winding. The word line is said to store a ZERO in that transformer.

Figure 2 is a simplified diagram of DROM/ROM showing one word line and six U cores. The bit outputs are indicated on Figure 2 for the word line shown. In the GE-PAC 30 DROM, each DROM word line is strung through 12 transformers. Each word line stores 12 binary bits. In the GE-PAC 30 ROM, each ROM word line is strung through 32 transformers. Thus, each ROM word line stores 32 binary bits. A strobe selection system selects one 16-bit word from the 32 bits addressed.

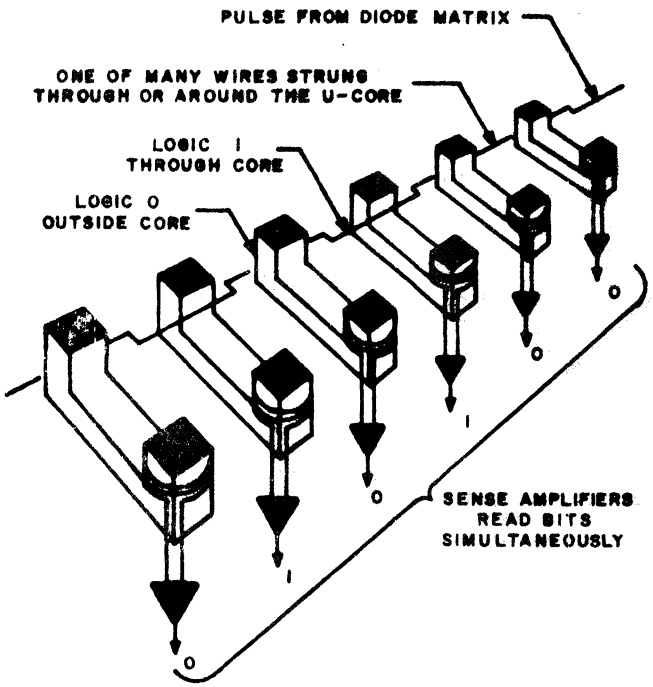


Figure 2. Basic ROM Layout

3. BLOCK DIAGRAM ANALYSIS

The basic GE-PAC 30-2, 512 word ROM is contained on two mother-boards: the ROM Switch (RMS) and the Decoder ROM Transformer (DRMT) boards. Refer to the block diagram, Figure 3. Address decoding is performed on the switch board. An 8 X 32 diode matrix is driven by the switch board. With one word line per diode, there are a total of 256 word lines. Each word line is strung through 32 U core transformers. Thus, each word line stores 32 bits, or two 16-bit ROM words. One 16-bit word is selected by strobing one of the two sets of 16 readout amplifiers. The selected 16 outputs are sent to the Processor double-rail, hence the 32 output lines. Two transformer boards may be driven in parallel if an additional 1024 words of ROM are needed.

The Decoder ROM Transformer board (DRMT) also contains the DROM Transformer and readout amplifiers. The switch board also does address decoding for the DROM. An 8 X 16 diode matrix on the DRMT is driven by the switch board. With one DROM word line per diode, there are a total of 128 word lines. Each DROM word line is strung through 12 U core transformers. Each DROM word line stores 12 bits. The output from the readout amplifiers is sent to the ROM Interface single-rail.

4. FUNCTIONAL DESCRIPTIONS

The following paragraphs provide functional descriptions which reference the FS38, FS39 and FS40 schematics included in Volume 2 of this manual.

Figure 4 shows the significance of the ROM address (RAH and RAL) bits. For example, RAH4 and 5 specify a transformer board if the expansion ROM (RMT) is used. RAH 6 and RAL 0:3 specify the X Switch, etc.

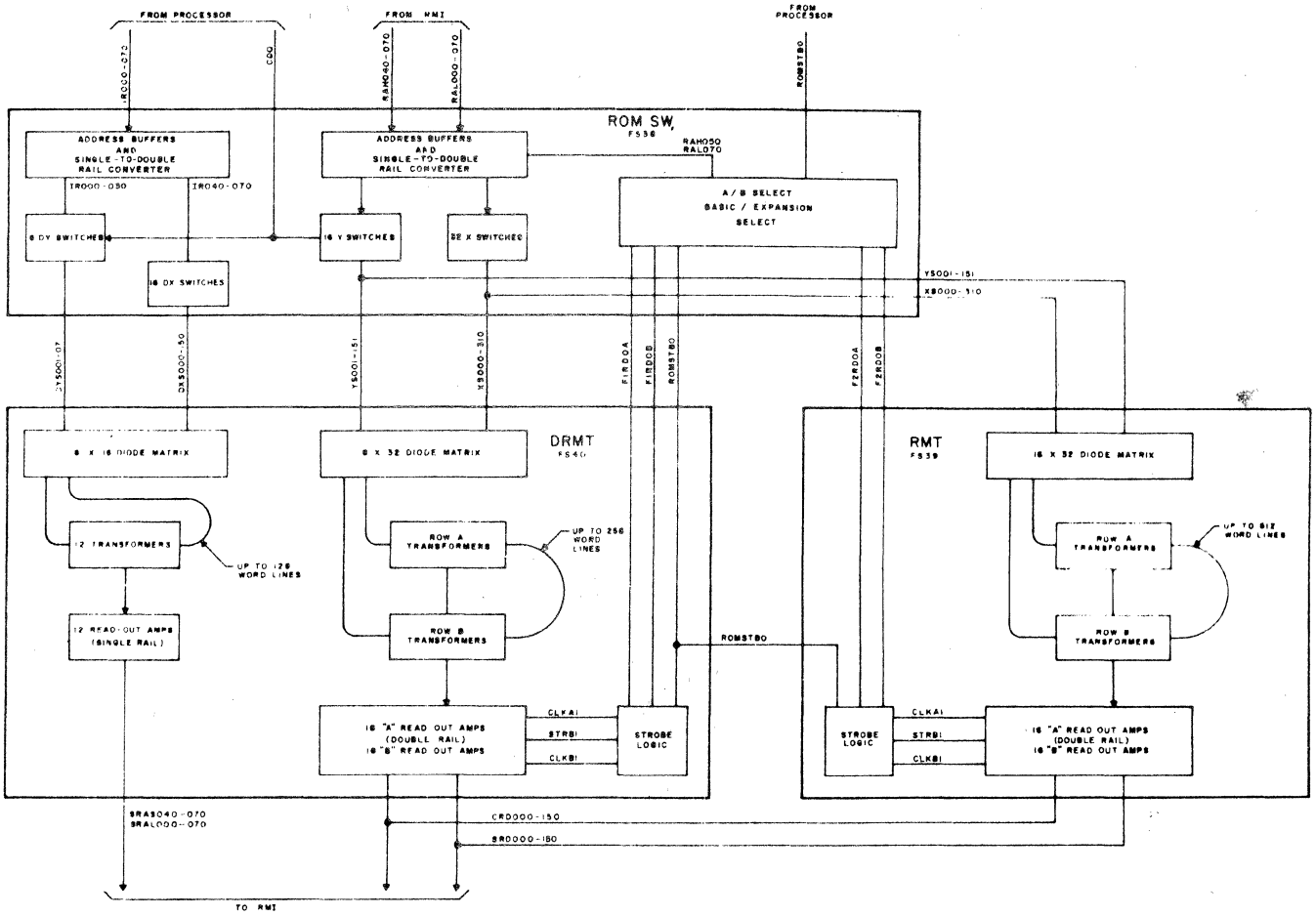


Figure 3. Block Diagram ROM

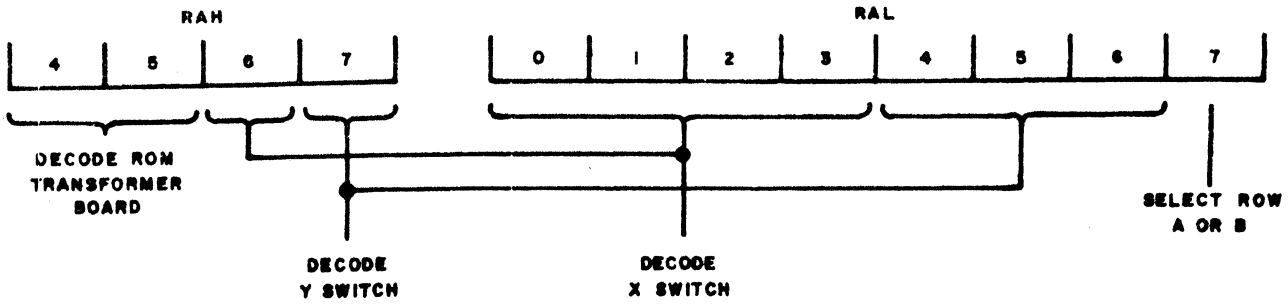


Figure 4. ROM Addressing

Refer to FS38-1. The ROM address buffers and single-to-double rail converters are located on the ROM switch board.

The 32 X Current Switches shown on FS38-2, 3, 4, and 5 are 19-006 integrated circuits. The IC's are positioned in locations 40, 41, 42, 43, 30, and 31. The switches are selected according to the truth table provided on Table 1.

TABLE 1. X SWITCH SELECTION DATA

RAH	RAL				MNEMONIC	SWITCH	
06	00	01	02	03			
0	0	0	0	0	XS00,0	X	Switch 00
0	0	0	0	1	XS01,0	X	Switch 01
0	0	0	1	0	XS02,0	X	Switch 02
0	0	0	1	1	XS03,0	X	Switch 03
0	0	1	0	0	XS04,0	X	Switch 04
0	0	1	0	1	XS05,0	X	Switch 05
0	0	1	1	0	XS06,0	X	Switch 06
0	0	1	1	1	XS07,0	X	Switch 07
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	0	0	XS28,0	X	Switch 28
1	1	1	0	1	XS29,0	X	Switch 29
1	1	1	1	0	XS30,0	X	Switch 30
1	1	1	1	1	XS31,0	X	Switch 31

The 16 Y Current Switches, shown on FS38-6 and 7 are on 35-039 daughter-boards in locations 10, 11, 12, and 13. The switches are selected one at a time, by the gated output from the 19-005 IC decoders in locations 20, 21, 22, and 23. The CD1 input to the decoders determines the pulse width and position of the Y Current pulses. (The output from each Y Current Switch is returned to -3 volts to insure back bias on the matrix diodes when the Y Switches are off.) The truth table on Table 2 specifies the selection of the Y Current Switches. Note that CD1 must be high in each case.

TABLE 2. Y SWITCH SELECTION DATA

RAH	RAL			MNEMONIC	SWITCH	
07	04	05	06			
0	0	0	0	YS001	Y	Switch 00
0	0	0	1	YS011	Y	Switch 01
0	0	1	0	YS021	Y	Switch 02
0	0	1	1	YS031	Y	Switch 03
0	1	0	0	YS041	Y	Switch 04
0	1	0	1	YS051	Y	Switch 05
0	1	1	0	YS061	Y	Switch 06
0	1	1	1	YS071	Y	Switch 07
1	0	0	0	YS081	Y	Switch 08
1	0	0	1	YS091	Y	Switch 09
1	0	1	0	YS101	Y	Switch 10
1	0	1	1	YS111	Y	Switch 11
1	1	0	0	YS121	Y	Switch 12
1	1	0	1	YS131	Y	Switch 13
1	1	1	0	YS141	Y	Switch 14
1	1	1	1	YS151	Y	Switch 15

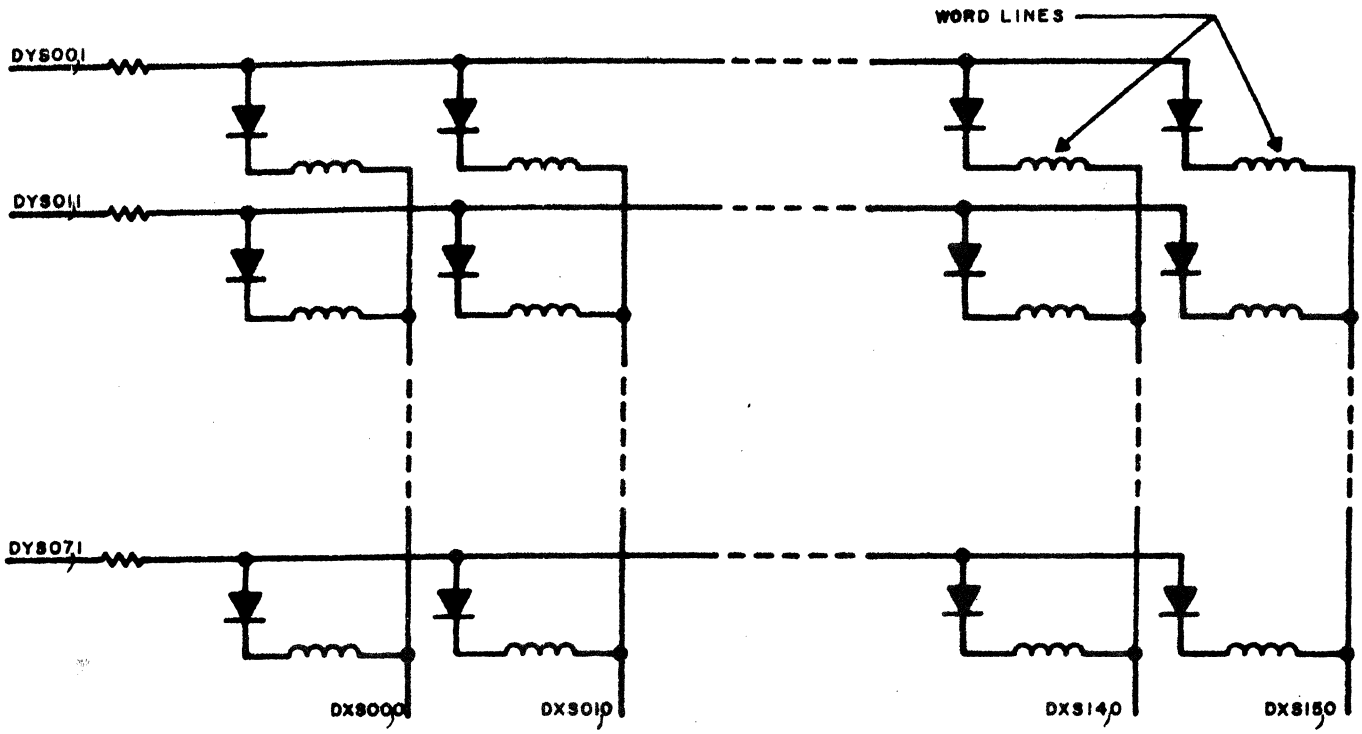


Figure 5. ROM Diode Matrix Layout

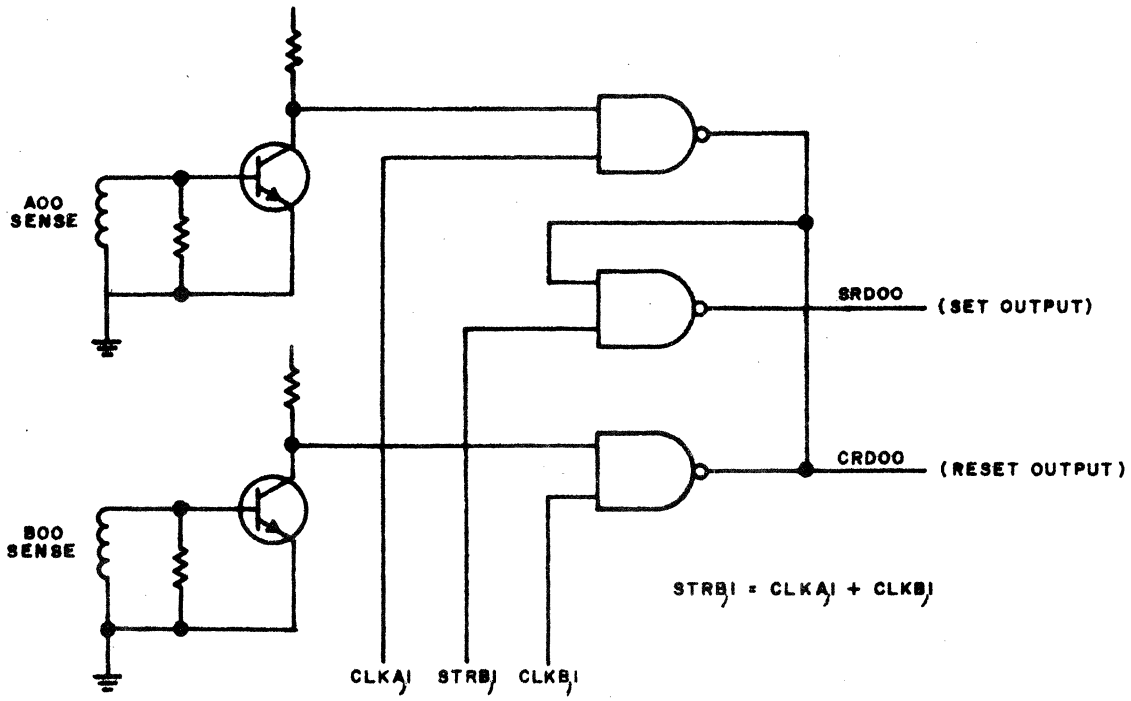


Figure 6. Read Amplifier Logic Diagram

The strobe select logic, shown on FS38-11, is performed on the switch board by 19-003 IC's in location 25. The output from this decoder define which transformer board is to be interrogated, and which set of 16 readout amplifiers on that board is to be strobed. RAL 07 determines which set of amplifiers, either the "A" set or the "B" set, on any transformer board is selected. RAH 4 and 5 determine which transformer board is selected as shown on Table 3.

The 8 X 32 diode matrix, shown on FS40-1, and the 16 X 32 diode matrix, shown on FS39-1, are driven by the Y Current Switches and the X Current Switches. A current limiting resistor is in series with each Y Current load. One word line in series with each of the 256 or 512 diodes is strung through the transformer array as indicated on Figure 5. Note that with one Y Switch and one X Switch turned on, only one diode is forward biased, allowing current to flow in one unique word line in each DROM/ROM transformer board.

The Strobe Amplifier on the Transformer board is a daughter-board in location 06 (FS40-5 and FS39-3). ROMSTB0 is the signal that determines the strobe width and position for either the "A" set of amplifiers, or the "B" set of amplifiers, F1RD0A or F1RD0B, respectively. STRB1 is active every time either CLKA1 or CLKB1 is active.

The ROM Read Out Amplifiers on the transformer board are contained on four 35-062 daughter-boards located in positions 46-47, 36-37, 26-27, and 16-17. Each amplifier has an input from both the A and B set of sense windings. The active set is selected by CLKA1 or CLKB1. Each readout amplifier provides a double-rail output to the RD register on the RMI (FS44). See Figure 6.

Figure 7 shows the significance of the Instruction Register bits. IR bits 0:3 select a DY Switch; IR bits 4:7 select a DX Switch.

Refer to FS38-8 and 10. The IR buffers and single-to-double rail converters are located on the ROM switch board.

INSTRUCTION REGISTER	
0 1 2 3	4 5 6 7
DECODE DY SWITCH	DECODE DX SWITCH

Figure 7. DROM Addressing

The 16 DX Current Switches shown on FS38-8 and 9 are 19-006 Integrated Cricuits. The IC's are positioned in locations 34, 35, 36, and 37. The switches are selected according to the truth table provided on Table 4.

TABLE 3. ROW SELECTION DATA

RAH		RAL		
04	05	07	MNEMONIC	ACTION
0	0	0	F1RD0A	Select A side, basic ROM
0	0	1	F1RD0B	Select B side, basic ROM
0	1	0	F2RD0A	Select A side second 1K ROM
0	1	1	F2RD0B	Select B side second 1K ROM

TABLE 4. DX SWITCH SELECTION DATA

IR				MNEMONIC	SWITCH
04	05	06	07		
0	0	0	0	DXS00,0	Decoder X Switch 00
0	0	0	1	DXS01,0	Decoder X Switch 01
0	0	1	0	DXS02,0	Decoder X Switch 02
0	0	1	1	DXS03,0	Decoder X Switch 03
0	1	0	0	DXS04,0	Decoder X Switch 04
0	1	0	1	DXS05,0	Decoder X Switch 05
0	1	1	0	DXS06,0	Decoder X Switch 06
0	1	1	1	DXS07,0	Decoder X Switch 07
1	0	0	0	DXS08,0	Decoder X Switch 08
1	0	0	1	DXS09,0	Decoder X Switch 09
1	0	1	0	DXS10,0	Decoder X Switch 10
1	0	1	1	DXS11,0	Decoder X Switch 11
1	1	0	0	DXS12,0	Decoder X Switch 12
1	1	0	1	DXS13,0	Decoder X Switch 13
1	1	1	0	DXS14,0	Decoder X Switch 14
1	1	1	1	DXS15,0	Decoder X Switch 15

The 8 DY Current Switches, shown on FS38-10, are on 19-039 daughter-boards in locations 05 and 06. The switches are selected, one at a time, by the gated output from 19-005 IC decoders in locations 15 and 16. The CD1 input to the decoders determines the pulse width and position of the DY Current pulses. The output from each Y Current Switch is returned to -3 volts to insure back bias on the matrix diodes when the DY Switches decoders are enabled by the DC lead ENDROM,0 which goes low when a DROM readout is desired. The truth table on Table 5 specifies the selection of the DY Current Switches. Note that CD,0 and ENDROM,0 must be active in each case. Note also, that not all combinations can be selected.

The 8 X 16 diode matrix, shown on FS40-2, is driven by the DY Switches and the DX Switches. A current limiting resistor is in series with each DY Current load. One word line in series with each of the 128 diodes is strung through the Decoder transformer array as indicated on Figure 8. Note that with one DY Switch and on DX Switch turned on, only one diode is forward biased, allowing current to flow in one unique word line.

The DROM Read Out Amplifiers on the DROM transformer board are fabricated with discrete components located near the 0 connector. Each amplifier has an input from a DROM sense winding. Each readout amplifier provides a single rail output to the RAS and RAL registers on the RMI (FS44).

Tables 6 and 7 provide a summary of the ROM and DROM addressing scheme.

TABLE 5. DY SWITCH SELECTION DATA

IR				MNEMONIC	SWITCH
0	1	2	3		
0	0	0	0	DYS00,1	Decoder Y Switch 00
0	0	1	0	DYS01,1	Decoder Y Switch 01
0	1	0	0	DYS02,1	Decoder Y Switch 02
0	1	1	0	DYS03,1	Decoder Y Switch 03
1	0	0	1	DYS04,1	Decoder Y Switch 04
1	1	0	0	DYS05,1	Decoder Y Switch 05
1	1	0	1	DYS06,1	Decoder Y Switch 06
1	1	1	0	DYS07,1	Decoder Y Switch 07

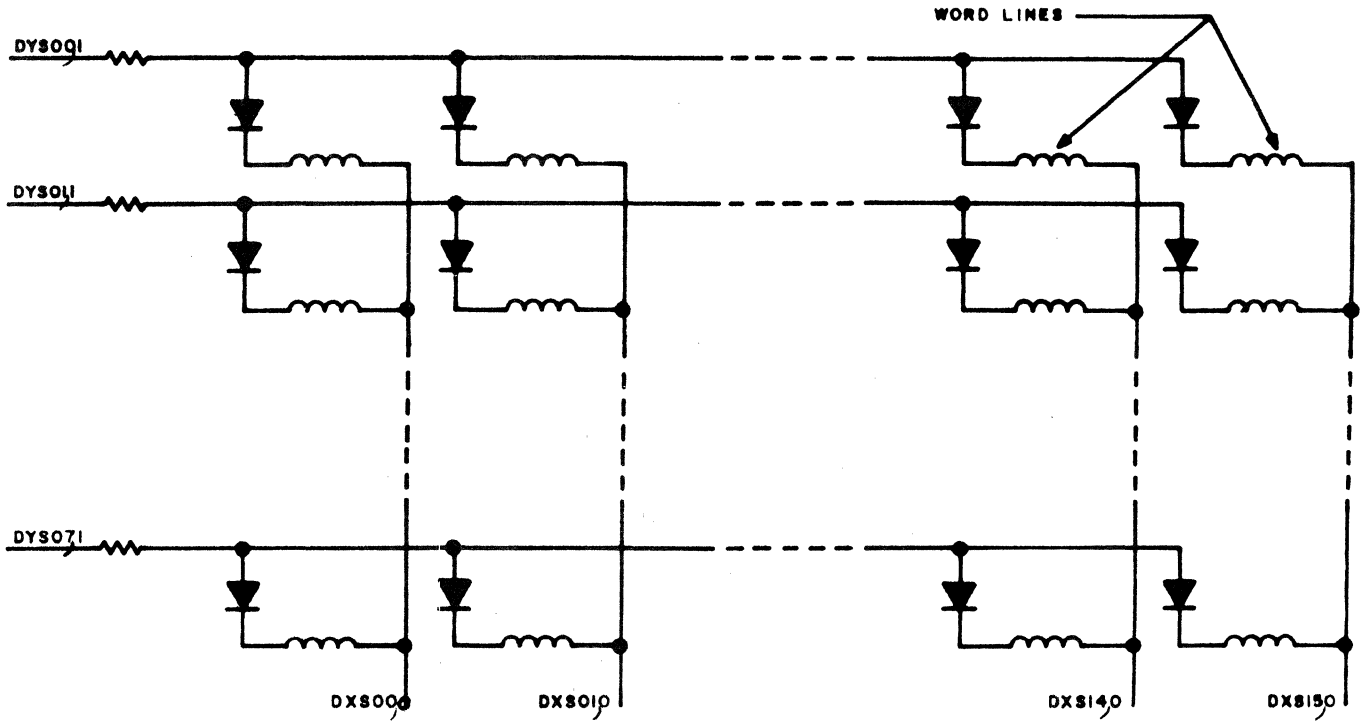


Figure 8. DROM Diode Matrix Layout

TABLE 6. ROM ADDRESSING SUMMARY

RAN				RAL									
04	05	06	07	00	01	02	03	04	05	06	07		
	0										0	FIRDA	BASIC ROM
	0										1	FIRDB	
	1										0	F2RDA	EXPANSION ROM
	1										1	F2RDB	
		0		0	0	0	0					XS00	X SWITCHES
		0		0	0	0	1					XS01	
		0		0	0	1	0					XS02	
		0		0	0	1	1					XS03	
		0		0	1	0	0					XS04	
		0		0	1	0	1					XS05	
		0		0	1	1	0					XS06	
		0		0	1	1	1					XS07	
		0		1	0	0	0					XS08	
		0		1	0	0	1					XS09	
		0		1	0	1	0					XS10	
		0		1	0	1	1					XS11	
		0		1	1	0	0					XS12	
		0		1	1	0	1					XS13	
		0		1	1	1	0					XS14	
		0		1	1	1	1					XS15	
		1		0	0	0	0					XS16	
		1		0	0	0	1					XS17	
		1		0	0	1	0					XS18	
		1		0	0	1	1					XS19	
		1		0	1	0	0					XS20	
		1		0	1	0	1					XS21	
		1		0	1	1	0					XS22	
		1		0	1	1	1					XS23	
		1		1	0	0	0					XS24	
		1		1	0	0	1					XS25	
		1		1	0	1	0					XS26	
		1		1	0	1	1					XS27	
		1		1	1	0	0					XS28	
		1		1	1	0	1					XS29	
		1		1	1	1	0					XS30	
		1		1	1	1	1					XS31	
			0					0	0	0		YS00	Y SWITCHES
			0					0	0	1		YS01	
			0					0	1	0		YS02	
			0					0	1	1		YS03	
			0					1	0	0		YS04	
			0					1	0	1		YS05	
			0					1	1	1		YS06	
			0					1	1	1		YS07	
			1					0	0	0		YS08	
			1					0	0	1		YS09	
			1					0	1	0		YS10	
			1					0	1	1		YS11	
			1					1	0	0		YS12	
			1					1	0	1		YS13	
			1					1	1	0		YS14	
			1					1	1	1		YS15	

TABLE 7. DROM SELECTION DATA

INSTRUCTION REGISTER								MNEMONIC
0	1	2	3	4	5	6	7	
0	0	0	0					DYS001
0	0	1	0					DYS011
0	1	0	0					DYS021
0	1	1	0					DYS031
1	0	0	1					DYS041
1	1	0	0					DYS051
1	1	0	1					DYS061
1	1	1	0					DYS071
				0	0	0	0	DXS000
				0	0	0	1	DXS010
				0	0	1	0	DXS020
				0	0	1	1	DXS030
				0	1	0	0	DXS040
				0	1	0	1	DXS050
				0	1	1	0	DXS060
				0	1	1	1	DXS070
				1	0	0	0	DXS080
				1	0	0	1	DXS090
				1	0	1	0	DXS100
				1	0	1	1	DXS110
				1	1	0	0	DXS120
				1	1	0	1	DXS130
				1	1	1	0	DXS140
				1	1	1	1	DXS150

5. MAINTENANCE

Clock timing adjustments, which may affect the ROM, are provided in the Processor Section of this manual. The following list provides suggested solutions to possible DROM/ROM problems. In each case, a fault indication is followed by possible causes of that indication.

5.1 All Zeros Read Out For All Addresses

1. No current in word lines. Caused by missing CD0 pulse.
2. Timing misadjusted (See Processor Section for timing procedure).

5.2 All Ones Read Out For All ROM Addresses

1. Grossly misadjusted timing. Readjust ROMSTB0 and CD0.

5.3 All Zeros Read Out For Two Adjacent ROM Addresses (For Example - Hex Address 000 and 001, or, 236 and 237), But Otherwise Readout Correct

1. Broken word line.
2. Open diode in matrix.

5.4 All Zeros Read Out For All Addresses Having RAH06 and RAL4:6 In One Common State

1. Bad Y Current Switch.

5.5 All Zeros Read Out For All Addresses Having RAH07 and RAL0:63 in One Common State

1. Bad X Current Switch

5.6 Consistent Failure Of One Bit For All Even ROM Addresses

1. Bad Read Out Amplifier, A side.
2. Open Sense Winding, A side.

5.7 Consistent Failure Of One Bit For All Odd ROM Address

1. Bad Read Out Amplifier, B side.
2. Open Sense Winding, B side.

5.8 Consistent Failure Of One Bit For All DROM Addresses

1. Bad Read Out Amplifier
2. Open Sense Winding.

5.9 Nonsensical Readout

1. Shorted diode in matrix.
2. More than one X Current Switch closed at a time.
3. More than one Y Current Switch closed at a time.
4. Timing incorrectly adjusted.

6. ROM MNEMONICS

The following list provides a brief description of each signal mnemonic found in the ROM. The source of each signal is also provided.

MNEMONIC	MEANING	LOCATION
A00N through A15N and A00P through A15P	A Row sense winding outputs	FS40-4A1, FS39-2A1
B00N through B15N and B00P through B15P	B Row sense winding outputs	FS40-4A2, FS39-2A2
CD1	Current Drive Pulse. Derived from CD0 from the Processor.	FS38-6D9

MNEMONIC	MEANING	LOCATION
CLKA/1	Clock to Row A Read Out Amplifiers.	FS40-5L7, FS39-3L7
CLKB/1	Clock to Row B Read Out Amplifiers.	FS40-5L5, FS39-3L5
CRD00 through CRD15	Clear ROM Data Register signals to the RMI.	FS40-4F2, FS39-2F2
DXS00/0 through DXS15/0	Decoder X Switch outputs to 8 X 16 matrix	FS38-8R1
DYS00/1 through DYS07/1	Decoder Y Switch output to 8 X 16 matrix	FS38-DR1
ENDROM/0	Enable lead to Decoder Y Switch selection	FS39-10R9
F1RDA/0	Decoded Strobe Enable RMT1-Row A	FS38-11P2
F1RDB/0	Decoded Strobe Enable RMT1-Row B	FS38-11P3
F2RDA/0	Decoded Strobe Enable RMT2-Row A	FS38-11P4
F2RDB/0	Decoded Strobe Enable RMT2-Row B	FS38-11P5
IR00/1 through IR07/1	Instruction Register inputs from the Processor. IR0:3 select a DY Switch. IR4:7 select a DX Switch.	FS38-8B1
N15	Negative 15 volt supply voltage	FS38-6R9
P5	Positive 5 volt supply voltage	

MNEMONIC	MEANING	LOCATION
RAH04/0 through RAH07/0	ROM Address High Register inputs from the RMI. RAH4 and 5 select the RMT board if 1024 extra ROM words are required. RAH6 and 7 help select the X and Y Switch respectively.	FS38-1L2
RAL00/0 through RAL07/0	ROM Address Low Register inputs from the RMI. RAL0:3 with RAH6 determine the X Switch. RAL4:6 with RAH7 determine the Y Switch. RAL7 selects Row A or Row B	FS38-1A2
SRAH04/0 through SRAH07/0	Set RAH signals to the RMI	FS40-3B1
SRAL00/0 through SRAL07/0	Set RAL signals to the RMI	FS40-3B1
SRD00 through SRD15	Set ROM Data Register signals to the RMI	FS40-4F2, FS39-2F2
STRB/1	Strobe signal. Derived from ROMSTB/0 from the Processor.	FS40-5K2, FS39-3K2
XS00/0 through XS31/0	Decoded X Switch outputs to 16 X 32 matrix.	FS38-2C9
YS00/1 through YS15/1	Decoded Y Switch outputs to 16 X 32 matrix.	FS38-6P1

THE INPUT/OUTPUT SYSTEM

TABLE OF CONTENTS

ILLUSTRATIONS

THE INPUT/OUTPUT SYSTEM

1. INTRODUCTION

Several methods of communication between the Processor and peripheral devices or systems are provided. The methods vary in speed, sophisitcation, and the amount of attention from the Processor required. Thus, the Input/Output (I/O) System may be tailored to communicate efficiently with the types of peripheral devices presently used with a particular system, and later may be simply expanded in the field to meet changing I/O requirements. This Section describes the basic I/O System, the Multiplexor Channel, and the most common peripheral Device Controller. For a general discussion of all I/O capabilities, see the Systems Interface Manual, Publication Number 29-003.

2. MULTIPLEXOR CHANNEL

Figure 1 is a block diagram of the Multiplexor Channel. The Multiplexor Channel is a byte oriented I/O system which communicates directly with up to 256 peripheral devices. The Multiplexor Bus consists of 27 lines; 8 data input, 8 data output, 8 control lines, 2 test lines, and an initialize line. The two test input lines from the Device Controllers are Synchronization (SYN) and Attention (ATN). The final line is System Clear (SCLR) to all Device Controllers. Each of the groups of lines is briefly described in the following paragraphs.

2.1 Data Available Lines (DAL's)

There are eight unidirectional DAL lines (DAL000:DAL070). Access from the Processor to the interface is accomplished via the DAL's for addressing an I/O Device Controller, for the output of data and for commands.

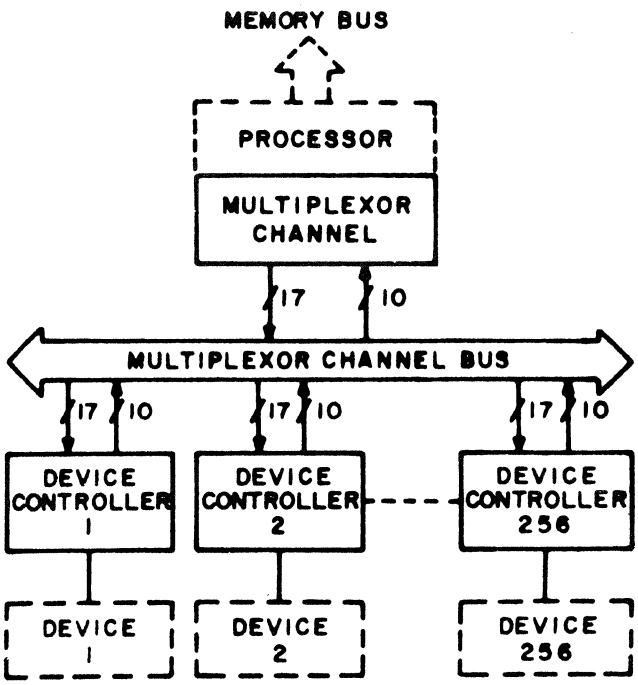


Figure 1. Multiplexor Channel, Block Diagram

2.2 Data Request Lines (DRL's)

There are eight unidirectional DRL lines (DRL000:DRL070). Access from the interface to the Processor is accomplished via the DRL's for receiving status bytes, data bits, or the address of a device controller which has generated an interrupt.

2.3 Control Lines

There are eight unidirectional control lines originating from the Processor. These lines are energized on a one-out-of-eight selection to control the use of the DRL's and DAL's. The specific function of each Control Line is:

1. ADRSϕ - When this line is active, it enables the appropriate Device Controller to respond to its address, which should be present on the DAL's.
2. DAϕ - Activating this line enables a previously addressed Device Controller to accept the data which should be present on the DAL's.
3. DRϕ - This line being active enables the addressed Device Controller to send data to the Processor via the DRL's.
4. SRϕ - Activating SRϕ enables a previously addressed Device Controller to send its status back to the Processor via the DRL's.
5. CMDϕ - When CMDϕ is active, the previously addressed Device Controller is enabled to receive a command from the Processor. The command should be present on the DAL's.
6. ACKϕ - This line is different from the other control lines because it is not a shared line. Its function, when active, is to allow an enabled device to send its address back to the Processor via the DRL's for the purpose of servicing an interrupt condition.
7. PPFϕ - CL07ϕ - On Processors equipped with the Primary Power Fail option (PPF), this line is active when the PPF circuit detects a loss of AC line power. Device controllers which must initiate a power-down sequence have 3 ms minimum before the logic power drops off.

The remaining control line CL06ϕ has no fixed functional assignment in the present system configuration.

2.4 System Synchronize (SYNϕ), Interrupt/Attention (ATNϕ) and System Clear (SCLRϕ) Lines

These are three shared unidirectional lines. The state of the SYNϕ and ATNϕ lines is tested by the Processor during program or interrupt control of the Device Controllers. The SCLRϕ line provides an Initialize signal to set up preferred states in the Device Controllers.

2.5 Sequence of Operations

The Multiplexor is a false bus, i.e., logical ONE is ground and logical ZERO is +5v. A typical sequence of operations over the Multiplexor Channel is:

1. The Processor sends the 8-bit address of a Device Controller on the DAL lines. The address appears on the bus to all Device Controllers.
2. The Processor then activates the ADRSϕ control line to specify to all Device Controllers that the DAL lines now provide an address (rather than data).

3. The Device Controllers use the ADRSØ bit to enable address decoders. Each Device Controller decodes its own address. Assuming that the DAL lines present the address of one of the Device Controllers tied to the Multiplexor Channel, the Device Controller decodes its address, sets its Address flip-flop, and responds by sending a SYN signal back to the Multiplexor Channel.

4. The Processor now removes the ADRSØ signal, and then the DAL signal. The Device Controller remains addressed until another Device Controller is addressed, or until a System Clear (SCLR) signal is received.

5. The Processor next activates the SRØ control line, and the Device Controller sends its status on the DRL lines, and a SYN signal.

6. If the status indicates that the Device Controller is capable of data transfer, the Processor proceeds.

7. For an output operation, the Processor places the 8 data bits on the DAL lines and then activates the DAØ control line. The Device Controller responds with a SYN signal when it has accepted the byte.

8. In the case of an input operation, the Processor activates the DRØ control line and the Device Controller places the byte on the DRL lines and sends a SYN signal.

The sequence provided here is simplified. The entire sequence for each type of I/O instruction is listed in Chapter 2 of the Systems Interface Manual, Publication Number 29-003. The final line to be introduced here is the Attention (ATN) line. The Attention line provides a means of interrupting the Processor. Each controller normally has an interrupt Queue flip-flop which may be set by conditions within either the device or the Device Controller. The output from the Queue flip-flop is sent to the Multiplexor Channel as ATN. Note that ATN may be initiated by any Device Controller, whether it is addressed or not. The Processor initiates an acknowledge cycle to determine which Device Controller caused the ATN signal. The interrupting device automatically returns its device number to the Processor. This interrupt feature is described in more detail later in this description.

3. BASIC DEVICE CONTROLLER

Each peripheral device is interfaced with the I/O System via a Device Controller. Therefore, each peripheral device has a unique Device Controller. The Device Controller converts from the GE-PAC 30 logic levels of 0 and +5 volts to the logic levels used by the peripheral device. The Device Controller also provides all control interface required. This section provides general information applicable to most Device Controllers. Later in this description, the Teletypewriter Device Controller is described. The Display Device Controller is described in a separate section of this manual. Other Device Controllers are described in separate maintenance manuals.

The Device Controller circuits used to communicate with the I/O bus system are shown in Figure 2. In a typical case, the DAL's and Control Lines are buffered by standard gates to drive the Address, Command, Control and ATN/ACK circuits. The signals back to the Processor on the DRL's are gated by power gates whose outputs are OR tied within the Device Controller and on the bus. The load resistors for the DRL's are located in the Processor. Diode-Transistor Logic (DTL) power gates used for bus drivers located in the Processor are capable of handling 25 DTL loads in addition to a 1K pull-up resistor. The ATN,0, SYN,0 and DRL bus lines are driven by power gates distributed throughout the Device Controller boards. On each line, the gate collectors are OR tied and share a common load resistor (located in the Processor) as shown on Figure 2. The basic rules for Device Controllers tying to the I/O buses are:

1. Only one DTL load is placed on each Device Controller input line from an I/O bus (DAL's, Control Lines, and SCLR,0 line).

2. Not more than two DTL power gates are ORed at a Device Controller output line to an I/O bus (ATN, SYN, and DRL's).

The previous two rules give the Processor a basic I/O drive capability of 25 Device Controllers. Additional buffering of the bus, within the Processor, is provided by expansions as shown on Figure 3. The Bus Buffers are connected in a series to preserve the daisy-chain priority assignment.

3.1 Addressing

One function of a Device Controller is to detect its address, and connect to the I/O bus when addressed properly. Figure 4 shows a typical Device Controller address circuit. The dotted lines around groups of logic on Figure 4 delineate daughter-boards. When a Device Controller is addressed, the 8-bit address code is placed on the Data Available Lines (DAL00,0 through DAL07,0). Two Model 35-040 daughter-boards buffer the lines and provide the true and false DAL lines. The Model 35-038 daughter-boards are wired with the address code of this particular Device Controller, and the 8 coded outputs are applied to an 8-input gate, Model 35-022. Thus, the Decoded Device output (DD,1) goes true. The address control line, ADRS,1, then strobes the DD,1 line into the Address flip-flop (Model 35-001).

The Synchronize signal is returned to the Processor, during the presence of ADRS,1, via the Address Sync line ADSY,0. The Model 35-022 gate is used here as an OR gate for returning the other device command Sync lines. The set output of the Address flip-flop, called Device Enable (DENB,1), is used to gate all other I/O control lines to the Device Controller. When another device is addressed, the decoded device line, DD,1, is low, causing the ADRS,1 strobe line to reset the Address flip-flop, and disabling the controller. Capacitor C1 on the SYNC return generates a delay of approximately 200 nanoseconds to allow the gate conditions to settle on the lines.

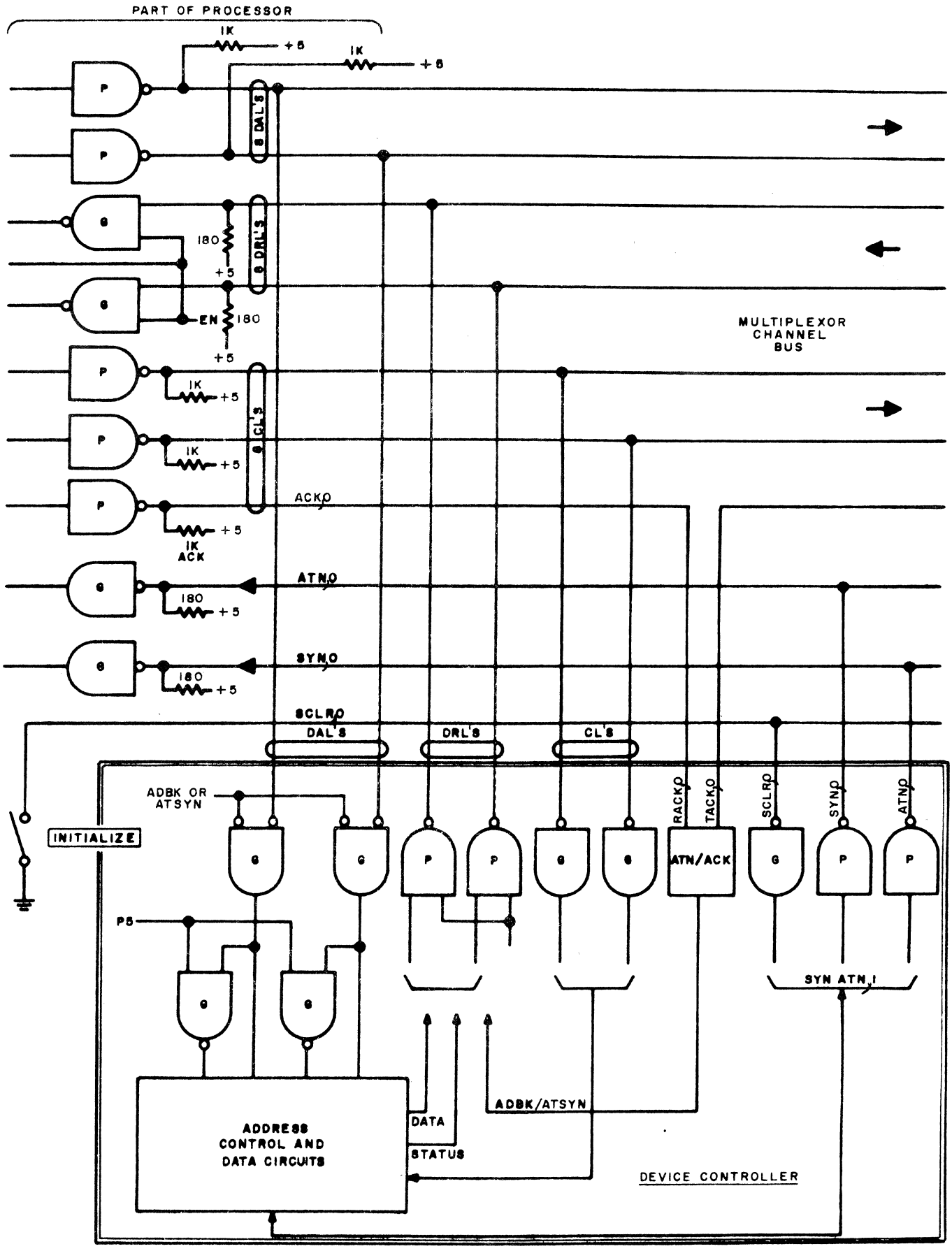


Figure 2. Multiplexor Channel, Logic Diagram

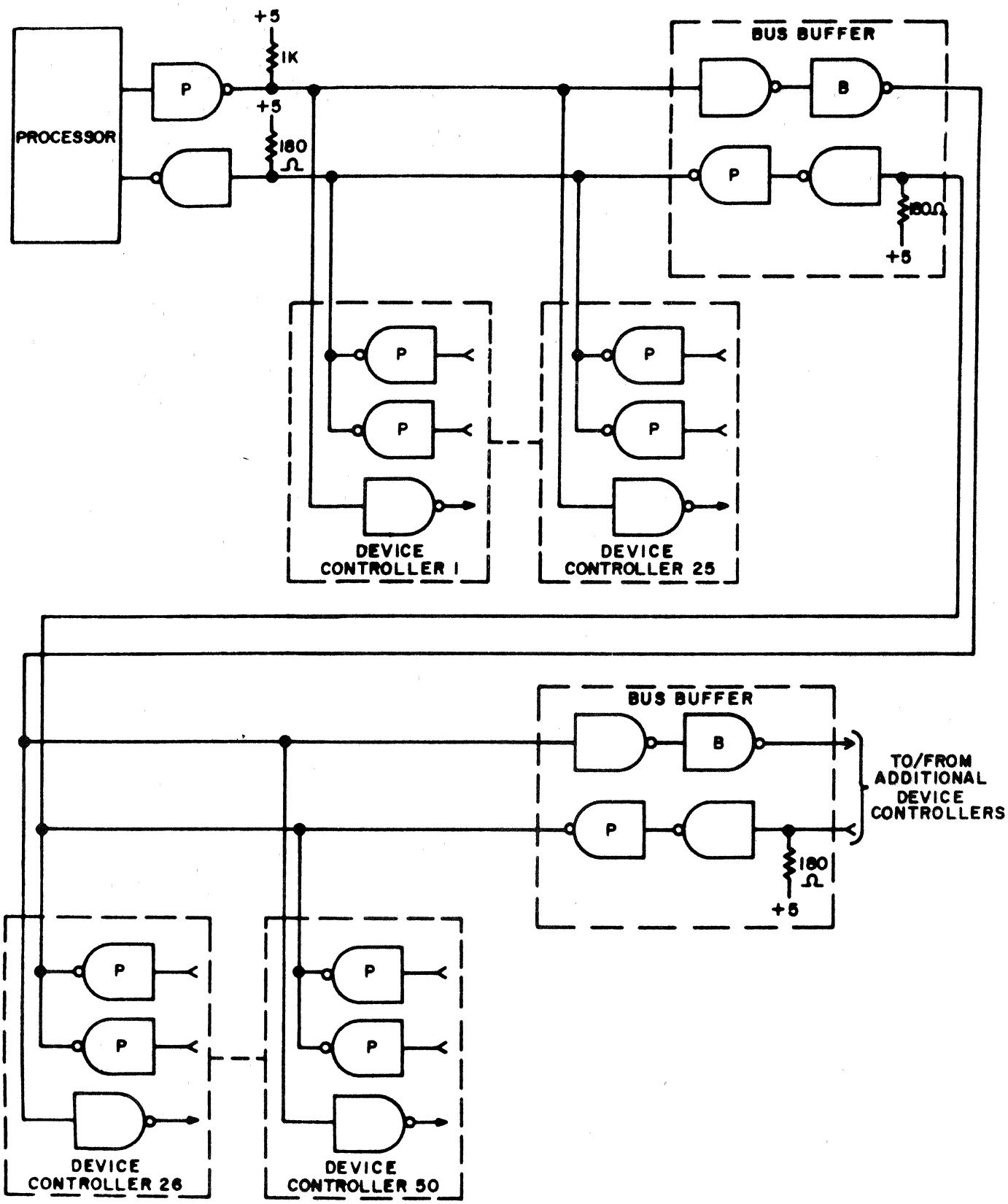


Figure 3. Multiplexor Channel, Bus Buffers

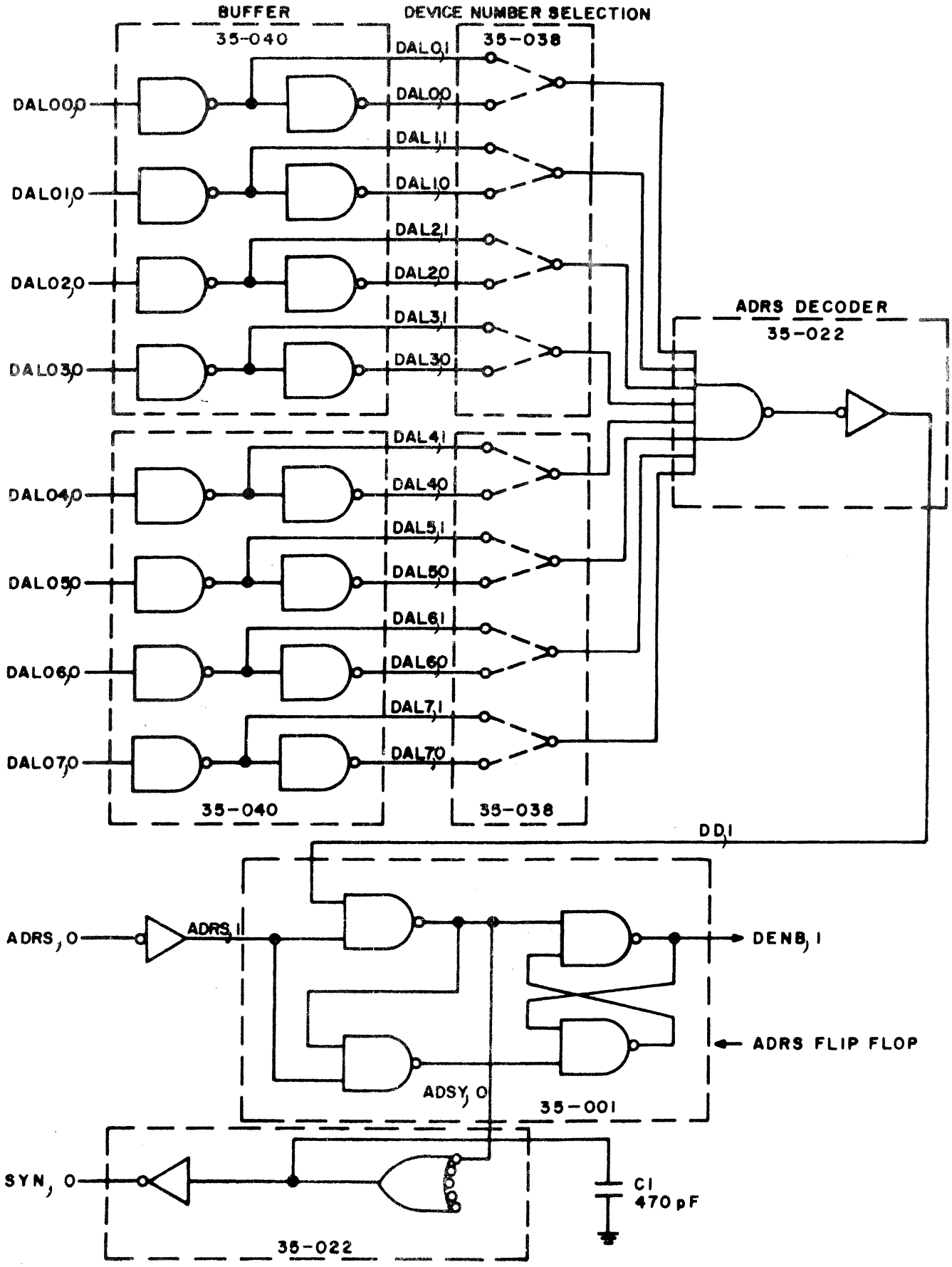


Figure 4. Device Addressing, Logic Diagram

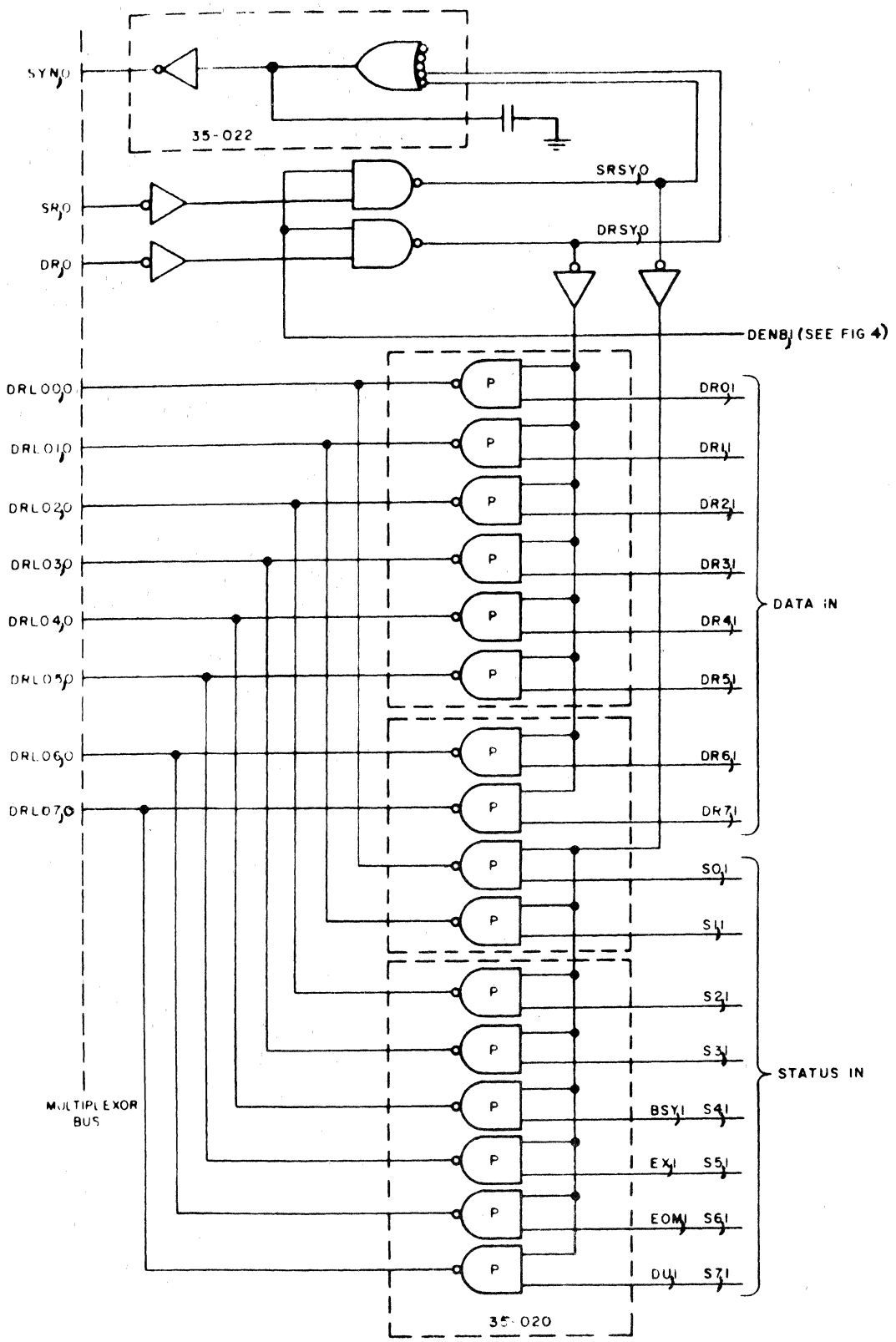


Figure 5. Data and Status Input, Logic Diagram

3.2 Data and Status Input

Figure 5 shows how a byte of data status may be read into the Processor from the Device Controller. When the device is addressed, DENBJ1 is high, enabling the Status Request (SR) or Data Request (DR) control line. The SR or DR control line, in turn, enables the status or data bytes onto the Data Request Lines (DRL00 through DRL07). The Model 35-020 daughter-board provides a convenient means of OR tying multiple data sources onto the DRL lines. Each of the control lines automatically generates a return sync signal SRSY0 or DRSY0. The Device Controller logic places a high on BSY1 until the data is ready and settled on the Data Request lines (DR01 through DR07). The Processor may now be synchronized to the device data rate by executing Sense Status instructions and branch looping on Busy until the Busy bit is low. Then, when the Busy bit is low, the program may execute a Read Data instruction. Device Synchronization can also be achieved by generating an interrupt when the data is ready.

The End of Medium (EOM) bit is normally set high at the termination of the device medium, such as End of Card. The Device Unavailable (DU) bit typically signifies that device power is not turned on.

The Examine Status (EX) bit is used to signify other appropriate device conditions. In this case, the user assigns S01 through S31 to appropriate conditions, such as Parity Error, etc.

3.3 Data and Command Output

Figure 6 shows how a byte of data and command may be output from the Processor. The buffered true and false Data Available Lines (DAL001 through DAL071 and DAL000 through DAL070) from Figure 4, feed to the set and reset inputs of the Data Register. When the device is addressed, DENBJ1 is high, enabling the control line DAGJ1 to strobe the data condition to the JK flip-flop Data Register. The DASY0 line also returns the Sync signal to the Processor. Either Model 35-015 or 35-016 daughter-boards may be used. The Model 35-015 contains four JK flip-flops and the Model 35-016 contains two JK flip-flops.

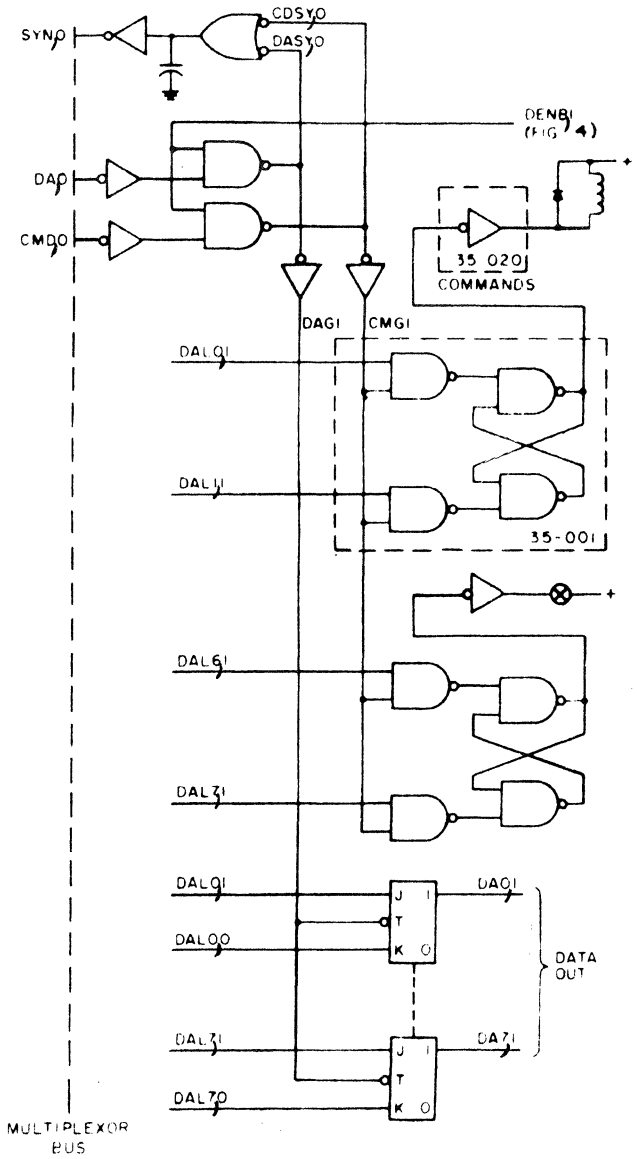


Figure 6. Data and Command Output, Logic Diagram

The Command lines are shown on Figure 6 as being used in the toggle mode. For example, a high on bit 0 (DAL00,1) sets a control relay when CMG,1 goes high. A high on bit 1 (DAL01,1) resets the relay. Bits 6 and 7 are shown operating an indicator. Other pairs of bits may be used to enable/disable interrupts, etc. The flip-flops may use the Model 35-001 daughterboard which contains four 2-input inverters.

3.4 Interrupt Control

Figure 7 shows a typical general purpose interrupt and interrupt acknowledge logic system. When an interrupt is generated, the Queue flip-flop is DC set via a differentiated negative going pulse. The output from the Queue flip-flop generates an Attention signal (ATN,0) to the Processor. The Processor responds with an Acknowledge Control Line which is received by the controller as Receive Acknowledge (RACK). Since the Queue flip-flop was set prior to receiving the RACK, the Gate G1 output disables G9, holding the G9 output high. The high output from G9 stops TACK,0 from sending the acknowledge to the next device. Thus, RACK,1 and the G2 output generate ATSY,0 via G3. ATSY,0 sends a SYNC back to the Processor, and also forces all inputs (DAL00,0 through DAL07,0) to zero. This causes the device number, wired in by the address strap board, to appear on the inputs of G10 through G17. Thus, the ATSY1 output from G4 enables the device number onto DRL00,0 through DRL07,0.

Capacitor C2 removes a 30 ns pulse which appears if the Queue flip-flop is set at the same instant that RACK,0 is received in response to another device interrupt. This pulse might otherwise reset the Queue flip-flop before the interrupt is serviced. The output from G4 also raises the acknowledge signal to the device. On receiving SYN,0, the Processor lowers RACK,1, causing the output from G4 to drop. This, in turn, causes the Queue flip-flop to reset.

NOTE

If the interrupt has not set the Queue flip-flop, the RACK,1 signal passes through G2 to TACK,0, and on to the next device.

If RACK,1 is high in response to another device, the output from G2 is low, disabling the interrupt from affecting G1. However, the interrupt remains in the Queue flip-flop, and is serviced after completion of the previous interrupt service.

The ENABLE line is an interrupt control device which may be set/reset via a Command Line flip-flop. The Enable flip-flop, if reset, disables the device such that it cannot interrupt the computer. An ARM function can be used to prevent the Queue flip-flop from being set.

3.5 Multiplexor Channel Timing

Both the Input and Output Operation on the Multiplexor Channel make use of "request-response" signalling. This allows the system to run at its maximum speed whenever possible, but permits a graceful slowdown if the length of the bus or the characteristics of a particular device controller require signals of longer duration. Device controller designs should keep Multiplexor Channel usage as fast as possible, consistent with practical circuit margins. This will give the greatest computer throughput when a system is configured with a number of peripheral devices.

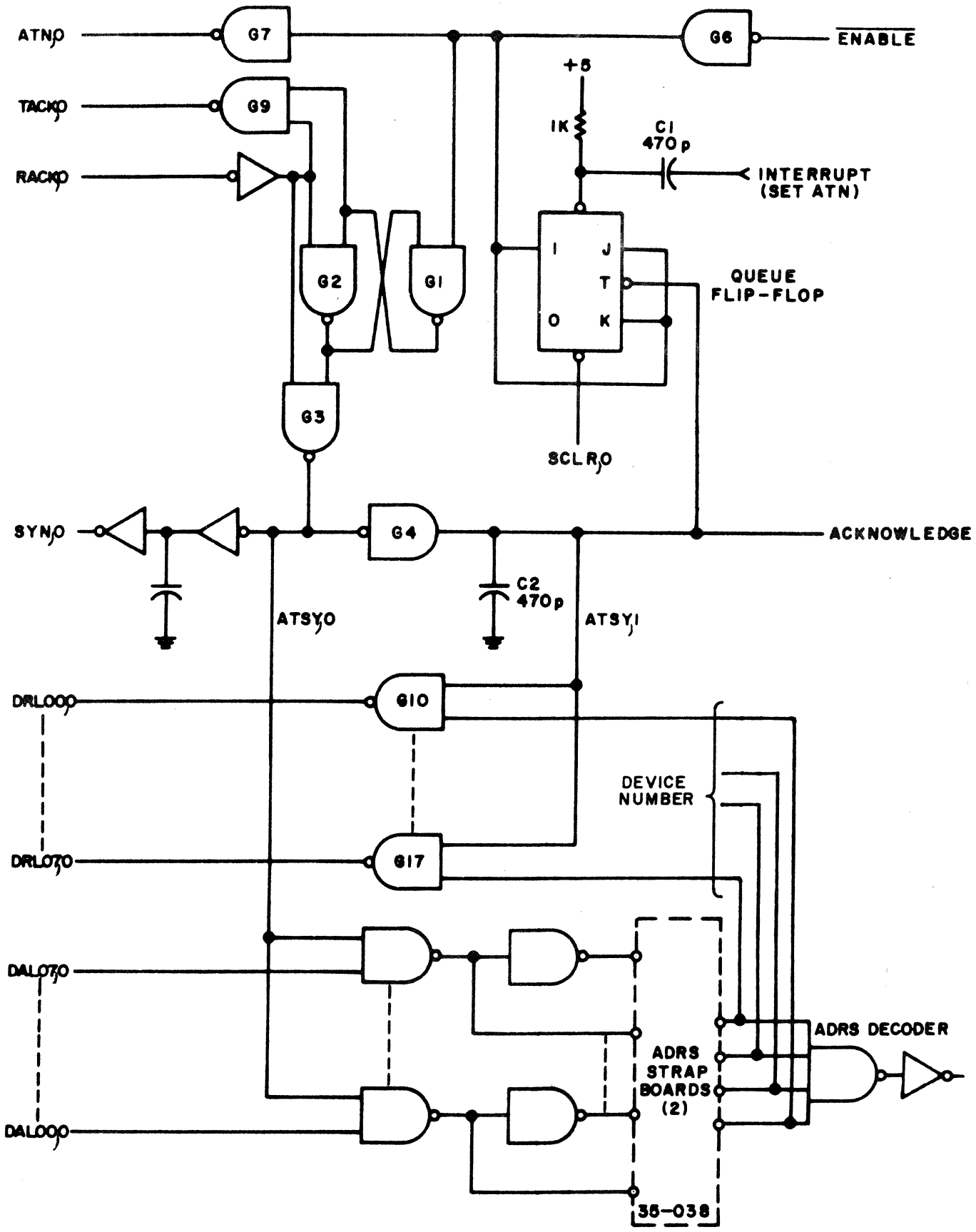
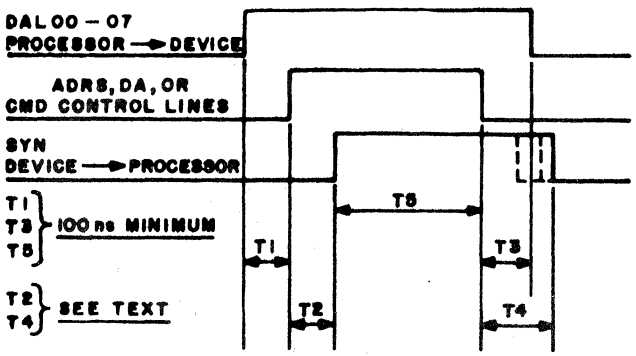
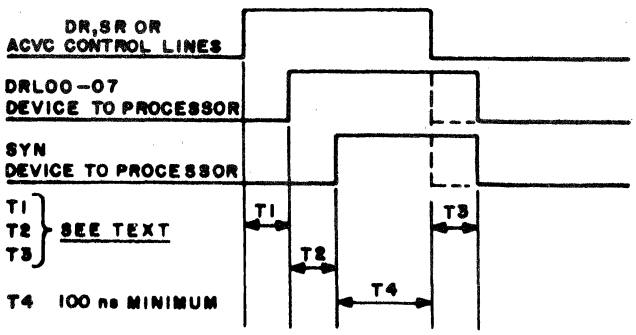


Figure 7. Interrupt Control, Logic Diagram

Typical operations are shown on Figure 8 for Input and Output. On the Output Operation, the Processor places a signal on the Data Available Lines followed by an appropriate Control Line signal. This stagger (T1) will vary depending upon which model Processor is in use, but it is guaranteed to be at least 100 ns. When the device controller has received the output byte, the SYN signal is returned to the Processor which then terminates the Control Line signal. Realizing that T5 is 100 ns minimum, the SYN delay T2 should be only long enough to guarantee proper reception of the output byte. The Control Line/DAL removal time (T3) is important where single-rail to double-rail operation is used - e.g. the ADRS flip-flop on Figure 4. A minimum of 100 ns is guaranteed for T3. For SYN generation as per Figures 4 and 5, the Control Line signal is DC coupled through the gates to form the SYN signal. The SYN removal time (T4) will be the delay through 4 DTL gates.



a. OUTPUT



b. INPUT

Figure 8. Multiplexor Channel Timing

It should be emphasized that the times shown on Figure 8 are defined for signals on the Multiplexor Channel. Within a given controller, one signal may flow through more gates than another signal and these delays must be considered.

For the Input Operation, the Processor places a signal on a Control Line. The Device Controller should gate signals to the DRL's as soon as possible to keep T1 minimum. The SYN delay (T2) must guarantee that the input byte is on the DRL's considering the slowest byte gates and the fastest SYN gates. The Processor will remove the Control Line signal when SYN is received with a minimum delay (T4) of 100 ns. With SYN and the byte gate DC coupled to the Control Line, the removal delay (T3) will be the sum of the corresponding gate delays.

4. TELETYPEWRITER DEVICE CONTROLLER

The ASR 33 Teletypewriter is one of the most popular peripherals with the GE-PAC 30 Digital System. This Section describes the Teletypewriter (TTY) Device Controller which interfaces both the ASR 33 and the continuous-service ASR 35 Models.

First, the characteristics of the Teletype writer machines must be introduced in order to fully understand the data handling and control features of this Device Controller. While a variety of models with different groups of peripheral devices are available, only the Automatic Send-Receive sets (ASR) is described here. The simpler units can be controlled by the same Device Controller.

The ASR Teletypewriter contains four peripheral devices; two data sources, a keyboard and paper tape reader; and two data receivers, a page printer and paper tape punch. The degree of operating flexibility depends upon the particular model. On the standard duty ASR 33 model, the punch is mechanically connected to the printer, so independent, simultaneous operation of these devices is not possible. The reader and keyboard are

electrically tied together, so the same restriction applies. However, the heavy duty ASR 35 models offer more freedom of use. In the Tape mode, for example, the reader and printer are on-line with the Processor, while the keyboard and punch are available for off-line tape perparation. A complete list of features is not provided here since the interface requirements do not change.

The description which follows covers Teletypewriter sets with the 8-level ASCII code at 100 WPM (110 BAUD), but other speeds and codes are easily accommodated. The line circuits in the ASR 33 consist of a Selector Magnet Driver (SMD) which amplifies the 20 ma serial input signal to drive the printer/punch, and a pair of contacts from the distributor and line break key which interrupt current in an external receiving circuit. The distributor converts parallel data from the keyboard/reader into serial line data. Two line options are possible. In the half-duplex option, the line circuits are connected in series so that any keyboard/reader data being sent to the computer is also printed. When loading data from tape, it is often unnecessary or undesirable to operate the printer, and its companion functions such as carriage return, line feed, bell, reader-on, reader-off, etc. The full-duplex option separates the Teletypewriter send and receive circuits. The processor can now control transmission to the printer so that keyboard/reader data may or may not produce a printed copy. This mode of operation is used with the TTY controller. The data feedback is controlled by the BLK (block) flip-flop in the Device Controller, and is described in more detail later.

The serial, ASCII code, Teletypewriter character sent to the TTY is shown in Figure 9. The eight data bits are preceded by a logical ZERO Start bit, and followed by two logical ONE Stop bits. This results in a stream of eleven bits per character.

Timing for 10 character/second (110 baud, 100 wpm) operation is illustrated. The 20 ma current level is determined by the SMD for data sent to the Teletypewriter. Distributor/line break contacts interrupt current in the Device Controller receiving circuit, where the receiver input filter and amplifier set the current at 20 ma. The 15v/20 ma circuit also reduces distributor contamination on the 33 ASR. While the bit timing within a character is fixed, the number of characters per second may vary. Data generated at the keyboard may come at almost any rate, while the tape reader operates at the 10 character/second maximum of the Teletype set. Computer data sent to the Teletype can also vary as long as the maximum rate is not exceeded. Timing circuits in the Device Controller control the data rate on output, and the Teletypewriter distributor controls the rate on input.

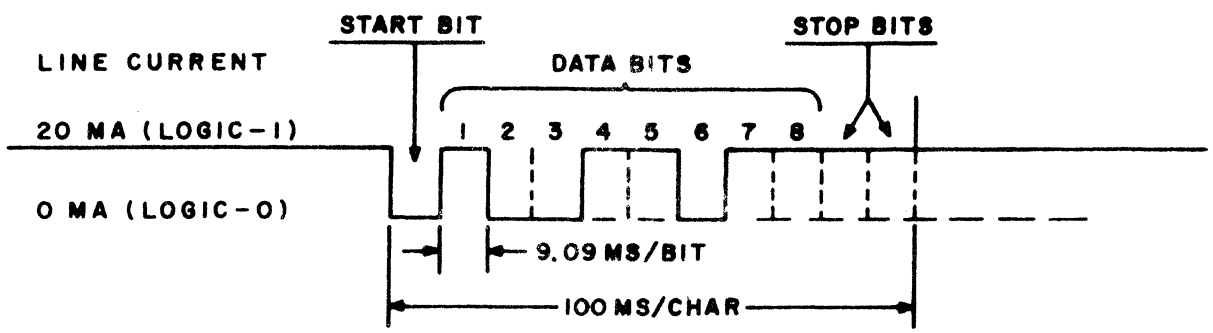


Figure 9. Serial Character to Teletypewriter (Letter Y - ASCII Code)

One further piece of information used by the Processor is the availability of the Teletypewriter. When the Teletypewriter is turned OFF, or is in the LOCAL mode (off-line), the line circuit terminals are shorted by the Teletypewriter line relay contacts. The SMD and distributor loops now appear the same as the idle condition between characters, as far as the Processor is concerned. Another terminal at the Teletypewriter can be examined for the presence of computer voltage switched by the Teletypewriter line relay. This Device Unavailable (DU) circuit is explained in more detail later.

The Teletypewriter Device Controller consists of one mother-board. This is of the copper (non-wire wrap) type with most of the integrated circuits mounted on the mother-board. The discrete circuits and some integrated circuits are mounted on daughter-boards.

N O T E

The Teletypewriter Device Controller, like all Device Controllers, can be located in any I/O slot. See the System Interface Manual, Publication Number 29-003, for more information on I/O System layout.

4.1 Circuit Descriptions

The following circuit descriptions reference the Teletypewriter Interface Functional Schematic, FS29, in Volume 2 of this manual.

4.1.1 Status and Commands.

The definitions for the Status and Command bytes for the TTY Interface are shown on Table 1. Since the TTY is both an input and output device, the Interface must be put in the Read Mode or Write Mode to properly generate the BSY status bit.

4.1.2 Initialization.

The Initialize signal (SCLR0, FS29-1) is equivalent to giving the DISABLE, BLOCK, READ, and PWR-OFF commands. The device enable (AD, FS29-1) and the interrupt queue (ATN, FS29-3) are also cleared by the SCLR0 signal.

4.1.3 Multiplexor Bus Communication Circuits.

The copper TTY Interface has its circuits for address, interrupt, and DRL cable drivers configured in almost the same manner as the copper portion of the Standard I/O Board. See the Systems Interface Manual, Publication Number 29-003.

Signals from the DAL bus are buffered and made available in both true and false form to be used by the address decoder, command register and data register. As shown on FS29-1, a group of 24 wire wrap pins in location 30 are used like 8 DPDT switches to set up the controller address. For the TTY, this is normally 02. A JK flip-flop is set or cleared at the end of the ADRS control signal under control of the decoder gates. The ATSYN0 line is normally high, so the DAL bus receivers are always active except during an interrupt acknowledge operation, which is described later.

The DRL bus drivers on FS29-2 place the contents of the 8 DRL XX0A lines on the DRL bus. The acknowledge interrupt address, data byte and status byte are OR tied on the DRL XX0A lines.

As shown on FS29-3, five Control Line signals are buffered and all but the ADRS signal are gated by the Address flip-flop. In addition, the DA and DR signals are also gated by the idle timer condition, TMG0. The gated control signals and ATSYN0 are ORed to generate the SYN0 signal. A capacitor delay is provided to guarantee that SYN0 is always later than any signals gated to the DRL bus.

TABLE 1
TELETYPEWRITER STATUS AND COMMAND BYTE DATA
HEX ADDRESS 02

BIT NUMBER	0	1	2	3	4	5	6	7
STATUS BYTE			BRK		BSY	EX		DU
COMMAND BYTE	DISABLE	ENABLE	UNBLOCK	BLOCK	WRT	READ	PWR ON	PWR OFF

BRK — The Break Bit is set when the Break key on the Teletypewriter is depressed, or the Teletypewriter is logically disconnected from the Controller.

BSY — The significance of the Busy bit depends upon whether a Read or a Write operation is in progress. During Write mode, BSY is normally low, and goes high only while data is being received by the device. During Read mode, BSY is normally high, and goes low only when data has been received from the device, but not yet been transferred to the Processor. During Read mode, BSY goes high again as soon as the Processor accepts the data.

EX — The Examine bit is set whenever BRK is set.

DU — The Device Unavailable bit is set whenever the Teletypewriter is in the OFF or LOCAL mode, or power is not connected to the Teletypewriter.

DISABLE — This command disables the Device Interrupt to the Processor from the Device Controller.

ENABLE — This command enables the Device Interrupt to the Processor from the Device Controller.

UNBLOCK — This command enables the printer to print data entered via either the keyboard or the tape reader.

BLOCK — This command disables the feature described above.

WRT, READ — The Write and Read commands are used to define the significance of the BSY bit.

PWR ON, PWR OFF — The Power On and Power Off commands are significant only with those Teletypewriters provided with an optional Power Control Box. The option permits switching Teletypewriter power under program control.

The interrupt Queue flip-flop, ATN, is set whenever the BSY status condition changes from true to false. If the controller has been enabled, EBL,0 will be low and the signal from the ATN flip-flop will be placed on the ATN,0 line to the Processor. See FS29-3. The interrupt acknowledge signal enters the controller on the Received Acknowledge line (RACK,0) and will be passed onto the next controller on the Transmit Acknowledge line (TACK,0), if the ATN,1 lead is low. This will be the case if the controller is disabled or is enabled without an interrupt in the Queue flip-flop. A pair of cross-coupled gates form a contention circuit to prevent multilation of the RACK,0/ TACK,0 pulse if the ATN flip-flop is set while the pulse is passing through the controller. A capacitor filter on the ATSYN,1 lead removes a short signal spike which could occur when the timing in the contention circuit causes a momentary race condition.

When the RACK,0 signal arrives and the ATN1 lead is high, ATSYN,0 and ATSYN,1 change to their active states - low and high respectively. Referring again to FS29-1, the low state on ATSYN,0 forces the 8 DAL XX,1 lines high and the 8 DAL XX,0A lines low. These signals passing through the 8 address straps produce an 8-bit byte with high levels where the controller address bits are one and low levels where bits are zero. This self-generated controller address is gated by the ATSYN,1 signal to the DRL XX,0A lines and then to the DRL bus. See FS29-2. ATSYN,0 is also used to generate the SYN signal.

4.1.4 General Operation - Writing and Reading. In the earlier description of the TTY operation, it was noted that the TTY is arranged for the full-duplex option, i.e. the sending and receiving circuits are separated. The block diagram of the TTY controller, FS29-12, shows the independent transmit and receive circuits in the TTY coupler with a common shift register and timer. The controller is basically half-duplex in operation providing controlled feedback of serial, received data to the TTY printer/punch.

The 11-bit shift register performs the serial to parallel conversion of data (and the inverse) in addition to generating a timer stop signal when the proper register patterns are detected by the end-of-character gates.

On output, the Data Available (DA) signal loads the register with all ones, loads zeros in the START bit flip-flop and wherever the data byte has zeros. At the end of the DA signal, the timer is started and the register is shifted at the data transfer rate, e.g. 110 bits per second for the Model 33/35 TTY. The BSY condition goes true. As data leaves one end of the register, zeros are shifted into the other. When ten zeros are detected, the register is shifted once more; the timer is stopped, the BSY condition goes false and an interrupt is stored in the ATN flip-flop.

The input operation begins when the START bit of a serial TTY character opens the input loop. The register is again loaded with all ones and the timer is started. The shift pulses are generated at the appropriate data rate, but displaced by half a bit time so the line is strobed into the register near the center of each bit. When the START bit reaches bit 7 of the register, the register is shifted once more and the timer is stopped. The input line is checked for the presence of the closed loop, STOP bit. If it is there, the BSY condition goes false and an interrupt is generated. Should line be open, the BSY condition goes false, the BRK and EX conditions go true, and an interrupt is generated. The timer will not be restarted on the continuing open line condition. The line break must be removed before the timer start circuit will respond to the next START bit. The BSY condition will go true when the contents of the register are transferred to the Processor by a DR signal.

Serial feedback to the printer of keyboard/reader data is controlled by the BLOCK flip-flop. If the UNBLOCK command has been given to the controller, the serial data out of the receiver is turned around and put on the transmit line to the TTY. The BLOCK function does not affect the normal computer output operation to the TTY.

4.1.5 Data Register, Timer and Control. The data shift register with parallel loading gates is shown on FS29-4. It has 11 stages designated DRX, DRY, DR00-07 and DRN. A common set line (SDR,0) permits loading all ones and a common shift or toggle line (TDR,1) shifts the data from DRX to DRY etc. The parallel load signal, DAG,1, strobes the load gates and clears DR00-07 wherever a zero appears in the corresponding bit of the data byte. The leading edge of DAG1 is differentiated (FS29-5) to generate the SDR,0 signal on an output operation. The DRN flip-flop is always cleared by DAG,0 to load the START bit. The trailing edge of DAG1A (FS29-6) toggles the Timer Gate flip-flop (TMG) set and the Device Transmit flip-flop (DT) clear. The DT flip-flop controls the flow of serial data into or out of the data register. With DT cleared, DT,1 will be low and block the data input to DRX (FS29-4) so that the J lead is low and the K lead is high. When the TDR,1 pulses shift the data toward DRN and the transmitter circuit, zeros are shifted into DRX and propagate through the register.

Regardless of the data contents of the register, there will be at least 2 ones being shifted, e.g. the STOP bits that originated in DRX and DRY. The end-of-character gates for output (FS29-5) put a low signal on EOC,0 when DRX, DRY, and DR00-07 all contain zeros and the TDR,1 goes high. The leading edge of EOC,1 clears TMG to stop the timer. The trailing edge produces one last shift of the register contents. The last TDR,1 pulse is quite short - about 500 ns. The ten preceding pulses are 1/8 of the bit time, e.g. 1.14 ms for 110 bits per second operation. Data from the last register stage (DRN) is gated by TMG,1 and DT,0 to the transmit line (TRNS,0 and TRNS,1) as shown on FS29-6.

On an input operation (see FS29-5 and FS29-6), the Timer Gate flip-flop (TMG) is set when the normal, closed loop, high state of the Device Data lead (DD,1) goes low. The Line Ready flip-flop (RDY) has been in the set state, and together with TMG,0 will have previously generated AST,0 and AST,1. As DD,0A goes high, the ST,0 pulse appears and sets TMG, sets DT, clears RDY, sets TMD, and produces SDR,1 and SDR,0 to load the register with all ones. The delay capacitor on the AST,0 lead keeps ST,0 in the low state for about 500 ns.

When the START bit has been shifted through the register into DR07, the next TDR,1 pulse gated by DR07,0 and DT,1 produces a low state on EOC,0. In the same manner as on output, the leading edge of EOC,1 clears TMG to stop the timer and limit the width of the EOC pulse to about 500 ns. The EOC,1 pulse tests the state of the input line by strobing DD,1. If the line is closed, DD,1 will be high and the RDY flip-flop will be set. Should the line be open, RDY will remain cleared. The TMG,0, RDY,0, and DD,0A conditions then generate the line break condition - BRK,0 and BRK,1. Even though DD,0A is still high, AST,1 remains low since RDY has not been set. This prevents the generation of ST,0, which would restart the timer again and again, bringing all zero characters into the register with an interrupt at the end of each timer operation. When the line break condition is removed, DD,1 and TMG,0 set RDY. Lead AST,1 now goes high, awaiting only the next START bit to generate ST,0.

Data from the receiver line, DD,0A, is gated by TMG,1 and BLKDT,0 signals to the transmit line (TRNS,0 and TRNS,1) as shown on FS29-6.

The timer on FS29-5 is a start-stop multivibrator consisting of the 35-037 network, the 35-027 dual differentiator, and the 35-020 gated pulse inverters. A regulated, temperature compensated power supply is derived from the P15 voltage line. The low-drift timing components (100K and .022MF) are adjusted so the basic timer oscillates at 440 Hz. This is divided by four with the counters TMC and TMD to produce TDR pulses at 110 Hz.

In the OFF state, the timer is clamped by TMG$_J$1A so that TMA$_J$0 is low, TMB$_J$0 high, and TMC and TMD cleared. This guarantees the proper pulse phasing when the timer starts. From this initial state, the end of the first TDR$_J$1 pulse occurs 9.09 ms after the TMG flip-flop is set, and produces the proper Output/Write mode timing.

In the Input/Read mode, the ST$_J$0 pulse sets TMG and sets TMD. This phase change in TMD means that the end of the first TDR$_J$1 pulse will occur 4.545 ms after the TMG flip-flop is set. The rest of the TDR pulses are still 9.09 ms apart. As noted earlier, the first ten TDR pulses will be 1.14 ms wide and the last pulse shortened to about 500 ns when the EOC condition clears TMG and TMG$_J$1A returns to the low state.

4.1.6 TTY Coupler Circuits. The transmit and receive loop circuits are both operated at 20 ma from the P15 power source. See FS29-7. The send and receive circuits in the TTY have no ground reference other than that provided by the TTY controller.

For the receive loop, current originates on the line power and protection circuit (35-083). The path to ground includes the RP lead, the distributor and line break contacts in the TTY, the RN lead and the transistor with filter network on the TTY coupler (35-024). In the idle condition, loop current flows to keep the transistor ON and the device data lead (DD$_J$0) in the low state. When either the distributor or the line break contacts in the TTY cause the loop to open, the transistor turns OFF and DD0 goes high. The 1 ms filter, which reduces distributor and line noise, produces a slowly changing waveform at the transistor collector. To prevent oscillation as the IC's move slowly through the active region, the final waveform is squared by positive DC feedback from DD$_J$0A to DD$_J$0.

The transmit loop is formed by the current source on the 35-083 board, the TP lead, the selector magnet driver (SMD) in the TTY, the TN lead, the 100 ohm resistor on the 35-024 board, and the gate connected to the TRNS$_J$0 lead. When the gate turns OFF to open the transmit line, both TP and TRNS$_J$0 are clamped to P5 by diodes.

The transmit and receive circuits have been designed with enough margin to allow up to 150 ohms of line resistance in each loop.

When the TTY line relay is operated, voltage from TP is returned on the TDU lead. The transistor on the 35-024 board turns ON and the Device Unavailable (DU$_J$1) lead goes to its false (low) state. Since the TDU signal will contain a portion of the signal sent to the TTY, a 10 ms filter at the transistor prevents false operation. The TDU signal will be removed when the TTY is in the OFF or LOCAL mode or if the cable to the TTY is open. This turns the transistor OFF and DU$_J$1 goes to the true (high) state.

Experience has shown that contact stagger on the TTY line relay can occasionally connect the internal, LOCAL loop power to the TTY coupler circuits in the controller and cause damage. Line protection diodes on the 35-083 board clamp the lines to ground for negative signals and to P5 or P15 for large, positive signals.

4.1.7 Status and Data Request Gates. The status and data bytes returned to the Processor are OR tied into the DRL XX,0A lines as described in 4.1.3 above. The gates for 4 bits of status and 8 data bits are shown on FS29-9. The BSY and DU conditions are gated to the DRL04,0A and DRL07,0A lines respectively. The line break (BRK) condition is gated to DRL02,0A and DRL05,0A to generate the BRK and EX bits.

The true contents of the data register are strobed by the DRG,1 signal onto the DRL XX,0A lines.

4.1.8 Command and Busy Circuits. Three Command flip-flops are loaded by the command byte from the Processor as shown on FS29-10. Since two bits are used for each flip-flop, they may be set or cleared independently or simultaneously in any combination. As described above, the ENABLE flip-flop controls the gating of the ATN flip-flop to the interrupt line, ATN,0 and the BLOCK flip-flop controls the keyboard to printer feedback on an input operation.

The READ flip-flop and the data control signals, DT,0 and TMG,1, generate the BSY,1 signal. This is gated back to the Processor as part of the status byte. It is also differentiated to set the ATN flip-flop whenever BSY,1 changes from the high to the low state. In the Output/Write mode, READ,1 will be low and BSY,1 will be true whenever TMG,1 is true, i.e., during the 100 ms period of time required to transmit a character to the TTY. For the Input/Read mode, BSY,1 will be true when TMG,1 is true and go false when timing is over. However, BSY,1 will again go true when the Processor executes a READ DATA instruction, which inputs a data byte and clears the DT flip-flop.

4.1.9 Power Control Option. For systems where unattended operation requires only occasional use of the TTY, it is desirable to remove power from the TTY when it is not needed. This is especially important with the Model 33 which is not a heavy duty machine. The optional, added circuits for power control are shown on FS29-8. A Power Control flip-flop (PWR), which is set and cleared with the OC instruction, is connected by a separate cable to a Power Control Box at the TTY. The TTY power switch is placed in the **LINE** position.

In the power off condition, the DU,1 lead will be true, the transistor in the DU Timer (35-126) is OFF, the delayed DU (DDU,1) line will be true and force a true condition on BSY,1. When the AC power is applied to the TTY, its signal line relay operates causing the DU,1 lead to go low. In about one second, the input capacitor in the DU timer receives enough charge to turn on the transistor, put a low signal on DDU,1 and allow the BSY,1 line to assume its proper condition as set up by the READ flip-flop. This time is sufficient to permit the motor driver mechanisms in the TTY to reach the proper speed before the computer attempts an output to the printer.

4.2 Maintenance

Refer to the appropriate vendor maintenance manuals for maintenance information on the Teletypewriter sets. The overall operation of both the Teletypewriter and the TTY interface may be checked by running the TTY Test Program as described in GE-PAC 30 Publication Number 06-004A12 (part of the Programming Manual, Publication Number 29-013).

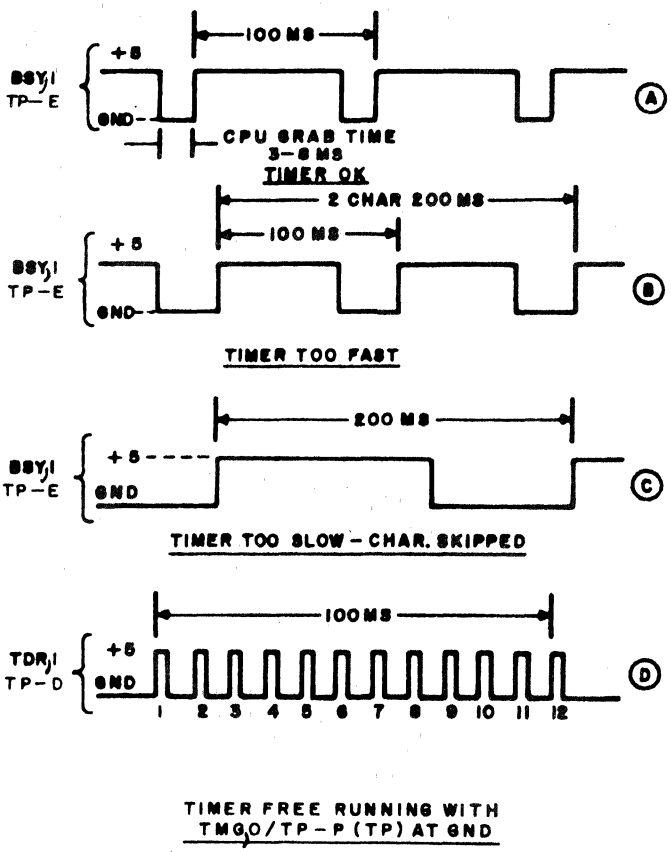
The only adjustment on the TTY Controller is the potentiometer on the 440 HZ gated timer. Check the timing as follows:

1. Initialize the system to put the TTY Interface into the Read mode.

2. Connect an oscilloscope to Test Point E; 2-v/cm on Y, 10MS/cm on X, internal, negative sync.

3. Generate a continuous stream of TTY data by running a tape on the reader, or by holding down the REPEAT and RUB-OUT keys on the keyboard.

4. The waveform should look like Figure 10A with negative pulses at a 100 ms rate and a width of 3 to 8 ms. This is the grab time.



TIMER FREE RUNNING WITH TMGJO/TP-P (TP) AT GND

Figure 10. TTY Controller Timing Adjustments

If the timer is too fast, the grab time will be excessively long as in Figure 10B and the computer will cause trouble on output by exceeding the TTY's receiving rate. Input characters will also be improperly received. If the timer is too slow, the pulse will be very small, or the Controller may skip characters (Figure 10C).

The timer can be adjusted with the TTY running as above. Turn the potentiometer on the daughter-board at location 45 until the grab time (negative pulse) is 6 ms wide at 25 degrees centigrade (room temperature).

An alternate check and adjustment may be made without running the TTY. Ground Test Point P and connect the oscilloscope to D (TDRJ1). A string of 11 positive pulses (Figure 10D) on the scope in 100 ms will occur when the timing is correct. This method is not as accurate as adjusting the grab time and should only be used as a first-order check or if a TTY machine is not available when the timer is adjusted.

If the timing is very slow and connot be adjusted, check for +15V on the TTY Interface. If still not proper, replace the 35-027 daughter-board in location 46, the 35-037 daughter-board in location 45, and/or the 35-020 daughter-board in location 44. Refer to FS29-5.

4.3 Mnemonics

The following alphabetical list briefly describes the mnemonics used in the TTY Device Controller. The source on FS29 for each signal is also provided.

MNEMONIC	MEANING	FS29 LOCATION
AD	Address flip-flop	1P2
ATN	Attention Queue flip-flop	3N7
ATSYN/1	Return Address Strobe - present when the Controller is responding to ACK	2N7
BLK	Block flip-flop	10K4
BRK	Line Break from TTY detected	6R5
BSY	Busy status circuit	10M7
CL's	Control Lines from Processor	3A3
ADRS/0	Address	
DA/0	Data Request	
DR/0	Data Request	
SR/0	Status Request	
CMD/0	Command	
CMG/0	CMD signal gated by AD flip-flop	3H3 10A7
DAG/0	DA signal gated by AD and TMG flip-flops	3H3
DAL's	Data Available Lines - I/O Bus from Processor	1A1
DD/1	Device Data - copy of signal from TTY	6A7
DR00 - DR07	Data Register flip-flops for data bits	4H5
DRG/0	DR signal gated by AD and TMG flip-flops	3H4

MNEMONIC	MEANING	FS29 LOCATION
DRL's	Data Request Lines - I/O Bus to Processor	2C5
DRN	Data Register flip-flop for Start bit	4P5
DRX	Data Register flip-flop for Stop bit	4E8
DRY	Data Register flip-flop for Stop bit	4G8
DT	Device Transmitting flip-flop	6K7
DU1	Device Unavailable	7P4
EBL	Enable Attention flip-flop	3P3
EOC	End Of Character	5G9
LSX	Least Significant hexadecimal digit - hex value of bits 4-7	1A4
MSX	Most Significant hexadecimal digit - hex value of bits 0-3	1A7
RACK$_0$	Received ACK signal - into the Controller	3A8
RDY	Ready flip-flop - checks for line break	6F4
SCLR$_0$	System Clear - initialize signal on I/O Bus	1A8
SDR$_0$	Set Data Register to all ONE's	4R6
SRG$_0$	SR signal gated by AD flip-flop	3H4
ST$_0$	Start Timer pulse in READ mode	6K2
SYN	System Synchronize	3A5
TACK$_0$	Transmitted ACK signal from the Controller	3A9
TDR$_1$	Toggle Data Register - shift pulse	4R6
TMA	Timer Oscillator flip-flop - 440 pps output	5H5
TMB	Timer Oscillator flip-flop - 440 pps output	5D5

MNEMONIC	MEANING	FS29 LOCATION
TMC	Timer Counter flip-flop - 220 pps output	5K4
TMD	Timer Counter flip-flop - 110 pps output	5L4
TMG	Timer Gate flip-flop - (active when timer is running)	6K3

THE POWER SYSTEM

1. INTRODUCTION

The power system in a GE-PAC 30 Digital System consists of a power supply, the power distribution buses, and cables which interconnect the two. The modular power supply is rugged and conservatively rated to provide long-term reliability. Three series regulators provide the +5 logic voltage and ±16 memory voltages. Regulation is ±1%. All outputs are current limited to 30% of the rated output in the event of shorts to ground. Reverse biased high current diodes are connected across all outputs to prevent circuit damage in case of accidental shorts between supply voltages.

Cabling between the power supply and the Processor is shown on Sheets 2 and 3 of Drawing 3-109R01B12 and on Drawing 17-011B06. The drawings are provided in Volume 2 of this manual.

2. POWER SUPPLY DESCRIPTION

Two power supply models have been provided with GE-PAC 30 Digital Systems: a purchased supply, and a supply manufactured by GE-PAC 30. The supplies are similar. Drawings for both supplies are provided in Volume 2 of this manual. The following descriptions are based on the vendor supply; however, most portions apply equally to the GE-PAC 30 supply.

2.1 Specifications

Input	115VAC ±10% or 230 VAC ±10%[1].	
	50 - 60 CPS	
	500 W	
Outputs	+5 VDC ±1%	20A
	+16 VDC ±1%	5A[2].
	-16 VDC ±1%	4A[2].

2.2 +5 Volt Supply

The +5 Volt Supply consists of the A3, A4, and A6 modules (located under the removeable protective screen). The +5 volt supply is designed to output 20 amperes, which is considerably more than the average system requires. It's adjustment is over a ±10% range and should be adjusted to +5.0 volts (trim-pot on the A6 module). This supply is equipped with over voltage protection in the form of a crowbar circuit across its output. If the output voltage rises above 7 volts, the crowbar circuit shunts the output to ground, blowing a fuse, and disabling the 5 volt supply.

2.3 16 Volt Supplies

The ±16 volt supplies are used primarily by the core memory, and for this reason are temperature programmed. A thermistor located on the MEO mother-board controls the +16 volt output as follows:

10° C produces 17.5 volts
50° C produces 15 volts

Linearity of this relationship is held to ±10%. The -16 volt supply tracks the +16 volt output. See the Memory System section of this manual for an adjustment procedure.

---

1. Selected by jumpers on TB2 and TB3 which are located under the rear cover. See Drawing D001-0184 in Volume 2.

2. The ±16 volt supplies are temperature programmed. See Section 2.3.

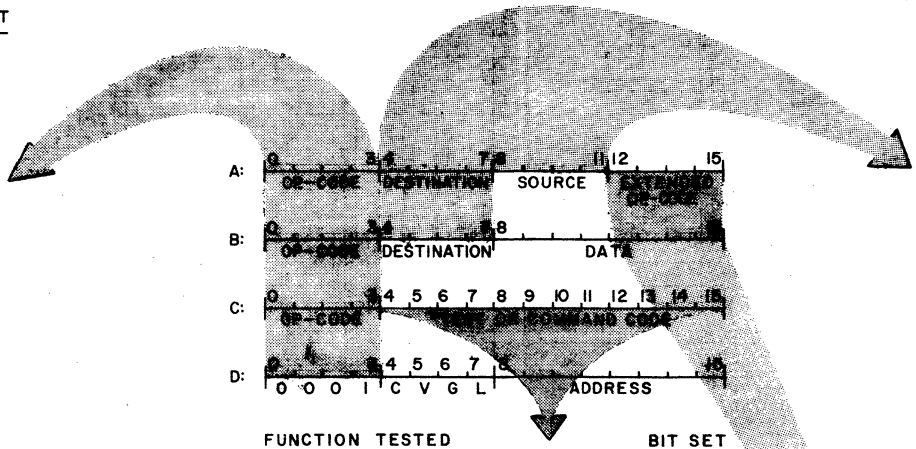
MODEL 30-1
GENERAL REFERENCES

The following index lists reference material provided in this section.

CORE MEMORY ALLOCATION FOR GENERAL REGISTERS AND PROGRAM STATUS WORDS

Hexadecimal Memory Address	Register Assignment
General Registers	
00 - 01	R0
02 - 03	R1
04 - 05	R2
06 - 07	R3
08 - 09	R4
0A - 0B	R5
0C - 0D	R6
0E - 0F	R7
10 - 11	R8
12 - 13	R9
14 - 15	R10
16 - 17	R11
18 - 19	R12
1A - 1B	R13
1C - 1D	R14
1E - 1F	R15
Hardware Registers	
20 - 21	Instruction Register
22 - 23	Instruction Address Register
24 - 25	Current PSW: Status and Condition Code
26 - 27	Current PSW: Instruction Address Counter
28 - 29	Display support: First operand
2A - 2B	Display support: Second operand
2C - 2D	Display support: Result
Program Status Words	
30 - 33	Old PSW: Illegal Instruction Interrupt
34 - 37	New PSW: Illegal Instruction Interrupt
38 - 3B	Old PSW: Machine Malfunction Interrupt
3C - 3F	New PSW: Machine Malfunction Interrupt
40 - 43	Old PSW: External Device Interrupt
44 - 47	New PSW: External Device Interrupt
48 - 4B	Old PSW: Divide Fault Interrupt
4C - 4F	New PSW: Divide Fault Interrupt
50	First user available memory location.

OP-CODE	INSTR.	FORMAT
0000	DO	A
0001	BRANCH	D
0010	TEST	C
0011	COMMAND	C
0100	LOAD	A
0101	LOAD IMM.	B
0110	OR	A
0111	OR IMM.	B
1000	AND	A
1001	AND IMM.	B
1010	EX. OR	A
1011	EX. OR IMM.	B
1100	ADD	A
1101	ADD IMM.	B
1110	SUBTRACT	A
1111	SUB. IMM.	B



DESTINATION/SOURCE CODE	REGISTER
0000	R0
0001	R1
0010	R2
0011	R3
0100	R4
0101	R5
0110	R6
0111	R7
1000	AR
1001	MDH
1010	MDL
1011	MAH
1100	MAL
1101	RAH/SDR
1110	RAL/SCR
1111	DFR

THE FOLLOWING REGISTERS CANNOT BE GATED TO THE "B" BUS: AR, MAH, MAL, RAH, RAL, SCR

IN ORDER FOR SDR OR SCR TO BE ADDRESSED, THE BANK FLIP-FLOP MUST BE SET.

IN ORDER FOR RAH OR RAL TO BE ADDRESSED, THE BANK FLIP-FLOP MUST BE RESET.

INSTR.	FORMAT	OP-CODE
ADD	A	1100
ADD IMM.	B	1101
AND	A	1000
AND IMM.	B	1001
BRANCH	D	0001
COMMAND	C	0011
DO	A	0000
EX. OR	A	1010
EX. OR IMM.	B	1011
LOAD	A	0100
LOAD IMM.	B	0101
OR	A	0110
OR IMM.	B	0111
SUBTRACT	A	1110
SUB. IMM.	B	1111
TEST	C	0010

FUNCTION TESTED	BIT SET
I/O INTERRUPT	5
I/O SYNC.	6
CONSOLE INTERRUPT	7
CONSOLE SINGLE MODE	8
UTILITY FLIP-FLOP	9
MEMORY PARITY FAIL	10
PRIMARY POWER FAIL	11

COMMANDS	BITS SET
READ MEMORY	7
WRITE MEMORY	6, 7
WRITE MEMORY, PROTECTED	5,6,7
SET BANK *	8
RESET BANK *	9
TRIGGER BANK *	8,9
SET UTILITY *	10
RESET UTILITY *	11
TRIGGER UTILITY *	10,11
RESET MEMORY PARITY FAIL *	12
SET WAIT ALARM *	13
RESET WAIT ALARM *	14
POWER DOWN	15

* = FLIP-FLOPS

EXTENDED OP.-CODE

FOR A,S,X,N,O
- 1XXX DISABLE AR TO ALU
- XIXX SET FLAGS
- XXIX CARRY INTO ALU
- XXXI CARRY OUT OF ALU

FOR "L" ONLY:
- 00XX LOAD
- 01XX SHIFT RIGHT
- 10XX SHIFT LEFT
- XXIX SHIFT WITH CARRY INTO ALU
- XXXI SHIFT WITH CARRY OUT OF ALU
- 11XX LOAD ONE'S COMPLEMENT

X = DON'T CARE STATE

MICRO-INSTRUCTION SUMMARY

3/4

SUMMARY OF USER INSTRUCTIONS - ALPHABETICAL BY NAME

INSTRUCTION	TYPE	MNEMONIC	OP CODE
Acknowledge Interrupt	RR	AIR	9F
Acknowledge Interrupt	RX	AI	DF
Add Halfword	RR	AHR	0A
Add Halfword	RX	AH	4A
Add Halfword Immediate	RS	AHI	CA
Add with Carry Halfword	RR	ACHR	0E
Add with Carry Halfword	RX	ACH	4E
AND Halfword	RR	NHR	04
AND Halfword	RX	NH	44
AND Halfword Immediate	RS	NHI	C4
Branch and Link	RR	BALR	01
Branch and Link	RX	BAL	41
Branch on False Condition	RR	BFCR	03
Branch on False Condition	RX	BFC	43
Branch on True Condition	RR	BTCR	02
Branch on True Condition	RX	BTC	42
Branch on Index Low or Equal	RS	BXLE	C1
Branch on Index High	RS	BXH	C0
Branch Unconditional	RR	BR	03
Branch Unconditional	RX	B	43
Branch on Overflow	RX	BO	424
Branch on Zero	RX	BZ	433
Branch on Not Zero	RX	BNZ	423
Branch on Equal	RX	BE	433
Branch on Not Equal	RX	BNE	423
Branch on Plus	RX	BP	422
Branch on Not Plus	RX	BNP	432
Branch on Low	RX	BL	428
Branch on Not Low	RX	BNL	438
Branch on Minus	RX	BM	421
Branch on Not Minus	RX	BNM	431

INSTRUCTION	TYPE	MNEMONIC	OP CODE
Branch on Carry	RX	BC	428
Compare Logical Halfword	RR	CLHR	05
Compare Logical Halfword	RX	CLH	45
Compare Logical Halfword Immediate	RS	CLHI	C5
Divide Halfword	RR	DHR	0D
Divide Halfword	RX	DH	4D
Exclusive OR Halfword	RR	XHR	07
Exclusive OR Halfword	RX	XH	47
Exclusive OR Halfword Immediate	RS	XHI	C7
Load Byte	RR	LBR	93
Load Byte	RX	LB	D3
Load Halfword	RR	LHR	08
Load Halfword	RX	LH	48
Load Halfword Immediate	RS	LHI	C8
Load Program Status Word	RX	LPSW	C2
Multiply Halfword	RR	MHR	0C
Multiply Halfword	RX	MH	4C
No Operation	RR	NOPR	02
No Operation	RX	NOP	42
OR Halfword	RR	OHR	06
OR Halfword	RX	OH	46
OR Halfword Immediate	RS	OHI	C6
Read Block	RR	RBR	97
Read Block	RX	RB	D7
Read Data	RR	RDR	9B
Read Data	RX	RD	DB
Output Command	RR	OCR	9E
Output Command	RX	OC	DE
Shift Left Arithmetic	RS	SLHA	CF
Shift Left Logical	RS	SLHL	CD
Shift Right Arithmetic	RS	SRHA	CE
Shift Right Logical	RS	SRHL	CC
Store Byte	RR	STBR	92
Store Byte	RX	STB	D2
Store Halfword	RX	STH	40

INSTRUCTION	TYPE	MNEMONIC	OP CODE
Subtract Halfword	RR	SHR	0B
Subtract Halfword	RX	SH	4B
Subtract Halfword Immediate	RS	SHI	CB
Subtract with Carry Halfword	RR	SCHR	0F
Subtract with Carry Halfword	RX	SCH	4F
Sense Status	RR	SSR	9D
Sense Status	RX	SS	DD
Write Block	RR	WBR	96
Write Block	RX	WB	D6
Write Data	RR	WDR	9A
Write Data	RX	WD	DA

SUMMARY OF USER INSTRUCTIONS - NUMERICAL BY OP CODE

OP CODE	TYPE	MNEMONIC	INSTRUCTION
01	RR	BALR	Branch and Link
02	RR	BTCR	Branch on True Condition
03	RR	BFCR	Branch on False Condition
04	RR	NHR	AND Halfword
05	RR	CLHR	Compare Halfword
06	RR	OHR	OR Halfword
07	RR	XHR	Exclusive OR Halfword
08	RR	LHR	Load Halfword
0A	RR	AHR	Add Halfword
0B	RR	SHR	Subtract Halfword
0C	RR	MHR	Multiply Halfword
0D	RR	DHR	Divide Halfword
0E	RR	ACHR	Add with Carry Halfword
0F	RR	SCHR	Subtract with Carry Halfword
40	RX	STH	Store Halfword
41	RX	BAL	Branch and Link
42	RX	BTC	Branch on True Condition
43	RX	BFC	Branch on False Condition
44	RX	NH	AND Halfword
45	RX	CLH	Compare Logical Halfword
46	RX	OH	OR Halfword
47	RX	XH	Exclusive OR Halfword
48	RX	LH	Load Halfword
4A	RX	AH	Add Halfword
4B	RX	SH	Subtract Halfword
4C	RX	MH	Multiply Halfword
4D	RX	DH	Divide Halfword
4E	RX	ACH	Add with Carry Halfword
4F	RX	SCH	Subtract with Carry Halfword
92	RR	STBR	Store Byte
93	RR	LBR	Load Byte
96	RR	WBR	Write Block
97	RR	RBR	Read Block
9A	RR	WDR	Write Data
9B	RR	RDR	Read Data
9D	RR	SSR	Sense Status
9E	RR	OCR	Output Command
9F	RR	AIR	Acknowledge Interrupt
C0	RS	BXH	Branch on Index High
C1	RS	BXLE	Branch on Index Low or Equal
C2	RX	LPSW	Load Program Status Word

OP CODE	TYPE	MNEMONIC	INSTRUCTION
C4	RS	NHI	AND Halfword Immediate
C5	RS	CLHI	Compare Logical Halfword Immediate
C6	RS	OHI	OR Halfword Immediate
C7	RS	XHI	Exclusive OR Halfword Immediate
C8	RS	LHI	Load Halfword Immediate
CA	RS	AHI	Add Halfword Immediate
CB	RS	SHI	Subtract Halfword Immediate
CC	RS	SRHL	Shift Right Logical
CD	RS	SLHL	Shift Left Logical
CE	RS	SRHA	Shift Right Arithmetic
CF	RS	SLHA	Shift Left Arithmetic
D2	RX	STB	Store Byte
D3	RX	LB	Load Byte
D6	RX	WB	Write Block
D7	RX	RB	Read Block
DA	RX	WD	Write Data
DB	RX	RD	Read Data
DD	RX	SS	Sense Status
DE	RX	OC	Output Command
DF	RX	AI	Acknowledge Interrupt

DISPLAY STATUS BYTE DATA

HEX ADDRESS 01

	Mode				Register Display			
	0	1	2	3	4	5	6	7
SINGLE	0	1	0	0				
RUN	1	0	0	0				
HALT	1	1	0	0				
MEM. WRITE	0	0	0	1				
MEM. READ	0	0	1	0				
ADRS	0	0	1	1				
OFF					0	0	0	0
ED DISPLAY					0	0	0	1
INSTR					0	0	1	0
PSW					0	1	0	0
R0, R1					1	0	0	0
R2, R3					1	0	0	1
R4, R5					1	0	1	0
R6, R7					1	0	1	1
R8, R9					1	1	0	0
R10, R11					1	1	0	1
R12, R13					1	1	1	0
R14, R15					1	1	1	1

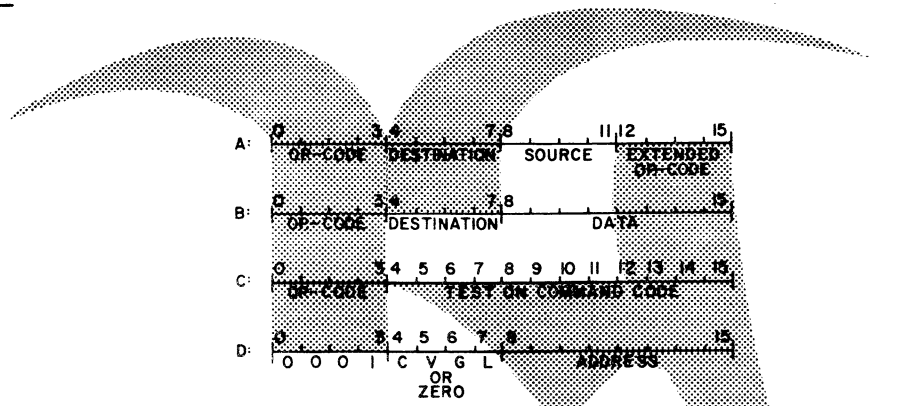
M O D E L 30-2
GENERAL REFERENCES

The following index lists reference material provided in this section.

CORE MEMORY ALLOCATION FOR GENERAL REGISTERS AND PROGRAM STATUS WORDS

Hexadecimal Memory Address	Register Assignment
Floating point Registers	
00 - 03	R0
04 - 07	R2
08 - 0B	R4
0C - 0F	R6
10 - 13	R8
14 - 17	R10
18 - 1B	R12
1C - 1F	R14
20 - 21	General Support Register
22 - 23	Pointer
24 - 25	Current PSW: Status and Condition Code
26 - 27	Current PSW: Instruction Address Counter
Program Status Words	
28 - 2B	Old PSW: Floating-point Divide fault Interrupt
2C - 2F	New PSW: Floating-point Divide fault Interrupt
30 - 33	Old PSW: Illegal Instruction Interrupt
34 - 37	New PSW: Illegal Instruction Interrupt
38 - 3B	Old PSW: Machine Malfunction Interrupt
3C - 3F	New PSW: Machine Malfunction Interrupt
40 - 43	Old PSW: External Device Interrupt
44 - 47	New PSW: External Device Interrupt
48 - 4B	Old PSW: Fixed-point Divide Fault Interrupt
4C - 4F	New PSW: Fixed-point Divide Fault Interrupt
50	First user available memory location

OP.-CODE	INSTR.	FORMAT
0000	DECODE	A
0001	BRANCH	D
0010	TEST	C
0011	COMMAND	C
0100	LOAD	A
0101	LOAD IMM.	B
0110	OR	A
0111	OR IMM.	B
1000	AND	A
1001	AND IMM.	B
1010	EX. OR	A
1011	EX. OR IMM.	B
1100	ADD	A
1101	ADD IMM.	B
1110	SUBTRACT	A
1111	SUB. IMM.	B



FUNCTION TESTED	BIT SET
I/O INTERRUPT	5
AUTO RESTART	6
CONSOLE INTERRUPT	7
CONSOLE SINGLE MODE	8
UTILITY FLIP-FLOP	9
MEMORY PARITY FAIL	10
PRIMARY POWER FAIL	11

COMMANDS	BITS SET
MULTIPLY	5
DIVIDE	4
REPEAT	4,5
READ MEMORY	7
WRITE MEMORY	6
WRITE MEMORY, PROTECTED	6,7
SET BANK *	8
RESET BANK *	9
TRIGGER BANK *	8,9
SET UTILITY *	10
RESET UTILITY *	11
TRIGGER UTILITY *	10,11
RESET MEMORY PARITY FAIL *	12
SET WAIT ALARM *	13
RESET WAIT ALARM *	14
POWER DOWN	15

* = FLIP-FLOPS

INSTR.	FORMAT	OP-CODE
ADD	A	1100
ADD IMM.	B	1101
AND	A	1000
AND IMM.	B	1001
BRANCH	D	0001
COMMAND	C	0011
DECODE	A	0000
EX. OR	A	1010
EX. OR IMM.	B	1011
LOAD	A	0100
LOAD IMM.	B	0101
OR	A	0110
OR IMM.	B	0111
SUBTRACT	A	1110
SUB. IMM.	B	1111
TEST	C	0010

DESTINATION/SOURCE

CODE	REGISTER
0000	MR0/RAH
0001	MR1/RAL
0010	MR2/YS
0011	MR3
0100	MR4
0101	MAR
0110	LOC
0111	PSW
1000	AR
1001	IR
1010	MDR
1011	FLR/IR4
1100	CNTR
1101	IO
1110	YD
1111	YDPI

THE FOLLOWING REGISTERS CANNOT BE GATED TO THE "B" BUS: AR, RAH, RAL, CNTR

TO ADDRESS MR0, MR1 OR MR2, THE BANK FLIP-FLOP MUST BE SET.

TO ADDRESS RAH, RAL OR YS, THE BANK FLIP-FLOP MUST BE RESET

FLR IS A DESTINATION ONLY.
IR4 IS A SOURCE ONLY.

EXTENDED OP.-CODE X = DON'T CARE STATE

FOR A,S,X,N,O:
1XXX DISABLE AR TO ALU
XIXX SET FLAGS
FOR A,S,X,N,O,L (NOT IO):
XXIX CARRY INTO ALU
XXXI CARRY OUT OF ALU

FOR "L" ONLY:
01XX SHIFT RIGHT
10XX SHIFT LEFT
11XX CROSS SHIFT
FOR "L" IO
XX01 ADDRESS/ACKNOWLEDGE
XX10 DATA AVAILABLE/DATA REQUEST
XX11 COMMAND/STATUS REQUEST

FOR "D" ONLY:
1XXX MEMORY READ
X1XX JAM FLR TO CCR
XXIX CLEAR FLR, CNTR, UT AND BANK
XXXI CHANGE PHASE

SUMMARY OF USER INSTRUCTIONS - ALPHABETICAL BY NAME

INSTRUCTION	TYPE	MNEMONIC	OP CODE
Acknowledge Interrupt	RR	AIR	9F
Acknowledge Interrupt	RX	AI	DF
Add Halfword	RR	AHR	0A
Add Halfword	RX	AH	4A
Add Halfword Immediate	RS	AHI	CA
Add Floating Point	RR	AER	2A
Add Floating Point	RX	AE	6A
Add with Carry Halfword	RR	ACHR	0E
Add with Carry Halfword	RX	ACH	4E
AND Halfword	RR	NHR	04
AND Halfword	RX	NH	44
AND Halfword Immediate	RS	NHI	C4
Auto Load	RX	AL	D5
Branch and Link	RR	BALR	01
Branch and Link	RX	BAL	41
Branch on False Condition	RR	BFCR	03
Branch on False Condition	RX	BFC	43
Branch on True Condition	RR	BTCR	02
Branch on True Condition	RX	BTC	42
Branch on Index Low or Equal	RS	BXLE	C1
Branch on Index High	RS	BXH	C0
Branch Unconditional	RR	BR	030
Branch Unconditional	RX	B	430
Branch on Overflow	RX	BO	424
Branch on Zero	RX	BZ	433
Branch on Not Zero	RX	BNZ	423
Branch on Equal	RX	BE	433
Branch on Not Equal	RX	BNE	423

INSTRUCTION	TYPE	MNEMONIC	OP CODE
Branch on Plus	RX	BP	422
Branch on Not Plus	RX	BNP	432
Branch on Low	RX	BL	428
Branch on Not Low	RX	BNL	438
Branch on Minus	RX	BM	421
Branch on Not Minus	RX	BNM	431
Branch on Carry	RX	BC	428
Compare Floating Point	RR	CER	29
Compare Floating Point	RX	CE	69
Compare Logical Halfword	RR	CLHR	05
Compare Logical Halfword	RX	CLH	45
Compare Logical Halfword Immediate	RS	CLHI	C5
Divide Floating Point	RR	DER	2D
Divide Floating Point	RX	DE	6D
Divide Halfword	RR	DHR	0D
Divide Halfword	RX	DH	4D
Exclusive OR Halfword	RR	XHR	07
Exclusive OR Halfword	RX	XH	47
Exclusive OR Halfword Immediate	RS	XHI	C7
Load Byte	RR	LBR	93
Load Byte	RX	LB	D3
Load Floating Point	RR	LER	28
Load Floating Point	RX	LE	68
Load Halfword	RR	LHR	08
Load Halfword	RX	LH	48
Load Halfword Immediate	RS	LHI	C8
Load Multiple	RX	LM	D1
Load Program Status Word	RX	LPSW	C2
Multiply Floating Point	RR	MER	2C
Multiply Floating Point	RX	ME	6C

INSTRUCTION	TYPE	MNEMONIC	OP CODE
Multiply Halfword	RR	MHR	0C
Multiply Halfword	RX	MH	4C
No Operation	RR	NOPR	020
No Operation	RX	NOP	420
OR Halfword	RR	OHR	06
OR Halfword	RX	OH	46
OR Halfword Immediate	RS	OHI	C6
Read Block	RR	RBR	97
Read Block	RX	RB	D7
Read Data	RR	RDR	9B
Read Data	RX	RD	DB
Output Command	RR	OCR	9E
Output Command	RX	OC	DE
Shift Left Arithmetic	RS	SLHA	CF
Shift Left Logical	RS	SLHL	CD
Shift Right Arithmetic	RS	SRHA	CE
Shift Right Logical	RS	SRHL	CC
Store Byte	RR	STBR	92
Store Byte	RX	STB	D2
Store Floating Point	RX	STE	60
Store Halfword	RX	STH	40
Store Multiple	RX	STM	D0
Subtract Halfword	RR	SHR	0B
Subtract Halfword	RX	SH	4B
Subtract Halfword Immediate	RS	SHI	CB
Subtract Floating Point	RR	SER	2B
Subtract Floating Point	RX	SE	6B
Subtract with Carry Halfword	RR	SCHR	0F
Subtract with Carry Halfword	RX	SCH	4F

INSTRUCTION	TYPE	MNEMONIC	OP CODE
Sense Status	RR	SSR	9D
Sense Status	RX	SS	DD
Write Block	RR	WBR	96
Write Block	RX	WB	D6
Write Data	RR	WDR	9A
Write Data	RX	WD	DA

SUMMARY OF USER INSTRUCTIONS - NUMERICAL BY OP CODE

OP CODE	TYPE	MNEMONIC	INSTRUCTION
01	RR	BALR	Branch and Link
02	RR	BTCR	Branch on True Condition
03	RR	BFCR	Branch on False Condition
04	RR	NHR	AND Halfword
05	RR	CLHR	Compare Halfword
06	RR	OHR	OR Halfword
07	RR	XHR	Exclusive OR Halfword
08	RR	LHR	Load Halfword
0A	RR	AHR	Add Halfword
0B	RR	SHR	Subtract Halfword
0C	RR	MHR	Multiply Halfword
0D	RR	DHR	Divide Halfword
0E	RR	ACHR	Add with Carry Halfword
0F	RR	SCHR	Subtract with Carry Halfword
28	RR	LER	Load Floating Point
29	RR	CER	Compare Floating Point
2A	RR	AER	Add Floating Point
2B	RR	SER	Subtract Floating Point
2C	RR	MER	Multiply Floating Point
2D	RR	DER	Divide Floating Point
40	RX	STH	Store Halfword
41	RX	BAL	Branch and Link
42	RX	BTC	Branch on True Condition
43	RX	BFC	Branch on False Condition
44	RX	NH	AND Halfword
45	RX	CLH	Compare Logical Halfword
46	RX	OH	OR Halfword
47	RX	XH	Exclusive OR Halfword
48	RX	LH	Load Halfword
4A	RX	AH	Add Halfword
4B	RX	SH	Subtract Halfword
4C	RX	MH	Multiply Halfword
4D	RX	DH	Divide Halfword
4E	RX	ACH	Add with Carry Halfword
4F	RX	SCH	Subtract with Carry Halfword
60	RX	STE	Store Floating Point
68	RX	LE	Load Floating Point
69	RX	CE	Compare Floating Point
6A	RX	AE	Add Floating Point
6B	RX	SE	Subtract Floating Point
6C	RX	ME	Multiply Floating Point
6D	RX	DE	Divide Floating Point
92	RR	STBR	Store Byte
93	RR	LBR	Load Byte

OP CODE	TYPE	MNEMONIC	INSTRUCTION
96	RR	WBR	Write Block
97	RR	RBR	Read Block
9A	RR	WDR	Write Data
9B	RR	RDR	Read Data
9D	RR	SSR	Sense Status
9E	RR	OCR	Output Command
9F	RR	AIR	Acknowledge Interrupt
C0	RS	BXH	Branch on Index High
C1	RS	BXLE	Branch on Index Low or Equal
C2	RX	LPSW	Load Program Status Word
C4	RS	NHI	AND Halfword Immediate
C5	RS	CLHI	Compare Logical Halfword Immediate
C6	RS	OHI	OR Halfword Immediate
C7	RS	XHI	Exclusive OR Halfword Immediate
C8	RS	LHI	Load Halfword Immediate
CA	RS	AHI	Add Halfword Immediate
CB	RS	SHI	Subtract Halfword Immediate
CC	RS	SRHL	Shift Right Logical
CD	RS	SLHL	Shift Left Logical
CE	RS	SRHA	Shift Right Arithmetic
CF	RS	SLHA	Shift Left Arithmetic
D0	RX	STM	Store Multiple
D1	RX	LM	Load Multiple
D2	RX	STB	Store Byte
D3	RX	LB	Load Byte
D5	RX	AL	Autoload
D6	RX	WB	Write Block
D7	RX	RB	Read Block
DA	RX	WD	Write Data
DB	RX	RD	Read Data
DD	RX	SS	Sense Status
DE	RX	OC	Output Command
DF	RX	AI	Acknowledge Interrupt

DISPLAY STATUS BYTE DATA

HEX ADDRESS 01

	Mode				Register Display			
	0	1	2	3	4	5	6	7
Variable	0	1	0	0				
Floating Variable	0	1	1	0				
RUN	1	0	0	0				
HALT	1	1	0	0				
Floating HALT	1	1	1					
MEM. WRITE	0	0	0	1				
MEM. READ	0	0	1	0				
ADRS	0	0	1	1				
OFF					0	0	0	0
ED DISPLAY					0	0	0	0
INSTR					0	0	1	0
PSW					0	1	0	0
R0, R1					1	0	0	0
R2, R3					1	0	0	1
R4, R5					1	0	1	0
R6, R7					1	0	1	1
R8, R9					1	1	0	0
R10, R11					1	1	0	1
R12, R13					1	1	1	0
R14, R15					1	1	1	1

TELETYPEWRITER STATUS AND COMMAND BYTE DATA
HEX ADDRESS 02

BIT NUMBER	0	1	2	3	4	5	6	7
STATUS BYTE			BRK		BSY	EX		DU
COMMAND BYTE	DISABLE	ENABLE	UNBLOCK	BLOCK	WRT	READ	PWR ON	PWR OFF

BRK — The Break bit is set when the Break key on the teletypewriter is depressed.

BSY — The significance of the Busy bit depends upon whether a Read or a Write operation is in progress. During Write mode, BSY is normally low, and goes high only while data is being received by the device. During Read mode, BSY is normally high, and goes low only when data has been received from the device, but not yet been transferred to the Processor. During Read mode, BSY goes high again as soon as the Processor accepts the data.

EX — The Examine bit is set whenever BRK is set.

DU — The Device Unavailable bit is set whenever the teletypewriter power is off, or the teletypewriter is in LOCAL mode.

DISABLE — This command disables the Device Interrupt.

ENABLE — This command enables the Device Interrupt.

UNBLOCK — This command enables the printer to print data entered via either the keyboard or the tape reader.

BLOCK — This command disables the feature described above.

WRT, READ — The Write and Read commands are used to define the significance of the BSY bit.

PWR ON — This command applies power to the Teletypewriter.

PWR OFF — This command removes power from the Teletypewriter.

HIGH SPEED PAPER TAPE READER STATUS AND COMMAND BYTE DATA
HEX ADDRESS 03

BIT NUMBER	0	1	2	3	4	5	6	7
STATUS BYTE	OVERFLOW			NMTN	BSY	EX		DU
COMMAND BYTE	DISABLE	ENABLE	STOP	RUN	INCR	SLEW	REV	FWD

OVERFLOW — The Overflow bit is available for use with paper tape readers which operate in the Slew mode. The bit is set if the next character is read before a Data Request (DR) is received for the present character.

NMTN — The No Motion bit is set any time the tape is not moving

BSY — The Busy bit is set anytime there is a character in the buffer and no Data Request (DR) has been received from the Processor.

EX — The Examine bit is set whenever either Overflow or NMTN is set.

DU — The Device Unavailable bit is set if Reader Power is off, or if the LOAD/READY lever on the reader is in the LOAD position.

DISABLE — This command disables the Device Interrupt.

ENABLE — This command enables the Device Interrupt.

STOP — The Stop command stops reader tape motion.

RUN — The Run command starts the reader tape motion.

INCR — The Increment command directs the reader to read in Increment mode. The tape is stepped to the next character after each character is input to the Processor.

SLEW — The Slew command applies only to readers capable of operation in the Slew mode. In Slew mode the tape is started and continues to run until a particular character or string of characters on the tape is sensed.

REV — The Reverse command applies only to bi-directional tape readers.

FWD — The Forward command directs the reader to move the tape forward.

CARD READER STATUS AND COMMAND BYTE DATA
HEX ADDRESS 04

BIT NUMBER	0	1	2	3	4	5	6	7
STATUS BYTE	OVERFLOW			NMTN	BSY	EX	EOM	DU
COMMAND BYTE	DISABLE	ENABLE	FEED					

OVERFLOW — The Overflow bit is set when a new column is strobed, and data from a previous column has not yet been transferred to the Processor. This bit is reset by the next FEED command.

BSY — The Busy bit is set when the controller is preparing data for transfer, but the data is not yet available.

EX — The Examine bit is set whenever either NMTN or OVERFLOW is set.

NMTN
EOM — Both the No Motion and the End Of Medium bits are set except for the time between a FEED command and the time the end of the card passes the reader photo cells.

DU — The Device Unavailable bit is set if the POWER or the FEED switch is OFF.

DISABLE — This command disables the Data Available Interrupt.

ENABLE — This command enables the Data Available Interrupt.

FEED — This command initiates a new card feed cycle.

MINI-TAPE STATUS AND COMMAND BYTE DATA
(Hex Address 06)

BIT NUMBER	0	1	2	3	4	5	6	7
STATUS BYTE	ERR	BOT	EOT	NMTN	BSY	EX	EOM	DU
COMMAND BYTE	DISABLE	ENABLE	STOP	RUN	INC	SLEW	WRITE	READ

ERR - The Error bit is set if:

1. A Parity read error is detected by the tape unit while reading.
2. The characters are not taken from interface before the next character arrives during Slew Read Mode.
3. The characters do not arrive at the interface fast enough during Slew Write Mode.

BOT - The BOT bit is set when the BOT TAB is Detected.

EOT - The EOT bit is set when the EOT TAB is Detected.

NMTN - The No Motion bit is set when the RUN flip-flop is Reset and tape unit is Ready. The bit is reset:

1. While the tape is in process of rewinding
2. While the interface is in the RUN Mode
3. When the tape unit is not Ready

BSY - Busy is set:

1. In Read Mode, when the interface is waiting for a character from the tape unit.
2. In Write Mode, when the tape unit is in the Process of Recording a character.

3. In Stop Mode, if the tape unit was given a Rewind Command.

Busy is reset:

1. In the Read Mode, if the interface is ready to transfer a character to the Processor.

2. In the Write Mode, if the tape unit is ready to accept another character.

EX - The Examine bit is set to indicate that one of the four most significant bits of the status byte is set.

EOM - The End of Medium bit is set when the tape unit is in the Read Mode and an END OF RECORD gap is detected. It is set while over the BOT tab.

DU - The Device Unavailable bit is set to indicate that the tape unit power is down, or that the tape unit is not set up properly (door open, tape not loaded, etc.).

DIS - Disables device interrupts

EN - Enables device interrupts

STOP - END recording or reading

RUN - START recording or reading

INC - Puts tape unit in the Incremental Mode

SLEW - Puts tape unit in the High Speed Mode

WRITE - Puts tape unit in the Write Mode

READ - Takes tape unit out of the Write Mode

The Command lines are subject to coding; if a code is generated, the individual lines associated with the bits are not activated:

CODE	FUNCTION
$2 \cdot 3 \cdot \overline{4}$	Rewind
$2 \cdot 3 \cdot 4$	Master Reset
$0 \cdot 1 \cdot \overline{2}$	Select Chan A
$0 \cdot 1 \cdot 2$	Select Chan B

TELETYPEWRITER/ASCII/HEX CONVERSION TABLE

HEX (MSD) →					8	9	A	B	C	D	E	F
(LSD) ↓	Teletypewriter Tape Channels →			8	Depends upon parity							
				7	0	0	0	0	1	1	1	1
				6	0	0	1	1	0	0	1	1
				5	0	1	0	1	0	1	0	1
	4	3	2	1								
ø	0	0	0	0	NULL	DC_0	SPACE	0	@	P		
1	0	0	0	1	SUM	X-ON	!	1	A	Q		
2	0	0	1	0	EOA	TAPE ON	"	2	B	R		
3	0	0	1	1	EOM	X-OFF	#	3	C	S		
4	0	1	0	0	EOT	TAPE OFF	$	4	D	T		
5	0	1	0	1	WRU	ERR	%	5	E	U		
6	0	1	1	0	RU	SYNC	&	6	F	V		
7	0	1	1	1	BELL	LEM	'	7	G	W		
8	1	0	0	0	FE_0	S_0	(	8	H	X		
9	1	0	0	1	HT/SK	S_1	)	9	I	Y		
A	1	0	1	0	LF	S_2	*	:	J	Z		
B	1	0	1	1	VT	S_3	+	;	K	[		
C	1	1	0	0	FF	S_4	,	<	L	\		ACK
D	1	1	0	1	CR	S_5	-	=	M	]		ALT. MODE
E	1	1	1	0	SO	S_6	.	>	N	↑		ESC
F	1	1	1	1	SI	S_7	/	?	O	←		DEL

ASCII/CARD CODE CONVERSION TABLE

GRAPHIC	8-BIT ASCII CODE	7-BIT ASCII CODE	CARD CODE	GRAPHIC	8-BIT ASCII CODE	7-BIT ASCII CODE	CARD CODE
SPACE	A0	20	0-8-2	@	C0	40	8-4
!	A1	21	12-8-7	A	C1	41	12-1
"	A2	22	8-7	B	C2	42	12-2
#	A3	23	8-3	C	C3	43	12-3
$	A4	24	11-8-3	D	C4	44	12-4
%	A5	25	0-8-4	E	C5	45	12-5
&	A6	26	12	F	C6	46	12-6
'	A7	27	8-5	G	C7	47	12-7
(	A8	28	12-8-5	H	C8	48	12-8
)	A9	29	11-8-5	I	C9	49	12-9
*	AA	2A	11-8-4	J	CA	4A	11-1
+	AB	2B	12-8-6	K	CB	4B	11-2
,	AC	2C	0-8-3	L	CC	4C	11-3
-	AD	2D	11	M	CD	4D	11-4
.	AE	2E	12-8-3	N	CE	4E	11-5
/	AF	2F	0-1	O	CF	4F	11-6
0	B0	30	0	P	D0	50	11-7
1	B1	31	1	Q	D1	51	11-8
2	B2	32	2	R	D2	52	11-9
3	B3	33	3	S	D3	53	0-2
4	B4	34	4	T	D4	54	0-3
5	B5	35	5	U	D5	55	0-4
6	B6	36	6	V	D6	56	0-5
7	B7	37	7	W	D7	57	0-6
8	B8	38	8	X	D8	58	0-7
9	B9	39	9	Y	D9	59	0-8
:	BA	3A	8-2	Z	DA	5A	0-9
;	BB	3B	11-8-6	[	DB	5B	12-8-2
<	BC	3C	12-8-4	\	DC	5C	11-8-1
=	BD	3D	8-6	]	DD	5D	11-8-2
>	BE	3E	0-8-6	↑	DE	5E	11-8-7
?	BF	3F	0-8-7	←	DF	5F	0-8-5

SYSTEM TEST SET INSTRUCTION MANUAL

TABLE OF CONTENTS

ILLUSTRATIONS

SYSTEM TEST SET INSTRUCTION MANUAL

Publication Number 29-012

1. INTRODUCTION

The Model 28-001 System Test Set (often referred to as a MOTS Box) is a maintenance test device which permits manual control of a GE-PAC 30 Digital System. Indicators on the Test Set display the contents of all pertinent registers and buses within the system. The System Test Set may be used in maintaining systems with standard Read-Only-Memories (ROMs), and is required if a special maintenance ROM (X-Ray ROM) is used. The plug-in X-Ray ROM is substituted for the normal ROM in the system under test. X-Ray ROM details are provided in separate publications. The System Test Set or System Test Set/X-Ray ROM combination permits rapid check-out and/or trouble analysis of a GE-PAC 30 Digital System. The System Test Set is compatible with all GE-PAC 30 Digital Systems. Unless otherwise noted, all comments in this manual apply equally to all Digital System Models.

2. SET-UP PROCEDURE

Use the following procedure to connect the System Test Set to the Digital System.

CAUTION

Remove power from the Digital System and disconnect it from its power source before proceeding.

1. Carefully remove the cable assemblies from their storage compartment in the System Test Set.

2. Connect the P1 through P3 cable connectors to the J1 through J3 connectors on the rear of the Test Set. Refer to Figure 1.

3. Connect the other end of the cables to the back panel of the Digital System. Refer to Figure 1.

4. The System Test Set derives its power from the Digital System. Connect the power cable as follows:

Voltage	Wire Color	Location[1]	Use
+15	Orange	TB4-2	Memory Voltage Reference
+ 5	Red	TB5-3	Lamp Supply
GRD	Black	TB5-6	Lamp Ground

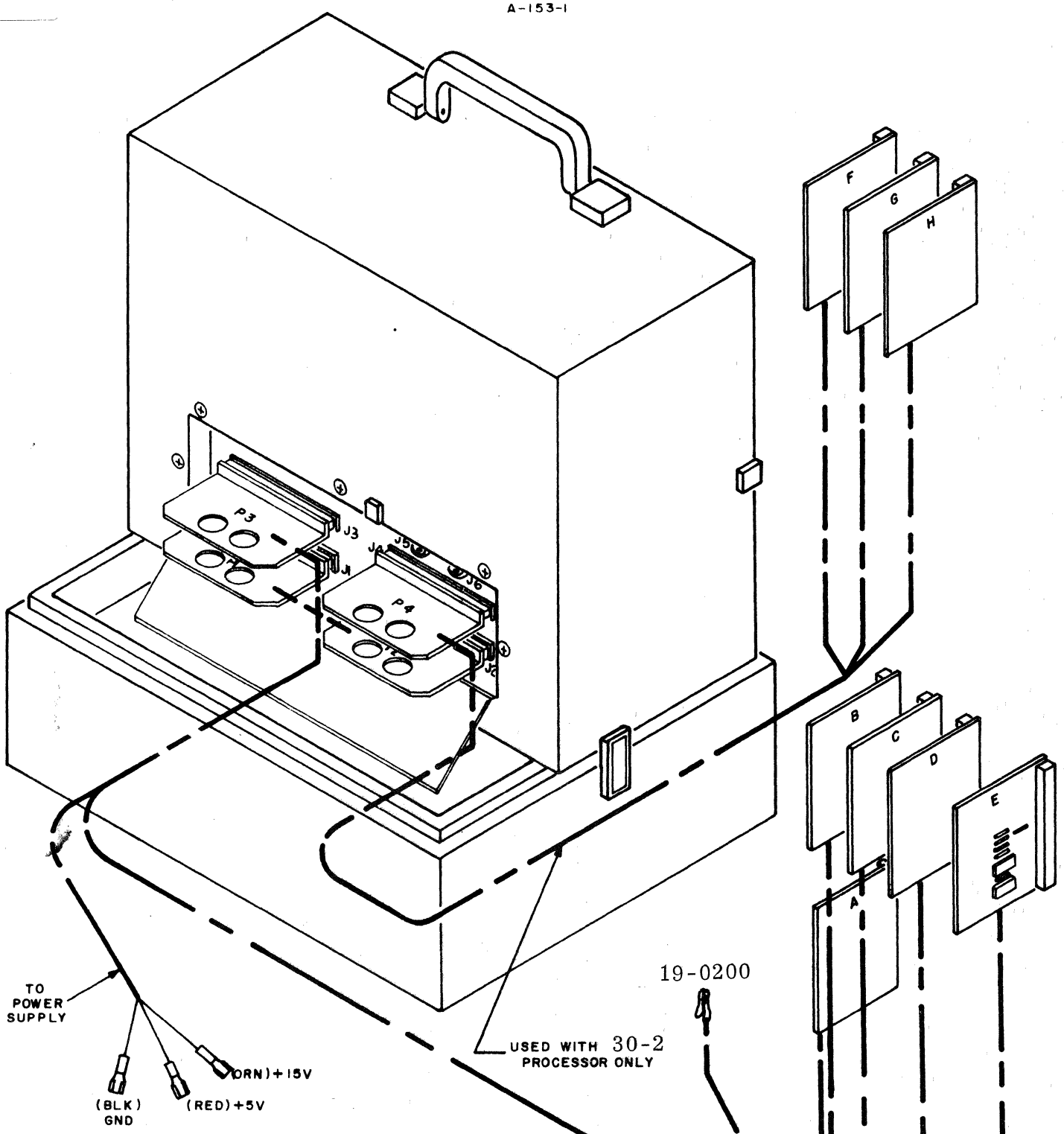
5. Clip the final lead (used for the memory threshold adjustment) to the back panel as follows:

GE-PAC 30-1 and -2 - 119-0200

6. Check that all connectors are mated properly and apply power to the system.

---

1. The locations provided are on the vendor power supply. If a GE-PAC 30 power supply is provided with the system, use points on the back panel. The supply voltages are marked on the back panel.



	CONNECTOR							
MODEL	A	B	C	D	E	F	G	H
30-01	RMI-0 ROWS 1,2	RMI-1 ROWS 1,2	BS1-1 ROWS 1,2	BS2-1 ROWS 1,2	CO2-1 ROWS 0,1	NA	NA	NA
30-01	RMI-0 ROWS 1,2	RMI-1 ROWS 1,2	BS1-1 ROWS 1,2	BS2-1 ROWS 1,2	CO2-1 ROWS 0,1	NA	NA	NA
30-02	RMI-0 ROWS 1,2	RMI-1 ROWS 1,2	BS3-1 ROWS 1,2	BS2-1 ROWS 1,2	CO1-1 ROWS 0,1	CO2-1 ROWS 1,2	BS1-1 ROWS 1,2	BS0-1 ROWS 1,2

Figure 1. System Test Set Cabling

3. CONTROLS AND INDICATORS

Each of the controls and indicators on the System Test Set control panel is described in the following paragraphs. Refer to Figure 2.

3.1 Controls

MEM VOLT — The MEMory VOLTage potentiometer is used to vary the bias in the core memory. The potentiometer is connected only when the MEM Switch is depressed.

EXT CLK — The EXTernal CLocK potentiometer is used to vary the frequency of a test clock which may be substituted for the system clock. The potentiometer is connected only when the EXT Switch is depressed.

MEM — When depressed, the MEMory Switch enables the MEM VOLT potentiometer which is used to test the bias margins in the core memory. This switch should normally be released - if not, marginal core memory operation may result.

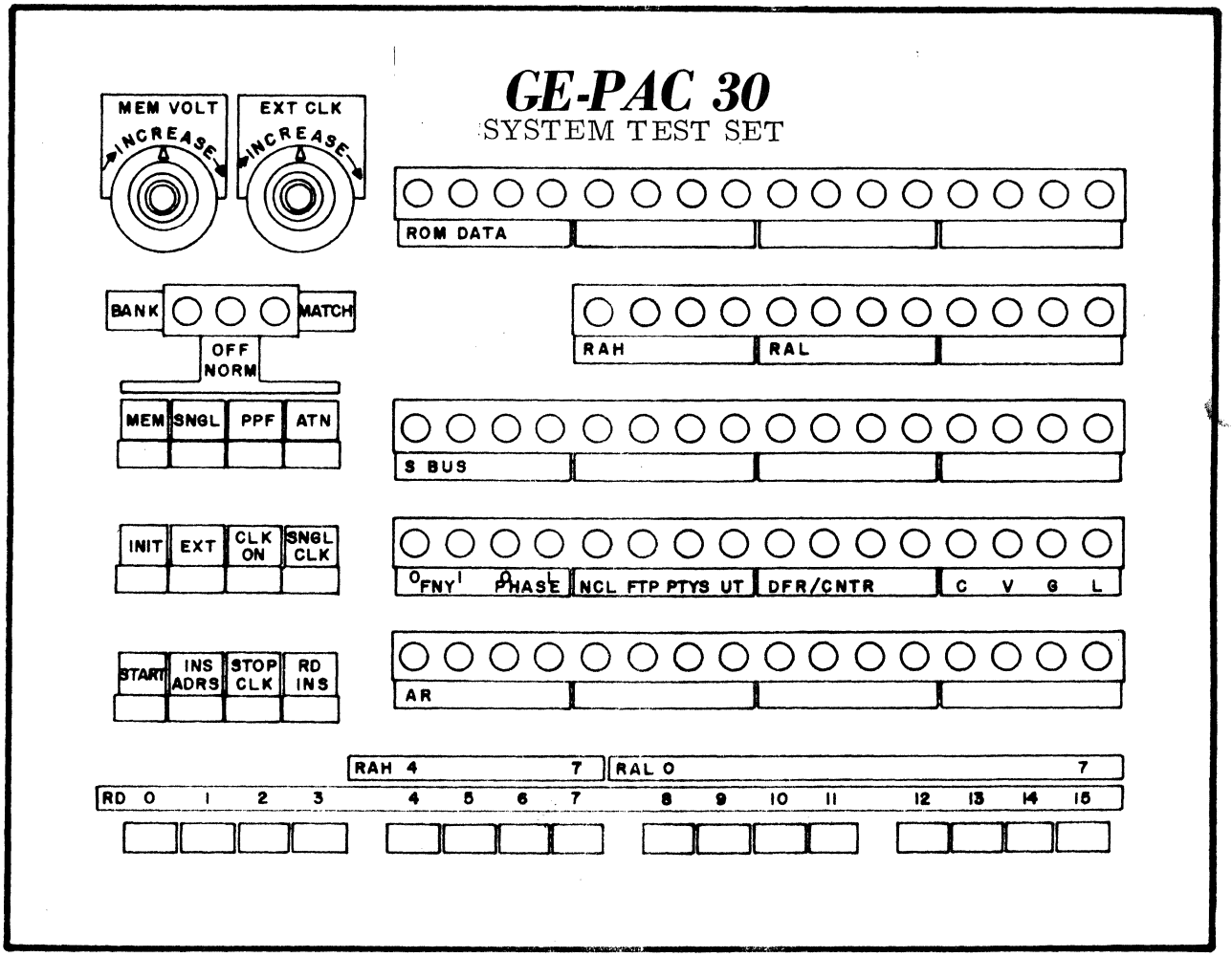


Figure 2. Control Panel Layout

ATN, PPF, and SNGL — These switches are used as sense switches for the X-Ray. (See Section 4.) If an X-Ray ROM is not in place in the computer, these switches should all be released. The OFF NORMal lamp lights if any are depressed.

INIT — The INITialize Switch performs the same function as the INITIALIZE Switch on the front panel, the Initialize relay in the Processor is released. This stops the clock momentarily, and clears the ROM address register and other flip-flops in the system.

EXT — When depressed, the EXTernal Switch enables the clock frequency to be controlled by the EXT CLK potentiometer. If the switch is released, the crystal-controlled clock in the Processor is enabled. This switch should not be operated while the clock is running.

CLK ON — Depressing the CLocK ON Switch, followed by depressing START, causes the system clock to run. The clock may be stopped by releasing the CLocK ON Switch, or on an address match when STOP CLocK is depressed.

SNGL CLK — If the clock is off, depressing the SiNGLe CLocK Switch generates a single clock pulse, to execute one micro-instruction at a time.

START — Depressing the START Switch permits the clock to start running if CLK ON is depressed. The first micro-instruction to be executed is the instruction currently in the ROM data register.

INH ADRS — Depressing the INHibit ADdReSs Switch stops the ROM address register clock when the Data Switches match the ROM address. Note that the Processor clock is still running and ROM readouts are still strobed into the ROM DATA register even though the ROM address register clock is stopped. This micro-instruction will be executed repeatedly.

STOP CLK — Depressing the STOP CLocK Switch enables a continuous match between the ROM address register and the 12 least significant bits of the Data Switches. The Processor clock is stopped when a match is detected.

RD INS — Operation of the ReaD INStruction momentary Switch causes the contents of the Data Switches to be loaded into the ROM DATA Register. This may be observed on the ROM DATA lamps. The switch is disabled when the clock is running, i.e. CLK ON depressed.

Data Switches — These 16 Data Switches, across the bottom of the panel, are used to perform a dual function:

1. The data set into these switches is inserted into the ROM DATA Register when RD INS is depressed with the clock stopped.

- or -

2. The right-most 12 bits may be used as a constant to match against the ROM address register.

3.2 Indicators

The indicators light when the associated bit or flip-flop is set.

ROM DATA — The 16 ROM DATA lamps indicate the contents of the last address read from the ROM.

BANK — Indicates the state of the Bank flip-flop in 30-1 or 30-2 Digital Systems.

OFF NORMAL — Indicates that at least one of the following switches are depressed: MEM, SNGL, PPF, or ATN.

MATCH — Indicates a match condition between the 12-bit ROM Address Register and Data Switches.

RAH — These lamps monitor the 4 most significant bits of the second rank of the ROM Address Register.

RAL — These lamps monitor the 8 least significant bits of ROM Address Register. Ordinarily, these lamps display the address of the next instruction to be executed.

S BUS — Indicates the contents of the S Bus. For the 30-1, only the right-most 8 bits are applicable. For the 30-2, all bits are used.

DFR/IF — For the 30-1, the right-most 8 bits display the contents of DFR. For the 30-2, the right-most 4 bits display the contents of the Flag Register; C, V, G, and L. The next 4 bits display the state of the Counter Register.

The remaining 8 bits in this row of lamps are used for 30-2 Systems only. Eight flip-flops are displayed as indicated on the display panel.

AR — Indicates the contents of the AR. For the 30-1, only the right-most 8 bits are applicable. For the 30-2, all 16 bits are used.

4. OPERATING PROCEDURES

The following procedures assume that an X-Ray ROM is used in conjunction with the Test Set.

4.1 Initial Start-Up

1. Connect the Systems Test Set as outlined in Section 2.
2. Before power is applied to the system, carefully remove the ROM mother-board and substitute the X-Ray ROM.
3. Slide the Display Controller mother-board out of the card file far enough to disconnect the back panel connectors. It is not necessary to remove the cables from the front of the Display Controller.
4. Release all switches on the Test Set except CLK ON. Depress CLK ON.

5. Turn the Digital System power on.

6. Depress the momentary START Switch. This starts the X-Ray running.

4.2 ROM Program Execution

The ROM program may be started at any address by using the following procedure to load the starting address of the ROM subroutine into the ROM Address Register.

1. Release the CLK ON Switch to stop the system clock.

2. Depress INIT to initialize the system.

3. Insert X'5D' (Load RAH) on Data Switches 0 through 7.

4. Set Data Switches 8 through 15 to the desired RAH address.

5. Depress the RD INS pushbutton and the SNGL CLK pushbutton to load the RAH Master Register.

6. Insert X'5E' (Load RAL) on Data Switches 0 through 7.

7. Set Data Switches 8 through 15 to the desired RAL address.

8. Depress the RD INS pushbutton and the SNGL CLK pushbutton to load the RAL Register and the RAH Slave Register. The starting address is now displayed on the RAH and RAL indicators.

9. Depress SNGL CLK. The system fetches the micro-instruction at the selected address and executes that instruction.

The system may be stepped through each micro-instruction by continuing to depress SNGL CLK, or CLK ON may be depressed to run the micro-program at normal speed.

4.3 Address Match

The System Test Set permits halting the system in either of two modes when the micro-program reaches a designated address. If an address is selected and STOP CLK is depressed, the system executes the instruction at the selected address, and then fetches the next instruction, but does not execute it. If, instead of STOP CLK, INH ADRS is depressed, the ROM Address Register is frozen at the selected address and the instruction at that address is executed repeatedly. Use the following procedure for address matching:

1. Set the desired match address on the Data Switches.

   NOTE

   Do not select an address at which the ROM micro-instruction is a Load RAL or an executed Branch.

2. Depress STOP CLK or INH ADRS depending upon the mode of the operation desired.

3. Resume operation after a match by simply releasing INH ADRS, or by releasing STOP CLK and depressing START.

4.4 X-Ray Control

When running the X-Ray ROM program, it is often desirable to have the program continuously loop through certain tests, or to loop through all X-Rays and only stop on errors. This is accomplished by using the PPF, ATN, and SNGL Switches as sense switches which are interrogated by the X-Ray program after a set of tests. The function of each switch is:

1. SNGL - When SNGL is released, the X-Ray continuously loops through the Processor X-Rays and the Memory X-Ray.

When SNGL is depressed, the X-Ray ROM loops in the Wait loop after each set of tests.

2. ATN - When ATN is depressed, the X-Ray loops in one set of tests. The operator selects the particular set of tests by stepping the system through the X-Rays with the EXECUTE pushbutton. Just prior to executing the test to be repeated, the ATN pushbutton is depressed.

3. PPF - If the PPF Switch is depressed, the micro-program loops in the Wait loop if the X-Ray detects an error. The AR indicators also display an error pointer code at this time. An error pointer code dictionary is provided at the end of Appendix 1 to this manual.

If the program is looping in the Wait loop as a result of the PPF and/or SNGL Switches, the program may be continued by depressing the EXECUTE pushbutton on the Display Panel twice.

The previous descriptions of the SNGL, ATN, and PPF Switch operation are applicable for the Processor X-Rays (START 0 through START 6), and for the Memory X-Ray (START 10). The START 7 X-Ray tests the Primary Power Fail circuit and the False Sync logic. During START 7, the SNGL, ATN, and PPF Switches have a different significance. To begin execution of the START 7 X-Ray, depress PPF and then INIT. If the SNGL Switch is depressed, the program loops at the end of the test (though not in the Wait loop). If the SNGL Switch is released, the micro-program continuously repeats START 7.

4.5 Core Memory Marginal Test

Use the following procedure to determine the core memory threshold margins:

1. Continuously loop through the START 10 X-Ray with the PPF Switch depressed to assure that no errors are detected.

2. Connect a voltmeter between ground and the VT testpoint on the Digital System back panel (pin 211-1 of MEM0 or MEM1).

3. Set the MEM VOLT potentiometer to its mid position and depress the MEM Switch.

4. Vary the MEM VOLT potentiometer to obtain readings from 3 volts to 5 volts. If the Memory X-Ray fails between these limits, the memory requires adjustment or repair. Refer to the Memory Section of the Digital System Maintenance Manual.

4.6 System Clock Marginal Test

If a trouble exists in the system, and it is suspected that the problem is related to speed, the variable clock on the System Test Set may be used as an aid to isolate the problem.

Use the following procedure to employ this feature.

1. Monitor the CD0 test point on the Digital System back panel with an oscilloscope.

2. Start the X-Ray running through all Processor tests, SNGL and ATN Switches released, PPF Switch depressed.

3. Release CLK ON, and PPF.

4. Turn the EXT CLK potentiometer fully counter-clockwise.

5. Operate the EXT, CLK ON, INIT, and PPF Switches.

When START is operated, the X-Ray program runs with the clock at approximately .2MHZ. If the program runs at this speed, increase the clock speed in discrete steps until a failure occurs. Five or ten seconds should be allowed between each setting to insure that the X-Ray completes at least 1 cycle.

NOTE

The clock in the Test Set is variable over a wide range and can be adjusted to be much faster than the computer crystal frequency. The crystal frequency should not be exceeded by more than 10% when running this test. The X-Rays should normally be run with the EXT Switch released.

5. FUNCTIONAL DESCRIPTION

The following description refers to the ten sheet System Test Set schematic provided in Appendix 1 at the rear of this manual.

Sheet 1 of the schematic shows the A Section of each of the Data/Address switches on the bottom of the System Test Set panel, and the interconnection of the switches to the Processor back panel. As shown on Figure 1 earlier in this manual, J1 and J2 are on the rear of the System Test Set. Connections P1 and P2 are on the cable assemblies, while the diodes shown enclosed within dotted lines are mounted on the connectors which mate with the back panel (RMI-0 and RMI-1). Note the momentary (non-latching) Read Instruction switch contacts shown in the lower left area of Sheet 1. When RD INS Switch is depressed, the condition of switches S00 through S15 is gated into the ROM Data Register.

Sheet 2 of the schematic shows the ROM DATA lamps on the System Test Set Control Panel, and the gates which drive them. Note that there is not gating envolved. Thus, the ROM DATA lamps continuously display the contents of the RD Register in the Digital System.

Sheet 3 shows the least significant 8 bits of the AR display and the S Bus display. Sheet 4 shows the most significant 8 bits of these displays, which are required only with 30-2 Digital Systems.

Sheet 5 shows part of the cable which is used with the 30-2. The sheet provides the Instruction Register display in such systems. Note the output from area R9 of this sheet to 6R9. This ground inhibits the inputs shown on the top right of Sheet 6.

The left half of Sheet 6 shows the RAH display logic. The four outputs to Sheet 8 are high when the RAH bit matches the corresponding S4 through S7 switch condition. For example, if S4 is depressed and RAH04 is set, the MRAH041 signal is high. The right half of Sheet 6 shows the DFR display logic if the Digital System is a 30-1 (see the preceding paragraph if the Digital System is a 30-2).

Sheet 7 shows the RAL display logic. Note that outputs to the address match logic on Sheet 8 are similar to those shown on Sheet 6 and described earlier in this Section.

The ROM address match logic is shown on the left side of Sheet 8. When the ROM address matches the configuration set in Switches 4 through 15, the output from the gate shown in area C5 goes high. This output is applied to four gates shown in area C6 through C9. Going from the top to bottom, the first gate lights the MATCH indicator on the Control Panel. If the INH ADRS Switch on the Control Panel is set, the next gate sends a signal to the Processor to inhibit incrementing the ROM address. If the STOP CLK Switch on the Control Panel is set, the output from the next gate stops the System Test Set clock (Sheet 9). The last gate provides an output to the ADD MATCH jack on the rear of the System Test Set. This output is typically used to trigger an oscilloscope at a selected ROM address.

The logic shown in the H and J area of Sheet 8 provides the Digital System clock to the CLOCK jack on the rear of the System Test Set, and to the Stop Clock gate discussed in the preceding paragraph. The PPF, ATN, and SNGL Switches are shown in areas L6 and M6. Note that in addition to supplying signals to the Processor, these three switches and the MEM Switch are ORed to light the OFF NORMAL indicator. The final circuit on Sheet 8 lights the BANK indicator when the Bank flip-flop in the 30-1 or 30-2 Processor is set.

Sheet 9 of the schematic shows the System Test Set clock and control logic. Assume first that the normal Processor internal clock is being used. In this case, the CLK ON Switch,shown in area G6, is depressed to remove the ground from IC05-10 (area C7). The CLKOFF0 signal is then forced high via gates at B7 and C2 when the START Switch at E7 is depressed. The Processor clock runs normally until a STP0 signal is received on an ROM address match, or until the CLK ON Switch is released.

If the External (System Test Set) clock is used, the EXT Switch,shown in area A8, is depressed. The normally-open contacts close to generate a CLKOFF0 signal which stops the Processor clock until EXT is released. The normally-closed contacts release the flip-flop shown in area C9. When CLK ON is depressed, followed by START, the flip-flop is set. The high output from the flip-flop enables the gate shown in area F9. The other input to the gate is the output from the external clock multivibrator circuit which is shown in the lower right area of Sheet 9. The output from the gate in area F9 generates the EXTCLK0 signal to the Processor via gates shown in areas G2 and H2. The clock continues to run at the frequency selected by the EXT CLK potentiometer shown in area M6, until the flip-flop shown in area C9 is reset. The flip-flop may be reset one of three ways: by SCLR1 from the Processor, by STP0 from Sheet 8, or by the CLK ON Switch (shown in area E6) being released. Note the SNGL CLK Switch shown in area H7. If the CLK ON Switch is released, the flip-flop shown in areas G8 and H8 is set each time the SNGL CLK Switch is depressed. The flip-flop output produces a clock pulse each time SNGL CLK is depressed.

The Initialize Switch, shown in area H6, produces a POW0 signal to the Processor each time it is depressed. The MEM VOLT potentiometer selects a memory threshold voltage which is applied to the Processor if the MEM Switch is depressed and the clip lead is installed.

6. MAINTENANCE

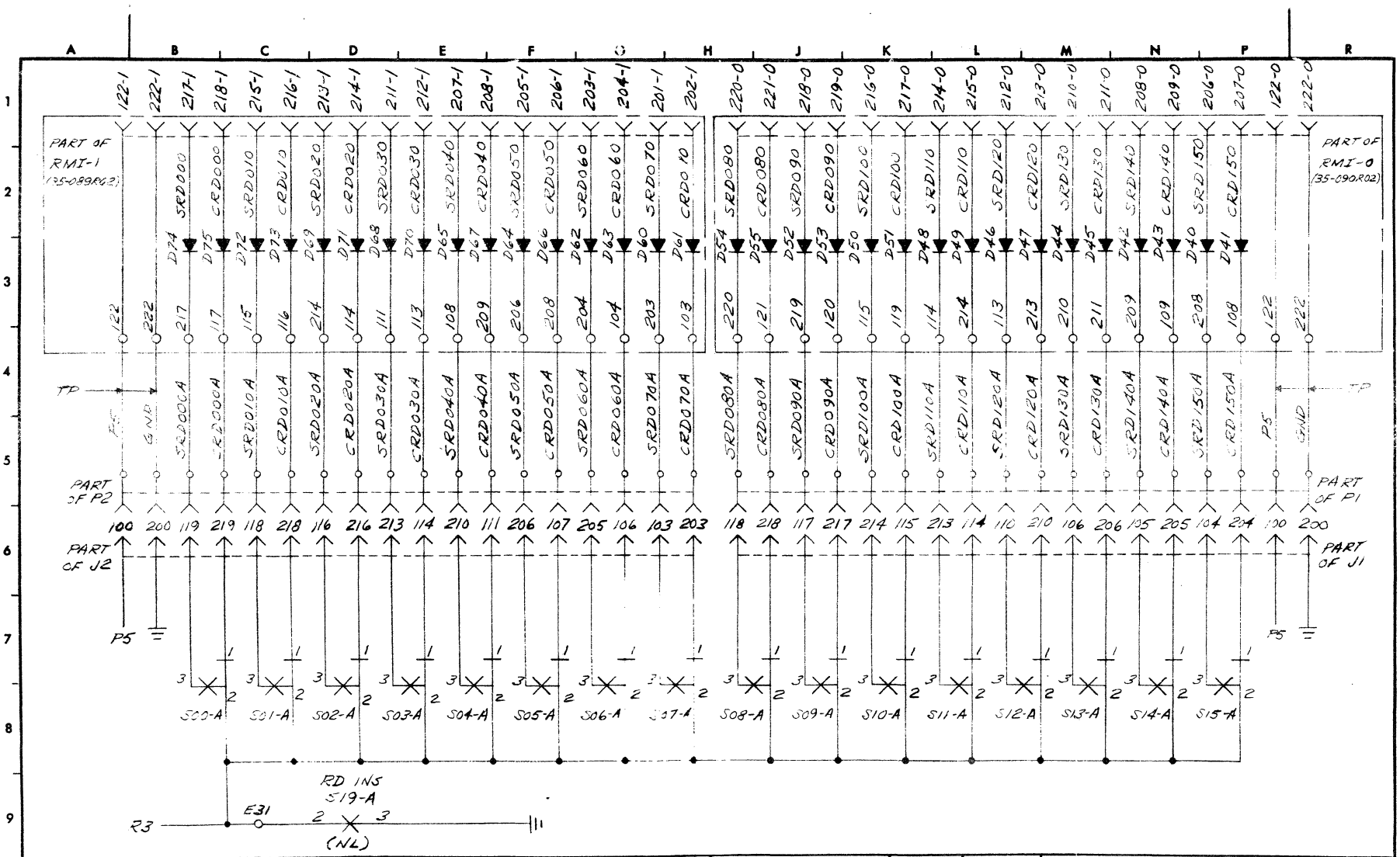
Maintenance on the System Test Set itself is normally restricted to the periodic replacement of indicator lamps. The lamps slip straight out from the front. It is normally possible to remove the lamp by hand. The replacement lamp, GE-PAC 30 Part Number 33-011 or Sylvania Part Number 12ESB, is simply pushed into place by hand.

If a trouble is encountered in the System Test Set, use the Functional Description provided in Section 5 and the schematics provided in Appendix 1 to locate the malfunction.

APPENDIX 1

DRAWINGS

This Appendix provides the schematic drawings necessary to maintain the System Test Set. The drawings are designated 28-001R01B08, Sheets 1 through 10.



NOTES 1. UNLESS OTHERWISE SPECIFIED, ALL APPARATUS IS PART OF TEST SET 39-11
2 UNLESS OTHERWISE SPECIFIED, RESISTORS ARE 1K, 1/4W AND ARE CONNECTED TO +5V.

NAME	TITLE	DATE

TITLE: SCHEMATIC SYSTEM TEST SET

TASK NO. 30-254 | DWG. NO. 28-001R01 REV | SHEET 1 OF 10

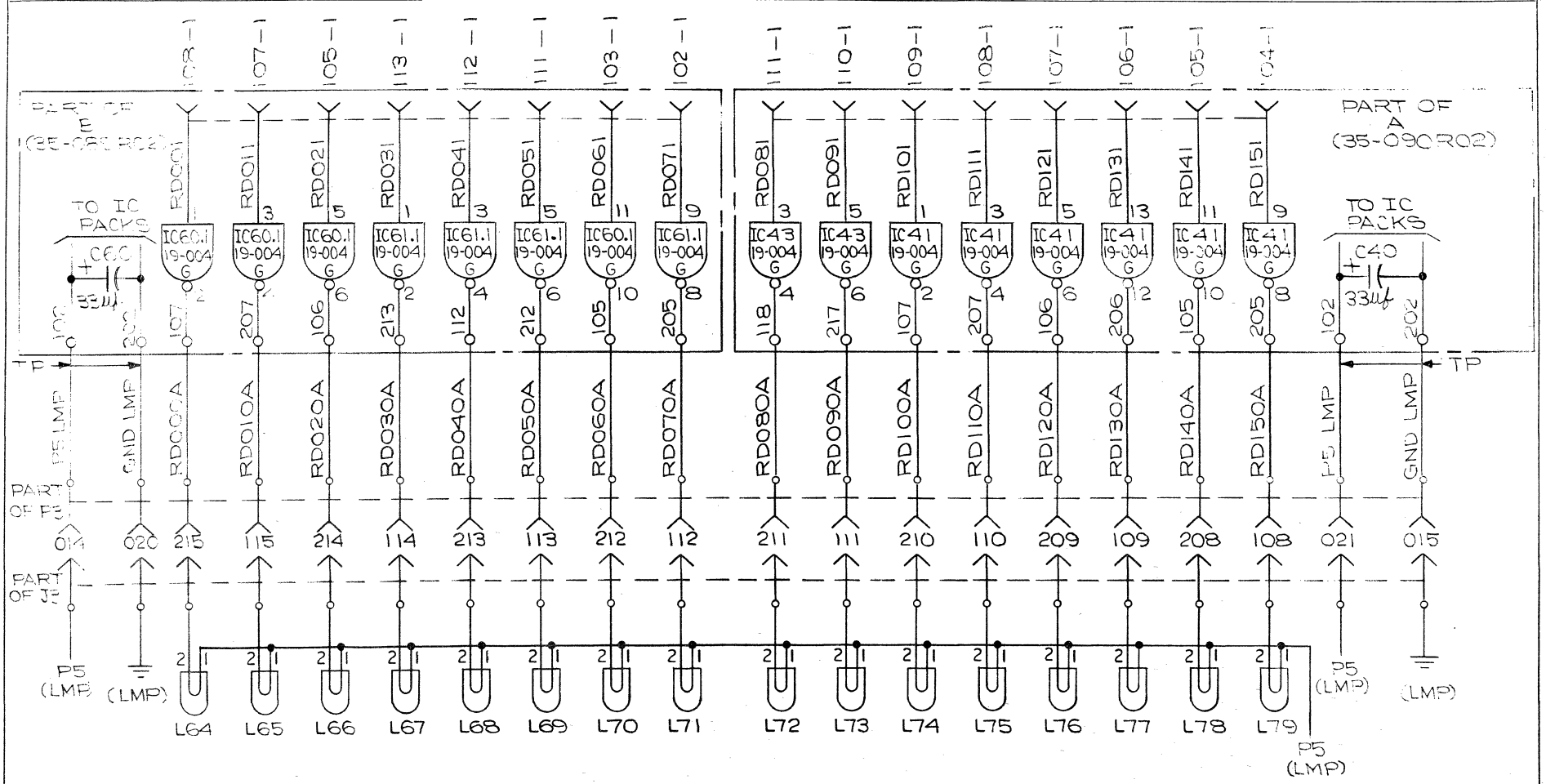
DIST.

UNLESS OTHERWISE SPECIFIED USE THE FOLLOWING:—
APPLIED PRACTICES | SURFACES | TOLERANCES ON MACHINED DIMENSIONS: FRACTIONS | DECIMALS | ANGLES
SURFACES: √
FRACTIONS: ± DECIMALS: ± ANGLES: ±

GENERAL ELECTRIC
PROCESS COMPUTER
PHOENIX

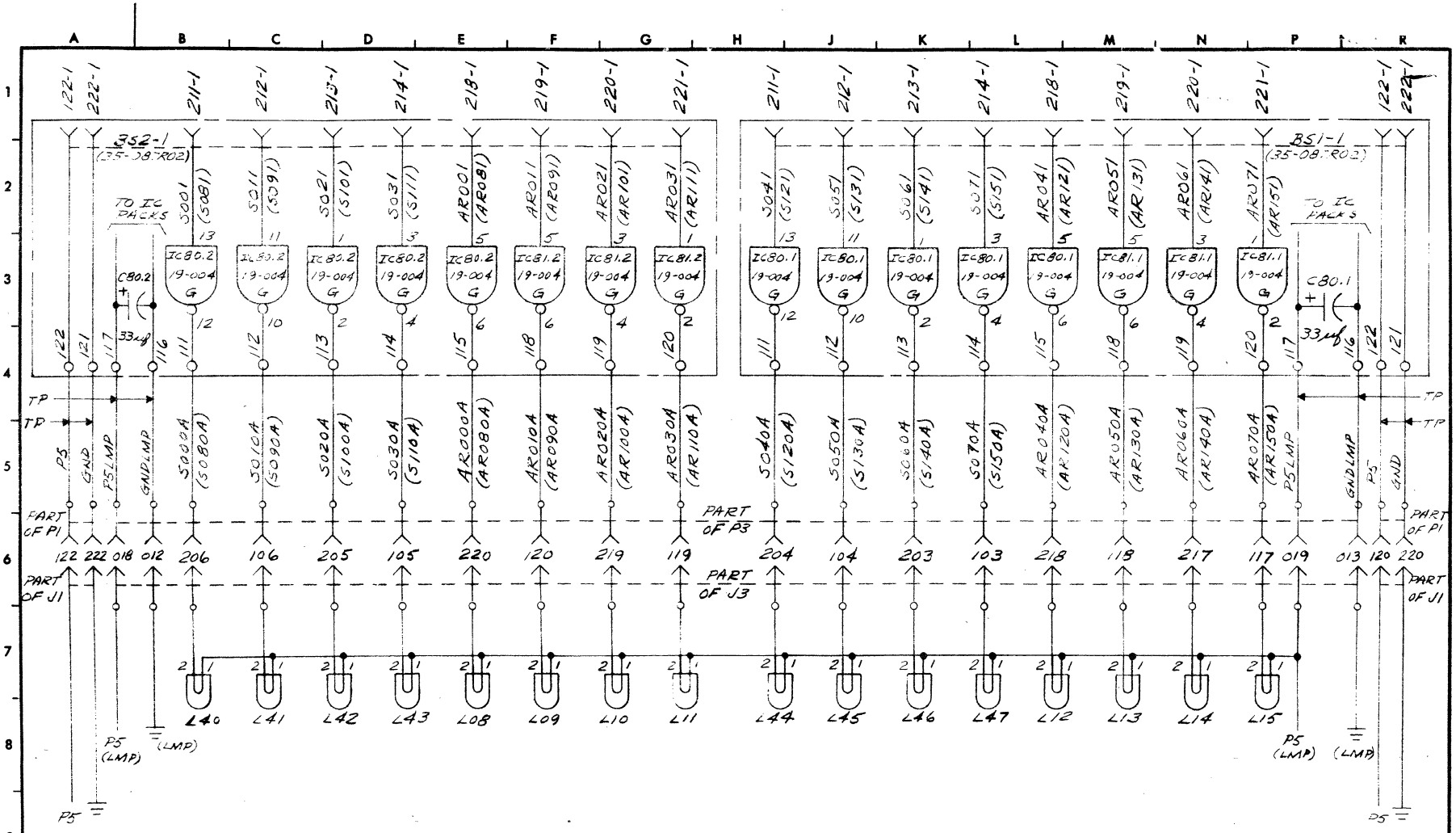
TITLE SCHEMATIC
SYSTEM TEST SET
FMF GE PAC 30 | FCF

DWG. NO.
CONT. ON SHEET SH. NO.



MADE BY | APPROVAL | DES. | CHKR. DR | REV.
ISSUED

DWG. NO.
CONT ON SHEET SH NO. 2



NOTES: SHEET NOTE: MNEMONIC DESIGNATIONS IN PARENTHESIS ARE FOR 30-02 USE ONLY.

NAME	TITLE	DATE

TITLE: SCHEMATIC SYSTEM TEST SET

TASK NO. 70-254 SHEET

DWG. NO. 28-001R01 303

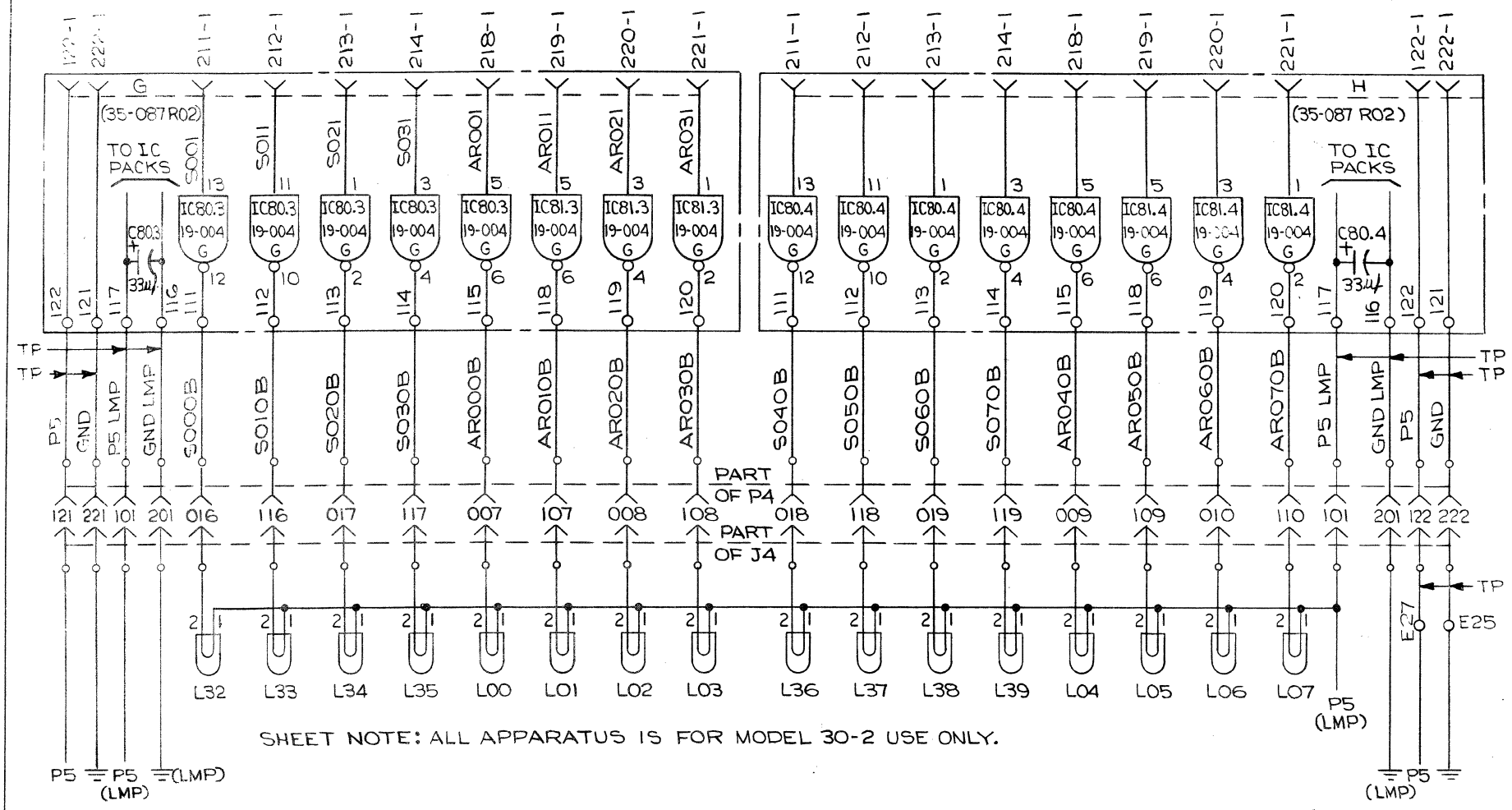
DIST.

UNLESS OTHERWISE SPECIFIED USE THE FOLLOWING:—
APPLIED PRACTICES | SURFACES | TOLERANCES ON MACHINED DIMENSIONS: FRACTIONS ± | DECIMALS ± | ANGLES ±
SURFACES: √

GENERAL ELECTRIC
PROCESS COMPUTER
PHOENIX

TITLE SCHEMATIC
SYSTEM TEST SET
FMF GE PAC 30 | 1 FCF

DWG. NO.
CONT. ON SHEET
SH. NO.



SHEET NOTE: ALL APPARATUS IS FOR MODEL 30-2 USE ONLY.

MADE BY
ISSUED
APPROVAL
DES.
CHKR. DR
REV.
DWG. NO.
CONT ON SHEET
SH NO. 4

DIST.

UNLESS OTHERWISE SPECIFIED USE THE FOLLOWING:—

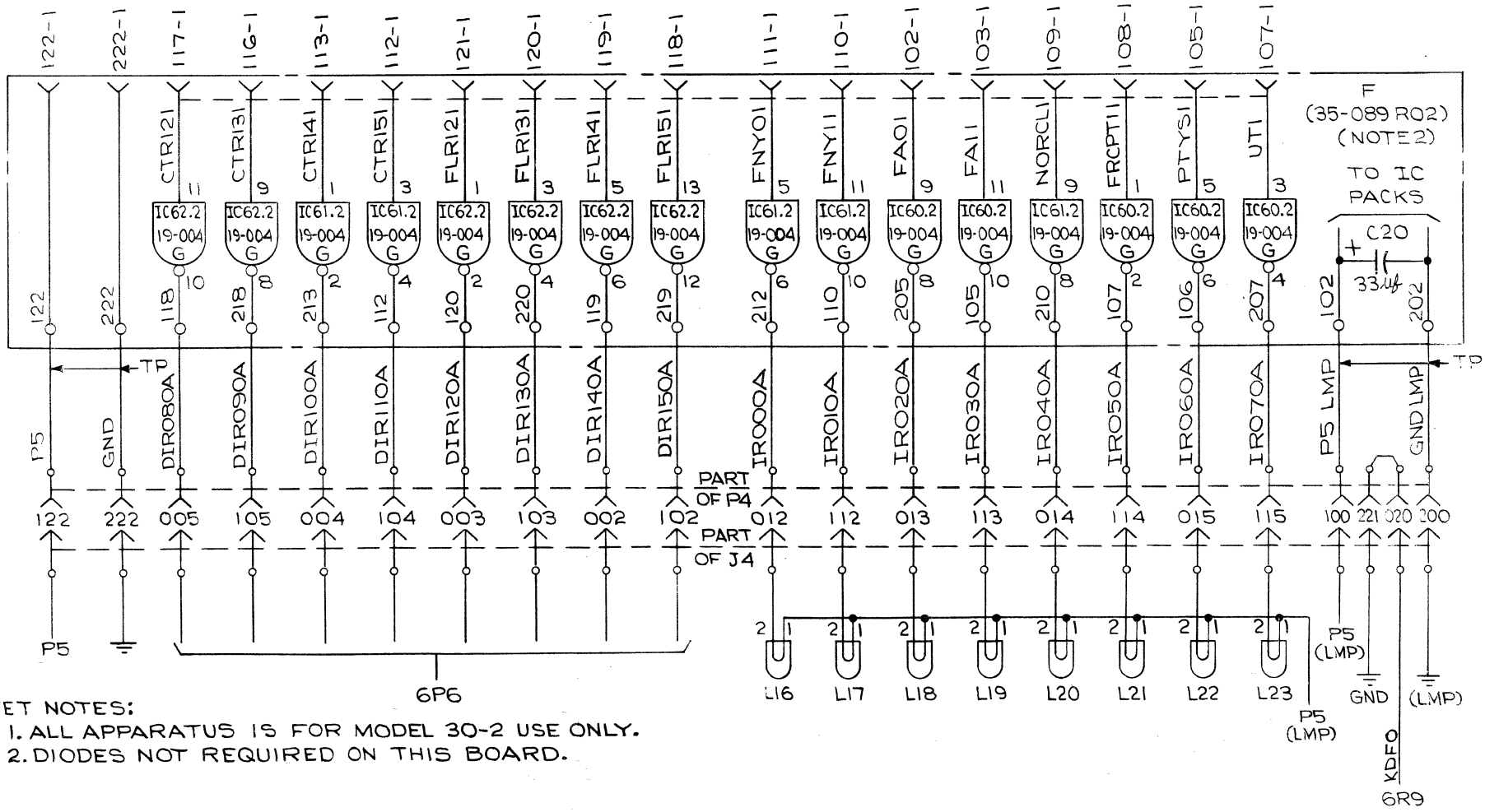
APPLIED PRACTICES	SURFACES	TOLERANCES ON MACHINED DIMENSIONS		
		FRACTIONS	DECIMALS	ANGLES
	√	±	±	±

GENERAL ELECTRIC
PROCESS COMPUTER
PHOENIX

TITLE SCHEMATIC
SYSTEM TEST SET
FMF GE-PAC 30 | FCF

DWG. NO.

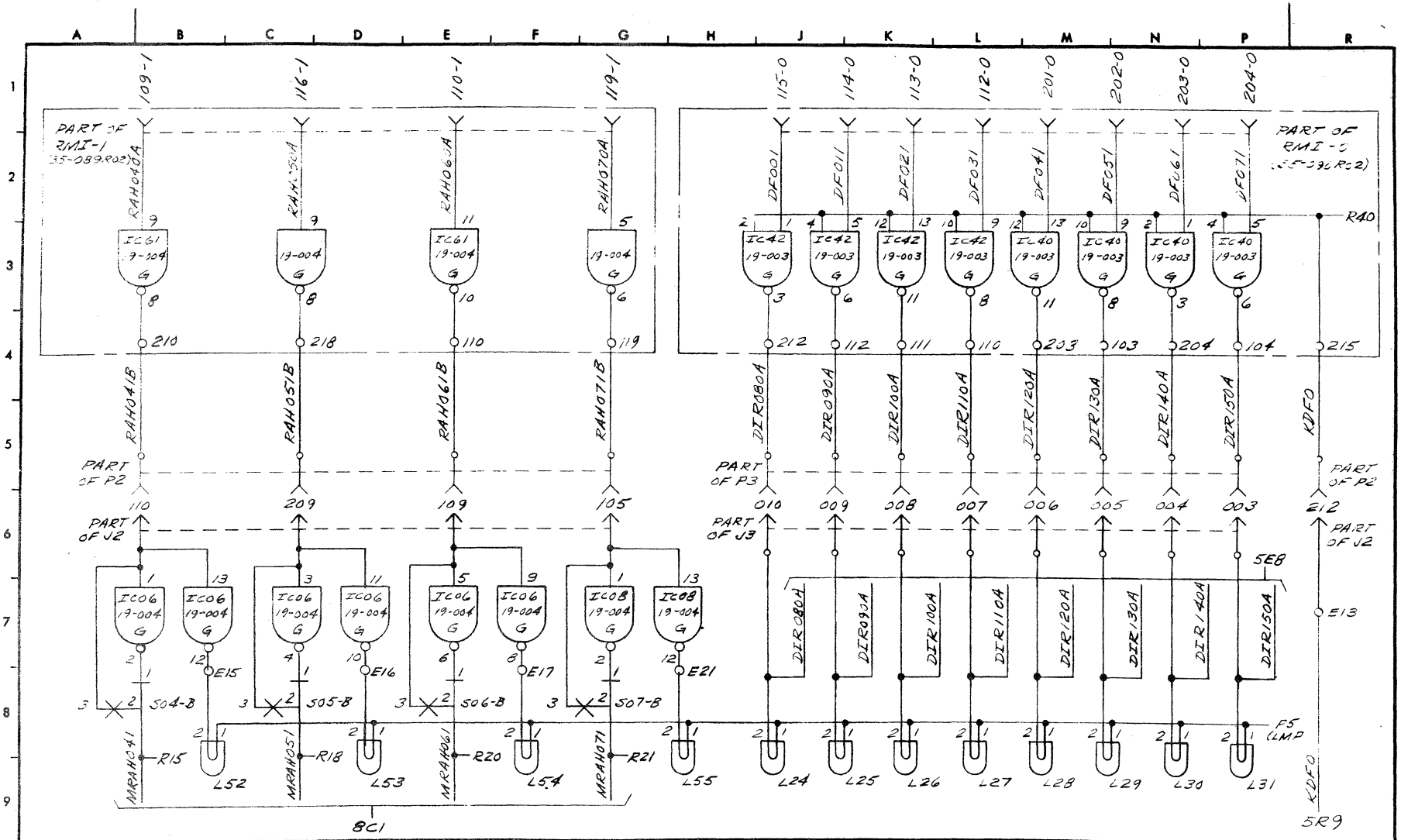
CONT. ON SHEET SH. NO.



SHEET NOTES:

1. ALL APPARATUS IS FOR MODEL 30-2 USE ONLY.
2. DIODES NOT REQUIRED ON THIS BOARD.

MADE BY	APPROVAL	DES.	REV.	DWG. NO.
ISSUED		CHKR. DR		CONT ON SHEET SH NO. 5

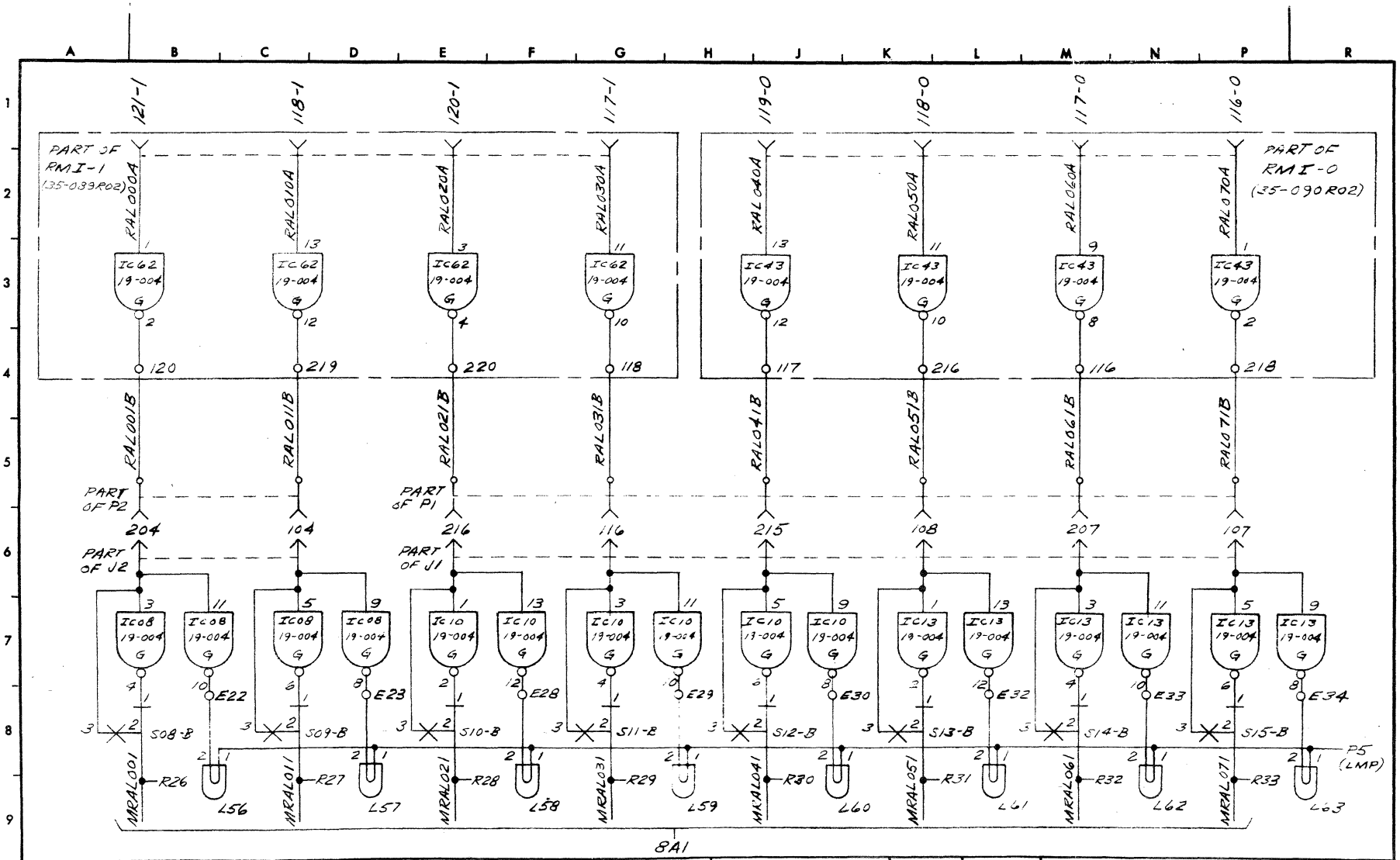


NOTES

NAME	TITLE	DATE

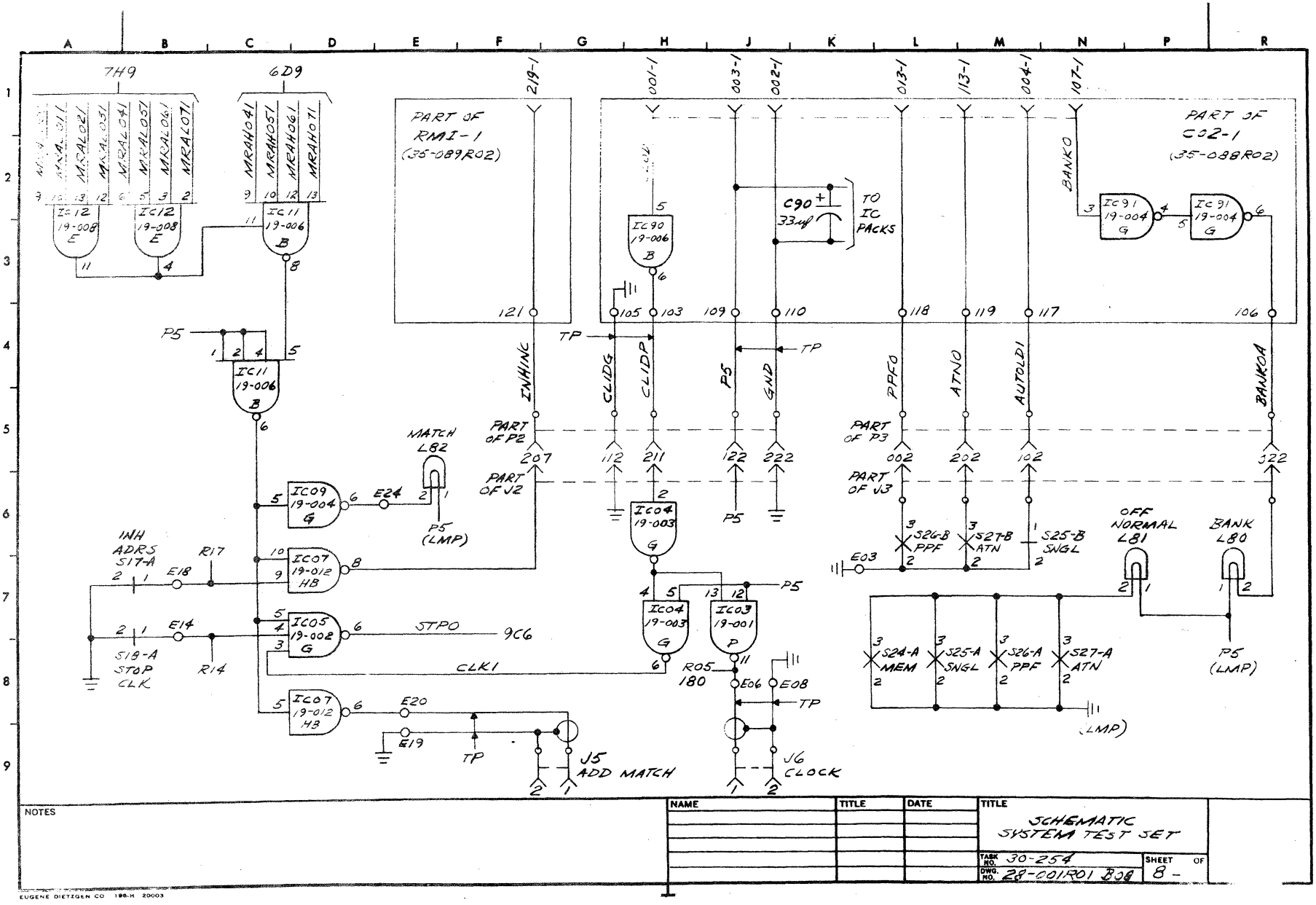
TITLE: SCHEMATIC SYSTEM TEST SET

TASK NO. 30-254 | DWG. NO. 29-001(P01)303 | SHEET 6 OF -



NOTES	NAME	TITLE	DATE	TITLE
				SCHEMATIC SYSTEM TEST SET
				TASK NO. 30 254 SHEET OF
				DWG. NO. 28-201R01 B08 1 -

EUGENE DIETZGEN CO 198-H 20003



NOTES

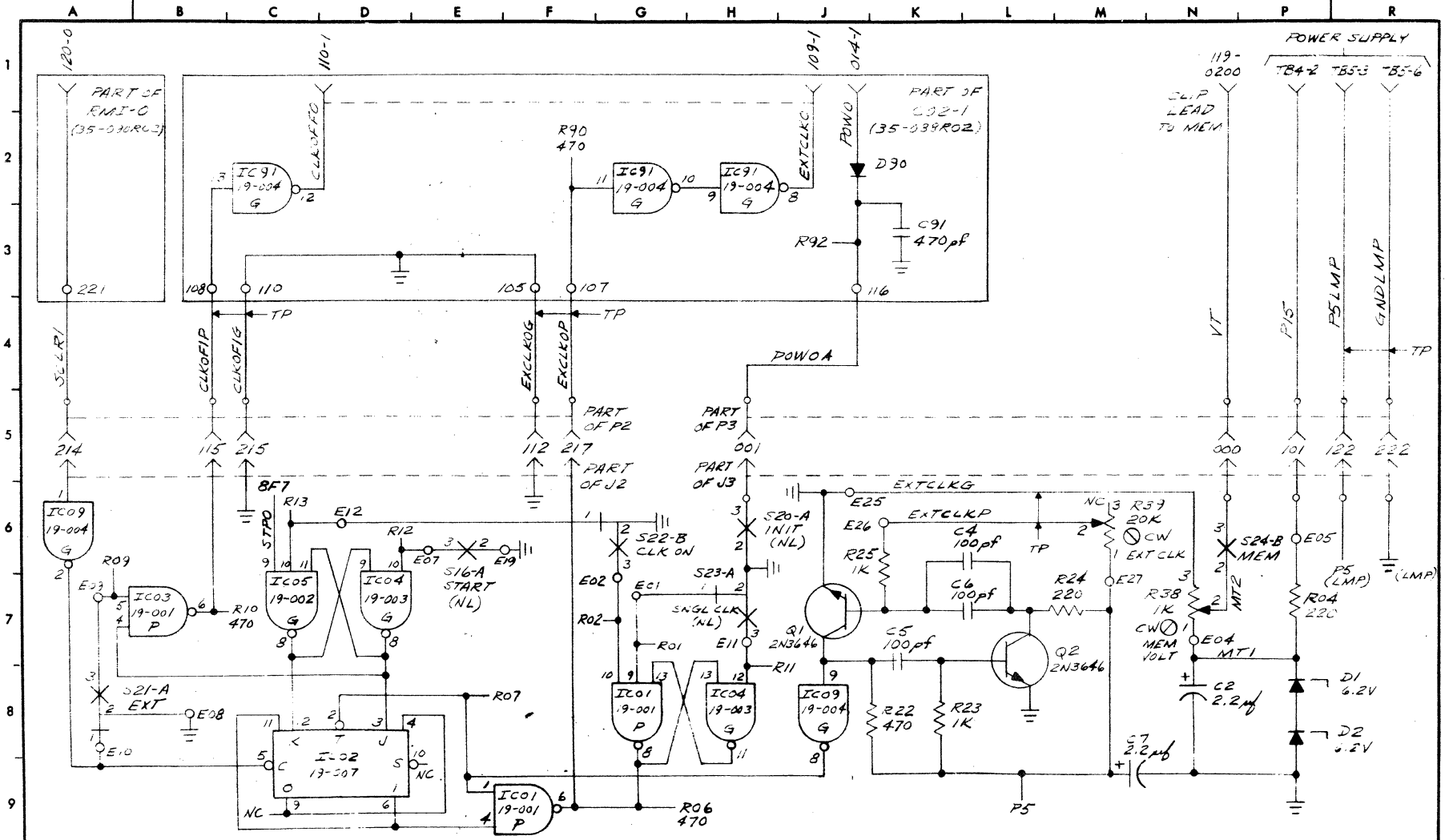
NAME	TITLE	DATE

TITLE: SCHEMATIC SYSTEM TEST SET

TASK NO. 30-254

DWG. NO. 28-001R01 B08

SHEET 8 OF


POWER SUPPLY
PART OF RMI-0 (35-010R03)
PART OF C02-1 (35-038R02)
IC91 19-004 G
IC09 19-004 G
IC03 19-001 P
IC05 19-002 G
IC04 19-003 G
IC02 19-007
IC01 19-001 P
IC01 19-003 G
IC09 19-004 G
Q1 2N3646
Q2 2N3646
D90
D1 6.2V
D2 5.2V
C91 470pf
C4 100pf
C5 100pf
C6 100pf
C2 2.2µf
C7 2.2µf
R90 470
R92
R09
R10 470
R07
R06 470
R11
R12
R13
R25 1K
R24 220
R22 470
R23 1K
R37 20K CW EXT CLK
R38 1K CW MEM VOLT
R04 220
S16-A START (NL)
S20-A INIT (NL)
S21-A EXT
S22-B CLK ON
S23-A SNGL CLK (NL)
S24-B MEM
MT1
MT2
BF7
STPO
SCLP LEAD TO MEM
VT
P15
P5LMP
GNDLMP
P5 (LMP)
(LMP)
TP
P5
SCLR1
CLKOFIP
CLKOFIG
EXCLKOG
EXCLKOP
POWO
POWOA
EXTCLKC
CLKOFFC
EXTCLKG
EXTCLKP
PART OF P2
PART OF P3
PART OF J2
PART OF J3
NOTES
NAME
TITLE
DATE
TITLE
SCHEMATIC SYSTEM TEST SET
TASK NO. 30-22-4
DWG. NO. 28-001R01 R08
SHEET
OF

CONNECTOR ASSIGNMENT FOR SYSTEM TEST SET

CABLE	CONNECTOR BOARD			COMPUTER LOCATION		
	TYPE	DESIG.		30-01 (WIREWRAP)	30-01 (COPPER)	30-02
17-035 ↓	35-089	RMI-1		11-1	10-1	10-1
	35-090	RMI-0		11-0	10-0	10-0
	35-087	BS1-1		13-1	11-1	18-1
	35-087	BS2-1		15-1	12-1	17-1
	35-088	CO2-1		19-1	14-1	13-1
17-038 ↓	35-087	BS3-1	—	—	—	16-1
	35-087	BS4-1	—	—	—	15-1
	35-089	CO1-1	—	—	—	19-1

NOTES

NAME	TITLE	DATE

TITLE: SCHEMATIC SYSTEM TEST SET

TASK NO. 3[illegible]57

DWG. NO. 28[illegible]

SHT 10

READER COMMENTS

The General Electric Company solicits your help in providing complete and accurate technical publications covering our Process Computer equipment. Please answer the questions listed here by checking the appropriate block. If your answer to any of these questions is "NO", please explain in "Comments" section below. Your comments and suggestions become the property of General Electric Company.

	YES	NO
• Is this publication adequate for your needs?	☐	☐
• Is the material		
Presented in clear text?	☐	☐
Conveniently organized?	☐	☐
Adequate detail?	☐	☐
Adequately illustrated?	☐	☐
Suitable for the technical level desired?	☐	☐

- What is your computer application? _____

_____

- What is your position? (Supervisor, Programmer, Technician, etc.) _____
- How is this publication used:

Familiarization of the subject? ☐ As reference material? ☐

For training purposes? ☐ For maintenance of equipment? ☐

Other (explain) _____

- Please give complete references (page number, line, etc.) with your comments. Please indicate if a reply is desired and include your proper mailing address.
- Your cooperation will be appreciated.

COMMENTS:

No postage necessary if mailed in the U. S. A.

Staple

YOUR ASSISTANCE, PLEASE

This document has been generated to help us serve you better. Your answers to the questions on the reverse side of this form, together with comments and recommendations, will be of great value to us in providing the best possible publications for your use. Your answers and comments will be carefully reviewed by the person who generated this publication, and may result in a revised publication. Your comments and recommendations become the property of General Electric Company.

Communications concerning Technical Publications should be directed to:

Manager, Technical Publications
GE Process Computer Department
2255 West Desert Cove Road
Phoenix, Arizona 85029

Fold Fold

FIRST CLASS
Permit No. 4091
Phoenix, Arizona

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

GENERAL ELECTRIC COMPANY
PROCESS COMPUTER DEPARTMENT
2255 West Desert Cove Road
Phoenix, Arizona 85029

Attention: Technical Publications

Cut Along Line

Fold Fold

Additional Comments: