HEWLETT PACKARD

Technical Information Package

13037 disc controller

Printed: AUG 1980

NOTICE

The information contained in this document is subject to change without notice.

This document has been prepared for use as a training supplement to facilitate the understanding of theory of operation to the component level. It is not the intent of this document, expressed or implied, to provide information that will enable repair to any level other than that which is specified in the Hewlett-Packard service manual.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Part No. 13037-90902

HEWLETT-PACKARD COMPANY
P.O. BOX 39, BOISE, IDAHO 83707, U.S.A.

Printed in U.S.A.

CONTENTS

CONTENTS (Continued)

INTRODUCTION

Information contained in this engineering supplement is intended to provide the customer with an in-depth knowledge as to the function and operation of the 13037 Disc Controller.

This document describes the 13037 Disc Controller as it is currently being shipped at the time of this printing. Information contained herein is subject to change without notice.

GENERAL DESCRIPTION

The 13037 Disc Controller is comprised of a discrete 16-bit microprocessor with externally addressable hardware functions physically located on the ECC/ROM and device controller boards. Although such functions are under the control of the microprocessor, some of these functions run asynchronously with respect to the microprocessor clock. For such hardware, the microprocessor will monitor various flags describing the condition of the circuitry, and at the appropriate times will update respective control signal latches to govern the desired processes (examples include reading information from the disc or decoding ECC patterns).

The device controller board is responsible for controller communications with the CPU (interface) and the disc drive. Additionally, the device controller hardware contains the circuitry needed to encode and precompensate data prior to its being written onto the disc (via the formatter), as well as to decode the data during subsequent reads.

The ECC/ROM board is comprised of two distinct sections: the ROM hardware (containing the microcoded instructions to be processed by the microprocessor) and the ECC hardware (error correction circuitry enabling the 13037 to correct up to a 32-bit burst of erroneous data read from the disc).

Internal communications of the 13037 microprocessor with the other controller hardware is provided by the microprocessor input/output (MIO) bus. This 16-bit bus is bidirectional, and concurrently connects the microprocessor board with the ECC/ROM and device controller PCA's. When the microprocesor wants to communicate with a particular hardware function of these other boards, the microprocessor sends the representative address over the five-bit external address bus. Function decoders on the ECC/ROM and device controller boards will respond by decoding this address and enabling the appropriate circuitry.

Figure 1 illustrates the relationship of each board to the bus lines previously described.

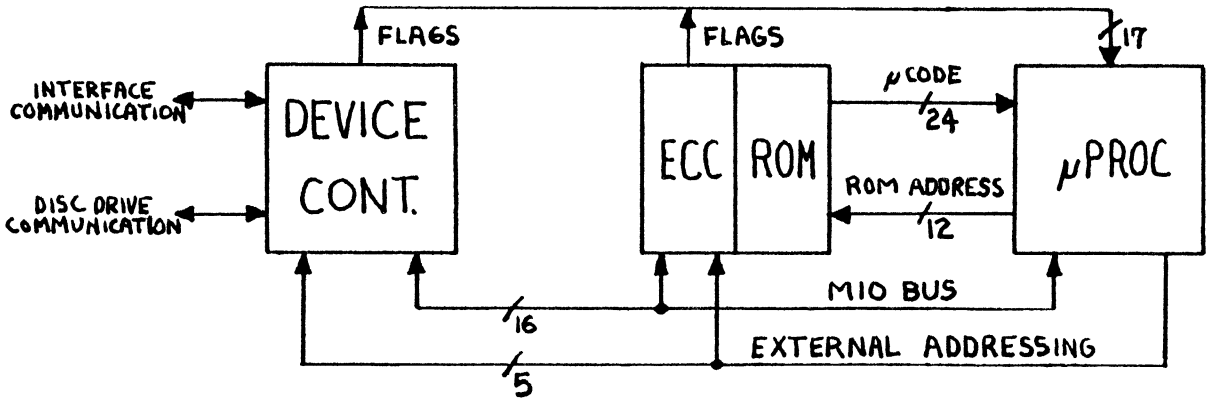


Figure 1

Figure 2 diagramatically relates each of the boards with specific hardware functions internal to these boards. Each of the boards is described below to the level represented by this block diagram. Component level discussions will follow.

The microprocessor board does not physcially contain the ROM hardware, but because it is essential for microprocessor operation, it is included in this section of the block diagram. The addressed microcoded instruction is made available by the ROM to the ROM output register (ROR) for subsequent latching at the beginning of the current instruction. At the same time, the ROM address register (RAR) is incremented to address the next instruction. In this way, the ROM can be setting up for this next instruction while the ROR holds the current instruction for subsequent command decoding.

The command decoding hardware interprets the current instruction as to type, then generates the appropriate control signals for proper execution of that command. One aspect of this decoding includes the determination of any appropriate branching. The branch select logic, in conjunction with the address gating, decides if the intended branch is to take place (by testing the condition code bit), and what address is to be used when branching. This address comes from one of four sources, as selected by the address gating hardware: 1) the three-level subroutine stack (during "return" operations from subroutines); 2) test inputs used in factory testing only; 3) a computed address from the arithmetic portion of the microprocessor (indirect branching; or 4) an immediate address taken directly from the lower twelve bits of the ROR (corresponding to a direct branch). Note that the subroutine stack functions during subroutine calls by buffering the corresponding return address for the "return" operations discussed above.

Other command decoding functions include the selection of a desired condition (either internal to the microprocessor board or one of the external flags previously described) to be latched into the condition code latch during a test instruction. This condition code bit (CCB) is tested by the branch select logic during branch instructions to decide if the intended branch is to be allowed (including subroutine calls and returns).

The arithmetic portion of the microprocessor board can process data from two sources: an eight-bit immediate operand (provided by the lower eight bits of the ROR), or register data (stored in the internal A and B registers). The arithmetic logic can function in byte (upper or lower) or full word modes. Byte modes must by used during immediate operand instructions, and though the immediate operand replaces any A register (A REG) source, a B register (B REG) must be used concurrently for any operation other than PASS or COMPLEMENT. For normal operations, A and/or B register sources are used and either byte or full word modes can be specified.

The registers are organized as two sets of four 16-bit registers (i.e., four A registers and four B registers). They receive data from the MIO bus for storage and source data to the arithmetic logic unit (ALU) for processing. The ALU will accept data as previously described and operate on these data as prescribed by the current instruction. The resulting computation is stored in a temporary register (T register) for later use, and subsequently passed to the shifter for any swap, shift, or rotate operation prior to its driving the MIO bus. If the resulting data is to be stored in one of the internal registers, it is now available on the MIO bus for writing into that register.

The process status register (PSR) monitors salient characteristics of the current arithmetic operation and latches that information for later review. This data can be individually sampled by the condition select logic in setting the condition code latch. In this way, branching can occur based on a former arithmetic result.

The external address bus is driven from the appropriate bits of the ROR. This data is used as previously described to allow the microprocessor to enable external communication paths over the MIO bus.

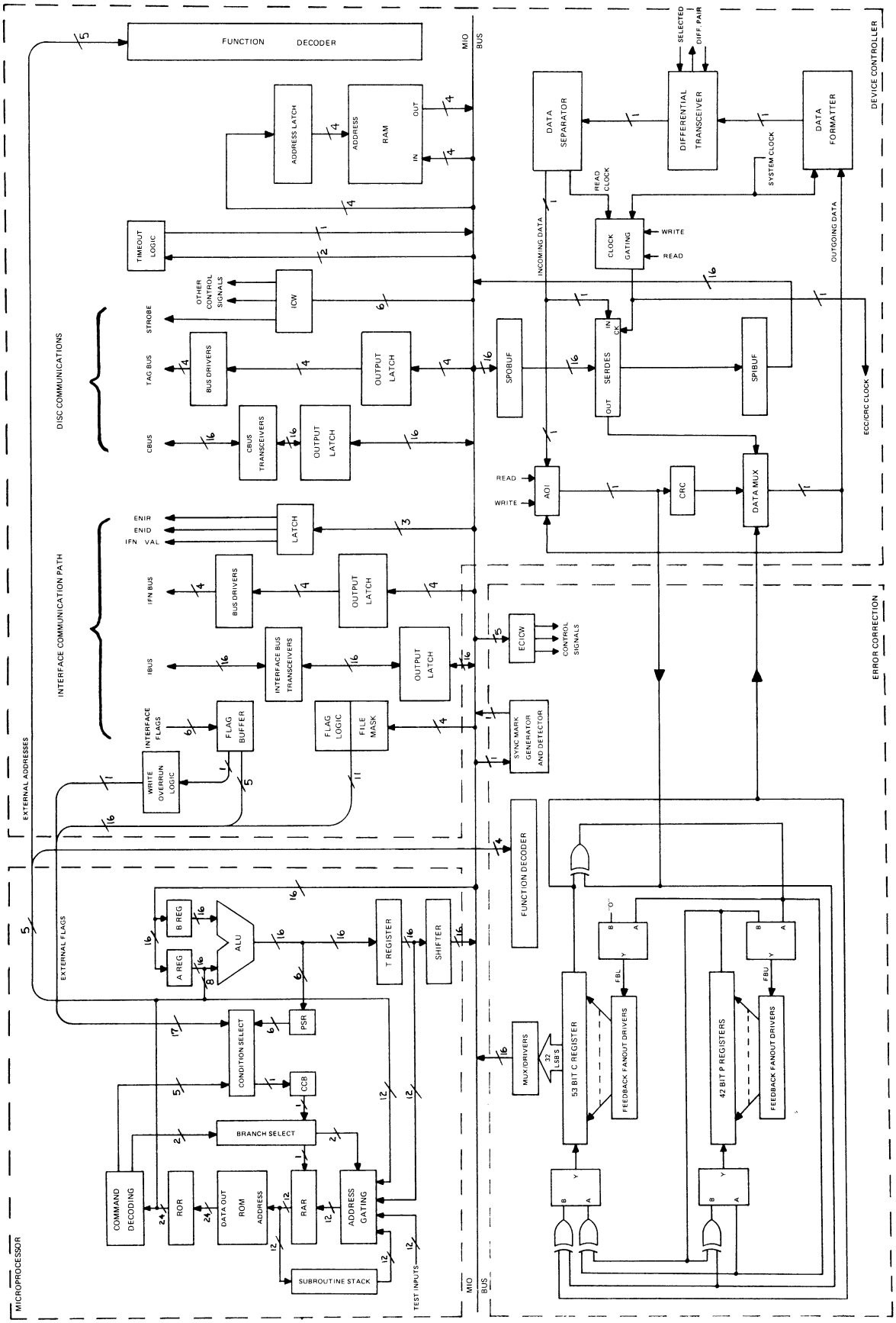

MICROPROCESSOR
FUNCTION DECODER
MIO BUS
ADDRESS LATCH
ADDRESS RAM
TIMEOUT LOGIC
OTHER CONTROL SIGNALS
ICW
DISC COMMUNICATIONS
STROBE
TAG BUS
CBUS
BUS DRIVERS
OUTPUT LATCH
CBUS TRANSCEIVERS
SPOBUF
SERDES
SPIBUF
DEVICE CONTROLLER
DATA SEPARATOR
DIFFERENTIAL TRANSCEIVER
DATA FORMATTER
SELECTED DIFF PAIR
READ CLOCK
SYSTEM CLOCK
CLOCK GATING
WRITE
READ
INCOMING DATA
OUTGOING DATA
ECC/CRC CLOCK
AOI
CRC
DATA MUX
INTERFACE COMMUNICATION PATH
ENIR
ENID
IFN VAL
LATCH
IFN BUS
IBUS
INTERFACE BUS TRANSCEIVERS
ECICW
CONTROL SIGNALS
INTERFACE FLAGS
FLAG BUFFER
FLAG LOGIC
FILE MASK
SYNC MARK GENERATOR AND DETECTOR
WRITE OVERRUN LOGIC
EXTERNAL ADDRESSES
EXTERNAL FLAGS
ERROR CORRECTION
FUNCTION DECODER
B REG
A REG
ALU
T REGISTER
SHIFTER
PSR
CONDITION SELECT
CCB
BRANCH SELECT
COMMAND DECODING
ROR
DATA OUT ROM ADDRESS
RAR
ADDRESS GATING
SUBROUTINE STACK
TEST INPUTS
MUX DRIVERS
32 LSB'S
53 BIT C REGISTER
42 BIT P REGISTERS
FEEDBACK FANOUT DRIVERS
FBL
FBU

Figure 2

The device controller board performs two major functions: communications with the CPU interface and disc drive, and the conditioning of data during disc reading and writing. These major functions are illustrated in the upper and lower portions of figure 2, respectively.

The function decoder monitors the external address bus to decide which function is being enabled by the microprocessor; in this way conflicts over use of the MIO bus are eliminated. Additionally, the internal control word (ICW) latch is updated by the microprocessor board to contain control signals relating to the particular function being performed (i.e., read, write, disc drive function bus validation, etc.). These control signals are used primarily by the data encoding and decoding hardware to establish the current modes of operation.

The RAM and its address latch are used to store 16 four-bit words of data for later use by the controller. This information includes disc drive to CPU mapping, a retry counter (used in recovering valid address information from a defective track), polling information, and other such quantities.

The timeout logic is a 1.8-second hardware timer used whenever the device controller has to wait for the CPU or disc drive to complete an intended function. Examples of such functions include waiting for the CPU interface to send or receive data, or waiting for the disc drive to complete an incremental (or decremental) seek during multiple track transfers. If the intended function does not complete within the 1.8-second period, the controller will enter a power on/timeout processor routine to disconnect from the malfunctioning device.

The file mask logic of the device controller is a series of one bit latches that retain information concerning sparing operations (sparing enabled) and multiple track transfers (direction of seek, cylinder or surface mode, etc.). This information is updated only by a power-on (or timeout) or by a CPU "Set File Mask" command to the controller. These one-bit data can be individually sampled as external flags by the microprocessor.

Additional external flags from the CPU interface are buffered by the flag buffer hardware. One flag, the overrun signal, is further qualified by the write overrun logic to detect any overruns internal to the device controller board during write operations.

The communications paths between the controller and the disc drive and CPU interface are organized in a basically identical manner. Each communication path has a four-bit function bus to describe the operation expected of the CPU interface or disc drive. These commands are transmitted over the interface function bus (IFN BUS) or the disc drive TAG BUS. These busses are unidirectional and are driven by the device controller board. Each bus has an output latch that (when enabled by the function decoder) stores the current MIO bus contents for subsequent transmission through the bus drivers. In this way the latched function remains constant independent of any changes on the MIO bus (until, of course, being updated as selected by the function decoder).

These functions must be validated by a control signal before the interface or disc drive will respond to them (in this way spurious noise will fail to initiate false operations). The interface function valid (IFN VAL) signal, one of three latched values, enables the interface to respond to the associated IFN BUS command. Additionally, the two other latched values are sent to the interface: the ENIR (enable interface receiver) signal enables the interface to accept information over the 16-bit interface bus (IBUS), a bidirectional data bus used to transfer additional data pertinent to the particular interface function being executed (i.e., a READ function would send read data to the CPU over the IBUS); the ENID (enable interface driver) signal allows the interface to send information over the IBUS (i.e., data words to be written onto the disc).

Validation of disc drive TAG BUS functions occurs with a true STROBE signal. This signal is updated with the ICW previously discussed. The disc drive control bus (CBUS), a 16-bit, bidirectional counterpart to the interface IBUS, has no equivalent ENIR or ENID signals. Instead, the most significant bit of the TAG BUS is used to establish direction of data transfer over the CBUS.

As with the function busses, the IBUS and CBUS require output latches to retain MIO bus information being outputted by the controller. When inputting information from these busses, the output latches are bypassed (information is passed directly over the IBUS and written into a microprocessor register).

Actual serial data transfer to or from the disc drive is handled asynchronously to the microprocessor by the lower half of the figure 2. At the heart of the transfer operation is the serializer-deserializer (SERDES), a 16-bit shift register with parallel loading capabilities.

In the write mode, the SERDES parallel output buffer (SPOBUF) gets updated with MIO bus data to be written as the next word onto the disc. Every sixteenth clock pulse, SERDES parallel loads the latched SPOBUF quantity and subsequently serially shifts it to the drive. The microprocessor then updates SPOBUF (via the function decoder) with the next word to be written.

In the read mode, SERDES serially inputs data as it is read from the disc. With every sixteenth bit the SERDES parallel input buffer (SPIBUF) latches the current word read from the disc. Before the next sixteenth bit, the function decoder will enable SPIBUF to put that word on the MIO bus for subsequent transmission to the interface.

It should be noted that in both the read and write modes previously described, a bit counter increments with every clock of SERDES. At every word boundary (every sixteenth bit) an EOW flag is set telling the microprocessor that a word has been read (SPIBUF has been updated), or a word is being written (SERDES has parallel loaded the SPOBUF quantity). In this way the microprocessor can synchronize itself with the somewhat asynchronous read/ write logic.

(The fact that microprocessor instructions are executed at a 5 MHz rate while data is transferred to/from the disc at a nominal 468.75k word/second rate reflects the asynchronous nature of the two operations.)

Prior to being written onto the disc, the serial data from SERDES is encoded to allow denser information packing onto the media. Additionally, a separate clock from the disc is not required as the MFM encoding technique is used (MFM combines the clock and data into one waveform and can be subsequently decoded into the original clock and data streams). The data formatter hardware takes SERDES serial data, encodes it, and precompensates it (to prevent disc media pulse crowding effects) before sending it to the differential transceiver. It should be noted that encoding of data is done with the system clock (divided down to the desired bit transfer rate).

Each of the eight data ports of the device controller is driven by a differential transceiver. These transceivers are connected in parallel, with possible conflicts eliminated by the fact that only the selected drive will enable its respective differential transceiver (only one drive is ever selected at any given time).

Besides sourcing formatted data to the drive during write operations, the differential transceiver receives read data from the drive and channels it to the data separator hardware. This hardware phase locks to the incoming waveform and derives from it the decoded data. Additionally, a clock is generated coincident with this decoded data that will track with any fluctuation in the return rate of that data. This read clock is gated through the clock gating hardware to clock SERDES in time with the incoming read data (during the write mode, formatted data is generated synchronous to the system clock, and thus the gating logic passes the system clock to SERDES).

As a supplemental means of determining the integrity of a read transfer, cyclic redundancy circuitry (CRC) is incorporated. This hardware uses data being written onto the disc (prior to its being formatted) to generate a characteristic CRC word. This word is subsequently ap-

pended to the written data field. During read operations, separated data is concurrently fed to SERDES and the CRC logic. A new CRC word is subsequently generated that will, if no errors were enountered, match the previously written CRC field. Absence of this match will cause a data error signal generation.

Because of the two possible sources of CRC input (separated data during read operations and outgoing data during write operations), and-or-invert (AOI) hardware is provided to establish the proper data path to the CRC.

In addition to CRC error detection, an error detection and correction technique is additionally provided to help correct misread or miswritten data. This error correction circuitry (ECC) monitors the same data as the CRC and hence is fed from the output of the AOI logic. As with CRC, during write operations a characteristic ECC pattern is generated and appended to the CRC field of the sector. This outgoing data requires a third write data path to the disc (i.e., data field path from the interface, CRC field path from the CRC hardware, and ECC field path from the ECC/ROM board); the DATA MUX must be included to appropriately select these data paths at the respective time that the corresponding fields are to be written onto the disc. It is the data mux output that is then sent to the data formatter for appropriate encoding.

The error correction circuitry (ECC) is comprised of one 53-bit shift register and one 42-bit shift register. In the encoding mode (i.e., during write operations when data sent to the disc is used to generate the characteristic ECC pattern), these two registers are joined in series and given appropriate feedback to create a 95-bit encoding register. In the decoding mode (i.e., during read operations when separated data is monitored in an attempt to detect errors), each register is simultaneously fed the separated read data. After the sector is clocked through, both registers will be identically zero if no error occurs. If any are non-zero, an error was detected by the respective non-zero register. Only when both registers are non-zero (i.e., both have detected an error) can error correction be attempted (both registers will be compared to see if they agree on the detected error).

Because of the varying configuration of the ECC hardware in the decoding and encoding mode, mux'es are included to attain the proper configuration for the respective operation. These mux'es affect not only the relationship of the registers, but the type of feedback as well. The control signals necessary to esablish these conditions are generated by the error correction internal control word (ECICW). This latch is updated by the MIO bus when enabled by the function decoder hardware (this hardware monitors the microprocessor generated external address lines, and performs the same type of function as the function decoder on the device controller board).

Once an error has been detected and the correction pattern generated, the correction data must be sent to the CPU. At this time the function decoder will enable the mux/drivers to select the appropriate word and send it across the MIO bus for subsequent transmission to the interface.

The sync mark generator and detector hardware is included to establish during write operations that a valid ECC field is being written (i.e., a 100376 sync word will be written as opposed to a 100377 sync word), and determine during read operations that a valid ECC field is present (found by monitoring the least significant bit of the sync word). This hardware was included because the initial controllers were to have error correction as an option, but it was later decided to be included as standard equipment.

This concludes the general description of the 13037 Disc Controller. Detailed circuit theory is found elsewhere in this document.

CONTENTS

MICROPROCESSOR PCA

1.0 MICROPROCESSOR CHARACTERISTICS

This document describes the microprocessor PCA of the 13037 Disc Controller. The function of the disc controller is to translate high-level commands from a 21XX or 3000 series computer into commands recognized by controller-compatible disc drives. Examples of such high-level commands are READ, WRITE, SEEK, and REQUEST STATUS.

The disc controller is a microprogrammed processing unit. The microprogram, or firmware, is stored in a read only memory. The microprocessor fetches instructions from the ROM, decodes them, manipulates and stores data, and creates the control signals which sequence the disc controller hardware.

Microprogramming has several advantages over the alternative method of implementing a disc controller by using a hardware sequenced state machine. Microprogramming allows a straightforward, orderly hardware design which is largely independent of specific controller characteristics. Complex algorithms in a microprogrammed machine do not require complex hardware, only appropriate firmware. Controller modification becomes a matter of changing firmware, which is much easier than modifying hardware.

The disc controller microprocessor resides on a single 3000-size printed circuit board. It communicates with the other disc controller modules via a central data (MIO) bus and clock, strobe, and addressing lines. Its current load is 5 amps.

2.0 MICROPROCESSOR DESCRIPTION

2.1 BLOCK DIAGRAM

Figure 1 is a block diagram of the microprocessor. The left half of figure 1 is instruction sequencing logic; the right half shows the registers, and arithmetic and logic modules.

Instructions are contained in the ROM. The command decoding block uses the current instruction to create control signals which operate the arithmetic/logic section. The ROM address register contains the address in ROM of the next instruction. This address is one greater than the previous address if a branch is not being executed. If a branch is being executed, the address gating network selects the proper branch address from one of four possible sources: the subroutine stack, an external address, the T register, or the immediate field of a branch instruction.

The condition select block tests a flag and sets the condition control bit (CCB). The branch select logic looks at the CCB to decide if the branch is to succeed or fail. It also determines what type of branch is to occur.

The right part of figure 1 shows the four A and four B registers which may act as inputs for the ALU. Note that an immediate operand may replace an A register in some instructions. The ALU output is temporarily held in the T register. Following this, the shifter rotates, shifts, or swaps bytes, putting the result on the MIO bus. This result is then directed either to an A or B register, or to an external destination.

The PSR block represents the processor status register, which latches six ALU conditions and makes them available to the conditional branch test circuitry.

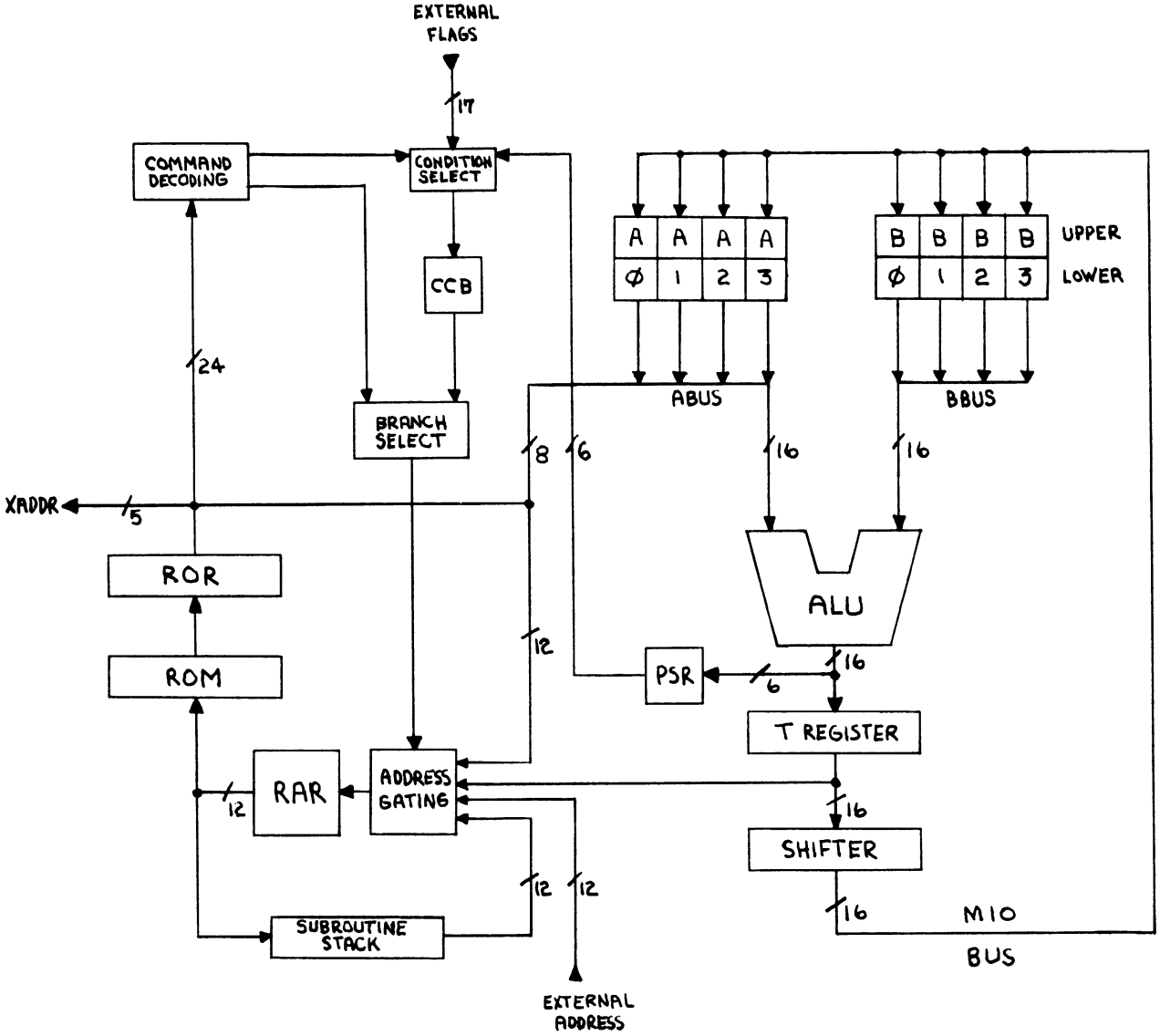


Figure 1. Microprocessor Block Diagram

2.2 MODULE DESCRIPTIONS

Microprocessor Module

This module contains all of the major decision-making logic contained in the controller. It can execute arithmetic/logical, input/output, and branch instructions. Each instruction will be executed in 200 ns. It controls and communicates with all other controller modules on the central MIO bus.

2.2.1. Processor Hardware Description

ALU

The ALU performs all of the functions listed below. ROM bits 16 - 20 and 14 in arithmetic and I/O instructions specify which operation is to be performed. All operations may be performed on a 16-bit (full word) or 8-bit (byte) basis. ALU status is available after each arithmetic instruction and is held in the Processor Status Register (PSR). Listed below are the operations incorporated by the 13037B firmware.

PLUS	INCREMENT	COMPLEMENT
MINUS	EXCLUSIVE OR	LOGICAL AND
PASS	INCLUSIVE OR	DECREMENT

BUSES

The ALU is driven by the "A" and "B" buses, each of which connects to four 16-bit registers, to be described later. The "A" bus also connects to the Immediate Operand field of the ROM instruction. (Note, therefore, that all two parameter immediate operations are with B registers.) All information on these buses is ground true.

The MIO bus is a 3-state bus which connects the output of the ALU-shift unit to the registers, as well as other modules. Information may be strobed out on this bus from a register or Immediate Operand, or it may be strobed into a register from the outside world or from the shift unit. Information on this bus is ground true.

REGISTERS

Eight 16-bit registers are implemented. Four of them drive the "A" bus (A0 – A3) and the other four (B0 – B3) drive the "B" bus. All are fed from the MIO bus.

Instructions may affect only the upper or the lower byte of these registers, or, in some cases, the entire 16 bits.

CONDITIONS

Seventeen external flags and six processor status indicators are available for use in conditional branches. They are all positive true. The external flags connect to the outside world via the MIO connector. Processor status is recorded during each arithmetic instruction in the Processor Status Register (PSR) and is available until the next arithmetic instruction. These conditions are: EQUAL, signifying that the contents of the A and B registers selected are equal; UOVER, signifying overflow (carry) out of the ALU upper byte; LOVER, indicating carry out of the lower byte of the ALU; TNZRO, meaning that the contents of the T-register are not zero; TMSB, the sign bit of the T-register; and TLSB, the LSB of the T-register. In addition, a condition which is always FALSE is provided for unconditional branches.

An SCC (set condition code) instruction should be executed before a branch to clock the selected condition into the condition code bit (CCB). A following branch will depend on the value set in the CCB by the last SCC instruction.

S-REGISTER

The S-register is a three-level LIFO (last-in-first-out), or push-down, stack. This allows for three-level subroutine jumps. When a CALL is executed, the current address plus 1 is pushed onto the stack; when a RTN is executed, the value on the top of the stack is put into the P-register and the stack is popped.

Care should be taken in programming not to exceed three levels of subroutine nesting, as only the last three return addresses are remembered; previous information is lost. Also, if more than three returns are made in succession, then the address which was at the bottom of the stack will be repeatedly jumped to.

T-REGISTER

The T-register holds the result from the ALU, again in ground true form. An indirect branch will load the contents of the T-register into the P-register, so care must be taken that the data in the T-register at that point has been complemented and is the correct sense.

SHIFTER

The shifter can: swap bytes; pass; and rotate, shift left, or shift right by one bit. Rotates and swaps are meaningless for byte instructions.

CLOCKS

The microprocessor runs on a 30 MHz clock in six phases, labeled T0 through T5. Timing is shown in figure 3 (next section). Each instruction is executed in 200 ns.

P-REGISTER AND ADDRESS GATING

The P-register is a 12-bit counter register which addresses the ROM. Normally the next instruction is in the current address plus one, so P is incremented during each instruction. In a branch, however, one of four different address sources (depending on the ROM instruction and the CCB) is loaded into the P-register.

This address may come from the BRANCH ADDRESS field of the ROM, the S-register, the T-register, or an external source. Which branch is taken is discussed in the description of the instructions.

ROM

A bipolar ROM with maximum access time of 130 ns is used to store the control program. The instruction word width is 24 bits. Up to 4K of control storage is addressable.

2.2.2 INSTRUCTION DESCRIPTION (Refer to figure 2)

OTI — Output Immediate Operand

The immediate operand contained in ROM bits 0 – 7 is passed through the ALU and shifter to the MIO bus. Information on the MIO bus is ground true. The external destination contained in ROM bits 8 – 12 is put on the external address bus and is positive true. External select is true for the entire instruction time. At T4 the UB or LB strobe (depending on ROM bit 13) is transmitted. The strobe occurs for 1.5 CPU clock periods (allowing for attenuation prior to reaching the device controller board) and is positive true. No shift is performed.

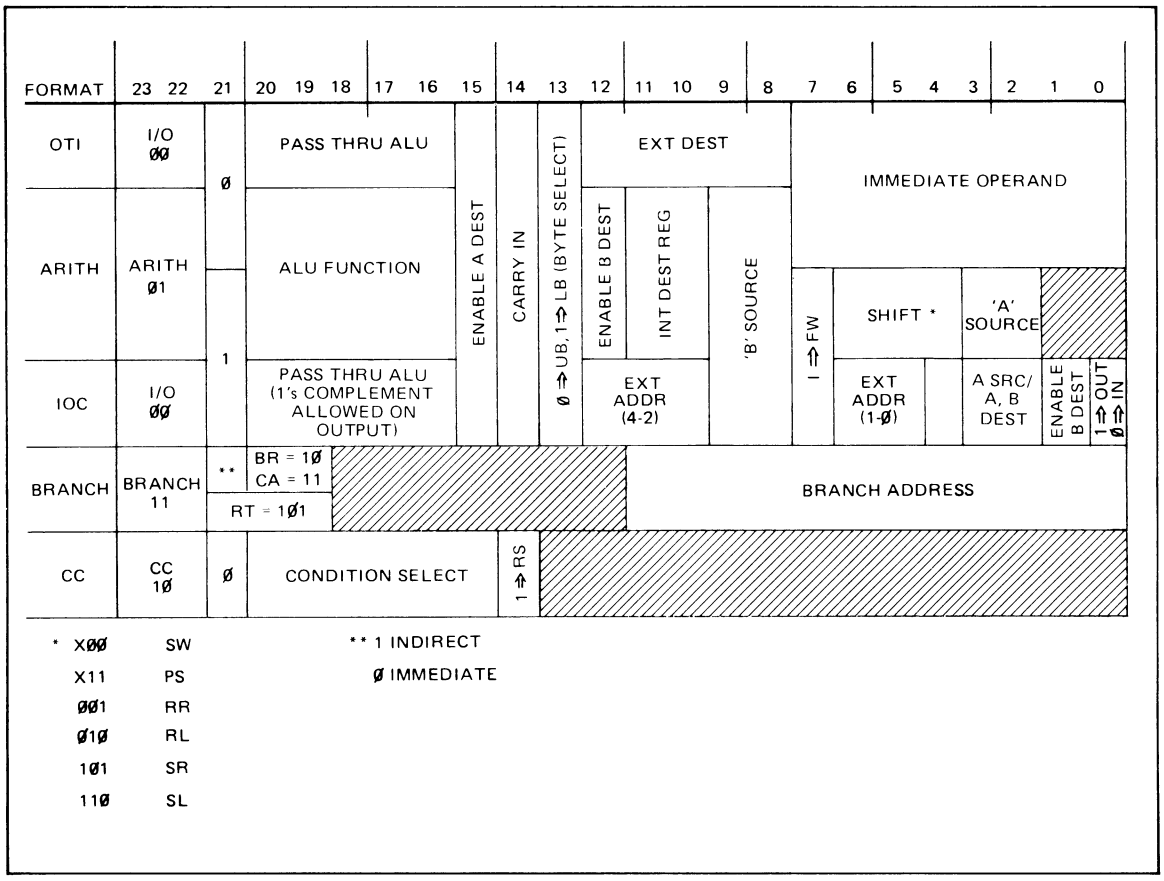


Figure 2. Instruction Format

U/L Field (ROM Bit 13)	If 0, immediate operand is put on the upper byte of the MIO bus; if 1, it is put on the lower byte.
EXT DEST Field (ROM Bits 8 – 12)	A five-bit field containing the address of the external destination of the immediate operand.
IMMEDIATE OPERAND Field (ROM Bits 0 – 7)	An eight-bit field containing binary information to be output to the address contained in the EXT DEST field.

ARITH — Arithmetic

A. With Immediate Operand (ROM Bit 21 = 0)

The ALU performs the operation specified by ROM bit 14 and ROM bits 16 – 20. It takes as inputs the immediate operand (ROM bits 0 – 7) from the A bus, and a B register specified by ROM bits 8 – 9. ROM bits 10 – 11 specify the number of the register into which the result is written, and ROM bits 15 and 12 specify whether an A register, a B register, both, or neither are written into. U/L (ROM bit 13) specifies which byte is affected. No shift is performed.

B. Without Immediate Operand (ROM Bit 21 = 1)

The ALU performs the operation specified by ROM bit 14 and ROM bits 16 – 20. It takes as inputs an A register specified by ROM bits 2 – 3 and a B register specified by ROM bits 8 – 9. A shift specified by ROM bits 4 – 6 is performed on the result. ROM bits 10 – 11 specify the number of the register into which the result is written, and ROM bits 15 and 12 specify whether an A register, a B register, both, or neither are written into. U/L (ROM bit 13) and FW (ROM bit 7) specify whether the upper byte, the lower byte, or both are affected.

ENABLE A DEST (ROM BIT 15) and
ENABLE B DEST (ROM BIT 12) FIELDS

Each bit, if set to 1, causes the result of the operation to be written into a register on the A or B bus, respectively. If both are set, an A and B register are paired and the effect is that of a single register available to both buses. If neither is set, the result is left in the T register until T3 of the next instruction. This allows computation of a final branch address without the need to write the information back into a register.

INT DEST REG FIELD (ROM BITS 10 – 11)

These bits contain the number of a register into which the result will be written. (ENABLE A DEST and ENABLE B DEST determine which group — A or B — that will be written into.)

A SOURCE (ROM BITS 2 – 3) and
B SOURCE (ROM BITS 8 – 9) FIELDS

Specify which register is put on each bus for an ALU operation.

IMMEDIATE OPERAND FIELD (ROM BITS 0 – 7)

This field represents an eight-bit number used in an ALU operation with a B register. Only one byte is affected (replaces A register).

U/L (ROM BIT 13) and
F/W (ROM BIT 7) FIELDS

These bits determine which byte(s) is (are) affected during an operation. If FW = 1, both bytes of the sources are used, and both bytes of the result are written in the destination. If only U or L is specified, only that byte of the operand and destination are affected. The other byte in both source and destination is left unaffected. (U/L = 0 specifies an upper byte; U/L = 1 specifies a lower byte.)

SHIFT FIELD (ROM BITS 4 – 6)

This three-bit field specifies what shift operation is to be done on the result from the ALU before the result is written into its destination:

ROM BIT			MNEMONIC	ACTION
6	5	4		
X	0	0	SW	Swap bytes (full word only)
X	1	1	PS	Pass no shift
0	0	1	RR	Rotate 1 bit right (full word only)
0	1	0	RL	Rotate 1 bit left (full word only)
1	0	1	SR	Shift 1 bit right
1	1	0	SL	Shift 1 bit left

IOC — Register I/O

A. Input to Register (ROM Bit 0 = 0)

Selects an external source and loads a selected register with the contents of that source. The external source address (ROM bits 5 – 6 and ROM bits 10 – 12) is put on the external address bus; external select (EX SEL) is true (high). An input strobe (INPUT) is sent out positive true. During T4, the data on the MIO bus (ground true) is written into the register(s) whose number is specified by ROM bits 2 – 3, and whose group is selected by ROM bits 15 and 1 (an A register, B register, both, or neither). If FW = 1 (ROM bit 7), the entire 16 bits of the MIO bus is recorded; if only U or L is true, then only the appropriate byte is recorded.

B. Output from Register (ROM Bit 0 = 1)

A selected register is output to a selected external destination. The source register is selected by ROM bits 2 – 3. Note that ROM bits 15 and 1 (enable A dest and enable B dest) are meaningless in this case. The assembler must know whether an A or B register is required, and insert PSA or PSB (Pass A or Pass B), or CMA or CMB (Complement A or Complement B) into the opcode field (ROM bit 14 and ROM bits 16 – 20). External select (EX SEL) is true during the entire instruction time (ground true). A shift specified by ROM bits 4 – 6 is performed on the output of the ALU. At time T4 UB OUT, LB OUT, or both (depending on U, L, and FW) is put out positive true.

ALU FUNCTION FIELD (ROM BIT 14 and ROM BITS 16 – 20)

The assembler must decide whether an A or B register is to be output, and insert PSA or PSB (or CMA or CMB) into this field; on input operations, this field is meaningless. For OTI, only Pass A may be used.

ENABLE A DEST (ROM BIT 15) and
ENABLE B DEST (ROM BIT 1) FIELDS

On input (ROM bit 0 = 0) conditions, the mode of operation is as described for the ARITH instruction; the fields are meaningless for output conditions (ROM bit 0 = 1).

U/L (ROM BIT 13) and
FW (ROM BIT 7) FIELDS

These are the same as described for ARITH instructions.

EXT ADDR FIELD (ROM BITS 0 – 4)

For output conditions (ROM bit 0 = 1), the field is as described for OTI. On input conditions (ROM bit 0 = 0), this field represents a five-bit address of an external source whose contents are to be stored in a processor register.

A/B SRC/DEST REG FIELD (ROM BITS 2 – 3)

On output conditions (ROM bit 0 = 1), this field contains the number address of the register to be output; it is used in conjunction with the ALU FUNCTION field in determining the specific register to be accessed. On input conditions (ROM bit 0 = 0), this field is used in conjunction with the ENABLE A DEST and ENABLE B DEST fields to specify the register(s) into which information is to be written.

I/O FIELD (ROM BIT 0)

A one-bit field specifying whether a register is to be input to or output from (0 = input, 1 = output).

BRANCH

A. JUMP (ROM BIT 20 = 1, ROM BIT 19 = 0)

If the CCB is false, the branch is not taken and the next instruction is executed. If the CCB is true, and if the branch is specified by bit 21 to be indirect, the contents of the T-register (temporary ALU result register) replace the contents of the ROM address register (i.e., the T-register contains the address of the next instruction). If bit 21 specifies a direct branch, then the address contained in ROM bits 0 – 11 is used as the address of the next instruction. The CCB is set true at the end of the instruction.

B. CALL (ROM BIT 20 = ROM BIT 19 = 1)

Operates just like a branch except that before the branch is taken (if and only if the CCB is true) the contents of the ROM address register are pushed onto the subroutine stack. The CCB is set true at the end of the instruction.

C. RTN (ROM BIT 21 = ROM BIT 19 = 1, ROM BIT 20 = 0)

If the CCB is true, the address in the top of the subroutine's stack replaces the address in the ROM address register, and the subroutine stack is popped. If more than three pops are made, the first-in address will always be used. If the CCB is false, the ROM address register is incremented. The CCB is set true at the end of the instruction.

SCC — Set Condition Code

Samples the condition specified (see below) and sets its value into the condition code bit (CCB). RS (reverse sense) may be used to negate the condition. Subsequent branches will be affected by the condition of this bit.

CONDITION SELECT	
01 0000 thru 01 1111	External Flags 00B – 17B
10 0000	External Flag 20B
10 0001	LOVER — Lower byte ALU carry
10 0010	EQUAL (A = B)
10 0011	UOVER (ALU overflow)
10 0100	TNZRO (T register ≠ 0)
10 0101	TMSB (Sign of T register results)
10 0110	TLSB (LSB of T register)
10 0111	FALSE (Unconditional False)

3.0 FUNCTIONAL DESCRIPTION

3.1 CLOCK (Refer to Sheet 3 of the microprocessor schematics)

U37 and its associated circuitry form an ECL oscillator. A 10 MHz crystal and a third harmonic tank circuit provides the 30 MHz master clock frequency. C7 is adjustable for a 30 MHz waveform at test point X1. U31 and U35 select either this internal clock or an external clock and route it to the clock phase ring counter, which is made up of U13, U14, U16, and U56. The ring counter generates the 6-phase clock waveforms shown in figure 3. It can be thought of as a shift register, of which only one bit at a time is ever high. On the rising edge of every 30 MHz clock pulse, the bit shifts to the next higher stage. During T5, all the inputs to U56 are high, so the D input to U13 is high. Thus on the next rising edge of the 30 MHz clock, T0 goes high, and the clock phases start a new cycle. Because each phase of the clock is 33.3 nsec, an entire instruction will require 200 nsec for completion.

U67 divides the 30 MHz clock to 15 MHz before transferring it to the device controller board. This scheme was employed to reduce the electromagnetic interface (EMI) that introduces noise into the system.

3.2 ROM

The microprocessor control memory consists of twenty-four 256 × 4 bit read only memory chips located on the ECC/ROM PCA. These ROM's are organized as four banks (horizontal rows) of six ROM's per bank. This gives an instruction word length of 24 bits, and allows 1024 instructions to be stored in ROM. Since the ROM address field is 12 bits wide, the ROM size could easily be expanded to 4096 words.

Bank selection is done by U91 of the ECC/ROM board, which looks at the two most significant bits of the ROM address field (RAR 8 and RAR 9). The ROM's may be either 3-state or open collector. If open collector, four resistor pull-up packs must be loaded.

3.3 INSTRUCTION FLOW (Refer to Sheet 1 of the microprocessor schematics)

Figure 4 outlines the mechanism by which the ROM is accessed. The ROM address register (U33, U53, and U73) contains the address of the next instruction in ROM. The ROM output register (U61, U71, U81, U91, and U155) latches the 24-bit instruction while it is being executed and a new instruction is being accessed in ROM.

3.3.1 TIMING

If the RAR is clocked when PE (parallel enable) is low, the RAR loads; i.e., each output (0_0, 0_1, 0_2, 0_3, on each 74161 IC) assumes the value of the input (P_0, P_1, P_2, P_3). If it is clocked (by a rising edge on CP) when PE is high, the output is incremented and no loading takes place. Incrementing the RAR causes access to the next sequential instruction in ROM; loading the RAR causes a branch, where the next instruction's address has no relation to the current instruction's address.

The RAR has been designed to increment on the rising edge of T0 of every instruction. Therefore we want PE on the RAR to be high as T0 rises. Similarly, we want PE to be low as T2 rises, because during a branch the RAR will be clocked at T2 in order to load it with the branch address. As shown in figure 5, flip-flop U55 satisifes these conditions, putting a square wave on PE which is high as T0 rises and low as T2 rises.

U75 creates the RAR clock: a rising edge on T0 always, and a rising edge on T2 during a successful branch instruction.

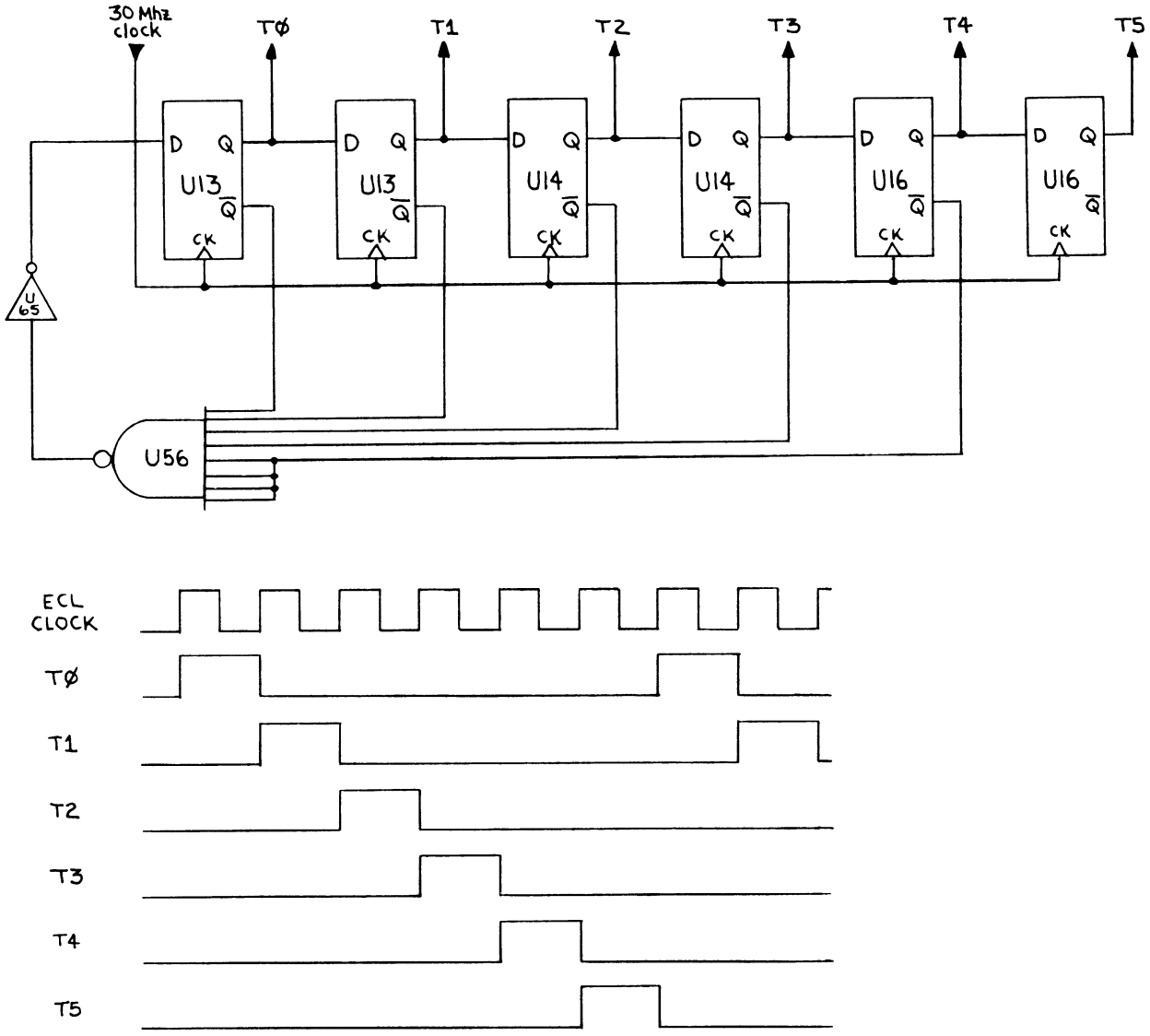


Figure 3. Clock Phase Generation

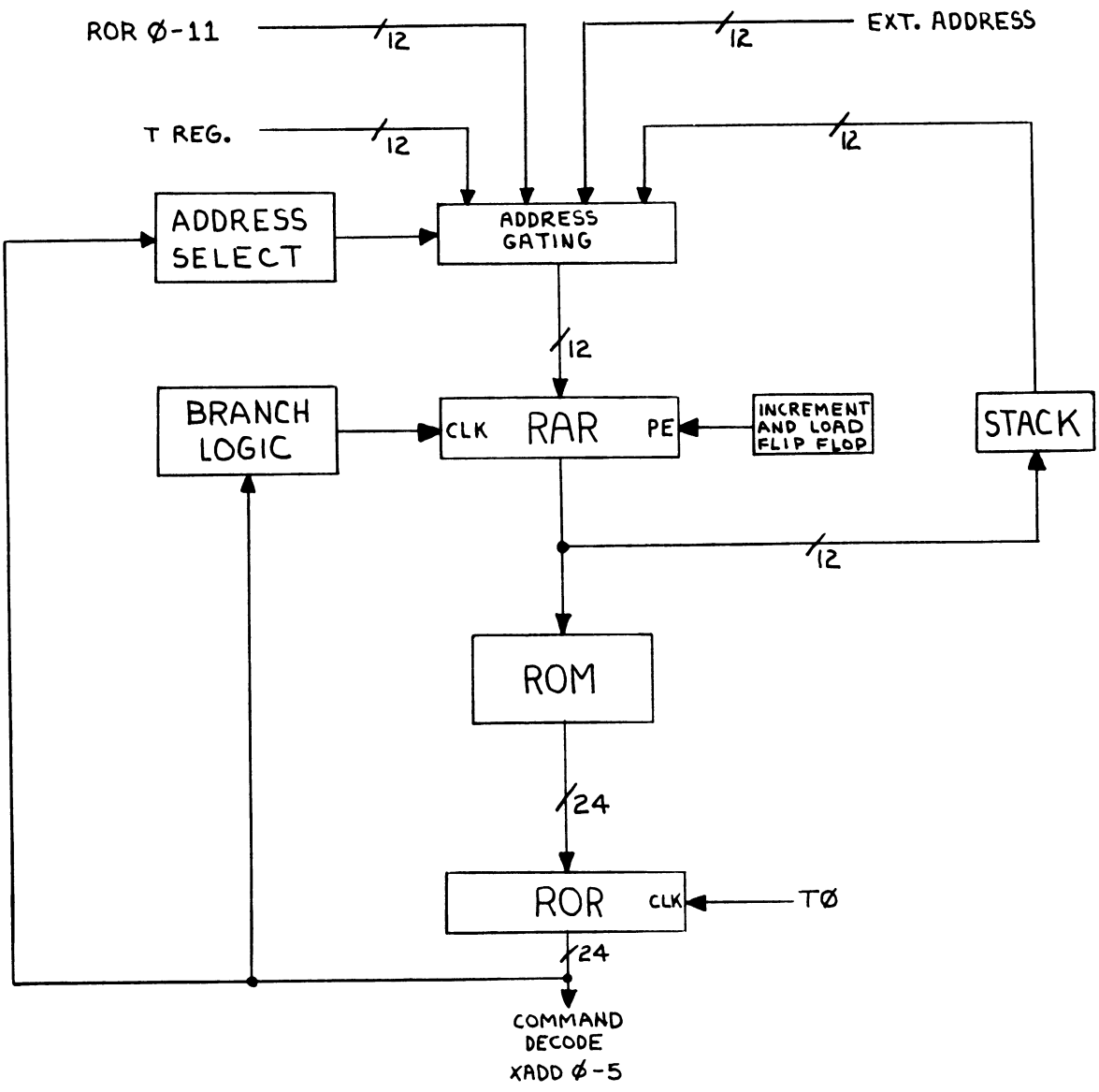


Figure 4. ROM Address Logic

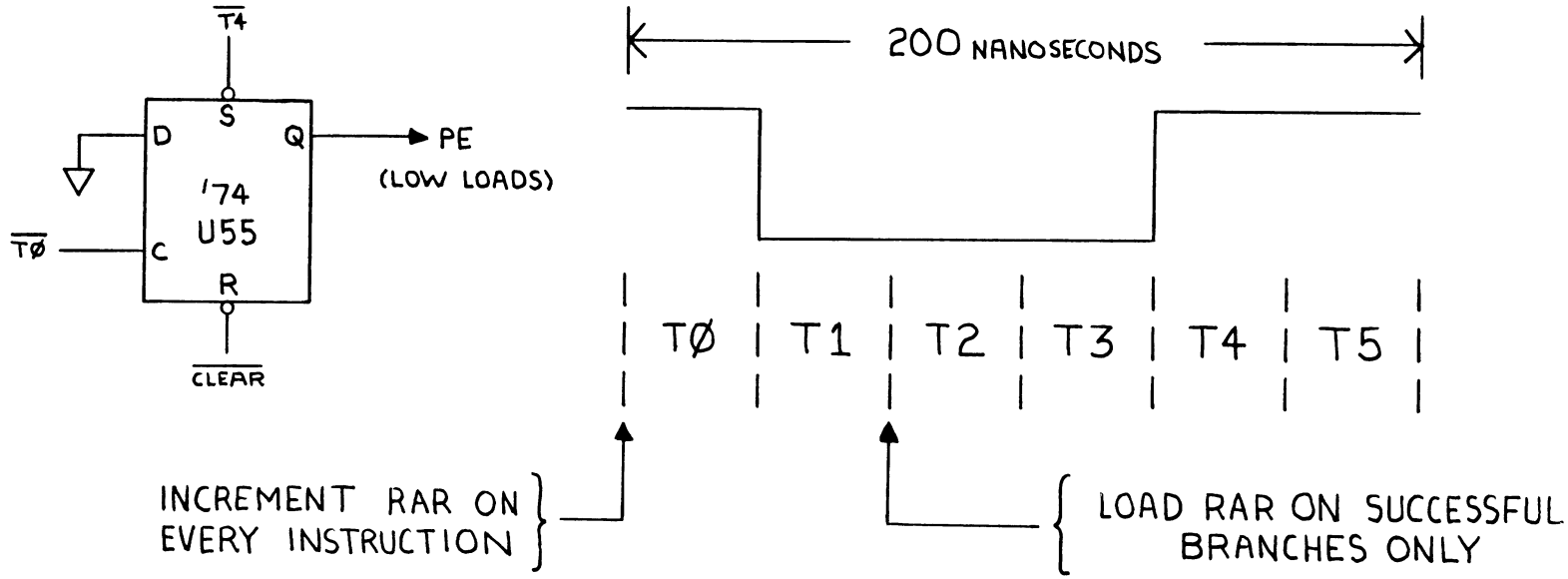


Figure 5. Increment/Load Flip-Flop and Timing

The ROR is always clocked on T0, so every instruction starts at T0. The ROM has completed its access; it is presenting the instruction word as the T0 clock rises and latches this word into the ROR. Recall that T0 is also the phase which increments the RAR. Thus we are strobing data out of the ROM on the same clock edge as we are changing the address to the ROM. No problem occurs since the ROM acess time is always greater than the ROR setup time. This scheme allows the ROM to be accessing for the full instruction time prior to the T0 edge which strobes its contents into the RAR. Two processes are thus occurring simultaneously: the instruction is being decoded and executed, and the next instruction is being looked up in ROM.

During a successful branch instruction, the RAR is loaded at T2 (refer again to figure 5).

3.3.2 RAR ADDRESS GATING

U22, U32, U42, U52, U62, and U72 are multiplexers which allow the RAR to be loaded with an address from one of four possible sources:

1. ROR bits 0 – 11 (corresponding to a direct branch); the destination address is the 12 least significant bits of the ROM instruction word.

2. an external address which has been keyed in from a service device attached to the controller; this feature allows an operator to start executing the microprogram at any desired address for diagnostic purposes.

3. the contents of the top level of the subroutine stack; this occurs when a return from a firmware subroutine is executed.

4. the contents of the T-register, which latches the output of the ALU on every instruction; this allows a microprogram to indirectly branch through a branch table in ROM, depending on the contents of the T-register.

Which of these four possible branches is taken depends on the levels on the 1Y and 2Y inputs (pins 2 and 14) of the RAR multiplexers. These levels are determined by U77 in accordance with the following table:

RAR Multiplexer

ROR 21	ROR 20	Selected
0	0	external address
0	1	direct branch
1	0	subroutine return
1	1	branch to T register contents

3.3.3 CONDITION CODE

Conditional branching in this microprocessor requires two instructions. The first instruction sets a condition code flip-flop (U55) depending on the state of an external flag or an internal condition. The second instruction is a branch instruction which succeeds when the condition code flip-flop is set, and fails when it is in the reset state. No loading of the RAR at T2 occurs on a branch that fails.

During a set condition code instruction, U121 or U131 selects an external flag (external to the microprocessor PCA) or a T-register condition, depending on bits 15 – 20 of the micro-instruction word. ROR 14 determines whether the selected condition is to be checked for true or false. It is exclusive-ORed with the output of U121 and U131 in U11 and fed to the condition code flip-flop U55. This data is clocked into the flip-flop only at T3 of a set condition code instruction. The preset input of this flip-flop goes low at T4 of a branch instruction, causing the flip-flop to set. This function was added so that if two branches are executed without an intervening SET CC instruction, the second branch will always succeed.

3.3.4 SUBROUTINE STACK

Subroutines are a useful feature of many programs because they allow a given section of code to be executed at many points within a program without the physical repetition of that code. When the programs are stored in a ROM, a means must be provided for storing the return addresses, so that the main program can resume at the correct place after the subroutine is complete. In this microprocessor, a 3-level, 12-bit subroutine stack provides this function. Composed of U23, U24, U34, U44, U54, U63, U64, and U74, it can store up to three return addresses, allowing three levels of subroutine nesting. These chips are wired so that during a CALL instruction, the contents of the RAR (which has been incremented to become the correct return address) is pushed onto the top of the stack (U34, U54, and U74). Also, the former contents of the top level of the stack are pushed onto level 2 (U24, U44, and U64), and the former contents of level 2 are pushed onto level 3 (U23 and U63). The contents of level 3 are lost during a CALL instruction. The opposite direction of data flow occurs during a RETURN instruction:

LEVEL 3 → LEVEL 3
LEVEL 3 → LEVEL 2
LEVEL 2 → LEVEL 1 (top)
LEVEL 1 → RAR

Every flip-flop in the stack is clocked at the rising edge of T2, on both a CALL and a RETURN instruction. The signal generated at pin 8 of U43 (STACKOP) supplies a positive-going edge at this time to clock U23 and U63. The other latches (U24, U34, U44, U54, U64, and U74) requires a negative edge to trigger, so U65 inverts STACKOP for them.

The WSEL (word select) input to the level 1 and level 2 flip-flops determines which set of inputs they read (and hence whether "popping" or "pushing" will take place). WSEL is high during a CALL and low during a RETURN.

4.0 ALU (Refer to Sheet 2 of the microprocessor schematics)

Arithmetic and logical operations are performed on two 16-bit words by the four ALU chips (U106, U126, U136, and U156). The function to be performed is determined by bits 16 through 20 of the arithmetic instruction word. These five bits are thus wired directly from the ROR to the M, S0, S1, S2, and S3 inputs of each of the ALU's. Full carry look-ahead is achieved by the 74S182 (U135). The ALU's always operate on two data words: one on the A bus, and one on the B bus (refer again to figure 1). No clocking is provided on the ALU chips. Rather, the data registers feeding the ALU's and the T-register following the ALU are clocked. Two sources exist for the data inputs to the ALU's: registers and immediate operands.

4.1 IMMEDIATE OPERANDS

Two types of instructions put immediate operands on the A bus: an immediate output, and an immediate arithmetic. U97 creates GIMOP (gate immediate operand) for these two instructions, which goes to one enable of U141 and U151 (the immediate operand drivers). The other enable of U151 goes to LB (lower byte), while the other enable of U141 goes to UB (upper byte). Immediate operands are only 8 bits long, and LB and UB determine whether they will be put onto the upper or lower byte of the A bus. Because the immediate operand replaces the A register during an immediate operand instruction, the A register outputs are not enabled so that there is no conflict.

4.2 REGISTERS

The four 16-bit A registers and four 16-bit B registers are implemented with eight 4 × 4 register files (U122, U123, U132, U133, U142, U143, U152, and U153). How these are assigned to upper and lower bytes of A and B registers is shown on sheet 2 of the microprocessor schematic. The data inputs to these registers (D_0, D_1, D_2, and D_3) are tied to the MIO bus. The data outputs go to the ALU inputs. The R_A and R_B control inputs specify the address (0, 1, 2, or 3) within each register file to be read out of. The W_A and W_B inputs specify one of four addresses to be written into. Two active low control enables, GW and GR, are also needed on each register file. They act as read and write clocks.

4.2.1 WRITE

When GW is low, data is written from the MIO bus into the register file. On sheet 2 of the microprocessor schematic, U124, U134, U144, and U154 combine $\overline{\text{AWEN}}$, $\overline{\text{BWEN}}$, $\overline{\text{LBG}}$, and $\overline{\text{UBG}}$ to lower the appropriate GW's during T4 of any instruction which requires writing into a register. Arithmetic instructions and I/O input instructions require writing into a register. The signal $\overline{\text{CLD}}$ (clock data) reflects this fact (it is the NOR of IOIN and ARITH on sheet 3 of the microprocessor schematic, and must be low to allow writing into a register).

$\overline{\text{LBG}}$ (lower byte gated) — same as LB except only goes low during T4.

$\overline{\text{UBG}}$ (upper byte gated) — same as UB except only goes low during T4.

$\overline{\text{AWEN}}$ — A register write enable.

$\overline{\text{BWEN}}$ — B register write enable.

4.2.2 READ

When GR is low, data is read out of the register file and onto the ALU input lines. $\overline{\text{LB}}$ is tied directly to GR of the B register lower byte, and $\overline{\text{UB}}$ goes directly to GR of the B register upper byte. Thus one byte or both bytes of the B register is always sent to the ALU inputs during an arithmetic or I/O instruction. U144, however, only allows $\overline{\text{LB}}$ or $\overline{\text{UB}}$ to drive GR low on the corresponding A register when GIMOP (gate immediate operand) is low. Thus the A inputs to the ALU are fed either by an immediate operand, or by the A register, but not both.

4.3 T-REGISTER

U94, U104, and U114 comprise a 16-bit register which latches the result of an ALU operation on the trailing edge of T2. U134 combines $\overline{\text{T2}}$ and $\overline{\text{ENSHIFT}}$ to clock this T-register. $\overline{\text{ENSHIFT}}$ is generated by U97 pin 8 to be low only when ROR 23 is low and IOIN (I/O INPUT) is low; i.e., during an ARITH, OTI, or I/O OUT instruction.

4.4 PROCESSOR STATUS REGISTER

During any instruction which clocks the T-register, the state of the T-register is latched in the 6-bit processor status register (U111). U95, U115, and U124 NAND together the 16 ALU outputs to produce TNZRO, which is high if any bit (or bits) of the ALU output is one. EQUAL goes high if the words on the A and B input busses to the ALU are the same. UOVER and LOVER go high if there is a carry out of the upper or lower bytes, respectively. MSB and LSB go high if the most significant bit or least significant bit, respectively, are ones.

The 6 outputs of the processor status register go to U121, where any one of them can be tested by a SET CC instruction.

4.5 SHIFTER

U82, U83, U92, U93, U102, U103, U112, and U113 form the shifter, which receives the output of the ALU, shifts it, and puts it onto the MIO bus. Allowable shifts are shift left, shift right, rotate left, rotate right, swap bytes, and pass. Except for bits 0 and 15, the shifter selects each bit from one of four possible sources. Bit i ($i \neq 0, 15$) comes from:

1. bit i of the T register (pass)
2. bit $i - 1$ of the T register (shift left, rotate left)
3. bit $i + 1$ of the T register (shift right, rotate right)
4. bit $i \pm 8$ of the T register (swap bytes)

This observation allows the shifter to be built from one-of-four data selectors, rather than a complex shift register. Each of the 8 shift chips contains two one-of-four data selectors. The inputs are wired to the T-register in accordance with the above table. The outputs of the shifter go directly to the MIO bus. U124 makes the distinction between shift and rotate for bits 1 and 15 in accordance with $\overline{\text{ROT}}$, which comes from ROR 6.

The select inputs to the shifter chips are functions of ROR 4, ROR 5 and PASS:

	ROR 6	**ROR 5**	**ROR 4**
Swap	X	0	0
Rotate right	0	0	1
Shift right	1	0	1
Rotate left	0	1	0
Shift left	1	1	0
Pass	X	1	1

From the above table, $\text{ROR 4} \cdot \overline{\text{ROR 5}}$ = rotate or shift right
$\overline{\text{ROR 4}} \cdot \text{ROR 5}$ = rotate or shift left
$\text{ROR 4} \cdot \text{ROR 5}$ = pass

PASS is GIMOP or I/O OUT (page 3 of the microprocessor schematics) since no shift should occur on instructions involving immediate operands or on I/O output instructions.

Pin 3 of U144 and pin 3 of U154 control the enables on the shifter chips. $\overline{\text{LB}}$ and $\overline{\text{UB}}$ determine whether the upper byte, the lower byte, or both bytes of the shifter output will be enabled onto the MIO bus. In addition, $\overline{\text{ENSHIFT}}$ must be low for the shifter to drive the MIO bus. This happens for ARITH, OTI, and I/O OUT instructions.

4.6 SIGNAL POLARITY

The MIO bus is ground true. A logical one is a zero volt signal and a logical zero is a 4.7 volt signal. The input bus to the ALU's is also ground true. The immediate operand drivers which drive this bus (U141 and U151) are inverters, since the immediate operand field of a micro-instruction comes out of the ROR (and ROM) plus true. The pull-up resistors on the 74170 register files ensure that when a file's output is not enabled, the line goes to +5.0 volts (which the input of the ALU regards as a logical zero). Thus when an ARITH instruction specifies operations on only one byte, the register files corresponding to the other byte are not enabled, and the ALU sees logical zeros on the inputs corresponding to the disabled byte.

5.0 INSTRUCTION DECODING

(Refer to Sheet 1 of the microprocessor schematics)

Besides the ROR and the clock circuitry, all of the chips on sheet 3 of the schematic perform instruction decoding. Based on the current instruction in the ROR, they generate the signals which operate the ALU. U86 decides which of the four basic types of instruction is being executed. It is a 2-line-to-4-line decoder which decodes bits 23 and 22 of the ROR into one of the following:

1. IOINST = $\overline{\text{ROR 23}} \cdot \overline{\text{ROR 22}}$ (also EX SEL to the device controller board)
2. ARITH = $\overline{\text{ROR 23}} \cdot \text{ROR 22}$
3. BRINST = $\text{ROR 23} \cdot \text{ROR 22}$
4. SET CC = $\text{ROR 23} \cdot \overline{\text{ROR 22}}$

5.1 U101

U101 is a quad 2-input data selector that selects XADD0 and XADD1 from one of two sources. During an OTI (output immediate) instruction, they come from ROR bits 8 and 9. During an IOC instruction, they come from ROR bits 8 and 9. During an IOC instruction, they come from ROR bits 5 and 6. In both cases, XADD0 and XADD1 go through the P2 connector to the device controller board, where they join XADD 2, 3, 4, and EX SEL in selecting an external register.

ID1 and ID2, which determine what word (0, 1, 2, or 3) of a register is written into, have two possible sources. During an ARITH instruction, they come from ROR bits 10 and 11. During an IOC instruction, ROR bits 2 and 3 are the sources.

5.2 I/O DECODING

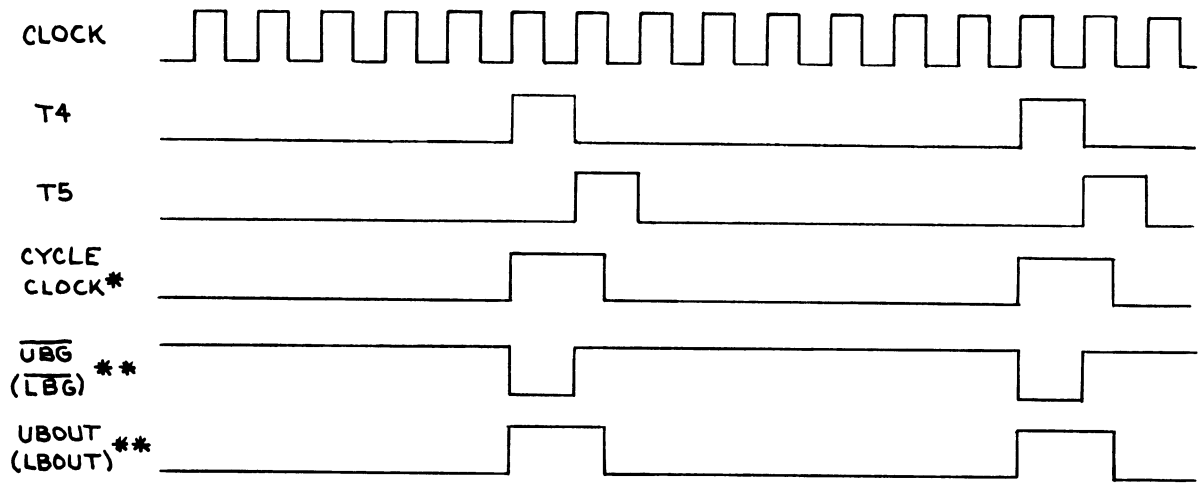
U96 (pins 11 and 12) separates all I/O instructions into either OTI (output immediate) or IOC (I/O control). U86 (pins 6 and 9) further divides IOC instructions into input (INPUT) or output (I/O OUT) types on the basis of ROR 0. I/O OUT is ORed with $\overline{\text{ROR 23}} \cdot \overline{\text{ROR 21}}$ to form PASS, which directs the shifter not to shift. INPUT is sent through P2 to the device controller board where it determines whether some of the registers drive or receive from the MIO bus.

The IOIN (as well as ARITH) command allows selection and subsequent writing into a register. The register group is selected by U145. The $\overline{\text{AWEN}}$ and $\overline{\text{BWEN}}$ signals are generated by ROR bit 15 and ROR bit 1, respectively (ROR bit 15 and ROR bit 12 for ARITH commands). As previously described, U101 provides gating for the number of the selected group (via ID1 and ID2) into which the information is written (i.e., A3 or B0, etc.).

5.3 BYTE SELECTION

U96 (pins 6 and 9) creates $\overline{LB}$ and $\overline{UB}$ based on ROR bit 13 (U/L field), ROR bit 7 (FW field), and ROR bits 21 and 23. $\overline{LB}$ specifies lower byte when it is low, and $\overline{UB}$ specifies upper byte. When both are low, a full word instruction is specified. U85 (pins 8 and 6) creates $\overline{LBG}$ $(\overline{LB \cdot T4})$ and $\overline{UBG}$ $(\overline{UB \cdot T4})$, which are used in the ALU section.

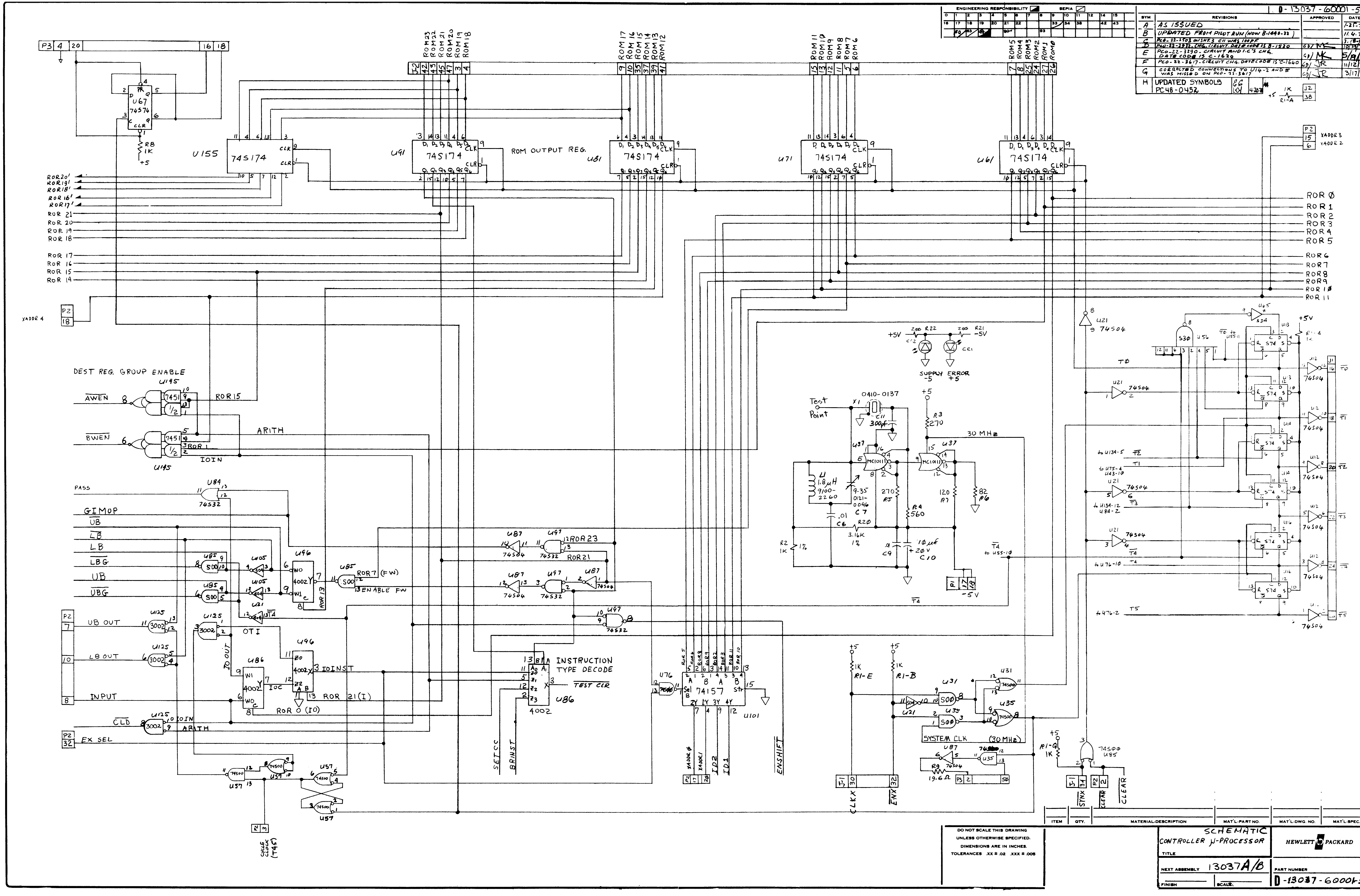
UBOUT and LBOUT are generated by U125 (pins 11 and 6) and sent to the device controller board. They are the same as UB and LB in that they become true at the beginning of T4, but they return false at the middle of T5 (instead of at the end of T4); this falling edge latches the contents of the MIO bus into the appropriate device controller register. The middle of T5 had to be chosen to ensure enough data propagation time, as attenution of the signal occurs before it reaches the device controller PCA. U57 creates this positive-going stretched pulse called CYCLE CLOCK, which remains high through the first half of T5 and which strobes LBOUT and UBOUT (see figure 6).

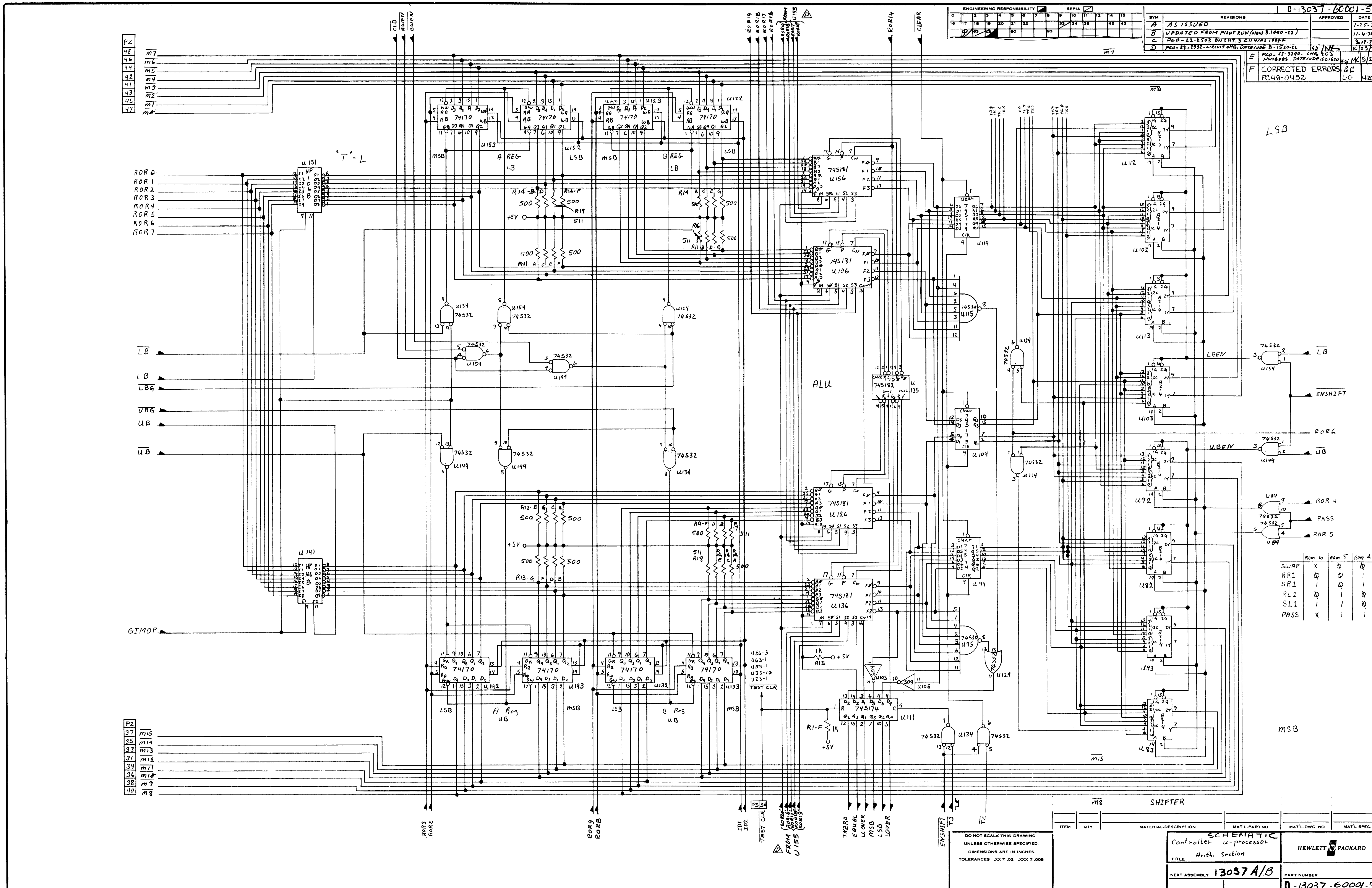


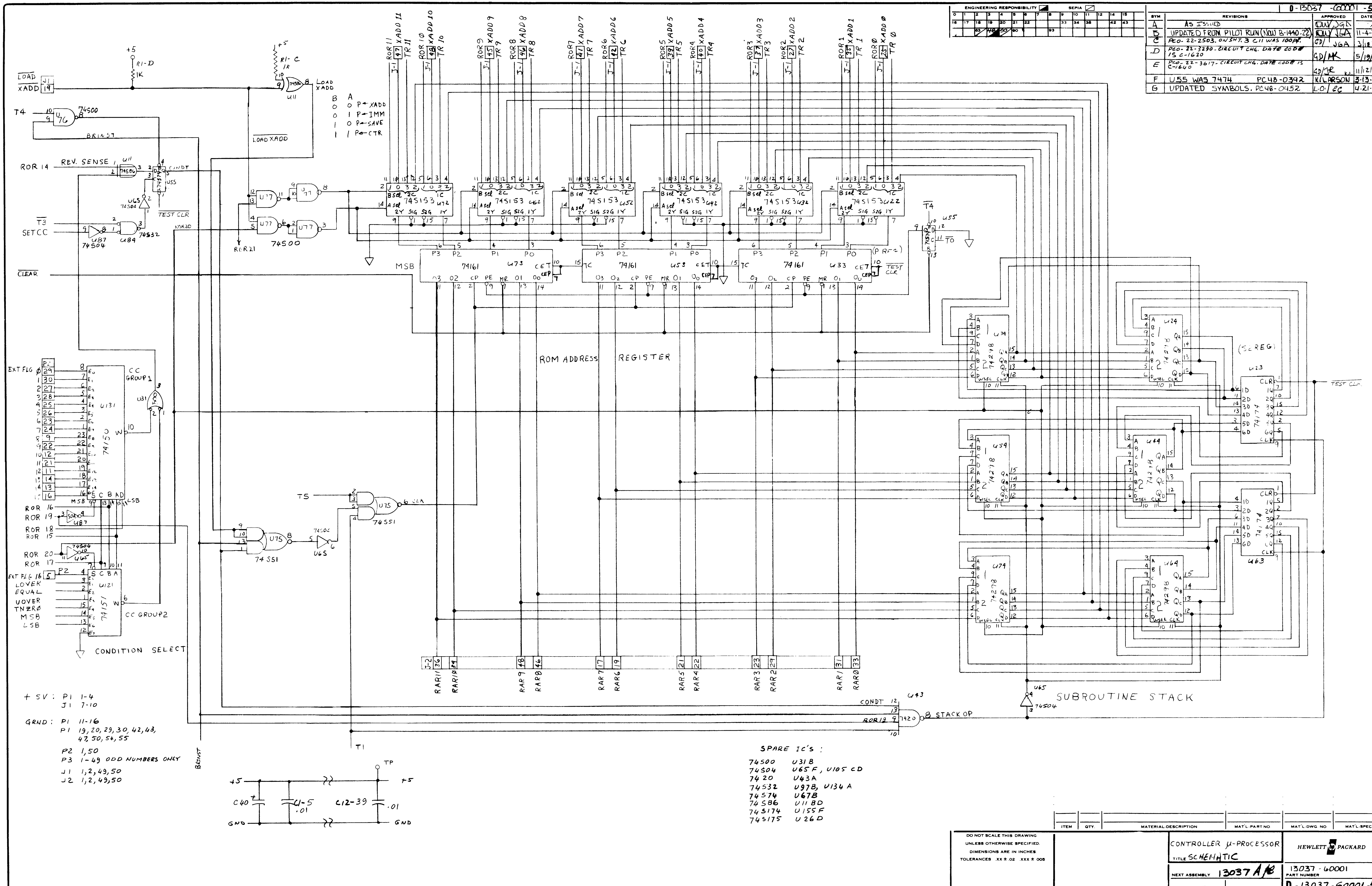
* ALWAYS PRESENT (GATED TO LBOUT AND UBOUT LOGIC THROUGH U57 BY AN OTI OR IOOUT INSTRUCTION).

** ONLY PRESENT WHEN ENABLED BY UB OR LB SELECTION (U/L FIELD OR FW FIELD).

Figure 6. Byte Selection Timing


D-13037-60001-53
SYM | REVISIONS | APPROVED | DATE
A AS ISSUED 1-25-73
B UPDATED FROM PILOT RUN (NOW B-1440-22) 11.4.74
C PCO-22-2503 ON SHT.3 C11 WAS 100PF
D PCO-22-2932. CHG. CIRCUIT. DATE CODE IS B-1530
E PCO-22-3290- CIRCUIT AND I.C'S CHG. DATE CODE IS C-1620
F PCO-22-3617- CIRCUIT CHG. DATE CODE IS C-1640 11/12/76
G CORRECTED CONNECTIONS TO U14-2 AND 5 WAS MISSED ON PCO-22-3617 3/17/77
H UPDATED SYMBOLS PC48-0452
ROM OUTPUT REG.
74S174
DEST REG. GROUP ENABLE
AWEN
BWEN
ARITH
IOIN
PASS
GIMOP
UB OUT
LB OUT
INPUT
CLD
EX SEL
INSTRUCTION TYPE DECODE
TEST CLR
ENABLE FW
IOINST
SET CC
BRINST
ENSHIFT
74157
Test Point
0410-0137
30 MHz
SUPPLY ERROR
SYSTEM CLK (30 MHz)
CLKX
ENX
SYNX
CLEAR
CYCLE CLOCK (T45)
DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED.
DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005
SCHEMATIC
CONTROLLER μ-PROCESSOR
NEXT ASSEMBLY 13037A/B
HEWLETT PACKARD
PART NUMBER
D-13037-60001-53


D-13037-60001-52
ENGINEERING RESPONSIBILITY
SEPIA
SYM
REVISIONS
APPROVED
DATE
A AS ISSUED 1-25-73
B UPDATED FROM PILOT RUN (NOW B-1440-22) 11-4-74
C PCO-22-2503 ON SHT. 3 C11 WAS 100PF.
D PCO-22-2932 - CIRCUIT CHG. DATECODE B-1530-22
E PCO-22-3290 - CHG. IC'S NUMBERS. DATECODE 1561620
F CORRECTED ERRORS PC48-0452
P2
m7
m6
m5
m4
m3
m2
m1
m8
CLD
AWEN
BWEN
ROR19
ROR18
ROR17
ROR16
FROM U155
ROR14
CLFAR
74170
A REG
B REG
LSB
MSB
LB
UB
"T" = L
u151
u141
ROR 0
ROR 1
ROR 2
ROR 3
ROR 4
ROR 5
ROR 6
ROR 7
+5V
74S181
74S182
74S32
74S30
74S174
ALU
LB
LBG
UBG
UB
GIMOP
LBEN
UBEN
ENSHIFT
ROR6
ROR 4
PASS
ROR 5
LSB
MSB
SHIFTER
m15
m14
m13
m12
m11
m10
m9
TEST CLR
P3 34
ROR3
ROR2
ROR9
ROR8
ID1
ID2
TMZRO
EQUAL
LOVER
MSB
LSD
LOVER
ENSHIFT
T3
T2
ROM 6
ROM 5
ROM A
SWAP X Ø Ø
RR1 Ø Ø 1
SR1 1 Ø 1
RL1 Ø 1 Ø
SL1 1 1 Ø
PASS X 1 1
ITEM
QTY.
MATERIAL-DESCRIPTION
MAT'L-PART NO
MAT'L-DWG. NO
MAT'L-SPEC.
DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED.
DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005
SCHEMATIC
Controller u-processor
Arith. Section
TITLE
HEWLETT PACKARD
NEXT ASSEMBLY 13037 A/B
PART NUMBER
D-13037-60001-52
FINISH
SCALE



D-13037-60001-51

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED		
B	UPDATED FROM PILOT RUN (NOW B-1440-22)		11-4-74
C	PCO-22-2503. ON SHT. 3 C11 WAS 100PF.	GD/ JGA	3/12/75
D	PCO-22-3290 - CIRCUIT CHG. DATE CODE IS C-1620	GD/MK	5/19/76
E	PCO-22-3617 - CIRCUIT CHG. DATE CODE IS C-1640	GD/JR	11/12/76
F	U55 WAS 7474 PC48-0392	K/LARSON	3-13-78
G	UPDATED SYMBOLS. PC48-0452	L.O./ EC	4-21-78

ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L PART NO	MAT'L DWG NO	MAT'L SPEC

DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED
DIMENSIONS ARE IN INCHES
TOLERANCES XX ± .02 XXX ± .005

CONTROLLER μ-PROCESSOR
TITLE SCHEMATIC

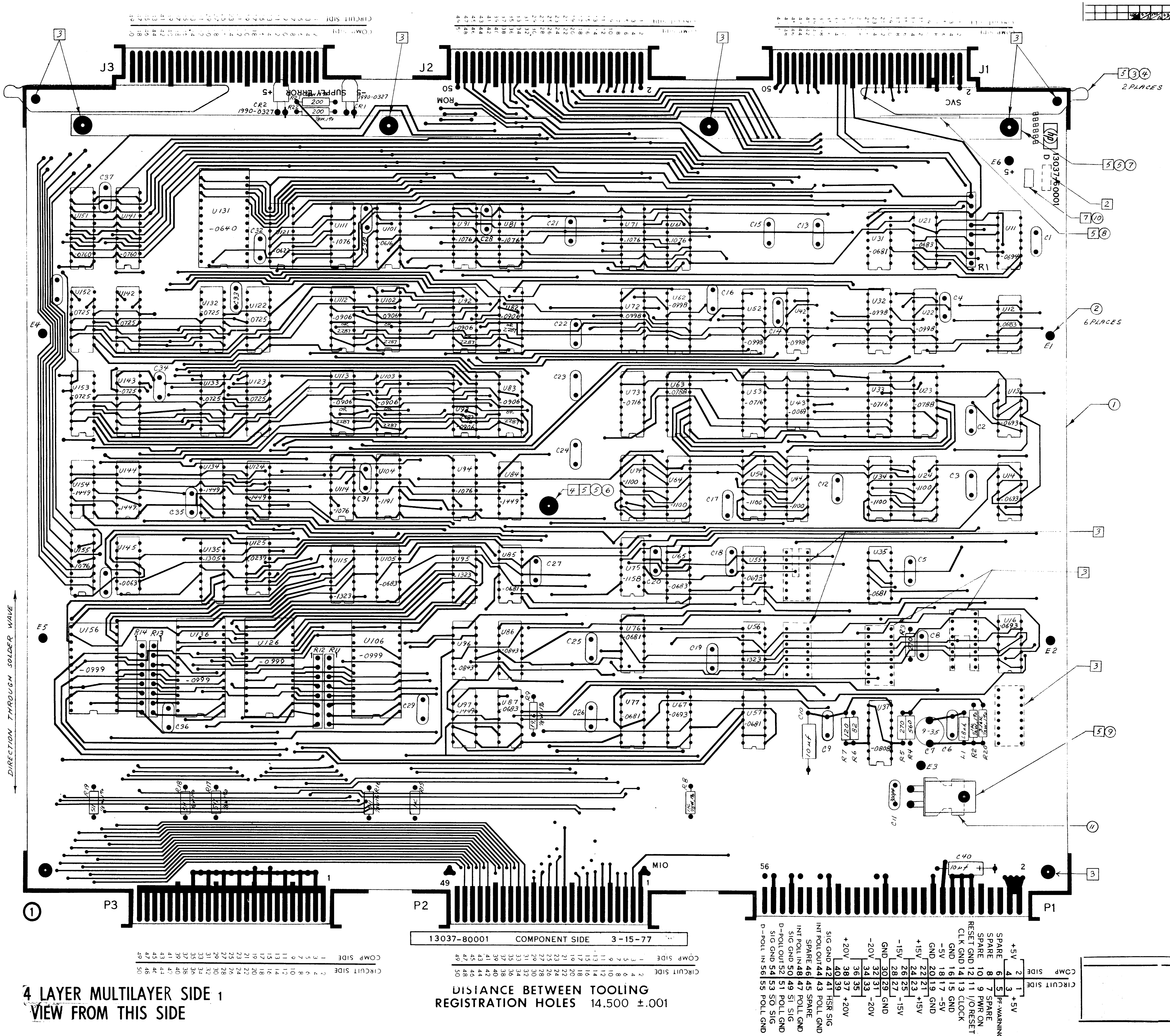
HEWLETT PACKARD

NEXT ASSEMBLY 13037 A/B

13037-60001
PART NUMBER

FINISH — SCALE —

D-13037-60001-51



F-13037-60001-E

A	AS ISSUED PC 48-0378
B	U55 WAS 1820-0077 PER PC 48-0392
C	CORRECTED DATE CODE. PC48-1198.
D	U12 WAS -0688, U156 SHORTENED, R1 NOW LABELED. PC48-1851.
E	ADDED 2287 TO LOC. U82, 83, 92, 93, 102, 103, 112, 113. PC48-2113.

NOTES:

1. UNLESS OTHERWISE SPECIFIED
   ALL RESISTANCE IN OHMS
   ALL RESISTORS 1/4W, 5%
   ALL RESISTOR NETWORKS 1810-0080
   ALL CAPACITANCE IN MICRO FARADS
   ALL CAPACITORS .01 CERAMIC DISC.
   ALL IC'S ARE 1820-

[2] MARK DATE CODE 1808

[3] MASK AS INDICATED PRIOR TO SOLDERWAVE

[4] USE SUPPORT FIXTURE DURING WAVE SOLDERING

[5] INSTALL ITEMS (3) THROUGH (9) IN TOUCH UP

6. CLEAN CONNECTORS WITH ALCOHOL IN TOUCH UP

[7] ITEM (10) STAMPED WITH MONTH & YEAR AND INSTALLED AFTER FINAL TEST

8. SCHEMATIC 2-13037-600[illegible] -51, -52, -53

ITEM	QTY	PART DESCRIPTION	PART NUMBER
11	1	SOCKET XTAL	1200-0546
10	1	BLANK LABEL	7120-5430
9	1	CRYSTAL	0410-0137
8	1	LABEL	7120-4325
7	1	BRACE, P.C. BOARD	5040-6058
6	1	SPACER	5020-7318
5	5	SCREW	0624-0077
4	2	EXTRACTOR	5040-6009
3	2	PIN	1480-0116
2	6	TERMINAL E1-E6	0360-0224
1	1	BOARD ETCHED	13037-80001

MICRO PROCESSOR ASSEMBLY DRAWING

HEWLETT hp PACKARD

SEE KARDEX

1 2X

13037-60001

F-13037-60001-E

CONTENTS

DEVICE CONTROLLER PCA

1.0 DEVICE CONTROLLER CHARACTERISTICS

The device controller PCA, like the microprocessor and ECC PCA's, resides on a single 3000-size printed circuit board. This board communicates with the microprocessor PCA via the MIO bus, five external address lines, and various flags; in turn the device controller sends and takes information from the disc drive and CPU interface (via various dedicated bus lines and flags), as well as controlling the operation of the ECC/ROM board. 5 volt current consumption is 4.5 amps.

Communication of data to or from the disc drive requires proper formatting or separating of that data. The process of formatting the information requires proper mixing of the 7.5 MHz data rate clock with the data to be recorded such that upon reading, these two signals can be properly separated. An MFM (modified frequency modulation) type encoding is used, employing the following rules:

1) a transition is made in the middle of all "1" bit cells;

2) a transition is made at the boundary of two adjacent "0" bit cells.

Such encoding helps minimize effects of pulse crowding as well.

Before the encoded data is passed to the disc drive for writing, precompensation of the data stream is employed to circumvent the effects that the disc media introduces upon recording. These effects include those of d.c. offset upon playback, and overlapping effects of crowded pulses.

As previously mentioned, reading data from the disc requires separating the data clock from the written data. Allowances also must be made for the fact that the speed of the disc upon playback may differ from that when recording. Thus, the separator locks onto the nominally 7.5 MHz data stream and, utilizing a phase lock loop and appropriate decoding hardware, obtains the original recorded data. The data clock separated from the encoded data is used to clock other hardware, including the ECC hardware (described in the next section) and the CRCC (cyclic redundancy code check) hardware.

The CRCC section of the device controller originally was to be the only resident error indicator employed on the 13037A. However, the ECC hardware became standard equipment before the disc controller was released, and hence the CRCC is used only to indicate any data error. (The ECC decides whether it is correctable, and what is needed to correct it.)

During a WRITE operation, serial data is shifted through the CRCC hardware; at the end of the data field, a characteristic CRCC word has been produced for the preceding data, and this 16-bit word is now serially transferred to the disc. Upon reading, all information through (and including) the CRCC word is clocked through the CRCC hardware; at the end of the CRCC word the contents of the CRCC register are examined, and if any bit is not zero, the CRCC hardware reports a data error via an appropriate flag.

In reading or writing operations, one encounters times when serial data must be converted to parallel data (i.e., during a READ command), and vice versa (i.e., during a WRITE command). Such conversions are handled in the SERDES (serial-deserializer) hardware when appropriate.

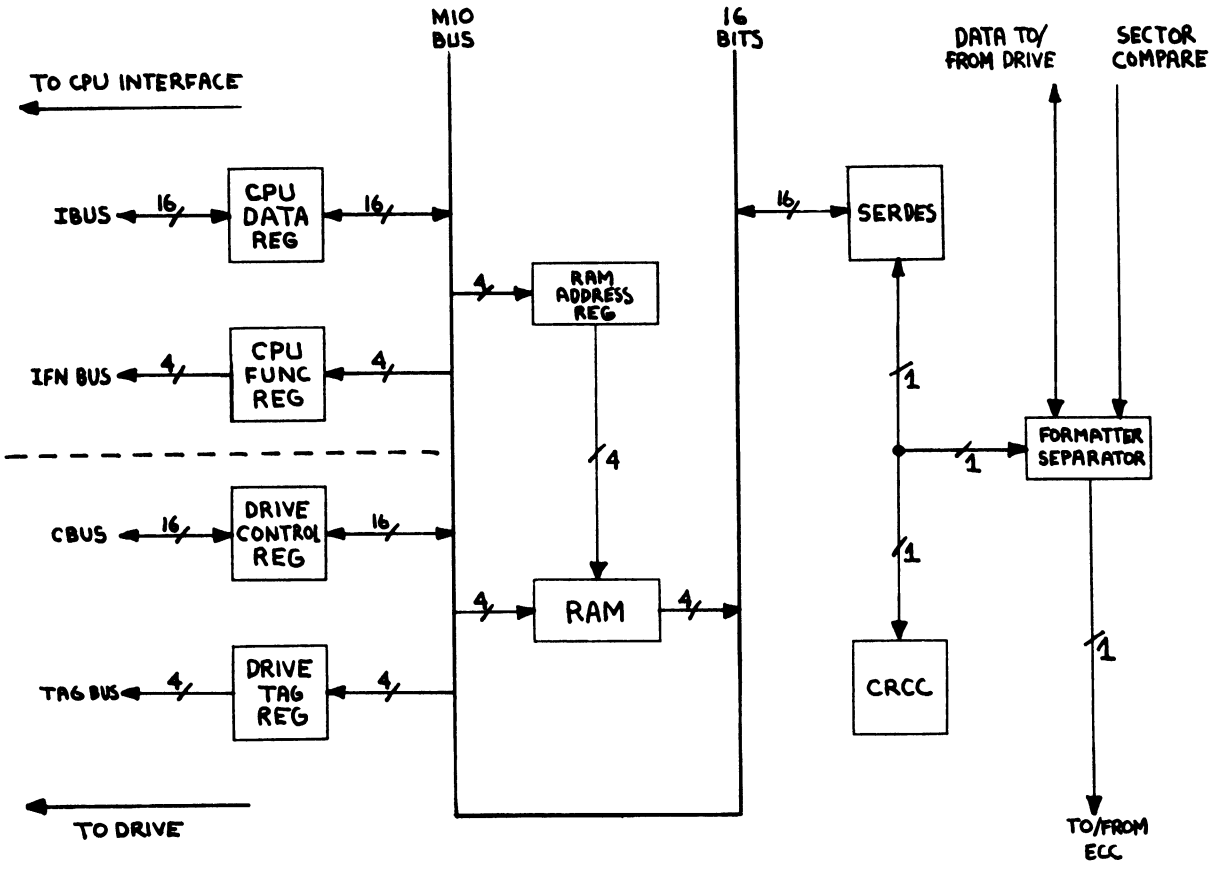


Figure 1. Device Controller Block Diagram

The SERDES hardware, as well as the bus buffers and latches, RAM and RAM address logic, and ICW (internal control word) hardware, is firmware controlled via the XADDR lines from the microcomputer board. These lines drive a MUX, enabling a specific hardware function selected by the microcode. When enabled, the selected hardware will (when appropriate) combine various external signals (i.e., LBOUT or UBOUT, INPUT, etc.) with relevant MIO bus lines to generate system flags, set up the file mask, validate data transfers, or perform other indicated tasks.

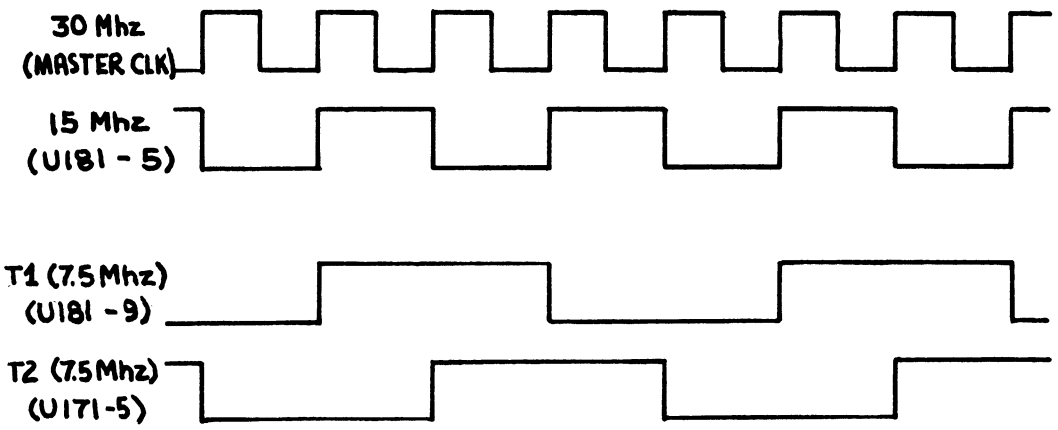
The ICW hardware, selected as described above and driven by the MIO bus, sets up various signals enabling reading, writing, and other functions (including enabling or disabling ECC shifts). The ICW (taken from the MIO bus) is latched into an ICW register until another microcode command overwrites that information.

A basic device controller block diagram is shown in Figure 1. The next section will describe these blocks and other hardware in more detail.

2.0 HARDWARE DESCRIPTION

2.1 FORMATTER/SEPARATOR CLOCK

U181 is a D-type flip-flop that functions as an R-S flip-flop with a 15 MHz input signal derived from the 30 MHz master clock on the microcomputer board (older versions take the 30 MHz directly from the master clock and incorporate U181 as a "divide-by-two" counter); this 15 MHz clock is divided by U171 and the other half of U181 into a two phase, 7.5 MHz clock incorporated by the formatter in encoding the data (the 7.5 MHz is used by the separator only as a sequence timing clock and PLL holding frequency). The relationship between the clock phases is shown below.



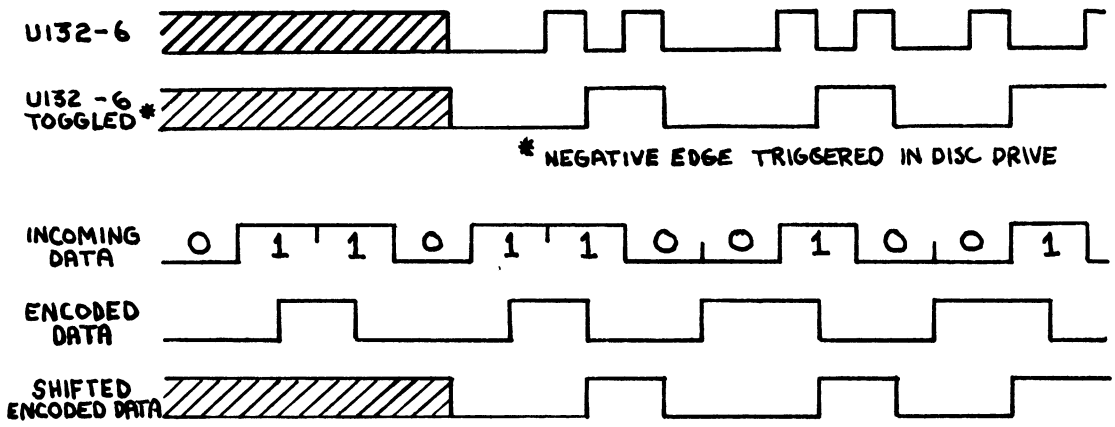
The falling edge of T2 will delineate the bit cell boundaries of the data when encoding, as will be described later in the next section.

2.2 DATA FORMATTER

With $\overline{\text{SECTOR COMPARE}}$ and WRITE both true, U189-6 provides a high WRITE GATE level (U142-2 generates a low $\overline{\text{WRITE GATE}}$ level) that enables various formatter hardware. Data to be written is provided at a 7.5 MHz system rate to U169-2, and appears at U161-4. U161 is a 74S175 quad D-type flip-flop incorporated as a 4-bit shift register through which incoming serial data is clocked to $\overline{T2}$. The various outputs are utilized in detecting "1's", "0" pairs, and the two data patterns requiring precompensation before writing onto the disc ("1101", "1011").

U143-11 goes high on detection of a "0" pair; this event is clocked into U151 by T1 (U181-9) "1's" are clocked into U141 by $\overline{T1}$. The cases of 1101 and 1011 are detected by U162-8 and U162-6 respectively; these events are clocked into U121 and U131 (both D-type flip-flops) by the output of U132-6 delayed 40 nanoseconds (more will be said about this later). U133-8 and U133-6, in conjunction with $\overline{T2}$ and T2, gate the "one" and "zero pair" signals through at the proper time for mixing by U132. U132-6 thus produces an encoded output which, when passed through the toggle flip-flop in the disc drive, gives the encoded signal to be written onto the disc.

Figure 2 outlines the relationship between the various signals for a sample data stream. (Shown below is the relationship between the encoded data and U132-6 clocking a toggle flip-flop.) The 3.5-bit cell delay of the formatter encoded data is the result of the combined delay of the shift register and event detection latches.



Although the data is successfully encoded at U132-6, precompensation must be applied to patterns 1101 and 1011 to circumvent media effects when writing (signals are delayed or advanced, respectively, to move the inner "one's" of 1101 and 1011 patterns away from the adjacent zero bit cell). This compensation is accomplished by passing the signal at U132-6 through U152, a 100-nanosecond delay line with 10-nanosecond taps. Normal data is delayed 30 nanoseconds and passed to U153 for subsequent gating. Detected 1101 patterns are gated through U153 as 40-nanosecond delayed data, while detected 1101 patterns are gated through U153 as 20-nanosecond delayed data. Normal, delayed, and advanced data appearing at U153-3, U153-8, and U153-6, respectively, are combined at U146-8 to provide the final WRITE DATA output.

U143-3 is used to disable the normal signal in passing to U153 when either a delayed or an advanced signal is to be passed through to U146. U121-9 and U131-9 (for detected 1101 events), and U131-5 and U121-5 (for detected 1011 events) are clocked by respective phases of the 40-nanosecond delayed data to set up for subsequent enabling of new delayed or advanced signals.

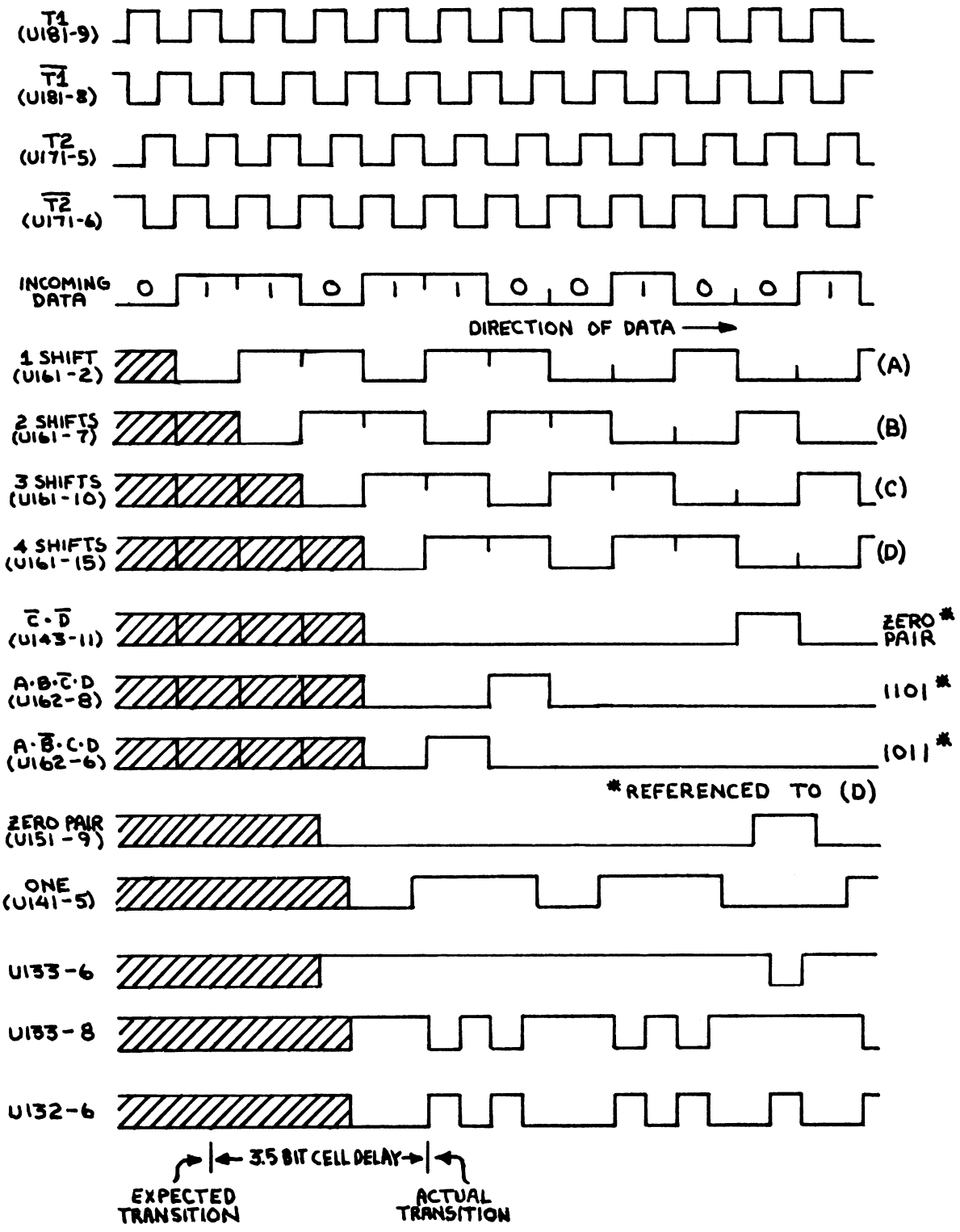
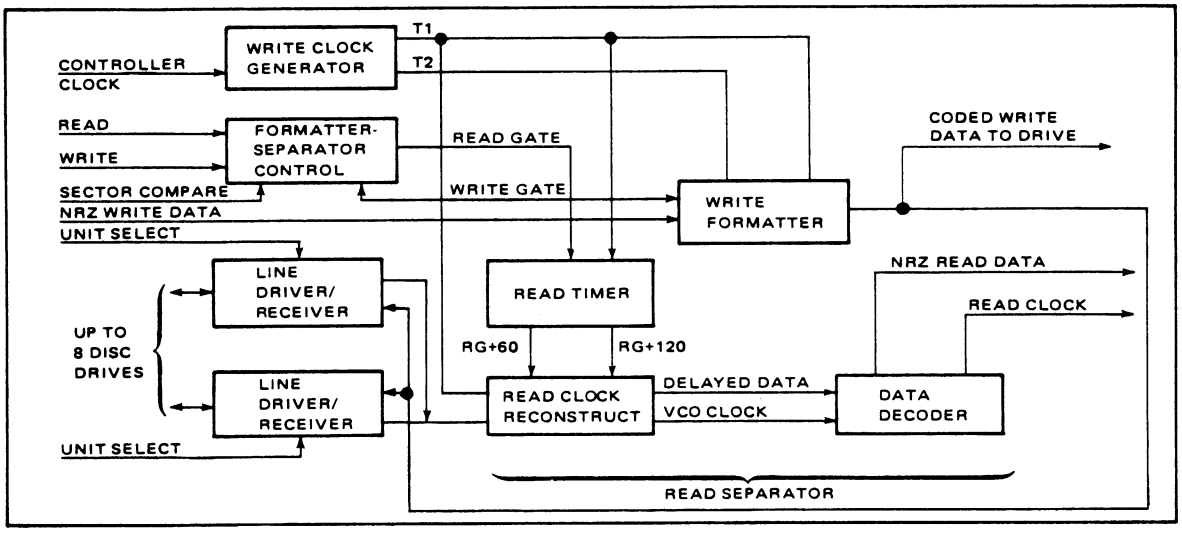

T1 (U181-9)
T1 (U181-8)
T2 (U171-5)
T2 (U171-6)
INCOMING DATA
0 1 1 1 0 1 1 1 0 0 1 1 0 1 0 1
DIRECTION OF DATA
1 SHIFT (U161-2)
(A)
2 SHIFTS (U161-7)
(B)
3 SHIFTS (U161-10)
(C)
4 SHIFTS (U161-15)
(D)
C·D (U143-11)
ZERO PAIR *
A·B·C·D (U162-8)
1101 *
A·B·C·D (U162-6)
1011 *
* REFERENCED TO (D)
ZERO PAIR (U151-9)
ONE (U141-5)
U133-6
U133-8
U132-6
3.5 BIT CELL DELAY
EXPECTED TRANSITION
ACTUAL TRANSITION

Figure 2

2.3 DATA SEPARATOR

The data separator employs both analog and digital hardware to allow the device to successfully separate data and clock signals from an incoming data stream whose rate is slightly varying. Shown below is the separator circuitry in relationship to the formatter/ separator section of the device controller.



2.3.1 SEPARATOR CONTROL AND CLOCK SELECT LOGIC

Prior to a true READ GATE (U189-11), U144-15 is held reset and U144-14 causes AND-OR output U174-6 to gate $\overline{T1}$ (U181-8) onto the DATA line of the PLL. At the same time U144-15 holds U145 reset, causing the SELECTED CLOCK line (U74-8) to carry the divided VCO output of U155-5. In this way, for non-READ operations, the PLL is held at the system storage rate of exactly 7.5 MHz.

With $\overline{\text{SECTOR COMPARE}}$ and READ both true, U189-11 provides a valid READ GATE signal, enabling U144-15 and gating T1 (U181-9) through U189-8 to be counted by U154 and U134. Note that U144-15 continues to hold U145 reset, so that the PLL will continue to lock onto the 7.5 MHz system clock until U144-15 goes high, as described below.

U154 and U134 are connected as divide-by-60 (at U134-11) and divide-by-120 (at U134-12) counters. Sixty pulses of T1 after the READ GATE goes high, U134-11 clocks U144-15 into a high state (U144-14 conversely going low); U164 and U174 consequently gate 70-nanosecond delayed data (from delay line U152) onto the DATA line (during the second 60 pulses of T1, the PLL locks up on the zero sync field read from the disc).

One hundred and twenty pulses of T1 after the READ GATE goes high, U134-12 clocks U145-11 high (U145-10 goes low), disabling counters U154 and U134 and enabling the next positive transition of the nominally 7.5 MHz VCO output (U112-5) to clock a high state through to U145-15. At this point, the clock select circuit of U175 is enabled, and U174 consequently gates U175-9 (the clock select circuit output) onto the SELECTED CLOCK line (U174-8). At this point T1 is disabled (U181-13 goes low), as it was not needed since the 60-bit count (when the DATA line was switched from T1 to 70-nanosecond delayed data). From now until $\overline{\text{SECTOR COMPARE}}$ is lost, the PLL tries to lock the VCO output onto the clock select output (which is generated by the incoming data as described below.)

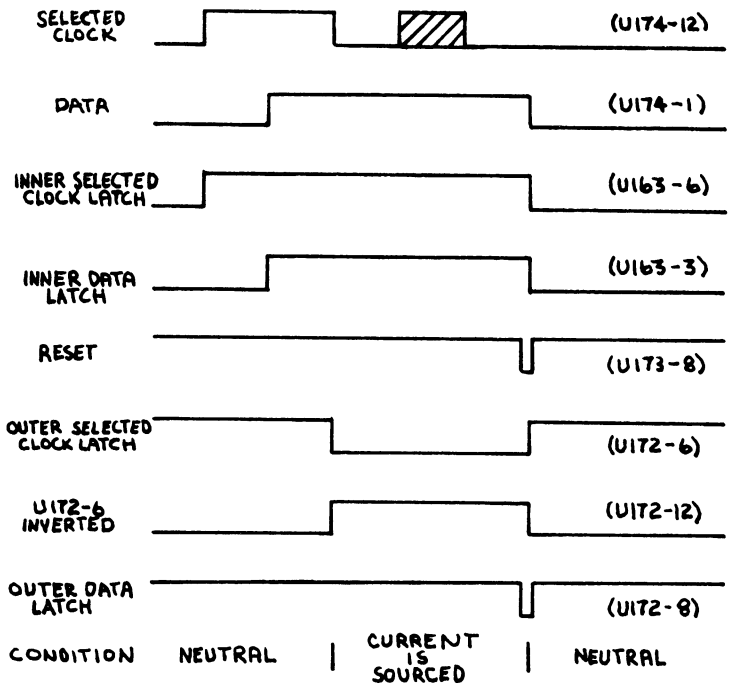
Data coming from the disc drive is generated by a bi-directional one-shot having an output pulse duration of 58 nanoseconds (nominal). After 120 pulses of T1, the clock select circuit (U175) receives the incoming data (undelayed) and generates from it a VCO dependent wavetrain coincident with the delayed incoming data. Both the delayed data and the generated wavetrain are fed to the PLL phase comparator (via the DATA line and the SELECTED CLOCK line, respectively) for use in matching the VCO frequency to that of the incoming data (this process will become evident later). Figure 3 shows the relationship between the VCO, incoming data, and clock select circuit pulsetrains.

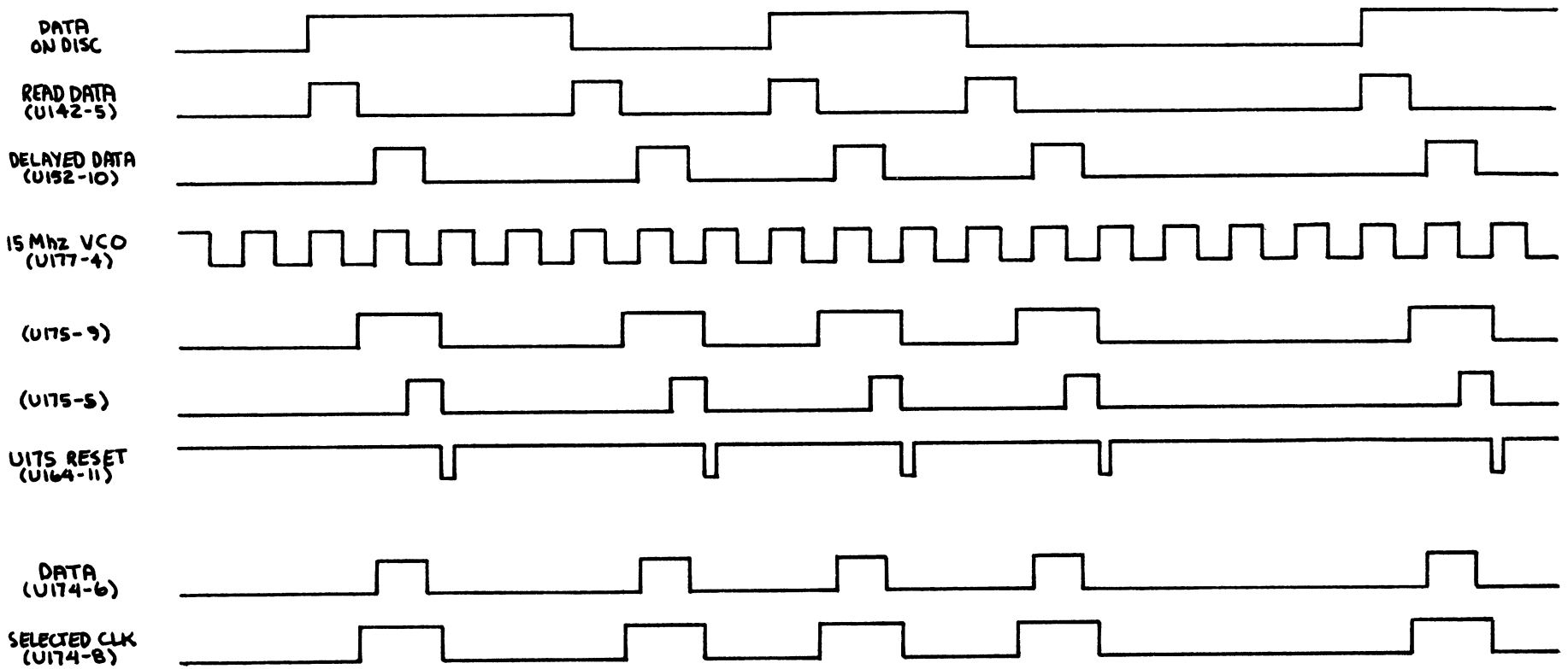
2.3.2 PHASE COMPARATOR AND VCO

Phase comparator logic (composed of U163, U172, U173, and U174) is redrawn in figure 4 to make circuit operation more apparent. U163 forms two inner latches, while U172 and U174 form two outer latches around those of U163; U173 provides a reset function for all latches (more of this will be described shortly). CR2 and CR3 make certain that current is sourced, sunk, or neither, depending on the conditions of U172-12 and U172-8 (see figure 4 insert).

When the SELECTED CLOCK provides a positive transition at U174-12, the upper inner latch (U163-6-8) becomes set; a subsequent negative transition will set the outer latch (U172-6 and U174-11). Further transitions of the SELECTED CLOCK line will produce no change in the state of either of these latches.

An analagous situation applies to DATA line latches. Note that transitions in either control lines (DATA or SELECTED CLOCK) effect only their corresponding logic. However, when all latches are set (i.e., both DATA and SELECTED CLOCK lines have encountered positive-going followed by negative-going transitions), U173 will go low, resetting all four latches to their original states. Thus the phase comparator will either source or sink current (depending on which control line received the first negative transition) between only the trailing edges of the DATA and SELECTED CLOCK signals. The timing diagram of figure 4 illustrates the various cases encountered during proper separator operation. Below is shown internal timing of the phase comparator for a specific case.





CLOCK SELECT CIRCUIT
WAVETRAINS

FIGURE 3

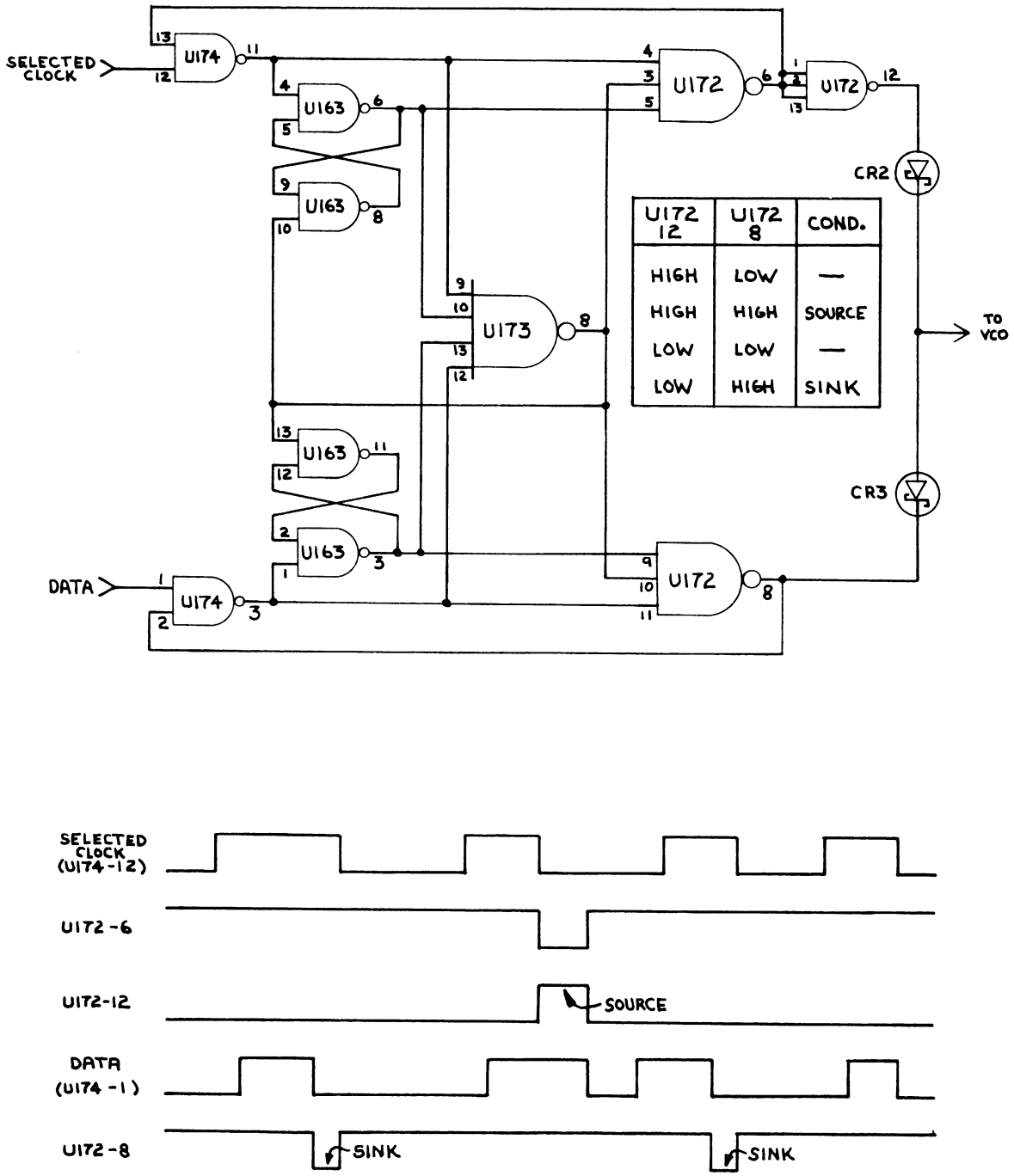


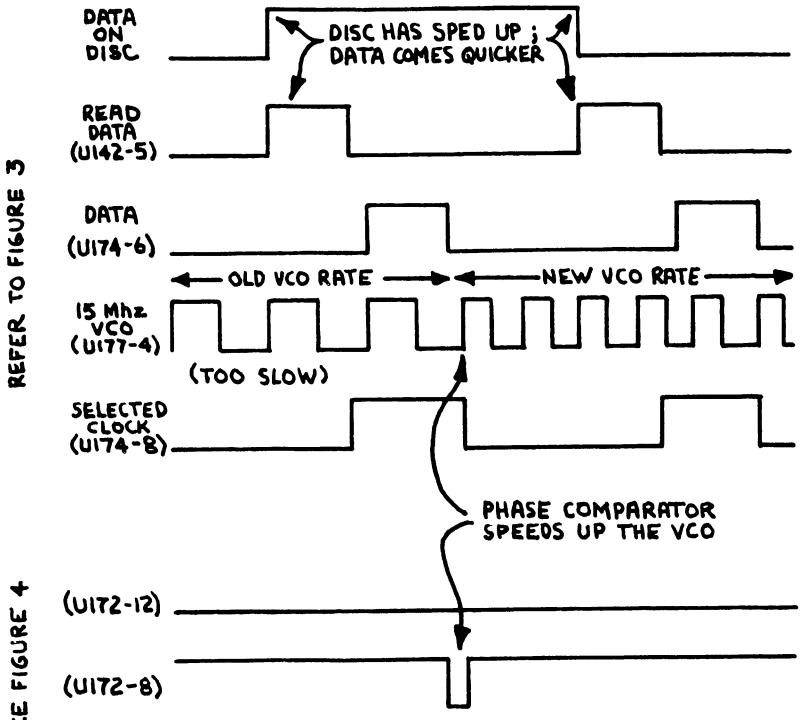
Figure 4. Phase Comparator Logic and Timing

Figure 5 shows the VCO circuit and a related block diagram. The phase comparator feeds an integrating amplifier (via the low pass filter); because the current sourced or sunk will remain relatively constant with time, the integrator is needed to convert the time that the current is sourced or sunk to a corresponding increase or decrease in voltage at the integrator's output (as described below).

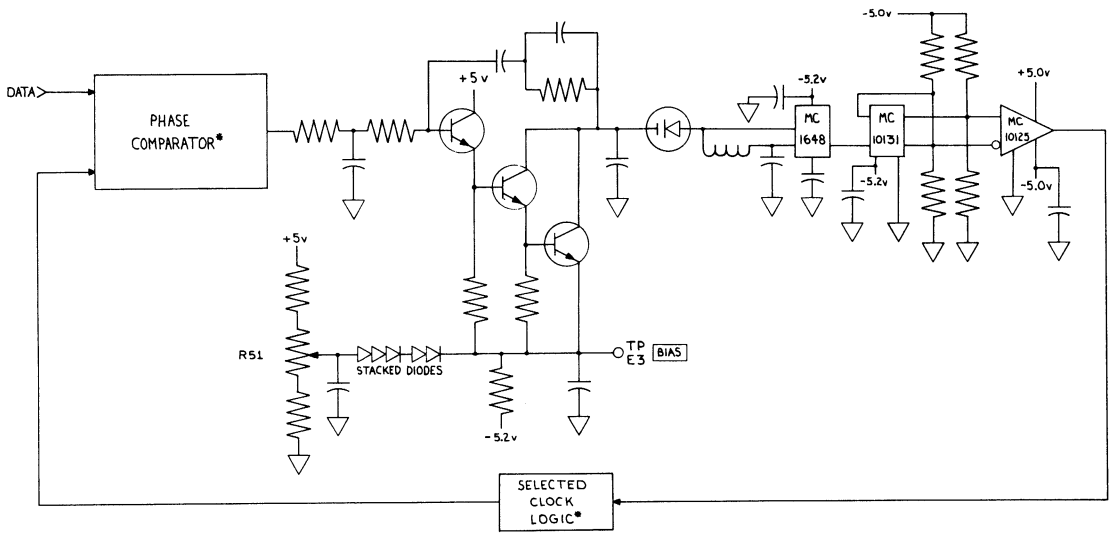
When the phase comparator sources current, C42 will change in an effort to maintain a relatively constant voltage at the base of Q4 (i.e., three base-emitter potentials above the a.c. ground of test point E3). As a result the capacitor potential will increase, causing the voltage at the collector of Darlington pair Q5 and Q6 to fall (similarly, sinking of current by the phase comparator will cause an increase in voltage at the output of the integrator by discharging C42 and thereby reducing the potential across it).

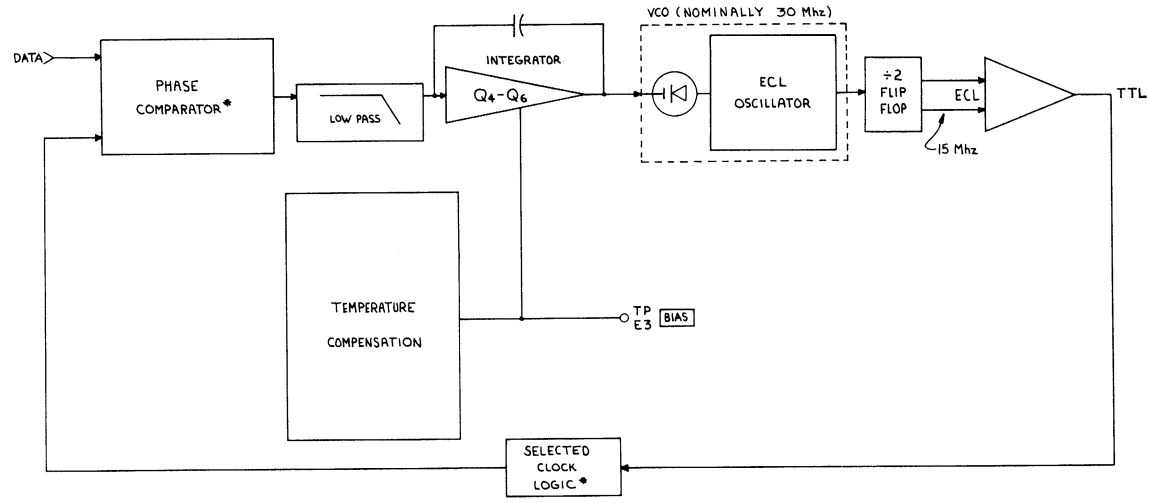
The cathode of the varicap remains basically fixed; therefore when the voltage output of the integrator falls, the reverse bias across the varicap will decrease. A smaller reverse bias will increase the capacitance associated with the varicap and, because it is in the tuned LC section of the 30 MHz ECL oscillator, will cause the VCO rate to slow down (a similar analysis will show that a rising integrator output will increase the reverse bias of the varicap, decreasing its associated capacitance and speeding up the VCO rate). Hence, the phase comparator can slow down the VCO by sourcing current (or speed it up by sinking current).

By speeding up or slowing down the VCO, the phase comparator can match the trailing edges of the DATA and SELECTED CLOCK lines, thus synchronizing the incoming data rate and the separator clock. Recall that the clock select circuit generates a falling edge that is time dependent on the VCO (refer to figure 3). Appropriately speeding up or slowing down the VCO will cause the trailing edge of the SELECTED CLOCK line (currently gating the output of the clock select circuit) to move in the direction of that of the DATA line (currently gating delayed data), as shown for various cases in figure 4. Illustrated below is a more complete timing diagram relating the clock select logic, phase comparator, and VCO for the cases of disc speed-up and slow-down.



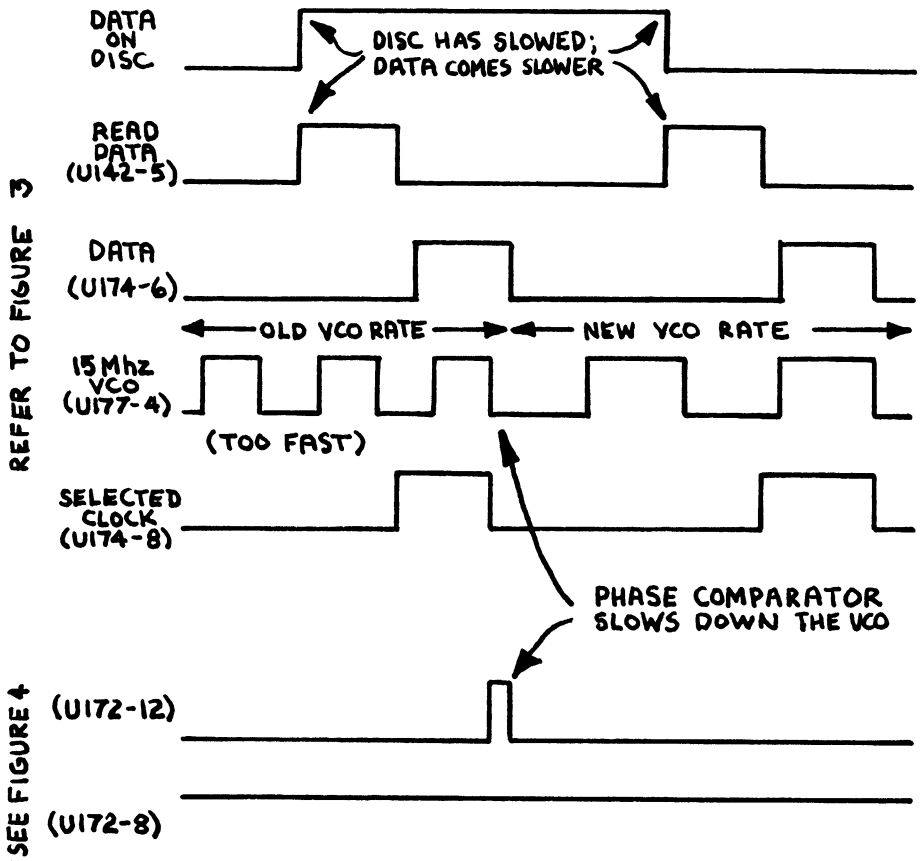
DISC SPEEDUP





* SEE PREVIOUS SECTIONS

Figure 5



DISC SLOWDOWN

A final point of interest concerning the VCO is the temperature compensation circuitry. The stacked diodes (CR5 and CR6) correct for potential differences across the transistors' base-emitter junctions due to variations in temperature. Initial adjustment is made at power on; R51 is adjusted until test point E3 is at zero potential. The phase comparator now neither sources or sinks current for constant VCO frequencies. The diodes will subsequently track with the transistors to maintain this condition throughout the specified operating temperature range of the controller.

2.3.3 FINAL DECODING LOGIC

U155-5 divides the nominally 15 MHz VCO output by two, returning a nominally 7.5 MHz VCO dependent square wave to the remaining decoder logic and to the PLL control logic. During the 60th to 120th count of T1, U155-5 is in the feedback loop of the PLL and subsequently becomes locked onto the zero sync field. U155-9 toggles 90° out of phase of U155-5 providing a "one's window" for subsequent decoding of "one's." The "one's window" at U155-9 has a duty cycle of slightly greater than 50% because advanced and retarded clocks to U155-13 are steered by U155-5 through U188. U156-5 will go high whenever the delayed data goes low during a "one's window," thus indicating a detected "one"; U157-6 will subsequently go low, latching this detected "one" into U166-5 for final outputting. With U157-6 low (U165-4 high) and U166-5 now high (U156-6 went high at the first detection of the "one"), U146-6 goes low resetting U156-5. The next trailing edge of the "one's window" will either provide clocking of another "one" into U166-5 or cause it to return to zero (depending on whether the subsequent circuitry detects a "one" or "zero" as the next bit). Final output of the decoded data occurs at U141-9; data at buffer U166-6 is clocked out through U141-9 via U155-5 (recall that U155-5 is the nominally 7.5 MHz VCO dependent output that became locked onto the zero sync

field; because transitions at "zero" bit cells occur at bit cell boundaries, it is this output that is used for final clocking of the decoded data from the separator).

U156-9 latches onto the first detected "one" after the sync field; this "one" by convention proceeds all sector information and data, as it provides the decoding logic with a way of syncing onto the relevant data that follows the introductory zero sync field. With U156-9 high, U157-3 gates the nominally 7.5 MHz VCO dependent output of U155-5 through U157-8 to provide the data clock for subsequent circuitry utilizing the decoded data.

Note that U155-9 and U166-5 are enabled by U144-15 after 60 counts of T1; after 120 bits, U156-9 is enabled, and U157-11 is gated to pass U166-6 as READ NRZ DATA.

Figure 6 provides overall timing for the final decoding logic.

2.4 SERDES, EOW, AND $\overline{\text{SYNC}}$

U98 and U128 form a 16-bit parallel load serial shift register for use by the SERDES hardware. U108, U118, and U138 form the SERDES parallel output buffer (SPOBF) while U97, U107, U117, and U127 form the parallel input buffer (SPIBF).

Serial data enters SERDES via U98-1-2; with each subsequent cycle of the DATA CLOCK (from the separator), $\overline{\text{DCLK 2}}$ will shift this data through the SERDES hardware. After 16 counts, U42 signals an EOW (end of word) and, after being delayed two bits by U52, causes U31-6 to generate a $\overline{\text{LOAD SPIBF}}$ signal that latches this 16-bit word onto the MIO bus via U97, U107, U117, and U127 (note that the SPIBF must first be enabled with an $\overline{\text{ENABLE SPIBF}}$ signal at pins 1 and 2 of each SPIBF latch). U52 incorporates the indicated two bit delay to compensate for the two bit delay of the separator hardware upon decoding read data.

Converse to the above case requiring incoming serial data from the disc to be converted to parallel data for use by the controller and CPU, parallel data to be written onto the disc must be serialized before being sent to the formatter. For this case U41-11 generates a $\overline{\text{LOAD}}$ signal at each EOW, causing the shift registers to load the current SPOBF contents (note that the SPOBF is updated by $\overline{\text{LBOUT1}}$; during clock phase T4 of appropriate lower byte instructions, the SPOBF is loaded with the current MIO bus contents). The formatter data clock (T2 now gated onto $\overline{\text{DCLK2}}$) continues to clock SERDES, with the serial data passing from U128-11 to MUX U51 (which selects from write data, ECC data, and CRCC data, depending on which requires writing onto the disc).

U51 selects appropriate data to be sent to the disc drive by monitoring the ENABLE CRCC SHIFT line. When the appropriate logic sends this line true, U51 deselects SERDES data and selects CRCC data; at the following $\overline{\text{SYNC}}$ signal U62-5 goes high, causing U51 to select serial ECC data from the ECC/ROM PCA. Note that SERIAL DATA OUT (U51-7) is sent to the formatter for encoding before being sent to the disc for subsequent writing.

The EOW latch (U72-3-11) is set either on sector overrun (generated by U62-9 when $\overline{\text{SECTOR COMPARE}}$ is lost) or when a $\overline{\text{SYNC}}$ pulse is encountered (refer to later paragraphs concerning the generation of this pulse). Resetting of the EOW flag is provided whenever an $\overline{\text{ENABLE SPOBF}}$ or $\overline{\text{ENABLE SPIBF}}$ signal becomes true; U148-11 will subsequently go low to reset the EOW latch (formed by U72-3-11). Figure 7 shows a simplified schematic representation of the EOW flag hardware, as well as the appropriate timing diagram.

$\overline{\text{SYNC}}$ hardware consists of U31, U41, U44, and U148. EOW pulses from U42-15 (on WRITE instructions), or once delayed EOW pulses from U52-5 (on READ commands), combine with the DATA CLOCK to provide the $\overline{\text{SYNC}}$ signal (U31-8). The delayed $\overline{\text{SYNC}}$ signal for READ operations is necessitated by the one bit delay of read data in the separator clock select

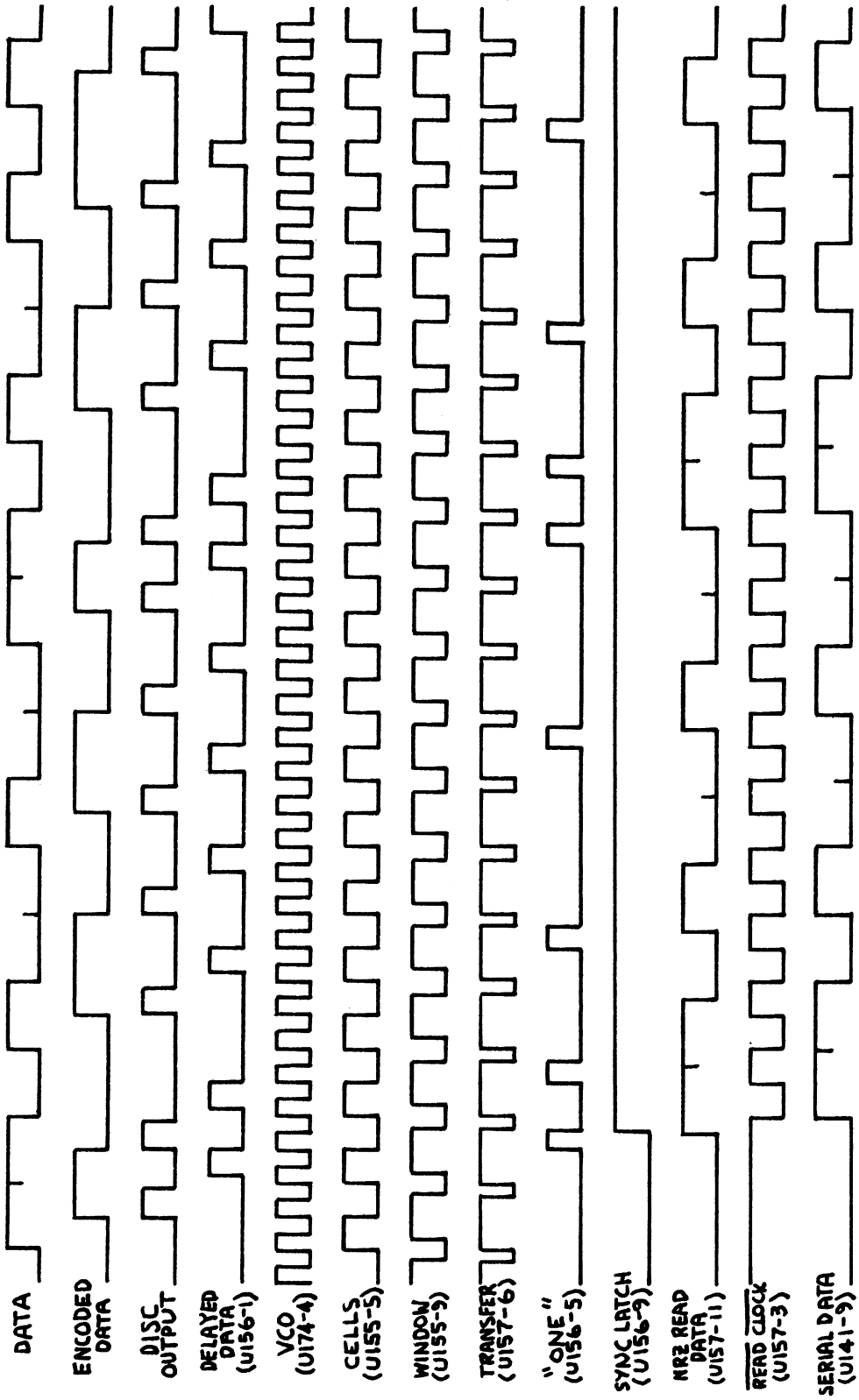


Figure 6. Decoder Logic Timing

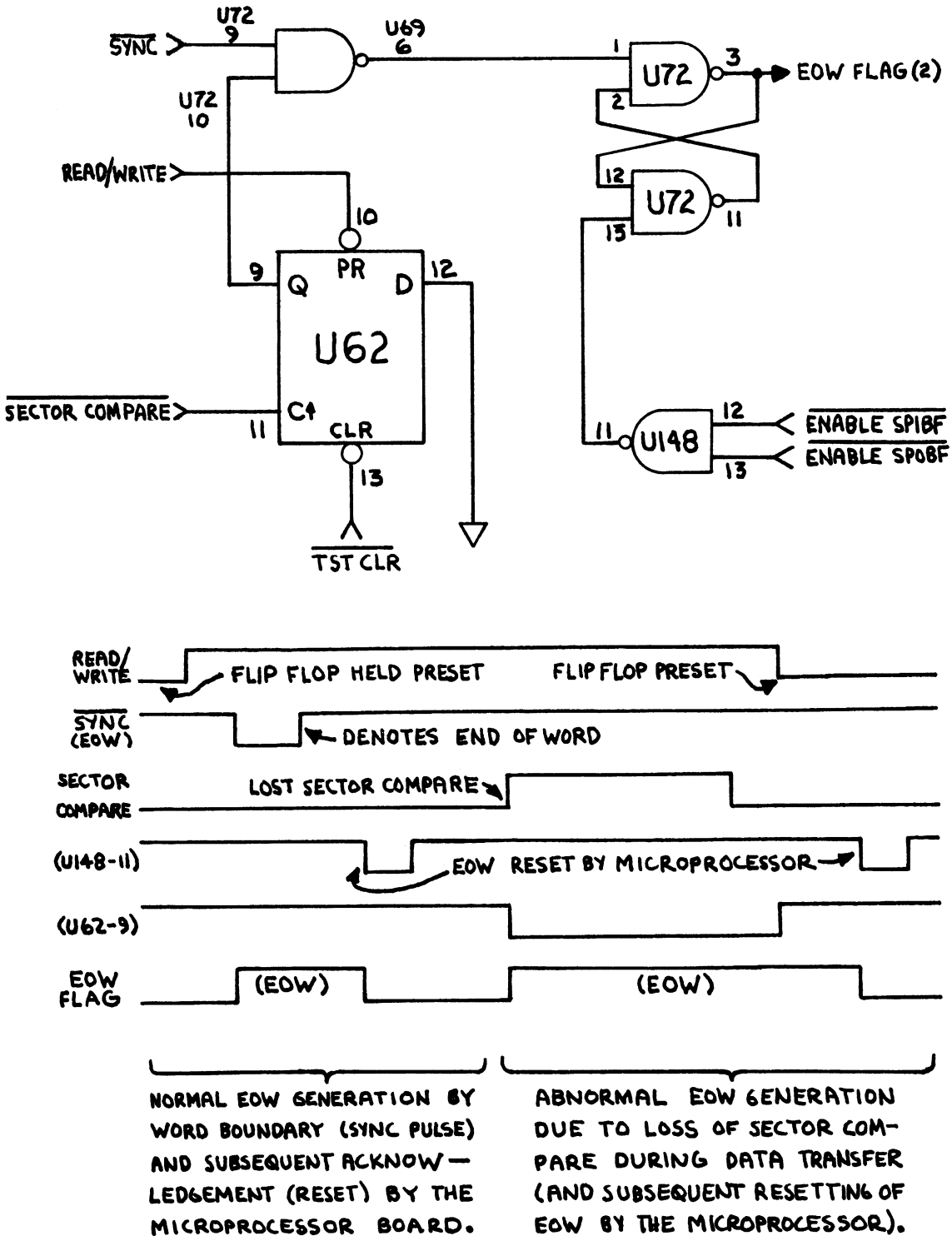


Figure 7. EOW Flag Circuit

circuit (see Section 2.3.1). (Because the $\overline{\text{SYNC}}$ pulse is incorporated into ECC timing considerations, it might prove useful to refer to Section 2.4.2 of the ECC description for further information.)

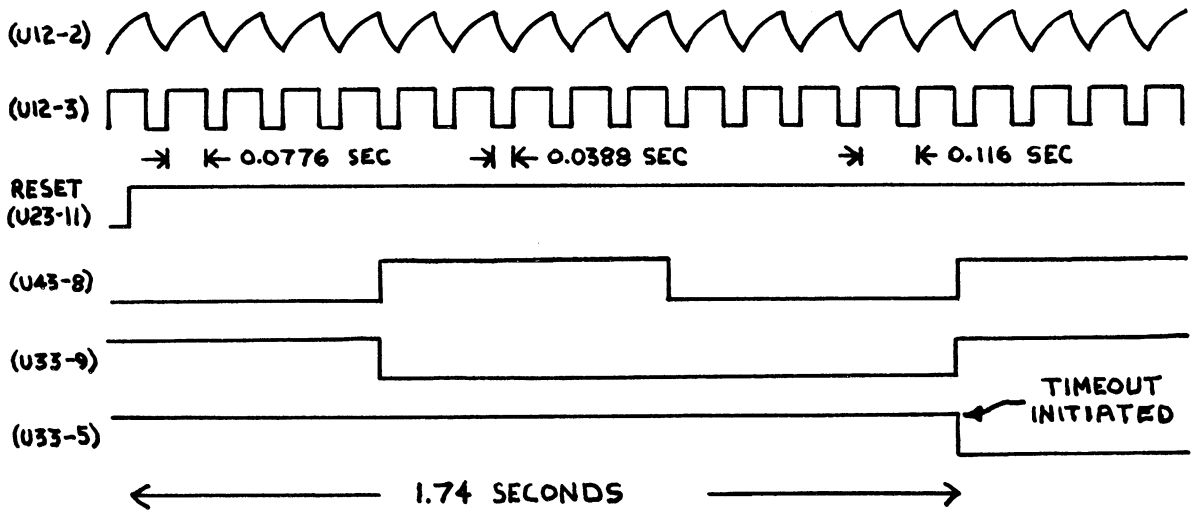
2.5 CRCC

The CRCC (cyclic redundancy code check) hardware is connected in a standard 16-bit CRCC configuration; U73-6-8-3 provide the various feedback paths around shift registers U54 and U74 to carry out the CRCC operation. U53 and U64 test the 16-bit CRCC generated polynomial for non-zero bits. If any are detected, U63-8 will go high indicating a CRCC error (the flag is tested at the appropriate time to indicate the existence of a data error).

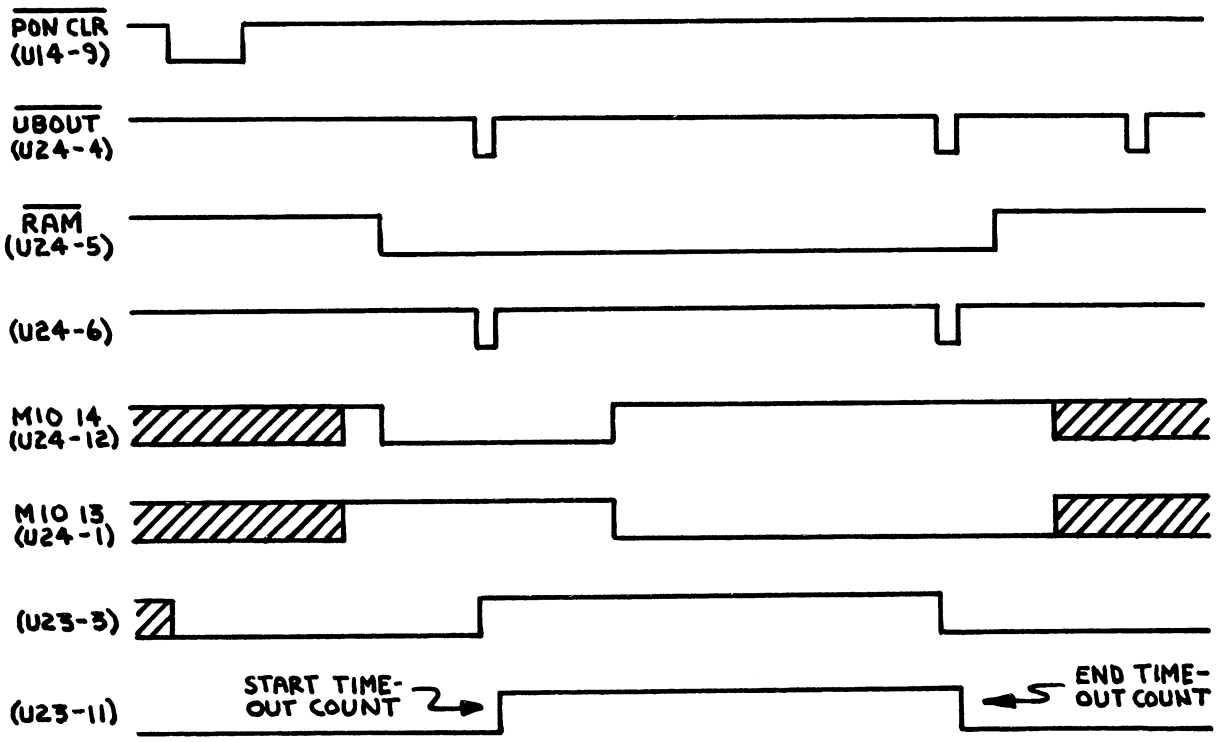
U39-6, U61-6, and U44-8 determine the source of data for CRCC processing. During a WRITE operation, serial data output from U51-7 is fed into the CRCC; otherwise serial data entering SERDES is concurrently gated into the CRCC hardware via U61.

2.6 TIMEOUT CIRCUITRY

U12 is a type 555 I.C. timer configured as an astable multi-vibrator with a period of 0.116 second (nominal) as shown below. The timer is deliberately operated at a rate greater than needed to generate the appropriate timeout period because the 555 is inherently more stable at higher frequencies (owing primarily to leakage currents in C3); as such, a divide circuit is needed to lengthen the 0.116-second period of the 555 to the desired 1.8-second (nominal) timeout period. U43 and U33 perform this dividing function.



Timeouts are initiated and reset by the gated latch formed by U14, U23, and U24. U23-3-6 form the basic latch, with U14-11-8 providing proper gating to disable timeouts during power-on clears or as a function of P2-4. U24-6-3-11 provide gating of MIO lines during appropriate intervals (i.e., during T4 of an upper byte instruction) to initiate or conclude the timing sequence. U23-11 provides the final gating of the reset/count signal to the dividing network. Below is illustrated the timing diagram for the timeout reset.



Note that U14-6 (with associated logic) will generate a true $\overline{\text{START}}$ signal whenever a timeout occurs; this signal is used by other hardware as will be described later.

2.7 RAM

The 7489 RAM (U159) functions as a scratchpad memory for the controller. In it are stored the data shown below.

BIT 3	2	1	0	WORD NO.	
X	I	I	I	15	CPU INTERFACE LAST POLLED FOR PENDING COMMAND
X	I	I	I	14	CPU INTERFACE CURRENTLY CONNECTED TO CONTROLLER
C	C	C	C	13	SECTOR COUNTER USED BY MICROPROGRAM
DRV		TYP		12	DISC DRIVE TYPE WORD
P	F	X	X	11	P = PLATTER PROTECT (READ ONLY) F = FORMAT OVERRIDE
DRV		ATN		8	PREVIOUS ATTENTION
I	I	I	H	7	CPU INTERFACE AND HOLD BIT FOR DISC DRIVES 0-7
				.	I = INTERFACE TO WHICH ASSOCIATED DRIVE (WORD NUMBER) WAS LAST CONNECTED
				.	H = SET INDICATES DISC DRIVE IS RESERVED FOR A SERIES OF OPERATIONS BY ASSOCIATED INTERFACE WORD NUMBER. NO OTHER CPU INTERFACE MAY ACCESS IT.
				.	
I	I	I	H	1	
I	I	I	H	0	NOTE THAT ONLY 8 DRIVES ARE MAPPED INTO RAM

U159 is addressed by the current contents of U158; MIO bus data (lines 0 through 3) is latched into U158 during LBOUT of a low RAMAD (RAM address) signal at U168-4 (thus 168-6 will latch MIO bus data into U158 during clock phase T4 of a proper microcode instruction). Data is written into the addressed memory only when RAM is low (thus enabling the memory enable of the RAM at U159-2) and LBOUT becomes true (thus during T4 of a lower byte instruction LBOUT goes low causing U159-3, the write enable of the RAM, to write the four lower MIO bits into the selected memory location). Note that although the write enable is set true during every lower byte instruction, data will not be written into RAM unless the memory enable is also true. Consequently data is read out of the RAM whenever U159-2 is low, but it is not gated onto the MIO bus unless U149 is enabled at U149-2-4-10-12 by U167-3 (this occurs only when the RAM has its memory enable pin true and the instuction decoding of the microprocessor PCA sets the INPUT line high).

2.8 FUNCTION DECODER

XADDR0-4 of the microprocessor PCA (available at P2-17-26-6-15-10) and the EXT SEL (P2-32) cause U59 to select one of fourteen (two being not currently assigned) internal functions of the device controller. Depending on the function selected, a particular gate of U48, U29, or U49 will be enabled to pass the appropriate control signal to the desired hardware. Described below are the various hardware implementations of these internal functions (note that the RAM, RAMAD, SPIBF ENABLE, and SPOBF ENABLE lines are generated by U59, but the corresponding hardware have already been described).

2.8.1 CONTROL BUS HARDWARE

Depending on whether CBUS information is going into or coming out of the controller, U59-5 (CBUS IN) or U59-3 (CBUS OUT) will be low. With U59-5 low, 8T26's U96, U106, U116, and U126 will be enabled in a receiver mode and consequently gate CBUS data onto the MIO bus (note that CBUS8 preceding these inverting bus drivers represents SECTOR COMPARE for other hardware logic).

Outgoing CBUS data must first be latched into U58, U68, and U78 (8T26's U96, U106, U116, and U126 are enabled as CBUS drivers by an appropriate TBUS command); with U59-3 low, U49-6 allows LBOUT to clock MIO data into these latches for subsequent loading onto the CBUS.

2.8.2 INTERFACE BUS HARDWARE

U59-4 being low enables the latching of MIO bus data into U57, U67, and U77 (this data is driven onto the MIO bus when 8T26's U56, U66, U76, and U86 are enabled in a driver configuration by interface function bus hardware). This operation will succeed only when INPUT (P2-8) is low (enabling U48-3); LBOUT passes through U48-6 (enabled by U59-4 being low) and performs the actual latching of data.

Note that U168-8 is subsequently enabled when U59-3 goes low; however INPUT (P2-8) must be high to allow U168-8 to enable 8T26's U56, U66, U76, and U86 in a receiver configuration (thus gating IBUS information onto the MIO bus). In this way, the INPUT signal can be used to decode a single IBUS EN command into an input or output operation without requiring another output of U59.

Whenever U59-4 goes low U69-2 subsequently goes high, allowing clock phase T4 (P2-3) to latch a high state at U18-9; consequently U21-6 will send IFCLK (J1-31) low, validating the individual word transfer on the IBUS. IFCLK will remain true until clock phase T4 of the next

instruction (unless that instruction sets U59-3 low), because U69-2 will go low as U59-3 goes high, thus allowing U18-9 to clock in a low state (U21-6 subsequently goes high).

It is important to remember that 8T26's U56, U66, U76, and U86 must be enabled prior to latching MIO bus data into U57, U67, and U77, as the validation strobe ($\overline{\text{IFCLK}}$) is sent only at this time.

2.8.3 INTERFACE FUNCTION BUS HARDWARE

U59-6 controls both the interface function bus and the clock offset enable. When U59-6 goes low, U29-8 allows LBOUT to latch MIO data into U47. U17 and U21 are always enabled, sending this latched data out on the four bit interface function bus ($\overline{\text{IFN0-3}}$) and the interface receiver ($\overline{\text{ENIR}}$) and interface driver ($\overline{\text{ENID}}$) enable control lines.

2.8.4 RETRY/INITIAL AND NEW TRACK FLAGS

U59-7 controls the retry-initial flag. When it is low, U29-6 passes $\overline{\text{LBOUT}}$ to allow clocking of MIO 0 data into U88-9. Similarly, U59-8 controls the new track flag; $\overline{\text{UBOUT}}$, when U59-8 is low, passes through U29-8, clocking MIO 15 data into U88-5. These flags appear at P2-22 and P2-26 respectively (representing flags 9 and 5).

2.8.5 TAG BUS HARDWARE

U59-5 will go low whenever tag bus data is to be updated. When this occurs, U49-8 passes $\overline{\text{LBOUT}}$ to latch MIO bits 0 to 3 into U27. U32 is always enabled, and consequently passes this data onto the tag bus.

Note that TBUS 3 (U27-10) will, when high, enable drivers U96, U106, U116, and U126 to pass latched MIO bus contents from U58, U68, and U78 onto the CBUS (refer back to section 2.8.1).

2.8.6 INTERNAL CONTROL WORD HARDWARE

The internal control word (ICW) generates the internal READ and WRITE commands, as well as CRCC functions, $\overline{\text{STROBE}}$ signals, etc.

When U59-10 goes low, U49-3 is gated to pass $\overline{\text{UBOUT}}$ to the ICW latches (U38 and U26-5). With a $\overline{\text{UBOUT}}$ pulse, MIO data is latched into U38 for ICW decoding, and U26 for the $\overline{\text{STROBE}}$ signal circuitry. U38 obtains the READ (MIO 8), WRITE (MIO 9), and READ/WRITE (MIO 10) signals directly. MIO bits 10 – 12 appear for processing by U16 (dual D-type flip-flop) and subsequent control of the CRCC hardware, as described below. Note that both U38 and U26-5 are cleared by a $\overline{\text{START}}$ pulse.

U16-5 and U16-9 decode certain ICW data to provide the INH CRCC CLK and EN CRCC SHIFT signals; the $\overline{\text{SYNC}}$ pulse will latch ICW data into U16-5-9 at the appropriate time during a READ or WRITE operation (by using the $\overline{\text{SYNC}}$ pulse as a clock, the circuitry can synchronize with the boundary of the last data word). Notice that a false READ/WRITE signal resets U16-5 (thus preventing any CRCC shifting). U16-9 reset and data functions are under the control of two distinct ICW bits (U38-7 and U38-5), while the data function of U16-5 is controlled by the same ICW bit as its reset (U38-10 and U38-12 both latch MIO 10 data).

$\overline{\text{STROBE}}$ hardware is controlled both by POWER ON and ICW signals (the former will be described in section 2.9). Each signal controls one of the two series transistors of U25; note that if any are "off," R39 and R40 will pull the $\overline{\text{STROBE}}$ signal to a high state (only when both are conducting will the $\overline{\text{STROBE}}$ line go true).

When the $\overline{\text{ICW CLK}}$ (U49-3) latches a "zero" (from MIO 13), U25-3 will invert it and cause its series transistor to conduct (normally the other transistor is already biased "on" by the POWER ON hardware); the $\overline{\text{STROBE}}$ line will go true unless a $\overline{\text{START}}$ pulse sets U26-5 to its normal high state (thus returning $\overline{\text{STROBE}}$ high).

2.8.7 FILE MASK AND EOC HARDWARE

The file mask and EOC (end of cylinder) flag are both selected when U59-13 goes low. A lower byte instruction will cause U29-3 to clock MIO 8 data onto U18-5 to generate the appropriate EOC flag; an upper byte instruction causes U29-11 to clock MIO bits 0, 1, 2, and 3 into U37-5, U37-9, and U26-9, respectively, thus generating the AUTSK (auto seek) flag, CYSRF (cylinder/surface) flag, SPREN (sparing enabled) flag, and seek operation's direction bit (notice that the latter is enabled onto MIO 14 only when an INPUT mode operation is in progress and the RAM is selected).

2.8.8 INTERFACE FUNCTION VALIDATION

Interface function bus commands are validated whenever U59-14 goes low. $\overline{\text{UBOUT}}$ is enabled through U48-11 to clock MIO 8 data into U36-5; this signal is subsequently inverted and driven to the interface by U21-11 (8T09 driver), which is always enabled. Note that this $\overline{\text{IFN VALID}}$ signal will remain in the current state until another instruction selects this hardware and clocks opposite data into U36-5.

2.9 POWER ON AND POWER FAIL CIRCUITRY

When controller power is applied, a POWER ON signal is sent to the controller via P1-9. Q2 will then conduct to drive the "power on" hardware and enable any subsequent $\overline{\text{STROBE}}$ signal.

Because the 13037B must be able to respond to the clear signal pulse train generated by 21XX CPU's, a digital low pass filter was incorporated in the clear circuitry. The falling edge of the clear signal at J1-2B causes U22-11 to go high. U22-4, 5 goes immediately high to cause U22-6 to generate a true $\overline{\text{PON CLR}}$ signal within the controller. When the clear signal returns high, U22-11 will return low; however, CR7 is now reverse biased, and its failure to conduct forces the RC time constant of R60 and C57. This cycle repeats throughout the clear pulse train, with a true $\overline{\text{PON CLR}}$ signal eventually winning out.

Note whenever POWER ON is true, Q1 is saturated and hence the POWER FAIL signal is false; if, however, POWER ON is lost causing the emitter of Q2 to go low, Q1 turns off to generate a true POWER FAIL signal (in effect J1-18, the POWER FAIL connection, acts as an open collector output that provides a CPU power fail signal).

2.10 DRIVE RECEIVERS AND TRANSMITTERS

Eight identical circuits are employed as interfaces between the disc and the controller. Each disc sends two differential signal lines and one select line to the device controller for subsequent processing. Figure 8 illustrates one of the eight transceiver circuits.

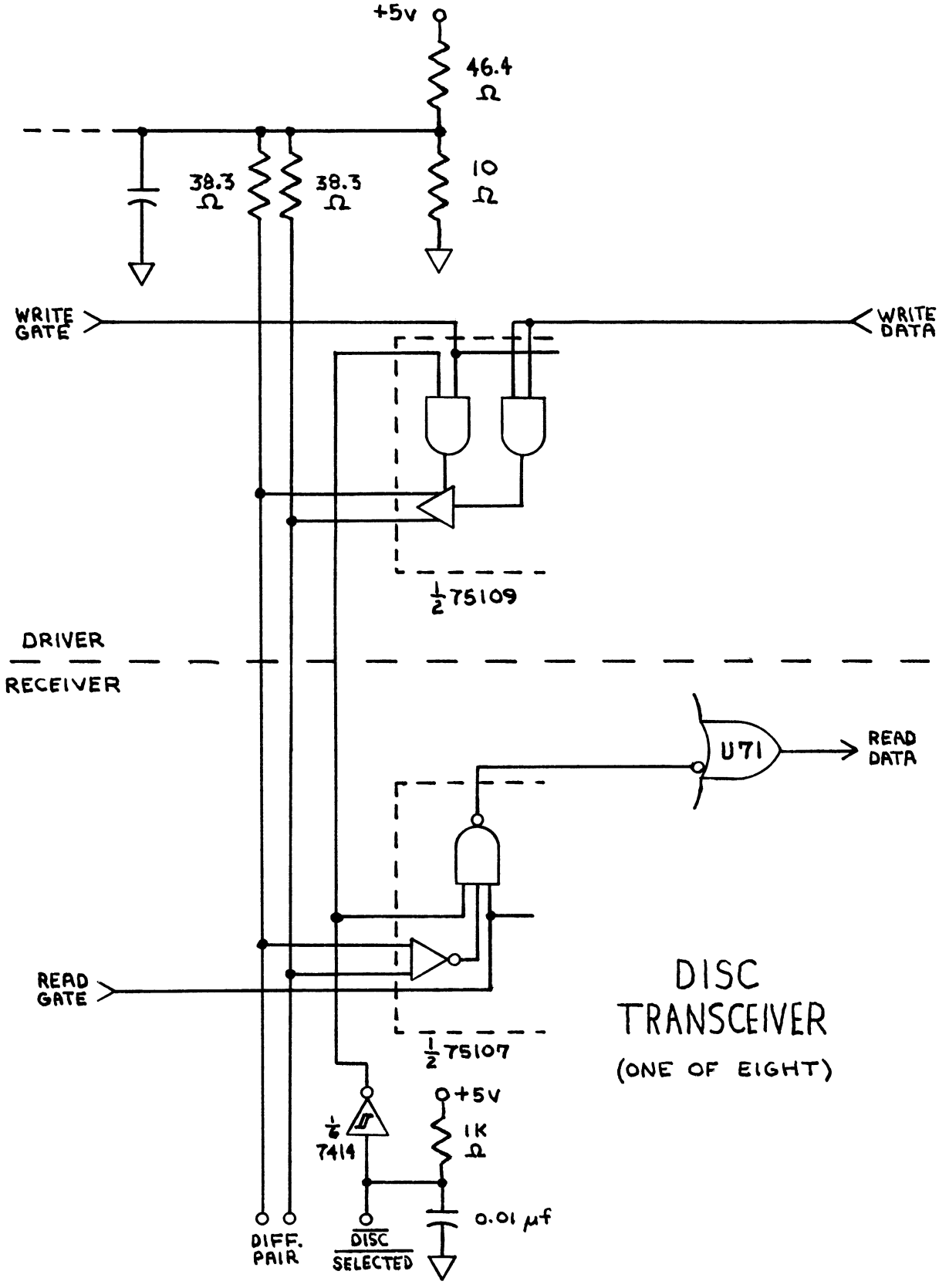

+5v
46.4 Ω
38.3 Ω
38.3 Ω
10 Ω
WRITE GATE
WRITE DATA
½ 75109
DRIVER
RECEIVER
U71
READ DATA
READ GATE
½ 75107
DISC TRANSCEIVER
(ONE OF EIGHT)
+5V
1K Ω
7414
0.01 μf
DIFF. PAIR
DISC SELECTED

Figure 8

The selected drive will send a $\overline{\text{DISC SELECTED}}$ signal to the appropriate transceiver hardware of the device controller. This signal, along with a READ GATE or WRITE GATE signal, will enable either the receiver (75107) or the driver (75109) circuit, respectively. The former will take data from the differential pair and pass it to U71; the latter will drive the differential pair with WRITE DATA. Note that because only one drive will be selected at any one time, there will be no conflict of READ DATA at U71.

2.11 FLAG BUFFER AND OVERRUN

All information between any device is sent ground true; the interface flags, therefore, need inverting and buffering for subsequent use by the controller. U19 provides both buffering and inversion of these flags, as well as the double buffering of the $\overline{\text{START}}$ signal (thus creating the $\overline{\text{CLEAR}}$ signal at P2-2).

Note that the incoming interface overrun flag (flag 7) is OR'ed with a controller-generated signal (at U35-6); during write operations the controller will generate an overrun error (independent of any overrun condition indicated by the interface flag) whenever a write is attempted without available data (unless, of course, an end-of-data condition is detected, whereupon the last word sent is continually written onto the remaining portion of the sector). U45 forms the controller overrun latch; it is enabled at U45-5 whenever the controller is in a write mode, and has as its input (set side) the output U35-11. This gate, in conjunction with U35-8, allows any $\overline{\text{IFCLK}}$ pulse (indicating a word transfer) to set the overrun latch unless U35-8 detects a DTRY (flag 1) or EOD (flag 6) condition (thus disabling the $\overline{\text{IFCLK}}$ pulse at U35-11). When the write mode is lost, the overrun latch is held reset until any subsequent write operation.

APPENDIX I

DISC DRIVE INTERFACE

CONTROLLER/DISC DRIVE INTERFACE

The Controller/Disc Interface will have the following groups of lines:

1. TAG BUS — Four lines specifying what function is to be performed.
2. CONTROL BUS — Sixteen bidirectional lines containing additional command information, drive conditions, etc.
3. COMMAND STROBE — Validates all Tag Bus functions.
4. FORMATTED DATA — A bidirectional line transmitting bit formatted data between the Controller and Disc Unit.

A schematic is shown in figure 4A-1.

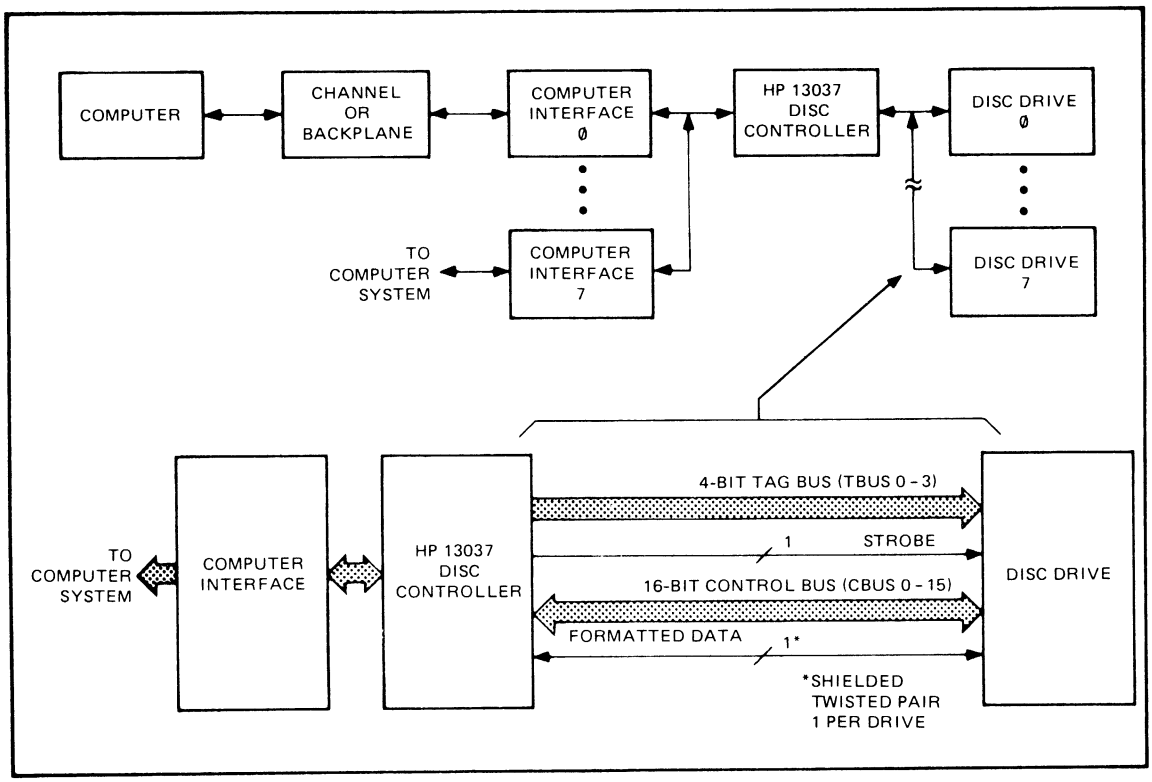


Figure 4A-1. Disc Drive Interface

TAG BUS FUNCTIONS

TAG BUS functions are summarized in table 4A-1.

The TAG BUS is a four-bit bus validated by command strobe, transmitting drive functions from the controller. All edge validations are by the leading edge of the strobe.

Table 4A-1. Tag Bus Functions Summary

TAG BUS					CONTROL BUS (Refer to Table A-2)	
CODE BIT				OPERATION		
3	2	1	0			
0	0	0	0	READ*	STATUS	INFORMATION FROM THE DISC DRIVE
0	0	0	1	WRITE*	STATUS	
0	0	1	0	REQUEST STATUS*	STATUS	
0	0	1	1	REQUEST ATTENTION	ATTENTION	
0	1	0	0	DISCONNECT		
0	1	0	1	CLEAR		
0	1	1	0	REQUEST SECTOR*		
0	1	1	1	NOT USED		
1	0	0	0	SEEK*	CYLINDER ADDRESS	INFORMATION FROM THE CONTROLLER
1	0	0	1	ADDRESS RECORD*	HEAD-SECTOR ADDRESS	
1	0	1	0	ADDRESS UNIT	UNIT ADDRESS	
1	0	1	1	RECALIBRATE*		
1	1	0	0	TRANSMIT SECTOR*	SECTOR ADDRESS	
1	1	0	1	OFFSET*	OFFSET	
1	1	1	0	CLEAR STATUS*	SELECT CLEAR	
1	1	1	1	NOT USED		

*Only selected units will respond.

Tag bus bit 3 determines whether the controller or the disc drive will place information on the control bus.

Units respond to commands only after being selected; the exceptions are Address Unit, Request Attention, Disconnect, and Clear.

READ

Select required. Drive gates status on the control bus. When this function is valid the disc drive waits for the leading edge of its internal sector compare signal, then transmits sector compare in the status word and bit-encoded data from the selected head on the formatted data lines.

WRITE

Select required. Drive gates status on the control bus. When this function is valid the disc drive waits for the leading edge of its internal sector compare signal, then transmits sector compare in the status word and accepts bit-encoded data for the selected head on the formatted data lines.

REQUEST STATUS

Select required. When this function is valid the drive gates the contents of its status register on the control bus. See table 4A-2 for status bit assignments.

REQUEST ATTENTION (REQUEST IDENTITY)

Select not required. When this function is valid the drive gates its attention bit on the line of the control bus corresponding to its unit number.

The Rotation Position Sensing (RPS) feature allows the drive to transmit this attention bit with a lookahead, requiring that the attention bit go true N sectors in advance (where N is any number between 0 and 15 inclusive). This delay is selectable on each disc drive by a strap. It should be noted that without RPS enabled, attention will remain true until disabled by the controller; with RPS, the attention is gated and remains true only during the time that the target sector (−N) passes underneath the heads.

DISCONNECT

Select not required. The disc drive responds to this function by clearing its select flip-flop. When this flip-flop is false the drive will not respond to select-required functions.

CLEAR

Select not required. The disc drive clears its attention bit, non-destructive read/write faults, plus any other appropriate bits (e.g., first status, sector address register).

REQUEST SECTOR

Select required. The disc drive transmits the contents of its "current address" counter and the head address register on the control bus.

Note: There are two current sector counters in the 7905 and 7906 drives; one is for the fixed disc and the other is for the removable disc (this is necessary as reloading a cartridge can cause sector skew between these discs).

SEEK

Select required. Control bus contains cylinder address. On the trailing edge of command strobe the drive clocks the control bus contents into its cylinder address register and initiates a seek to that address. Upon completion of the seek, the attention bit is set. If the address is illegal the drive sets seek check and attention.

ADDRESS RECORD

Select required. Control bus contains head-sector address. On the leading edge of the command strobe the drive clocks the head address field of the control bus into its head address register. On the trailing edge the sector address field is clocked from the control bus into the sector address register (note that an illegal address in either or both fields will not be latched; instead an attention and seek check will be generated by the trailing edge of the command strobe).

ADDRESS UNIT

Select not required. Control bus contains 3-bit unit number. Disc drive responds by setting its select flip-flop if the control bus contents match the disc drive address. If this bit is true it responds to all select-required commands.

RECALIBRATE

Select required. The disc drive positions its heads over cylinder 0 and clears its "current cylinder address" register.

TRANSMIT SECTOR

Select required. Control bus contains sector address. On the leading edge of command strobe, the drive clocks the sector address only into its sector address register. The head address register is not modified.

Note: This function was designed to permit flexibility in defective/alternate track operations by allowing the sector address to be changed without the head address being touched.

OFFSET

Select required. On the leading edge of command strobe, the drive clocks the control bus contents into its position offset register. The Drive Busy status bit goes true while the heads move to their offset position. See table 4A-2 for bit significance.

Note: Designed for marginal data recovery purposes, OFFSET allows 63 increments, in either direction, from the nominal track center carriage position. The physical amount for each increment depends on the characteristics of the particular drive and/or surface being used. The offset may be modified by another offset, or cancelled by a subsequent SEEK or RECALIBRATE function.

CLEAR STATUS

Select required. Control bus contains bits specifying which drive status to clear. The drive clears the appropriate status bits in response to this function. See table 4A-2 for CLEAR STATUS bit assignments.

CONTROL BUS FUNCTIONS

CONTROL BUS functions are summarized in table 4A-2.

The CONTROL BUS is a sixteen-bit bidirectional bus used to transmit control information and modifiers between the disc drive and the controller. Depending on the tag bus, the control bus contains one of the following:

ATTENTION

A line-per-drive response. Each drive sets its attention bit when it loads, unloads, completes a seek, faults, or sets seek check in response to an illegal cylinder address. The bit is cleared by a clear command or a clear status command with bit 0 on. When attention is put on the control bus, each disc unit drives only the bus line corresponding to its logical unit number.

Table 4A-2. Control Bus Functions Summary

BUS LINE	CLEAR STATUS	OFFSET	CYLINDER	HEAD-SECTOR	UNIT	STATUS	ATTENTION
0	Attention	1	1	Sector 1	1	Drive Busy	Unit 0
1	First Status	2	2	2	2	Drive Ready	1
2		4	4	4	4	Seek Check	2
3		8	8	8		First Status	3
4		16	16	16		Drive Fault	4
5		32	32	32		Format	5
6			64			Protected	6
7		Sign	128			Attention	7
8			256	Head 1		Sector Compare	
9			512	2		Drive Type 1	
10				4		Drive Type 2	
11				8		Drive Type 4	
12						Drive Type 8	
13							
14							
15							

UNIT

Valid unit numbers are 0 to 7.

CYLINDER

Valid cylinder numbers depend on drive type.

HEAD-SECTOR

Valid head and sector numbers depend on drive type.

OFFSET

Valid offset numbers are 0 to 63, positive or negative. Negative offsets are in sign-magnitude form. +0 = −0.

CLEAR STATUS

Bit 0 on specifies that the attention bit should be cleared; bit 1 on specifies that the first status bit be cleared.

STATUS

Contains the current drive status:

Drive Busy — True if the heads are not positioned and settled over a valid track.

Drive Ready — True if the heads are over the disc (and, for the 7906, the temperature compensation requirements have been satisfied).

Seek Check — Set if the controller transmits an invalid cylinder, head, or sector address, or if a seek is attempted while the drive is still executing a previous seek command. Cleared when a valid operation is performed to correct the error. Any attention bit can then be reset by the controller.

Note: The bounds checking is done by the disc drive.

First Status — Set when drive first loads the heads on the disc, i.e., by going from a "not ready" to a "ready" condition. Cleared by a clear status command with bit 1 set.

Note: This is designed to provide separate status for the controller for unexpected events.

Drive Fault — Set when the drive detects either a read/write or servo fault; neither can be caused by the controller under normal circumstances. Non-destructive read/write faults can be cleared by a clear command; servo faults and destructive read/write faults require operator intervention.

Format — Transmits the condition of the format switch located at the front of the drive. This switch may be set to the "format" position to allow track initialization, or the controller designer may use it for any other system design purpose.

Protected — Transmits the condition of the platter protect switch located on the front of the disc drive corresponding to the currently addressed head.

Attention — The condition of the attention bit.

Sector Compare — True while a read or write command is true following the coincidence of a leading edge of the drive's internal signal comparing the sector address register and the "current sector address" counter. Cleared when the command is removed or when the disc drive determines that it is not safe to continue the transfer because the sector has passed.

Drive Type — Four encoded bits which allow controller to determine number of heads and number of sectors per track.

COMMAND STROBE LINE

Normally the leading edge validates all functions on the tag bus. The function is valid only if this line is true. The trailing edge is used when the drive is validating the contents of the CBUS.

FORMATTED DATA LINES

A differential pair used to transmit serial bit formatted data to and from the disc drive. A separate pair is used for each disc drive. Each differential pair has a shield that becomes grounded when its respective drive is selected. This shield is used to enable its corresponding data port on the 13037.

DRIVERS/RECEIVERS

All lines are ground true. FORMATTED DATA uses differential pairs. Termination of all interface lines will be provided at the disc drive located physically farthest down the cable from the controller. All signals are transmitted on twisted pairs. With the exception of FORMATTED DATA, one side of each twisted pair is terminated to Logic Ground at the DISC UNIT and at the CONTROLLER.

SERIAL BIT ENCODED DATA

Coding Scheme — Delay modulation, self clocking with one transition per bit cell. A data pulse is written for each "1" data bit. The data pulse occurs in the middle of the bit cell.

A clock pulse is written only between two consecutive "0" data bits. This clock bit occurs at the cell boundary between the "0" data bits.

Write Clock Frequency — The Write Clock Frequency is 7.50 megahertz ±0.01%.

Bit Cell Time — 133.33 nanoseconds nominal.

Read Data Pulse Width — 50 nanoseconds nominal.

Sector Timing — The leading edge of any sector pulse relative to its nominal location shall not vary more than ±5 microseconds (this includes sector pulse circuitry; deviation between heads on the same disc will be less than 4 microseconds).

APPENDIX II

COMPUTER INTERFACE

SIGNAL LINES:

Clear

Hard clear on controller logic. May be generated by CPU power up/down, manual preset button, or programmed signal not otherwise interpreted by the controller. May be disabled on some interfaces. (This signal is not mandatory.)

Note: Single pulse hardware Clear signals must remain true for at least 500 μs; digital low-pass filtering of this signal within the 13037 will respond to pulse trains of sufficient duration with periods less than this 500 μs criterion.

Data Bus (IBUS 0-15)

Sixteen bit bus used to transmit commands, data, status, and addresses between the controller and interface in both directions.

ENID (Enable Interface Drivers)

Interface must be selected. Enables data bus drivers on interface.

ENIR (Enable Interface Receivers)

Enables data bus receivers on the interface.

Function Bus (IFN 0-3)

Four bit bus validated by IFVLD. Transmits interface functions from the controller.

IFCLK (Interface Clock)

Validates word transfers on the data bus while IFIN, IFGTC, IFOUT, or WRTIO function is valid. Trailing edge must be used. When IFOUT is true, controller reads data at the leading edge.

IFVLD (Interface Function Valid)

Validates function on the function bus. The function is valid only if this line is true. If an edge is used, it must be the trailing edge.

Flags from Interface

Flags are gated out by an interface whose select bit is set.

CMRDY (Command Ready)

True if a command is present in the command register. Cleared by IFGTC.

DTRDY (Data Ready)

True if data is present in the data buffer. Cleared when the data buffer is emptied.

EOD (End of Data)

True if the channel has completed a block transfer and the data buffer is empty. Not set during data chaining until the entire transfer is complete. Cleared when next command is transferred from the CPU to the interface. Note that while completing a sector transfer, the controller may continue to fetch from the data buffer even though EOD is set; therefore, the next command may not be preloaded into that data buffer.

OVRUN (Read Overrun)

Data buffer was full and controller tried to overwrite, or data buffer was empty and CPU attempted to fetch. Setting is prohibited by EOD, and is cleared when the next command is sent to the interface.

Note: Write Overrun is detected by the controller whenever the data buffer is empty and the controller attempts to fetch (unless EOD is also set).

XFRNG (Transfer No Good)

False unless the interface or channel detects an error condition (other than overrun) which prohibits transfer of data. Cleared by STINT.

INTOK (Interrupt OK)

True if the interface may be interrupted due to a disc drive attention request. Other interrupt conditions do not examine this flag.

Function Bus Orders

Orders listed are validated by IFVLD. Any edge validations are by trailing edge.

BUSY

Interface must be selected. ENIR is true. Clocks bit 0 of the data bus into a bit on the interface which, when bit 0 is high, signifies that the controller has accepted a command and is executing it, and the requested disc drive (if any) is available. (Bit 0 low – not busy.) Use IFVLD as clock. (This function is not mandatory.)

DSCIF (Disconnect Interface)

All interfaces respond. Used to clear interface select bit. When the select bit is false, the interface will not respond to select-required commands, or gate its flags or data bus out to the controller. Use IFVLD as clock. (This function is mandatory only for multiple-CPU subsystems.)

DVEND (Device End)

Interface must be selected. Transmitted by the controller to terminate an operation it considers retryable (correctable or uncorrectable data error on read operations, overrun on any data operation). May be used to increment a retry counter, must signal that the current controller command has completed. Use IFVLD as clock.

OVRUN (Read Overrun)

Data buffer was full and controller tried to overwrite, or data buffer was empty and CPU attempted to fetch. Setting is prohibited by EOD, and is cleared when the next command is sent to the interface.

Note: Write Overrun is detected by the controller whenever the data buffer is empty and the controller attempts to fetch (unless EOD is also set).

XFRNG (Transfer No Good)

False unless the interface or channel detects an error condition (other than overrun) which prohibits transfer of data. Cleared by STINT.

INTOK (Interrupt OK)

True if the interface may be interrupted due to a disc drive attention request. Other interrupt conditions do not examine this flag.

Function Bus Orders

Orders listed are validated by IFVLD. Any edge validations are by trailing edge.

BUSY

Interface must be selected. ENIR is true. Clocks bit 0 of the data bus into a bit on the interface which, when bit 0 is high, signifies that the controller has accepted a command and is executing it, and the requested disc drive (if any) is available. (Bit 0 low – not busy.) Use IFVLD as clock. (This function is not mandatory.)

DSCIF (Disconnect Interface)

All interfaces respond. Used to clear interface select bit. When the select bit is false, the interface will not respond to select-required commands, or gate its flags or data bus out to the controller. Use IFVLD as clock. (This function is mandatory only for multiple-CPU subsystems.)

DVEND (Device End)

Interface must be selected. Transmitted by the controller to terminate an operation it considers retryable (correctable or uncorrectable data error on read operations, overrun on any data operation). May be used to increment a retry counter, must signal that the current controller command has completed. Use IFVLD as clock.

IFIN (Data In, Controller to Interface)

Interface must be selected. ENIR is true. Enables transfers into the CPU from the controller. While IFIN is true, the trailing edge of IFCLK validates individual word transfers. This function is used for all transfers into the CPU.

IFGTC (Get Command from Interface)

Interface must be selected. ENID is true. Used to gate the contents of the command register on the interface onto the data bus. Clears CMRDY flag.

IFIN (Data In, Controller to Interface)

Interface must be selected. ENIR is true. Enables transfers into the CPU from the controller. While IFIN is true, the trailing edge of IFCLK validates individual word transfers. This function is used for all transfers into the CPU.

IFGTC (Get Command from Interface)

Interface must be selected. ENID is true. Used to gate the contents of the command register on the interface onto the data bus. Clears CMRDY flag.

IFOUT (Data Out, Interface to Controller)

Interface must be selected. ENID is true. Gates the interface data buffer onto the data bus. Data is accepted by the controller at the leading edge of IFCLK. Transfer validation is indicated by the trailing edge of IFCLK. Used in all word fetches to the controller except command fetch which uses IFGTC.

IFPRF (Pre-Fetch Command from Interface)

Interface must be selected. ENID is true. Gates the contents of the command register of the interface onto the data bus (as in IFGTC) but does not clear the CMRDY flag or otherwise alter the environment on the interface. Thus the same command can be obtained any number of times with this order. IFGTC should be used when the command is being obtained for the last time.

RQSRV (Request Service)

Interface must be selected. Sets service request explicitly at the end of any block transfer (in or out) involving more than the command word (e.g., SEEK, READ, STATUS, etc.). While a block transfer is in progress, the interface may not fetch the next command unless it goes to a buffer other than the data buffer. RQSRV signals the interface that the controller has finished with its block transfer and it is all right to fetch the next command. Use IFVLD as clock. (This function is not mandatory.)

SELIF (Select Interface)

Interface need not be selected. ENIR is true. A three-bit address is present on the lower three bits of the data bus. If it matches the jumpered address on the interface, the interface uses IFVLD to set its select bit. When the select bit is true the interface responds to the select-required commands, responds to ENID, and gates its flags to the controller. (Mandatory only for multiple-CPU subsystems. Otherwise the interface may have its select bit wired true.)

SRTRY (Set Retry Counter)

Interface must be selected. ENIR is true. One byte of mode control data is present on the lower byte (8 bits) of the data bus. The lower 4 bits of this byte is interpreted by the controller and should be ignored by the interface. The upper 4 bits of this lower byte is for use by the interface (e.g., as a retry counter setting). This byte is an image of the lower byte passed in a Set File Mask controller instruction. Use IFVLD as clock. (This function is not mandatory.)

STDFL (Set Data Flag)

Interface must be selected. Sets a bit on the interface:

To request direct transfer of a single word (SEEK, ADDRESS RECORD, STATUS, REQUEST SYNDROME, VERIFY, READ WITH OFFSET, REQUEST SECTOR, LOAD TIO REGISTER).

When a data operation is complete (COLD LOAD, READ, READ FULL SECTOR, VERIFY, WRITE, WRITE FULL SECTOR, INITIALIZE, READ WITH OFFSET, READ WITHOUT VERIFY).

To signal completion of a single-transaction command (SET FILE MASK, WAKEUP, CLEAR). May be used to set interrupt request. Use IFVLD as clock. (This function is not mandatory.)

STINT (Set Interrupt)

Interface must be selected. Sets interrupt request bit on the interface when the controller discovers an error condition, or when a drive is requesting attention.

WRTIO

Interface must be selected. ENIR is true. Data bus (full sixteen bits) contains controller status to be clocked into a CPU-testable status register. Use IFCLK trailing edge as clock. (This function is not mandatory.)

Function Bus Summary

FUNCTION BUS	MNEMONIC	FUNCTION DESCRIPTION	DATA BUS CONTENTS
00	STINT	Interrupt Request	
01	STDFL	Information Request	
02	IFPRF	Pre-fetch Command	Command Word
03	SELIF	Select Interface	Interface Address
04	SRTRY	Set Retry Counter	Retry Count
05	DVEND	Device End	
06		(Reserved)	
07	RQSRV	Request Service	
10		(Reserved)	
11	DSCIF	Disconnect Interface	
12	WRTIO	Write TIO	Status-1
13	IFOUT	Data Out (Write to Disc)	Data
14	IFIN	Data In (Read from Disc)	Data
15	IFGTC	Get Command	Command Word
16	BUSY	Set/Clear Controller Busy Bit	Busy Status
17		(Reserved)	

INTERRUPTS

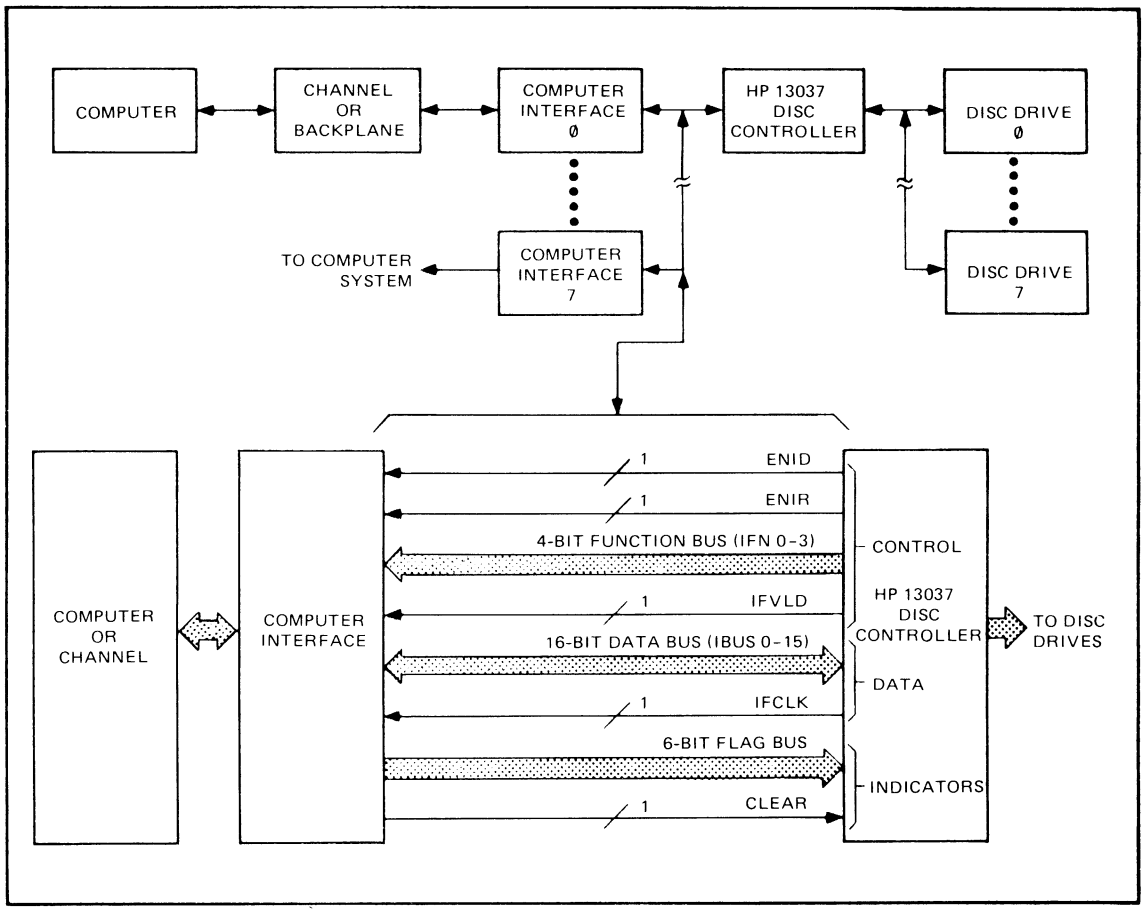
The controller generally will request an interrupt (i.e., transmit STINT or DVEND) whenever the S1 field of the Status-1 word becomes non-zero (when the S1 field indicates a drive attention status, the INTOK flag must also be true for STINT to be transmitted). This includes driver status changes (access ready error), incorrect conditions encountered during an operation (Status-2 error, address miscompare, defective or spare track access error, etc.) or program error (end of logical cylinder, illegal opcode). Additionally, at the end of a data transfer, the controller transmits STDFL to the interface, which may cause an interrupt.

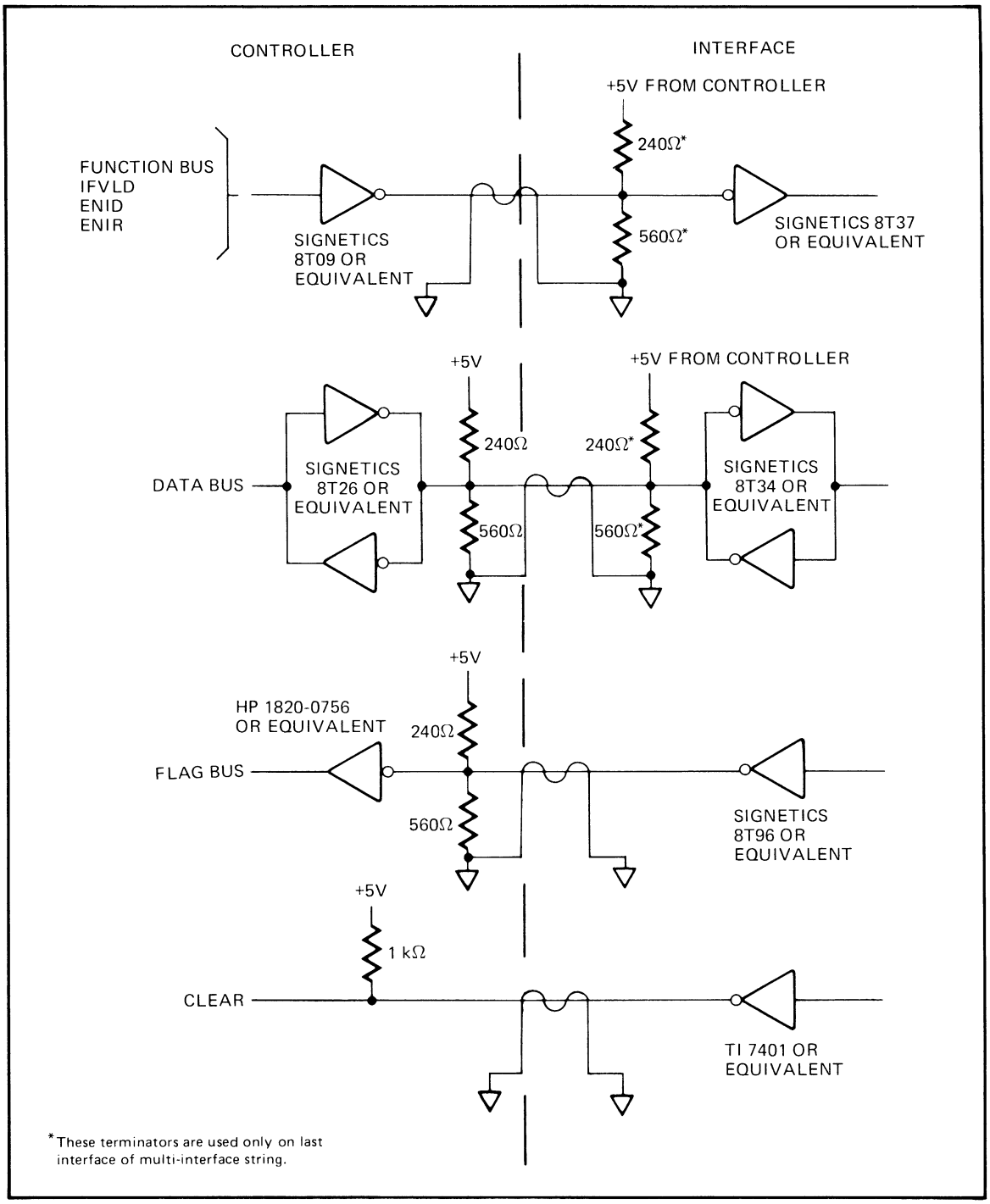
On any interrupt or command completion the CPU should examine the TIO register (if used) or perform a status operation to determine the cause and, if applicable, the unit number.

Overrun, Possibly Correctable Data Error, and Uncorrectable Data Error are considered retryable, so rather than transmit STINT, the controller transmits DVEND to the interface and increments the retry counter (if used). However, if the retry counter rolls over, the interface will interrupt the CPU. If the S1 field indicates that a Possibly Correctable Data Error has occurred, the CPU may attempt to correct this error by issuing a REQUEST SYNDROME command to the controller.

Note that interrupts generated for non-zero S1 fields of the Status-1 word will cause the controller to clear any attention bit from the drive before accepting any subsequent commands.

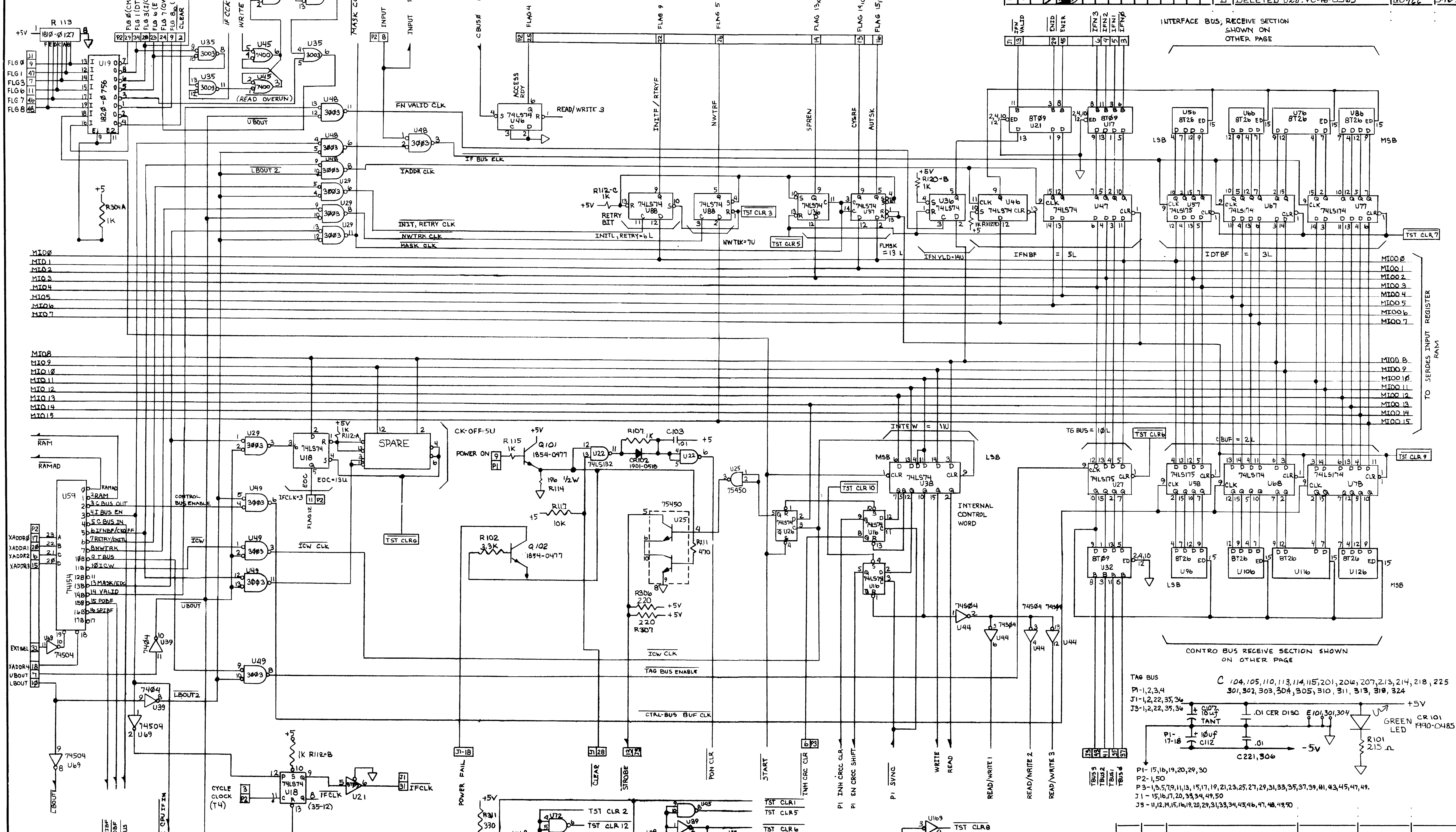
Controller Interface




CONTROLLER
INTERFACE
+5V FROM CONTROLLER
240Ω*
FUNCTION BUS
IFVLD
ENID
ENIR
SIGNETICS
8T09 OR
EQUIVALENT
560Ω*
SIGNETICS 8T37
OR EQUIVALENT
+5V
+5V FROM CONTROLLER
240Ω
240Ω*
DATA BUS
SIGNETICS
8T26 OR
EQUIVALENT
560Ω
560Ω*
SIGNETICS
8T34 OR
EQUIVALENT
+5V
HP 1820-0756
OR EQUIVALENT
240Ω
FLAG BUS
560Ω
SIGNETICS
8T96 OR
EQUIVALENT
+5V
1 kΩ
CLEAR
TI 7401 OR
EQUIVALENT
*These terminators are used only on last
interface of multi-interface string.

D-13037 -60028 -51

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED PC48-0457	ND/EC	4-24-78
B	DELETED U28. PC48-0565	LO./EC	5-16-78



ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.

TOLERANCES .XX ± .02 .XXX ± .005

SEE CORP. STD. 608

TITLE: DEVICE CONTROLLER MIO BUS-OUTPUT SECTION SCHEMATIC

HEWLETT PACKARD

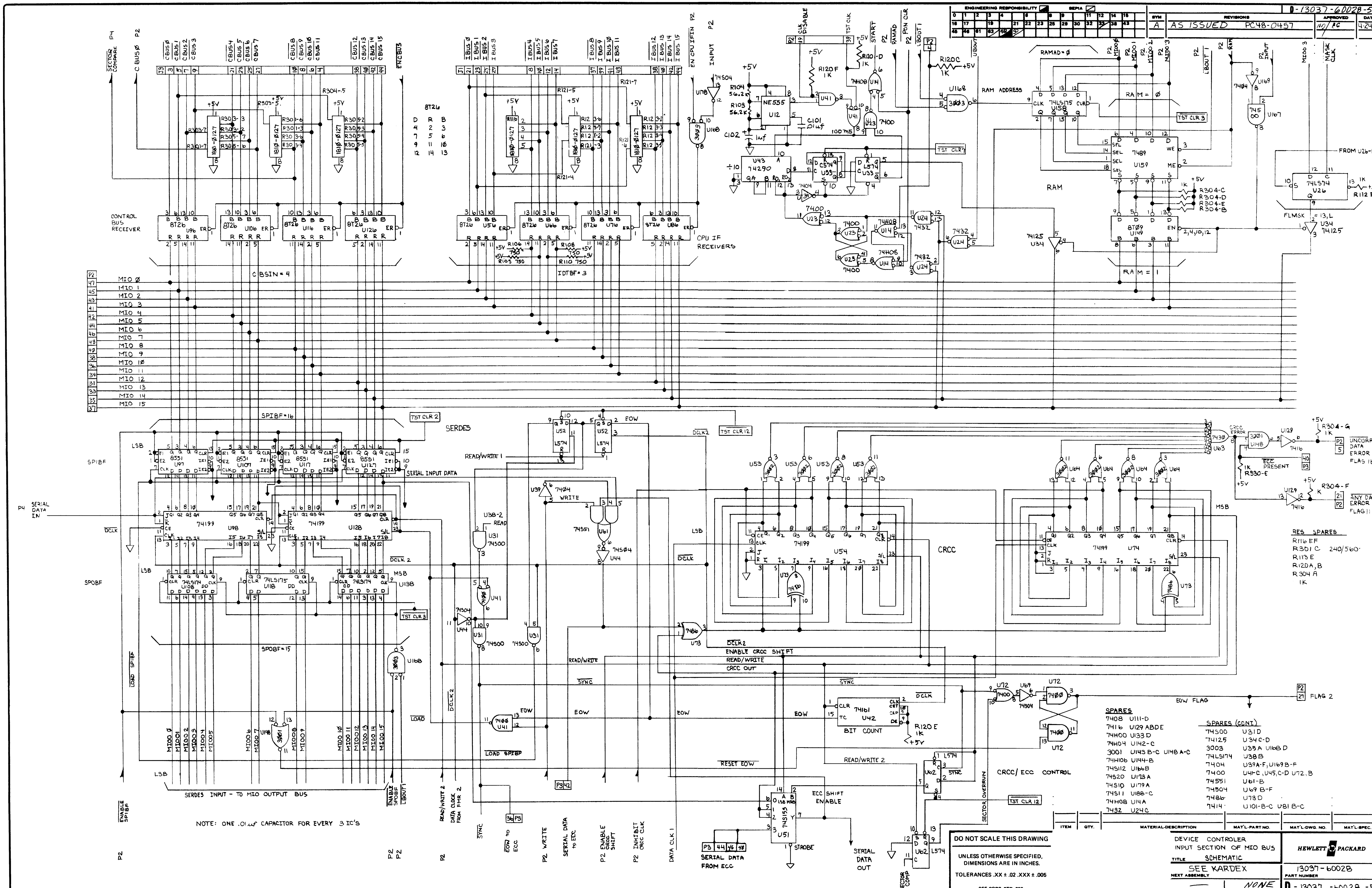
NEXT ASSEMBLY: SEE KARDEX

PART NUMBER: 13037-60028

FINISH: —

SCALE: NONE

D -13037 -60028 -51


D-13037-60028-52
ENGINEERING RESPONSIBILITY
SEPIA
A AS ISSUED PC48-0457 4-24-78
CONTROL BUS RECEIVER
CBSIN = 4
CPU IF RECEIVERS
IDTBF = 3
RAM ADDRESS
RAMAD = Ø
RAM = Ø
RAM = 1
RAM
TST CLR 3
TST CLR 7
FLMSK = 13, L
FROM U26-1
MIO Ø
MIO 1
MIO 2
MIO 3
MIO 4
MIO 5
MIO 6
MIO 7
MIO 8
MIO 9
MIO 1Ø
MIO 11
MIO 12
MIO 13
MIO 14
MIO 15
SPIBF = 16
TST CLR 2
SERDES
SPIBF
SERIAL DATA IN
SERIAL INPUT DATA
SPOBF
SPOBF = 15
READ/WRITE 1
EOW
TST CLR 12
DCLK 2
CRCC
UNCORR DATA ERROR FLAG 16
ECC PRESENT
ANY DATA ERROR FLAG 11
RES SPARES
R116 EF
R301 C 240/560
R113 E
R120A, B
R304 A
1K
ENABLE CRCC SHIFT
READ/WRITE
CRCC OUT
SYNC
BIT COUNT
EOW FLAG
FLAG 2
LOAD SPIBF
RESET EOW
READ/WRITE 2
CRCC/ECC CONTROL
ECC SHIFT ENABLE
STROBE
SERIAL DATA FROM ECC
SERIAL DATA OUT
SECTOR OVERRUN
SECTOR COMP
SERDES INPUT - TO MIO OUTPUT BUS
NOTE: ONE .01uf CAPACITOR FOR EVERY 3 IC'S
ENABLE SPIBF
READ/WRITE 2
DATA CLOCK FROM FMR 2
EOW TO ECC
WRITE
SERIAL DATA TO ECC
ENABLE CRCC SHIFT
INHIBIT CRCC CLK
DATA CLK 1
SPARES
7408 U111-D
7416 U129 ABDE
74H00 U133D
74H04 U142-C
3001 U143 B-C U148 A-C
74H106 U144-B
74S112 U166B
74S20 U173A
74S10 U179A
74S11 U188-C
74H08 U14A
7432 U24C
SPARES (CONT)
74S00 U31D
74125 U34C-D
3003 U35A U168D
74LS174 U38B
7404 U39A-F, U169B-F
7400 U41-C, U45, C-D U72, B
74551 U61-B
74S04 U69 B-F
7486 U73D
7414 U101-B-C U81B-C
ITEM QTY. MATERIAL-DESCRIPTION MAT'L-PART NO. MAT'L-DWG. NO. MAT'L-SPEC.
DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005
SEE CORP. STD. 608
DEVICE CONTROLER INPUT SECTION OF MIO BUS
TITLE SCHEMATIC
SEE KARDEX
NEXT ASSEMBLY
HEWLETT PACKARD
13037-60028
PART NUMBER
FINISH
SCALE NONE
D-13037-60028-52

ENGINEERING RESPONSIBILITY — SEPIA

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED PC48-0457	NO EC	4-24-78
B	R314 WAS 464. PC48-0926.	LO EC	8-29-78
C	CR305 WAS 1901-0463. PC48-1247.	LO EC	12-29-78

D-13037-60028-53

FLIP-FLOP STANDARD

+5V R331 1K — P3-16 — TST CLR 11 — 30 MHZ P3-2 — WRITE DATA TEST P3-10 — SERIAL DATA OUT — R332 1K — P3-18 — R330-B 1K — R330-G 1K — R330-C 1K — WRITE TEST P3-14 — U38-15 WRITE — SECTOR COMPARE — READ U38-2 — READ TEST P3-22 — R330-A 1K — WRITE GATE — READ GATE — DATA — SELECTED CLOCK — DELAY — T2 — ZERO PAIR — ONE — ABCD — C·D — DATA ENABLE — TST CLR 4 — TST CLR 8 — TST CLR 1 — ADVANCE — DELAY — SYNC BIT

R120-G 1K — R330-D 1K — P3-20 — P3-4 — P3-50 CLOCK INHIBIT — WRITE DATA — CR301 1901-0040 — P3-38 TEST OUTPUT — WRITE GATE — READGATE — READ DATA — DATA CLOCK #2 (J12-13) — DATA CLOCK #1 — P3-12 WRITE CLOCK — P3-32 28830 DATA CLOCK — P3-24 READ CLOCK — P3-26 READ NRZ DATA — SERIAL DATA IN (U98-1) — READ/WRITE3 (U44-12)

P1-25, 26, 27, 28 -15V — R122 56 2W — Q103 1826-0179 — C106 2.2mf 0160-0128 — 1205-0011 HEATSINK — C111 4.7mF — -5.2V

R322 6.8Ω — R323 10Ω 2100-3345 — R321 46.4Ω — CR302 1-0518 — CR303 1-0518 — R312 825 — R313 825 — C314 430pF — C315 .1 — R314 681 — 1000pf C316 — R315 464 — +5 — E303 — CR306 0122-0094 — CR305 1901-0866 — CR304 1901-0460 — C312 330mF 6V 0180-1714 — R316 1K — R317 1K — R320 100 — -5.2V — C320 .02 — E302 TP BIAS — C317 .01 — C321 .1 — C322 .01 — C323 .1 — C325 .01 — .56mH L301 9100-2232 — R324 121 — R326 121 — R325 82.5 — R327 82.5 — C307 .01 — R330-F 1K — +5V — -5.0

ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.

TOLERANCES .XX ± .02 .XXX ± .005

SEE CORP. STD. 608

DEVICE CONTROLLER FORMATTER / SEPARATOR

TITLE SCHEMATIC

SEE KARDEX — NEXT ASSEMBLY

— — FINISH — NONE — SCALE

HEWLETT PACKARD

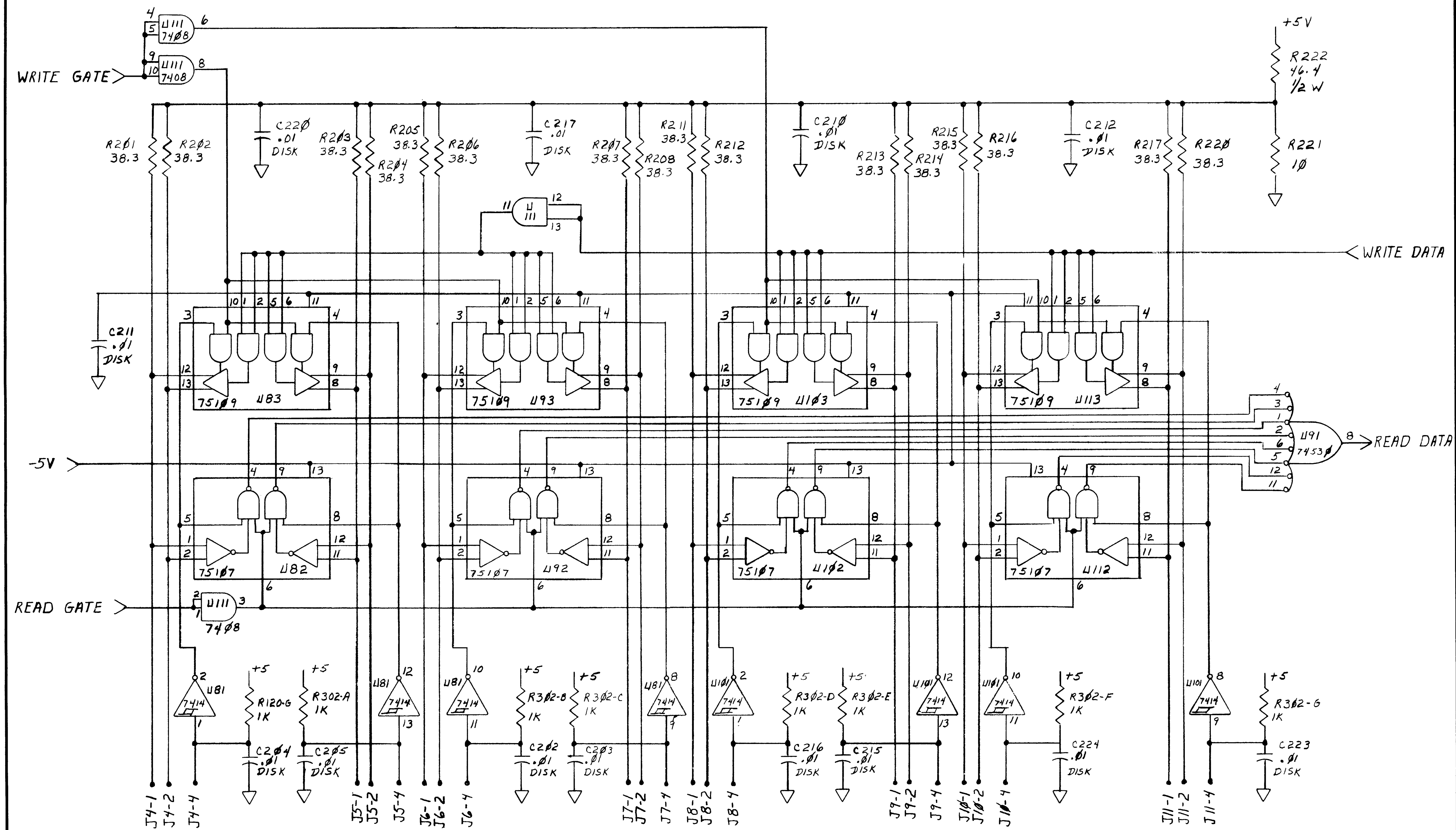
13037-60028 — PART NUMBER

D-13037-60028-53

ENGINEERING RESPONSIBILITY	SEPIA
0 1 2 3 4 6 8 9 11 12 14 15	
16 17 19 21 22 23 25 29 30 32 33 38 43	
45 46 61 63 48 40	

C-13037-60028-54

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED. PC48-0457	ND/EC	4-24-78



ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.

TOLERANCES .XX ± .02 .XXX ± .005

SEE CORP. STD. 608

DEVICE CONTROLLER SCHEMATIC

TITLE

SEE KARDEX

NEXT ASSEMBLY

FINISH — SCALE NONE

HEWLETT PACKARD

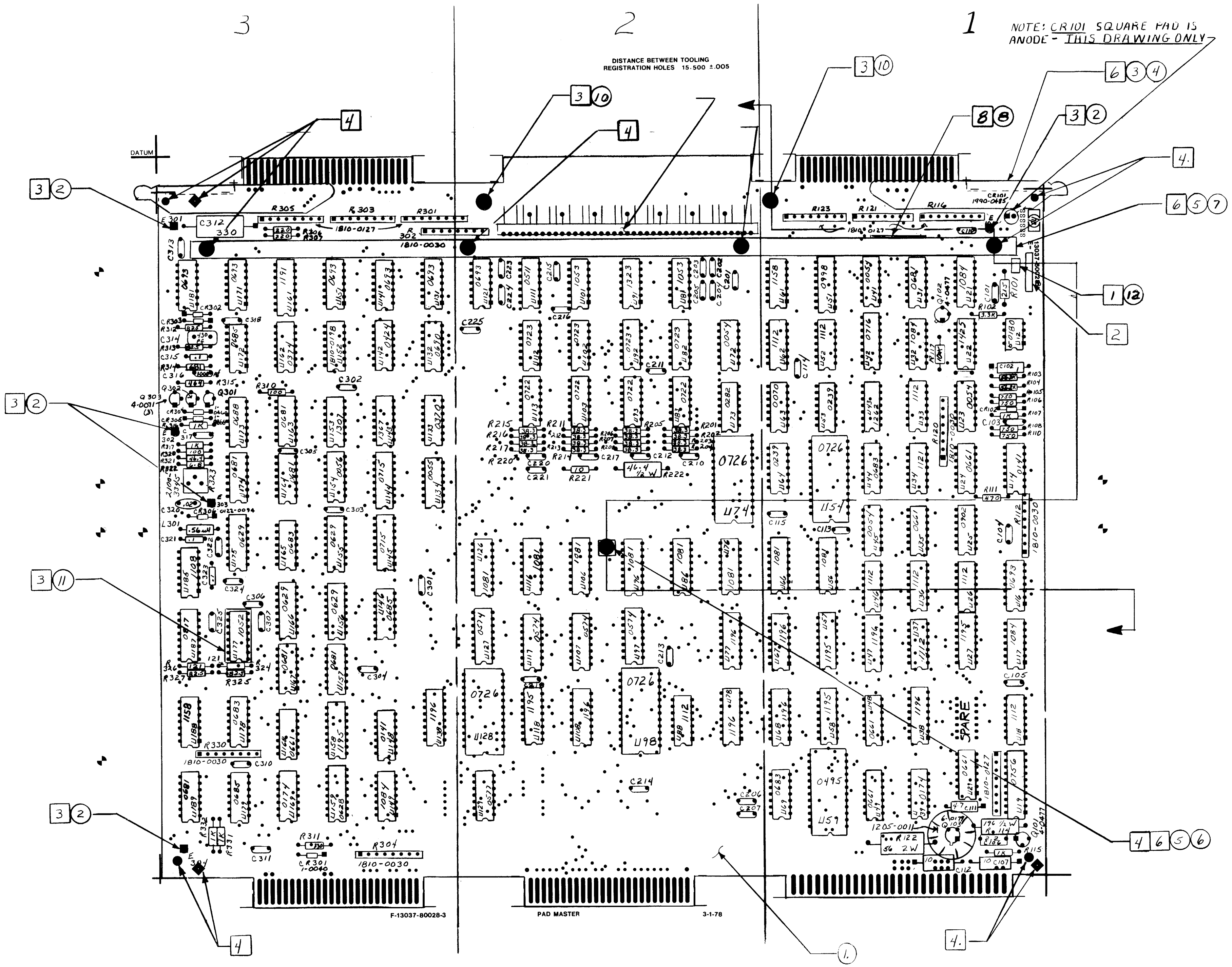
13037-60028

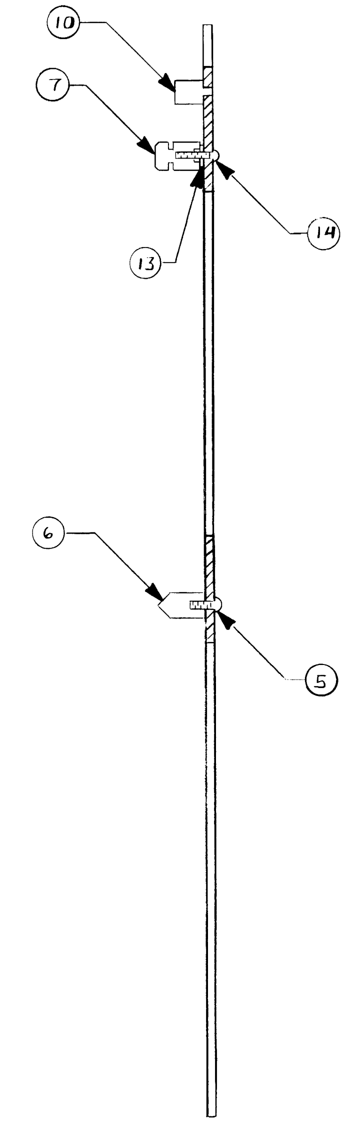
PART NUMBER

C-13037-60028-54

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED PC48-0457	WD/EG	4-18-78
B	SHOW U28 AS SPARE. U88 WAS 0686 PC48-0565	L.O./EG	5-16-78
C	R314 WAS 464, DATE CODE WAS 1815. PC48-0926.	L.O./EG	8-29-78
D	CR305 WAS 1901-0463. PC48-1247.	L.O./EG	12-29-78
E	ADD SECTION A-A, ADDED STIFFNER AND BUSHING 0340-0500 DELETED 5040-6058 BRACE DATE CODE WAS 1835 PC48-1542	TT	3-9-79

D-13037-60028-1





SECTION A-A

NOTES:

1. INSTALL ITEM (13) AFTER LINE TEST.
2. OPR. 264; MARK ASSEMBLY DATE CODE 1905.
3. OPR. 265; INSTALL ITEMS (2) 3 PLCS., (10) 2 PLCS., (7) 1 PLC. BEFORE WAVE SOLDER.
4. OPR. ; MASK PRIOR TO LOADING.
5. USE SUPPORT FIXTURE DURING WAVE SOLDER.
6. OPR. ; TOUCH UP, INSTALL ITEMS (3) THRU (7).
7. CLEAN CONNECTOR WITH ALCOHOL IN TOUCH UP.
8. INSTALL ITEM (8) AFTER OVEN BURN-IN.

REFERENCE DRAWINGS } SCHEMATIC D-13037 -80028 -51 52,53
C-13037 -80028 -54

UNLESS NOTED OTHERWISE:

ALL RESISTANCE IN OHMS
ALL RESISTORS 1/4 W ±1%
ALL CAPACITANCE IN MICROFARADS
ALL CAPACITORS .01 CER. DISC
ALL IC's 1820-______
ALL DIODES 1901-0518

	ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.
E	14	4	4-40 x 0.375 SCREW	2200-0143		
E	13	4	SHOLDED BUSHING	0340-0500		
	12	1	LABEL ORIGIN	7120-6830		
	11	1	SOCKET 16 PIN	1200-0482		
	10	2	STANDOFF 6-32X.312	0380-0198		
	9	1	CONNECTOR	1251-4219		
	8	1	LABEL	7120-7348		
E	7	1	BOARD STIFFENER	13037-20006		
	6	1	SPACER	5020-7318		
	5	5	SCREW	0624-0077		
	4	2	EXTRACTOR	5040-6009		
	3	2	PIN	1480-0116		
	2	5	TERMINAL E101 + E301-E305	0360-0294		
	1	1	BOARD ETCHED	13037-80028		

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.

TOLERANCES .XX ± .02 .XXX ± .005

SEE CORP. STD. 608

TITLE: DEVICE CONTROLLER ASSY. DWG.

HEWLETT PACKARD

NEXT ASSEMBLY: SEE KARDEX

PART NUMBER: 13037-60028

FINISH — SCALE —

D-13037-60028-1

CONTENTS

ERROR CORRECTION PCA

1.0 GENERAL DESCRIPTION

The 13037-60024 error correction (ECC) PCA is a single 3000 I/O-type printed circuit board installed within the 13037 disc controller. This board requires 3.5 amps.

Serial binary data is supplied to the ECC hardware along with the data clock during a WRITE data operation, and at the end of the data, 95 bits of check information are retrieved from the ECC hardware, to be written in the last six word-fields of a sector on the disc. These six words contain useful information that enables the 13037 controller to ensure data integrity or recovery in case of errors on subsequent READ data operations.

Serial binary data along with the data clock is supplied to the ECC hardware during certain READ operations (VERIFY command or RECC), and in the event that possibly correctable data errors are detected during the transfer, the CPU interface may request the 13037 controller to attempt to correct the errors. This is accomplished by issuing a single command RQSYN to the controller, and at the end of its execution seven words are returned to the interface that pertain to the recovery from the erroneous data transfer.

Every single burst error of length ≤ 32 bits in a sector can be corrected; every single burst error of length > 32 but ≤ 48 bits can be detected without being mis-corrected and 99.99% of all other errors will be detected.

2.0 BLOCK DIAGRAM AND FUNCTIONAL DESCRIPTION

The ECC hardware listens to the ECC firmware commands only; consequently, the proper execution of the ECC commands enables one to determine the proper functioning of the hardware. What follows is a general description and a command-by-command description of the hardware and its functions.

The ECC hardware basically consists of thirteen 8-bit 74199 shift registers (U23, U26, U53, U83, U106, U113, U126, U143, U146, U173, and U176) with appropriate feedback through XOR gates (U33, U34, U36, U37, U63, U64, U66, U67, U85, U86, U93, U94, U123, U136, U153, U156, U157, U183, U186, U187, and U168). The feedback is fanned out to the XOR gates or the 74199 D-inputs via two levels of 74S04 inverters U138, U158). There is a 74S157 multiplexer IC (U169) which, when applied with a "select" control signal (U108 pin 14), enables the thirteen 74199's to be interconnected to form two configurations:

A. The ENCODER. A single serial 95-bit shift register. This configuration is obtained for any WRITE data operations only.

B. The DECODER. Two serial-shift-registers: One 42 bits long, called the P-register (U113, U83, U53, U23, U173, and U143); and the other 53 bits long, called the C-register (U106, U76, U56, U26, U176, U146, and U126). Both have the serial DATA applied simultaneously to their inputs. This configuration is attained for any READ & VERIFY data operation only.

   See figure 1-A-B-C.

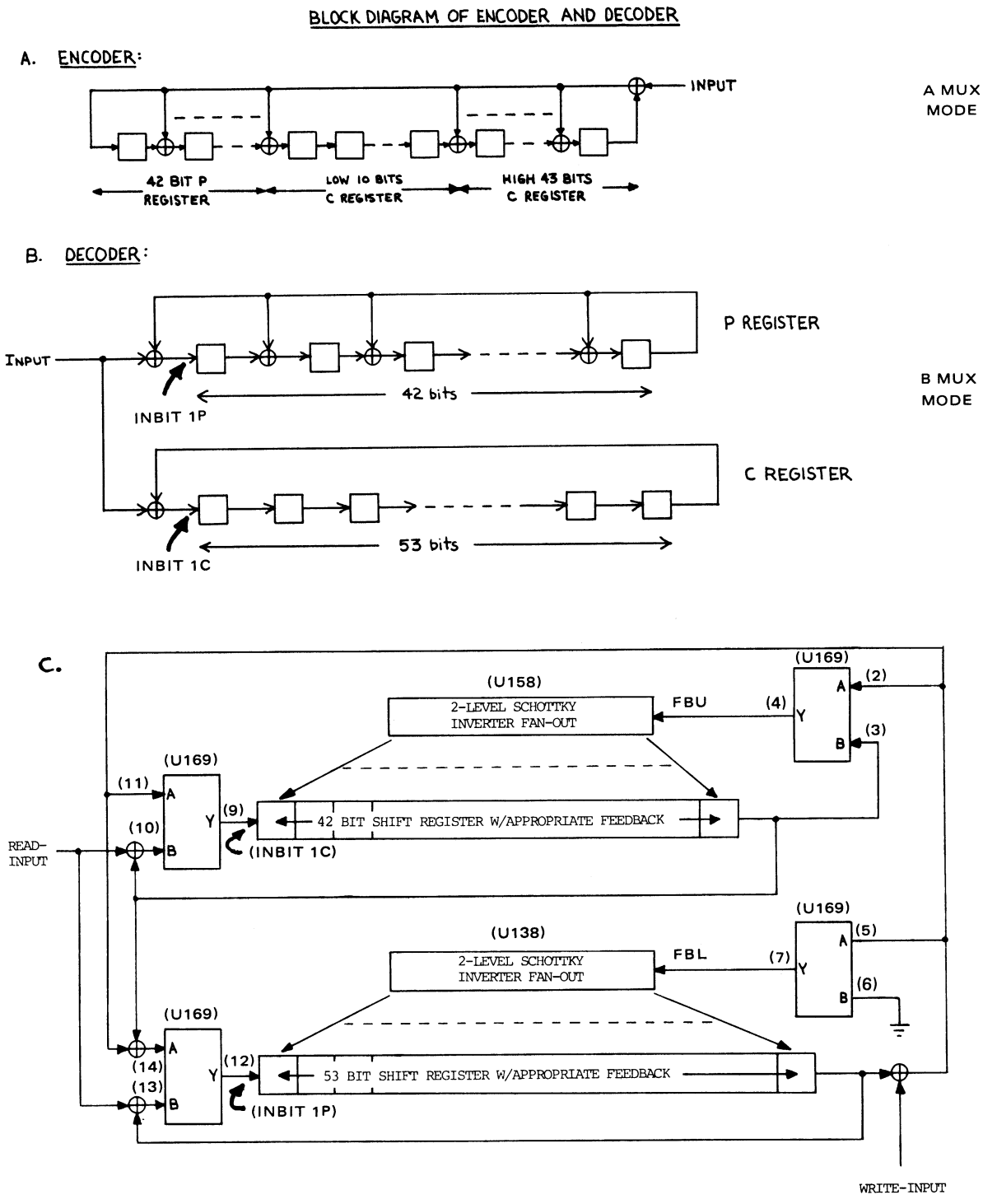


Figure 1. Block Diagram of ENCODER and DECODER

Functionally the ECC assembly can be organized into three different parts and corresponding modes of operation:

2.1 WRITE 1 SECTOR OPERATION

This is intiated by issuing the WECC firmware command, which produces an ECC internal control word (ECICW) interpreted as *WRITE with ECC* and subsequently latched into U108 for further decoding. The effect of the WECC command is:

a) To send out a low level to the multiplexer to select the encoder configuration.

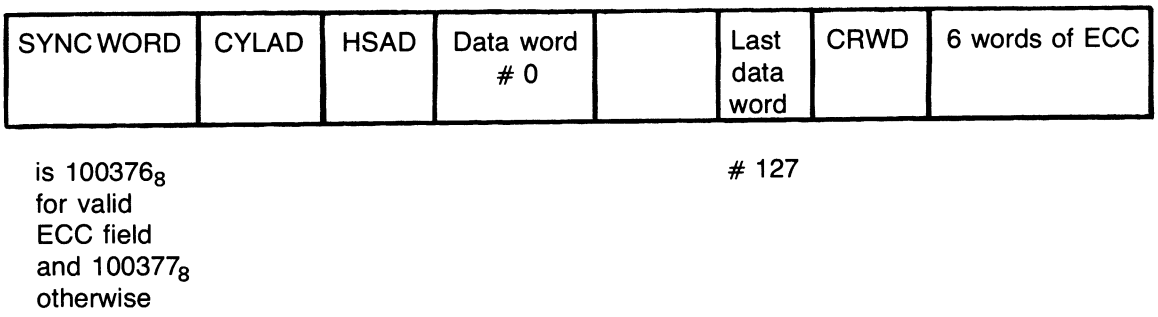
b) To reset the 7493A end-of-word counter IC (U159), and to reset latches U88 and U129 so that the INHIBIT on the shift register is removed and the shift registers are put into "load" mode.

c) To clear the shift registers.

d) To clear the data buffer, U177 (explained later).

e) To clear the "clock-suppressor", U181 and U182 (explained later).

Recall that the data stream looks as follows:



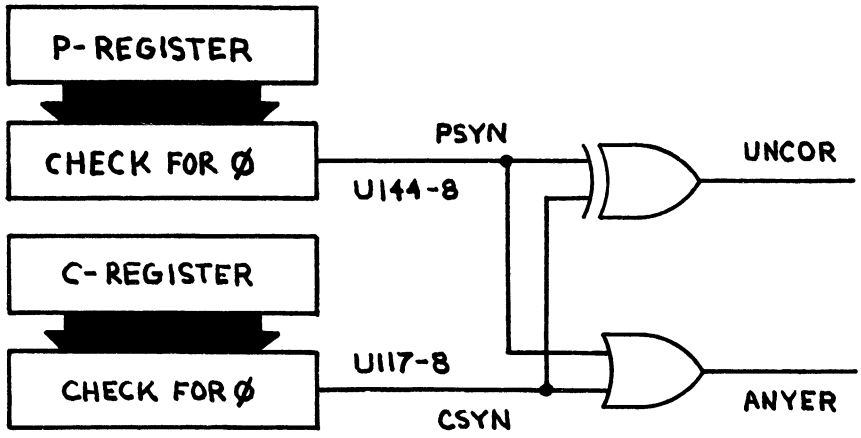
The last valid bit of "data" to be encoded by the ECC hardware during a WRITE operation is the last bit of the CRC word. Consequently after clocking in the last CRC bit, we must put the shift registers in "shift" mode. For this to happen, a firmware ECC command is executed during the transfer of the last data word (# 127). It is called ESC or "encoder sync control." Upon strobing this command into the ECICW latch (U108), the corresponding effect is the removal of the reset from the end-of-word counter (U159), and the initiation of the count of two EOW's (that of the last data word and that of the CRC word). Upon decoding this count, a latch (U88) is set so as to put the shift registers in shift mode, and the ensuing clock pulses will shift out six words of valid ECC information. This completes the description of the WRITE and the ENCODER's operation.

2.2 READ 1 SECTOR OPERATION

This is initiated by issuing a firmware command called RECC, which is interpreted by the ECICW as *READ with ECC*. The effect of this command is the same as that of the WECC command except that the multiplexer is selected in such a manner as to have the shift registers configured as a DECODER.

Again, analagous to the write operation, we must be able to control the mode of operation of the shift registers; however, in this case we continue to clock the registers in "load" mode only, being careful to INHIBIT clocking as soon as the last bit of the last ECC word has been clocked into the registers. To do this, a firmware ECC command is executed during the transfer of the

CRC word. It is called DSC or "decoder sync control." Upon validation of this command in the ECICW, the corresponding effect consists of the removal of the reset from the EOW counter and the initiation of the count of seven EOW's: that of the CRC word and each of the six ECC words. Upon decoding this count, a latch (U129) is set so as to INHIBIT the shift registers from clocking any further. The contents of the shift registers are examined in hardware by the check-for-0 logic, and the firmware examines two ground true flags (UNCOR and ANYER) to determine the integrity of the data transfer as follows:



If $\overline{\text{UNCOR}}$ is true, then an uncorrectable data error is reported. If $\overline{\text{ANYER}}$ is true, then a "possibly correctable data error" is reported. Otherwise, there was no error. This completes the description of the READ and the DECODER's error-detection process.

2.3 ERROR CORRECTION

Having detected a possibly correctable data error, the controller awaits a command from the corresponding CPU interface. In the case that it receives a RQSYN command (which is a request to find information required to correct the error), the controller now invokes the correction algorithm by calling the firmware subroutine RQSYN. Notice that in following read data operation we still have retained the contents of the shift registers, since it is these that contain the residue information required to find the error-patterns and the displacement.

The process consists of three parts, and each is described as follows:

2.3.1 BURST POSITIONING

This consists of positioning the error pattern in the lower leftmost 32 bits of the C-register, and is accomplished by executing these initial commands:

CLKHH — which takes the clock high and clears the data buffer (explained later), since we perform all the shifting with 0-input.

CLKHI — which insures that the clock is high while removing the INHIBIT from the registers and putting them in load mode.

At this point a hardware flag is checked to see if the burst is already positioned. If so, we go to seciton 2.3.2; otherwise we must shift the C-register until the burst is positioned or the shift-bound (54 shifts) expires. In order to shift the C-register exclusively, the command FRPSH (freeze P shift) is initially executed outside the burst positioning loop, which applies the INHIBIT on the P-register while maintaining the clock high. Through the use of a pair of

firmware instructions (namely CCLKH and CCLKL) the C-register clock is taken successively high and low to enable shifting with feedback, at each step adding an appropriate constant modulo to a displacement register until the burst is positioned or the shift-bound expires. If the shift-bound expires before retrieving the pattern, the error is uncorrectable.

2.3.2 MATCHING THE POSITIONED BURST

This consists of shifting the contents of the P-register with 0-input until its contents match those of the C-register, or until the shift-bound of the P-register (43 shifts) expires. The matching process is performed in IC comparator types 8130 and 8160 (U134, U74, U45, U174, and U164). The process is entirely analogous to burst positioning, using the analogous firmware instructions named PCLKH and PCLKL. If a match is not found, we return to section 2.3.1

2.3.3 ADJUSTING THE PATTERN TO WORD BOUNDARIES

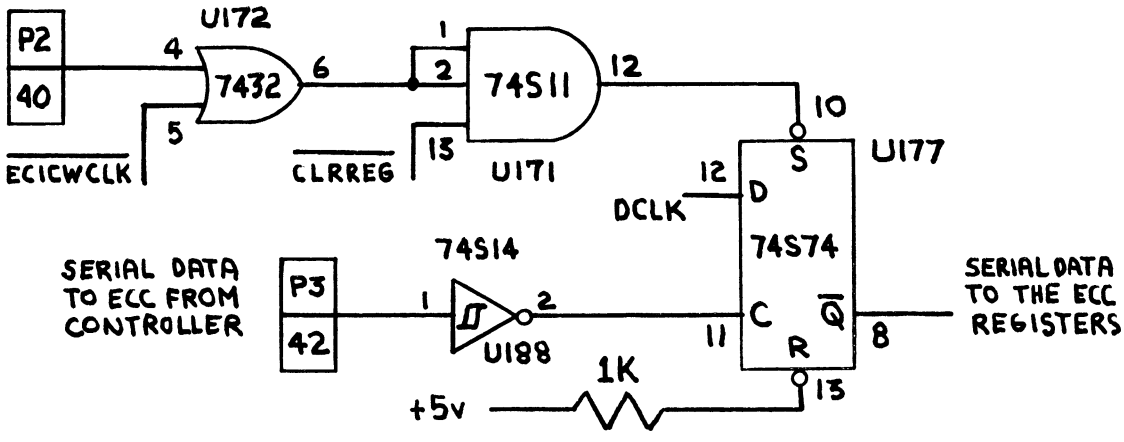
The displacements from the individual loops (previous two parts) are added together and reduced modulo a constant. The lower four bits are masked out and the scratch register shifted four bits to the right, thus performing a division by 16 so as to obtain the word displacement and bit-within-word displacement. This pattern is obtained from the shift registers via the MIO bus and is placed into two scratch registers; a third register is used along with the first two to contain three words of error pattern. The 32 bits are then left shifted in the three registers to adjust the patterns to word boundaries.

The patterns are retrieved from the shift registers onto the MIO bus via the 8123 quad two-input multiplexer IC's (U107, U97, U87, and U77). See figure 2.

2.4 HARDWARE

2.4.1 DATA BUFFER

The time interval between two successive bits of the data stream is only 133 nanoseconds; the propagation delay time for a data bit put out by the device controller before it reaches the ECC board, combined with the propagation delay through the ECC routing logic becomes prohibitive. Thus, it becomes necessary to buffer the data bit as soon as it enters the ECC board, synchronize it with the data clock, and then send it to the shift registers. The buffering is done with the following hardware:



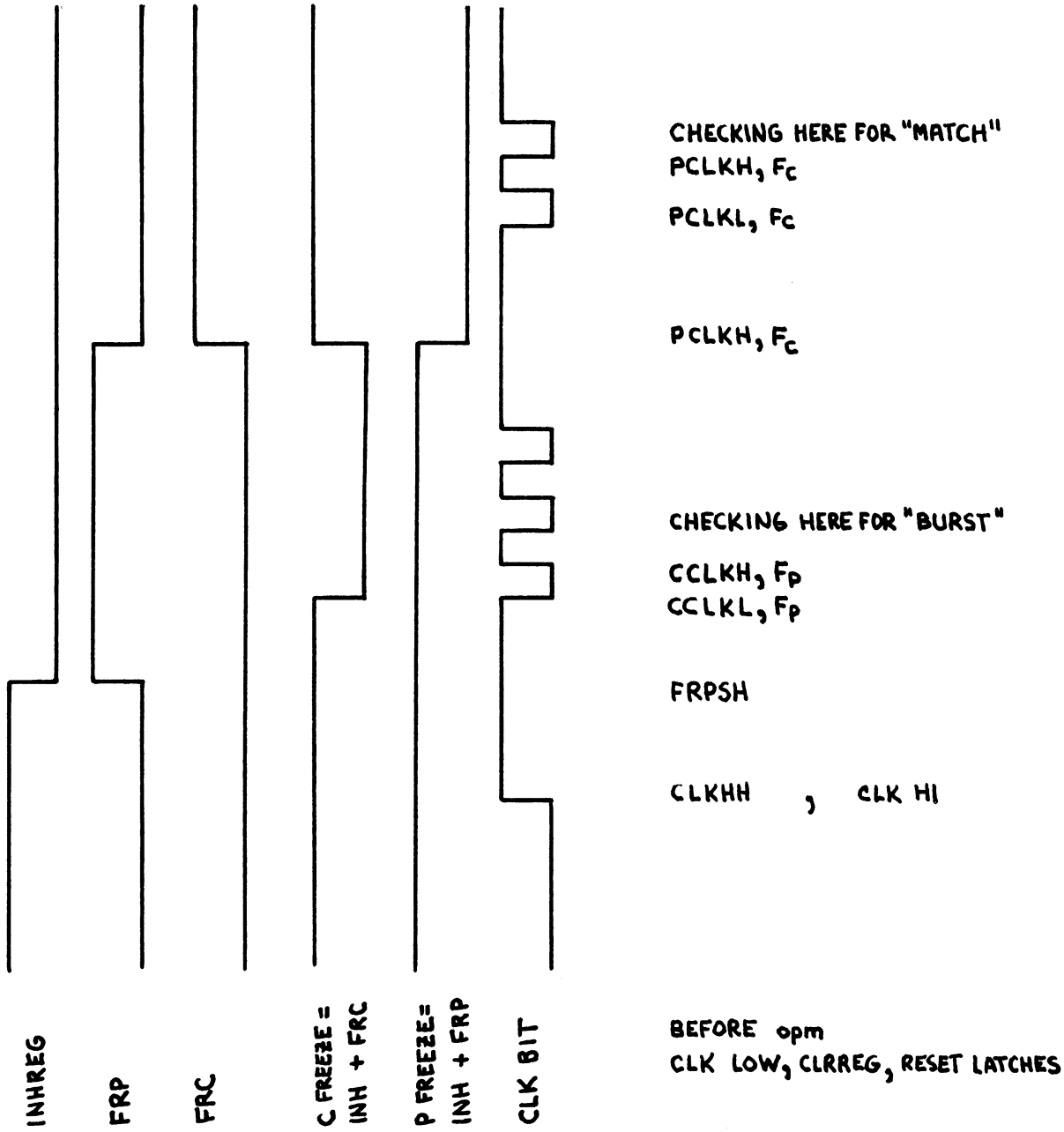


Figure 2. Error Correction Timing Diagram

This scheme has two effects:

1) The propagation delay due to the controller's routing logic is now eliminated, and consequently the delay that is due solely to the ECC logic is well within the bound of 133 ns.

2) When the last valid data bit (i.e., CRCLSB) comes out of the device controller, it is clocked into the data buffer, and is written onto the disc (it comes from the data FMTR/SPRTR); the ECC shift registers then clock in the second to the last CRC bit. An extra clock pulse is subsequently required to clock out the last CRC bit from the data buffer to the ECC registers. In so doing, an illegal bit of data gets written onto the disc occupying the MSB position of the first ECC word. This has the following repercussions:

   (i) In order to provide the extra clock pulse, the ENECC SHIFT latch must be set after a one-bit delay. This is an ENCODER related situation.

   (ii) On readback, we must suppress the clock pulse corresponding to the phony bit written in the ECC field. This situation is DECODER related.

2.4.2 CLOCK SUPPRESSOR

For ENCODER operations, the ECC hardware and the FMTR/SPRTR lag behind the device controller by one bit time. Thus all such control signals need to be delayed by one bit time. (Note: The device controller is accepting data from the CPU in "real time" — i.e., without delay.)

For READ operations, the device controller buffers the data coming from the FMTR/SPRTR for one bit time and then the ECC hardware again buffers it for one bit time. Nevertheless, since the ECC is synchronized with the device controller, the net delay with respect to the INHIBIT ECCCLK signal is only one bit time. The ECC, however, lags the FMTR/SPRTR in actual data bits by two bit times. This has to be taken into account in order to suppress the exact clock pulse corresponding to the phony bit as shown in the READ/WRITE timing diagram.

The clock suppressor hardware is on the following page:

2.4.3 TIMING OF READ AND WRITE OPERATIONS

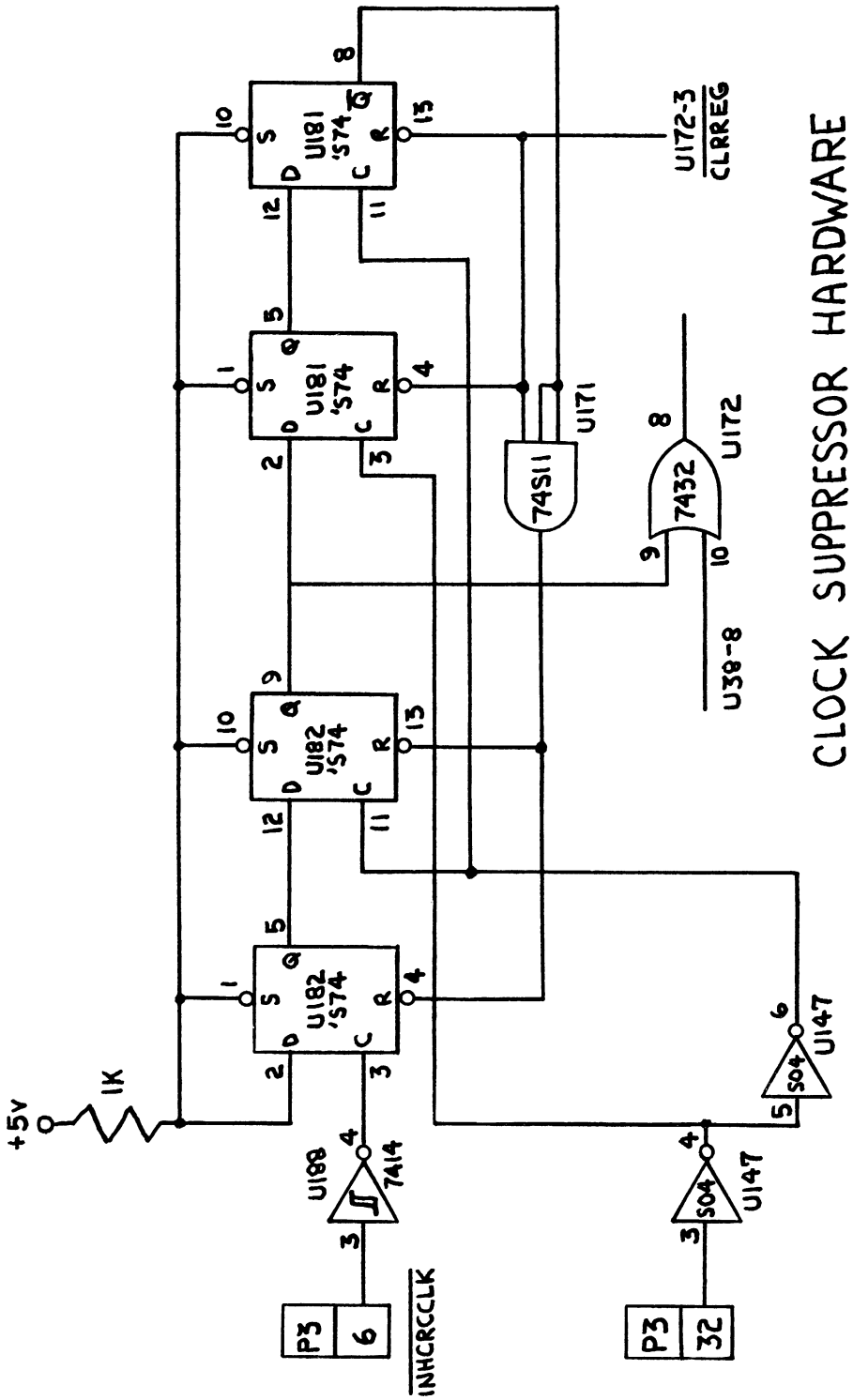
The timing sequence for WRITE and READ operations (see figure 3) is as follows.

WRITE operations:

(A1): SERDES → (puts out) bit 1
ECCSR (shift register) ← (accepts) dummy "0"
FMTR (formatter or data separator) ← bit 0
and ECCBUF (one-bit data buffer) ← bit 0

(A2): SERDES → bit 2
ECCSR ← bit 0
FMTR, ECCBUF ← bit 1

(A3): SERDES → bit 3
ECCSR ← bit 1
FMTR, ECCBUF ← bit 2


CLOCK SUPPRESSOR HARDWARE
+5V
1K
U181 '574
U182 '574
U171
74S11
U172
7432
U172-3
CLRREG
U38-8
U188
7414
U147
S04
P3
6
P3
32
INHCRCCLK

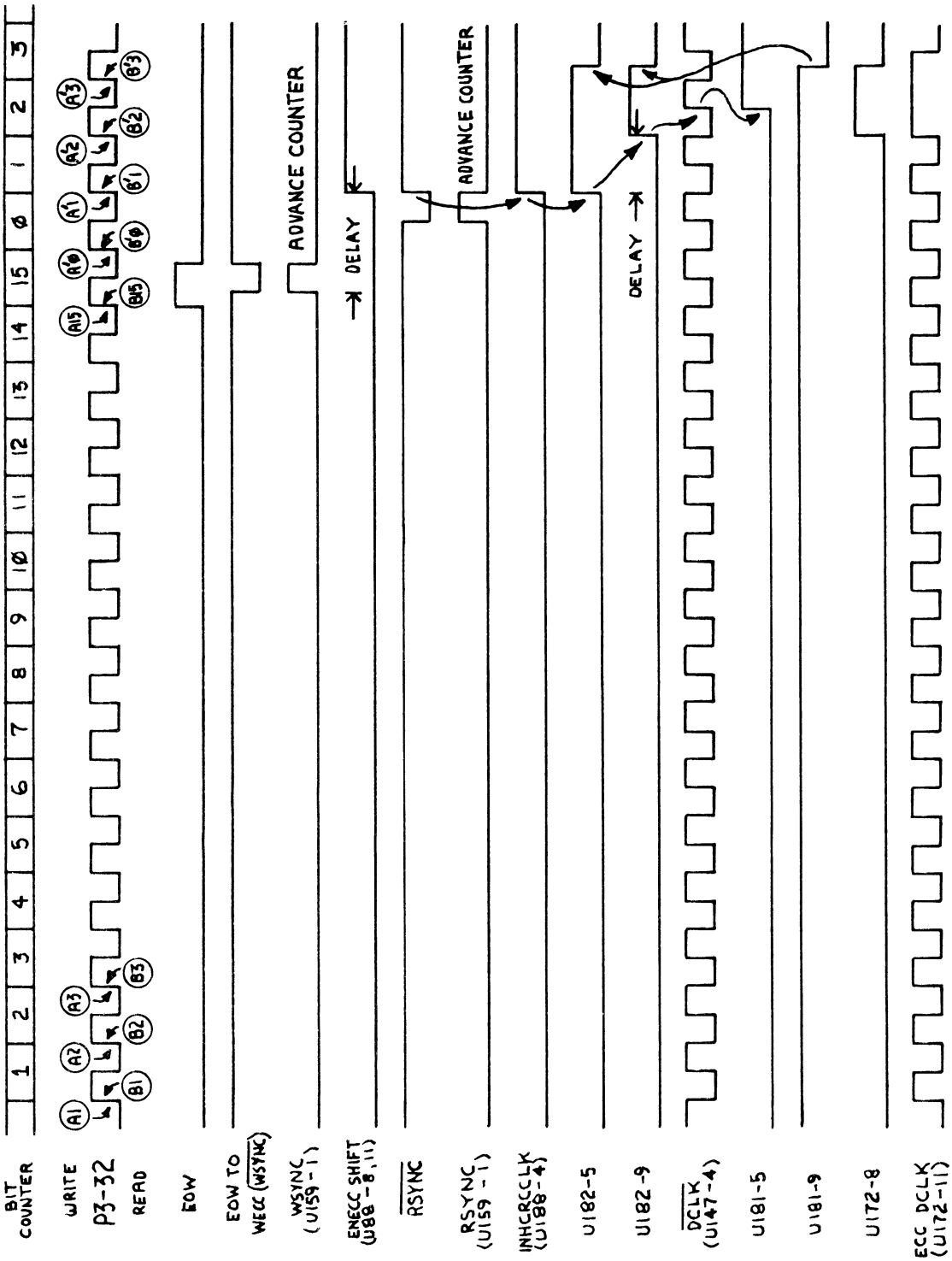


Figure 3. Timing Sequence for READ and WRITE Operations

(A15): SERDES → bit 15
ECCSR ← bit 13
FMTR, ECCBUF ← bit 14

(A'0): SERDES → bit 0
ECCSR ← bit 14
FMTR, ECCBUF ← bit 15

(A'1): SERDES → bit 1
ECCSR ← bit 15
FMTR, ECCBUF ← bit 0

Thus, if bit 15 was CRCLSB, then bit 0 here is the phony ECC MSB that gets unavoidably written and must be ignored on READ by suppressing the corresponding positive edge of the DCLK seen by ECC shift registers.

READ operations:

(B1): FMTR → bit 0
SERDES, ECCBUF ← dummy 0
ECCSR ← dummy 0

(B2): FMTR → bit 1
SERDES, ECCBUF ← bit 0
ECCSR ← dummy 0

(B3): FMTR → bit 2
SERDES, ECCBUF ← bit 1
ECCSR ← bit 0

(B15): FMTR → bit 14
SERDES, ECCBUF ← bit 13
ECCSR ← bit 12

(B'0): FMTR → bit 15
SERDES, ECCBUF ← bit 14
ECCSR ← bit 13

(B'1): FMTR → bit 0
SERDES, ECCBUF ← bit 15
ECCSR ← bit 14

(B'2): FMTR → bit 1
SERDES, ECCBUF ← bit 15
ECCSR ← bit 14

(B'3): FMTR → bit 2
SERDES, ECCBUF ← bit 1
ECCSR ← bit 0

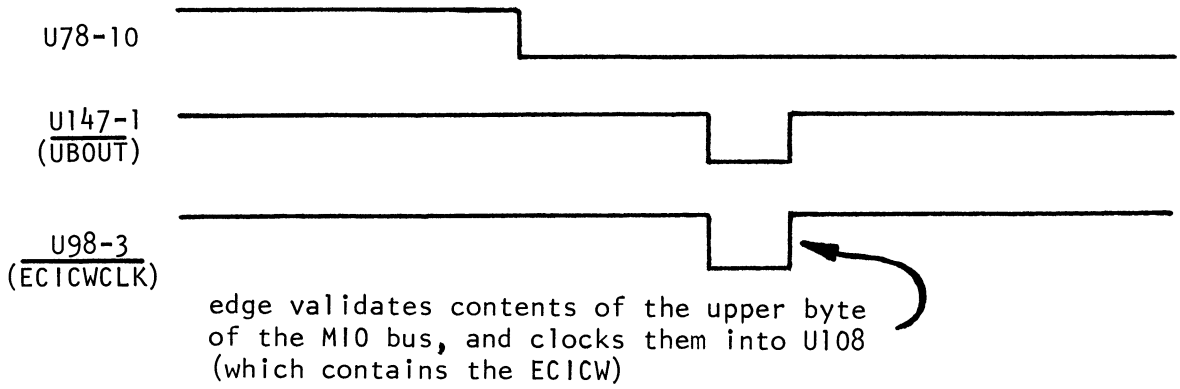
Thus, if bit 15 was CRCLSB, then bit 0 here is the phony ECC MSB that should not be clocked into the ECC shift registers, and the positive edge of clock pulse corresponding to it (shown in figure 3 at B'3) should be suppressed. This is accomplished by the "clock suppressor" made from U181, U183, U147, and U171.

3.0 EXPANDED DESCRIPTION

3.1 FIRMWARE COMMANDS

Presented below is a detailed description of the ECC hardware in conjunction with the firmware commands executed.

Whenever the ECICW is addressed in either a DO or DI (data out or data input) instruction, the address decoder (U78) puts out a low level on U78-10. This, combined with $\overline{\text{UBOUT}}$ (the signal for upper byte instructions on the MIO bus) provides the pulse $\overline{\text{ECICWCLK}}$ at U98-3 as follows:

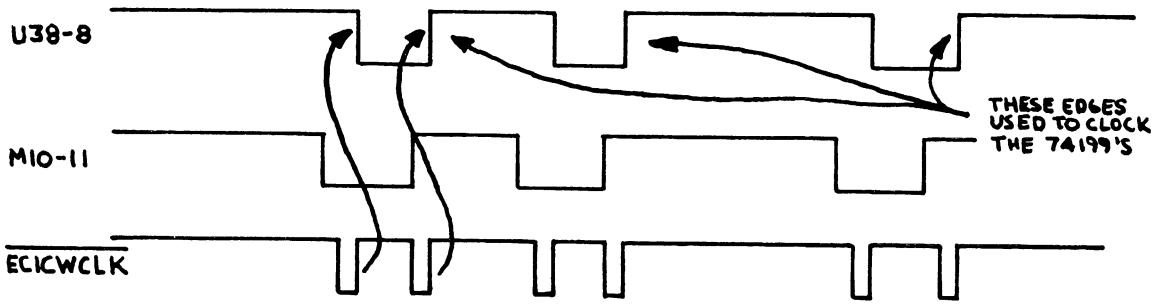


The ECC firmware commands are:

OCTAL	ECC Firmware Command	MIO Bus Contents							
		Bit 15	14	13	12	11	10	9	8
26	WECC	0/H	0/H	0/H	1/L	0/H	1/L	1/L	0/H
0	ESC	0/H	0/H	0/H	0/H	0/H	0/H	0/H	0/H
226	RECC	1/L	0/H	0/H	1/L	0/H	1/L	1/L	0/H
200	DSC	1/L	0/H	0/H	0/H	0/H	0/H	0/H	0/H
211	CLKHH	1/L	0/H	0/H	0/H	1/L	0/H	0/H	1/L
230	CLKHI	1/L	0/H	0/H	1/L	1/L	0/H	0/H	0/H
312	FRPSH	1/L	1/L	0/H	0/H	1/L	0/H	1/L	0/H
300	CCLKL	1/L	1/L	0/H	0/H	0/H	0/H	0/H	0/H
310	CCLKH	1/L	1/L	0/H	0/H	1/L	0/H	0/H	0/H
240	PCLKL	1/L	0/H	1/L	0/H	0/H	0/H	0/H	0/H
250	PCLKH	1/L	0/H	1/L	0/H	1/L	0/H	0/H	0/H

Since the MIO bus is ground true, a logical "0" will be a high level and a logical "1" will be a low level. It should also be pointed out what the various MIO bits 8 through 15 signify to the ECC assembly when validated into the ECICW by $\overline{\text{ECICWCLK}}$.

MIO Bus Bit Number	Description
15	If 0/H is on the input U108-13, the $\overline{Q}$ output U108-14 goes low, which helps to select the A-inputs on the mux U169 to provide the ENCODER configuration. If 1/L is on U108-13, we obtain a high at U108-14 which helps to select the B-inputs on U169, thus providing the DECODER configuration.
14	When 0/H is on the input U108-12, the $\overline{Q}$ output U108-11 goes low; when 1/L is on the input, this output goes high, enabling in this case to INHIBIT the clock on the P-register (this signal is called F_P or "freeze P).
13	When 0/H is on the input U108-4, the $\overline{Q}$ output U108-3 goes low; when 1/L is on the input, this output goes high, enabling in this case to INHIBIT the clock on the C-register (this signal is called F_C or "freeze C").
12	When 0/H is on the input U108-5, the Q output U108-7 goes high; the $\overline{Q}$ output U108-6 goes low, thus removing the reset from the EOW Counter U159 and enabling it to count. When 1/L on the input U108-5, the Q output U108-7 goes low, which helps to reset the ENECCSHIFT/LOAD latch U88 and also the INHREG latch U129; the $\overline{Q}$ output U108-6 goes high and resets the EOW counter U159 (the $\overline{Q}$ output is called "$\overline{SC}$" or "$\overline{\text{sync control}}$").
11	When 0/H is on the input U38-12, the Q output U38-8 goes low; when 1/L is on the input U38-12, the Q output U38-8 goes high. Successive applications of highs and lows enables us to provide "clock" pulses to the shift registers for shifting under firmware control (this signal is called "CLK" or clock).



10	When 1/L is at the inputs U172-2 and U98-4-12, the outputs are pulses at U172-3, U98-6-11, which are used to clear the data buffer, the clock suppressor, and the P and C registers (this signal is called "$\overline{\text{CLRREG}}$" or "$\overline{\text{clear the registers}}$").
9	This line is not used and as such has a "don't care" level.
8	When 1/L is at the input U172-4, its effect is to clear the one-bit data buffer U177 to enable the shifting with feedback of the P and C registers without input (this signal is called "$\overline{\text{CLRDATABUFFER}}$").

3.2 WRITE OPERATIONS

Let us consider what happens for write operations such as WRITE and INITIALIZE.

Initially the WECC command is put out on the upper byte of the MIO bus via the instruction D0: ECCICW ← WECC, U and the bus contents are validated by $\overline{\text{ECICWCLK}}$.

The effect of executing this instruction is as follows. U108-14 goes low so that the A-inputs are selected at mux U169 (i.e., the registers are in ENCODER configuration); also U139-8 is held high and thus prevents the INHIBIT-REG latch from being set. U108-15 goes high, thus removing the reset from U149-13 (whose function is to delay the count of two EOW's by one bit time).

U108-11 (F_P) goes low thus removing the inhibit from P-register.

U108-3 (F_C) goes low thus removing the inhibit from C-register.

U108-6 (SC) goes high thus resetting the EOW counter U159.

U108-7 (SC) goes low thus resetting the INHREG latch U129 and resetting the ENSHIFT/LOAD latch U88 (puts the shift registers into load mode). U38-8 goes low, thus allowing the data clock (as opposed to the firmware-supplied clock) to be passed to the shift registers via U172-11 and U118-3-8.

U172-3 and U98-6-11 put out a negative pulse which clears the one-bit data buffer (U177-8), the clock suppressor (U181), and the P and C registers.

After this command we execute the instruction DI BUF3 ← ESYNC, L which puts a logical "0" in the LSB of the sync word. See U28-1-2-3 and MIO 0.

During the transfer of the last data word we execute the instruction DO ECICW ← ESC, U and its effect is as follows:

U108-14 remains low for the ENCODER configuration.

U108-15 remains high.

U108-11 remains low.

U108-3 remains low.

U108-6 goes low, thus removing the reset from U159-2-3 and allowing it to count subsequent EOW's.

U108-7 goes high, thus removing the reset from the ENSHIFT/LOAD latch U88.

Now after counting two EOW pulses, U159-8 goes high and gets clocked onto U149-12 at the next positive edge of DDCLK; the output U149-9 goes high, U88-3 goes low, and the latch U88 gets set. Consequently, U88-8-11 both go high, putting the shift-registers into the shift mode; corresponding clocks applied to the shift-registers will cause the ECC parity bits to be written onto the disc. See figure 4.

Note that for all write operations U108-14 is low and U108-15 is high in order to maintain the ENCODER configuration.

3.3 READ OPERATIONS

Now let us consider read operations such as READ and VERIFY (in VERIFY, no data is passed to the CPU, yet the ECC hardware treats it just as if it were a READ command).

Throughout READ and VERIFY data operations in the controller the ECC board is maintained in the DECODER configuration; thus all ECC commands related to these operations

will maintain U108-14 high and U108-15 low. U108-15 being low prevents the ENSHIFT/ LOAD latch U88 from being set; thus the shift-registers will remain in the load mode throughout these operations.

Initially the RECC command is put out by executing a DO ECICW ← RECC, U instruction.

U108-3-11 both go low, so as to remove their individual INHIBIT's from the C and P registers, respectively.

U108-6 goes high and resets the U159 EOW counter as in WECC.

U108-7 goes low and resets the latches as in WECC.

U38-8 goes low to allow the data clock from P3-32 to pass through as in WECC.

U172-3 and U98-6-11 all issue a negative clear pulse as in WECC.

The next command executed is DO DSYNC ← REG0, L. The purpose of this instruction is to decode the LSB of the SYNC word in U38 to determine whether the sector being transferred or verified has valid data in the ECC field. See U38-2-3-6.

During the transfer of the CRC word, we execute the instruction DO ECICW ← DS, U and its effect is as follows.

Leaving everything else unchanged, U108-6 goes low, thus removing the reset from U159-2-3 and allowing it to count subsequent EOW's. U108-7 goes high thus removing the reset from the INHREG latch U129 and its one-bit delay buffer U149-2-3-5. Note that since U108-14 is high for read operations, it enables U139-8 to decode the count of seven EOW's; since U108-15 is low it disables the latch U88 from being set.

After counting seven EOW's U139-8 goes low; after the next data clock positive edge U149-5 goes low and sets the latch U129-8. The shift-registers are now inhibited and the operation is completed. See figure 5.

This completes the description of all data operations, and should be understood in conjunction with the timing diagram in figure 3.

3.4 ERROR RECOVERY

3.4.1 DETECTION

At the end of the data transfer or verification, the following instructions are executed to determine the integrity of the operation:

CC = UNCOR, RS	/Was there an uncorrectable data error? Check flag P2-5.
BR = UDTER	/Yes, set up appropriate status and interrupt.
CC = ANYER, RS	/No, was there a (possibly) correctable data error? Check flag P2-21.
BR: CDTER	/Yes, set up appropriate status and interrupt.
RT: 0	/No data error was detected.

Flag $\overline{\text{ANYER}}$ at P2-21 will be low if any data error (uncorrectable or possibly correctable) was detected at the end of the operation.

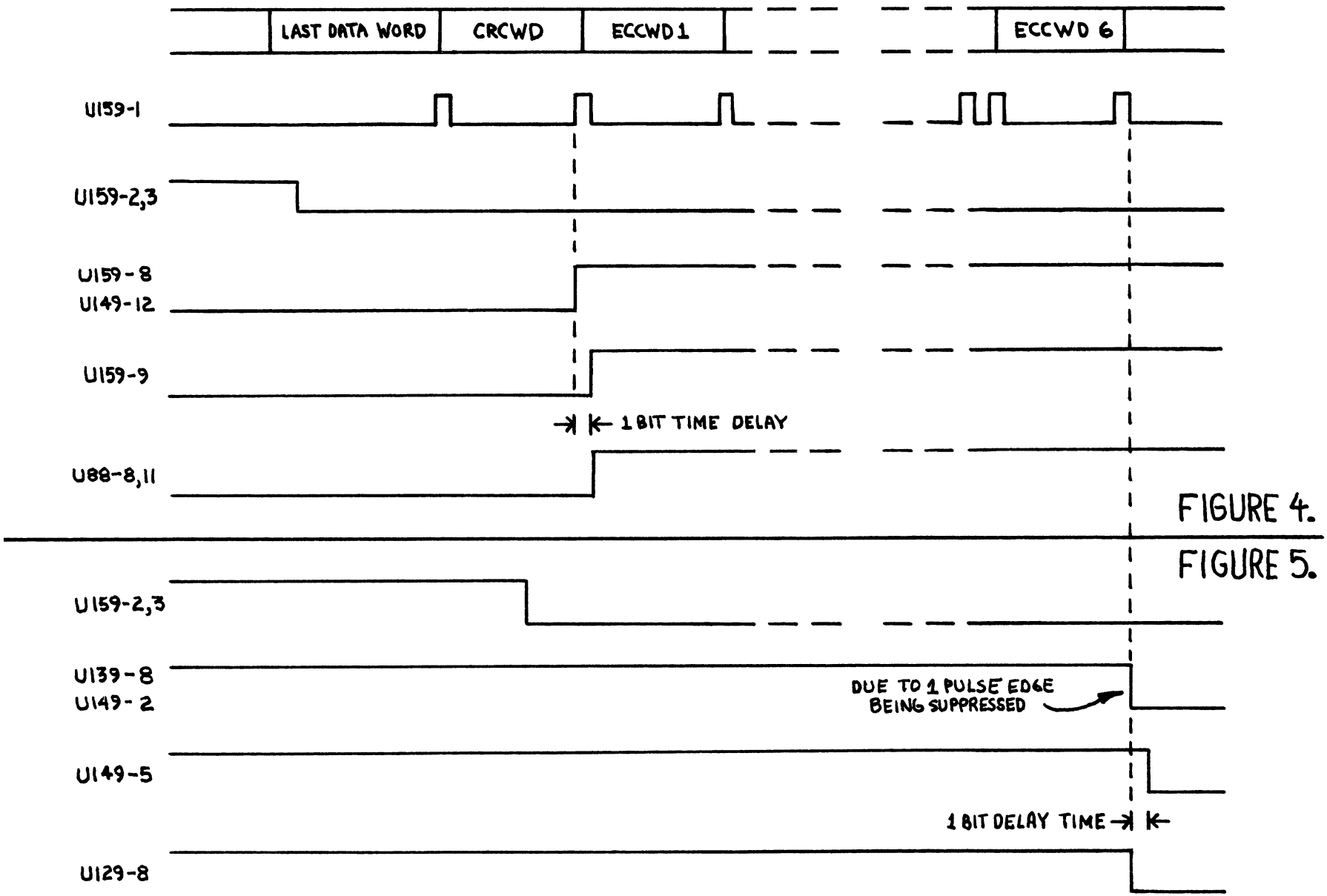


FIGURE 4.

FIGURE 5.

3.4.2 CORRECTION

Error correction is invoked by a branch to the RQSYN routine in the ROM. Again note, U108-14 is high throughout this routine to maintain the DECODER configuration.

A. Burst Positioning

Initially three instructions are executed — they are:

1) DO ECICW ← CLKHH, U

   U38-8 is taken high, so that the clock inputs on the shift-registers are high.

   U172-6 outputs a negative pulse to clear the data buffer so that the registers may be clocked with "0" input only.

2) DO ECICW ← CLKHI, U

   While the clock input to the registers is held high by maintaining U38-8 high, U108-7 goes low to reset latches U88 and U129 so as to permit the subsequent clocking of the shift-registes in load mode.

3) DO ECICW ← FRPSH, U

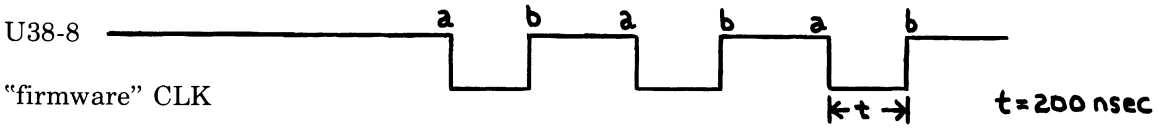
   While maintaining U38-8 high (line the clock inputs on the shift registers), U108-11 (F_P) goes high while U108-3 (F_C) goes low, thus relieving the inhibit on the C-register and applying the inhibit to the P-register to enable selective and exclusive clocking of the C-register.

Following this initial set-up, a set of instructions are conditionally executed as below:

```
C   CC = BURST, RS
    BR: PLOOP

    DO ECICW ← CCLKL, U
    DO ECICW ← CCLKH, U
```

This pair of instructions, when executed, provides the clocking mechanism required to clock the enabled (uninhibited) shift-register



After each "clocking" of the C-register the condition C is checked again, and if the flag called "BURST" at U119-9 goes low, we skip this set of instructions and branch to the next phase; otherwise, we continue until the shift bound (of 53) expires (at which point we report "uncorrectable error" status and exit from the RQSYN routine).

B. Burst Matching

The next phase consists of executing an initial instruction

DO ECICW ← PCLKH, U

which makes U108-11 (F_P) go low, thus removing the INHIBIT from the P-register; also U108-3 (F_C) goes high, thus applying the INHIBIT to the C-register.

Then a pair of similar clocking instructions are conditionally executed as follows:

```
P   CC = MATCH, RS
    BR:  GTPAT

    DO ECICW ← PCLKL, U
    DO ECICW ← PCLKH, U
```

which accomplish the same function as CCLKL and CCLKH on the P-register (which is not now inhibited).

After each "clocking" of the P-register the condition P is checked again, and if the flag called "MATCH" (at U119-13) goes low, we skip this set of instructions and branch to the next phase to obtain the error pattern; otherwise, we continue until the P-register shift bound (43) expires, at which point we return to the previous phase (Section A) again.

The flag "BURST" can be traced back to the shift registers, and employs some of the "check-for-0" circuitry. It is ground true and goes low whenever the Q output of U106-4 is high and all the Q outputs of U176, U146, and U126-4-6-8-10-15 are low.

The flag "MATCH" can also be traced back to the shift registers and employs the comparators U134, U74, U45, U174, and U164. It is also a ground true flag and goes low only when all the corresponding outputs of the P-register and the C-register match. Refer again to figure 2 for a general idea.

C. Pattern Retrieval

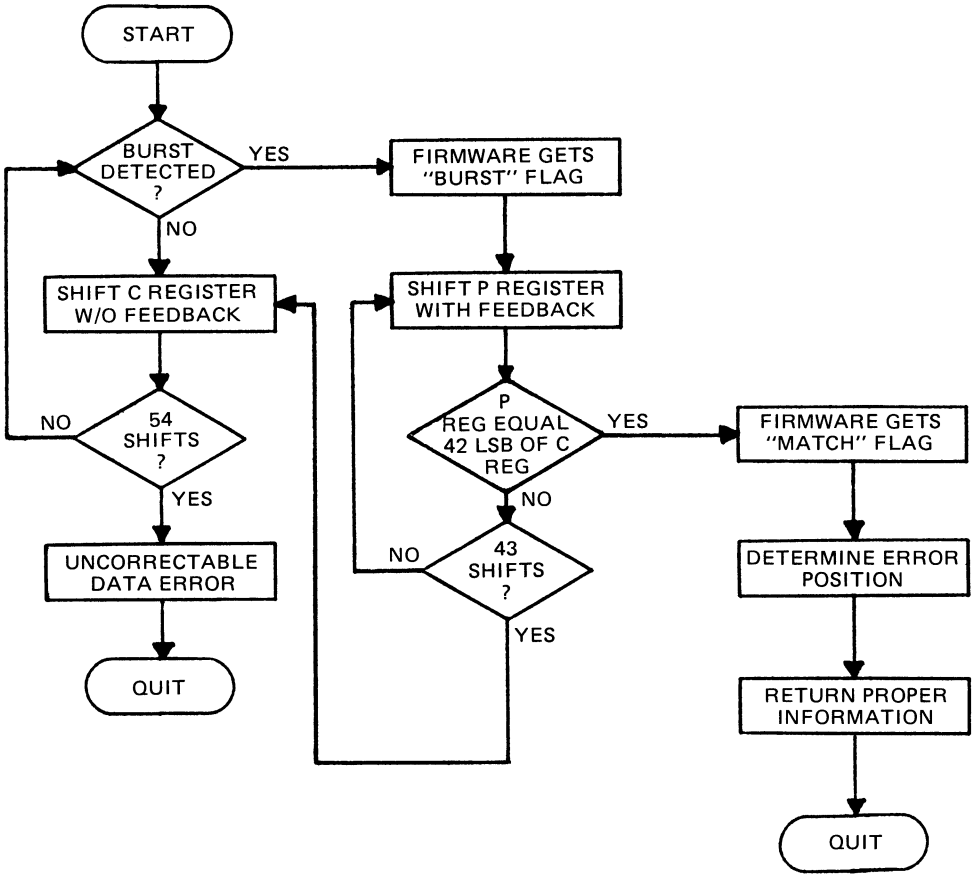
In this phase the main instructions executed are:

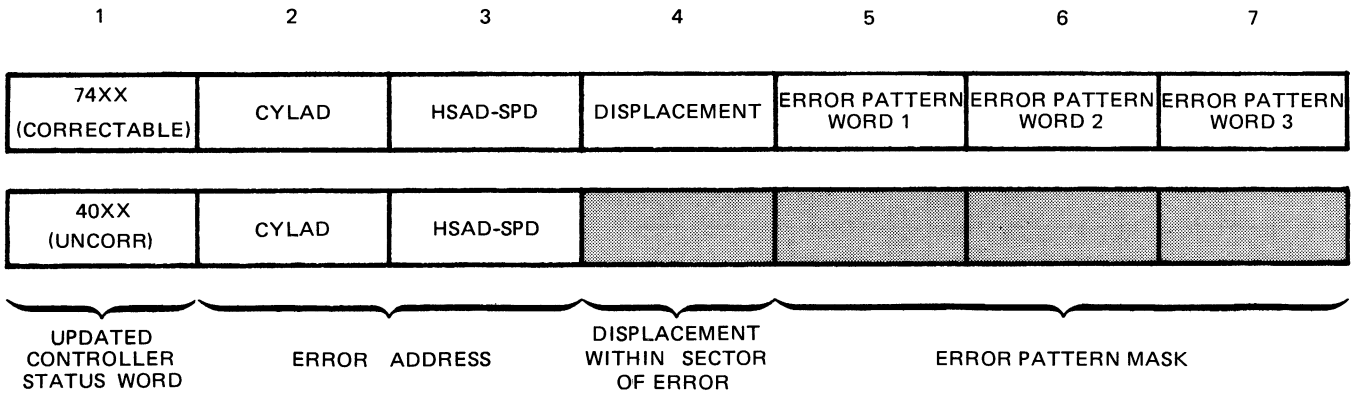
```
DI XREG ← WORD 1
DI YREG ← WORD 2
```

The address decoder U78 decodes the external addresses (WORD 1 and WORD 2) and enables the three-state multiplexers U107, U97, U87, and U77, selecting them successively so that the contents at their I_0 inputs are first put on the bus (then those at the I_1 inputs are put out).

D. Final Correction

This is the last phase of the entire operation and does not involve the ECC hardware at all. The patterns are now adjusted to word boundaries under firmware control and passed to the CPU interface for subsequent XORing with the original data (thus accomplishing the final error correction).



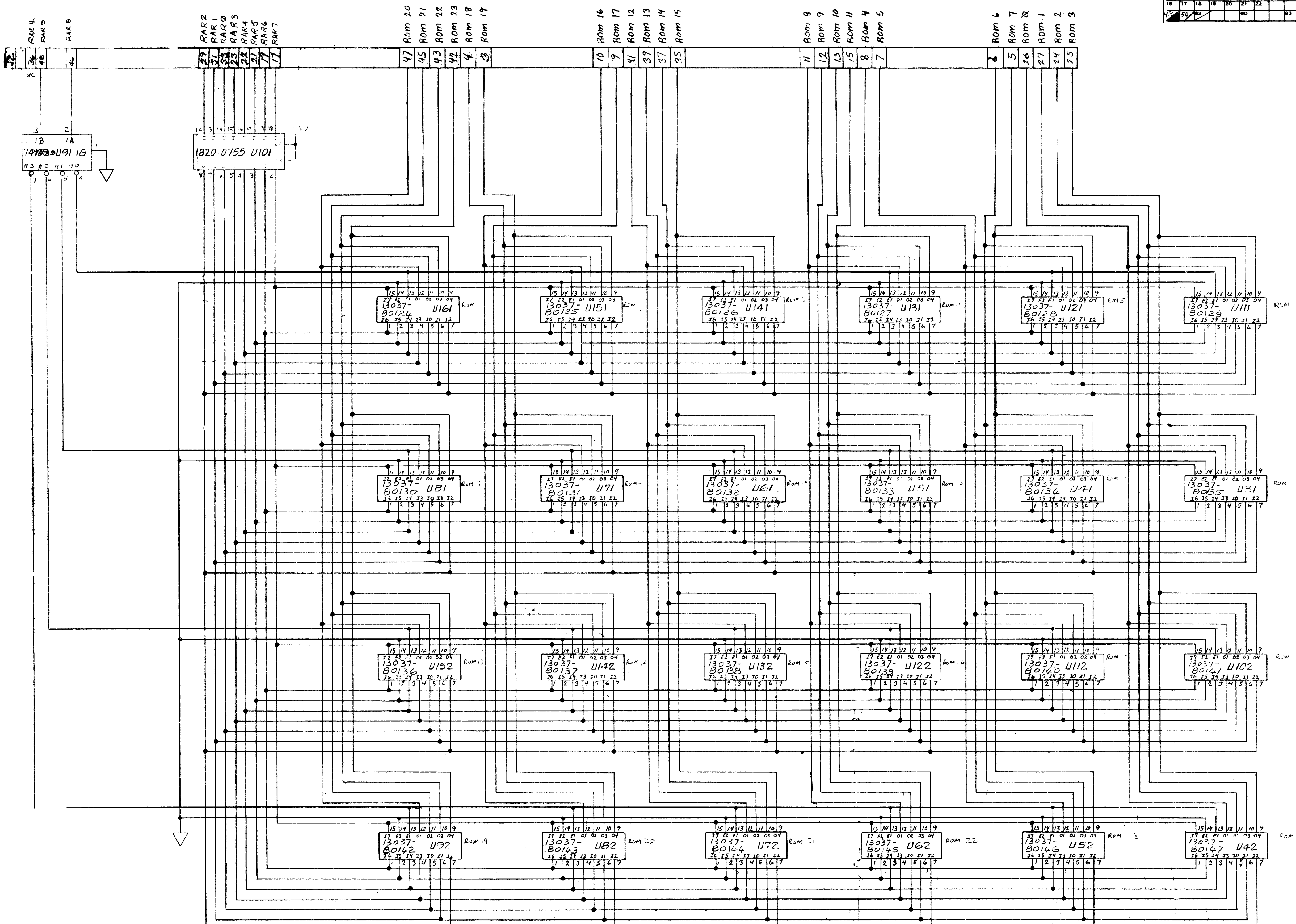


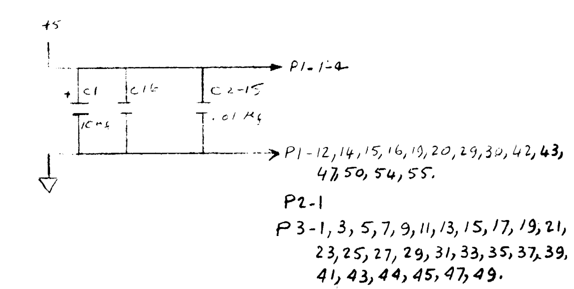
In actuality,

BIT DISP = [1591 X (# C SHIFTS) + 689 X (# P SHIFTS)] MOD 2279

Because there are 2208 bits in a sector if the sync field is excluded, 71 bits of the zero sync field must be shifted in to accommodate the 2279 bit modular arithmetic. Note that these 71 zero bits do not affect the initially cleared state of the registers until the actual data (beginning with the sync word) is shifted through. However, the required bit displacement calculation surpasses the capabilities of the 16 bit arithmetic logic, so the firmware employs a modified algorithm to keep intermediate calculations within the limits of the microprocessor logic.

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED PCOA 22-3486	JR	1/22/
B	PCO-22-3661-CIRCUIT CHG. DATE CODE SAME (1627)	JR	12/15/
C	PCO-22-3718 ROM PART NO. WERE 13037-80100 / -80123, DATE CODE IS 1650	JD/JR	1/3/77





ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC

DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED.
DIMENSIONS ARE IN INCHES
TOLERANCES .XX ± .02 .XXX ± .005

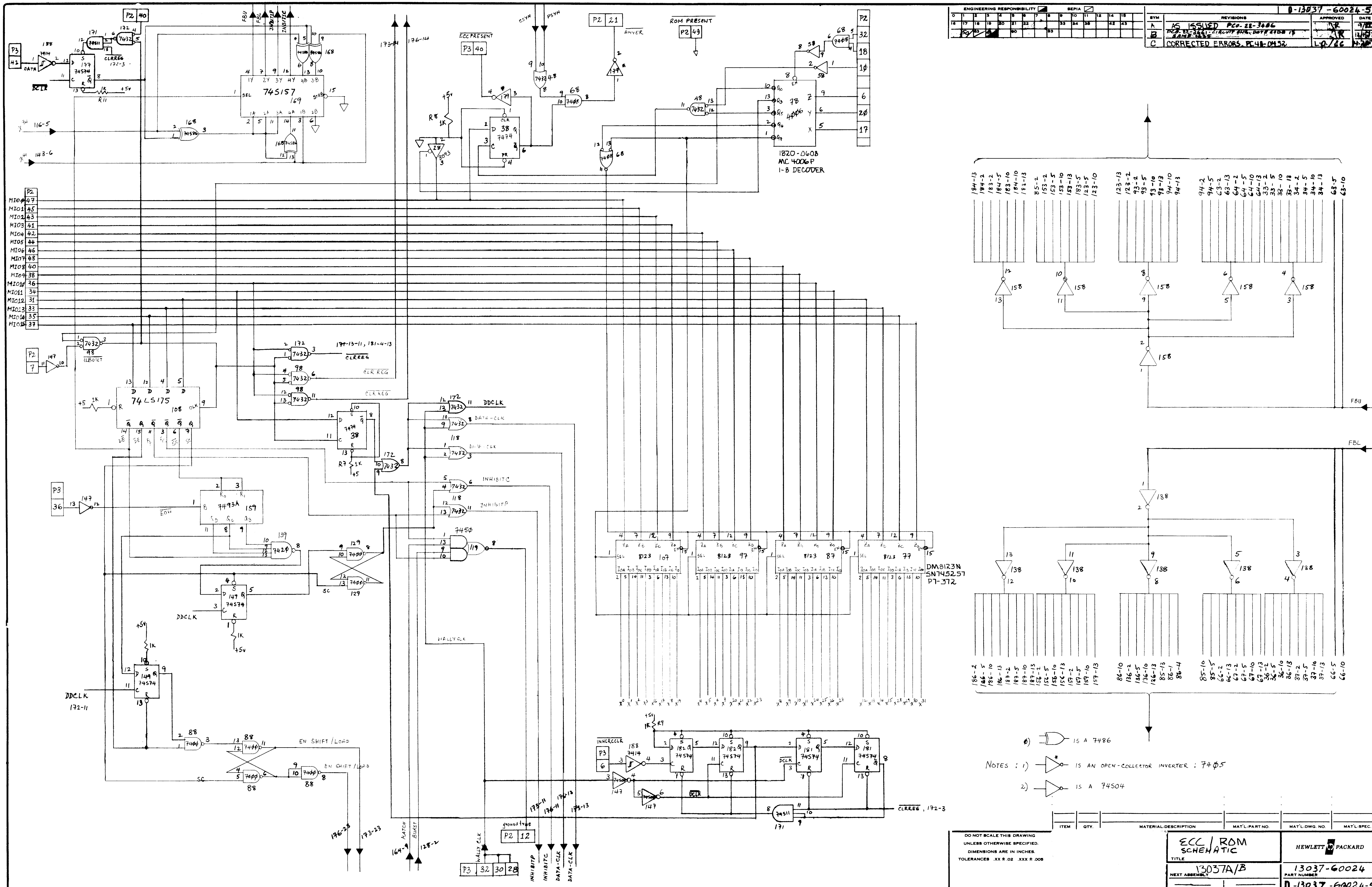
TITLE: ECC / ROM SCHEMATIC

NEXT ASSEMBLY: 13037 A/B

HEWLETT PACKARD

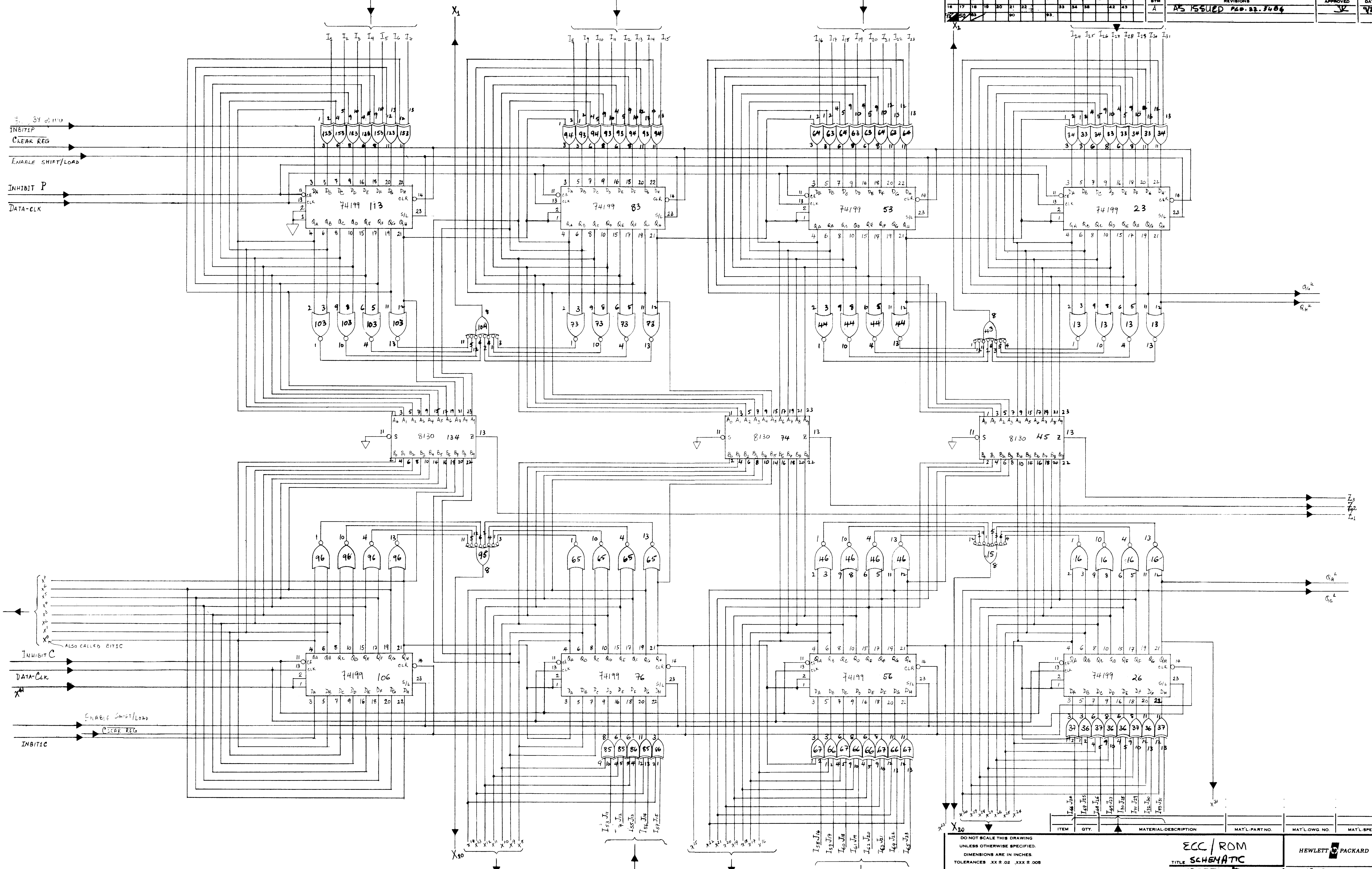
PART NUMBER: 13037-60024

D-13037-6[illegible]4-5


ENGINEERING RESPONSIBILITY
SEPIA
REVISIONS
APPROVED
DATE
A AS ISSUED PCO-22-3486
C CORRECTED ERRORS. PC48-0432
D-13037-60024-53
74S157
74LS175
7493A
7474
MC 4006P
1-8 DECODER
1820-0608
DM8123N
SN74S257
8123
ECC PRESENT
ROM PRESENT
CLRREG
DDCLK
DATA-CLK
INHIBITC
INHIBITP
EN SHIFT/LOAD
WALLY CLK
MATCH
BURST
INHERCCLK
FBU
FBL
NOTES : 1) IS AN OPEN-COLLECTOR INVERTER : 7405
2) IS A 74S04
IS A 7486
DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED.
DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005
ITEM
QTY.
MATERIAL-DESCRIPTION
MAT'L-PART NO.
MAT'L-DWG. NO.
MAT'L-SPEC.
ECC/ROM SCHEMATIC
TITLE
13037A/B
NEXT ASSEMBLY
FINISH
SCALE
HEWLETT PACKARD
13037-60024
PART NUMBER
D-13037-60024-53

D-13037-60024-S1

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED PCO. 22. 3486	[illegible]	[illegible]



ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.

DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED.
DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005

TITLE ECC / ROM SCHEMATIC

NEXT ASSEMBLY 13037A/B

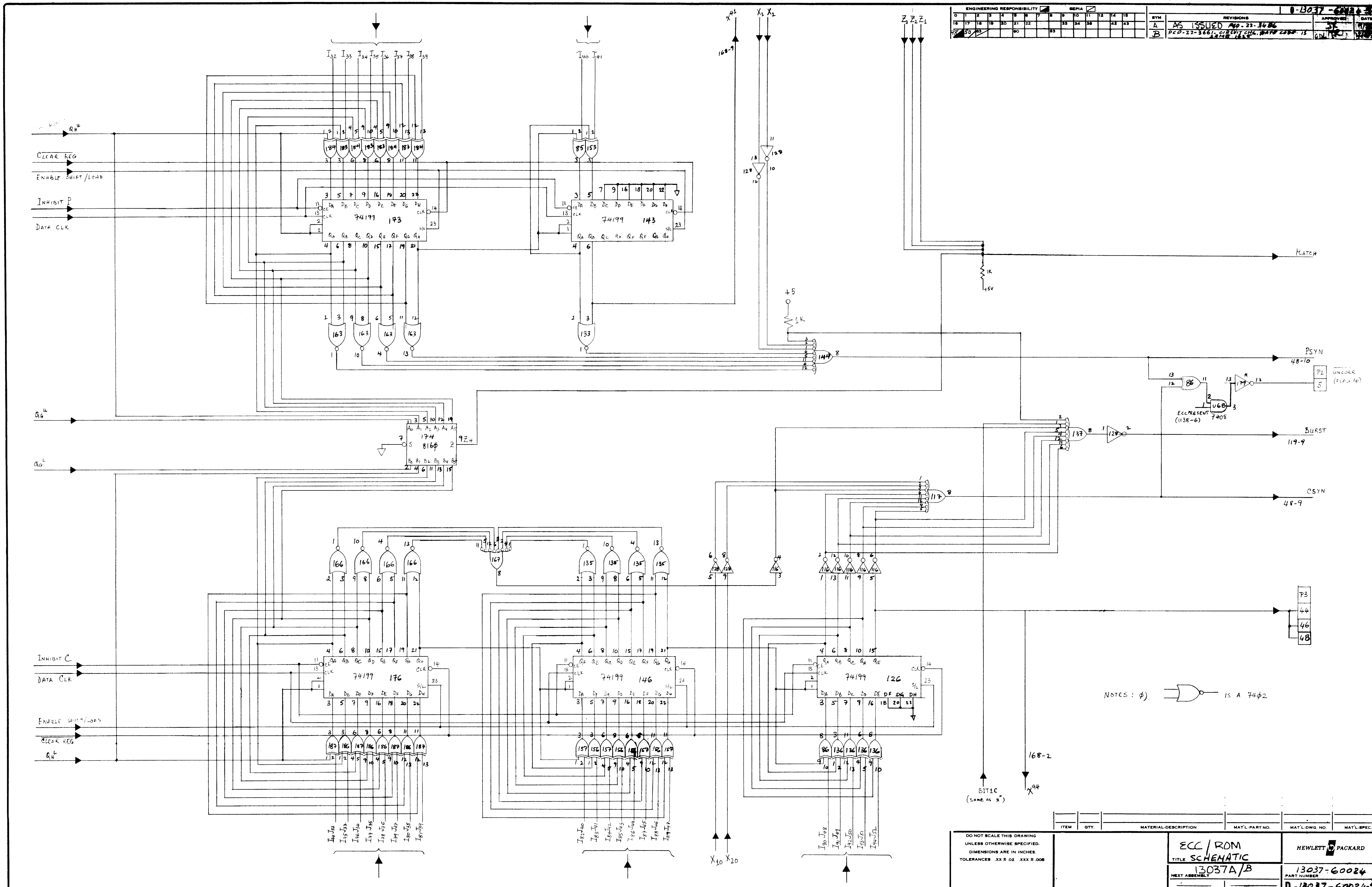
HEWLETT PACKARD

PART NUMBER 13037-60024

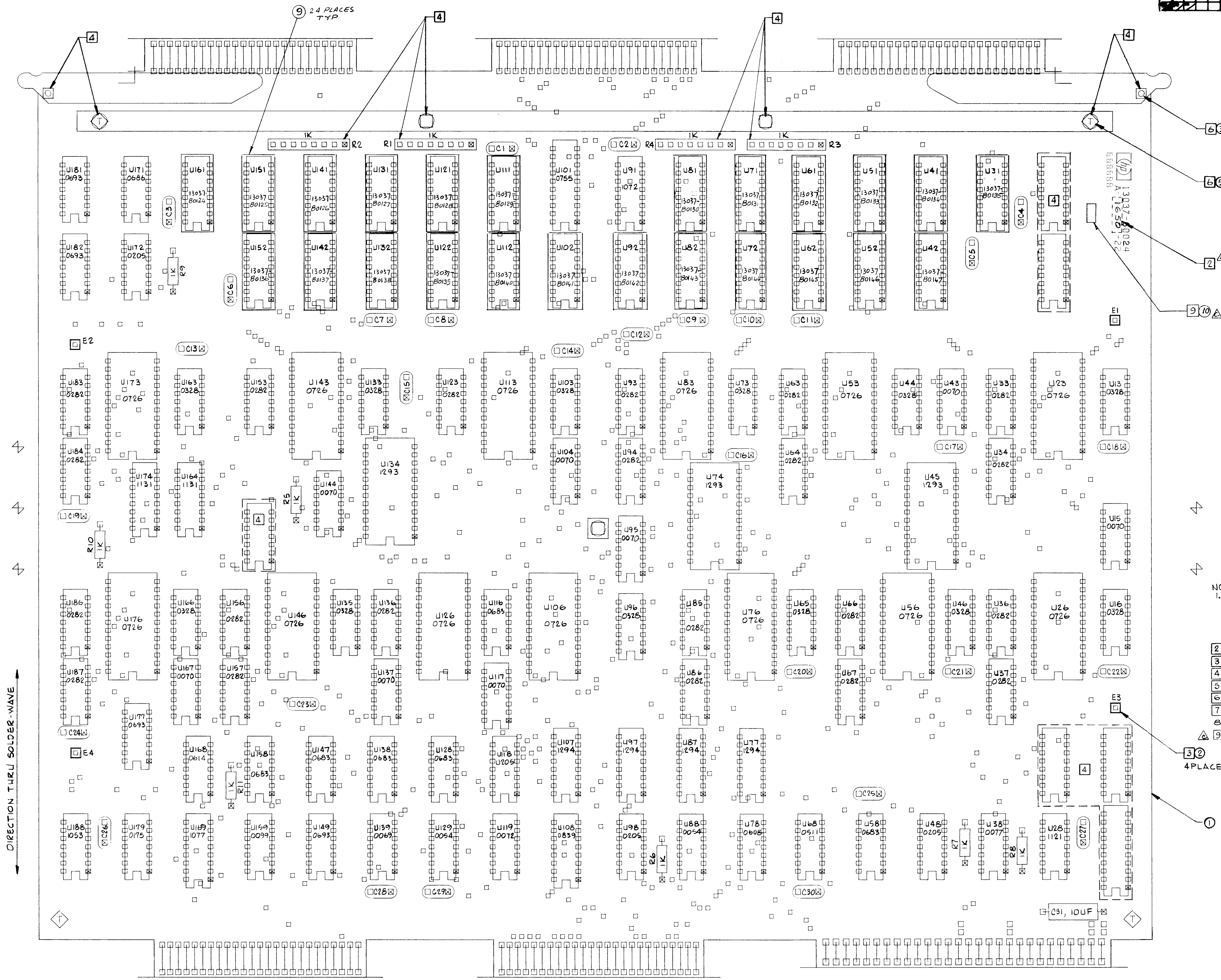
FINISH

SCALE

D-13037-60024-S1


ECC / ROM SCHEMATIC
13037A/B
13037-60024
D-13037-60024-52
HEWLETT PACKARD
DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED.
DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005
ITEM
QTY.
MATERIAL-DESCRIPTION
MAT'L-PART NO.
MAT'L-DWG. NO.
MAT'L-SPEC.
NEXT ASSEMBLY
PART NUMBER
FINISH
SCALE
ENGINEERING RESPONSIBILITY
SEPIA
SYM
REVISIONS
APPROVED
DATE
A AS ISSUED
B
CLEAR REG
ENABLE SHIFT/LOAD
INHIBIT P
DATA CLK
INHIBIT C
DATA CLK
ENABLE SHIFT/LOAD
CLEAR REG
74199
173
143
176
146
126
8160
174
MATCH
PSYN
48-10
UNCORR
BURST
119-9
CSYN
48-9
ECC PRESENT
7408
NOTES: ϕ) IS A 74ϕ2
168-2
168-9
BIT1C
1K
+5
+5V
P2
P3
44
46
48

ENGINEERING RESPONSIBILITY	SEPIA			
LTR	REVISIONS		APPROVED	DATE
A	AS ISSUED PCO-22-[illegible]		[illegible]	[illegible]
B	PCO-22-3717, ECO'S [illegible] -80143-DATE CODE WAS 1825		GD/[illegible]	[illegible]
C	PCO-22-3720: ADD. NOTE 9 & ITEM 10		GD/JR	1/18/77



NOTES:
1. UNLESS OTHERWISE SPECIFIED:
   ALL RESISTANCE IN OHMS
   ALL RESISTORS 1K, 1%
   ALL CAPACITANCE IN MICROFARADS
   ALL CAPACITORS .01UF CERAMIC DISCS
   ALL IC'S 1820-

2. STAMP DATE CODE (OPER 33)
3. INSTALL ITEM ② 4 PLACES (OPER 35)
4. MASK AS INDICATED PRIOR TO LOADING
5. USE SUPPORT FIXTURE DURING WAVE SOLDER
6. INSTALL ITEMS ③ THRU ⑥ IN TOUCH-UP
7. INSTALL ITEM ⑧ PER DWG D-5951-4413-1
8. SCHEMATIC D-13037-60024-51-54
9. ITEM ⑩ STAMPED WITH MONTH & YEAR & INSTALLED AFTER FINAL TEST.

ITEM	QTY.	MATERIAL-DESCRIPTION	MATL-PART NO.	MATL-DWG. NO.	MATL-SPEC.
10	1	BLANK LABEL	7120-5480		
9	24	SOCKET 16 PIN	1200-0482		
8	1	LABEL	7120-5738		
7	1	P.C. BRACE	5040-6058		
6	4	SPACER	5020-7318		
5	5	SCREW	0624-0077		
4	2	EXTRACTOR	5040-6009		
3	2	PIN	1480-0116		
2	4	TERMINAL E1-E4	0360-0294		
1	1	BOARD, ETCHED	13037-80004		

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED.
DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005

TITLE: ECC/ROM ASSY PERM. PRINT

HEWLETT PACKARD

NEXT ASSEMBLY: 12995A-001, 13037A/B 13037M

PART NUMBER: 13037-60024

FINISH

SCALE 2/1

F-13037-60024-1

SHEET 1 OF 1

CONTENTS

21XX INTERFACE PCA

The following describes the operation of the 13037-60023 Disc Interface PCA. The schematic diagrams for this assembly are numbered C-13037-60023-51, 52, and 53; these should be referenced to fully understand this circuit description.

1.0 BUS INTERFACE AND DECODE LOGIC

(These functions are shown in schematic diagram page 13037-60023-52.

1.1 IBUS CHANNEL HARDWARE

The 16-bit bidirectional Interface Bus (IBUS) connecting the Interface PCA to the disc controller is shown at the bottom right of the page. The bus is driven by three-state logic and transmits data with ground-true coding (0 volts = logical 1). Four inverting bus transceiver IC's (U55, U65, U94, and U95) connect the IBUS to the interface data paths.

In the Read mode, controller data on the IBUS is received by these devices and placed on the transceiver OUTBUS, to be sent to the Select Code Comparator, Busy Bit FF, and the FIFO buffer. In the Write mode, data from the FIFO is taken from the transceiver INBUS and sent over the IBUS to the controller. The OUTBUS always contains IBUS data; with the IF-SELECTED line true and a true ENID signal, the 8T34 bus drivers then drive the IBUS with INBUS information (i.e., at this time U63-8 enables the 8T34 drivers at pins 7 and 9 of each chip). Note that the OUTBUS and INBUS are coded positive true (5 volts = logical 1).

1.2 IFN FUNCTION BUS HARDWARE

The unidirectional ground true four-bit Interface Function Bus (IFN0-3) and signals Interface Function Valid (IFVLD), Interface Clock (IFCLK), and Enable Interface Drivers (ENID) are received by 8T37 inverting gates U31 and U71. These devices have Schmidt-trigger hysteresis inputs for maximum noise rejection. The above signals, as well as the 16 IBUS lines, are terminated with resistor voltage dividers (R4, R7, R8, R11, R12, and R13) which provide the proper termination impedance (on the 13037-60025 add-on Interface PCA, these resistors are omitted). The terminator resistors are powered by the disc controller so that powering down the terminated CPU in a multiple-computer configuration will not affect the buses.

Note that the Enable Interface Receiver (ENIR) signal, while not used by this card, is still terminated. Also, the IFCLK signal must be terminated by one-half of the normal impedance (two dividers in parallel) for proper matching.

1.2.1 FUNCTION VALIDATION

The function placed on the IFN bus by the controller is decoded by U41. Functions are validated by the IFVLD signal which is used to enable the decoder via pins U41-18 and 19. The outputs of the decoder are normally high and will go low when the particular function is decoded. Note that this low state will persist only as long as the IFVLD signal is true (typically

600 nsec, though it can be as long as 350 μs; only during setting of the interrupt is IFVLD as short as 400 nanoseconds, due to the fact that we are interested only in the trailing edge to latch in this interrupt signal at the interface).

1.2.2 INTERFACE SELECTION AND DESELECTION

Since up to eight CPU's can be connected to one disc controller, most interface functions will not be executed unless the interface is selected (selection is the process by which the controller tells the interface whether it is to respond to an Interface Function). Flip-flop U72-4 indicates the state of interface selection (a "one" indicates the unselected state and a "zero" indicates selection). Note that this flip-flop enables all U41 decoded IFN bus instructions except those that select (IFSEL) and deselect (DSCIF) the interface.

The interface is deselected at CPU power-on by the backplane signal PON (U72-2), or by the reception of a Disconnect Interface (DSCIF) Function decoded by U41-10 and applied to U72-3.

To select an interface, the controller puts the binary address (0 – 7) on the three lowest bits of the IBUS and issues a Select Interface (IFSEL) on the IFN bus. There are three address selection jumpers on the PCA that determine the select address to which the card will respond. When the IFSEL function is decoded by U41-4, comparator U44 checks for a match between the jumper value and the controller's select address. If a match occurs, the "=OUT" pin (U44-6) will go high, placing the Select FF in the Selected state; this state is indicated by the Select LED CR1, which is driven by gates U32-1 and U32-13.

1.2.3 INTERFACE BUSY BIT

The Busy Bit FF at the upper right corner of the page is used by the controller to indicate to the CPU that a command is currently being executed. The BUSY signal (generated by U41-16) will clock the lowest bit of the IBUS into this FF. Note that when the interface is unselected, the FF is always cleared (U62-1).

1.2.4 FLAG FUNCTIONS

The function STINT, STDFL, and DVEND are logically OR'ed by U21-12 and conditioned by SELECTED (U12-8) to produce the SETFLAG signal, used by the CPU Flag Logic of the board (see section 3.4).

1.2.5 DATA DIRECTION CONTROL

The READ and WRITE signals developed by U51-4 and U51-1 are used to control the direction of data through the FIFO buffer (section 2.3). READ and WRSRQEN (U51-10) are used by the SRQ logic to request DMA transfer cycles (section 2.8).

1.2.6 IFCLK PULSE DEBOUNCING

In order to prevent ringing on the IFCLK signal from entering (or removing) a data word to (or from) the FIFO more than once, the IFCLK Debouncer circuit is included (shown at the upper left corner of the page). Quad flip-flop U102 is wired as a four-bit shift register; it is clocked by the 10 MHz on-card oscillator (section 2.5), and cleared at power on by the POPIO signal.

When an IFCLK pulse is received, FF U72-9 is set. This "one" is shifted four times by U102 (taking 400 nsec); when it reaches the fourth stage, FF U72 is reset by U102-14 (the "one" complemented), and further IFCLK transitions are prevented from setting the FF again by gate U63-11. (Note that IFCLK pulses validate individual IBUS word transfers.)

When the "zero" at U72-9 reaches the fourth stage (another 400 nsec), the debounce circuitry is again ready to accept another IFCLK pulse. Thus the first transition of IFCLK will produce a debounced IFCLK (DBIFCLK) that is 400 nsec long. This signal is then used to enter or remove data from the FIFO buffer (section 2.6). Note that almost all IFCLK signals are 400 nsec in duration; the exceptions are as follows: retry counter sets during cold load read and set file mask instructions, and controller busy bit sets during the setup or BSYST subroutines (each time these signals are 200 nsec). Note that these 200 nsec signals will not affect the FIFO.

RC filter R9-C12, with a delay of about 50 nsec, prevents random noise pulses that might appear on IFCLK from triggering the debounce circuitry. Since IFCLK is unsynchronized to the 10 MHz clock signal, the output of the debouncer could be skewed as much as 100 nsec from the input. Added to the 50 nsec RC filter delay, this means that DBIFCLK could trail IFCLK by as much as 150 nsec. This skew, however, will cause no problems in the transfer of data due to the long data hold time (greater than one microsecond) of the 13037 Controller.

2.0 FIRST-IN FIRST-OUT (FIFO) BUFFER

(The FIFO buffer and related logic is shown on schematic page 13037-60023-53.)

2.1 FIFO HARDWARE AND CONTROL

All data and commands exchanged by the disc controller and the interface pass through the FIFO buffer, shown at the right side of the schematic. Comprised of LSI IC's U45, U84, U75, and U105, this buffer can store up to 16 words. A four-bit slice of all 16-bit words is stored in each of the four FIFO IC's.

Three signals control the operation of the FIFO. Master Reset (MR) on pin 11 will clear all internal FIFO pointers, control logic and flags. Parallel Load (PL) on pin 2 will cause the word present at the D inputs to be loaded. Transfer Out Parallel (TOP) on pin 13 will pull the next word in the buffer to the outputs (if no new word is in the FIFO, the old value will remain).

Two flags are provided by the FIFO to indicate the level of fullness. Output Register Empty (ORE) indicates that no data words are in the buffer, and Input Register full (IRF) signals that no space is available for additional data words.

When the PL pin of the FIFO IC's is pulsed, the data is loaded into an internal input register. The IRF signal will go low to indicate that loading is occurring. IRF is passed from each FIFO IC to the next via the IES input (pin 9). When IRF of U105 goes low, this indicates that all four of the input registers are loaded. This signal then feeds Transfer To Stack (TTS, pin 10) of each FIFO IC, "dumping" the word into the fall-through stack to the first available level. If the stack is not full, IRF will then return to the high state.

In a similar manner, the ORE signal is cascaded between FIFO IC's via the OES input (pin 15). When ORE of U105 goes high, this indicates that all four IC's have valid data at their outputs.

2.2 DTRDY FLAG GENERATION

The ORE signal is used to drive the Data Ready (DTRDY) flag to the controller, indicating that data is valid to the FIFO. In all, five flags are driven by tri-state driver U82 onto the uni-directional Flag Bus. The outputs of U82 are enabled when the interface is selected. The remaining four flags will be mentioned in later sections.

2.3 FIFO I/O

The 16-bit output of the FIFO is sent to one of two destinations. During a Write operation, FIFO data is sent via the IBUS transceivers to the disc controller. For Read functions the FIFO word is placed on the CPU input bus (IOBI) by driver gates U46, U56, U76, U85, U86, U96, and U106. These backplane drivers are enabled by the computer signal IOISC, derived by the CPU Flag Logic (section 3.2).

Similarly, the input data to the FIFO can come from two sources: the IBUS transceivers during Read; or the CPU output bus (IOBO) during Write. Input multiplexer U54, U64, U74, and U104 select between these sources as directed by the decoded READ signal. These devices also contain data latches, which ensure that the data is stable before a PL pulse is applied to the FIFO. Note that the 16 IOBO signals, since they come from the computer's CTL backplane, require pull-down resistors to the −2 volt supply to ensure a proper logical zero voltage level.

2.4 INTERFACE CLOCK

The on-card 10 MHz oscillator is shown at the upper left area of the schematic. This crystal-controlled signal is used to clock the IFCLK Debouncer, the FIFO Synchronizer-Sequencer, and the FIFO pointer.

2.5 FIFO SYNCRHONIZATION AND SEQUENCING

The FIFO Synchronizer-Sequencer is used to control the loading and unloading of the FIFO buffer. Hex flip-flop U92 is connected to form two separate shift registers; the two-bit register on the left generates the TOP signal for unloading, and the three-bit register on the right generates PL for loading.

When the computer inputs a word from the FIFO (U83-4) or the controller sends an IFCLK in the Write mode (U83-8), gate U83-6 will send a "one" to shift register input U92-13. After the trailing edge of this input, gate U81-4 will generate a TOP pulse to remove a word from the FIFO and decrement the FIFO pointer.

Similarly, when the computer enters a word into the FIFO (U83-1) or the controller sends an IFCLK in the Read mode (U83-11), gate U83-3 will load the data into the Input Register and apply a "one" to shift register input U92-3. Some time (100 − 200 nsec) after the trailing edge of this input, gate U81-1 generates a PL pulse to load the FIFO and increment the pointer (the added delay for loading is necessary to ensure that the data has settled at the outputs of the input register before loading the FIFO).

When the computer and controller are transferring data (since the two devices are unsynchronized), it is quite possible that one device will try to load a word into the FIFO at the same time that the other is removing one. In this case the FIFO Synchronizer-Sequencer will generate a PL pulse and a TOP pulse at the same instant. The LSI FIFO devices are able to handle this situation with no resulting loss of data. This condition is also automatically handled by the FIFO pointer.

2.6 FIFO POINTER LOGIC

Shift register U101 functions as a pointer to indicate the number of words contained in the FIFO at any time. The register is configured as a Johnson ring counter, which has the following count sequence:

COUNT	QA	QB	QC	QD
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1

The pointer is reset to zero whenever the FIFO is cleared by a Master Reset pulse. The 10 MHz oscillator continually clocks the register; mode inputs S0 and S1 determine the direction of data shift. In the quiescent case (no loading or unloading of the FIFO), both mode inputs are low and thus no shifting occurs. When a PL pulse is generated by U81-1, S0 goes high, causing the register to shift to the right, "increasing" the count. Similarly, a TOP pulse will set S1 high and a left shift "decrement" will occur. The PL and TOP pulses are present for only one clock period (100 nsec), so that only one count will be recorded per FIFO transaction.

If both PL and TOP pulses occur simultaneously, both mode inputs will be high. In this case, the register will perform a parallel load function. Since the inputs have been connected to the corresponding outputs, the present count value will be reloaded into the register. Thus a simultaneous loading and unloading of the FIFO will cause the pointer to be unchanged, which is exactly the desired result.

Two values of the counter are decoded for other circuit operation. Gate U81-10 generates the FIFO0 signal to indicate that the FIFO is empty, and U81-13 generates FIFO1 to signal that one word is present.

Note: The ORE signal can also indicate that the FIFO is empty. However, because this signal momentarily pulses every time that something is removed from the FIFO, gate U81-10 was required to achieve steady-state signal.

2.7 OVERRUN DETECTION

The Overrun Detector determines when data has been lost due to overflowing or underflowing of the FIFO. And-or-invert gate U73-6 is used to detect this condition, which is stored in the Overrun FF U62-8. When the interface is neither in the Read nor Write mode, gate U63-6 clears the overrun condition. If the FIFO Sequencer tries to load a word (PL, U73-4) when the FIFO is full (IRF, U93-6), or remove a word (TOP, U73-4) when the buffer is empty (ORE, U93-8), gate U73-6 will generate an UNDER/OVERFLOW pulse. If this occurs before the end of the block transfer (EOD1, U42-2), gate U42-1 will set the Overrun FF. Flag Bus driver U82-9 then forwards this condition to the controller.

2.8 SERVICE REQUEST GENERATION

The SRQ circuitry generates a Service Request to the CPU, telling the DMA channel to transfer a word to or from the interface. The SRQ flip-flop (U43-8) is clocked by CPU backplane signal ENF, which occurs at phase T2 of every instruction cycle.

2.8.1 READ MODE SRQ GENERATION

Gate U73-8 determines when an SRQ should be generated. In the Read mode, the SRQ logic will try to keep the FIFO empty. Thus whenever the ORE signal (U73-9) indicates that the FIFO is not empty, U73-8 will go low, causing a Service Request to occur. However, if the computer is currently reading the last word in the FIFO, no SRQ should be generated, since at the end of the current CPU read no words will be left. Gate U63-3 detects this condition and inhibits the SRQ signal for the current CPU cycle by clearing the SRQ FF (U43-10).

2.8.2 WRITE MODE SRQ GENERATION

In the Write mode the SRQ logic tries to keep the FIFO stocked with words for the controller. Although the FIFO can hold up to 16 words, the Write fullness level has been limited to five to prevent unnecessary hogging of CPU memory cycles. Gate U73-13 uses the rightmost bit of the FIFO pointer to determine when four words are in the buffer; when this level has been reached, SRQ is shut off. Since SRQ can be generated as the fourth word is arriving, a fifth word can be obtained, achieving the maximum Write level of five words.

3.0 CPU FLAG LOGIC

(These functions are shown in schematic diagram page 13037-60023-51.)

3.1 CHANNEL SELECTION

In the upper left corner of the schematic, gate U25-12 combines CPU backplane signals IOG, LSCL, and LSCM to produce the Channel Select (CHANSEL) signal. CHANSEL indicates that the current CPU instruction is addressed to this interface card, and therefore is used to qualify most of the other backplane signals shown on this schematic page (the suffix "SC" is appended to these signals to indicate that they have been qualified; e.g., CLC becomes CLCSC). Since the backplane signals are driven by CTL logic, pull-down resistors to the −2 volt supply are needed to insure a valid logical-zero level.

3.2 CPU INPUT MODE

The IOI signal indicates that the CPU is executing an input instruction. Flip-flop U43-5 (at the right of the page) conditions IOI with CHANSEL to eliminate backplane skew between these two signals. Gate U35-6 then reconstructs the IOISC signal, which is used to gate a FIFO word onto the CPU input bus.

3.3 COMMAND GENERATION

To send the controller a command, the CPU first sends a Clear Control (CLCSC) to the interface, which will reset the FIFO and set the COMMAND FOLLOWS FF, U33-5. On the following output instruction, IOOSC will cause gate U13-6 to place the CMRDY flip-flop (U23-4) in the "true" state. When the FIFO flag ORE indicates that the command is valid at the FIFO output, gate U42-10 and Flag Bus driver U82-7 will forward the CMRDY flag to the controller. The outputting of the command will also clear the COMMAND FOLLOWS FF by clocking in a zero. The CMRDY FF is cleared by the Power-on Preset signal (POPIO, U23-2), by the controller function Get Command (CLRCMRDY, U23-3), or by the CLRCMRDY signal generated at the BUS and Interface Decode Logic (previoulsy described).

3.4 INTERFACE FLAG GENERATION AND TEST

The FLAG BUFFER flip-flop (U14-9) latches the interface flag condition asynchronously to CPU backplane timing. This FF is reset by signals CLFSC (U26-6) or when an interrupt from the interface is acknowledged by the CPU (U26-3). Gate U26-8 logically OR's these conditions to reset this FF (U14-10). The FLAG BUFFER FF can be set by these signals: POPIO, STFSC, or the controller initiated CSETFLAG. Flip-flop U33-8 detects the trailing edge of the controller command; if the FLAG FF is not currently set, U42-4 will allow the controller to set the FLAG BUFFER FF via gate U25-8.

The contents of the FLAG BUFFER are loaded into the FLAG FF at time T2 by gate U13-11. The FLAG FF is cleared by the backplane signal CLFSC, which also clears the Trailing Edge Detect FF.

When the CPU executes an SFS (Skip on Flag Set) instruction, U34-6 will send the state of the FLAG FF over the backplane SKF signal. An SFC (Skip on Flag Clear) command will allow U34-12 to report the state of the Busy Bit FF (section 1.2.3).

3.5 INTERRUPT REQUEST GENERATION

The CONTROL FF (U14-13) acts as an interrupt mask for the interface card. It is set by a CPU STC instruction (U14-15) and cleared by the CLC instruction (U14-14). The state of the CONTROL FF is sent by Flag Bus driver U82-3 to the controller as the INTOK flag. If true, the controller can generate a subsequent interrupt.

The INT REQ flip-flop (U14-4) is used to generate interrupts to the CPU. It is reset at cycle T2 of each instruction by signal ENF (U14-1). INT REQ will be set at cycle T5 by signal SIR (U35-13) if all of the following conditions are met:

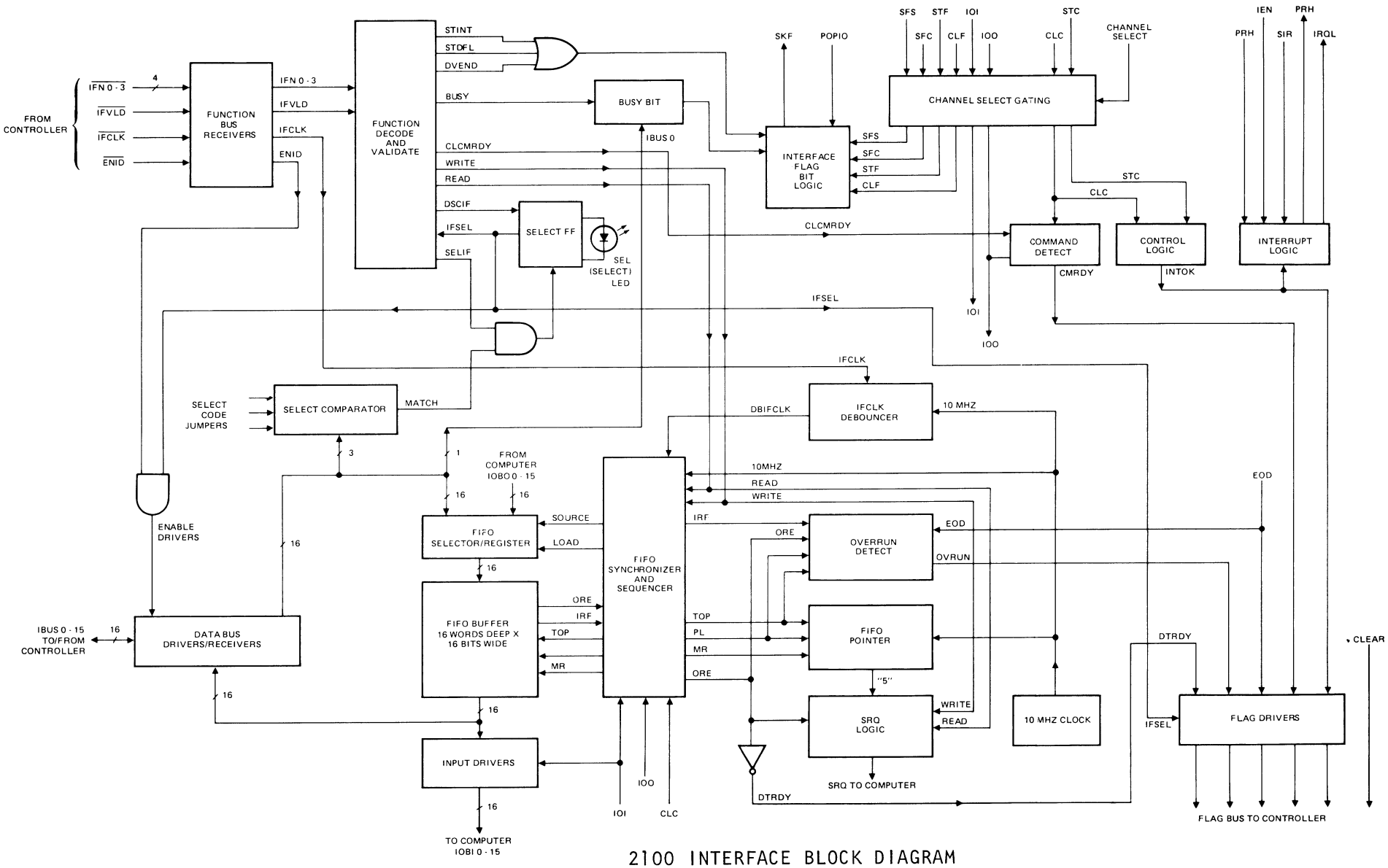
1) The PRH interrupt priority from other interface cards is true (U35-12).
2) The CONTROL FF is set (U25-5).
3) IEN is true (U25-3).
4) The FLAG FF is set (U25-4).
5) The FLAG BUFFER FF is set (U34-9).

The INT REQ signal is sent via backplane driver U15-9 and U16-9 to the CPU.

3.6 END OF TRANSFER

When the computer DMA channel has sent or received the last word of a block transfer, a pulse will appear on the EDT backplane signal. Gate U36-6 conditions this signal with CHANSEL and sets the EOD1 FF, U23-13. In the Read mode, the output of gate U22-3 will be high, allowing gate U22-11 to immediately set the EOD2 FF, U23-9. Flag Bus driver U82-5 forwards the End of Data flag to the controller. In the Write mode, however, there may still be words left in the FIFO to be sent to the controller before EOD can be signaled. Gate U22-6 uses the FIFO0 output of the FIFO pointer to detect when all words have been transferred. If the FIFO was empty at the time that the last word arrived, the FIFO pointer must be given time to update before the EOD flag is set: flip-flop U23-7 (at the right of the page) causes the EOD logic to wait until the next CPU cycle before gate U22-6 decides that the FIFO is empty. When the last word is finally removed by the controller, EOD will then be signaled.

The EOD2 is also set by signal CLCSC, which resets the FIFO and tells the interface that a COMMAND FOLLOWS. EOD2 is cleared whenever the controller fetches a command from the interface (CLRCMRDY, U23-10).



2100 INTERFACE BLOCK DIAGRAM

APPENDIX I
2100 INTERFACE LOGIC AND PROGRAMMING

A. CPU COMMANDS AND THE 2100 INTERFACE

Certain CPU commands work directly with the interface card in the selected channel that the command references. These commands are reproduced in table 6A-1 with the corresponding effects they have on the 2100 interface. Note that no select codes are shown.

Figure 6A-1 represents a simplified version of the 2100 interface's CPU flag logic. Certain CPU backplane signals (at the top of the illustration) correspond directly to entries in table 6A-1 (such signals are STC, CLC, STF, CLF, SFS, and SFC). Other signals are generated as a result of certain of the commands in the table (IOI becomes true for LIA/B and MIA/B instructions, while IOO becomes true for OTA/B instructions). Other signals govern interface channel selection and CPU/interface communications as will be described below.

B. 2100 INTERFACE HARDWARE

Figure 6A-1 is subdivided into numbered sections by broken lines and will be discussed separately. It should be noted that only the CPU flag logic is shown; FIFO logic, service request generation, and function decoding of 13037 bus signals are not shown.

Table 6A-1. CPU Commands to 2100 Interface

COMMAND	FUNCTION
STC	Sets control FF and turns on device on channel specified by S.C.
CLC	Clears control FF and turns off device (aborts data transfer if programmed).
STF	Sets when data transfer finished.
CLF	Clear flag FF on interface card.
SFS	Test if data transfer is completed.
SFC	Test if data transfer still in progress.
LIA/B	Transfer data from I/O interface buffer to A/B register.
MIA/B	Merge data from I/O interface buffer to A/B register.
OTA/B	Transfer data from A/B register to specified interface card.

Before any CPU signals can be recognized by the interface, the channel in which the interface card resides must be selected. The backplane signals necessary to do this (IOG, LSCL, and LSCM) are combined in section (2) to create the qualifying CHANSEL signal. This signal is used internally by the interface to qualify other CPU-generated control signals.

To input or output information through the interface, the FIFO circuitry must be told the fact that a transfer is to occur, and which direction that transfer is to take. This communication takes place via the IOISC (I/O input to CPU, with interface channel selected) and IOOSC (I/O output from CPU, with interface channel selected). These signals are generated directly by IOI and IOO, respectively.

When outputting information through the interface, the CPU must inform the controller of any words that are meant to represent one of the twenty-one 13037 instructions. Section (7) contains the required hardware to generate the necessary CMRDY flag. The CONTROL flip-flop (section (3)) is first cleared by the CLSC signal (generated by a CPU CLC command to the proper channel). The CLSC signal clears the output register of interface FIFO, and also presets the COMMAND FOLLOWS flip-flop of section (7), thus latching the fact that the next IOO from the CPU will correspond to a 13037 instruction. When that IOO is issued, the CMRDY flip-flop becomes reset, and when the FIFO acknowledges the receipt of the command word (ORE — output register empty — goes false), the CMRDY flag goes true. As the controller fetches the command (via the IFGTC function), CLRCMRDY sets the CMRDY flip-flop and causes the CMRDY flag to disappear. Further transfers will not be interpreted as commands unless preceded by a CLC. (Note that POPIO — power on preset I/O — also sets the CMRDY flip-flop to disable the CMRDY flag; additionally, a CRS — clear signal, not shown — will reset the COMMAND FOLLOWS flip-flop.)

Section (4) represents the actual flag logic, and interacts with section (3) during interrupt generation. STINT, DVEND, and STDFL generated by the 13037 are OR'ed at the interface so as to allow the FLAG DETECT flip-flop of section (4) to latch the occurrence of any of those signals. In order to detect any occurrence of these flags, the CPU must first generate the CLFSC signal (CLF to the proper channel) to reset the FLAG DETECT, FLAG BUFFER, and FLAG flip-flops. When a subsequent flag is detected by the FLAG DETECT flip-flop, the FLAG BUFFER flip-flop becomes set so that the following ENF signal (generated automatically at phase T2 of the CPU cycle) can update the FLAG flip-flop. The flag is then held for later interrupt generation by section (3), or flag test by section (5). Note that the FLAG BUFFER flip-flop is additionally set by POPIO or STFSC (though the latter is used only during diagnostic testing), and is additionally reset whenever a requested interrupt is acknowledged by the CPU IAK signal.

Actual interrupts are allowed whenever the CONTROL flip-flop of section (3) is in the true (set) state. Setting the CONTROL flip-flop is accomplished by the CPU's issuance of a STC instruction to the interface channel (STCSC). With the CONTROL flip-flop set, the INTOK flag to the 13037 goes true to inform the 13037 that an interrupt can be subsequently generated. Additionally, INTOK is combined with IEN (the CPU-generated interrupt enable signal) to qualify any 13037-generated flag detected by the 2100 interface as being an interrupt. This new signal will be used to set the IRQ (interrupt request) flip-flop whenever the FLAG BUFFER flip-flop is set and the daisy chain priority hardware of section (1) determines that this interface has the highest priority of those interfaces currently requesting an interrupt. The IRQ flip-flop then generates the IRQL and FLGR signals to the CPU.

After interrupt request receipt, the IAK signal from the CPU resets the FLAG BUFFER flip-flop as previously described. The IRQ flip-flop is automatically reset at the appropriate time (and the FLAG flip-flop of section (4) is set with any new flag held by the FLAG BUFFER flip-flop) by the CPU hardware-generated ENF signal.

In cases where flags are not to be interpreted as interrupts (i.e., during transfers of parameters as previously described), the CPU does not set the CONTROL flip-flop (recall that a CLCSC signal resets this flip-flop as the initial 13037 command is sent). Instead, the CPU monitors the condition of section (4)'s FLAG flip-flop by issuing a SFS instruction to the proper channel. The AND gate of section (5), when the channel is selected and the SFS instruction has been executed, will gate the contents of the FLAG flip-flop over the SKF (skip flag) line to the CPU. Note, however, that the AND gate of section (6) also can gate the status of the interface

NOT BUSY bit to the CPU over the SKF line. Thus, whenever the CPU executes a SFC instruction to the interface channel, the SKF line will inform the CPU as to the status of this bit (remember that this bit is true whenever the 13037 is currently executing an instruction; it should be noted, however, that because the 13037 accepts and executes most commands extremely fast, the SFC instruction is really only useful for the CPU to determine if a DMA transfer is still in progress). Hence, for the 2100 interface, SFS (skip if flag set) and SFC (skip if flag clear) test different functions, not complementary aspects of the same function.

C. BASIC 2100 INTERFACE PROGRAMMING EXAMPLE

Based on the above discussion of the 2100 interface hardware, it can be seen that certain instruction sequences must be generated by the CPU in order to obtain the expected results. To illustrate some of these sequences, the following example is included to illustrate the way in which a CPU could issue a VERIFY command to the 13037.

LDA	CMD	CPU updates the A and B registers with the VERIFY command (from CPU register CMD) and the sector count (from CPU register SCTC)
LDB	SCTC	
CLC,C		sets up first word transferred as being a 13037 instruction and clears the interface flag logic
OTA		outputs VERIFY command to interface
SFS		waits for 13037 to accept the command and subsequently request the sector count by issuing a STDFL command to the interface
JMP	*-1	
OTB,C		outputs sector count to interface and clears the previously detected flag
SFS		waits for the 13037 to indicate command completion (13037 issues either STINT, DVEND, or STDFL when done)
JMP	*-1	

Two important aspects of the above program segment should be recognized. First, command completion was tested by the second SFS instruction. Although current command execution could be tested by the SFC instuction, some commands are executed so quickly that only command completion can be tested; thus SFC should be used only to determine the state of any DMA transfers. Second, command completion does not indicate in what state the command had completed; recall that the interface OR's together STDFL (normal completion), DVEND (data transfer error), or STINT (any other error) to produce the single interface flag. Therefore a REQUEST STATUS command should be issued to determine the completion state of any transfer operation.

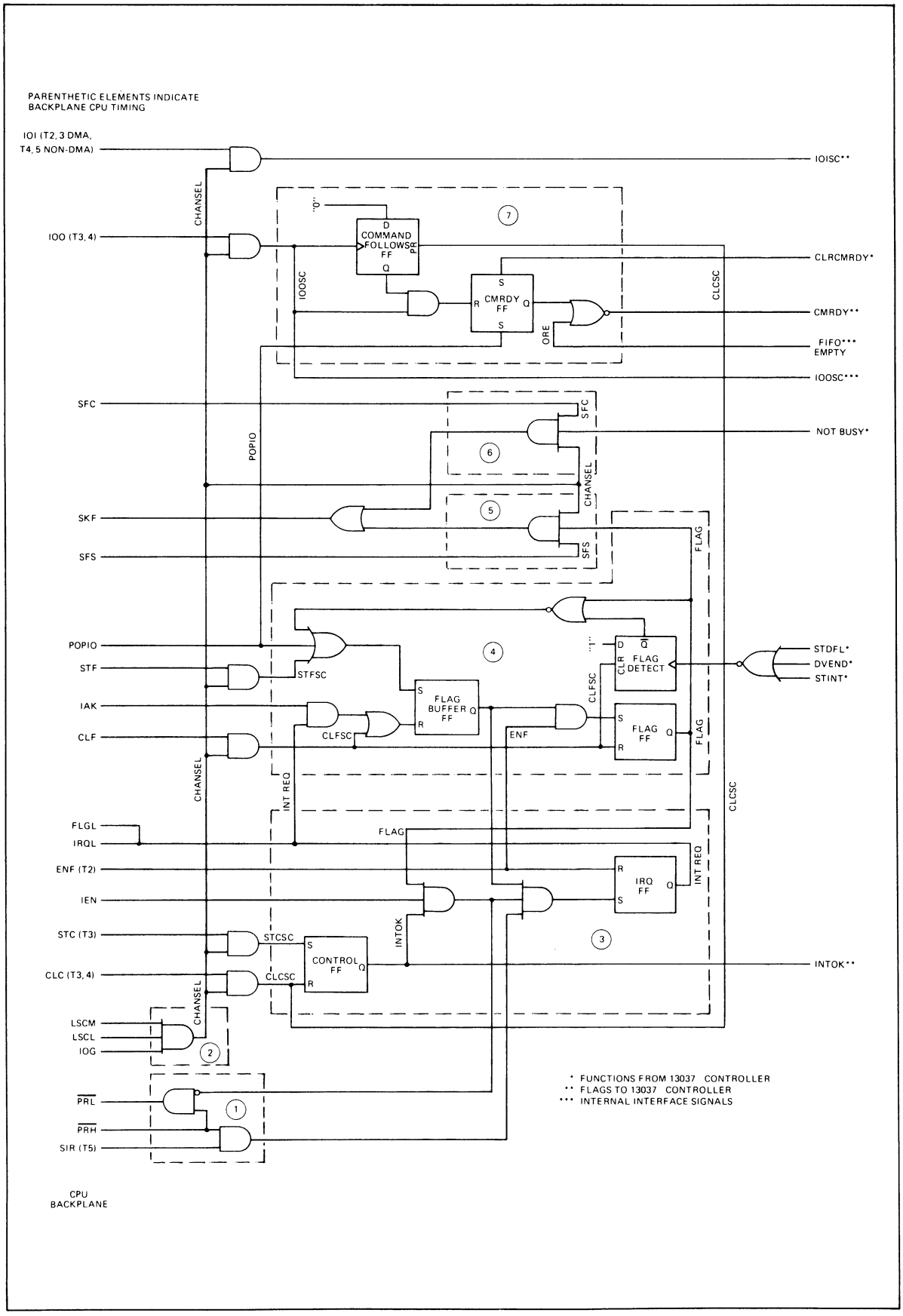


Figure 6A-1. 2100 Interface CPU Flag Logic

ENGINEERING RESPONSIBILITY														
0	1	2	3	4		6		8	9		11	12	14	15
16	17		19		21	22	23	25	29	30	32	33	38	43
45	46	61	63	65	50									

SEPIA

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED	JR	9/10/76

DBIFCLK

IFCLK DEBOUNCER

POFIO

10MHz

74LS175 U102

74279 U72

U63

R9 147

200pf C12

FUNCTION DECODER

74154 U41

(OCTAL) ØØ STINT

Ø1 STDFL

Ø5 DVEND

Ø3 IFSEL

11 DSCLR

13 IFOUT

14 IFIN

15 IFGTC

16 BUSY

U22

SELECTED

U21

CSET FLAG

WRITE

READ

CLRMDRY

U51

U53

WRSRQEN

SEL

CR1 LED

R3 121

7401 U32

8T37 U31

IFCLK IFNØ IFN1 IFN2 IFN3 IFVLD

ENID

8T37 U71

R8 240

R8 560

R12 560

R12 240

C11 10µf

R8 560

R7 240

R7 560

R4 240

R4 560

R12 240

R12 560

R13 240

R13 560

R11 10K

PON

P1 66

.01 C20

IFSELECTED

SELECT FF U72 74279

SELECT CODE COMPARATOR

7485 U44

Busy BIT FF

7474 U62

NOTBUSY

1K R5

+5

OUTBUS 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Positive True Data

INBUS Ø 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

8T34 BUS U55

8T34 BUS U94

8T34 BUS U65

8T34 BUS U95

Ground True Data

J1 16 2 3 B C 7 15 T 23 11 12 13 14 D E F H 19 20 21 22 W X Y Z J1

IFCLK IFNØ IFN1 IFN2 IFN3 IFVLD ENID ENIR +5 FROM CNTRLR

IBUS Ø IBUS 1 IBUS 2 IBUS 3 IBUS 4 IBUS 5 IBUS 6 IBUS 7 IBUS 8 IBUS 9 IBUS 10 IBUS 11 IBUS 12 IBUS 13 IBUS 14 IBUS 15

+5 GND P1 39 40 1,2 85 86

C14 10µf .01µf

E1, E2, E3

C1-7, 9, 10, 13, 15-20, 22-25

1 9 10 17 18 A J K L M N P U V J1

For 13037-60025, omit R4, R7, R8, R11, R13, R13

ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.

TOLERANCES .XX ± .02 .XXX ± .005

21XX-13037 Interface Schematic

TITLE Bus Interface / Decode

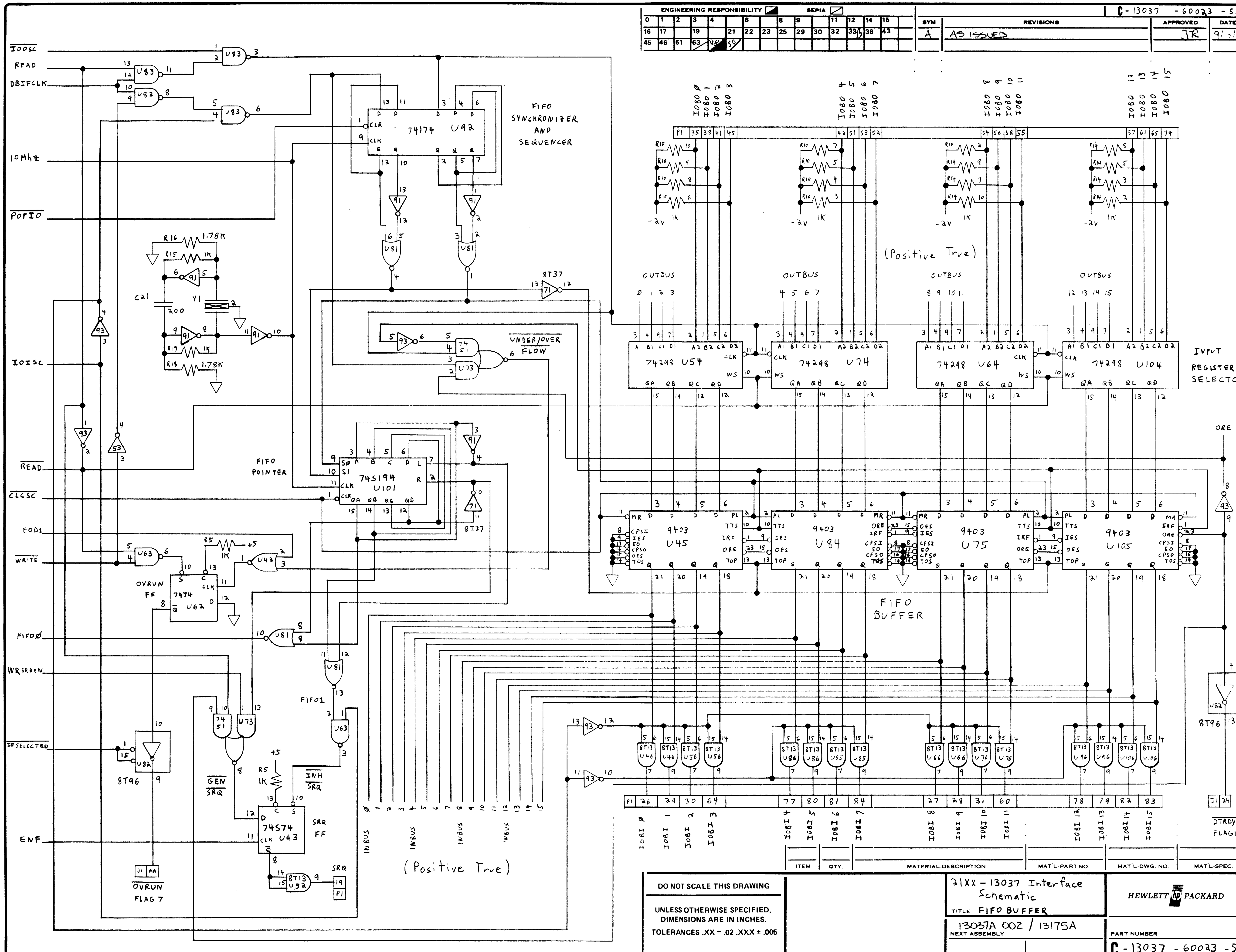
HEWLETT PACKARD

13037A 002 / 13175A

NEXT ASSEMBLY

FINISH SCALE

PART NUMBER C-13037-60023-52

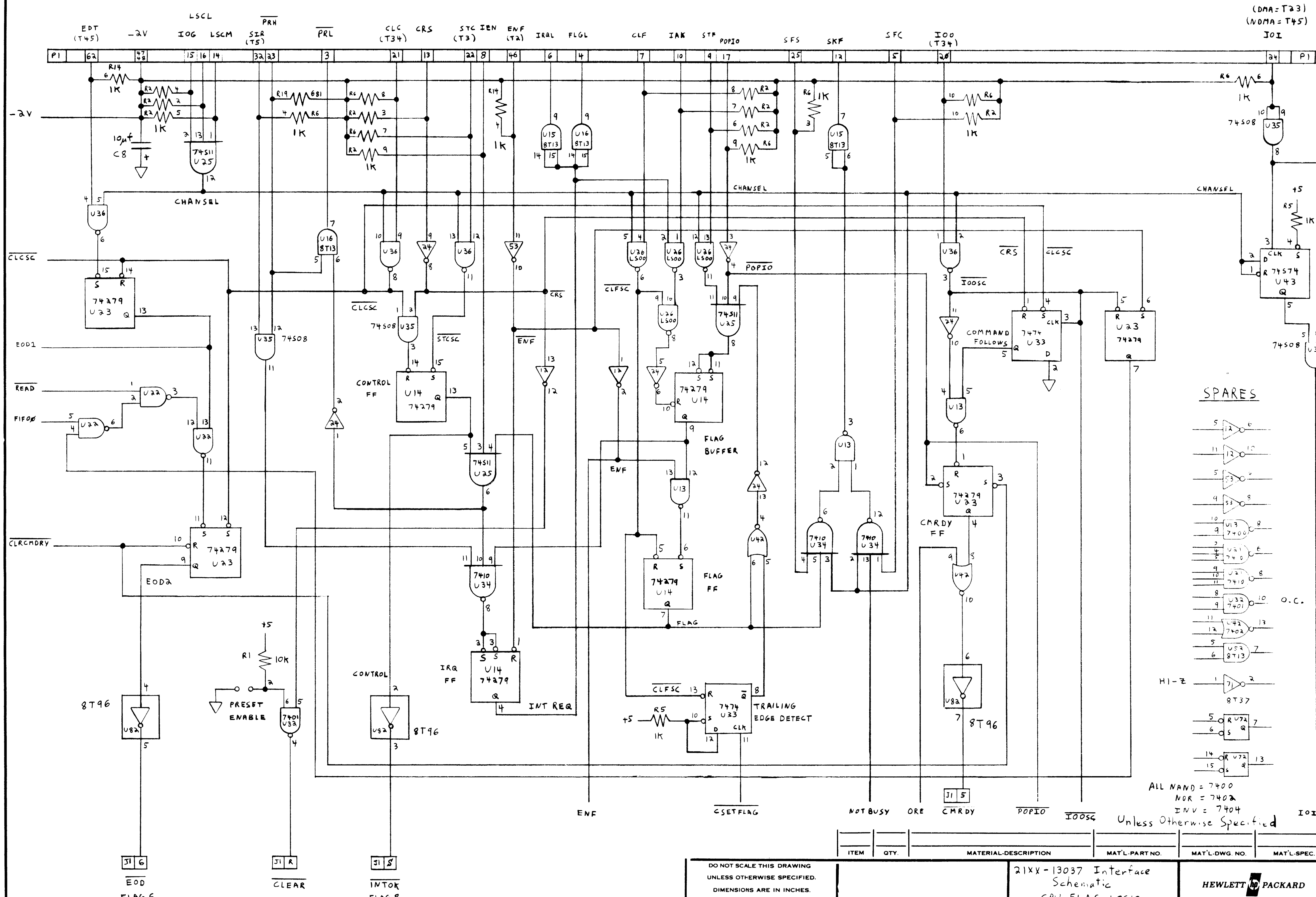


IOOSC
READ
DBIFCLK
10MHz
POPIO
IOISC
READ
CLCSC
EOD1
WRITE
FIFO0
WRSREEN
IFSELECTED
ENF
FIFO SYNCHRONIZER AND SEQUENCER
74174 U92
UNDER/OVER FLOW
FIFO POINTER
74S194 U101
OVRUN FF
7474 U62
GEN SRQ
INH SRQ
74S74 U43
SRQ FF
OVRUN FLAG 7
8T96
8T37
(Positive True)
OUTBUS
74298 U54
74298 U74
74298 U64
74298 U104
INPUT REGISTER/SELECTOR
9403 U45
9403 U84
9403 U75
9403 U105
FIFO BUFFER
INBUS
DTRDY FLAG1
ORE
ENGINEERING RESPONSIBILITY
SEPIA
REVISIONS
APPROVED
DATE
A AS ISSUED
DO NOT SCALE THIS DRAWING
UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005
21XX - 13037 Interface Schematic
TITLE FIFO BUFFER
13037A 002 / 13175A
NEXT ASSEMBLY
FINISH
SCALE
HEWLETT PACKARD
PART NUMBER
C - 13037 - 60023 - 53
ITEM
QTY.
MATERIAL-DESCRIPTION
MAT'L-PART NO.
MAT'L-DWG. NO.
MAT'L-SPEC.

C-13037-60023-51

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED	JTR	9-10-76

ENGINEERING RESPONSIBILITY — SEPIA



ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.

DO NOT SCALE THIS DRAWING UNLESS OTHERWISE SPECIFIED. DIMENSIONS ARE IN INCHES. TOLERANCES .XX ± .02 .XXX ± .005

21XX-13037 Interface Schematic

TITLE CPU FLAG LOGIC

13037A 002 / 13175A

NEXT ASSEMBLY

FINISH — SCALE

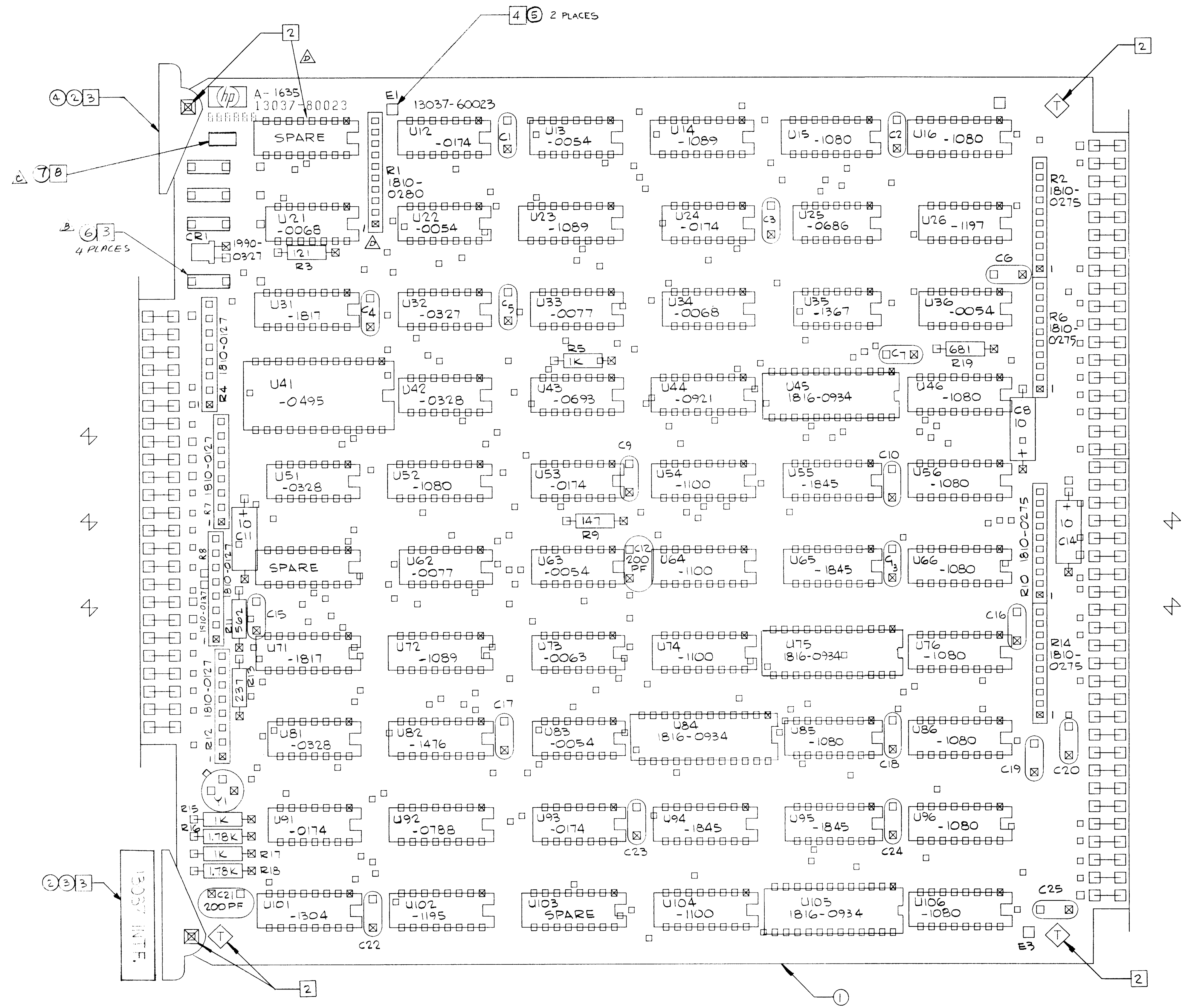
HEWLETT PACKARD

PART NUMBER C-13037-60023-51

D-13037 - 60023 - 1			
SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED	JR	[illegible]/76
B	CHG. NOTE 3, ADD. ITEM 6, CORRECTED POLARITY OF Y1	JR	11/15/76
C	PCO-22-3720-ADD. NOTE 8 & ITEM 7	GD/JR	1/17/77
D	ADDED NO. 1 TO SHOW ORIENTATION OF R1 AND NOTE 2 TO SPARE IC PC48-0033	J Chu / J.CHU	8-31-77

ENGINEERING RESPONSIBILITY / SEPIA

0	1	2	3	4		6		8	9		11	12	14	15
16	17		19		21	22	23	25	29	30	32	32 B	38	43
45	46	61	63	48	50 B									



NOTES:

1. UNLESS OTHERWISE SPECIFIED
   - ALL RESISTANCE IN OHMS
   - ALL RESISTORS 1/8W 1%
   - ALL CAPACITANCE IN UF
   - ALL CAPACITORS .01 CERAMIC DISC
   - ALL ICS ARE 1820-
2. MASK AS INDICATED PRIOR TO LOADING.
3. (B) INSTALL ITEMS ② ③ ④ & ⑥ IN TOUCH-UP
4. INSTALL ITEM ⑤ 2 PLACES (OPER 35)
5. SCHEMATIC C-13037-60023-51, -52, -53
6. MARK ③ PER DWG. A-5950-5668-1
7. (B) CLEAN CONNECTORS WITH ALCOHOL IN TOUCH UP
8. (C) ITEM ⑦ STAMPED WITH MONTH & YEAR & INSTALLED AFTER FINAL TEST

	ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.
C	7	1	BLANK LABEL	7120-5480		
B	6	4	JUMPER PLUG	1258-0124		
	5	2	TERMINAL E1, E3	0360-0294		
	4	1	EXTRACTOR	5040-6001		
	3	1	EXTRACTOR, RED	5040-6065		
	2	2	PIN	1480-0116		
	1	1	BOARD, ETCHED	13037-80023		

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.

TOLERANCES .XX ± .02 .XXX ± .005

SEE CORP. STD. 608

TITLE: 21XX INTERFACE ASSY DWG

PERM. PRINT

HEWLETT PACKARD

NEXT ASSEMBLY: 13037A 002 / 13175A

PART NUMBER: 13037-60023

FINISH: DRI-FILM

SCALE: 2/1

D-13037 -60023 - 1

SHEET 1 OF 1

CONTENTS

POWER REGULATOR PCA

1.0 +5 VOLT REGULATOR

1.1 OVERALL CONCEPT

A simplified block diagram of the +5 regulator is shown in figure 1. Unregulated 12 volt DC power is delivered through a "unity" gain power amplifier to the logic board "load" via the Power Distribution circuit board. A unity gain differential amplifier U3D senses the load voltage right at the distribution board output and provides a "single-ended" replica to the minus input of the 723 comparison amplifier. The 723's high gain error amplifier compares the "load voltage" to a stable 5 volt reference (derived from the 723's internal reference) and drives the Power Amplifier to achieve an error null: the negative feedback thus maintains the load voltage essentially equal to the 5 volt reference source, independent of load current and unregulated +12 and +20 volt power variations.

1.2 VOLTAGE REFERENCE AND VOLTAGE STABILITY

The 723 contains an internal temperature compensated zener circuit which provides a stable 7.15 ±5% reference output at pin 6. This voltage is divided down to 5 volts by the R27, R10, R26 string. Potentiometer R10 provides a limited adjustment range to compensate for the 723 and U3D offset voltages, as well as the tolerance on the reference voltage, divider string, and R22-R24. Resistors R33 and R34 connect the sense amplifier to the 5 volt outputs on the regulator PCA for use with earlier Power Distribution boards that do not have remote sensing (13037-60020). When remote sensing is present, R33 and R34 are effectively shorted to the *remote* nodes by traces on the Power Distribution PCA so that the remote voltage is sensed. Note that adjustment of R10 is made with voltmeter connected to the *sense* test points (+5 and COM) rather than at the output of U3D, to avoid errors introduced by the sense amplifier.

The input characteristics of the 723 and U3D amplifiers and the voltage reference stability are such that the major contribution to voltage drift with temperature comes from thermal coefficient of the divider string and differential amplifier resistors; overall worst case performance is ± 0.04%/°C maximum, with typical performance on the order of ± 0.01%/°C (± 70 mv worst case over the temperature range). Long term stability will also be primarily determined by these resistors; 25 mv/1000 hrs. (0.5%) is the expected upper limit.

Ripple rejection to the reference output from the unregulated +20 volts on the 723 is typically 100 microvolts/volts; C14 attenuates high frequency noise and improves rejection at high frequency.

Resistor R25 provides impedance match for the input bias current of the 723 amplifier, and in combination with C12 provides filtering (82 KC) to reject high frequency noise and extraneous signals which could disturb loop operation.

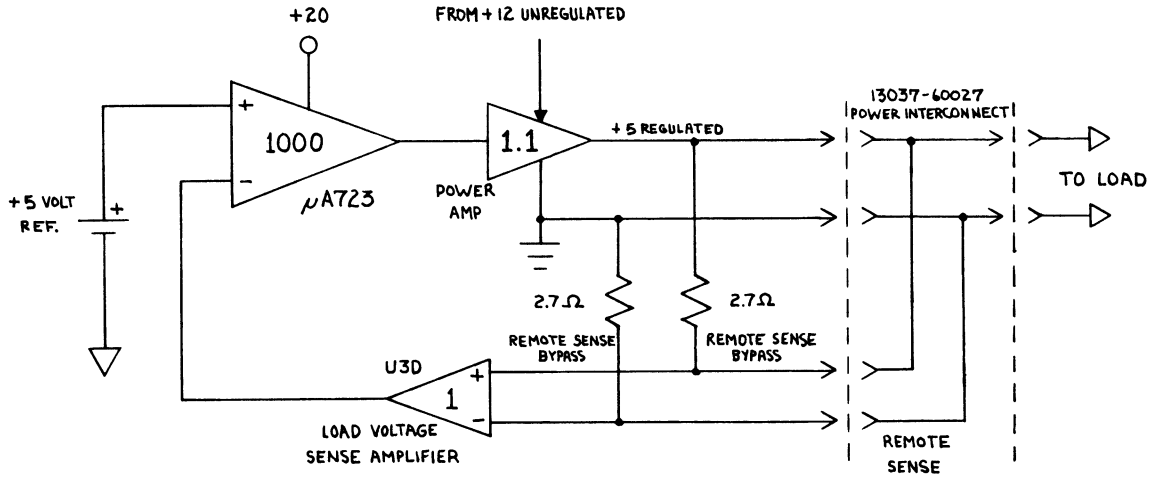
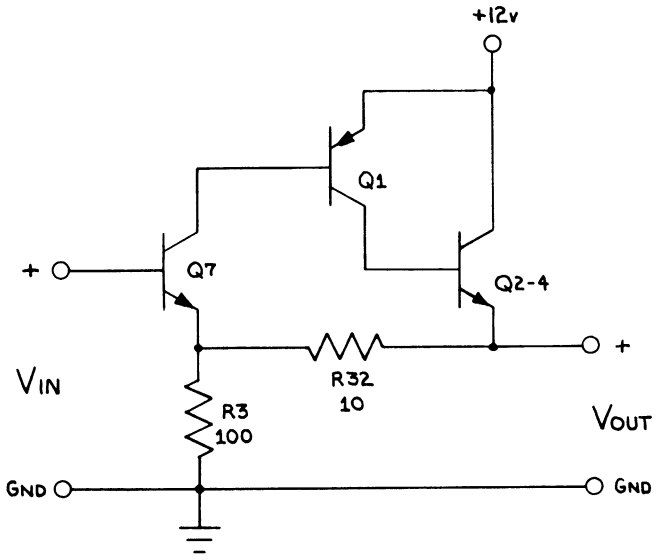


Figure 1

2.0 POWER AMPLIFIER

Transistors Q7, Q1, and Q2-Q4 are the active elements of a power amplifier shown below in simplified form.



Q2-Q4 and Q1 comprise a complimentary Darlington pair drive by Q7; series feedback (R3 and R32) provides a stabilized voltage gain of 1.1, independent of the transistor betas. Since this amplifier is in turn part of the overall negative feedback loop maintaining the load voltage, the local feedback employed here desensitizes the gain of the overall loop with respect to the Q7, Q1, and Q2-Q4 parameters.

Referring to the schematic, Q2, Q3, and Q4 are power transistors driven in parallel by Q1. Resistors R4-R6 give local emitter degeneration to insure that the output current splits equally among the three (as well as to provide current sense information). R40 and R35 set the DC operating point of Q7 and Q1.

3.0 FREQUENCY RESPONSE AND LOOP STABILITY

The closed loop bandwidth of the power amp is nominally 100KC, but is slightly dependent on load current. The 650 microfarad output capacitor C13 plays an important role in determining the magnitude of the high frequency response. Above 3KC, C13 becomes resistive with R5 (approximately 0.1Ω). Minimum DC level impedance is 0.25Ω and as a practical matter will not get much lower than that below 200KC with normal (0.01 – 20 microfarad) bypassing on the logic cards. C13 thus becomes the "load" at the higher frequencies and serves to desensitize the open loop response to variations in actual load impedance. C13 is chosen a wet-slug tantalum which has fairly predictable series resistance characteristics. R9 assures a minimum load current of 185 ma.

The high frequency open loop gain is further controlled by local "Miller" feedback around Q1 via C21 and R36, making the high frequency transimpedance of Q1 independent of beta. In addition, the negative feedback lowers the output impedance of the Q1 stage, which serves to broad-band the Q2-Q4 output transistors. R36 introduces a "zero" at 60 KC and above that, the gain roll-off comes from the transistors (primarily Q2-Q4, which have a low f_t). C20 introduces a "zero" at 300 KC to give additional phase margin.

To summarize, the high frequency open loop gain characteristics of the power amp are determined largely by the fixed elements employed, rather than the transistor parameters; and since the closed loop bandwidth of this stage is much higher than that of the primary voltage control loop, variations in actual gain crossover frequency (attendant with temperature dependence of C13's series resistance) will not affect overall operation of the regulator.

3.1 VOLTAGE CONTROL LOOP

The 723's error amp is a single differential pair (biased at 160 microamps each) driving the double emitter follower pair shown on the schematic; minimum DC voltage gain will be 1000 (B = 18). Zener diode CR5 assures sufficient collector-base bias for the input pair over the temperature range. The combination of R5 and CR4 limit the maximum 723 output current to a safe value: loss of +12 volt power to the output amp causes the 723 to attempt to drive the load through R32, as opposed to AC coupling R32. The clamp transistor (pins 2 and 3) is for over-current protection and is normally cut off.

3.2 FREQUENCY RESPONSE

The closed loop bandwidth of the voltage control loop is 10KC and is again determined by fixed elements. C18 sets the AC gain in the 100 cps – 60KC range (gm of the input pair is relatively fixed) since the power amp and sense amp are broad band. R29 introduces a "zero" at 60KC which roughly matches the pole at 100KC from the power amplifier. C19 provides roll-off above 500KC. The resultant curve thus has a constant 6 db/octave roll-off for at least a decade above and below gain crossover, to yield plenty of phase margin in the face of small variations introduced by component tolerances, input amplifier bias current, and power amplifier frequency response.

The following is true of the measured output impedance vs. frequency from 1KC on up. At the lower frequencies, the decreasing reactance of C13 is offset by the falling loop gain (in both loops) so the curve is fairly flat; above 10KC the impedance rises to 0.1Ω (the series R of C13) as the power amp runs out of loop gain. At frequencies below 1kC the output impedance will be much lower (theoretically $0.02 \times 10^{-3}\Omega$ at 10 CPS with minimum beta transistors) and will be determined largely by the resistance of current traces leading from the remote voltage sensing point.

4.0 OVER CURRENT PROTECTION

The +5 regulator is protected against load faults by an over current detect circuit which latches the 723 output to zero volts for current surges lasting longer than 100 microseconds. When the fault is removed, the +5 output can be restored by turning AC power switch Off and then On.

During that course of the design, current fold-back circuitry was implemented. It was determined, however, that some *transient* fault conditions can result in instantaneous over voltage transients; latching the output to zero was felt to be a safer approach, and therefore subsequently implemented.

4.1 CIRCUIT DETAILS

The current through each output transistor is sensed via R4, R5, R6 voltage drops, and applied to differential amplifier U3A via the R37, R38, R39 summing resistors. The gain setting resistors R45, R46, R47 are chosen to yield 5 volts output for 21 amps output current (the values depart from ideal due to I-R drops on the circuit board and the position of R47 sense lead). The amplifier output is filtered via R44 and C22 (τ = 500 microseconds) and applied to the input of comparator U3B. The reference for this comparator is the 5.0 volts output voltage reference at R10. For input voltages less than 5 volts, the comparator output is low (< 0.5 volt); for inputs above 5.0 volts (> 21.5 amps) the U3B output swings positive, coupling positive feedback to the input via CR12 and R42; the output thus drives itself to the +20 volt supply. Should the input signal return below 5 volts, the output stays latched at +20 volts. Dividers R41 and R28 apply the comparator signal to the 723's clamp transistor; clamping occurs nominally at an 8.5 volt comparator output.

Experimentally, the output current transient following the application of a short circuit is a pulse, peaking at 40 amps and lasting about 100 seconds. C22 was chosen to be the largest value that did not increase the pulse duration.

5.0 OVERVOLTAGE PROTECTION

The regulated +5 volt output is protected against overvoltage conditions which could damage TTL logic. Such conditions could arise from component failure within the regulator, or accidental short circuits while servicing.

A sensing amplifier compares the output voltage to an independent 6.2 volt reference, and triggers an SCR crowbar circuit when overvoltage occurs. The crowbar instantly clamps the output to ground via a power rectifier diode, and in the longer term interrupts the power to the power amplifier by blowing the 25 amp secondary fuse. This latter step is necessary to limit the power dissipation in the crowbar itself, since the currents attendant with collector-base shorts in the output stage can be enormous.

5.1 CIRCUITRY

Transistor Q5 compares the +5 volt output on the regulator board to the 5.62 ±5% reference provided by CR3. R13 and C8 provide filtering for transients lasting less than 1 microsecond. For voltages above 6.2 volts (nominal), Q5 conducts, triggering the triac amplifier SCR2 at about 2 ma. The triac in turn quickly shorts R1 to the +20 volt supply delivering the 100 ma of current necessary to trigger the crowbar SCR1. C7 is for speedup and provides 1.5 amp instantaneous trigger current to assure uniform current flow within the SCR.

Once triggered, the SCR1 anode current builds up rapidly and anode voltage drops to 2 volts. CR15 is now forward biased, and clamps the regulator output at 3.5 volts. R7 protects the SCR by limiting the average current to 70 – 90 amps (depending on line voltage), and the 25 amp fuse blows after 0.6 second (±50%) nominally.

Typical dynamic performance following application of a direct short from +12 to the regulator output is that the overvoltage circuit will respond within 3 microseconds to limit the maximum output voltage rise to 7.0 volts; thus the output voltage is greater than 5.0 volts for less than 10 microseconds total.

6.0 THERMAL DESIGN DATA

The power amp output transistors Q2-Q4 are mounted directly to the head sink using Wakefield 120 Thermal compound. The heat sink runs at 12 volts DC, avoiding the requirement for insulating washers. This saves up to 16°C on maximum junction temperature, and a more reliable thermal contact is established.

The worst case power dissipation for the output transistors occurs at hi-line 60 Hz with the unregulated filter capacitor C17 at its maximum value. Total worst case dissipation at 20 amps is 111 watts, but due to possible current unbalance the power handled by an individual device could be 8% above the average for the three. Also note that at hi-line *50 Hz,* the dissipation is only 1 – 2% lower than 60 Hz; 50 Hz will be the worst case cooling situation if fan output is frequency dependent.

7.0 +12 VOLT UNREGULATED SUPPLY

The +12 volt unregulated supply provides the DC power to the +5 volt regulator power amplifier, which in turn controls the flow of power to the logic board load. The transformer center-tapped secondary voltage is full-wave rectified by power rectifier diodes CR1 and CR2 and filtered by the 80K microfarad (+80 −10%) capacitor C17. Power output to the regulator is taken through the 25 amp fuse.

Rectifier dissipation at 20 amps load is 15 watts each, worst case; allowing 1.7°C/W from junction to heat sink, the maximum allowed heat sink temperature for 170° junction is 140°C at 20 amps.

The power transformer has a 12:43:1 primary-secondary turns ratio, and at 25°C has an equivalent secondary resistance of 13 mΩ for each leg (3 mΩ reflected from primary). Allowing 80° winding temperature rise, transformer I -R dissipation will be 13 watts maximum at 20 amps.

8.0 −20 VOLT UNREGULATED SUPPLY

The −20 volt unregulated supply provides unregulated power to the Device Controller logic board via the Power Distribution board. Nominal load current taken from this supply is 115 ma.

CR16, CR17 and C15, C3 comprise the negative voltage doubler operating from one side of the transformer secondary.

9.0 +20 VOLT UNREGULATED SUPPLY

The +20 volt unregulated supply is used to power the 723 regulator chip, the LM 320 op-amp, and the overvoltage trigger circuit. Nominal load current is 25 ma.

The circuit comprising CR6, CR8, C4, and C6 is identical to that for the −20 volt supply.

10.0 −12 VOLT UNREGULATED SUPPLY

The −12 volt unregulated supply provides power to the −5 volt regulator for distribution to the Device Controller board. Nominal load current is 310 ma. CR7 and CR9 full wave rectify the transformer output and C5 provides filtering.

11.0 −5 VOLT REGULATOR

The −5 volt regulator is an LM 320K T03 package mounted on the heat sink, providing an output of −5 volts ±5%. This device is short-circuit proof and contains internal overtemperature shut down. At 500 ma, maximum allowed heat sink temperature is 115°C, based on T_j maximum of 125°C and 3.8°C/W thermal resistance (0.8°C/W for insulating washer).

12.0 POWER-ON CIRCUIT

In addition to the regulated and unregulated output voltages, the Power Regulator board provides a Power-On logic signal output (PON) to the logic cards to provide initialization during power up/down sequencing.

At power turn-on, an internal time delay keeps PON low until 150 msec (nominal) after the +5 volt output has stabilized. At power turn-off PON goes low when the −5 volt regulator output drops, typically 6 msec before the loss of +5 volts. At very low input line voltage (brown-out) PON goes low whenever the 5 volt regulator output drops more than 0.25 volts below the reference (loss of regulation) or when the −5 volt output drops below 4.0 volts ±0.5; the line voltage at which this occurs depends on the 5 volt DC load current being taken (number of boards in controller), but is typically 92 VAC with a 20 amp load.

12.1 CIRCUIT OPERATION

Differential amplifier U3C compares the +5 volt output (sensed by R14) to a voltage that is 5% less than the 5 volt regulator reference (via the R15 − R48 divider). Pin 8 will be low as long as the 5 volt output exceeds the divided reference. Should the +5 output drop more than 0.25 volt below the reference, pin 8 swings rapidly high to the +20 volt supply, charging C9 through CR10. If the loss of regulation is due to low line, the +5 output will go in and out of regulation on the peaks and valleys of the +12 volt ripple. CR11 connects C9 as an integrating capacitor for negative-going excursions of pin 8, so that pin 8 stays high despite the ripple. R16 provides 0.1 volt hysteresis to assure clean hi-to-low transition at pin 8 as the line voltage is restored.

At power turn-on, the reference voltage comes up before the +5 output, so that U3 pin 8 comes on high. After regulation is achieved, pin 8 ramps to its low state via the C9 integration delay (150 ms typical).

The turn-on delay circuit is interfaced to a TTL logic level by transistor Q6 and R20. R19 and R18 bias the base slightly negative, but R18 turns Q6 on in the event of loss of −5 volts. CR13 and CR14 protect pin 8 from being pulled negative (op-amp requirement).

APPENDIX I
OUTPUT SPECIFICATIONS

+5 Volts regulated	−0 to 20 amps
Voltage accuracy:	±1% — adjustable
Ripple and noise:	less than 1 millivolt RMS
Output impedance:	less than 1 mv/amp
Line regulation:	less than 10 mvolts change, hi to low line
Current limit:	shut down protected at 23 amps ±2 amps
Overvoltage:	crowbar clamp at 6.2 volts ±0.2 volt
Remote voltage sense:	when used with 13037-60027
−5 Volts regulated	−0 to 350 ma
Voltage accuracy:	±5%
Ripple and noise:	less than 5 mv RMS
Output impedance:	less than 70 mΩ
Line regulation:	less than 20 mv change, hi to low line
−20 Volts unregulated	0 − 150 ma
DC output voltage:	18 volts nominal at 120 ma (nominal line)
Output impedance:	50 volts/amp
Ripple:	3.1 volts pp at 150 ma (lower line — 50 CPS)

PON — A logic signal to initialize remote logic at power turn-on; transistor to ground with 1K pull-up. Low, until 60 msec ±40 msec after +5 volts stabilizes at power turn-on.

APPENDIX II
TROUBLESHOOTING AIDS AT TURN-ON

1. Secondary Fuse Blows

It is suggested at this point that an external +7 volt power supply be used to supply +12 unregulated power. Disconnect fuse, and connect external +7 volts to heat sink side of fuse clip, and set *current* limit to 300 ma. This will save a lot of fuses while you are fixing the problem.

Ground the gate of SCR1 with a clip lead and see if +5 output really does go above 6 volts. If not, the problem is in the overvoltage circuit.

A. Overvoltage Circuit Problems

Check the CR3 zener voltage. Look carefully to see that SCR2 and Q1 are loaded correctly.

B. Overvoltage Really Occurring

First, verify that CR4 and CR15 are loaded correctly and the +20v is okay. Then, disable Q5 with a clip lead from base to ground; this disables the overvoltage clamp. Now check that the reference voltage at pin 5 of U2 is 5.0 volts ±0.5 volt.

If it's okay, place a short from U2 pin 13 to ground (available at R29-CR4 anode). Verify that voltage drop across R5 is less than 50 millivolts. If it isn't, CR4 is marked backwards or U2 is faulty.

If that looks okay, check Q7 base voltage. It should be less than 10 mv; if not, Q7 is faulty. If Q7 base is zero, verify that +5 output is less than 0.5 volt. If it is, proceed to next paragraph; if it isn't, check V_{be} of Q1 (across R40).

If it isn't less than 0.2 volt, look for faulty Q7 or shorted C23 or C21. If Q1 looks okay, determine which of Q2, Q3, or Q4 is carrying current by measuring the drop across R4, R6, and R8. If all three are on, look for external shorts and then replace Q1. If just one is on, replace it. If none are on, the problem is CR15. Note: A likely problem to occur is shorts from burrs or shavings in the transistor mounting holes on heat sink.

If +5 volt output is less than 0.5 volt with a short introduced at U2 pin 11, check error amp U3D as follows.

Remove shorts at U2 pins 11 and 13 and let the +5 output rise to the 7 volt limit of the external supply. Verify that the voltage difference from +5 output to U3 pin 14 (and U2 pin 4) is less than 50 millivolts. If not, check R21-R24 before replacing U3. If error amp output looks okay, replace U2.

2. No +5 Output

First, check +12 volt fuse and the +20 supply; then verify that CL test point (U3 pin 7) is low. If CL is high, the problem is in the overcurrent circuit U3A and B. To verify this, short a clip lead from U2 pin 2 to ground; this should restore the +5 output. If it doesn't, trace the voltage through the regulator loop using techniques similar to those employed in section 1.

For example, a clip lead to ground at U2 pin 4 should cause the +5 output to limit on the unregulated supply. If it doesn't, check voltages at U2 pin 10 and the base of Q7 with the clip lead in place.

Alternately, check to see that the error amp output is zero when the output is zero.

3. PON Problems

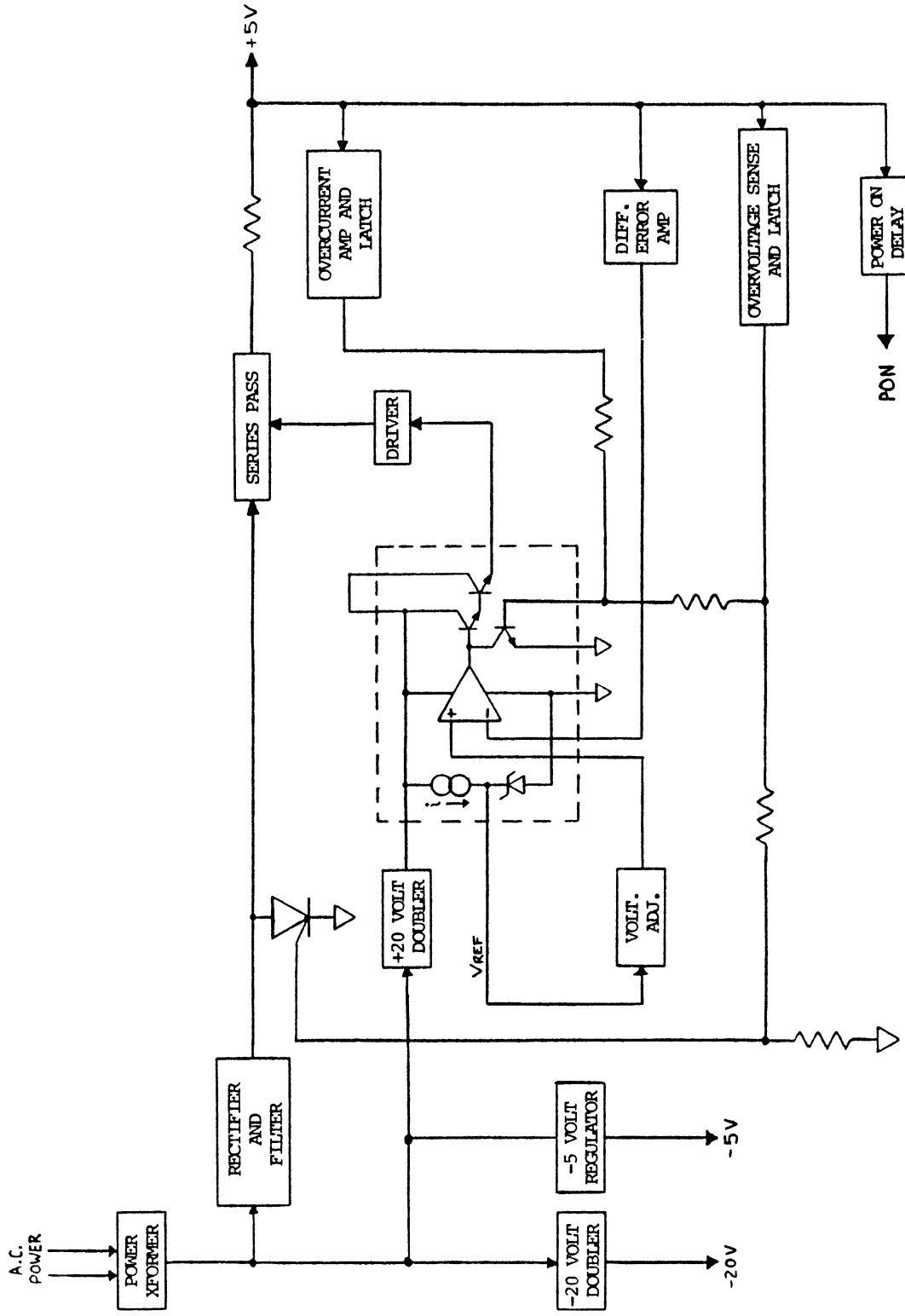
First, be sure +5 and −5 are okay.

A. PON Low All the Time

Verify that U3 pin 8 is low — it should be if the +5 volt regulator is working.

B. PON High All the Time

Check CR14 loading; check to see that U3 pin 8 is high at power turn-on and then ramps low (trigger scope from +5 start-up transient).



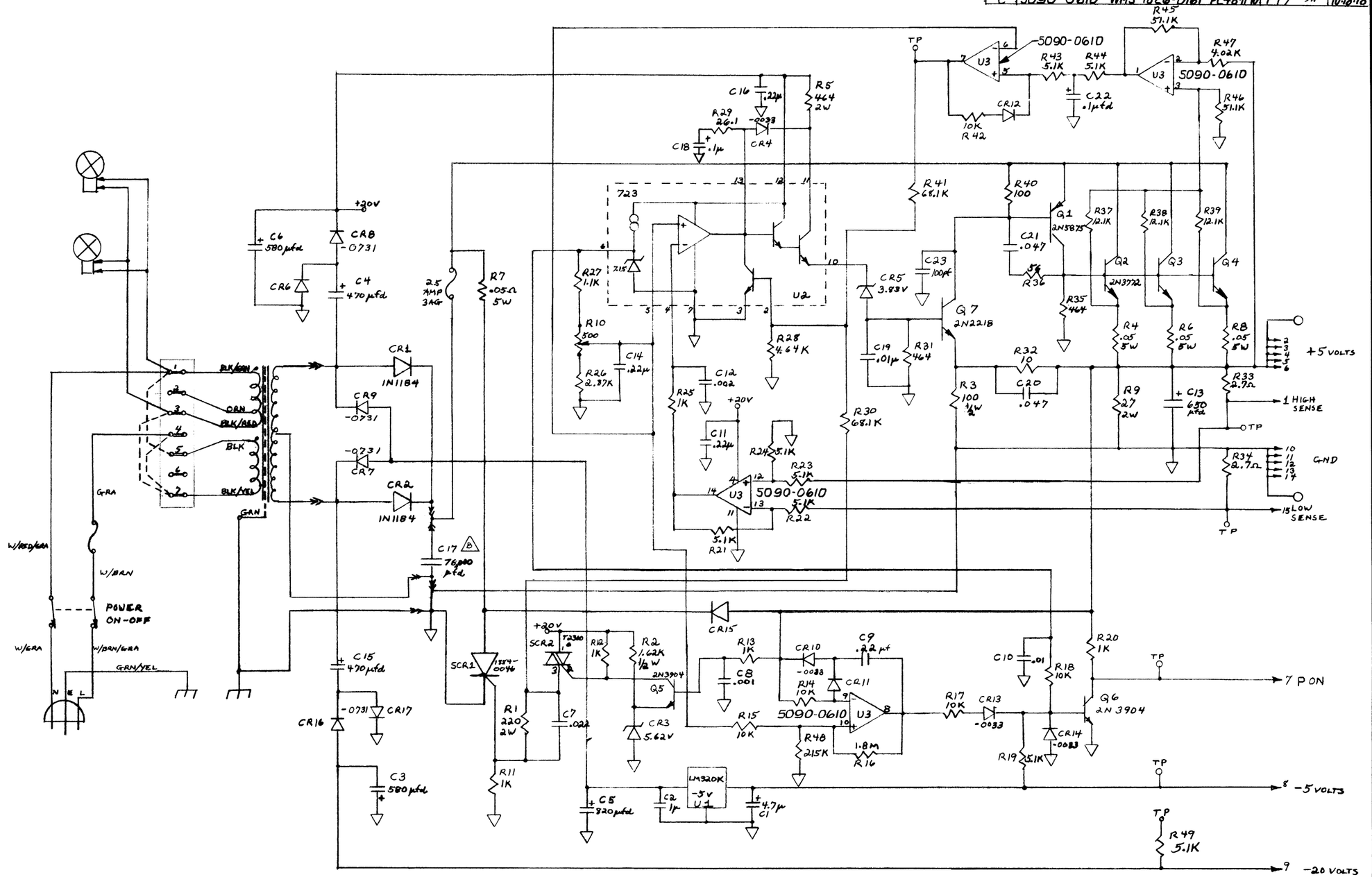
POWER SUPPLY BLOCK DIAGRAM

ENGINEERING RESPONSIBILITY ■ SEPIA ▨

0	1	2	3	4		6		8	9		11	12	14	15
16	17		19		21	22	23	25	29	30	32	33	38	43
45	46	61	63	46	45									

C-13037 - 60026 - 51

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED PER PC48-0060	JR	6/29/77
B	CHANGED VALUE OF FILTER CAPACITOR PC48-0050 KL	K.LARSON	10-8-77
C	5090-0610 WAS 1826-0161 PC48-1140 TT / SH		10-18-78



ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.

TOLERANCES .XX ± .02 .XXX ± .005

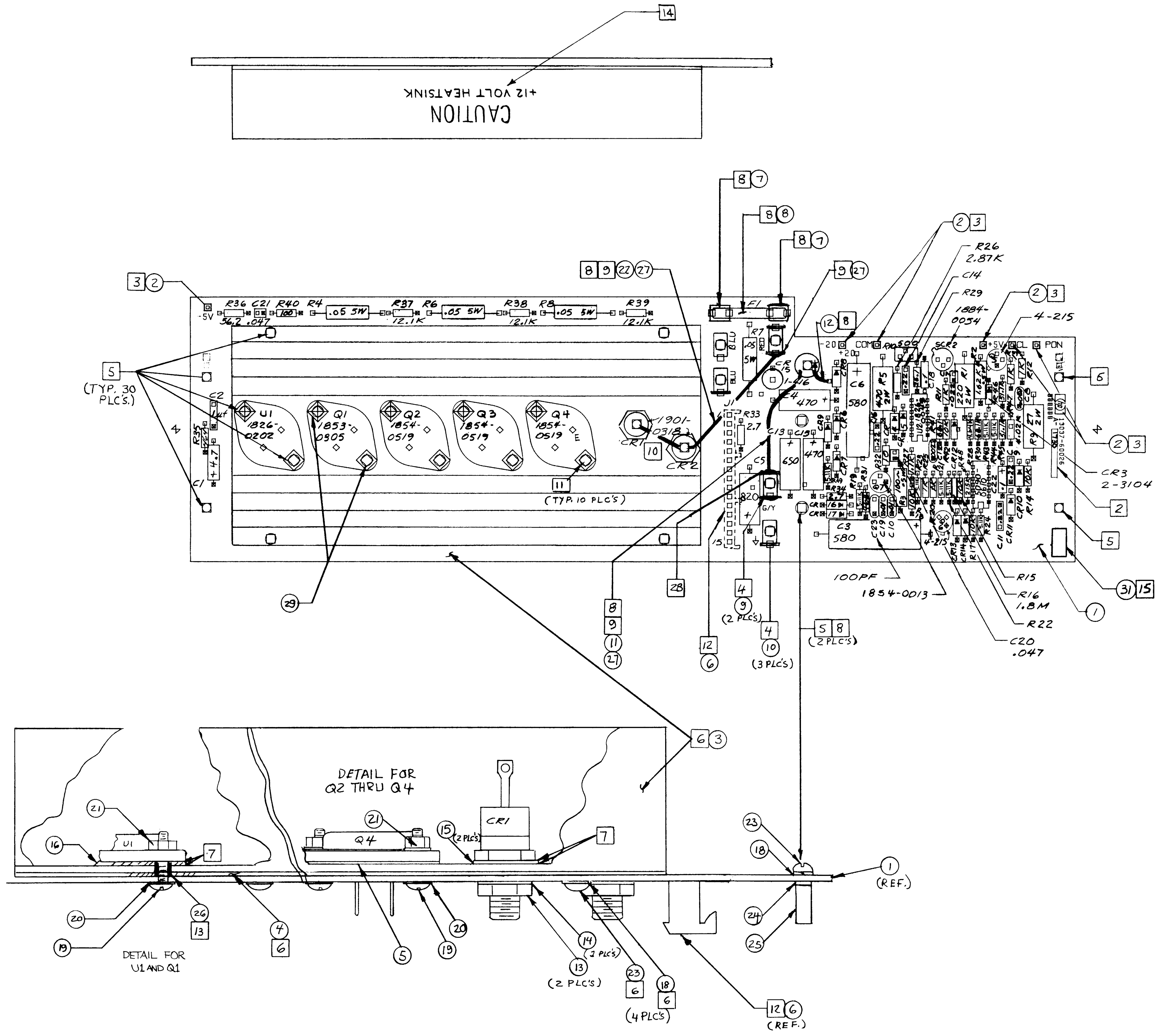
SEE CORP. STD. 608

TITLE	POWER REGULATOR BD 13037 REV A DESIGN
NEXT ASSEMBLY	13037B
FINISH	NONE
SCALE	NONE
HEWLETT PACKARD	
PART NUMBER	13037-60026
	C-13037-60026-51

SHEET OF

D-13037-60026-1

SYM	REVISIONS	APPROVED	DATE
A	AS ISSUED PER PC48-0060	[illegible]	[illegible]
B	CORRECTED RL,R3,R28 VALUES, PC48-0444	L.O./BG	[illegible]
C	CHANGED HARDWARE. PC48-0489	E.O./BG	4-[illegible]
D	ITEM 21 WAS 8150-2344, PC48-[illegible]	[illegible]	[illegible]
E	5090-0610 WAS 1826-0161 PC48-1140	[illegible]	10-20-78



NOTES:

1. UNLESS OTHERWISE SPECIFIED
   ALL RESISTANCE IN OHMS,
   ALL RESISTORS 1/4 W, 1% M.F.
   ALL UNMARKED RESISTORS ARE 5.1K (R19, 21-24, 43, 44 & 49).
   ALL CAPACITANCE IS IN MICROFARADS.
   CR6-9, 16 & 17 ARE 1901-0731
   CR4, & 10-14 ARE 1901-0033.

[2] MARK DATE CODE 1730 & "MADE IN USA."

[3] INSTALL ITEM (2) 6 PLACES.

[4] RIVET ITEMS (9) & (10), THEN HAND SOLDER BEAD AROUND BASE OF LUGS AND ON TOP OF RIVET BEFORE LOADING COMPONENTS (7 PLCS.)

[5] MASK PRIOR TO WAVE SOLDERING.

[6] INSTALL HEAT SINK, ITEM (3) INSULATOR, ITEM (4) WITH SCREWS, ITEM (17) & WASHERS, ITEM (18) (4 PLCS)

[7] USE THERMAL COMPOUND, ITEM (5) ON BOTH SIDES OF XSISTOR INSULATORS, ITEM (16) AND SHOULDER WASHER, ITEM (15) USED ON CR1 & 2.

[8] INSTALL IN TOUCH-UP

[9] INSTALL LUGS TO WIRE BEFORE SOLDERING WIRE TO COMPONENTS.

[10] TORQUE FOR 1/4 x 28 TO 35-40 IN. LBS. MAX.

[11] TORQUE 12 IN. LBS. (#6 HARDWARE)

[12] CHECK FOR FLATNESS – MUST.

[13] INSULATOR CONTACTS BOARD SURFACE TO COMPONENT SURFACE THRU ALL INSULATORS.

[14] RUBBER STAMP CAUTION NOTE ON HEATSINK WITH CABLE MARKING INK BEFORE ASSEMBLY

[15] ITEM (31) PRINTED WITH MONTH AND YEAR, DATE CODE, AND INSTALLED AFTER FINAL TEST.

ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.
31	1	LABEL - "MADE IN USA"	7120-6830		
30	5	RIVET	0361-0662		
29	2	SCREW #6-32 X .612	2360-0229		
28	1	LUG CRIMP	0362-0393		
27	2	LUG CRIMP	0362-0394		
26	1/2 IN	INSUL. TUBING	0890-0334		
25	2	STANDOFF	0380-1085		
24	2	WASHER INT #8	2190-0009		
23 (C)	6	SCREW 8-32 X .312	2510-0101		
22	6 IN.	WIRE #12 AWG RED	8150-2154		
21	10	NUT SELF-LOCK 6-32	2420-0001		
20	10	LOCK WASHER #6	2190-0006		
19	8	SCREW 6-32 X .500	2360-0201		
18	6	LK. WASH #8	2190-0073		
17 (C)					
16	2	INSULATOR - XSTR	1200-0077		
15	2	WASHER, INS.	1200-0088		
14	2	LK WASHER 1/4 SPLIT LOCK	2190-0032		
13	2	NUT 1/4-28	2950-0036		
12 (B)	1 IN.	WIRE #24 AWG BLK	8150-0447		
11	6 IN.	WIRE #24 AWG GRN/YEL	8150-2624		
10	3	TERMINAL .250 TAB 1	0360-1263		
9	2	TERMINAL .250 TAB 2	0360-1264		
8	1	FUSE, 25A	2110-0250		
7	2	FUSE CLIP	2110-0551		
6	1	CONNECTOR, 15 PIN, F, J1	1251-3852		
5	A/R	COMPOUND, THERMAL	6040-0239		
4	1	INSULATOR, H.S.	13037-00007		
3	1	HEAT SINK	13037-20005		
2	6	TERMINAL	0360-0294		
1	1	PCB, ETCHED	13037-80026		

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.
TOLERANCES .XX ± .02 .XXX ± .005
SEE CORP. STD. 608

TITLE: POWER REGULATOR ASSY DWG

HEWLETT PACKARD

NEXT ASSEMBLY: SEE KARDEX

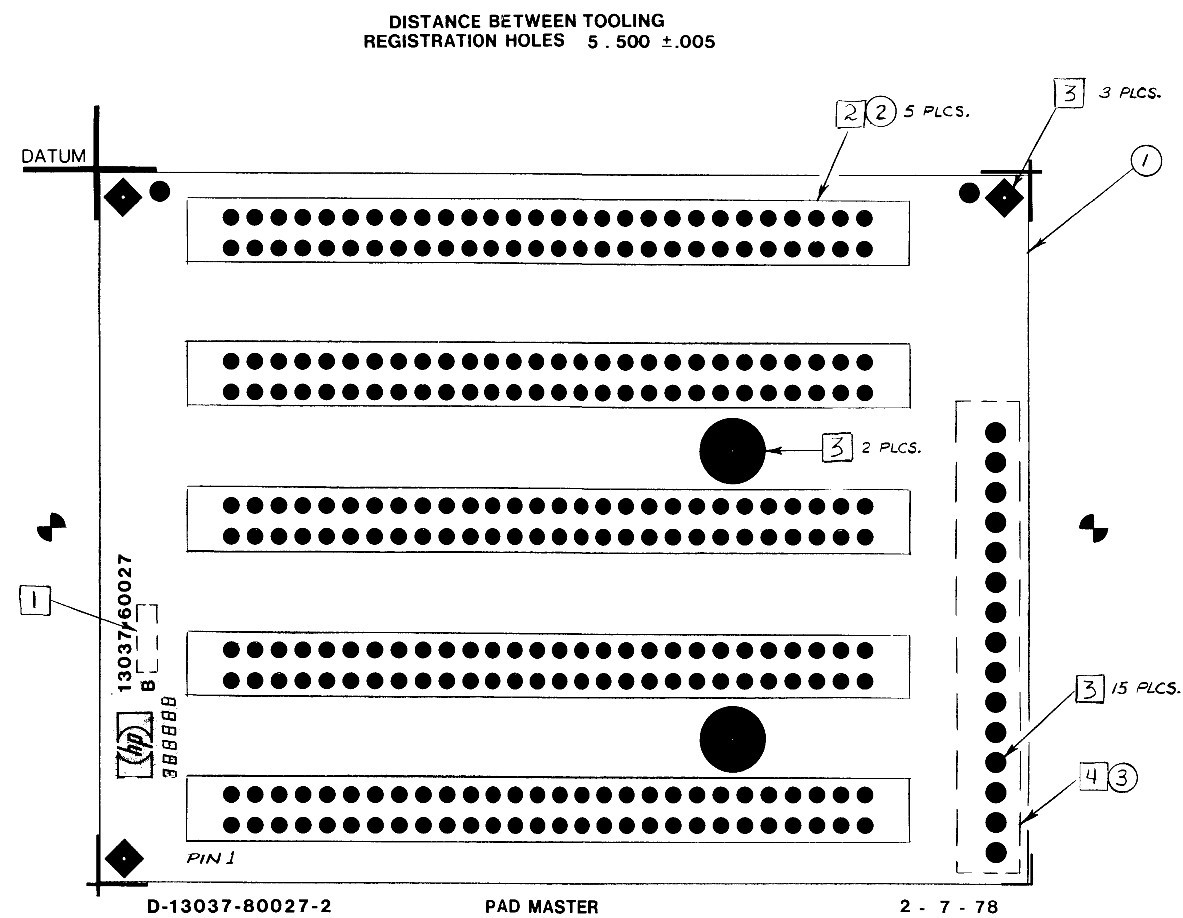
PART NUMBER: 13037-60026

FINISH: NONE SCALE: 1/1

D-13037-60026-1

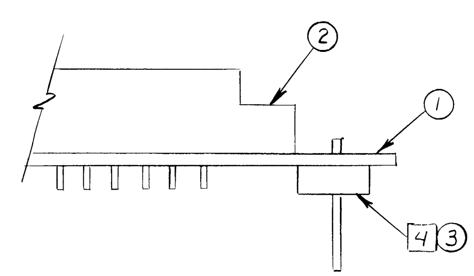
D-13037-60027-1

SYM	REVISIONS	APPROVED	DATE
C	REDRAWN PER PC48-0631	L.O./EC	6-23-78



NOTES:

1. OPR. 264 ; MARK ASSEMBLY DATE CODE 1735
2. OPR. 265 ; INSTALL ITEMS (2)
3. OPR. 265 ; MASK PRIOR TO LOADING.
4. OPR. 266 ; TOUCH UP, INSTALL ITEMS (3)



ITEM	QTY.	MATERIAL-DESCRIPTION	MAT'L-PART NO.	MAT'L-DWG. NO.	MAT'L-SPEC.
3	1	CONNECTOR, 15 PIN	1251-3243		
2	5	CONNECTOR, 56 PIN	1251-3856		
1	1	BOARD, ETCHED	13037-80027		

DO NOT SCALE THIS DRAWING

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES.

TOLERANCES .XX ± .02 .XXX ± .005

SEE CORP. STD. 608

TITLE: PWR. INTERCONNECT BD. ASSLY. DRAWING

NEXT ASSEMBLY: SEE KARDEX

FINISH: —— SCALE: 2x

HEWLETT PACKARD

PART NUMBER: 13037-60027

D-13037-60027-1

CONTENTS

Page

```
1
2                       /
3                       /
4                       /
5                       /
6                       /
7                       /
8                       /
9                       /
10                      /
11                      /
12                      /              1       333333      0000 0    333333   777777777
13                      /             11      3      3    0    0   3       3           7
14                      /            1 1             3   0     0 0          3          7
15                      /           1  1             3   0    0  0          3         7
16                      /              1        33333   0    0  0    33333          7
17                      /              1             3  0  0    0          3       7
18                      /              1             3  0  0    0          3      7
19                      /              1             3  0 0     0          3     7
20                      /              1      3      3   0     0   3      3    7
21                      /           1111111    333333   0 0000      333333   7
22                      /
23                      /
24                      /
25                      /
26                      /
27                      /       CCCCC   N       N  TTTTTTTTT RRRRRR    L          RRRRRR
28                      /      C     C  N       N       T     R     RR  L          R     RR
29                      /     C         NN      N       T     R       R L          R       R
30                      /     C         N N     N       T     R     RR  L          R     RR
31                      /     C         N  N    N       T     RRRRRR    L          RRRRRR
32                      /     C         N   N   N       T     R   R     L          R   R
33                      /     C         N    N N       T     R    R    L          R    R
34                      /     C         N     NN       T     R     R   L          R     R
35                      /      C     C  N      N       T     R      R  L          R      R
36                      /       CCCCC   N      N       T     R       R LLLLLLLL R       R
37                      /
38                      /
39                      /   ************************************************************
40                      /   * (C) COPYRIGHT HEWLETT-PACKARD COMPANY 1978.  ALL RIGHTS     *
41                      /   * RESERVED.  NO PART OF THIS PROGRAM MAY BE PHOTOCOPIED,      *
42                      /   * REPRODUCED OR TRANSLATED TO ANOTHER PROGRAM LANGUAGE WITHOUT *
43                      /   * THE PRIOR WRITTEN CONSENT OF HEWLETT-PACKARD COMPANY.       *
44                      /   ************************************************************
```

INTRODUCTION

```
45
46                      /
47                      /
48                      /
49                      /
50                      /     THIS MICROPROGRAM CONTROLS THE OPERATION OF THE 13037  EXPANDED
51                      /   CAPACITY DISC CONTROLLER.  THE PROGRAM SUPPORTS OPERATION IN
52                      /   EITHER A 3000 OR 2100 CPU ENVIRONMENT.
53                      /     CONTROLLER ARCHITECTURE IS SUCH THAT ANY ONE OF 8 CPU'S CAN
54                      /   ADDRESS ANY ONE OF 11 DISC DRIVES.  THE CPU CAN, UNDER SOFTWARE
55                      /   CONTROL, HOLD OR RELEASE ITS SELECTED DRIVE FOLLOWING COMPLETION
56                      /   OF THE CURRENT DISC OPERATION.  WHILE A CPU HOLDS A DRIVE, NO
57                      /   OTHER CPU MAY ACCESS IT.  SUBJECT TO THIS RESTRICTION, OVERLAPPING
58                      /   SEEKS ARE SUPPORTED BY THIS CONTROLLER.
59                      /     DRIVE ATTENTION BITS ARE SET WHENEVER A SEEK COMPLETES, WHEN
60                      /   THE DRIVE IS LOADED OR UNLOADED, OR WHEN THE DRIVE FAULTS.
61                      /   DURING A CONTROLLER WAIT LOOP, THE DRIVES ARE POLLED TO CHECK
62                      /   THESE BITS.  THE LOWEST-NUMBERED DRIVE WHOSE ATTENTION BIT IS
63                      /   SET GENERATES A CPU INTERRUPT.
64                      /     CPU ADDRESSING IS SIMILAR WHEN CHECKING FOR A PENDING COMMAND,
65                      /   EXCEPT THAT THE CPU'S ARE POLLED STARTING WITH THE NEXT CPU ABOVE
66                      /   THE ONE WHOSE COMMAND HAS JUST COMPLETED (INSTEAD OF STARTING WITH
67                      /   CPU 0).  THIS REQUIRES A HARDWARE MEMORY, BUT INSURES THAT HIGHER-
68                      /   NUMBERED CPU'S WILL NOT BE SHUT OUT BECAUSE OF MULTIPLE OPERATIONS
69                      /   ON LOWER-NUMBERED CPU'S.
70                      /     MICROPROGRAM STRUCTURE ASSUMES A 469 KWORD DISC TRANSFER RATE
71                      /   (7.5 MHZ BIT RATE) AND A MICROINSTRUCTION CYCLE OF 200 NS.  THIS
72                      /   ALLOWS A MAXIMUM OF 10 MICROINSTRUCTIONS BETWEEN EACH DISC WORD I-O.
```

```
                        EXTERNAL ADDRESSES
                /
                /
                /
                /
        EQU : CBSIN   ←   4          / COMMAND BUS INPUT BUFFER.
        EQU : CBUF    ←   2          / CONTROL BUS OUTPUT BUFFER.
        EQU : CLKOF   ←   5          / READ CLOCK OFFSET BUFFER.
        EQU : DSYNC   ←   25         / READ-DECODE ECC SYNCWORD MARK.
        EQU : ECICW   ←   20         / ECC INTERNAL CONTROL WORD.
        EQU : EOC     ←   13         / END-OF-CYLINDER INDICATOR.
        EQU : ESYNC   ←   23         / WRITE-ENCODE ECC SYNCWORD MARK.
        EQU : FLMSK   ←   13         / CONTROLLER MODE (FILE) MASK:
                                     /   BIT 0 ==> ALLOW INCREMENTAL SEEK
                                     /  AT END OF PLATTER OR CYLINDER.
                                     /   BIT 1 ==> CYLINDER MODE (SET),
                                     /             PLATTER MODE (RESET).
                                     /   BIT 2 ==> ENABLE TRACK SPARING.
        EQU : IDTBF   ←   3          / CPU INTERFACE DATA BUFFER.
        EQU : IFNBF   ←   5          / CPU INTERFACE FUNCTION BUFFER.
        EQU : IFVLD   ←   14         / INTERFACE COMMAND VALIDATION CLOCK.
        EQU : INITL   ←   6          / FLAG SET DURING INITIALIZE.
        EQU : INTCW   ←   11         / CONTROLLER INTERNAL CONTROL WORD.
        EQU : NWTRK   ←   7          / NEW TRACK FLAG -- SET WHENEVER
                                     /   CONTROLLER SWITCHES TO NEW TRACK.
        EQU : RAM     ←   1          / INTERFACE-CPU ASSOCIATION RAM (4X8).
        EQU : RAMAD   ←   0          / ADDRESS REGISTER FOR ABOVE RAM.
        EQU : RETRY   ←   6          / ADDRESS RECORD RETRY BIT (MIO 0).
        EQU : SKDIR   ←   1          / AUTO SEEK DIRECTION FLAG, BIT 14:
                                     /   1 = DECREMENTAL SEEK,
                                     /   0 = INCREMENTAL SEEK.
        EQU : SPIBF   ←   16         / CONTROLLER ← SERIAL-PARALLEL CONVERTER.
        EQU : SPOBF   ←   15         / PARALLEL-SERIAL CONVERTER ← CONTROLLER.
        EQU : TGBUS   ←   10         / CONTROLLER-TO-DISC TAG BUS.
        EQU : TIMER   ←   1          / TIME-OUT CIRCUIT TO AVOID
                                     /   CONTROLLER HANG-UPS.
        EQU : WORD1   ←   26         / 1ST WORD OF ERROR PATTERN, MSB'S.
        EQU : WORD2   ←   27         / 2ND WORD OF ERROR PATTERN, LSB'S.
```

```
                        EXTERNAL FLAGS
                /
                /
                /
                /
        EQU : ACRDY   ←   4          / ACCESS READY (HEADS OVER VALID TRACK).
        EQU : ANYER   ←   13         / ANY DATA ERROR -- TRUE IF CSYN OR
                                     /   PSYN <> 0.
        EQU : AUTSK   ←   17         / AUTO INCR-DECR SEEK ENABLED.
        EQU : BURST   ←   12         / C-LOOP CONTROL FLAG.
        EQU : CMRDY   ←   0          / COMMAND READY (AVAILABLE).
        EQU : CYSRF   ←   16         / CYLINDER MODE (1) OR SURFACE MODE (0).
        EQU : DTRDY   ←   1          / DATA READY (AVAILABLE).
        EQU : EOCF    ←   14         / "END-OF-CYLINDER" FLAG.
        EQU : EOD     ←   6          / END OF DATA TRANSFER.
        EQU : EOW     ←   2          / END OF DATA WORD.
        EQU : INITF   ←   11         / SET DURING INIT, WFS, RFS TO
                                     /   PREVENT VRFY-SPARE WHEN
                                     /   SWITCHING TRACKS.
        EQU : INTOK   ←   10         / O.K. TO INTERRUPT CPU WITH
                                     /   DRIVE ATTENTION STATUS.
        EQU : MATCH   ←   12         / P-LOOP CONTROL FLAG.
        EQU : NWTRF   ←   5          / NEW TRACK FLAG -- SET WHENEVER
                                     /   CONTROLLER SWITCHES TO NEW TRACK.
        EQU : OVRUN   ←   7          / DATA OVERRUN.
        EQU : RTRYF   ←   11         / RETRY FLAG -- ALLOWS ONE SEEK
                                     /   TO LOGICAL ADDRESS WHEN AN
                                     /   ADDRESS ERROR IS DETECTED.
        EQU : SPREN   ←   15         / TRACK SPARING ENABLED.
        EQU : UNCOR   ←   20         / UNCORRECTABLE DATA ERROR -- TRUE
                                     /   IF CSYN OR PSYN = 0.
        EQU : XFRNG   ←   3          / 3000 I-O CHANNEL ERROR FLAG.
```

```
143                         DISC AND INTERFACE COMMANDS
144                 /
145                 /
146                 /
147                 /
148                 /  DISC COMMANDS:
149                 /  EQU VALUE SHOWN IS INVERTED FOR GROUND TRUE LOGIC.
150                 /
151                       EQU : ADREC  ←   6          / ADDRESS RECORD COMMAND.
152                       EQU : CLRCM  ←   12         / CLEAR COMMAND.
153                       EQU : CLST   ←   1          / CLEAR SELECTED STATUS.
154                       EQU : DCCOM  ←   13         / DISCONNECT COMMAND.
155                       EQU : OFFST  ←   2          / CYLINDER OFFSET COMMAND.
156                       EQU : RCCOM  ←   4          / RECALIBRATE COMMAND.
157                       EQU : RDCOM  ←   17         / READ COMMAND.
158                       EQU : REQAT  ←   14         / REQUEST ATTENTION COMMAND.
159                       EQU : RSCOM  ←   11         / REQUEST SECTOR ADDRESS COMMAND.
160                       EQU : SEEKC  ←   7          / SEEK COMMAND.
161                       EQU : SLECT  ←   5          / SELECT COMMAND.
162                       EQU : STATC  ←   15         / STATUS COMMAND.
163                       EQU : WRTCM  ←   16         / WRITE COMMAND.
164                       EQU : XSECT  ←   3          / TRANSMIT SECTOR COMMAND.
165                 /
166                 /
167                 /  INTERFACE COMMANDS:
168                 /  EQU VALUES SHOWN INCLUDE BITS TO ENABLE-DISABLE THE COMMAND AND
169                 /  DATA BUS BUFFERS TO THE INTERFACE AS WELL AS BITS TO ENABLE THE
170                 /  LINE DRIVERS OR RECEIVERS ON THE INTERFACE.
171                 /
172                 /
173                 /         7 6  5 4 3  2 1 0
174                 /         - -  - - -  - - -
175                 /       ! E    E E F  F F F !
176                 /       ! C    I I N  N N N !
177                 /       ! D    D R 3  2 1 0 !
178                 /         - -  - - -  - - -
179                 /
180                 /
181                 /    WHERE ECD     = ENABLE CONTROLLER-INTERFACE DATA BUS DRIVERS.
182                 /          EID     = ENABLE DATA BUS DRIVERS ON INTERFACE.
183                 /          EIR     = ENABLE DATA AND COMMAND BUS RECEIVERS ON INTERFACE.
184                 /          FN3-FN0 = CONTROLLER FUNCTION (COMMAND) TO INTERFACE.
185                 /
186                 /  ALL VALUES SHOWN ARE GROUND TRUE (0 SIGNIFIES AN ACTIVE BIT).
187                 /                     -----------
188                 /
189                       EQU : BUSY   ←   41         / SET-CLEAR CONTROLLER BUSY STATUS BIT.
190                       EQU : DSCIF  ←   266        / DISCONNECT INTERFACE.
191                       EQU : DVEND  ←   272        / INCREMENT RETRY COUNTER.
192                       EQU : IFIN   ←   43         / DATA IN (DISC TO INTERFACE).
193                       EQU : IFGTC  ←   222        / GET COMMAND FROM INTERFACE.
194                       EQU : IFOUT  ←   224        / DATA OUT (INTERFACE TO DISC).
195                       EQU : IFPRF  ←   235        / PREFETCH COMMAND FROM INTERFACE.
196                       EQU : RQSRV  ←   270        / SERVICE REQUEST (3000 ONLY).
197                       EQU : SELIF  ←   54         / SELECT INTERFACE.
198                       EQU : SRTRY  ←   53         / SET RETRY COUNTER.
199                       EQU : STDFL  ←   276        / SET DATA FLAG (2100).
200                       EQU : STINT  ←   277        / SET INTERRUPT REQUEST (ERROR OR ATTENTION)
201                       EQU : WRTIO  ←   45         / WRITE TIO COMMAND.
```

```
                    REGISTER ASSIGNMENTS
/
/
/
/
          REG : ACC    <-   A0        / DISPLACEMENT ACCUMULATOR FOR ECC.
          REG : ATTN   <-   A0        / ATTENTION BITS FROM DISC DRIVES.
          REG : BIAS   <-   B0        / SHORTENED CODE BIAS.
          REG : BUF1   <-   A1        / USED
          REG : BUF2   <-   B0        /   FOR TEMPORARY
          REG : BUF3   <-   B3        /   QUANTITIES.
          REG : CACC   <-   A0        / C-DISPLACEMENT ACCUMULATOR.
          REG : CCONS  <-   B3        / HOLDS CHINESE REMAINDER THEOREM
                                      /   CONSTANT FOR C-REGISTER.
          REG : CAPM   <-   B0        / CODE LENGTH MODULUS.
          REG : CLIM   <-   A1        / COUNTER FOR C-LOOP CONTROL.
          REG : COMAD  <-   B0        / COMMAND ADDRESS FOR EXEC ROUTINE.
          REG : COMWD  <-   A3        / COMMAND WORD - USED BY EXEC ROUTINE.
          REG : COUNT  <-   A1        / TEMP CNTR FOR POWER-ON, RQSYN ROUTINES.
          REG : CYLAD  <-   B2        / CYLINDER ADDRESS REGISTER.
          REG : DUMMY  <-   1         / NOT ASSIGNED TO A OR B GROUP -- USED
                                      /   FOR DUMMY I-O TO CLEAR EOW FLAG,
                                      /   AND TO SET T-REG FOR COND BRNCH.
          REG : EWDCT  <-   A1        / TEMPORARY - ERROR WORD COUNTER.
          REG : HSAD   <-   A2        / HEAD-SECTOR ADDRESS REGISTER.
          REG : HSADB  <-   B3        / TEMPORARY B-REGISTER HSAD.
          REG : IFCNO  <-   A0        / INTERFACE (CPU PORT) NUMBER.
          REG : PACC   <-   A3        / P-DISPLACEMENT ACCUMULATOR.
          REG : PCONS  <-   B3        / HOLDS CHINESE REMAINDER THEOREM
                                      /   CONSTANT FOR P-REGISTER.
          REG : PLHED  <-   A3        / SAVES PHYSICAL HEAD AND LOGICAL
                                      /   HEAD DURING INITIALIZE.
          REG : PLIM   <-   A1        / COUNTER FOR P-LOOP CONTROL.
          REG : REGO   <-   A0        / TEMPORARY REGISTER.
          REG : SCNT   <-   A3        / SECTOR COUNT FOR VERIFY ROUTINE.
          REG : STATR  <-   B1        / CONTROLLER STATUS REGISTER.
          REG : TCYLD  <-   A0        / SAVES CYLINDER ADDR DURING VERIFY.
          REG : THSAD  <-   B0        / SAVES HEAD-SECTOR ADDR -- VERIFY.
          REG : TSTTM  <-   A3        / TEMP TO TEST FOR TIME-OUT INTRPT.
          REG : WCNT   <-   A1        / SECTOR WORD COUNTER.
          REG : XREG   <-   A3        / LEAST SIGNIF. WORD OF ERROR PATTERN.
          REG : YREG   <-   B0        / NEXT  SIGNIF. WORD OF ERROR PATTERN.
          REG : ZREG   <-   B3        / MOST  SIGNIF. WORD OF ERROR PATTERN.
```

```
                        CONSTANTS (SOME ARE 1'S COMPLEMENT)
/
/
/
/
        EQU : BIT3   ←   10          / BIT 3 MASK.
        EQU : CAPMU  ←   10          / NATURAL CODE LENGTH = 2279
        EQU : CAPML  ←   347         /   (DEC), UPPER & LOWER BYTES.
        EQU : CATN   ←   376         / CLEAR SELECTED ATTENTION
        EQU : CCLKH  ←   310         / FREEZE P, CLOCK HI FOR C SHIFT.
        EQU : CCLKL  ←   300         / FREEZE P, CLOCK LO FOR C SHIFT.
        EQU : CCONU  ←   6           / CHINESE REMAINDER THEOREM CONSTANT
        EQU : CCONL  ←   67          /   FOR C, UPPER AND LOWER BYTES.
        EQU : CFST   ←   375         / CLEAR SELECTED FIRST STATUS.
        EQU : CLBSY  ←   1           / CLEAR CONTROLLER BUSY BIT.
        EQU : CLICW  ←   37          / CLEAR INTERNAL CONTROL WORD.
        EQU : CLKHH  ←   211         / TAKE CLOCK HIGH ONCE.
        EQU : CLKHI  ←   230         / TAKE CLOCK HI FOR INHIBITS.
        EQU : CLRCL  ←   1           / CLEAR INTERFACE FUNCTION CLOCK.
        EQU : CSTRB  ←   37          / CLEAR DISC COMMAND STROBE.
        EQU : CURNT  ←   16          / ADDRESS OF CURRENT I'FACE IN RAM.
        EQU : DRTYP  ←   14          / ADDR OF DRIVE TYPE WORD IN RAM.
        EQU : DSC    ←   200         / START COUNT OF 7 EOWS TO FREEZE.
        EQU : ERWCT  ←   371         / # OF CHECK WORDS (TWO'S COMP.).
        EQU : ESC    ←   0           / START COUNT OF 2 EOWS TO SHIFT.
        EQU : FALSE  ←   377         / POSITIVE-TRUE FALSE.
        EQU : FRPSH  ←   312         / CLOCK HI, CLEAR INHREG, FREEZE P.
        EQU : FULSC  ←   166         / FULL SECTOR WDCNT, TWO'S COMP.
                                     /   (INCLUDES SYNC, CYLAD, HSAD,
                                     /   CRC, AND ECC WORDS).
        EQU : MXDRV  ←   365         / MAX # OF DRIVES, TWO'S COMP.
        EQU : NSEC   ←   15          / RAM ADDR  OF VERIFY RETRY COUNTER.
        EQU : PCLKH  ←   250         / FREEZE C, CLOCK HI FOR SHIFT P.
        EQU : PCLKL  ←   240         / FREEZE C CLOCK LO FOR SHIFT P.
        EQU : PCONU  ←   2           / CHINESE REMAINDER THEOREM CONSTANT
        EQU : PCONL  ←   261         /   FOR P, UPPER AND LOWER BYTES.
        EQU : RDONL  ←   13          / ADDRESS OF READ-ONLY
                                     /   SWITCH STATUS IN RAM.
        EQU : POLIF  ←   17          / ADDRESS OF LAST INTERFACE POLLED.
        EQU : PRVAT  ←   10          / RAM ADDR OF DRV WITH PREV ATTN.
        EQU : RCC1   ←   76          / READ AND CYCLIC CHECK.
        EQU : RCC2   ←   72          / SAME WITH ACRDY CHECK ENABLED.
        EQU : RDWRT  ←   13          / CLR ICW, KEEP RD-WRT BIT, CRCC.
        EQU : RECC   ←   226         / READ WITH ECC.
        EQU : RERWD  ←   52          / READ (SHIFT IN) ERROR WORDS.
        EQU : RWCT   ←   201         / DATA WORD COUNT - 1, TWO'S COMP.
        EQU : SETCL  ←   0           / SET INTERFACE FUNCTION CLOCK.
        EQU : START  ←   100         / START CONTROLLER TIME-OUT CKT.
        EQU : STBSY  ←   0           / SET CONTROLLER BUSY BIT.
        EQU : STOP   ←   200         / STOP CONTROLLER TIME-OUT CKT.
        EQU : STRB   ←   77          / SET DISC COMMAND STROBE.
        EQU : TRUE   ←   0           / POSITIVE-TRUE TRUE.
        EQU : WAKST  ←   2           / DRIVE AVAILABLE STATUS.
        EQU : WCC1   ←   75          / WRITE AND CYCLIC CHECK.
        EQU : WCC2   ←   71          / SAME WITH ACRDY CHECK ENABLED.
        EQU : WCT    ←   200         / DATA WORD COUNT, TWO'S COMP.
        EQU : WECC   ←   26          / WRITE WITH ECC.
        EQU : WERWD  ←   61          / WRITE (SHIFT OUT) ERROR WORDS.
        EQU : WKUPC  ←   26          / WAKEUP COMMAND CODE.
```

```
                          POWER ON CLEAR, TIME-OUT PROCESSOR
                /
                /
                /
                /
                      ORG = 0
                /
                /
                /  CLEAR RAM ONLY WHEN POWER IS TURNED ON OR WHEN A HARD CLEAR
                /  IS RECEIVED FROM THE CPU.  LEAVE IT ALONE IF WE GET HERE VIA
                /  A TIME-OUT INTERRUPT.
                /
0000 64000001           BR : NXTAD                      / SET CC BIT IF CLEAR.
0001 66001625 NXTAD     CA : IFDSC                      / PREVENT INTERRUPT AT POWER-ON.
0002 10100454           DI : TSTTM  ←   TIMER,U         / SEE IF THIS WAS
0003 37706074       TSTTM =       PASS TSTTM,U          /   TIME-OUT INTERRUPT.
0004 44540000           CC = TMSB             ,RS
0005 64000025           BR : PWRON                      / NO, GO DO POWER-ON SEQUENCE.
                                                        / YES, FALL INTO TIME-OUT SQNC.
                /
                /
                /  THIS SECTION IS ENTERED IN CASE OF AN INTERFACE TIME-OUT.  ANY
                /  DRIVE HOLD BITS WHICH MAY HAVE BEEN SET BY THE INTERFACE ARE
                /  CLEARED SO THAT OTHER INTERFACES MAY USE THE DRIVE.
                /
0006 27722012         COUNT = 12    PASS       ,L     / TEST ONLY RAM ADDRESSES 10-0.
0007 07620016           DO : RAMAD  ←   CURNT,L         / FIND OUT WHO CURRENT I'FC IS
0010 10120440           DI : REGO   ←   RAM  ,L         /   (IT'S JUST TIMED OUT).
0011 17620405 CLHLD     DO : RAMAD  ←   COUNT,L         / GET I'FC ASSOCIATED WITH
0012 10020456           DI : BUF3   ←   RAM  ,L         /   CURRENT DRIVE.
0013 36437520         BUF3 =        PASS BUF3 ,L,SR   / CLEAR IT'S HOLD BIT.
0014 36223460        DUMMY = REGO   XOR  BUF3 ,L      / SEE IF DRIVE BELONGS TO
0015 44400000           CC = TNZRO                      /   CURRENT INTERFACE.
0016 64000021           BR : NTEQL                      / NO, DON'T MODIFY RAM.
0017 36437540         BUF3 =        PASS BUF3 ,L,SL   / YES, RESTORE I'FC NUMBER WITH
0020 16421455           DO : RAM    ←   BUF3 ,L         /   HOLD BIT CLEAR AND WRITE BACK.
0021 30122064 NTEQL COUNT =         DEC  COUNT,L      / SET UP NEXT RAM WORD.
0022 44100000           CC = LOVER                      / DONE THEM ALL (ROLLED THROUGH 0)?
0023 64000011           BR : CLHLD                      / NOT YET, DO ANOTHER ONE.
0024 64000035           BR : CLR1                       / YES, RESET FLMSK, POLL INTERFACES.
                /
                /
                /  THIS SECTION CLEARS THE INTERFACE-DISC DRIVE ASSOCIATIVE RAM
                /  WHEN POWER IS TURNED ON OR WHEN THE INTERFACE SENDS A HARD
                /  CLEAR COMMAND.
                /
0025 27722020 PWRON COUNT = 20      PASS       ,L     / SET UP LOOP COUNTER.
0026 30122064 CLRAM COUNT =         DEC  COUNT,L      / GET NEXT RAM ADDRESS (17 OCTAL-0).
0027 44400000           CC = TNZRO                      / SET CC BIT UNLESS LAST ONE (0).
0030 17620405           DO : RAMAD  ←   COUNT,L         / ADDRESS THE CURRENT RAM WORD.
0031 07620400           DO : RAM    ←   0    ,L         / CLEAR IT.
0032 64000026           BR : CLRAM                      / LOOP IF NOT DONE (SEE CC AT *-3).
0033 07624012           DO : TGBUS  ←   CLRCM,L         / DONE, CLEAR ALL DRIVES.
0034 66001641           CA : STROB
0035 07602777 CLR1      DO : CLKOF  ←   377  ,U         / CLEAR ANY CLOCK OFFSET.
0036 07625777           DO : FLMSK  ←   FALSE,L         / RESET THE FILE MASK.
                                                        / FALL INTO POLL LOOP.
```

```
                                POLL LOOP
                        /
                        /
                        /
                        /
                        /   POLL LOOP -- WAIT HERE BETWEEN EVENTS WHICH DO NOT REQUIRE
                        /   RETENTION OF PAST HISTORY (DISC OR CONTROLLER STATUS, CYLAD,
                        /   FILE MASK, ETC).  WHILE IN THIS LOOP, THE CPU'S ARE POLLED
                        /   FOR NEW COMMANDS AND THE DRIVES ARE POLLED FOR ATTENTION.
                        /
0037 66001614 POLL      CA : FLUSH                  / CLEAR COMMAND BUFFER.
0040 66001620 POLL1     CA : STPCL                  / TURN OFF ALL CLOCKS.
0041 66001704           CA : BSYCL                  / CLEAR THE CONTROLLER BUSY BIT.
0042 66001640           CA : DISCO                  / DISCONNECT THE DISC DRIVES
0043 66001625           CA : IFDSC                  /   AND THE INTERFACE PORTS.
                        /
                        /  THIS SECTION EXAMINES THE COMMAND READY FLAG OF EACH INTERFACE,
                        /  STARTING WITH THE ONE FOLLOWING THE CURRENTLY-SELECTED INTERFACE.
                        /  IN THIS WAY, THE INTERFACES ARE POLLED SERIALLY, AND NO COMBINATION
                        /  OF OPERATIONS ON LOWER-NUMBERED INTERFACES CAN SHUT OUT OPERATIONS ON
                        /  HIGHER-NUMBERED ONES.  THE CURRENTLY CONNECTED CPU (INTERFACE) NUMBER
                        /  IS STORED IN RAM(14).  (SEE SUBROUTINE "UNIT" FOR RAM FORMAT).
                        /
0044 66001651 POLL2     CA : CLRST                  / CLEAR STATUS REGISTER (STATR).
0045 07620017           DO : RAMAD  ←   POLIF,L     / LOOK UP LAST INTERFACE
0046 10020456           DI : BUF3   ←   RAM  ,L     /   TO BE POLLED.
0047 22237401        BUF3 = 1       PLUS BUF3 ,L    / BUMP TO NEXT ONE.
0050 27437407        BUF3 = 7       AND  BUF3 ,L    / RESET TO 0 IF WE GET 8
                                                    /   (INSURES HOLD BIT NOT SET).
0051 16421455           DO : RAM    ←   BUF3 ,L     / UPDATE "LAST INTERFACE" IN RAM.
0052 07620016           DO : RAMAD  ←   CURNT,L     / ALSO UPDATE "CURRENT INTERFACE".
0053 16421455           DO : RAM    ←   BUF3 ,L
0054 15221555           DO : IDTBF  ←   BUF3 ,L,C   / NOW SELECT NEW INTERFACE.
0055 66001627           CA : IFSEL
0056 66000424           CA : CKXFR                  / CHECK FOR 3000 CHANNEL ERROR.
0057 42000000           CC = CMRDY                  / INTERFACE HAVE COMMAND WAITING?
0060 64000147           BR : EXEC                   / YES, GO DO IT, LEAVE ATTN ON.
0061 66001625 POLL3     CA : IFDSC                  / NO, OR DRIVE NOT AVAILABLE,
                                                    /   DISCONNECT INTERFACE.
0062 07624014           DO : TGBUS  ←   REQAT,L     / REQUEST ATTENTION FROM DRIVES.
0063 66001642           CA : STRB1                  / VALIDATE COMMAND, LEAVE STROBE ON.
0064 10102200           DI : ATTN   ←   CBSIN       / CHECK DRIVE ATTENTION WORD.
0065 34100260        ATTN =         CMP  ATTN
0066 44440000           CC = TNZRO          ,RS     / ANY DRIVES NEED HELP?
0067 64000044           BR : POLL2                  / NO, CONTINUE POLLING INTERFACES.
```

```
406                          SET ATTENTION AND INTERRUPT
407                  /
408                  /
409                  /
410                  /
411                  /    ONE OR MORE DRIVES HAS ATTENTION SET.  STARTING WITH ONE
412                  /  DRIVE AFTER THE ONE LAST SERVICED, CHECK FOR ATTN IN ASCEND-
413                  /  ING ORDER (0 FOLLOWS 7), AND REPORT THE FIRST ONE WITH
414                  /  ATTENTION.  BUILD THE ATTENTION STATUS AND UNIT NUMBER IN
415                  /  A TEMPORARY B-REGISTER.  PUT INTO STATR ONLY IF WE GET TO
416                  /  LABEL ATTN3.  THIS AVOIDS LEAVING ATTENTION STATUS LYING
417                  /  AROUND FOR COMMANDS WHICH DON'T CLEAR STATR (E.G. RQSYN).
418                  /
419 0070 07604437            DO : INTCW  ←   CSTRB,U     / TURN OFF ATTENTION STROBE.
420 0071 37610200          BUF2 =        PASS ATTN ,SW    / SET DUPLICATE DRIVE ATTENTION
421 0072 36500060          ATTN =        PASS BUF2 ,U     /   BYTES IN BOTH HALVES OF ATTN.
422 0073 07620010            DO : RAMAD  ←   PRVAT,L     / LOOK UP LAST DRIVE TO
423 0074 10120444            DI : BUF1   ←   RAM  ,L     /   HAVE ATTENTION BIT SERVICED.
424 0075 37636064          BUF3 =        PASS BUF1 ,L     / ACCUMULATE DRIVE NUMBER IN BUF3,L.
425 0076 37700320 ATTN1    ATTN =        PASS ATTN ,SR    / SHIFT AND IGNORE ATTENTION BITS
426 0077 30122064          BUF1 =        DEC  BUF1 ,L     /   UNTIL WE COME TO DRIVE BEYOND
427 0100 44100000            CC = LOVER                   /   LAST ONE WE REPORTED.
428 0101 64000076            BR : ATTN1
429 0102 22237401 ATTN2    BUF3 = 1      PLUS BUF3 ,L     / FOUND IT, NOW START COUNTING.
430 0103 27437407          BUF3 = 7      AND  BUF3 ,L     / RESET TO 0 IF WE GET 8.
431 0104 37700320          ATTN =        PASS ATTN ,SR    / TEST CURRENT ATTN LSB, SHIFT.
432 0105 44640000            CC = TLSB             ,RS    / THIS DRIVE HAVE ATTENTION SET?
433 0106 64000102            BR : ATTN2                   / NO, CONTINUE SEARCH.
434 0107 16421455            DO : RAM    ←   BUF3 ,L     / YES, SAVE ITS NUMBER FOR NEXT TIME.
435 0110 16421415            DO : RAMAD  ←   BUF3 ,L     / LOOK UP CPU FOR THIS DRIVE.
436 0111 10120440            DI : IFCNO  ←   RAM  ,L
437 0112 37700320         IFCNO =        PASS IFCNO,SR    / POSITION CPU FIELD.
438 0113 07620016            DO : RAMAD  ←   CURNT,L     / MAKE IT THE CURRENT CPU IN RAM.
439 0114 17620041            DO : RAM    ←   IFCNO,L
440 0115 14020141            DO : IDTBF  ←   IFCNO,L,C   / CONNECT CONTROLLER TO NEW CPU
441 0116 66001627            CA : IFSEL
442 0117 15221515            DO : CBUF   ←   BUF3 ,L,C   /   AND TO DRIVE WHICH HAS
443 0120 66001347            CA : UNIT2                   /   ATTENTION BIT SET.
444 0121 66000424            CA : CKXFR                   / CHECK FOR 3000 CHANNEL ERROR.
445 0122 27616037          BUF3 = 37     PASS       ,U    / BUF3,U GETS ATTENTION STATUS.
446 0123 42040000            CC = CMRDY             ,RS   / IS NEXT COMMAND ALREADY THERE?
447 0124 64000131            BR : ATTN3                   / NO, GO INTERRUPT.
448 0125 66001666            CA : PRFCH                   / YES, PRE-FETCH IT.
449 0126 26222026         DUMMY = WKUPC XOR  COMAD,L      / TEST IT FOR WAKE-UP COMMAND.
450 0127 44400000            CC = TNZRO
451 0130 64000150            BR : EXEC1                   / NO, GO EXECUTE NORMAL COMMAND.
452 0131 36413660 ATTN3    STATR =       PASS BUF3        / YES, OR NO COMMAND, NOW SET STATR.
453 0132 66001702            CA : TRNOF                   / CLOCKS OFF, WRT TIO, CLEAR BUSY.
454 0133 07622677            DO : IFNBF  ←   STINT,L     / SET TO INTERRUPT CPU.
455 0134 43040000            CC = INTOK            ,RS   / WILL CPU ALLOW INTERRUPT?
456 0135 64000040            BR : POLL1                   / NO, CLR STATR, POLL UNTIL IT DOES.
457
458                  /
459                  /
460                  /
461                  /
462                  /  THE FOLLOWING USE OF IFVLD IS FOR THREE REASONS:
463                  /  1.  THIS IS A REAL-TIME CONSIDERATION, WHERE WE WANT A MINIMUM
464                  /      LATENCY BETWEEN THE TIME WE TEST THE INTOK FLAG AND WHEN
465                  /      WE INTERRUPT.  THIS IS IN CASE THE PROGRAMMER HAS CHANGED
466                  /      INTOK JUST AFTER WE TESTED IT.  THE CALL TO ICLK2 AVOIDS
467                  /      THE 600 NSEC DELAY BEFORE THE LEADING EDGE OF IFVLD.
468                  /  2.  THE 600 NSEC DELAY IN ICLCK IS NOT REQUIRED HERE, SINCE THE
469                  /      "STINT" FUNCTION WENT OUT 700 NSEC AGO (SEE PREVIOUS PAGE).
470                  /  3.  A DIRECT "IFVLD _ CLRCL" SHOULD FOLLOW RATHER THAN A CALL TO
471                  /      ICLK3.  THE FUNCTION IS VALIDATED ON THE TRAILING EDGE OF IFVLD,
472                  /      AND THE DIRECT CALL MAKES THE TRAILING EDGE OCCUR EARLIER.
473                  /
474 0136 66001633            CA : ICLK2                   / VALIDATE INTERRUPT COMMAND.
475 0137 07606001            DO : IFVLD  ←   CLRCL,U     / THEN TURN IT OFF.
476 0140 66001604 ATTN4      CA : SWAT3                   / CLEAR THE DRIVE'S ATTENTION BIT.
477                  /
478                  /  WAIT HERE FOR NEXT COMMAND FROM INTERFACE.
479                  /
480 0141 66001640 CMDWT      CA : DISCO                   / DISCONNECT ALL DRIVES.
481 0142 07600500            DO : TIMER  ←   START,U     / TURN ON TIME-OUT CIRCUIT.
482 0143 66000424 CMDW1      CA : CKXFR                   / CHECK FOR 3000 CHANNEL ERROR.
483 0144 42040000            CC = CMRDY             ,RS   / WAIT FOR NEW COMMAND
484 0145 64000143            BR : CMDW1                   /   OR TIME OUT.
485 0146 07600600            DO : TIMER  ←   STOP ,U     / GOT IT, TURN OFF TIME-OUT
486                                                       /   CIRCUIT AND FALL INTO EXEC.
```

```
                    COMMAND FORMATS
/
/
/
/       1 1 1 1 1 1
/       5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0  BIT (2100 NUMBERING)
/       - - - - - - - - - - - - - - - -
/     !       C C C C C                 !  ADREC, CLEAR, END,
/     !                                 !    LTIO, REQAD, RQSYN
/     !       C C C C C         U U U U !  RQSCT, STAT
/     !       C C C C C H D     S E C T !  COLD LOAD
/     !       C C C C C R T R Y M A S K !  SET FILE MASK
/     !       C C C C C 1       U U U U !  SEEK, RECALIBRATE
/     ! S P D C C C C C 0       U U U U !  WRITE ADDRESS (INITIALIZE)
/     !       C C C C C 0       U U U U !  (ALL OTHER COMMANDS)
/       - - - - - - - - - - - - - - - -
/
/      WHERE  C = FIVE-BIT COMMAND OPCODE:
/
/        UBYTE  LBYTE  DESCRIPTION
/        -----  -----  -----------
/
/            0      0  COLD LOAD READ (BOOT)
/          400      1  RECALIBRATE
/         1000      2  SEEK
/         1400      3  REQUEST STATUS
/         2000      4  REQUEST SECTOR ADDRESS
/         2400      5  READ
/         3000      6  READ FULL SECTOR
/         3400      7  VERIFY
/         4000     10  WRITE
/         4400     11  WRITE FULL SECTOR
/         5000     12  CLEAR
/         5400     13  WRITE ADDRESS (INITIALIZE)
/         6000     14  ADDRESS RECORD
/         6400     15  REQUEST SYNDROME
/         7000     16  READ WITH OFFSET
/         7400     17  SET FILE MASK
/        10000     20  (NOT USED)
/        10400     21  (NOT USED)
/        11000     22  READ WITHOUT VERIFY
/        11400     23  LOAD TIO REGISTER
/        12000     24  REQUEST DISC ADDRESS
/        12400     25  END (RETURNS CONTROLLER TO POLL LOOP)
/        13000     26  WAKEUP
/
/             S = FLAG AS SPARE TRACK.
/             P = FLAG AS PROTECTED TRACK (WRITE COMMAND MAY ABORT).
/             D = FLAG AS DEFECTIVE TRACK.
/            HD = HEAD ADDRESS FOR COLD LOAD READ COMMAND.
/          SECT = SECTOR ADDRESS FOR COLD LOAD READ COMMAND.
/          RTRY = DATA TRANSFER RETRY COUNTER (3000 ENVIRONMENTS).
/          MASK = FILE MASK BITS (SEE DESCRIPTION IN SFMSK PROCESSOR).
/             U = DISC DRIVE (UNIT) ADDRESSED BY THE COMMAND.  ERRORS
/                 MAY OCCUR IF U = 10 (OCTAL).
/         BIT 7 = HOLD (BUSY) BIT.  IF SET, RESERVES DRIVE "U" FOR
/                 INTERFACE ISSUING THE COMMAND.  COMMANDS FROM OTHER
/                 INTERFACES TRYING TO ACCESS THIS DRIVE WILL BE DEFERRED
/                 UNTIL THE DRIVE IS AVAILABLE.
```

```
                          OBTAINING COMMAND AND EXECUTING BRANCH VECTOR
                /
                /
                /
                /
0147 66001666 EXEC      CA : PRFCH                      / PRE-FETCH THE NEXT COMMAND.
0150 27616037 EXEC1   BUF3 = 37     PASS       ,U      / CLEAR S, P, AND D BITS
0151 37505470         HSAD = HSAD   AND   BUF3 ,U      /   FROM HSAD.
0152 07623377           DO : INITL  ←    FALSE,L       / CLEAR INITIALIZE FLAG.
                /         DO : RETRY  ←    FALSE,L       DUMMY AS LONG AS RETRY = INITL.
0153 22222351        DUMMY = 351    PLUS COMAD,L      / CHECK FOR ILLEGAL OPCODE.
0154 44140000           CC = LOVER             ,RS
0155 64000161           BR : EXEC2                      / OPCODE O.K.
0156 66001614 ILLGL     CA : FLUSH                      / ILLEGAL, FLUSH IT FROM
0157 66001651                                           /   INTERFACE COMMAND BUFFER.
                        CA : CLRST                      / CLEAR STATUS-1.
0160 64001771           BR : OPCER                      / GO INTERRUPT - ILLEGAL OPCODE.
                /
                /
                /  CODING FROM HERE TO END OF PAGE MUST RESIDE IN ADDRESSES <= 377B.
                /
0161 22236164 EXEC2   BUF3 = CVECT PLUS COMAD,L      / ADD TABLE ADDRESS TO
0162 35221414           NO : BUF3              ,L     /   GET ADDRESS INTO
0163 74000000           BR : I                        /   BRANCH VECTOR.
                /
                /  INDIRECT BRANCH THROUGH COMMAND VECTOR TO START OF
                /  APPROPRIATE ROUTINE.
                /
0164 64000215 CVECT     BR : BOOT                       / COLD LOAD READ.
0165 64000233           BR : RECAL                      / RECALIBRATE (RESTORE HOME).
0166 64000241           BR : SEEK                       / SEEK.
0167 64000301           BR : STATS                      / REQUEST STATUS.
0170 64000346           BR : RQSCT                      / REQUEST SECTOR.
0171 64000407           BR : READ                       / READ.
0172 64000613           BR : RFS                        / READ FULL SECTOR.
0173 64000507           BR : VRFCM                      / VERIFY SECTOR(S).
0174 64001002           BR : WRITE                      / WRITE.
0175 64001032           BR : WFS                        / WRITE FULL SECTOR.
0176 64001262           BR : CLR                        / CLEAR.
0177 64000642           BR : INIT                       / WRITE ADDRESS (INITIALIZE).
0200 64000264           BR : RECRD                      / ADDRESS RECORD.
0201 64001106           BR : RQSYN                      / REQUEST SYNDROME.
0202 64000460           BR : ROFST                      / READ WITH OFFSET.
0203 64001252           BR : SFMSK                      / SET FILE MASK.
0204 64000156           BR : ILLGL                      / ILLEGAL OPCODE (WAS CLRUB).
0205 64000156           BR : ILLGL                      / ILLEGAL OPCODE (WAS UNTIN).
0206 64000416           BR : RNVFY                      / READ WITHOUT VERIFY.
0207 64001100           BR : LTIO                       / LOAD TIO REGISTER.
0210 64001266           BR : REQAD                      / REQUEST DISC ADDRESS.
0211 64000037           BR : POLL                       / END OF COMMAND SEQUENCE
                                                        /   (RETURN TO POLL LOOP).
                                                        / (FALL INTO WAKEUP COMMAND IF HERE).
```

```
598                          WAKEUP, COLD LOAD READ (BOOT) COMMANDS
599                  /
600                  /
601                  /
602                  /
603                  /  THE WAKEUP COMMAND IS ISSUED BY AN INTERFACE WHEN IT WANTS TO
604                  /  ACCESS A DRIVE CURRENTLY BEING HELD BY ANOTHER INTERFACE.  AS
605                  /  LONG AS THE DRIVE CONTINUES TO BE HELD BY THE OTHER INTERFACE,
606                  /  THE WAKEUP COMMAND FUNCTIONS AS A NOP (CONTROLLER RETURNS TO
607                  /  POLL).  WHEN THE DRIVE IS AVAILABLE, THE FIRST INTERFACE IS
608                  /  INTERRUPTED ("AWAKENED") WITH A "DRIVE AVAILABLE" STATUS AND
609                  /  DRIVE NUMBER.  WHEN THE DRIVE BECOMES AVAILABLE, ITS HOLD BIT
610                  /  WILL BE UPDATED FROM BIT 7 (HOLD BIT FIELD) OF THE WAKEUP
611                  /  COMMAND.  NOTE:  IF THE DRIVE IS BEING HELD BY THE INTERFACE
612                  /  ISSUING THE WAKEUP COMMAND, "DRIVE AVAILABLE" STATUS IS
613                  /  RETURNED IMMEDIATELY.
614                  /
615 0212 66001317 WAKUP    CA : UNIT                    / TRY TO CONNECT DRIVE.
616 0213 22212402       STATR = WAKST PLUS STATR,U     / SET UP "DRIVE AVAILABLE" STATUS.
617 0214 64001257          BR : INTRP                   / GO WAKE UP THE INTERFACE.
618                  /
619                  /  BOOT TESTS THE HOLD BIT OF UNIT 0.  IF CLEAR, OR IF UNIT 0 IS HELD
620                  /  BY THE CURRENT INTERFACE, BOOT SEEKS TO DRIVE 0, CYLINDER 0, HEAD
621                  /  AND SECTOR SPECIFIED IN COMMAND WORD.  IT THEN READS UNTIL THE
622                  /  INTERFACE SIGNALS END OF DATA.  IF THE DRIVE IS HELD BY ANOTHER
623                  /  INTERFACE, BOOT RETURNS TO POLL WITH THE COLD LOAD COMMAND STILL
624                  /  PENDING ON THE INTERFACE.
625                  /
626 0215 34100274 BOOT   REGO =       CMP  COMWD       / SAVE HD-SECT, CLEAR UPPER BYTE.
627 0216 27726377       COMWD = 377  PASS      ,L      / FORCE UNIT 0.
628 0217 66001317          CA : UNIT                   / TRY TO SELECT IT.  RETURN
629                                                    /   ONLY IF IT'S AVAILABLE.
630 0220 36414660       CYLAD =       PASS STATR       / SET CYLINDER 0 (CLRST HAS BEEN
631                                                    /   CALLED AND UNIT # = 0).
632 0221 37704340        HSAD =       PASS REGO ,SL    / SHIFT HEAD NUMBER
633 0222 37704350        HSAD =       PASS HSAD ,SL    /   INTO UPPER BYTE.
634 0223 37724130        HSAD =       PASS HSAD ,L,SR  / RESTORE SECTOR NUMBER
635 0224 37724130        HSAD =       PASS HSAD ,L,SR  /   IN LOWER BYTE.
636 0225 66000247          CA : SEEK1                  / DO THE SEEK.
637 0226 07625403          DO : FLMSK  ←   3     ,L    / AUTO TRACK SPARING, SURFACE MODE,
638                                                    /   NO AUTO INCR-DECR SEEK.
639 0227 07622453          DO : IFNBF  ←   SRTRY,L    / SET THE RETRY COUNTER ALSO.
640 0230 07621570          DO : IDTBF  ←   170   ,L    / NO RETRIES ALLOWED, 3000 BOOT
641 0231 66001630          CA : ICLCK                  /   HAS NO PROVISION FOR THEM.
642 0232 64000410          BR : RBOOT                  / NOW READ THE SECTOR(S).
```

```
643                          RECALIBRATE, SEEK COMMANDS, SEEK1, SEEK2, SEEK3
644                  /
645                  /
646                  /
647                  /
648                  /  THE RECALIBRATE (RESTORE HOME) COMMAND ADDRESSES THE UNIT AND
649                  /  CAUSES IT TO RESET ITSELF TO CYLINDER 0.  IT THEN RELEASES
650                  /  THE UNIT.  NOTE:  THE CPU MUST WAIT FOR AN INTERRUPT RESPONSE
651                  /  AFTER OUTPUTTING THIS COMMAND, ELSE THE COMMAND MAY BE
652                  /  OVERWRITTEN ON THE INTERFACE BEFORE THE CONTROLLER PICKS IT UP.
653                  /
654 0233 66001317 RECAL    CA : UNIT                    / TRY TO CONNECT DRIVE.
655 0234 66001547          CA : SWAIT                   / SEE IF THE DRIVE WORKS.
656 0235 07624004          DO : TGBUS  ←   RCCOM,L      / YES, PUT OUT RECALIBRATE COMMAND.
657 0236 66001641          CA : STROB
658 0237 66001674          CA : WTIO                    / UPDATE THE TIO REGISTER.
659 0240 64000040          BR : POLL1                   / POLL INTERFACES WHILE WAITING.
660                  /
661                  /
662                  /  THE SEEK COMMAND ACCEPTS TWO ADDRESS WORDS FROM
663                  /  THE CPU.  THE FIRST IS THE CYLINDER ADDRESS (16
664                  /  BITS) AND THE SECOND IS THE HEAD (UPPER BYTE)-
665                  /  SECTOR (LOWER BYTE) ADDRESS.  THEY ARE STORED
666                  /  IN CONTROLLER REGISTERS CYLAD AND HSAD AS WELL AS
667                  /  BEING SENT TO THE DISC DRIVE.
668                  /
669 0241 66001317 SEEK     CA : UNIT                    / TRY TO CONNECT DRIVE.
670 0242 66000272          CA : RCDRD                   / GET CYLAD AND HSAD.
671 0243 66000247          CA : SEEK1                   / GO OUTPUT THE SEEK.
672 0244 66001674          CA : WTIO                    / UPDATE 3000 STATUS REGISTER.
673 0245 66001623          CA : SRQ                     / SRQ 3000 FOR NEXT ORDER.
674 0246 64000040          BR : POLL1                   / RETURN TO POLL LOOP.
675                  /
676                  /
677                  /  SUBROUTINE SEEK1 DOES THE ACTUAL SEEK COMMAND FOR THE SEEK AND
678                  /  COLD LOAD READ COMMANDS.  IT IS SEPARATED OUT SO THAT THE
679                  /  AUTOMATIC TRACK SWITCHING ROUTINES CAN CALL IT AND THEN WAIT
680                  /  FOR SEEK COMPLETION.  THE USER-CALLABLE SEEK COMMAND PROCESSOR
681                  /  (MIDDLE OF PAGE) RETURNS TO THE POLL LOOP AFTER THE SEEK IS
682                  /  ISSUED.
683                  /    SEEK1 IS A TWO-LEVEL SUBROUTINE WITH THREE ENTRY POINTS,
684                  /  SEEK1, SEEK2, AND SEEK3.  SEEK1 CHECKS DISC STATUS, THEN
685                  /  SEEKS TO CYLAD AND HSAD.  SEEK2 AND SEEK3 ARE ONE-LEVEL
686                  /  SUBROUTINES USED BY "SPARE" AND "DECS2".
687                  /
688 0247 66001551 SEEK1    CA : GSTAT                   / SEE IF DRIVE
689 0250 27422002       DUMMY = 2      AND  BUF2 ,L     /   WILL RESPOND
690 0251 44400000          CC = TNZRO                   /   AND IF NOT,
691 0252 64001761          BR : STER                    /   SET STATUS-2 ERROR AND QUIT.
692 0253 15201311          DO : CBUF   ←   CYLAD,C      / PUT OUT CYLINDER ADDRESS
693 0254 07624007          DO : TGBUS  ←   SEEKC,L      /   AND SEEK COMMAND
694 0255 66001641          CA : STROB                   /   AND STROBE IT OUT.
695 0256 07603777          DO : NWTRK  ←   FALSE,U      / CLEAR NEW TRACK FLAG HERE (FOR
696                                                     /   BOOT CMD, NO CLR FOR DECS2).
697 0257 66000320          CA : GDTYP                   / GET DRIVE TYPE FOR OUTA1.
698 0260 37616270 SEEK2  BUF3 =       PASS HSAD        / SET UP FOR OUTA1, SEEK3.
699 0261 07605777 SEEK3    DO : EOC    ←   FALSE,U      / CLEAR END-OF-CYLINDER FLAG.
700 0262 07624006          DO : TGBUS  ←   ADREC,L      / DRIVE ADDRESS RECORD COMMAND.
701 0263 64001727          BR : OUTA1                   / SEND IT TO DRIVE.
```

```
                        ADDRESS RECORD COMMAND, TERM, RCORD
                /
                /
                /
                /
                /  ACCEPTS TWO ADDRESS WORDS AS IN "SEEK" COMMAND, BUT PLACES
                /  THEM ONLY IN CYLAD AND HSAD (NOT TRANSMITTED TO DRIVE).
                /
0264 66001610 RECRD     CA : SETUP                  / CLEAR STATUS, SET CONTROLLER BUSY.
0265 66000272           CA : RCORD                  / GET AND STORE CYLAD AND HSAD.
                                                    / FALL INTO TERM.
                /
                /
                /  TERM IS THE NORMAL COMPLETION POINT OF CONTROLLER OPERATIONS
                /  WHICH CANNOT ALLOW THE CURRENT CPU TO BE DISCONNECTED, NAMELY
                /  ALL OPERATIONS EXCEPT "END" COMMANDS OR WAITING FOR "SEEK" AND
                /  "RECALIBRATE COMMANDS TO BE COMPLETED.  THE "END" COMMAND ALLOWS
                /  THE SOFTWARE DRIVER TO DETERMINE WHEN TO RELEASE THE CONTROLLER
                /  AND LET IT RESUME POLLING OF OTHER INTERFACES.  OPERATIONS WHICH
                /  COMPLETE AT TERM REQUIRE A 3000 SRQ BUT NO 2100 DATA FLAG.
                /
0266 66000424 TERM      CA : CKXFR                  / CHECK FOR 3000 CHANNEL ERROR.
0267 66001702           CA : TRNOF                  / CLOCKS OFF, WRT TID, CLEAR BUSY.
0270 66001623           CA : SRQ                    / SRQ THE 3000 CHANNEL.
0271 64000141           BR : CMDWT                  / GO WAIT FOR NEXT COMMAND.
                /
                /
                /  SUBROUTINE RCORD DOES THE ACTUAL WORK OF THE ADDRESS RECORD
                /  COMMAND.  IT IS SEPARATED OUT BECAUSE IT IS ALSO CALLED BY SEEK.
                /
0272 07605777 RCORD     DO : EOC    ←   FALSE,U     / CLEAR THE END-OF-CYLINDER
0273 07603777           DO : NWTRK  ←   FALSE,U     /   AND NEW TRACK FLAGS.
0274 66001654           CA : GET                    / GET CYLAD FROM CPU.
0275 34014264       CYLAD =       CMP   BUF1
0276 66001654           CA : GET                    / GET HSAD FROM CPU.
0277 34104264       HSAD =        CMP   BUF1
0300 72000000           RT : 0
```

```
                        STATUS COMMAND FORMATS
                /
                /
                /
                /  RETURNS ENCODED TERMINATION STATUS AND UNIT NUMBER IN THE
                /  FOLLOWING FORMAT:
                /
                /       1 1 1 1 1 1
                /       5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0   BIT (2100 NUMBERING)
                /       - - - - - - - - - - - - - - - -
                /      !S P D E E E E E         U U U U!
                /       - - - - - - - - - - - - - - - -
                /
                /       WHERE S = SPARE TRACK.
                /             P = PROTECTED TRACK.
                /             D = DEFECTIVE TRACK.
                /             E = ENCODED TERMINATION STATUS.
                /   00  (0000UU)  NO ERROR
                /   01  (000400)  ILLEGAL OPCODE
                /   02  (0010UU)  DRIVE AVAILABLE STATUS (***NOT AN ERROR***)
                /   07  (0034UU)  CYLINDER COMPARE ERROR
                /   10  (0040UU)  UNCORRECTABLE DATA ERROR
                /   11  (0044UU)  HEAD-SECTOR COMPARE ERROR
                /   12  (005000)  I-O PROGRAM ERROR (3000 ONLY)
                /   14  (0060UU)  END OF CYLINDER
                /   16  (0070UU)  DATA OVERRUN
                /   17  (0074UU)  (AFTER DATA OP OR VRFY):  POSS CORRECTABLE DATA ERR
                /   17  (0074UU)  (AFTER REQUEST SYNDROME):  ERROR WAS CORRECTED
                /   20  (1100UU)  ATTEMPT TO DIRECTLY ACCESS SPARE TRACK
                /   21  (0304UU)  TRACK FLAGGED DEFECTIVE & AUTO SPARING NOT ALLOWED
                /   22  (0110UU)  ACCESS NOT READY DURING DATA OPERATION
                /   23  (0114UU)  DISC DRIVE STATUS ERROR
                /   26  (0130UU)  ATTEMPT TO WRITE ON PROT TRACK & FORMAT SWITCH OFF
                /   27  (0134UU)  UNIT UNAVAILABLE
                /   37  (0174UU)  DRIVE ATTENTION
                /
                /            UU = UNIT NUMBER.
                /
                /
                /   RETURNS CURRENT DRIVE STATUS IN THE FOLLOWING FORMAT:
                /
                /      1 1 1 1 1 1
                /      5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0   BIT (2100 NUMBERING)
                /      - - - - - - - - - - - - - - - -
                /     !E     DRV TYP S S S S S S S S!
                /      - - - - - - - - - - - - - - - -
                /
                /     WHERE S = DYNAMIC UNIT STATUS:
                /               BIT 0  *DRIVE BUSY
                /                   1  *DRIVE NOT READY
                /                   2  *SEEK CHECK
                /                   3   FIRST STATUS
                /                   4  *FAULT
                /                   5   FORMAT
                /                   6   PROTECTED
                /                   7   ATTENTION
                /
                /           DRV TYP = 4 BITS USED TO DETERMINE LAST AVAIL HD & SECT.
                /           E = ERROR (TRUE IF ANY BIT MARKED "*" IS TRUE).
```

```
                        STATUS COMMAND
                /
                /
                /
                /
0301 66001341 STATS    CA : UNIT1                    / GET UNIT, SET BUSY.
0302 35302660        BUF1 =       CMP  STATR         / MAKE STATUS +TRUE FOR CPU.
0303 66001304          CA : PUT                      /   THEN GIVE IT AWAY.
0304 66001551          CA : GSTAT                    / GET DRIVE STATUS.
0305 27422027       DUMMY = 27    AND  BUF2 ,L
0306 44440000          CC = TNZRO          ,RS      / ...ANY ERROR?
0307 64000311          BR : NOERR
0310 26610200        BUF2 = 200   OR   BUF2 ,U      / YES, SET ERROR BIT (BIT 15).
0311 35302260 NOERR  BUF1 =       CMP  BUF2         / NO, MAKE +TRUE FOR CPU.
0312 66001304          CA : PUT
0313 66001651          CA : CLRST
0314 66001751          CA : USTAT                    / PUT CURRENT UNIT# IN STATR,L.
0315 07621375          DO : CBUF   ←   CFST ,L       / CLEAR FIRST STATUS.
0316 66001606          CA : CST
0317 64000371          BR : RQSC2                    / WAIT FOR CPU TO ACCEPT 2ND WORD.
```

```
818                          SUBROUTINE GDTYP
819                 /
820                 /
821                 /
822                 /
823                 /    SUBROUTINE GDTYP RETRIEVES THE DRIVE TYPE WORD FROM RAM(12)
824                 /  (PLACED THERE BY SUBROUTINE "UNIT").  ON EXIT BUF2,U CONTAINS
825                 /  -(LAST AVAILABLE SURFACE) (TWO'S COMPLEMENT) AND BUF2,L
826                 /  CONTAINS -(LAST AVAILABLE SECTOR) (TWO'S COMPLEMENT).
827                 /    THE MAPPING IS IMPLEMENTED AS A VECTOR SO THAT OTHER DRIVES
828                 /  MAY BE EASILY ADDED.  AS OF THIS DATE, THE 7925 (DRIVE TYPE
829                 /  WORD = 3, LAST AVAIL HEAD = 8, LAST AVAIL SECTOR = 63), THE
830                 /  7905A (DRIVE TYPE WORD = 2, LAH = 2, LAS = 47), THE 7920 (DRIVE
831                 /  TYPE WORD = 1, LAH = 4, LAS = 47), THE 7905B (DRIVE TYPE WORD = 0,
832                 /  LAH = 3, LAS = 47), AND TWO ADDITIONAL LARGER CAPACITY DRIVES
833                 /  (DRIVE TYPE WORD = 5, LAH AND LAS NOT DEFINED, AND DRIVE TYPE
834                 /  WORD = 4, LAH = 14, LAS = 63) ARE DEFINED.  NOTE THAT CHANGES TO
835                 /  ONE DRIVE TYPE (OR THE ADDITION OF NEW DRIVE TYPES) MAY AFFECT
836                 /  TABLE ENTRIES FOR OTHER DRIVE TYPES.  BE WARNED!!!!!!
837                 /    NOTE:  THIS CODING MUST RESIDE IN ADDRESSES <= 377B.
838                 /
839 0320 07620014 GDTYP    DO : RAMAD  ←   DRTYP,L     / RETRIEVE THE
840 0321 10020442          DI : BUF2   ←   RAM  ,L     /   DRIVE TYPE WORD.
841 0322 36430140        BUF2 =       PASS BUF2 ,L,SL / MUST DOUBLE BECAUSE DRIVE
842 0323 21560343        REGO = DVECT SUB  BUF2 ,L     /   VECTOR ENTRIES ARE IN PAIRS.
843 0324 27610000        BUF2 = 0     PASS       ,U     / MUST CLEAR BUF2 FOR
844 0325 27630000        BUF2 = 0     PASS       ,L     /   TABLE TO WORK PROPERLY.
845 0326 15206241          DO : SPOBF  ←   BUF2 ,C     / CLEAR SPOBF IN CASE OF WRITE.
846 0327 34020000          NO : REGO            ,L
847 0330 74000000          BR : I                       / BRANCH TO DRIVE VECTOR.
848                 /
849 0331 22210000        BUF2 = 0     PLUS BUF2 ,U     / DTW = 5, LAST HEAD UNDEFINED,
850 0332 22230000        BUF2 = 0     PLUS BUF2 ,L     /   LAST SECTOR UNDEFINED.
851 0333 22210372        BUF2 = 372   PLUS BUF2 ,U     / DTW = 4, LAST HEAD = 14,
852 0334 22230000        BUF2 = 0     PLUS BUF2 ,L     /   LAST SECTOR = 63.
853 0335 22210372        BUF2 = 372   PLUS BUF2 ,U     / 7925 -- LAST HEAD = 8,
854 0336 22230360        BUF2 = 360   PLUS BUF2 ,L     /   LAST SECTOR = 63.
855 0337 22210002        BUF2 = 2     PLUS BUF2 ,U     / 7905A -- LAST HEAD = 2,
856 0340 22230000        BUF2 = 0     PLUS BUF2 ,L     /   LAST SECTOR = 47.
857 0341 22210377        BUF2 = 377   PLUS BUF2 ,U     / 7920 -- LAST HEAD = 4,
858 0342 22230000        BUF2 = 0     PLUS BUF2 ,L     /   LAST SECTOR = 47.
859 0343 22210375 DVECT  BUF2 = 375   PLUS BUF2 ,U     / 7905B -- LAST HEAD = 3,
860 0344 22230321        BUF2 = 321   PLUS BUF2 ,L     /   LAST SECTOR = 47.
861 0345 72000000          RT : 0
```

```
862                          REQUEST SECTOR COMMAND
863                  /
864                  /
865                  /
866                  /
867                  /  THIS COMMAND RETURNS THE ADDRESS OF THE SECTOR
868                  /  OVER WHICH THE HEADS ARE POSITIONED WHEN THE
869                  /  COMMAND IS ISSUED.  THE DISC DRIVE RETURNS THE PHYSICAL SECTOR
870                  /  ADDRESS, WHICH WE MUST CONVERT TO THE LOGICAL ADDRESS BEFORE
871                  /  SENDING IT OFF TO THE CPU.
872                  /
873 0346 66001651 RQSCT     CA : CLRST                    / CLEAR THE STATUS REGISTER.
874 0347 66001751          CA : USTAT                    / PUT CURRENT UNIT# IN STATR,L,
875                                                       /   CHECK FOR LEGAL DRIVE REQUEST.
876 0350 66001341          CA : UNIT1                    / CONNECT THE DESIRED DRIVE.
877 0351 27422002       DUMMY = 2      AND   BUF2 ,L     / CONTINUE ONLY IF DRIVE IS READY.
878                                                       /   (BUF2 HAS DRY STAT AFTER UNIT1).
879 0352 44400000          CC = TNZRO
880 0353 64001761          BR : STER                     / NOT READY, ABORT AND REPORT IT.
881 0354 66000375          CA : RQSC3                    / O.K., GET PHYSICAL HEAD AND SECTOR.
882 0355 37616210        BUF3 =         PASS HSAD ,SW    / PUT LOGICAL HEAD IN BUF3,L.
883 0356 27437437        BUF3 = 37      AND   BUF3 ,L    / KILL ANY S, P, OR D BITS.
884 0357 31477464        BUF3 = BUF1    SUB   BUF3 ,L    / TAKE SECTOR - HEAD.
885 0360 44100000          CC = LOVER                    / IS THAT < 0?
886 0361 64000366          BR : RQSC1                    / NO, SAFE TO SEND IT.
887 0362 66000320          CA : GDTYP                    / YES, SO MUST ADD BACK
888 0363 35230060        BUF2 =         CMP   BUF2 ,L    /   NUMBER OF SECTORS PER TRACK.
889 0364 22322002        BUF1 = 2       PLUS  BUF2 ,L    / BUF1 = # SECTORS PER TRACK.
890 0365 32237464        BUF3 = BUF1    PLUS  BUF3 ,L
891 0366 27702377 RQSC1  BUF1 = 377     PASS        ,U   / ZERO TOP HALF OF BUF1.
892 0367 35323460        BUF1 =         CMP   BUF3 ,L    / COMPLEMENT LOW BUF3 INTO BUF1.
893 0370 66001304          CA : PUT                      / GIVE IT TO CPU.
894 0371 07600500 RQSC2    DO : TIMER  ←    START,U      / RUN TIMER WHILE
895 0372 42100000 RQSC5    CC = DTRDY                    /   WAITING FOR CPU
896 0373 64000372          BR : RQSC5                    /   TO ACCEPT DATA.
897 0374 64000266          BR : TERM                     / TERM CALLS TRHOF TO STOP TIMER.
898                  /
899                  /
900                  /  SUBROUTINE RQSC3 IS SEPARATED OUT FROM THE REST OF THE REQUEST
901                  /  SECTOR COMMAND SO THAT THE INITIALIZE COMMAND MAY ALSO USE IT
902                  /  TO GET THE PHYSICAL HEAD ADDRESS FROM THE DRIVE.
903                  /
904 0375 07624011 RQSC3     DO : TGBUS  ←    RSCOM,L      / SEND "REQUEST SECTOR" TO DISC.
905 0376 66001642          CA : STRB1                    / VALIDATE IT, LEAVE STROBE ON.
906 0377 07600500          DO : TIMER  ←    START,U      / TURN ON TIME-OUT CIRCUIT.
907 0400 10102204 RQSC4    DI : BUF1   ←    CBSIN        / READ ADDRESS.
908 0401 10002202          DI : BUF2   ←    CBSIN        / READ AGAIN IN CASE IT CHANGED.
909 0402 36202264       DUMMY = BUF1   XOR   BUF2
910 0403 44400000          CC = TNZRO
911 0404 64000400          BR : RQSC4                    / IF SO, TRY AGAIN.
912 0405 34102264        BUF1 =         CMP   BUF1       / COMPLEMENT FOR MICROPROCESSOR,
913 0406 64001647          BR : STRB2                    /   TURN OFF STROBE, AND RETURN.
```

```
914                          READ, RNVFY COMMANDS, SUBROUTINES DSTRT, DSTR1, CKXFR, REDE1
915                  /
916                  /
917                  /
918                  /
919                  /  THE READ COMMAND READS ONE SECTOR AND PASSES THE DATA TO THE CPU.
920                  /  RBOOT IS THE ENTRY POINT USED BY THE COLD LOAD READ COMMAND, AND
921                  /  READ1 IS THE ENTRY POINT USED BY THE READ-WITHOUT-VERIFY COMMAND.
922                  /
923 0407 66001317 READ      CA : UNIT                    / TRY TO CONNECT DRIVE.
924 0410 66001707 RBOOT     CA : STRT1                   / SET PROPER DISC ADDRESS.
925 0411 66000422 READ1     CA : DSTRT                   / GO SET UP READ DATA PATH.
926 0412 66000430 READ2     CA : REDE1                   / THEN GO READ A SECTOR.
927 0413 66000771           CA : EWRF1                   / SET UP NEXT SECTOR ADDRESS,
928                                                      /   CHECK FOR END OF TRANSFER.
929 0414 66001503           CA : SECTR                   / NOT THE END, ACCESS NEXT SECTOR.
930 0415 64000411           BR : READ1                   / THEN GO BACK AND READ IT.
931                  /
932                  /
933                  /  THIS COMMAND READS THE ADDRESSED SECTOR WITHOUT VERIFICATION
934                  /  OF THE PRECEDING SECTOR.  IT IS INTENDED AS THE CONTINUATION
935                  /  PORTION OF A WRAPAROUND READ AND MUST BE USED WITH CAUTION
936                  /  SINCE NO SAFEGUARDS AGAINST ADDRESS OR TRACK STATUS ERRORS ARE
937                  /  PROVIDED.  THE ADDRESS AND TRACK STATUS WILL BE VERIFIED,
938                  /  HOWEVER, IF THE TRANSFER CONTINUES BEYOND A TRACK BOUNDARY.
939                  /
940 0416 66001317 RNVFY     CA : UNIT                    / TRY TO CONNECT DRIVE.
941 0417 66001714           CA : STRT2                   / SET PROPER DISC ADDRESS.
942 0420 07623377           DO : INITL  ←   FALSE,L      / FORCE VERIFY IF WE SWITCH TRACKS.
943 0421 64000411           BR : READ1                   / NOW GO READ.
944                  /
945                  /
946                  /  SUBROUTINE DSTRT SETS UP THE READ DATA PATH TO THE INTERFACE.
947                  /  IT IS SEPARATED OUT SO THAT IFNBF PROP TIME ONLY OCCURS AT
948                  /  THE START OF TRANSFER, NOT BETWEEN SECTORS.  THE ENTRY POINT
949                  /  AT DSTR1 IS USED BY THE INIT, WRITE, AND WFS ROUTINES.  FOR
950                  /  THESE ROUTINES, IFNBF IS SET TO IFOUT.
951                  /
952 0422 07622443 DSTRT     DO : IFNBF  ←   IFIN ,L      / SET UP INTERFACE DATA PATH.
953 0423 66001631 DSTR1     CA : ICLK1                   / VALIDATE COMMAND, LEAVE IT ON.
954                                                      / FALL INTO CKXFR.
955                  /
956                  /
957                  /  SUBROUTINE CKXFR CHECKS THE 3000 I-O CHANNEL ERROR FLAG.
958                  /
959 0424 42340000 CKXFR     CC = XFRNG            ,RS    / ANY ERROR IN THE 3000 CHANNEL?
960 0425 72000000           RT : 0                       / NO, RETURN.
961 0426 66001651           CA : CLRST                   / YES, CLEAR THE STATUS-1 WORD.
962 0427 64001766           BR : IOPER                   / GO INTERRUPT - I-O PROG ERR.
963                  /
964                  /
965                  /  SUBROUTINE REDE1 TRANSFERS ONE SECTOR FROM DISC TO CPU.  IT
966                  /  IS CALLED AFTER THE DATA PATH IS SET UP BY DSTRT.
967                  /
968 0430 66000530 REDE1     CA : READ3                   / READ USING VRFY CODE.
969                                                      / THEN FALL INTO VCHK.
```

```
 970                          SUBROUTINES VCHK, CLROF; RETRYABLE ERROR STATUS SIEVE
 971                 /
 972                 /
 973                 /
 974                 /
 975                 /  DATA TRANSFER IS FINISHED.  NOW CHECK FOR ERRORS.
 976                 /
 977 0431 44040000 VCHK     CC = UNCOR            ,RS   / THE ULTIMATE SIN--
 978 0432 64000437          BR : UDTER                  /   UNCORRECTABLE.
 979 0433 43300000          CC = ANYER                  / ANY OTHER ERROR (CORRECTABLE)?
 980 0434 72000000          RT : 0                      / NO, RETURN.
 981                                                    / YES, FALL INTO RETRY SIEVE.
 982                 /
 983                 /
 984                 /    ON DATA ERROR OR OVERRUN, THE SIGNAL DVEND (DEVICE END)
 985                 /  INCREMENTS A RETRY COUNTER ON THE 3000 INTERFACE AND CAUSES
 986                 /  THE INTERFACE TO SIGNAL "DEVICE END" TO THE CHANNEL.
 987                 /  EXCEPTION:  IF THE COUNTER ROLLS OVER (OUT OF RETRIES) THE
 988                 /  INTERFACE INTERRUPTS THE CPU.  ON A 2100 INTERFACE, DVEND
 989                 /  SETS THE FLAG, ALWAYS CAUSING AN INTERRUPT.
 990                 /    THE ENTRY POINT AT NRTER IS FROM THE NON-RETRYABLE ERROR
 991                 /  SIEVE.  WE CLEAR ANY DRIVE OFFSET AT THIS POINT, SINCE WE
 992                 /  MAY HAVE HAD AN ERROR ABORT FROM A READ WITH OFFSET COMMAND.
 993                 /  CLEARING THE DRIVE OFFSET GENERATES A DRIVE ATTENTION, SO
 994                 /  WE WAIT FOR THAT AND THEN CLEAR IT TO AVOID CONFUSING THE
 995                 /  USER WITH A POSSIBLE EXTRA ATTENTION.
 996                 /
 997 0435 22212401 CDTER STATR = 1      PLUS STATR,U     / CORRECTABLE DATA ERROR    (17).
 998 0436 22212406 OVERR STATR = 6      PLUS STATR,U     / DATA OVERRUN              (16).
 999 0437 22212410 UDTER STATR = 10     PLUS STATR,U     / UNCORRECTABLE DATA ERROR  (10).
1000 0440 66001702          CA : TRNOF                  / PUT EVERYTHING IN A KNOWN STATE.
1001 0441 07622672          DO : IFNBF  ←   DVEND,L     / THEN BUMP COUNTER ON 3000 INTERFACE
1002                                                    /   (OR SET FLAG ON 2100 INTERFACE).
1003 0442 66001630          CA : ICLCK                  / VALIDATE THE COMMAND.
1004 0443 66001623          CA : SRQ                    / NOW SRQ 3000 FOR NEXT ORDER.
1005 0444 66001604 NRTER    CA : SWAT3                  / CLEAR ATTENTION BIT.
1006 0445 66000454          CA : CLROF                  / CLEAR ANY DRIVE OFFSET.
1007 0446 07600500          DO : TIMER  ←   START,U     / TURN ON TIME-OUT CIRCUIT.
1008 0447 66001551 WTATN    CA : GSTAT                  / WAIT FOR ATTENTION BIT
1009 0450 27422202       DUMMY = 202    AND  BUF2 ,L    /   FROM DRIVE (ONLY IF
1010 0451 44440000          CC = TNZRO            ,RS   /   DRIVE IS READY).
1011 0452 64000447          BR : WTATN
1012 0453 64000140          BR : ATTN4                  / GO CLEAR IT AND WAIT
1013                                                    /   FOR THE NEXT COMMAND.
1014                /
1015                /
1016                /  SUBROUTINE CLROF CLEARS THE DRIVE OFFSET.  IT IS CALLED FOR
1017                /  NORMAL COMPLETION OF THE ROFST COMMAND, AND IS ALSO CALLED
1018                /  BY THE ERROR INTERRUPT PROCESSOR IN CASE A READ-WITH-OFFSET
1019                /  COMMAND WAS ABORTED BEFORE NORMAL COMPLETION.
1020                /
1021 0454 07624002 CLROF     DO : TGBUS  ←   OFFST,L     / PUT OUT OFFSET COMMAND.
1022 0455 07621377          DO : CBUF   ←   377  ,L     / OFFSET PARAMETER = 0.
1023 0456 07602777          DO : CLKOF  ←   377  ,U     / ALSO TURN OFF SEPARATOR OFFSET.
1024 0457 64001641          BR : STROB
```

```
1025                            READ WITH OFFSET COMMAND
1026                    /
1027                    /  THIS COMMAND REQUESTS AN OFFSET PARAMETER FROM THE CPU AND
1028                    /  COMMANDS THE DRIVE TO OFFSET THE HEADS FROM TRACK CENTER.
1029                    /  THE OFFSET IS REMOVED AT COMMAND COMPLETION, IF A CONTROLLER-
1030                    /  GENERATED SEEK OCCURS (AUTOMATIC TRACK SWITCHING AT END OF
1031                    /  TRACK OR SPARE TRACK ACCESS), OR IF AN ERROR OCCURS.  THIS COMMAND IS
1032                    /  USED ONLY IN ERROR RE-READ SITUATIONS.  NOTE THAT SPARE TRACKS CANNOT
1033                    /  BE READ WITH OFFSET   THE REASONING IS AS FOLLOWS:
1034                    /    1.  A SEEK BY THE DRIVE REMOVES ANY OFFSET, THEREFORE TO READ A SPARE
1035                    /  TRACK WITH OFFSET THE OFFSET MUST BE APPLIED AFTER THE SPARE TRACK IS
1036                    /  REACHED.
1037                    /    2.  THE OFFSET PARAMETER MUST BE READ IN BEFORE THE STRT1 ROUTINE IS
1038                    /  CALLED.  THIS IS BECAUSE THE SOFTWARE EXPECTS TO SEE A HANDSHAKE FLAG
1039                    /  ON THE INTERFACE WHEN THE OFFSET PARAMETER IS ACCEPTED, SO THAT IT CAN
1040                    /  PROPERLY START UP DMA.  SINCE THERE ARE ERROR INTERRUPT EXITS FROM STRT1
1041                    /  (WHICH ALSO SET THE INTERFACE FLAG), THE SOFTWARE WILL ASSUME THIS ERROR
1042                    /  FLAG IS REQUESTING THE PARAMETER RATHER THAN TERMINATING THE COMMAND,
1043                    /  UNLESS THE PARAMETER HAS ALREADY BEEN READ IN.
1044                    /    3.  THE STRT1 ROUTINE (AND THE SUBROUTINES CALLED BY STRT1) USE ALL 8
1045                    /  WORKING REGISTERS IN THE MICROPROCESSOR.  THERE IS NO PLACE TO STORE THE
1046                    /  OFFSET PARAMETER DURING STRT1.
1047                    /    4.  THEREFORE THE OFFSET PARAMETER MUST BE REQUESTED FROM THE INTERFACE
1048                    /  AND SENT TO THE DRIVE BEFORE STRT1 IS CALLED, TO REMOVE THE INTERFACE
1049                    /  FLAG AMBIGUITY.  BUT THIS MEANS THAT ANY CONTROLLER-GENERATED SEEK TO A
1050                    /  SPARE TRACK WILL REMOVE THE OFFSET.  HENCE SPARE TRACKS CANNOT BE READ
1051                    /  WITH OFFSET.
1052                    /
1053 0460 66001317 RDFST     CA : UNIT                   / TRY TO CONNECT DRIVE.
1054 0461 66001654           CA : GET                    / GET OFFSET FROM CPU.
1055 0462 66001623           CA : SRQ                    / SRQ FOR READ ON 3000.
1056 0463 66001315           CA : DFLST                  / HANDSHAKE THE 2100 ALSO.
1057 0464 66001547           CA : SWAIT                  / COMPLETE SEEK BEFORE OFFSETTING.
1058 0465 17602445           DO : CLKOF  ←   BUF1 ,U     / SET UP READ CLOCK OFFSET.
1059 0466 07624002           DO : TGBUS  ←   OFFST,L     / PREPARE OFFSET COMMAND
1060 0467 17600705           DO : CBUF   ←   BUF1        /   AND THE PARAMETER.
1061 0470 66001641           CA : STROB                  / SEND THEM TO THE DRIVE.
1062 0471 66001547           CA : SWAIT                  / WAIT FOR DRIVE TO FINISH OFFSET.
1063 0472 66001707           CA : STRT1                  / SET PROPER DISC ADDRESS.
1064 0473 66000422 RDFS1     CA : DSTRT                  / GO SET UP READ DATA PATH.
1065 0474 66000430           CA : REDE1                  / NOW GO READ ONE SECTOR
1066 0475 66001461           CA : INCHS                  / THEN SET UP NEXT ADDRESS.
1067 0476 42600000           CC = EOD                    / WAS THAT ALL?
1068 0477 64000502           BR : RDFS2                  / YES, GO CLEAR OFFSET.
1069 0500 66001503           CA : SECTR                  / NO, SET UP NEXT SECTOR
1070 0501 64000473           BR : RDFS1                  /   AND GO READ IT.
1071                    /
1072 0502 66000454 RDFS2     CA : CLROF                  / ALL DONE, SO CLEAR OFFSET.
1073 0503 66001547           CA : SWAIT                  / WAIT FOR OFFSET TO COMPLETE
1074                                                     /   THEN FALL INTO RDFS3.
1075                    /
1076                    /
1077                    /  RDFS3 NOTIFIES THE CPU THAT THE OPERATION IS COMPLETE.  THE
1078                    /  HP3000 INTERFACE SERVICE REQUESTS, AND THE 2100 INTERFACE SETS ITS
1079                    /  FLAG.
1080                    /
1081 0504 07606001 RDFS3     DO : IFVLD  ←   CLRCL,U     / TURN OFF INTERFACE FUNCTION.
1082 0505 66001315           CA : DFLST                  / 2100 FLAG SET HERE.
1083 0506 64000266           BR : TERM                   / 3000 IS SRQ'ED AT TERM.
```

```
1084                          VERIFY COMMAND, SUBROUTINES VRFY, VRFY1, VRFY2, READ3
1085                 /
1086                 /
1087                 /
1088                 /
1089                 /  VRFCM PERFORMS AN ADDRESS VERIFICATION
1090                 /  AND CYCLIC CHECK ON EACH SECTOR UNTIL SCNT = 0.
1091                 /  SCNT IS REQUESTED FROM THE CPU AT THE BEGINNING
1092                 /  OF THE OPERATION.
1093                 /
1094                 /
1095 0507 66001317 VRFCM    CA : UNIT                    / TRY TO CONNECT DRIVE.
1096 0510 66001654          CA : GET                     / GET SECTOR COUNT FROM CPU.
1097 0511 34106264        SCNT =       CMP  BUF1
1098 0512 66001707          CA : STRT1                   / SET PROPER DISC ADDRESS.
1099 0513 66000527 VLOOP    CA : VRFY1                   / NOW VERIFY EACH SECTOR IN TURN,
1100 0514 66000431          CA : VCHK                    /   CHECKING DATA ERRORS.
1101 0515 66001461          CA : INCHS                   / SET UP NEXT SECTOR.
1102 0516 30106274        SCNT =       DEC  SCNT        / THEN SEE IF DONE.
1103 0517 44440000          CC = TNZRO           ,RS
1104 0520 64000504          BR : ROFS3                   / ...YES, DO HOUSEKEEPING.
1105 0521 66001503          CA : SECTR                   / ...NO, GET READY TO CONTINUE
1106 0522 64000513          BR : VLOOP                   / AND DO THE NEXT ONE.
1107                 /
1108                 /
1109                 /
1110                 /
1111                 /  THE FOLLOWING SUBROUTINE IS USED TO RECOVER DATA FROM THE DISC
1112                 /  DURING ALL READ AND VERIFY ROUTINES EXCEPT READ FULL SECTOR.
1113                 /  FOUR ENTRY POINTS ARE USED, DEPENDING ON JUST HOW MUCH THE
1114                 /  ROUTINE IS TO DO OTHER THAN READ THE DISC.  VRFY1 IS THE ENTRY
1115                 /  POINT THAT CAUSES THE ROUTINE TO PASS DATA THROUGH THE HARDWARE
1116                 /  CRC AND ECC REGISTERS WITHOUT ANY SPARING OPERATIONS, BUT UPDATING
1117                 /  TCYLD AND THSAD.  VRFY2 IS THE ENTRY POINT USED FOR SPARING,
1118                 /  SINCE IT SAVES THE TARGET SECTOR ADDRESS AND RESETS THE VERIFY
1119                 /  RETRY COUNTER TO SCRATCH.  VRFY IS THE PRIMARY ENTRY POINT AND
1120                 /  RESETS THE ADDRESS ERROR RETRY BIT AS WELL.  A FOURTH ENTRY POINT
1121                 /  AT READ3 IS CALLED BY ALL THE READ ROUTINES (EXCEPT RFS) TO
1122                 /  TRANSFER DATA TO THE INTERFACE.  THE READ ROUTINES MUST SET UP
1123                 /  THE DATA TRANSFER PATH BEFORE CALLING READ3.  NOTE THAT ENTRY
1124                 /  AT VRFY1 OR EARLIER PROHIBITS ANY DATA TRANSFER BECAUSE IFVLD
1125                 /  IS CLEARED.
1126                 /    FOLLOWING ANY FLAVOR OF VRFY OR READ3, CALLS TO VCHK CHECK
1127                 /  DATA ERRORS DETECTED DURING EXECUTION, AND A CALL TO SPARE WILL
1128                 /  USE THE INFORMATION SAVED BY VRFY TO ACCESS A SPARE TRACK IF
1129                 /  NECESSARY.
1130                 /    TIMING:  DO NOT LESSEN THE NUMBER OF INSTRUCTIONS BETWEEN
1131                 /  READ3 AND READ3+3 OR BETWEEN READ3+3 AND READ3+8.  THESE
1132                 /  INSTRUCTIONS ARE SPACED DELIBERATELY TO ALLOW TGBUS AND
1133                 /  COMMAND BUS PROP TIMES, BUT WITHOUT THE ARTIFICIAL DELAY
1134                 /  INTRODUCED BY A CALL TO STRB1.  THIS CUTS DOWN ON WASTED
1135                 /  TIME BETWEEN SECTORS OF A MULTIPLE-SECTOR TRANSFER.
```

```
1136                          SUBROUTINES VRFY, VRFY1, VRFY2
1137                  /
1138 0523 07623377 VRFY      DO : RETRY  ←   FALSE,L     / CLEAR RETRY ONLY AT 1ST ENTRY.
1139 0524 37702210 VRFY2   BUF1 =       PASS HSAD ,SW   / SAVE SECTOR FIELD OF HSAD.
1140 0525 07620015           DO : RAMAD ←   NSEC ,L     / CLEAR PREVIOUS SECTOR VERIFY
1141 0526 07620777           DO : RAM   ←   377  ,L     /   RETRY COUNTER.
1142 0527 07606001 VRFY1     DO : IFVLD ←   CLRCL,U     / STOP ANY TRANSFER.
1143 0530 07624017 READ3     DO : TGBUS ←   RDCOM,L     / SEND READ COMMAND TO DRIVE.
1144 0531 27722201         WCNT = RWCT  PASS      ,L     / SET UP DATA FIELD LENGTH.
1145 0532 10007304           DI : DUMMY ←   SPIBF       / MAKE SURE EOW IS CLEAR.
1146 0533 07604476           DO : INTCW ←   RCC1 ,U     / READ AND CYCLIC CHECK.
1147 0534 07610226           DO : ECICW ←   RECC ,U     / READ WITH ECC.
1148 0535 07600500           DO : TIMER ←   START,U     / TURN ON TIME-OUT CIRCUIT.
1149 0536 34000000           NO :                       / ALLOW RCC1 TO SET UP AT DRIVE.
1150 0537 34000000           NO :
1151 0540 07604472           DO : INTCW ←   RCC2 ,U     / RCC WITH ACRDY CHECK ENABLED.
1152 0541 42440000           CC = ACRDY          ,RS    / DRIVE ACCESS READY?
1153 0542 64001762           BR : ACRER                 / NO, ESCAPE HERE.
1154 0543 42240000 VSYNC     CC = EOW            ,RS    / YES, WAIT FOR SYNC WORD.
1155 0544 64000543           BR : VSYNC
1156 0545 10107300           DI : REGO  ←   SPIBF       / THIS RESETS EOW BESIDES ALLOW-
1157 0546 17632041           DO : DSYNC ←   REGO ,L     /   ING ECC TO OVERRIDE CRCC.
1158 0547 42240000 VCYLD     CC = EOW            ,RS    / GET AND SAVE CYLINDER ADDRESS.
1159 0550 64000547           BR : VCYLD
1160 0551 10107300           DI : TCYLD ←   SPIBF
1161 0552 34100260         TCYLD =       CMP  TCYLD       / COMPLEMENT TO GROUND TRUE.
1162 0553 42240000 VHSAD     CC = EOW            ,RS    / GET AND SAVE HEAD-SECTOR ADDRESS.
1163 0554 64000553           BR : VHSAD
1164 0555 10007302           DI : THSAD ←   SPIBF
1165 0556 35210260         THSAD =       CMP  THSAD       / COMPLEMENT TO GROUND TRUE.
1166 0557 42240000 VDATA     CC = EOW            ,RS    / READ DATA.
1167 0560 64000557           BR : VDATA
1168 0561 10007316           DI : BUF3  ←   SPIBF
1169 0562 16401755           DO : IDTBF ←   BUF3
1170 0563 16401755           DO : IDTBF ←   BUF3        / (400 NSEC IFCLK PULSE).
1171 0564 33762064         WCNT =        INC  WCNT ,L    / CHECK FOR END OF DATA.
1172 0565 44140000           CC = LOVER          ,RS
1173 0566 64000557           BR : VDATA                 / NOT YET.
1174 0567 42240000 VDTA1     CC = EOW            ,RS    / WAIT FOR ONE MORE BEFORE FREEZE.
1175 0570 64000567           BR : VDTA1
1176 0571 10007316           DI : BUF3  ←   SPIBF       / GET WORD, CLEAR EOW.
1177 0572 07610200           DO : ECICW ←   DSC  ,U     / TELL ECC TO GET READY.
1178 0573 07604452           DO : INTCW ←   RERWD,U     / SET TO GET ERROR WORDS.
1179 0574 16401755           DO : IDTBF ←   BUF3
1180 0575 16401755           DO : IDTBF ←   BUF3        / (400 NSEC IFCLK PULSE).
1181 0576 27722371         EWDCT = ERWCT PASS      ,L    / INITIALIZE ERROR WORD COUNTER.
1182 0577 42240000 VERWD     CC = EOW            ,RS    / WAIT FOR ERROR WORD.
1183 0600 64000577           BR : VERWD
1184 0601 10007304           DI : DUMMY ←   SPIBF       / THEN THROW IT AWAY.
1185 0602 33762064         EWDCT =       INC  EWDCT,L    / BUT DON'T FORGET TO COUNT IT.
1186 0603 44140000           CC = LOVER          ,RS    / LAST ERROR WORD?
1187 0604 64000577           BR : VERWD                 / NO, GO BACK FOR NEXT ONE.
1188 0605 42700000 VRFY3     CC = OVRUN                 / YES, A SLOWPOKE OUT THERE?
1189 0606 64000436           BR : OVERR                 / OVERRUN, TOO BAD ABOUT THAT.
1190 0607 07604413           DO : INTCW ←   RDWRT,U     / NO, TURN OFF DATA TRANSFER, BUT
1191                                                    /   KEEP ACRDY STATUS, CRCC.
1192 0610 42440000           CC = ACRDY          ,RS    / DRIVE GO NOT READY DURING OP?
1193 0611 64001762           BR : ACRER                 / AIN'T THAT A SHAME!
1194 0612 72000000           RT : 0                     / NO, READ OR VRFY DONE, RETURN.
```

```
1195                          READ FULL SECTOR COMMAND
1196                  /
1197                  /
1198                  /
1199                  /
1200                  /  RFS (READ FULL SECTOR) READS FROM THE DISC AND PASSES
1201                  /  TO THE CPU THE SYNC WORD, THE ADDRESS WORDS, THE DATA,
1202                  /  AND THE ERROR WORDS WITH NO ERROR CHECKING.  THE SECTOR
1203                  /  ADDRESS AND TRACK STATUS ARE NOT VERIFIED.
1204                  /
1205 0613 66001317 RFS       CA : UNIT                    / TRY TO CONNECT DRIVE.
1206 0614 66001714           CA : STRT2                   / SET PROPER DISC ADDRESS.
1207 0615 66000422           CA : DSTRT                   / GO SET UP READ DATA PATH.
1208 0616 07624017 RFULL     DO : TGBUS  ←   RDCOM,L      / GIVE READ COMMAND TO DRIVE.
1209 0617 66001642           CA : STRB1                   / TURN ON STROBE, LEAVE IT ON.
1210 0620 07606000           DO : IFVLD  ←   SETCL,U      / MAKE SURE TRANSFER OCCURS.
1211 0621 10007304           DI : DUMMY  ←   SPIBF        / MAKE SURE EOW IS CLEAR.
1212 0622 07600500           DO : TIMER  ←   START,U      / TURN ON TIME-OUT CIRCUIT.
1213 0623 07604472           DO : INTCW  ←   RCC2 ,U      / READ AND CYCLIC CHECK.
1214 0624 27722166         WCNT = FULSC PASS      ,L      / SET UP SECTOR WORD LENGTH.
1215 0625 42440000           CC = ACRDY             ,RS
1216 0626 64001762           BR : ACRER                   / DRIVE BUSY, ABORT.
1217 0627 42240000 RFDAT     CC = EOW               ,RS   / WAIT FOR WORD FROM DISC AND
1218 0630 64000627           BR : RFDAT                   /   PASS IT TO THE CPU.
1219 0631 10007316           DI : BUF3   ←   SPIBF
1220 0632 16401755           DO : IDTBF  ←   BUF3
1221 0633 16401755           DO : IDTBF  ←   BUF3         / (400 NSEC IFCLK PULSE).
1222 0634 33762064         WCNT =       INC  WCNT ,L
1223 0635 44140000           CC = LOVER             ,RS   / END OF SECTOR?
1224 0636 64000627           BR : RFDAT                   / NOT YET.
1225 0637 66000770           CA : EWRFS                   / YES, READ OFF, DO END SECTOR CHECKS.
1226 0640 66001503           CA : SECTR                   / ADDRESS NEXT SECTOR
1227 0641 64000616           BR : RFULL                   /   AND GO READ IT.
```

```
1228                          INITIALIZE COMMAND
1229                    /
1230                    /
1231                    /
1232                    /
1233                    /  INITIALIZE WRITES A SECTOR WITHOUT FIRST VERIFYING THE
1234                    /  ADDRESS FIELD AND ERROR BYTES OF THE PREVIOUS SECTOR.  IT IS
1235                    /  ALSO USED TO FLAG TRACKS SPARE, PROTECTED, AND-OR DEFECTIVE
1236                    /  (COMMAND WORD BITS 15-13 RESPECTIVELY).  THE OPERATION WILL
1237                    /  FAIL (STATUS-2 ERROR) IF THE DISC SURFACE IS HARDWARE
1238                    /  PROTECTED OR IF THE FORMAT SWITCH IS OFF.
1239                    /    WHEN A TRACK IS BEING INITIALIZED DEFECTIVE, THE
1240                    /  (LOGICAL) HEAD ADDRESS IN HSAD IS NOT NECESSARILY THE SAME AS
1241                    /  THE PHYSICAL HEAD BEING ADDRESSED.  SINCE THE HEAD ADDRESS IN
1242                    /  HSAD IS THE BASIS FOR SKEWING SECTORS WHEN ADDRESSING THE DRIVE,
1243                    /  WE MUST PUT THE PHYSICAL HEAD IN HSAD WHEN ADDRESSING THE DRIVE
1244                    /  AND THE LOGICAL HEAD IN WHEN WRITING HSAD ON TO THE SECTOR.
1245                    /  THIS ASSURES PROPER SKEWING, I.E., ON ANY TRACK, SECTORS ARE
1246                    /  SKEWED BY THE LOGICAL HEAD (THE ONE PASSED DURING THE LAST
1247                    /  SEEK OR ADDRESS RECORD COMMAND).
1248                    /
1249 0642 66001317 INIT     CA : UNIT                    / TRY TO CONNECT DRIVE.
1250 0643 66001714          CA : STRT2                   / SET PROPER DISC ADDRESS.
1251 0644 66000775          CA : CKWRT                   / MAKE SURE WRITING IS ALLOWED.
1252 0645 34010274        BUF2 =       CMP  CONWD       / SET UP S, P, AND D BITS
1253 0646 27412340       STATR = 340   AND  BUF2 ,U     /   IN STATR AND HSAD.
1254 0647 36704470        HSAD = HSAD  OR   STATR,U
1255 0650 37706210       PLHED =       PASS HSAD ,SW    / PUT LOGICAL HEAD IN LOWER BYTE
1256 0651 37706070       PLHED =       PASS HSAD ,U     /   AND UPPER BYTE (ASSUME SAME
1257                                                    /   LHEAD AND PHEAD TO START).
1258 0652 27402040       DUMMY = 40    AND  BUF2 ,U     / ISOLATE THE D BIT.
1259 0653 44440000          CC = TNZRO           ,RS    / IS IT ON?
1260 0654 64000662          BR : INIT1                  / NO, LHEAD = PHEAD, CONTINUE.
1261 0655 66000375          CA : RQSC3                  / YES, GO FETCH PHYSICAL HEAD.
1262 0656 37706064       PLHED =       PASS BUF1 ,U     / ! PHYSICAL HEAD ! LOGICAL HEAD !
1263 0657 37704074        HSAD =       PASS PLHED,U     / ADDR LOGICAL SCTR WITH PHYS HEAD.
1264 0660 66000320          CA : GDTYP                  / GET DRIVE TYPE FOR OUTSC.
1265 0661 66001725          CA : OUTSC
1266 0662 07622624 INIT1    DO : IFNBF  ←   IFOUT,L     / SET INTERFACE FOR DATA OUTPUT.
1267 0663 66000423          CA : DSTR1                  / GO SET UP WRITE DATA PATH.
1268 0664 44700000 INIT2    CC = FALSE                  / LOOP EXIT FAILS FROM HERE.
1269 0665 37706214 INIT3 PLHED =       PASS PLHED,SW    / ! LOGICAL HEAD ! PHYSICAL HEAD !
1270 0666 37704074        HSAD =       PASS PLHED,U     / HSAD _ LHEAD FOR WRITING OR EXIT.
1271 0667 64000504          BR : ROFS3                  / FAILS IF WE'RE STILL LOOPING.
1272 0670 66000705          CA : WRONE                  / INITIALIZE THE SECTOR.
1273 0671 37706214       PLHED =       PASS PLHED,SW    / ! PHYSICAL HEAD ! LOGICAL HEAD !
1274 0672 37704074        HSAD =       PASS PLHED,U
1275 0673 66001461          CA : INCHS                  / SET UP NEXT SECTOR ADDRESS.
1276 0674 42540000          CC = NWTRF           ,RS    / DID WE SWITCH TRACKS?
1277 0675 64000700          BR : INIT4                  / NO, CONTINUE.
1278 0676 37706210       PLHED =       PASS HSAD ,SW    / YES, ASSUME NORMAL INIT (NO
1279 0677 37706070       PLHED =       PASS HSAD ,U     /   D BIT), UPDATE PLHED.
1280 0700 42600000 INIT4    CC = EOD                    / ALL THE DATA TRANSFERRED?
1281 0701 64000665          BR : INIT3                  / YES, GO END THE OPERATION.
1282 0702 66001503          CA : SECTR                  / NO, GO SET FOR NEXT SECTOR.
1283 0703 66001024          CA : WSTA2                  / PLATTER PROTECT SWITCH SET?
1284 0704 64000664          BR : INIT2                  / NO, GO INITIALIZE ANOTHER SECTOR.
```

```
1285                          WRITE ONE SECTOR SUBROUTINE
1286                 /
1287                 /
1288                 /
1289                 /
1290                 /  SUBROUTINE WRONE WRITES ONE SECTOR ON THE DISC.  IT ASSUMES THAT
1291                 /  THE SECTOR HAS BEEN PREVIOUSLY ADDRESSED AND THAT THE DATA PATH
1292                 /  TO THE INTERFACE IS ALREADY SET.  IT ALSO ASSUMES THAT ALL INTER-
1293                 /  LOCKS OF A PARTICULAR WRITE OPERATION HAVE BEEN SATISFIED.  WRONE
1294                 /  IS NOT USED FOR THE "WRITE FULL SECTOR" COMMAND.
1295                 /    TIMING:  DO NOT LESSEN THE NUMBER OF INSTRUCTIONS BETWEEN
1296                 /  WRONE AND WRONE+3 OR BETWEEN WRONE+3 AND WSYNC-4.  THESE
1297                 /  INSTRUCTIONS ARE SPACED DELIBERATELY TO ALLOW TGBUS AND
1298                 /  COMMAND BUS PROP TIMES, BUT WITHOUT THE ARTIFICIAL DELAY
1299                 /  INTRODUCED BY A CALL TO STRB1.  THIS CUTS DOWN ON WASTED
1300                 /  TIME BETWEEN SECTORS OF A MULTIPLE-SECTOR TRANSFER.
1301                 /
1302 0705 07624016 WRONE    DO : TGBUS  ←   WRTCM,L     / PUT OUT WRITE COMMAND.
1303 0706 07606000          DO : IFVLD  ←   SETCL,U     / MAKE SURE TRANSFER OCCURS.
1304 0707 07600500          DO : TIMER  ←   START,U     / TURN ON TIME-OUT CIRCUIT.
1305 0710 07604475          DO : INTCW  ←   WCC1 ,U     / WRITE AND CYCLIC CHECK.
1306 0711 07610026          DO : ECICW  ←   WECC ,U     / WRITE WITH ECC.
1307 0712 27636366        BUF3 = 366    PASS      ,L     / SYNC FIELD WORD COUNTER
1308                                                    /   (9 REPS, WRITES 12 WORDS).
1309 0713 27616177        BUF3 = 177    PASS      ,U     / UPPER HALF OF SYNC WORD.
1310 0714 27722200        WCNT = WCT    PASS      ,L     / DATA FIELD LENGTH (1'S COMP).
1311 0715 07604471          DO : INTCW  ←   WCC2 ,U     / WCC1 WITH ACRDY CHECK ENABLED.
1312 0716 10007304          DI : DUMMY  ←   SPIBF       / CLEARS EOW FLAG FOR SYNC FIELD
1313                                                    /   (HDWE INITIALIZES SERDES TO 0).
1314 0717 42440000          CC = ACRDY            ,RS   / IS DRIVE O.K.?
1315 0720 64001762          BR : ACRER                  / NO, ESCAPE HERE.
1316 0721 42240000 WSYNC    CC = EOW              ,RS   / YES, READY TO TRANSFER WORD?
1317 0722 64000721          BR : WSYNC                  / NO, LOOP UNTIL WE ARE.
1318 0723 10007304          DI : DUMMY  ←   SPIBF       / YES, CLEAR EOW FOR NEXT
1319                                                    /   SYNC FIELD WORD.
1320 0724 22237401        BUF3 = 1      PLUS BUF3 ,L
1321 0725 44140000          CC = LOVER            ,RS
1322 0726 64000721          BR : WSYNC                  / IF NOT DONE, DO ANOTHER.
1323 0727 42240000 WSYN1    CC = EOW              ,RS   / WAIT FOR LAST ONE.
1324 0730 64000727          BR : WSYN1                  /   BEFORE PUTTING OUT SYNC WORD.
1325 0731 10031556          DI : BUF3   ←   ESYNC,L     / PUT OUT ECC MARK IN SYNC WORD.
1326 0732 16407655          DO : SPOBF  ←   BUF3        / PUT OUT SYNC WORD.
1327 0733 42240000 WCYLD    CC = EOW              ,RS   / WAIT FOR IT TO BE ACCEPTED.
1328 0734 64000733          BR : WCYLD
1329 0735 15207251          DO : SPOBF  ←   CYLAD,C     / THEN OUTPUT CYLINDER ADDRESS.
1330 0736 42240000 WHSAD    CC = EOW              ,RS   / WAIT FOR BUFFER TO ACCEPT IT.
1331 0737 64000736          BR : WHSAD
1332 0740 14007251          DO : SPOBF  ←   HSAD ,C     / OUTPUT HEAD-SECTOR ADDRESS.
1333 0741 42240000 WDATA    CC = EOW              ,RS
1334 0742 64000741          BR : WDATA
1335 0743 10001756          DI : BUF3   ←   IDTBF       / WRITE DATA FIELD IN THIS LOOP
1336 0744 10001756          DI : BUF3   ←   IDTBF       /   (400 NSEC IFCLK PULSE).
1337 0745 16407655          DO : SPOBF  ←   BUF3
1338 0746 33762064        WCNT =        INC   WCNT ,L     / COUNT THE DATA WORD.
1339 0747 44140000          CC = LOVER            ,RS   / END OF DATA FIELD?
1340 0750 64000741          BR : WDATA                  / NO, GO BACK FOR NEXT WORD.
1341 0751 42240000 WDTA2    CC = EOW              ,RS   / WAIT FOR LAST DATA WORD
1342 0752 64000751          BR : WDTA2                  /   TO BE ACCEPTED.
1343 0753 10007304          DI : DUMMY  ←   SPIBF       / RESET EOW.
```

```
1344                              WRITE ONE SECTOR SUBROUTINE, EWRFS, CKWRT
1345                    /
1346                    /
1347                    /
1348                    /
1349 0754 07610000            DO : ECICW  ←   ESC  ,U     / START COUNTING 2 EOWS TO FREEZE.
1350 0755 07604461            DO : INTCW  ←   WERWD,U     / SET TO WRITE ERROR WORDS.
1351 0756 27722371         EWDCT = ERWCT PASS       ,L     / INITIALIZE ERROR WORD COUNTER.
1352 0757 42240000 WTEWD      CC = EOW                ,RS   / WAIT FOR PREVIOUS WORD TO GO OUT.
1353 0760 64000757            BR : WTEWD
1354 0761 10007304            DI : DUMMY  ←   SPIBF       / RESET EOW.
1355 0762 33762064         EWDCT =       INC  EWDCT,L     / COUNT THE ERROR WORD.
1356 0763 44140000            CC = LOVER              ,RS   / LAST ONE?
1357 0764 64000757            BR : WTEWD                    / NO, GO BACK FOR NEXT ONE.
1358 0765 42240000 WRLWD      CC = EOW                ,RS   / YES, WAIT FOR IT TO BE WRITTEN.
1359 0766 64000765            BR : WRLWD
1360 0767 64000605            BR : VRFY3                    / DO END-OF-SECTOR CHECKS, RETURN.
1361                    /
1362                    /
1363                    /  SUBROUTINE EWRFS IS SOME COMMON CODE AT THE END OF THE WFS
1364                    /  AND RFS COMMAND PROCESSORS.  THE ENTRY AT EWRF1 IS USED BY THE
1365                    /  WRITE COMMAND AND READ COMMAND PROCESSORS.
1366                    /
1367 0770 66000605 EWRFS     CA : VRFY3                    / GO DO END-OF-SECTOR CHECKS.
1368 0771 66001461 EWRF1     CA : INCHS                    / ALL O.K., SET UP NEXT SECTOR.
1369 0772 42600000            CC = EOD                    / END OF DATA?
1370 0773 64000504            BR : ROFS3                  / YES, GO END THE OPERATION.
1371 0774 72000000            RT : 0                      / NO, RETURN AND CONTINUE.
1372                    /
1373                    /
1374                    /  SUBROUTINE CKWRT IS SOME COMMON CODE FROM THE INIT AND WFS
1375                    /  COMMAND PROCESSORS.
1376                    /
1377 0775 66001024 CKWRT     CA : WSTA2                    / MAKE SURE THE DRIVE CAN WRITE.
1378 0776 27423404         DUMMY = 4      AND  BUF3 ,L     / "FORMAT" SWITCH MUST BE ON TOO.
1379 0777 44440000            CC = TNZRO              ,RS
1380 1000 64001761            BR : STER                   / IT'S OFF, CAN'T DO IT.
1381 1001 72000000            RT : 0
```

```
1382                          WRITE COMMAND, SUBROUTINES WSTA1, WSTA2
1383                  /
1384                  /
1385                  /
1386                  /
1387                  /  WRITES A SECTOR OF INFORMATION ONTO THE DISC.
1388                  /  FIRST VERIFIES THE ADDRESS FIELD OF THE PRECEDING
1389                  /  SECTOR.  OPERATION FAILS IF SURFACE IS PROTECTED.
1390                  /
1391 1002 66001317 WRITE     CA : UNIT                    / TRY TO CONNECT DRIVE.
1392 1003 66001707           CA : STRT1                   / SET PROPER DISC ADDRESS.
1393 1004 66001014           CA : WSTA1                   / SEE IF WE CAN WRITE.
1394 1005 07622624           DO : IFNBF   ←   IFOUT,L     / ALL O.K., SET UP I'FC DATA PATH.
1395 1006 66000423           CA : DSTR1
1396 1007 66000705 WRTOK     CA : WRONE                   / GO WRITE ONE SECTOR.
1397 1010 66000771           CA : EWRF1                   / SET UP NEXT SECTOR ADDRESS,
1398                                                      /   CHECK FOR END OF TRANSFER.
1399 1011 66001503           CA : SECTR                   / NOT END, SET UP NEXT SECTOR.
1400 1012 66001014           CA : WSTA1                   / RECHECK P, D BITS, READ-ONLY.
1401 1013 64001007           BR : WRTOK                   / ALL O.K., GO WRITE NEXT SECTOR.
1402                  /
1403                  /
1404                  /    SUBROUTINE WSTA1 CHECKS THE P-BIT.  IF IT IS SET, THE FORMAT
1405                  /  SWITCH MUST ALSO BE SET TO ALLOW WRITING.  FOLLOWING WSTA1, WE
1406                  /  FALL INTO WSTA2 TO CHECK THE PLATTER PROTECT (7905) OR READ-
1407                  /  ONLY (7920) SWITCH.  NOTE:  DUPLICATING THE TWO RAM ACCESS
1408                  /  INSTRUCTIONS (ONCE IN WSTA1, AGAIN IN WSTA2) COSTS NO EXTRA
1409                  /  CODE AND SAVES 400 NSEC IN EXECUTION TIME OVER ALTERNATE METHODS.
1410                  /
1411 1014 27402500 WSTA1 DUMMY = 100   AND  STATR,U
1412 1015 44440000           CC = TNZRO          ,RS    / IS THE P-BIT SET?
1413 1016 64001024           BR : WSTA2                 / NO, GO CHECK READ ONLY SWITCH.
1414 1017 07620013           DO : RAMAD   ←   RDONL,L   / YES, CHECK IF
1415 1020 10020456           DI : BUF3    ←   RAM  ,L   /   FORMAT SWITCH IS ON.
1416 1021 27423404        DUMMY = 4       AND  BUF3 ,L
1417 1022 44440000           CC = TNZRO          ,RS
1418 1023 64001760           BR : WPER                  / NO, CAN'T WRITE ON P-TRACK.
1419                                                    / FALL INTO WSTA2 IF HERE.
1420                  /
1421                  /
1422                  /  SUBROUTINE WSTA2 CHECKS THE PLATTER PROTECT WORD IN RAM TO SEE
1423                  /  IF THE PLATTER PROTECT (READ-ONLY) SWITCH IS SET FOR THE
1424                  /  CURRENT SURFACE.
1425                  /
1426 1024 07620013 WSTA2     DO : RAMAD   ←   RDONL,L   / ACCESS THE
1427 1025 10020456           DI : BUF3    ←   RAM  ,L   /   READ-ONLY WORD.
1428 1026 27423410        DUMMY = 10      AND  BUF3 ,L
1429 1027 44400000           CC = TNZRO                 / BIT 3 ≠ 0 MEANS
1430 1030 64001761           BR : STER                  /   READ-ONLY SWITCH SET.
1431 1031 72000000           RT : 0                     / ALL O.K., RETURN.
```

```
1432                          WRITE FULL SECTOR, LOAD TIO COMMANDS
1433                  /
1434                  /
1435                  /
1436                  /  THE WFS (WRITE FULL SECTOR) COMMAND WRITES THE ENTIRE SECTOR
1437                  /  (INCLUDING PRE- AND POSTAMBLE, BUT NOT SYNC FIELD) FROM DATA
1438                  /  SUPPLIED BY THE CPU.  FORMAT SWITCH MUST BE SET TO OVERRIDE.
1439                  /    TIMING:  SAME CONSIDERATIONS HERE AS FOR WRONE.
1440                  /
1441 1032 66001317 WFS       CA : UNIT                    / TRY TO CONNECT DRIVE.
1442 1033 66001714           CA : STRT2                   / SET PROPER DISC ADDRESS.
1443 1034 66000775           CA : CKWRT                   / MAKE SURE WRITING IS ALLOWED.
1444 1035 07622624           DO : IFNBF  ←   IFOUT,L      / ALL O.K., SET UP DATA PATH.
1445 1036 66000423           CA : DSTR1
1446 1037 07624016 WFULL     DO : TGBUS  ←   WRTCM,L      / SEND WRITE COMMAND TO DRIVE.
1447 1040 07606000           DO : IFVLD  ←   SETCL,U      / MAKE SURE TRANSFER OCCURS.
1448 1041 07604475           DO : INTCW  ←   WCC1 ,U      / WRITE (WITH "CYCLIC CHECK").
1449 1042 10007304           DI : DUMMY  ←   SPIBF        / CLEARS EOW FLAG FOR SYNC FIELD
1450                                                      /   (HDWE INITIALIZES SERDES TO 0).
1451 1043 27636366         BUF3 = 366    PASS       ,L     / SYNC FIELD WORD COUNTER.
1452 1044 27722166         WCNT = FULSC  PASS       ,L     / DATA FIELD WORD COUNTER.
1453 1045 07600500           DO : TIMER  ←   START,U      / TURN ON TIME-OUT CIRCUIT.
1454 1046 07604471           DO : INTCW  ←   WCC2 ,U      / WCC1 WITH ACRDY CHECK ENABLED.
1455 1047 42440000           CC = ACRDY            ,RS    / WAIT FOR FIRST WORD OUT.
1456 1050 64001762           BR : ACRER
1457 1051 42240000 WFSYN     CC = EOW              ,RS
1458 1052 64001051           BR : WFSYN
1459 1053 10007304           DI : DUMMY  ←   SPIBF        / CLEAR EOW.
1460 1054 22237401         BUF3 = 1      PLUS BUF3 ,L     / COUNT THE SYNC FIELD WORD
1461 1055 44140000           CC = LOVER            ,RS
1462 1056 64001051           BR : WFSYN                   /   AND GO BACK FOR MORE,
1463 1057 42240000 WFDAT     CC = EOW              ,RS    /   OR ELSE START THE DATA FIELD.
1464 1060 64001057           BR : WFDAT
1465 1061 10001756           DI : BUF3   ←   IDTBF
1466 1062 10001756           DI : BUF3   ←   IDTBF        / (400 NSEC IFCLK PULSE).
1467 1063 16407655           DO : SPDBF  ←   BUF3         / SEND OUT THE DATA WORD.
1468 1064 33762064         WCNT =        INC  WCNT ,L
1469 1065 44140000           CC = LOVER            ,RS    / DATA TRANSFER DONE?
1470 1066 64001057           BR : WFDAT                   / NOT YET.
1471 1067 42240000 WFWAT     CC = EOW              ,RS    / YES, WAIT FOR LAST WORD TO GO OUT.
1472 1070 64001067           BR : WFWAT
1473 1071 10007304           DI : DUMMY  ←   SPIBF        / CLEAR EOW
1474 1072 42240000 WFWA2     CC = EOW              ,RS    /   THEN WAIT FOR SERDES TO CLEAR.
1475 1073 64001072           BR : WFWA2
1476 1074 66000770           CA : EWRFS                   / WRITE OFF, DO END SECTOR CHECKS.
1477 1075 66001503           CA : SECTR                   / MORE TO DO, ADDRESS NEXT SECTOR.
1478 1076 66001024           CA : WSTA2                   / CHECK NEW READ-ONLY STATUS.
1479 1077 64001037           BR : WFULL                   / O.K., GO WRITE THE SECTOR.
1480                  /
1481                  /
1482                  /  LTIO IS A DIAGNOSTIC ROUTINE THAT LOADS THE TIO REGISTER WITH
1483                  /  THE CONTENTS OF A WORD PASSED BY THE 3000 CPU.
1484                  /
1485 1100 66001610 LTIO      CA : SETUP                   / CLEAR STATUS, SET CONTROLLER BUSY.
1486 1101 66001654           CA : GET                     / GET THE PATTERN WORD.
1487 1102 66001675           CA : WTIO1                   / SEND IT TO THE TIO REGISTER.
1488 1103 66001623           CA : SRQ                     / ASK FOR THE NEXT INSTRUCTION.
1489 1104 66001704           CA : BSYCL                   / CLEAR THE CONTROLLER BUSY BIT.
1490 1105 64000141           BR : CMDWT                   / GO WAIT FOR NEXT COMMAND.
```

```
                        REQUEST SYNDROME COMMAND
                /
                /   THIS SECTION OF CODING IMPLEMENTS THE REQUEST SYNDROME (RQSYN)
                /  COMMAND.  THE RQSYN COMMAND RETURNS 7 WORDS OF INFORMATION:
                /
                /    1)  AN UPDATED CONTROLLER STATUS WORD (7400B IF THE ERROR WAS
                /  CORRECTED, 4000B IF IT WAS UNCORRECTABLE.  IN THE LATTER CASE,
                /  WORDS 4-7 CONTAIN GARBAGE AND SHOULD BE IGNORED).
                /    2-3)  THE DISC ADDRESS (CYLAD AND HSAD-MINUS-SPD-BITS) WHERE
                /  THE DATA ERROR OCCURRED (USEFUL IN MULTIPLE-SECTOR TRANSFERS).
                /    4)  THE DISPLACEMENT WITHIN THE SECTOR WHERE THE ERROR PATTERN
                /  (WORDS 5-7) SHOULD BE APPLIED.
                /    5-7)  THE ERROR PATTERN, USED TO CORRECT DATA IN THE USER'S
                /  BUFFER.
                /
                /    THE DISPLACEMENT MAY VARY FROM -5 (TWO'S COMPLEMENT) TO +135.
                /  IT REPRESENTS THE OFFSET THAT SHOULD BE ALGEBRAICALLY ADDED TO
                /  THE BASE (FIRST WORD) ADDRESS OF THE DATA BUFFER CORRESPONDING
                /  TO THE SECTOR IN ERROR (USE WORDS 2 AND 3).  THIS GETS THE
                /  ADDRESS OF THE FIRST WORD TO WHICH THE ERROR PATTERN SHOULD BE
                /  APPLIED.  NOTE THAT IF THE DISPLACEMENT < 0 OR > 125, SOME OR
                /  ALL OF THE SECTOR ERRORS ARE OUTSIDE THE DATA AREA.  TO AVOID
                /  MIS-CORRECTION IN THIS CASE, FOLLOW THE ALGORITHM SHOWN IN THE
                /  13037 PRODUCT SPECIFICATIONS (HOPEFULLY ALSO THE OPERATING MANUAL).
                /
                /    THE ERROR PATTERN WORDS ARE MASKS TO BE EXCLUSIVE-OR'ED WITH
                /  DATA ALREADY IN THE USER'S BUFFER.  BITS IN ERROR WILL BE 1 IN
                /  THE MASK, CAUSING THE CORRESPONDING BITS IN THE DATA BUFFER TO
                /  BE COMPLEMENTED.  THE ERROR PATTERN IS ALWAYS ADJUSTED TO A WORD
                /  BOUNDARY, SO NO BIT SHIFTING WITHIN WORDS IS REQUIRED.
                /
                /    THE RQSYN COMMAND SHOULD ONLY BE INVOKED IF BITS 12-8 OF THE
                /  CONTROLLER STATUS (STATUS-1) WORD FROM THE PREVIOUS COMMAND
                /  ARE 0 111 1 (STATUS = 74UU (OCTAL)).  THIS INDICATES:
                /
                /    1)  THE ECC FEATURE IS INSTALLED IN THE CONTROLLER,
                /    2)  THE SECTOR IN ERROR WAS WRITTEN USING A CONTROLLER IN
                /  WHICH ECC WAS INSTALLED (GUARANTEEING A VALID ECC FIELD), AND
                /    3)  THE ERROR IS POSSIBLY CORRECTABLE.
                /
                /  IF ANY OF THE ABOVE CONDITIONS ARE FALSE, AND A DATA ERROR
                /  OCCURS, CONTROLLER STATUS WILL BE 0 100 0 (40UU (OCTAL)).  A RQSYN
                /  COMMAND ISSUED HERE WILL RETURN STATUS = 4000B AND GARBAGE IN
                /  WORDS 4-7.  (TO OBTAIN THE DISC ADDRESS OF THE SECTOR IN ERROR
                /  WHEN STATUS = 40XXB, USE THE REQUEST ADDRESS (REQAD) COMMAND.
                /    IF CONDITION 1 OR 2 IS FALSE (NO ECC), A 16-BIT CRC (ALWAYS
                /  PRESENT) IS USED.  IT CANNOT CORRECT ERRORS, BUT WILL DETECT:
                /
                /    1)  ALL SINGLE-16-BIT-OR-LESS BURST ERRORS.
                /    2)  99.997% OF ALL SINGLE-17-BIT BURST ERRORS.
                /    3)  99.9985% OF ALL BURST ERRORS > 17 BITS LONG.
                /
                /    IF ECC IS INSTALLED AND IS VALID FOR THE SECTOR IN ERROR, THE
                /  HARDWARE AND THIS CODING WILL:
                /
                /    1)  CORRECT ALL SINGLE-BURST ERRORS <= 32 BITS LONG.
                /    2)  DETECT ALL SINGLE-BURST ERRORS > 32 BITS BUT <= 48 BITS
                /  LONG WITHOUT MIS-CORRECTING THEM.
                /    3)  DETECT 99.999% OF ALL SINGLE-BURST ERRORS > 48 BITS LONG.
```

```
1550                              REQUEST SYNDROME COMMAND
1551                    /
1552                    /
1553                    /
1554                    /
1555 1106 66001611 RQSYN     CA : BSYST                          / SET CONTROLLER BUSY.
1556 1107 27700000          CACC = 0      PASS        ,U       / START WITH C-ACCUMULATOR SET TO 0.
1557 1110 27720000          CACC = 0      PASS        ,L
1558 1111 27412740          STATR = 340   AND   STATR,U       / PRESERVE S, P, AND D BITS.
1559 1112 26612417          STATR = 17    OR    STATR,U       / ASSUME IT'S CORRECTABLE AT FIRST.
1560 1113 27610010          CAPM = CAPMU PASS         ,U      / SET UP CODE LENGTH MODULUS (2279)
1561 1114 27630347          CAPM = CAPML PASS         ,L      /   FOR BOTH BYTES.
1562 1115 07610211            DO : ECICW  ←   CLKHH,U         / CLOCK HIGH ONCE TO AVOID
1563                                                          /   RACE CONDITION IN ECICW.
1564 1116 07610230            DO : ECICW  ←   CLKHI,U         / CLOCK MUST BE HIGH BEFORE INHIBIT.
1565 1117 27722312          CLIM = 312    PASS        ,L      / C-LOOP COUNTER = -53 (DEC) -1
1566                                                          /   (BECAUSE WE TEST LIMIT BEFORE
1567                                                          /   WE TEST FOR BURST).
1568 1120 43240000 MOREC      CC = BURST              ,RS     / HAS BURST OCCURRED ALREADY?  WITHOUT
1569 1121 64001152            BR : PLOOP                      /   ANY C-SHIFTS?  GO MATCH.
1570 1122 07610312 CLOOP      DO : ECICW  ←   FRPSH,U         / FREEZE P-REGISTER, SET CLOCK HIGH
1571                                                          /   (OFF), REMOVE REGISTER INHIBIT.
1572 1123 27616006          CCONS = CCONU PASS        ,U      / SET UP CHINESE REMAINDER THEOREM
1573 1124 27636067          CCONS = CCONL PASS        ,L      /   C-CONSTANT (1591) IN BOTH BYTES.
1574 1125 07610300            DO : ECICW  ←   CCLKL,U         / FREEZE P, CLOCK LOW-HIGH TO
1575 1126 07610310            DO : ECICW  ←   CCLKH,U         /   SHIFT C ONCE WITH FEEDBACK.
1576 1127 32301660          CACC = CACC   PLUS CCONS          / ADD THE CONSTANT TO C-ACCUMULATOR.
1577 1130 66001200            CA : ACMOD                      / REDUCE CACC MOD CAPM.
1578 1131 33762064          CLIM =        INC   CLIM ,L       / COUNT THE NON-BURST.
1579 1132 44140000            CC = LOVER              ,RS     / CHECK LIMIT - DONE?
1580 1133 64001120            BR : MOREC                      / NO, GO SEE IF LAST SHIFT GAVE
1581                                                          /   US A BURST.
1582                    /
1583                                                          / DONE, CAN'T GET A CORRECTABLE
1584                                                          /   PATTERN, GONNA HAVE TO QUIT.
1585 1134 27412750          STATR = 350   AND   STATR,U       / UNCORRECTABLE DATA ERROR.
1586 1135 35302660          BUF1 =        CMP   STATR         / SEND UDTER STATUS AND DRIVE NO.
1587 1136 66001304            CA : PUT                        / SENDS C(BUF1) TO INTERFACE.
1588 1137 66001271            CA : RQAD1                      / SEND OUT CYLAD, HSAD.
1589 1140 34102260 CDER1    BUF1 =        CMP   ACC           / SEND OUT WORD DISPLACEMENT.
1590 1141 66001304            CA : PUT
1591 1142 36503660          BUF1 =        PASS ZREG           / SEND OUT FIRST WORD OF PATTERN.
1592 1143 66001304            CA : PUT
1593 1144 36502260          BUF1 =        PASS YREG           / SEND OUT SECOND WORD OF PATTERN.
1594 1145 66001304            CA : PUT
1595 1146 37702274          BUF1 =        PASS XREG           / SEND OUT LAST WORD OF PATTERN.
1596 1147 66001304            CA : PUT
1597 1150 66001461            CA : INCHS                      / ADDRESS THE NEXT SECTOR.
1598 1151 64000371            BR : RQSC2                      / NOW TERMINATE.
1599
1600                    /
1601                    /
1602                    /
1603                    /
1604 1152 27706000 PLOOP    PACC = 0      PASS        ,U      / START P-LOOP WITH P-ACCUMULATOR
1605 1153 27726000          PACC = 0      PASS        ,L      /   SET TO 0.
1606 1154 07610250            DO : ECICW  ←   PCLKH,U         / FREEZE C-REGISTER, SET CLOCK HIGH
1607                                                          /   (OFF) INITIALLY.
1608 1155 27616002          PCONS = PCONU PASS        ,U      / SET UP CHINESE REMAINDER THEOREM
1609 1156 27636261          PCONS = PCONL PASS        ,L      /   P-CONSTANT (689) IN BOTH BYTES.
1610 1157 37702204          PLIM =        PASS CLIM   ,SW     / PLIM AND CLIM ARE IN SAME WORD.
1611 1160 27722325          PLIM = 325    PASS        ,L      / P-LOOP COUNTER = -43 (DEC) BECAUSE
1612                                                          /   WE MUST LEAVE P-REGISTER LIKE WE
1613                                                          /   FOUND IT IF NO MATCH.
1614 1161 64001172            BR : COMP2                      / GO CHECK FOR IMMEDIATE MATCH.
1615 1162 07610240 MOREP      DO : ECICW  ←   PCLKL,U         / FREEZE C, CLOCK LOW-HIGH TO
1616 1163 07610250            DO : ECICW  ←   PCLKH,U         /   SHIFT P ONCE WITH FEEDBACK.
1617 1164 32307674          PACC = PACC   PLUS PCONS          / ADD THE CONSTANT TO P-ACCUMULATOR.
1618 1165 31546274          PACC = PACC   SUB  CAPM           / REDUCE MOD CAPM.
1619 1166 44540000            CC = TMSB               ,RS     / SUBTRACT TOO MUCH?
1620 1167 64001171            BR : COMP1                      / NO, IT'S OK.
1621 1170 32306274          PACC = PACC   PLUS CAPM           / YES, ADD IT BACK.
1622 1171 33762064 COMP1    PLIM =        INC   PLIM ,L       / COUNT THE NON-MATCH.
1623 1172 43240000 COMP2      CC = MATCH              ,RS     / GET MATCH DUE TO LAST SHIFT?
1624 1173 64001205            BR : GTPAT                      / YES, GO GET PATTERN.
1625 1174 44140000            CC = LOVER              ,RS     / NO MATCH, CHECK LIMIT - DONE?
1626 1175 64001162            BR : MOREP                      / NO, GO BACK, TRY TO MATCH AGAIN.
1627 1176 37702204          CLIM =        PASS PLIM   ,SW     / YES, RESTORE CLIM.
1628 1177 64001122            BR : CLOOP                      / RESET PACC TO 0, TRY NEW
1629                                                          /   PATTERN.  RESET TO 0 IS DONE
1630                                                          /   ON ENTRY INTO P-LOOP.
```

```
1631                    /           REQUEST SYNDROME COMMAND
1632                    /
1633                    /    SUBROUTINE ACMOD GROUPS SOME COMMON CODE FROM THE CLOOP AND GTPAT
1634                    /  ROUTINES, THEREBY SAVING ONE WORD.  ****NOTE****:  THIS PLOY ONLY
1635                    /  WORKS BECAUSE THE SYMBOLIC REGISTERS ACC AND CACC ARE THE SAME
1636                    /  PHYSICAL REGISTER.  IT CANNOT BE USED FOR THE SIMILAR CODING IN PLOOP.
1637                    /  IF ACC AND CACC ARE EVER A DIFFERENT PHYSICAL REGISTER, IT WILL BE
1638                    /  NECESSARY TO RESTORE THIS CODING TO IN-LINE CODE IN CLOOP AND GTPAT.
1639                    /
1640 1200 31540260 ACMOD    ACC = ACC    SUB   CAPM          / REDUCE ACC (CACC) MOD CAPM.
1641 1201 44540000           CC = TMSB                ,RS    / SUBTRACT TOO MUCH?
1642 1202 72000000           RT : 0                          / NO, RETURN.
1643 1203 32300260          ACC = ACC    PLUS CAPM          / YES, ADD IT BACK.
1644 1204 72000000           RT : 0                          / THEN RETURN.
1645
1646                    /
1647                    /
1648                    /
1649                    /
1650 1205 35302660 GTPAT   BUF1 =         CMP   STATR        / DATA ERROR CORRECTABLE, SEND
1651 1206 66001304           CA : PUT                        /   STATUS AND DISC ADDRESS HERE TO
1652 1207 66001271           CA : RQAD1                      /   PRESERVE STATR LATER.
1653 1210 37616274        PCONS =         PASS PACC          / TEMPORARILY GET A B-REGISTER.
1654 1211 32301660          ACC = CACC    PLUS PCONS         / ADD THE TWO DISPLACEMENTS.
1655 1212 66001200           CA : ACMOD                      / REDUCE ACC MOD CAPM.
1656 1213 27630231         BIAS = 231     PASS        ,L     / SET UP SHORTENED CODE BIAS
1657 1214 27610000         BIAS = 0       PASS        ,U     /   PLUS 48 PRE-AMBLE BITS.
1658 1215 31450260         BIAS = ACC     SUB   BIAS         / SUBTRACT BIAS FROM ORIGINAL BIT
1659                                                         /   DISPLACEMENT.
1660 1216 44540000           CC = TMSB                ,RS    / REMEMBER IF IT WAS NEGATIVE.
1661 1217 27436017         BUF3 = 17      AND   BIAS ,L      / MASK OUT LOWER ORDER 4 BITS
1662                                                         /   WHICH IS 15-# SHIFTS REQUIRED TO
1663                                                         /   PLACE ERROR PATTERN ON WORD BOUNDARY
1664 1220 21563420        COUNT = 20     SUB   BUF3 ,L      / START WITH 16-# SHIFTS REQUIRED,
1665                                                         /   BECAUSE WE DECREMENT FIRST
1666                                                         /   (AT WLOOP).
1667 1221 36500320          ACC =        PASS BIAS ,SR      / THEN DIVIDE MODIFIED BIT DISPLACEMENT
1668 1222 37700320          ACC =        PASS ACC  ,SR      /   BY 16 (SHIFT RIGHT 4 TIMES) TO
1669 1223 37700320          ACC =        PASS ACC  ,SR      /   GET THE WORD DISPLACEMENT.
1670 1224 37700320          ACC =        PASS ACC  ,SR
1671 1225 64001230           BR : COMP3                     / WAS DISPLACEMENT NEGATIVE?
1672 1226 27610360         BIAS = 360     PASS        ,U     / YES, EXTEND THE SIGN BIT
1673 1227 36700060          ACC = BIAS    OR    ACC   ,U     /   BY PUTTING IN 1'S.
1674 1230 10013302 COMP3     DI : YREG     ←     WORD1        / LOAD THE 16 MSB'S OF PATTERN.
1675 1231 10113754           DI : XREG     ←     WORD2        / LOAD THE 16 LSB'S OF PATTERN.
1676                                                         /   (HARDWARE SWAPS THESE WORDS END-
1677                                                         /   FOR-END DURING LOADING).
1678 1232 27616377         ZREG = 377     PASS        ,U     / CLEAR ZREG INITIALLY.  WORK IN +TRUE
1679 1233 27636377         ZREG = 377     PASS        ,L     /   TO SAVE A WORD OF CODE.
1680 1234 30122064 WLOOP  COUNT =         DEC   COUNT,L      / COUNT DOWN # SHIFTS REQUIRED.
1681 1235 44440000           CC = TNZRO               ,RS    / DONE SHIFTING?
1682 1236 64001140           BR : CDER1                      / YES, GO SEND DISPLACEMENT & DATA.
1683 1237 36417740         ZREG =         PASS ZREG ,SL      / NO, SHIFT ZREG ONCE WITH 1-FILL.
1684 1240 36410340         YREG =         PASS YREG ,SL      / SHIFT YREG ONCE WITH 1-FILL.
1685 1241 44540000           CC = TMSB                ,RS    / LOOK AT PREVIOUS YREG MSB.
1686 1242 64001244           BR : SHFTX                      / 1 (+TRUE), DON'T WORRY ABOUT ZREG.
1687 1243 26237401         ZREG = 1       XOR   ZREG ,L      / 0, SO SHIFT A 0 INTO ZREG LSB.
1688 1244 37706354 SHFTX   XREG =         PASS XREG ,SL      / SHIFT XREG LEFT ONCE WITH 1-FILL.
1689 1245 44540000           CC = TMSB                ,RS    / LOOK AT PREVIOUS XREG MSB.
1690 1246 64001250           BR : ADJX                       / 1 (+TRUE), DON'T WORRY ABOUT YREG.
1691 1247 26230001         YREG = 1       XOR   YREG ,L      / 0, SO SHIFT A 0 INTO YREG LSB.
1692 1250 33766074 ADJX    XREG =         INC   XREG ,L      / XREG MUST END UP WITH 0-FILL.
1693 1251 64001234           BR : WLOOP                      / GO SEE IF WE'RE DONE SHIFTING.
```

```
1694                            SET FILE MASK, CLEAR COMMANDS
1695                  /
1696                  /
1697                  /
1698                  /
1699                  /  THE SET FILE MASK (SFMSK) COMMAND ROUTES BITS 7-0 OF THE COMMAND
1700                  /  WORD TO APPROPRIATE DESTINATIONS.  BITS 3-0 OF THIS WORD ARE
1701                  /  SAVED IN THE FILE MASK REGISTER AND ARE INTERPRETED AS FOLLOWS:
1702                  /
1703                  /    BIT 0:  ALLOW AUTOMATIC INCREMENTAL-DECREMENTAL SEEK.
1704                  /    BIT 1:  SET ==> CYLINDER MODE, CLEAR ==> SURFACE MODE.
1705                  /    BIT 2:  ALLOW AUTOMATIC TRACK SPARING.
1706                  /    BIT 3:  IF BIT 0 SET, SET ==> DECREMENT, CLEAR ==> INCREMENT.
1707                  /            IF BIT 0 CLEAR, BIT 3 IS IGNORED.
1708                  /
1709                  /  BITS 7-4 ARE SENT TO THE 3000 INTERFACE AS A RETRY COUNTER FOR
1710                  /  DATA TRANSFER OPERATIONS IN WHICH ONE OR MORE ERRORS OCCUR.
1711                  /
1712 1252 66001610 SFMSK     CA : SETUP                    / SET UP STATUSES (STATI???).
1713 1253 17625555           DO : FLMSK  ←   COMWD,L      / PUT FILE MASK OUT TO REGISTER.
1714 1254 07622453           DO : IFNBF  ←   SRTRY,L      / SET UP RETRY COUNT.
1715 1255 17601755           DO : IDTBF  ←   COMWD
1716 1256 66001630           CA : ICLCK
1717 1257 66001702 INTRP     CA : TRNOF                    / CLOCKS OFF, WRT TIO, CLEAR BUSY.
1718 1260 66001315           CA : DFLST                    / SET THE 2100 DATA FLAG.
1719 1261 64000141           BR : CMDWT                    / GO WAIT FOR NEXT COMMAND.
1720                  /
1721                  /
1722                  /    THE CLEAR COMMAND PROCESSOR CLEARS ALL DISC DRIVES.  IT MUST
1723                  /  BE USED WITH EXTREME CAUTION (IF AT ALL) IN MULTI-CPU SYSTEMS.
1724                  /
1725 1262 66001610 CLR       CA : SETUP                    / CLR STATUS, SET BUSY, FLUSH CMND.
1726 1263 07624012           DO : TGBUS  ←   CLRCM,L      / PUT OUT CLEAR COMMAND AND
1727 1264 66001641           CA : STROB                    /   VALIDATE IT.
1728 1265 64001257           BR : INTRP                    / GO CLEAN UP.
1729                  /
1730                  /
1731                  /  BECAUSE OF CONTROL STORE SPACE LIMITATIONS, THE CLEAR UNIT BUSY
1732                  /  COMMAND (CLRUB) HAS BEEN DELETED.  ITS FUNCTION, CLEARING THE
1733                  /  HOLD BIT OF DRIVES ATTACHED TO INTERFACES WHOSE CPU HAS DIED,
1734                  /  IS NOW AUTOMATICALLY PROGRAMMED AS PART OF THE TIME-OUT
1735                  /  PROCESSOR.  SHOULD IT BE (POSSIBLE-REQUIRED-DESIRED) (CHOOSE
1736                  /  ONE OR MORE) TO USE IT AGAIN IN THE FUTURE, LISTINGS DATED
1737                  /  6-19-75 AND EARLIER CONTAIN THE MICROCODE USED IN THE COMMAND
1738                  /  PROCESSOR.  THERE WERE NO KNOWN BUGS IN THE COMMAND AT THAT TIME.
1739                  /
1740                  /
1741                  /  TO IMPLEMENT LONGER DISC AND INTERFACE CLOCK STROBES, THE
1742                  /  REQUEST UNIT ALLOCATION AND TIME OUT INTERRUPTS COMMAND
1743                  /  HAS BEEN DELETED.  SHOULD IT BECOME POSSIBLE TO USE IT
1744                  /  AGAIN IN THE FUTURE, LISTINGS DATED 4-11-75 AND EARLIER
1745                  /  CONTAIN THE MICROCODE USED IN THE COMMAND PROCESSOR AND
1746                  /  ASSOCIATED SUBROUTINES.  THERE WERE NO KNOWN BUGS IN THE
1747                  /  COMMAND AT THAT TIME.
```

```
1748                          REQUEST DISC ADDRESS COMMAND, SUBROUTINE PUT
1749                  /
1750                  /
1751                  /    THE REQAD (REQUEST DISC ADDRESS) COMMAND IS USED WHENEVER THE
1752                  /  USER WANTS TO KNOW THE DISC ADDRESS CURRENTLY BEING POINTED TO
1753                  /  BY THE CONTROLLER.  IN GENERAL, THIS WILL BE THE CONTENTS OF THE
1754                  /  CONTROLLER REGISTERS CYLAD AND HSAD.  THE CONTROLLER'S DISC
1755                  /  ADDRESS IS ALWAYS SUCH THAT A SUBSEQUENT DATA TRANSFER (OR VERIFY)
1756                  /  OPERATION MAY OCCUR WITHOUT RE-ADDRESSING THE CONTROLLER WITH A
1757                  /  SEEK OR ADDRESS RECORD COMMAND.  THIS MEANS THAT IN CASE OF AN
1758                  /  ERROR DURING THE DATA TRANSFER, CYLAD AND HSAD POINT TO THE SECTOR
1759                  /  IN WHICH THE ERROR OCCURRED.  THIS INFORMATION IS USEFUL TO THE
1760                  /  SYSTEM DURING MULTIPLE-SECTOR TRANSFERS, AND ALSO ALLOWS RE-READS.
1761                  /  IF NO ERROR OCCURRED, CYLAD AND HSAD USUALLY POINT TO THE NEXT
1762                  /  SECTOR TO BE TRANSFERRED.
1763                  /    AT THE END OF A TRACK OR CYLINDER, THE CONTENTS OF CYLAD AND HSAD
1764                  /  ARE ONLY PARTIALLY UPDATED BY THE INCHS ROUTINE.  THE REST OF THE
1765                  /  UPDATING OCCURS WHEN THE NEXT TRANSFER BEGINS.  THIS IS TO AVOID
1766                  /  A POSSIBLE END OF CYLINDER ERROR (SEE THE DESCRIPTION FOR
1767                  /  SUBROUTINE SECTR).  A REQAD ISSUED HERE WOULD RETURN THE WRONG
1768                  /  LOGICAL ADDRESS TO THE USER, SO THE END OF CYLINDER AND NEW TRACK
1769                  /  FLAGS (EOCF AND NWTRF) ARE USED TO MODIFY THESE ADDRESSES
1770                  /  APPROPRIATELY BEFORE SENDING THEM OFF.
1771                  /    SUBROUTINE RQAD1 IS SEPARATED OUT SO THAT THE RQSYN PROCESSOR CAN
1772                  /  USE IT TOO.  EOCF AND NWTRF ARE ALWAYS FALSE DURING RQSYN.
1773                  /
1774 1266 66001611 REQAD    CA : BSYST                    / SET CONTROLLER BUSY, SAVE STATUS.
1775 1267 66001271          CA : RQAD1                    / SO "RQSYN" CAN USE ALSO.
1776 1270 64000371          BR : RQSC2                    / GO END THE OPERATION.
1777                  /
1778 1271 35303260 RQAD1  BUF1 =       CMP  CYLAD         / SEND OUT CYLINDER ADDRESS.
1779 1272 43440000          CC = EOCF           ,RS       / INCREMENT IT BY 1 IF AT EOC.
1780 1273 64001275          BR : RQAD2                    / DON'T BOTHER IT OTHERWISE.
1781 1274 30102264        BUF1 =       DEC  BUF1          / EOC, ADD 1 TO +TRUE CYLAD.
1782 1275 66001304 RQAD2    CA : PUT                      / THERE IT GOES...
1783 1276 37616270       HSADB =       PASS HSAD          / PUT HSAD IN -B- REGISTER
1784 1277 27417437       HSADB = 37    AND  HSADB,U       /   SO WE CAN MASK OFF SPD BITS.
1785 1300 42540000          CC = NWTRF           ,RS       / RESET SECTOR TO 0 IF AT END OF
1786 1301 64001303          BR : RQAD3                    /   TRACK, ELSE LEAVE IT ALONE.
1787 1302 27636000       HSADB = 0     PASS       ,L      / END OF TRACK, SET SECTOR = 0.
1788 1303 35303660 RQAD3  BUF1 =       CMP  HSADB         / GIVE HSAD TO BUF1
1789                                                      /   AND FALL INTO PUT.
1790                  /
1791                  /
1792                  /  SUBROUTINE PUT WRITES ONE WORD TO THE INTERFACE.  THE WORD
1793                  /  IS IN BUF1 ON ENTRY.  DFLST SETS THE 2100 INTERFACE FLAG.
1794                  /  PUT IS A SECOND-LEVEL SUBROUTINE.
1795                  /
1796 1304 07600500 PUT      DO : TIMER  ←    START,U      / TURN ON TIME-OUT CIRCUIT.
1797 1305 42100000 PUT1     CC = DTRDY                    / INTERFACE BUFFER EMPTY?
1798 1306 64001305          BR : PUT1                     / NO, WAIT UNTIL IT IS.
1799 1307 07622443          DO : IFNBF  ←    IFIN ,L      / YES, SET UP TO PASS THE WORD.
1800 1310 66001631          CA : ICLK1                    / CAN'T CALL DSTRT -- TOO DEEP.
1801 1311 66000424          CA : CKXFR                    / CHECK FOR 3000 CHANNEL ERROR.
1802 1312 17600745          DO : IDTBF  ←    BUF1         / O.K., SEND THE DATA WORD
1803 1313 17600745          DO : IDTBF  ←    BUF1         /   (400 NSEC IFCLK PULSE).
1804 1314 66001621          CA : STPC1                    / TURN OFF TIME-OUT AND IFVLD.
1805 1315 07622676 DFLST    DO : IFNBF  ←    STDFL,L      / SEND OUT "SET FLAG" COMMAND.
1806 1316 64001630          BR : ICLCK
```

```
1807                        DRIVE NUMBER VERIFICATION AND SELECT ROUTINE
1808                /
1809                /
1810                /
1811                /
1812                /    THE FOLLOWING DIAGRAM IS THE LAYOUT OF THE 64-BIT (16X4) RAM WHICH
1813                /  MAPS THE DISC UNIT NUMBER (SAME AS THE RAM WORD NUMBER EXCEPT FOR
1814                /  NUMBERS 11-15) INTO THE NUMBER OF THE INTERFACE WHICH LAST
1815                /  ACCESSED THAT DRIVE.  WORD 15 CONTAINS THE NUMBER OF THE
1816                /  INTERFACE LAST POLLED TO SEE IF A COMMAND WAS PENDING.  WORD
1817                /  14 CONTAINS THE NUMBER OF THE INTERFACE CURRENTLY CONNECTED TO
1818                /  THE CONTROLLER.  WORD 13 IS A SECTOR COUNTER USED BY THE
1819                /  "SPARE" SUBROUTINE.  WORD 12 HOLDS THE DRIVE TYPE, RETURNED IN
1820                /  BITS 12-9 OF THE DRIVE STATUS WORD.  THIS WORD DETERMINES HOW
1821                /  MANY HEADS PER DRIVE AND SECTORS PER TRACK THE DRIVE HAS.  IF
1822                /  BIT 3 OF WORD 11 IS SET, THE PLATTER PROTECT SWITCH FOR THE
1823                /  CURRENT SURFACE (7905) OR THE READ-ONLY SWITCH FOR THE ENTIRE
1824                /  DISC PACK (7920, 7925) IS ON, LOCKING OUT WRITING OF ANY KIND.
1825                /  IF BIT 2 IS ON, THE DRIVE'S FORMAT SWITCH IS SET TO OVERRIDE.
1826                /  WORD 8 HOLDS THE NUMBER OF THE LAST DRIVE FOR WHICH THE
1827                /  CONTROLLER PROCESSED AN ATTENTION REQUEST.  COMMAND WORDS
1828                /  SHOULD NOT REFER TO UNIT 8 OR ERRORS MAY OCCUR.
1829                /    ALL INFORMATION IS STORED GROUND TRUE.
1830                /
1831                /
1832                /         BIT  3 2 1 0     WORD
1833                /              ---------
1834                /             ! X I I I !   15
1835                /             ! X I I I !   14
1836                /             ! C C C C !   13
1837                /             ! DRV TYP !   12
1838                /             ! RD ONLY !   11
1839                /             ! I I I H !   10
1840                /             ! I I I H !    9
1841                /             ! PRV ATN !    8
1842                /             ! I I I H !    7
1843                /             .         .
1844                /             .         .
1845                /             ! I I I H !    1
1846                /             ! I I I H !    0
1847                /              ---------
1848                /
1849                /
1850                /    WHERE I IS A 3-BIT NUMBER (0-7) REPRESENTING THE INTERFACE
1851                /            TO WHICH THE ASSOCIATED DRIVE (WORD NUMBER) WAS LAST
1852                /            CONNECTED.  FOR WORDS 15 AND 14 ONLY, THE I-FIELD IS
1853                /            AS DESCRIBED ABOVE THE DIAGRAM.  WORDS 15 AND 14 ARE
1854                /            THE SAME EXCEPT WHEN THE CONTROLLER IS RESPONDING TO
1855                /            A DRIVE ATTENTION BIT.  THERE CAN BE NO DISC DRIVE
1856                /            11-15.
1857                /
1858                /          H IF SET MEANS THAT THE INTERFACE HAS RESERVED (HELD) THE
1859                /            DRIVE FOR A SERIES OF OPERATIONS.  WHILE H IS SET,
1860                /            NO OTHER INTERFACE MAY ACCESS THAT DRIVE.
1861                /
1862                /          C IS A SECTOR COUNTER USED BY SUBROUTINE "SPARE".
1863                /
1864                /          X IS NOT USED.
```

```
1865                          DRIVE NUMBER VERIFICATION AND SELECT ROUTINE
1866                  /
1867                  /
1868                  /
1869                  /
1870                  /    SUBROUTINE UNIT EXAMINES THE UNIT NUMBER FIELD OF THE CPU
1871                  /  COMMAND WORD.  THE FIELD IS TESTED FOR VALIDITY
1872                  /  (0 <= UNIT <= 10), THEN THE WORD IN RAM CORRESPONDING TO
1873                  /  THAT DRIVE IS EXAMINED.  IF THE DRIVE HAS NOT BEEN RESERVED
1874                  /  FOR USE (HOLD BIT NOT SET) BY ANOTHER CPU, IT IS
1875                  /  ASSIGNED TO THE CURRENT CPU.  THE HOLD BIT WILL BE SET ONLY
1876                  /  IF THE HOLD BIT IN THE COMMAND WORD IS SET.  IF THE DRIVE HAS
1877                  /  BEEN RESERVED BY ANOTHER CPU, THE COMMAND IS LEFT PENDING AND
1878                  /  THE CONTROLLER RESUMES POLLING.  IF THE DRIVE IS AVAILABLE OR IS
1879                  /  HELD BY THE CURRENT CPU, IT IS CONNECTED TO THE CONTROLLER
1880                  /  (0 <= UNIT <= 7 ONLY) AND THE CONTROLLER BUSY BIT IS SET ON THE
1881                  /  CURRENT INTERFACE.  THE DRIVE TYPE IS SET IN RAM(12) FROM BITS
1882                  /  12-9 OF THE DRIVE STATUS WORD.
1883                  /    UNIT MUST BE CALLED FROM LEVEL 0, SINCE IT CALLS BSYST WHICH IS
1884                  /  TWO LEVELS DEEP.
1885                  /
1886 1317 66001651 UNIT      CA : CLRST                    / CLEAR STATUS REGISTER (STATR).
1887 1320 07620016           DO : RAMAD  ←   CURNT,L       / GET THE CURRENT INTERFACE NUMBER
1888 1321 10120444           DI : BUF1   ←   RAM  ,L       /   IN THE LOW THREE BITS OF BUF1.
1889 1322 66001751           CA : USTAT                    / PUT CURRENT UNIT# IN STATR,L,
1890                                                       /   CHECK FOR LEGAL DRIVE REQUEST.
1891 1323 16420405           DO : RAMAD  ←   STATR,L       / DRIVE NUMBER O.K., FIND OUT
1892 1324 10020456           DI : BUF3   ←   RAM  ,L       /   IF DRIVE IS BEING HELD.
1893 1325 36437520         BUF3 =        PASS BUF3 ,L,SR   / TEST HOLD BIT.
1894 1326 44640000           CC = TLSB             ,RS
1895 1327 64001333           BR : DRVOK                    / HOLD BIT CLEAR, SAFE TO USE.
1896 1330 36223464        DUMMY = BUF1  XOR  BUF3 ,L       / HOLD BIT SET, CPU'S BETTER MATCH.
1897 1331 44400000           CC = TNZRO                    / DO THEY?
1898 1332 64000061           BR : POLL3                    / NO, GO RESUME POLLING.
1899 1333 37616344 DRVOK BUF3  =        PASS BUF1 ,SL     / O.K. TO USE DRIVE, MERGE CURRENT
1900 1334 36402060        DUMMY =        PASS COMAD,U      /   INTERFACE NUMBER WITH HOLD BIT
1901 1335 44540000           CC = TMSB             ,RS     /   FROM COMMAND AND SAVE IN RAM.
1902 1336 64001340           BR : SVRAM                    / INTERFACE WANTS TO RELEASE DRIVE.
1903 1337 26637401         BUF3 = 1      OR   BUF3 ,L      / IT WANTS TO HOLD DRIVE, SET HOLD BIT
1904 1340 16421455 SVRAM     DO : RAM    ←   BUF3 ,L       / UPDATE RAM FOR THIS DRIVE.
1905 1341 66001611 UNIT1     CA : BSYST                    / ENTRY POINT FROM STATUS OR RQSCT,
1906                                                       /   SET CONTROLLER BUSY.
1907 1342 66001640           CA : DISCO                    / DISCONNECT EVERYONE.
1908 1343 27422410        DUMMY = 10    AND  STATR,L      / CONNECT DRIVE ONLY IF
1909 1344 44400000           CC = TNZRO                    /   0 <= UNIT <= 7.
1910 1345 72000000           RT : 0                        / > 7, RETURN HERE.
1911 1346 17621515           DO : CBUF   ←   COMWD,L       / PUT DRIVE NUMBER ON COMMAND BUS.
1912 1347 07624005 UNIT2     DO : TGBUS  ←   SLECT,L       / ENTRY POINT FROM ATTN ROUTINE.
1913 1350 66001641           CA : STROB                    / SELECT THE DRIVE.
1914 1351 66001551           CA : GSTAT                    / FIND OUT WHAT TYPE IT IS
1915 1352 27502036         BUF1 = 36     AND  BUF2 ,U      /   (IN BUF2 BITS 12-9).
1916 1353 37702204         BUF1 =        PASS BUF1 ,SW     / POSITION IT INTO BUF1(3-0)
1917 1354 37702324         BUF1 =        PASS BUF1 ,SR
1918 1355 07620014           DO : RAMAD  ←   DRTYP,L       /   AND SAVE IT FOR ALL
1919 1356 17620445           DO : RAM    ←   BUF1 ,L       /   OPERATIONS ON THIS DRIVE.
1920 1357 72000000           RT : 0
```

```
1921                          TRACK SPARING ROUTINE
1922                  /
1923                  /
1924                  /
1925                  /
1926                  /  SUBROUTINE SPARE IS THE ROUTINE THAT IMPLEMENTS THE ACTUAL
1927                  /  SPARING ALGORITHM.  IT FOLLOWS A CALL TO VRFY AND USES THE
1928                  /  ADDRESSES SAVED IN THSAD AND TCYLD TO ACCESS THE SPARE TRACK.
1929                  /  THE ENTRY POINT ("SPARE") IS NOT AT THE TOP OF THE CODE SO
1930                  /  THAT DTERR CAN FALL INTO SPARE IN CASE OF RETRIES.
1931                  /
1932                  /
1933                  /
1934                  /
1935                  /  IN THE PROCESS OF TRYING TO SPARE, A DATA ERROR HAS BEEN ENCOUNTERED
1936                  /  SO WE CAN'T RELY ON THE CONTENTS OF THSAD AND TCYLD BEING VALID.
1937                  /  UP TO THE NEXT 15 SECTORS ARE TRIED, UNTIL A GOOD ONE IS FOUND.
1938                  /  IF ALL 16 SECTORS HAVE DATA ERRORS, THE PROCESS IS ABORTED WITH AN
1939                  /  ERROR STATUS.
1940                  /
1941 1360 07620015 DTERR     DO : RAMAD  ←    NSEC ,L     / GET SECTOR RETRY COUNTER,
1942 1361 10120440           DI : REGO   ←    RAM  ,L     /   UPDATE AND CHECK.
1943 1362 30120060         REGO =        DEC  REGO ,L     / DECREMENT RETRY COUNTER
1944 1363 44140000           CC = LOVER            ,RS    /   AND CHECK FOR UNDERFLOW.
1945 1364 64001416           BR : DTER2                   / TOO MANY RETRIES => QUIT.
1946 1365 17620041           DO : RAM    ←    REGO ,L     / STORE UPDATED RETRY COUNT.
1947 1366 37702204         BUF1 =        PASS BUF1 ,SW    / RESTORE BUF1 FOR NEXT VRFY.
1948 1367 66000320           CA : GDTYP                   / GET DRIVE TYPE FOR INCS
1949                                                      /   (AVOIDS 4-LEVEL NESTING).
1950 1370 66001720           CA : INCS1                   / SET UP NEXT SECTOR
1951 1371 66000527           CA : VRFY1                   /   AND TRY IT FOR SIZE.
1952 1372 37702204 SPARE   BUF1 =        PASS BUF1 ,SW    / TO GET SECT ADDR RIGHT AGAIN.
1953 1373 43340000           CC = ANYER            ,RS    / WAS THERE A DATA ERROR?
1954 1374 64001360           BR : DTERR                   / YES, SEE IF WE'VE TRIED 16 SECTORS
1955 1375 27412340        STATR = 340    AND  THSAD,U     / NO, SET S, P, AND D BITS IN STATR.
1956 1376 27402040        DUMMY = 40     AND  THSAD,U     / CHECK FOR DEFECTIVE TRACK.
1957 1377 44400000           CC = TNZRO                   / SET CC IF D-BIT SET.
1958 1400 37616070         BUF3 =        PASS HSAD ,U     / DO A LITTLE COMMON HOUSEWORK
1959 1401 27505437         HSAD = 37     AND  BUF3 ,U     /   BEFORE BRANCHING, CLEAR SPD
1960 1402 27410037        THSAD = 37     AND  THSAD,U     /   BITS FROM HSAD AND THSAD.
1961 1403 64001420           BR : DFTRK                   / BRANCH HERE ONLY IF D-BIT SET.
1962 1404 36202270        DUMMY = HSAD   XOR  THSAD       / NO D-BIT,
1963 1405 44400000           CC = TNZRO                   /   DO HEAD-SECTORS MATCH?
1964 1406 64001435           BR : HSER                    / NO, THIS TRACK'S IN PAIN.
1965 1407 36203260        DUMMY = TCYLD XOR  CYLAD        / HSAD O.K., SET TO CHECK CYLAD.
1966 1410 44400000           CC = TNZRO                   / DO CYLINDERS MATCH?
1967 1411 64001441           BR : CYLER                   / NO, WE'RE HURTING BAD.
1968 1412 07623377           DO : RETRY  ←    FALSE,L     / ALL OK, CLEAR RETRY BIT.
1969 1413 36704470         HSAD = HSAD   OR   STATR,U     / REPLACE S, P, D, BITS IN HSAD.
1970 1414 37724064         HSAD =        PASS BUF1 ,L     / RESTORE TARGET (DTERR MAY MODIFY HSAD).
1971 1415 72000000           RT : 0                       /   AND RETURN (NO SPARING NECESSARY).
1972                  /
1973                  /
1974                  /  THERE'S NO HOPE LEFT AT ALL; WE SEEM TO HAVE RUN OUT OF RETRIES.
1975                  /  THE ONLY THING WE CAN DO IS TELL SOMEONE ABOUT THE ERROR.
1976                  /
1977 1416 37724064 DTER2   HSAD =        PASS BUF1 ,L     / RESTORE TARGET SECTOR ADDRESS.
1978 1417 64000437           BR : UDTER                   / REPORT UNCORRECTABLE ERROR.
```

```
1979                              TRACK SPARING ROUTINE
1980                    /
1981                    /
1982                    /
1983                    /    A SECTOR WITH NO ERROR BUT MARKED DEFECTIVE MEANS THAT A SPARE
1984                    /  TRACK SHOULD BE ACCESSED BECAUSE THERE IS AN IRRECOVERABLE ERROR
1985                    /  ON SOME OTHER SECTOR.  SPARE TRACK ACCESS OCCURS HERE.  TO AVOID
1986                    /  AN INFINITE LOOP IF THE D-BIT IS SET BUT THE ADDRESS FIELD POINTS
1987                    /  TO THE SAME DEFECTIVE CYLINDER AND HEAD, THIS CONDITION IS TESTED
1988                    /  AND, IF FOUND TO BE TRUE, THE OPERATION IS ABORTED AS IF SPARE
1989                    /  TRACK ACCESS WERE NOT ENABLED.
1990                    /
1991 1420 43540000 DFTRK     CC = SPREN            ,RS    / IS SPARING ENABLED?
1992 1421 64001763           BR : DEFER                   / NO, FLAG ERROR AND QUIT.
1993 1422 36203260        DUMMY = TCYLD XOR  CYLAD        / SPARING ENABLED, TEST FOR
1994 1423 44400000           CC = TNZRO                   /   SPARE ADDRESS = DEFECTIVE ADDR.
1995 1424 64001430           BR : DFTR1                   / CYLAD'S DON'T MATCH, ALL O.K.
1996 1425 36202070        DUMMY = HSAD  XOR  THSAD,U      / CYLAD'S MATCH, HEADS BETTER NOT.
1997 1426 44440000           CC = TNZRO            ,RS
1998 1427 64001763           BR : DEFER                   / THEY DO, THROW THE BUM OUT.
1999 1430 37630064 DFTR1 THSAD =        PASS BUF1 ,L      / RESTORE TARGET SECTOR ADDRESS
2000 1431 37724064        HSAD =        PASS BUF1 ,L      /   (IN CASE OF VRFY RETRIES).
2001 1432 36416260        BUF3 =        PASS THSAD        / SET UP SPARE HEAD-SECT ADDRESS.
2002 1433 14000301           DO : CBUF   ←   TCYLD,C      / SEEK TO THE SPARE TRACK.
2003 1434 64001447           BR : SPAR1                   / DO THE SEEK, THEN RECHECK.
2004                    /
2005                    /
2006                    /  AN ADDRESS ERROR--WE MAY HAVE INCREMENTED OFF THE END OF
2007                    /  A SPARE TRACK, SO RETRY ONCE.
2008                    /
2009 1435 43140000 HSER      CC = RTRYF            ,RS    / HEAD-SECTOR MISMATCH, 2ND TIME?
2010 1436 64001443           BR : CYLR1                   / NO, WE STILL HAVE A CHANCE.
2011 1437 66001744           CA : SBTCK                   / YES, SHOW NO (MORE) MERCY,
2012                                                      /   CHECK IF S-BIT SET.
2013 1440 64001767           BR : HSER1                   / NO, REAL LIVE HEAD-SECTOR ERROR.
2014                    /
2015 1441 43100000 CYLER     CC = RTRYF                   / CYLINDER MISMATCH, 2ND TIME?
2016 1442 64001457           BR : CYLRT                   / YES, THAT'S ONE TIME TOO MANY!
2017 1443 07623000 CYLR1     DO : RETRY  ←   TRUE ,L      / SET RETRY BIT (WE WON'T BE BACK!)
2018 1444 15201311           DO : CBUF   ←   CYLAD,C      / RETRY THE OPERATION BY SEEKING
2019                                                      /   TO THE LOGICAL ADDRESS.
2020 1445 37724064        HSAD =        PASS BUF1 ,L      / RESTORE TARGET SECTOR ADDRESS
2021 1446 37616270        BUF3 =        PASS HSAD         /   (IN CASE OF VRFY RETRIES).
2022 1447 07624007 SPAR1     DO : TGBUS  ←   SEEKC,L
2023 1450 66001641           CA : STROB
2024 1451 66000320           CA : GDTYP                   / GET DRV TYPE FOR SEEK3.
2025 1452 66000261           CA : SEEK3                   / FINISH THE SEEK.
2026 1453 66001725           CA : OUTSC                   / (REQUIRED IF DFTRK-SPAR1,
2027                                                      /   REDUNDANT IF HSER OR CYLER).
2028 1454 66001547           CA : SWAIT                   / WAIT FOR SEEK COMPLETION,
2029 1455 66000524           CA : VRFY2                   /   THEN RE-VERIFY NORMALLY.
2030 1456 64001372           BR : SPARE                   / FINALLY, RECHECK.
2031                    /
2032                    /
2033                    /  CYLINDER ADDRESS ERROR, CHECK IF SPARE TRACK INDICATOR IS SET.
2034                    /
2035 1457 66001744 CYLRT     CA : SBTCK                   / CHECK IF S-BIT SET.
2036 1460 64001770           BR : CYLR2                   / NO, GEN-U-WINE CYLINDER ERROR.
```

```
                        INCREMENT HEAD-SECTOR
                /
                /
                /
                /  THE FOLLOWING ROUTINES (INCHS, SECTR AND DECS, TOGETHER WITH
                /  INCS) ARE RESPONSIBLE FOR SELECTING THE NEXT CYLINDER, HEAD AND
                /  SECTOR TO BE ACCESSED.  INCS (INCREMENT SECTOR) INCREMENTS THE
                /  SECTOR FIELD OF HSAD BEFORE IT IS OUTPUT (THE LAST LOGICAL SECTOR
                /  IS INCREMENTED TO 0 WITHOUT CHANGING THE HEAD ADDRESS).  (INCS IS
                /  LOCATED AFTER STRT2 TO ELIMINATE A BRANCH INSTRUCTION.)  SIMILARLY
                /  DECS DECREMENTS THE SECTOR FIELD OF HSAD, WITH 0 DECREMENTING TO
                /  THE LAST LOGICAL SECTOR WITHOUT CHANGING THE HEAD ADDRESS.  AN
                /  ADDITIONAL USE OF DECS IS DESCRIBED FARTHER DOWN THIS PAGE.  INCHS
                /  (INCREMENT HEAD AND SECTOR) AND SECTR ARE MORE COMPLICATED AND
                /  DEPEND ON A NUMBER OF FACTORS:
                /
                /  IT IS THE FUNCTION OF INCHS TO DETERMINE WHETHER END-OF-CYLINDER
                /  OR END-OF-TRACK HAS BEEN REACHED, BASED ON ADDRESS MODE AND SECTOR
                /  ADDRESS, AND TRANSMIT THE NEXT SECTOR OR HEAD-SECTOR TO THE DRIVE
                /  AS REQUIRED.  THE ADDRESS MODES ARE:
                /
                /     1.  SURFACE--HEADS WILL NOT BE SWITCHED AT THE END OF
                /         ANY TRACK.  THE OPERATION MAY CONTINUE ON THE SAME
                /         HEAD, NEXT CYLINDER.
                /
                /     2.  CYLINDER--A CYLINDER CONSISTS OF ALL TRACKS AT A GIVEN
                /         ADDRESS.  HEAD SWITCHING WILL BE ALLOWED IN A DOWNWARD
                /         MANNER, WITH HEAD 0 OF THE NEXT CYLINDER FOLLOWING THE
                /         HIGHEST-NUMBERED HEAD OF THE CURRENT TRACK.  STORAGE IS
                /         CONSIDERED ALL FIXED OR ALL REMOVABLE.
                /
                /  THE HEAD ADDRESS IS TRANSMITTED ONLY IF THE CONTROLLER SWITCHES
                /  HEADS, OTHERWISE ONLY THE SECTOR ADDRESS IS TRANSMITTED SINCE
                /  THE LOGICAL AND PHYSICAL HEAD ADDRESSES MAY NOT AGREE IF SPARING
                /  HAS OCCURED.  INCHS IS CALLED AT THE END OF EVERY SECTOR.
                /
                /  SECTR IS CALLED AFTER INCHS ONLY IF THERE IS MORE DATA TO TRANSFER.
                /  ITS FUNCTION IS TO PERFORM AN INCREMENTAL OR DECREMENTAL SEEK (AND
                /  VERIFY) IF REQUIRED, AND A NOP IF NOT.  IF A NEW TRACK IS STARTED
                /  DURING ANY DATA TRANSFER EXCEPT WFS, RFS OR INIT (WHETHER BY SEEK
                /  OR BY SWITCHING HEADS WHILE IN CYLINDER MODE), A SECTOR IS VERIFIED
                /  (AND A SPARE TRACK IS ACCESSED IF NECESSARY) BEFORE CONTROL RETURNS
                /  TO THE DATA TRANSFER COMMAND PROCESSOR.  IF INCREMENTAL (DECREMENTAL)
                /  SEEK IS REQUIRED BUT IS NOT ALLOWED BY THE FILE MASK, AN END-OF-
                /  CYLINDER (EOC) ERROR INTERRUPT IS TAKEN.
                /
                /  DECS IS CALLED AT THE START OF ALL DATA TRANSFERS.  IT OPERATES
                /  AS DESCRIBED IN THE FIRST PARAGRAPH UNLESS EOC OR THE NEW TRACK
                /  FLAG (NWTRF) IS SET.  IF EOC IS STILL SET FROM THE PREVIOUS DATA
                /  TRANSFER, THE OPERATION IS CONSIDERED A CONTINUATION (THAT IS,
                /  NO SEEK OR ADDRESS RECORD COMMAND WAS PERFORMED SINCE THEN) AND
                /  INCREMENTAL (DECREMENTAL) SEEK OCCURS IF ALLOWED.  IF NWTRF IS
                /  STILL SET, THE OPERATION IS ALSO A CONTINUATION AND DECS IS A
                /  NOP.  THAT IS, THE LAST SECTOR PROCESSED WAS NUMBER N-1 (ON AN
                /  N SECTOR PER TRACK DISC), AND THE FIRST SECTOR TO BE VERIFIED
                /  ON THE NEXT TRACK IS ALSO NUMBER N-1.
                /
                /  EOC AND NWTRK ARE CLEARED BY SEEK, ADDRESS RECORD, OR INCREMENTAL
                /  SEEK DONE BY SECTR OR DECS.
```

```
2096                            SUBROUTINES:  INCHS, SECTR
2097                  /
2098                  /
2099                  /
2100                  /
2101 1461 66000320 INCHS    CA : GDTYP                    / GET # HDS, SCTRS, THIS DRIVE.
2102 1462 32222070       DUMMY = HSAD  PLUS BUF2 ,L     / CHECK FOR LAST SECTOR.
2103 1463 44140000          CC = LOVER           ,RS
2104 1464 64001724          BR : NTLST                    / NO, JUST INCREMENT SECTOR.
2105 1465 07603400          DO : NWTRK  ←   TRUE ,U     / YES, SET NEW TRACK FLAG.
2106                  /
2107                  /  END OF TRACK.  USE MODE TO DETERMINE WHETHER END-OF-CYLINDER.
2108                  /
2109 1466 43640000          CC = CYSRF           ,RS    / SURFACE MODE?
2110 1467 07624003          DO : TGBUS  ←   XSECT,L     / IF SO, SET TO RE-SEND SECTOR ONLY
2111 1470 64001501          BR : SURFC                    / SUCCEEDS ONLY IF SURFACE MODE.
2112                  /
2113                  /  CYLINDER MODE
2114                  /
2115 1471 37616070       HSADB =        PASS HSAD ,U     / PUT HSAD IN B REGISTER AND
2116 1472 27503437        BUF1 = 37     AND  HSADB,U     /   DELETE S, P, AND D BITS.
2117 1473 32202064       DUMMY = BUF1   PLUS BUF2 ,U     / CHECK FOR LAST SURFACE.
2118 1474 44340000          CC = UOVER           ,RS    / SET CC IF NOT LAST SURFACE.
2119 1475 22305401        HSAD = 1      PLUS HSADB,U     / ASSUME NOT (JUST BUMP HEAD).
2120 1476 07624006          DO : TGBUS  ←   ADREC,L     / SET UP HEAD-SECTOR COMMAND.
2121 1477 64001726          BR : OUTAD                    / SUCCEEDS ONLY IF NOT LAST SURF.
2122 1500 27505740        HSAD = 340    AND  HSADB,U     / LAST SURFACE, SET HEAD 0
2123 1501 07605400 SURFC    DO : EOC    ←   TRUE ,U     /   AND END-OF-CYLINDER.
2124 1502 64001726          BR : OUTAD                    / THEN GO SEND HEAD-SECTOR (CYL)
2125                                                      /   OR SECTOR ONLY (SURFC).
2126                  /
2127                  /
2128                  /  SUBROUTINE SECTR PROPERLY BELONGS IN INCHS, SINCE IT FINISHES
2129                  /  SETTING THE NEXT DISC ADDRESS.  IT HAS BEEN SEPARATED OUT TO
2130                  /  AVOID END-OF-CYLINDER ERRORS (AUTO SEEK DISABLED) OR AN EXTRA
2131                  /  SEEK (ENABLED) IF TRANSFER ENDS AT EOC.  SECTR MUST BE CALLED
2132                  /  FROM LEVEL 0, SINCE IT CALLS SPARE WHICH IS TWO LEVELS DEEP.
2133                  /
2134 1503 43400000 SECTR    CC = EOCF                   / END-OF-CYLINDER?
2135 1504 66001530          CA : DECS2                  / YES, DO AUTO SEEK IF ALLOWED.
2136 1505 42540000          CC = NWTRF           ,RS    / NO, DID WE SWITCH TRACKS?
2137                                                      /   (NOTE:  EOCF ==> NWTRF).
2138 1506 72000000          RT : 0                      / NO, RETURN.
2139 1507 07603777          DO : NWTRK  ←   FALSE,U     / YES, CLEAR THE NEW TRACK FLAG.
2140 1510 43100000          CC = INITF                  / SECTR CALLED BY INIT, WFS, RFS?
2141 1511 64001717          BR : INCS                   / YES, SET SCTR 0, SKIP VERIFY
2142                                                      /   (SECTOR ADDRESS IS ALREADY
2143                                                      /   HIGHEST NUMBERED SECTOR IF
2144                                                      /   WE'RE HERE, SO "INCS" WILL
2145                                                      /   SET TO 0, AS WE WANT).
2146 1512 64001711          BR : STRT3                  / OTHERWISE VERIFY ONE SECTOR
2147                                                      /   AND ACCESS SPARE IF REQUIRED.
```

```
2148                              SUBROUTINES:  DECS, DECS2
2149                    /
2150                    /
2151                    /
2152                    /
2153                    /  DECS DECREMENTS THE SECTOR ADDRESS.  FOR FULLER DESCRIPTION,
2154                    /  SEE COMMENTS UNDER "INCREMENT HEAD-SECTOR".
2155                    /
2156 1513 66000320 DECS      CA : GDTYP                    / GET DRIVE TYPE FOR OUTSC, SEEK2.
2157 1514 43400000           CC = EOCF                     / END-OF-CYLINDER STILL SET?
2158 1515 64001527           BR : DECS1                    / YES, ASSUME I-O CONTINUATION.
2159 1516 42500000           CC = NWTRF                    / NO, HAVE WE JUST SWITCHED TRACKS?
2160 1517 07603777           DO : NWTRK  ←   FALSE,U       / YES, CLEAR THE NEW TRACK FLAG
2161 1520 72000000           RT : 0                        /   AND ASSUME AN I-O CONTINUATION.
2162 1521 30124070         HSAD =        DEC  HSAD ,L      / NO, JUST DECREMENT.
2163 1522 44100000           CC = LOVER                    / UNDERFLOW SECTOR 0?
2164 1523 64001725           BR : OUTSC                    / NO, DON'T RESET SECTOR FIELD.
2165 1524 35236060         BUF3 =        CMP  BUF2 ,L      / YES, SET LAST SECTOR...
2166 1525 22325401         HSAD = 1      PLUS BUF3 ,L
2167 1526 64001725           BR : OUTSC                    /   ...AND GIVE IT AWAY.
2168                    /
2169 1527 07603777 DECS1     DO : NWTRK  ←   FALSE,U       / CLEAR THE NEW TRACK FLAG
2170                                                       /   AND FALL INTO DECS2.
2171                    /
2172                    /
2173                    /  SUBROUTINE DECS2 CHECKS WHETHER AUTO TRACK SWITCHING IS
2174                    /  ALLOWED, AND IF SO, WHICH DIRECTION.  IT THEN PERFORMS THE
2175                    /  INDICATED SEEK AND WAITS FOR COMPLETION BEFORE RETURNING.
2176                    /
2177 1530 43740000 DECS2      CC = AUTSK            ,RS     / AUTO INCR-DECR SEEK OK?
2178 1531 64001765           BR : EOCER                    / NO, END-OF-CYLINDER ERROR.
2179 1532 36503260         BUF1 =        PASS CYLAD        / YES, PUT CYLAD IN -A- REGISTER.
2180 1533 10000456           DI : BUF3   ←   SKDIR,U       / GET DIRECTION FLAG (BIT 14).
2181 1534 27403500        DUMMY = 100    AND  BUF3 ,U      / ISOLATE IT.
2182 1535 44440000           CC = TNZRO            ,RS     / INCREMENT OR DECREMENT?
2183 1536 30102264         BUF1 =        DEC  BUF1         / ASSUME DECREMENT.
2184 1537 64001542           BR : DECS3                    / IF TRUE, THIS WILL SUCCEED.
2185 1540 33742264         BUF1 =        INC  BUF1         / NO, INCREMENT WANTED, MUST
2186 1541 33742264         BUF1 =        INC  BUF1         /   INC TWICE TO WIPE OUT DEC.
2187 1542 37614264 DECS3 CYLAD =        PASS BUF1         / RESTORE CYLAD.
2188 1543 15201311           DO : CBUF   ←   CYLAD,C       / MUST DUPLICATE 3 INSTRUCTIONS
2189 1544 07624007           DO : TGBUS  ←   SEEKC,L       /   FROM SEEK1 TO AVOID 4-LEVEL
2190 1545 66001641           CA : STROB                    /   SUBROUTINE NESTING.  SEEK HERE.
2191 1546 66000260           CA : SEEK2                    / FINISH ADDRESSING THE SECTOR.
2192                                                       / FALL INTO SWAIT TO WAIT FOR SEEK
2193                                                       /   COMPLETION.
```

```
2194                                SUBROUTINES:  SWAIT, GSTAT, SWAT3, CST
2195                 /
2196                 /
2197                 /    SUBROUTINE SWAIT WAITS FOR A SEEK TO COMPLETE SO THAT THE
2198                 /  CONTROLLER CAN ISSUE ANOTHER COMMAND TO THE DRIVE.  AT COMPLETION,
2199                 /  IT SAVES THE STATUS OF THE READ-ONLY SWITCH (BIT 6) AND THE FORMAT
2200                 /  SWITCH (BIT 5) FOR THE WRITE ROUTINES TO EXAMINE, AND CLEARS THE
2201                 /  ATTENTION BIT.  THE ENTRY AT SWAT3 ALLOWS OTHER ROUTINES TO CLEAR
2202                 /  THE DRIVE'S ATTENTION BIT.
2203                 /
2204                 /    SUBROUTINE GSTAT GETS STATUS FROM THE DRIVE AND TURNS OVER
2205                 /  THE DRIVE READY BIT SO IT IS TRUE IF THE DRIVE IS NOT READY.
2206                 /    SWAIT AND GSTAT ARE AT THE THIRD LEVEL SINCE THEY ARE CALLED
2207                 /  FROM "SPARE" WHICH IS CALLED FROM "SECTR".  GSTAT FIRST SETS THE
2208                 /  CCB FALSE TO INSURE THAT A CALL TO GSTAT WILL RETURN IMMEDIATELY
2209                 /  AFTER FETCHING STATUS.
2210                 /
2211 1547 07600500 SWAIT    DO : TIMER  ←   START,U    / TURN ON TIME-OUT CIRCUIT
2212 1550 64001552          BR : SWAT1                /   (ONLY FOR SWAIT).
2213 1551 44700000 GSTAT    CC = FALSE                / INSURE RETURN FROM GSTAT.
2214 1552 07624015 SWAT1    DO : TGBUS  ←   STATC,L   / SEND STATUS COMMAND TO DRIVE.
2215 1553 34000000          NO :                      / MUST INSERT PROP DELAY TIME
2216 1554 34000000          NO :                      /   HERE -- SWAIT IS TOO DEEP
2217                                                  /   TO CALL STRB1, ALSO THE CC
2218                                                  /   BIT MAY BE FALSE (GSTAT).
2219 1555 07604477          DO : INTCW  ←   STRB ,U    / VALIDATE THE STATUS COMMAND.
2220 1556 34000000          NO :                      / ALLOW 1 USEC FOR STATUS SET-UP.
2221 1557 34000000          NO :
2222 1560 34000000          NO :
2223 1561 34000000          NO :
2224 1562 10002202          DI : BUF2   ←   CBSIN     / READ IN STATUS
2225 1563 07604437          DO : INTCW  ←   CSTRB,U   /   AND TURN OFF COMMAND.
2226 1564 26230002        BUF2 = 2      XOR   BUF2 ,L   / TURN OVER DRIVE READY
2227                                                  /   (READY _ NOT READY)
2228 1565 35210260         BUF2 =       CMP   BUF2     /   AND THEN THE WHOLE THING.
2229 1566 64001570          BR : SWAT2                / IF SWAIT, DON'T RETURN YET.
2230 1567 72000000          RT : 0                    / IF GSTAT, THIS WILL SUCCEED.
2231 1570 27422026 SWAT2 DUMMY = 26     AND   BUF2 ,L   / CHECK IF DRIVE IS
2232 1571 44400000          CC = TNZRO                /   ABLE TO COMPLETE SEEK.
2233 1572 64001761          BR : STER                 / NO, DRIVE NRDY, SK CHK OR FAULT.
2234 1573 27422001       DUMMY = 1      AND   BUF2 ,L   / YES, IS ACCESS READY
2235 1574 44400000          CC = TNZRO                /   (SEEK COMPLETED)?
2236 1575 64001552          BR : SWAT1                / NO, WAIT UNTIL IT IS.
2237 1576 27520140        REGO = 140    AND   BUF2 ,L   / YES, ISOLATE READ-ONLY
2238 1577 37720120        REGO =        PASS  REGO ,L,SR / AND FORMAT SWITCH STATUSES
2239 1600 37720120        REGO =        PASS  REGO ,L,SR / AND SAVE THEM IN RAM.
2240 1601 37720120        REGO =        PASS  REGO ,L,SR
2241 1602 07620013          DO : RAMAD  ←   RDONL,L
2242 1603 17620041          DO : RAM    ←   REGO ,L
2243 1604 07600600 SWAT3    DO : TIMER  ←   STOP ,U    / TURN OFF TIME-OUT CIRCUIT.
2244 1605 07621376          DO : CBUF   ←   CATN ,L    / THEN CLEAR ATTENTION TO AVOID
2245                                                  /   2ND INTERRUPT.  FALL INTO CST.
2246                 /
2247                 /
2248                 /  CST CLEARS DRIVE ATTENTION AND-OR FIRST STATUS BIT.  IT ASSUMES
2249                 /  THE APPROPRIATE CLEAR BIT(S) ARE IN CBUF ON ENTRY.
2250                 /
2251 1606 07624001 CST      DO : TGBUS  ←   CLST ,L    / PUT OUT CLEAR STATUS COMMAND.
2252 1607 64001641          BR : STROB                / VALIDATE IT AND RETURN.
```

```
2253                                SUBROUTINES:  SETUP, BSYST, FLUSH, STPCL, SRQ, IFDSC, IFSEL
2254                    /
2255                    /
2256                    /
2257                    /
2258                    /  SUBROUTINE BSYST SETS THE CONTROLLER BUSY BIT ON THE CURRENT
2259                    /  INTERFACE, THEN CLEARS THE CURRENT COMMAND FROM THE INTERFACE.
2260                    /  THE SEPARATE ENTRY POINT AT FLUSH CLEARS THE COMMAND WITHOUT
2261                    /  SETTING THE CONTROLLER BUSY BIT (REQUIRED FOR UNIT UNAVAILABLE
2262                    /  AND ILLEGAL OPCODE ERRORS).  CLEARING THE COMMAND HERE ALLOWS
2263                    /  THE CONTROLLER TO REPEATEDLY ACCESS COMMANDS ON THE INTERFACE.
2264                    /  BSYST IS ONLY CALLED WHEN ALL PERTINENT COMMAND INFORMATION HAS
2265                    /  BEEN TRANSFERRED TO THE CONTROLLER AND IT IS ABLE TO BEGIN
2266                    /  EXECUTION.  THE ENTRY POINT AT SETUP CLEARS THE CONTROLLER STATUS
2267                    /  REGISTER FIRST.  SUBROUTINE STPCL (STOP CLOCKS) IS SEPARATED FROM
2268                    /  TRNOF SO THAT THE POLL LOOP CAN CALL IT WITHOUT UPDATING THE 3000
2269                    /  TIO REGISTER.  PLACING STPCL HERE ELIMINATES ONE BRANCH TO ICLK3.
2270                    /  THE FIRST TWO INSTRUCTIONS ARE IRRELEVANT IF WE FALL IN FROM FLUSH.
2271                    /
2272 1610 66001651 SETUP     CA : CLRST
2273 1611 07621400 BSYST     DO : IDTBF  ←   STBSY,L     / SET THE CONTROLLER BUSY BIT.
2274 1612 07622441           DO : IFNBF  ←   BUSY ,L
2275 1613 66001630           CA : ICLCK
2276 1614 07622622 FLUSH     DO : IFNBF  ←   IFGTC,L     / FETCH COMMAND AND DISCARD IT
2277 1615 66001631           CA : ICLK1                  /   (CLEARS I'FC COMMAND BUFFER).
2278 1616 10001744           DI : DUMMY  ←   IDTBF
2279 1617 10001744           DI : DUMMY  ←   IDTBF       / (400 NSEC CLOCK PULSE).
2280 1620 07604437 STPCL     DO : INTCW  ←   CLICW,U     / TURN OFF THE DRIVE COMMAND.
2281 1621 07600600 STPC1     DO : TIMER  ←   STOP ,U     / TURN OFF TIME-OUT CIRCUIT.
2282 1622 64001636           BR : ICLK3                  / TURN OFF INTERFACE FUNCTION.
2283                    /
2284                    /
2285                    /  SUBROUTINE SRQ FORCES A SERVICE REQUEST TO THE 3000 I-O
2286                    /  CHANNEL AFTER EVERY TRANSFER OF CONTROL INFORMATION OTHER THAN
2287                    /  THE COMMAND WORD, AND AFTER EVERY DATA TRANSFER.
2288                    /
2289 1623 07622670 SRQ       DO : IFNBF  ←   RQSRV,L
2290 1624 64001630           BR : ICLCK
2291                    /
2292                    /
2293                    /  SUBROUTINE IFDSC DISCONNECTS ALL CPU INTERFACES.
2294                    /
2295 1625 07622666 IFDSC     DO : IFNBF  ←   DSCIF,L     / SEND "DISCONNECT INTERFACE" FUNCTION
2296 1626 64001630           BR : ICLCK
2297                    /
2298                    /
2299                    /  SUBROUTINE IFSEL SELECTS THE INTERFACE WHOSE NUMBER IS IN IDTBF.
2300                    /
2301 1627 07622454 IFSEL     DO : IFNBF  ←   SELIF,L     / PUT OUT SELECT COMMAND
2302                                                     /   AND FALL INTO IFCLK.
```

```
2303                          SUBROUTINES:  ICLCK, DISCO, STROB, CLRST
2304                  /
2305                  /
2306                  /  SUBROUTINE ICLCK SENDS A 600 NSEC CLOCK (IFVLD) TO THE INTERFACE TO
2307                  /  VALIDATE THE COMMAND IN IFNBF.  THE LEADING EDGE OF THE CLOCK
2308                  /  IS ALSO DELAYED 600 NSEC TO ALLOW THE CONTENTS OF IFNBF TO
2309                  /  PROPAGATE DOWN THE CABLE.  ENTERING AT ICLK1 INSERTS ALL OF
2310                  /  THE ABOVE DELAYS BUT TRANSMITS ONLY THE LEADING EDGE OF IFVLD.
2311                  /  THIS FEATURE IS USED BY THE DATA TRANSFER COMMANDS AND BY THE
2312                  /  GET, PUT, AND FLUSH ROUTINES.  THE ENTRY AT ICLK3 SIMPLY TURNS
2313                  /  OFF IFVLD.
2314                  /
2315 1630 44700000 ICLCK    CC = FALSE                    / INSURE NO RETURN FROM ICLK1.
2316 1631 34000000 ICLK1    NO :                          / ALLOW 600 NSEC PROP TIME.
2317 1632 34000000          NO :
2318 1633 07606000 ICLK2    DO : IFVLD  ←   SETCL,U       / VALIDATE THE COMMAND ON IFNBF.
2319 1634 34000000          NO :                          / LEAVE IT ON FOR 600 NSEC.
2320 1635 72000000          RT : 0                        / THIS SUCCEEDS FOR ICLK1 OR ICLK2.
2321                  /
2322 1636 07606001 ICLK3    DO : IFVLD  ←   CLRCL,U       / DISABLE THE IFNBF COMMAND
2323 1637 72000000          RT : 0                        /   AND RETURN.
2324                  /
2325                  /
2326                  /  SUBROUTINE DISCO DISCONNECTS A DRIVE AT THE END OF AN OPERATION.
2327                  /
2328 1640 07624013 DISCO    DO : TGBUS  ←   DCCOM,L       / PUT OUT DISCONNECT COMMAND
2329                                                      /   AND FALL INTO STROB.
2330                  /
2331                  /
2332                  /  SUBROUTINE STROB OPERATES SIMILARLY TO ICLCK, WITH CORRESPONDING
2333                  /  ENTRY POINTS, BUT VALIDATES THE TAG BUS COMMAND TO THE DISC.
2334                  /
2335 1641 44700000 STROB    CC = FALSE                    / INSURE NO RETURN FROM STRB1.
2336 1642 34000000 STRB1    NO :                          / ALLOW 600 NSEC PROP TIME.
2337 1643 34000000          NO :
2338 1644 07604477          DO : INTCW  ←   STRB ,U       / VALIDATE THE TAG BUS COMMAND.
2339 1645 34000000          NO :                          / LEAVE THERE FOR 600 NSEC.
2340 1646 72000000          RT : 0                        / THIS SUCCEEDS ONLY FOR STRB1.
2341                  /
2342 1647 07604437 STRB2    DO : INTCW  ←   CSTRB,U       / DISABLE THE TAG BUS COMMAND.
2343 1650 72000000          RT : 0                        /   AND RETURN.
2344                  /
2345                  /
2346                  /  SUBROUTINE CLRST CLEARS STATUS FOR ALL COMMANDS BUT
2347                  /  STATUS REQUEST.
2348                  /
2349 1651 27612000 CLRST STATR = 0      PASS       ,U     / CLEAR UPPER BYTE.
2350 1652 27632000       STATR = 0      PASS       ,L     / CLEAR LOWER BYTE.
2351 1653 72000000          RT : 0
```

```
2352                              SUBROUTINES:  GET, PRFCH, WTIO, TRNOF, BSYCL
2353                    /
2354                    /
2355                    /
2356                    /
2357                    /  SUBROUTINE GET READS ONE WORD FROM THE INTERFACE AND RETURNS IT
2358                    /  IN BUF1.  GET IS A SECOND-LEVEL SUBROUTINE.
2359                    /
2360 1654 66001315 GET       CA : DFLST                    / TELL CPU WE'RE READY.
2361 1655 07600500           DO : TIMER  ←   START,U       / TURN ON TIME-OUT CIRCUIT.
2362 1656 42140000 GET1      CC = DTRDY            ,RS     / WAIT FOR TRANSFER
2363 1657 64001656           BR : GET1                     /   OR TIME-OUT INTERRUPT.
2364 1660 07622624           DO : IFNBF  ←   IFOUT,L       / CAN'T CALL DSTR1 -- TOO DEEP.
2365 1661 66001631           CA : ICLK1                    / BUT WE CAN CALL ICLK1.
2366 1662 66000424           CA : CKXFR                    / CHECK FOR 3000 CHANNEL ERROR.
2367 1663 10101744           DI : BUF1   ←   IDTBF         / INPUT ONE WORD
2368 1664 10101744           DI : BUF1   ←   IDTBF         /   (400 NSEC CLOCK PULSE).
2369 1665 64001621           BR : STPC1                    / TURN OFF TIMER, CLEAR VALIDATION.
2370                    /
2371                    /
2372                    /  SUBROUTINE PRFCH PRE-FETCHES THE NEXT COMMAND FROM THE INTER-
2373                    /  FACE, THAT IS, IT OBTAINS THE COMMAND BUT DOES NOT TELL THE
2374                    /  INTERFACE ABOUT IT.  THIS ALLOWS COMMANDS TO BE ACCESSED
2375                    /  REPEATEDLY IN CASE THE DESIRED DRIVE IS NOT AVAILABLE.  ALL
2376                    /  COMMANDS ARE EVENTUALLY CLEARED VIA A CALL TO SUBROUTINE FLUSH
2377                    /  OR BSYST.
2378                    /
2379 1666 07622635 PRFCH     DO : IFNBF  ←   IFPRF,L       / PRE-FETCH THE COMMAND (ENABLES
2380 1667 66001631           CA : ICLK1                    /   INTERFACE DRIVERS, BUT
2381 1670 10101754           DI : COMWD  ←   IDTBF         /   FUNCTION IS NOT ASSIGNED).
2382 1671 34010214        COMAD =        CMP  COMWD,SW     / COMPLEMENT COMMAND, EXCHANGE BYTES.
2383 1672 27430037        COMAD = 37     AND  COMAD,L      / MASK OFF S, P, AND D BITS.
2384 1673 64001636           BR : ICLK3                    / TURN OFF IFVLD, RETURN.
2385                    /
2386                    /
2387                    /  SUBROUTINE WTIO WRITES THE CONTROLLER STATUS REGISTER (STATR)
2388                    /  TO THE 3000 TIO REGISTER.
2389                    /
2390 1674 35302660 WTIO    BUF1 =         CMP  STATR
2391 1675 07622445 WTIO1     DO : IFNBF  ←   WRTIO,L       / SEND "WRITE TIO" TO INTERFACE.
2392 1676 66001631           CA : ICLK1                    / VALIDATE THE INTERFACE COMMAND.
2393 1677 17600745           DO : IDTBF  ←   BUF1          / NOW SEND THE DATA WORD
2394 1700 17600745           DO : IDTBF  ←   BUF1          /   (400 NSEC IFCLK PULSE).
2395 1701 64001621           BR : STPC1                    / TURN OFF TIME-OUT AND IFVLD.
2396                    /
2397                    /
2398                    /  SUBROUTINE TRNOF DOES A LOT OF THE BUSY WORK OF SHUTTING
2399                    /  THINGS DOWN AT THE END OF A CONTROLLER OPERATION.
2400                    /
2401 1702 66001620 TRNOF     CA : STPCL                    / STOP ALL THE CLOCKS.
2402 1703 66001674           CA : WTIO                     / SEND STATUS TO 3000 TIO REGISTER.
2403 1704 07621401 BSYCL     DO : IDTBF  ←   CLBSY,L       / SEND CLBSY FUNCTION TO DATA BUFFER.
2404 1705 07622441           DO : IFNBF  ←   BUSY ,L       / SEND SET-CLR COMMAND TO INTERFACE.
2405 1706 64001630           BR : ICLCK                    / CLEAR THE CONTROLLER BUSY BIT.
```

```
2406                                SUBROUTINES:  STRT1, STRT2, INCS
2407                 /
2408                 /
2409                 /
2410                 /
2411                 /  STRT1 AND STRT2 SET UP THE INITIAL SECTOR FOR DATA TRANSFER
2412                 /  OPERATIONS.  THEY MUST BE CALLED FROM LEVEL 0.  STRT1 IS CALLED IF
2413                 /  PREVIOUS SECTOR VERIFICATION AND-OR SPARE TRACK ACCESSING IS DONE
2414                 /  BEFORE DATA TRANSFER.  STRT2 SKIPS THIS PROCESS.  THE ENTRY AT
2415                 /  STRT3 IS USED BY THE SECTR SUBROUTINE WHENEVER WE SWITCH TRACKS
2416                 /  DURING A DATA TRANSFER.
2417                 /
2418 1707 66001513 STRT1     CA : DECS                    / CHECK FOR EOC, SEEK IF REQ'D.
2419 1710 66001547           CA : SWAIT                   / MAKE SURE OF NO SEEK CHECK.
2420 1711 66000523 STRT3     CA : VRFY                    / VERIFY PRECEDING SECTOR.
2421 1712 66001372           CA : SPARE                   / ACCESS SPARE TRACK IF REQD.
2422 1713 64001717           BR : INCS                    / RE-ADDRESS PROPER SECTOR.
2423                 /
2424 1714 66001513 STRT2     CA : DECS                    / CHECK FOR EOC, SEEK IF REQ'D.
2425 1715 66001547           CA : SWAIT                   / MAKE SURE OF NO SEEK CHECK.
2426 1716 07623000           DO : INITL  ←   TRUE, L      / SKIP VERIFY IF WE SWITCH TRACKS.
2427                                                      / FALL INTO INCS.
2428                 /
2429                 /
2430                 /  SUBROUTINE INCS INCREMENTS THE SECTOR ADDRESS WITHOUT CHANGING
2431                 /  THE HEAD OR CYLINDER ADDRESSES.  THE VARIOUS OTHER ENTRY POINTS
2432                 /  (NTLST, OUTSC, OUTAD, OUTA1) ARE USED BY OTHER ROUTINES TO
2433                 /  TRANSMIT SECTOR ADDRESSES TO THE DRIVE.  THE MAPPING FROM
2434                 /  CONTROLLER LOGICAL TO DRIVE PHYSICAL SECTOR ADDRESSES OCCURS
2435                 /  HERE ONLY.  (THE INVERSE MAPPING FOR THE RQSCT COMMAND IS DONE IN
2436                 /  THAT COMMAND PROCESSOR).
2437                 /
2438 1717 66000320 INCS      CA : GDTYP                   / GET DRIVE TYPE FOR LATER.
2439 1720 32222070 INCS1 DUMMY = HSAD  PLUS BUF2 ,L      / CHECK FOR LAST SECTOR.
2440 1721 44140000           CC = LOVER          ,RS
2441 1722 64001724           BR : NTLST
2442 1723 27724377 LAST    HSAD = 377   PASS       ,L    / YES, SET SECTOR 0.
2443 1724 33764070 NTLST   HSAD =       INC  HSAD ,L     / NO, JUST INCREMENT SECTOR.
2444 1725 07624003 OUTSC     DO : TGBUS  ←   XSECT,L     / PUT OUT SECTOR ADDRESS ONLY.
2445 1726 37616270 OUTAD   BUF3 =       PASS HSAD        / INTO BUF3 FOR COMPATIBILITY
2446                                                      /   WITH SEEK2, SEEK3.
2447 1727 22322377 OUTA1   BUF1 = 377   PLUS BUF2 ,L     / SET UP -(# SECTORS PER TRACK).
2448 1730 32223464        DUMMY = BUF1  PLUS BUF3 ,L     / CHECK FOR ILLEGAL
2449 1731 44100000           CC = LOVER                  /   SECTOR ADDRESS.
2450 1732 64001742           BR : OUTA2                  / YES, DON'T MODIFY IT.
2451 1733 27501437         REGO = 37    AND  BUF3 ,U    / O.K., CONVERT LOGICAL ADDRESS
2452 1734 37700200         REGO =       PASS REGO ,SW   /   TO PHYSICAL ADDRESS BY ADDING
2453 1735 32237460         BUF3 = REGO  PLUS BUF3 ,L    /   HEAD ADDRESS AND TAKING THE
2454 1736 32223464        DUMMY = BUF1  PLUS BUF3 ,L     /   RESULT MOD (# SCTRS PER TRACK).
2455 1737 44140000           CC = LOVER          ,RS    / IS SCTR ADDR > # SCTRS PER TRACK?
2456 1740 64001742           BR : OUTA2                  / NO, SAFE TO SEND IT TO DRIVE.
2457 1741 32237464         BUF3 = BUF1  PLUS BUF3 ,L    / YES, SUBTRACT (# SCTRS PER TRACK).
2458 1742 15201715 OUTA2     DO : CBUF   ←   BUF3 ,C     / SHIP OUT HEAD-SECTOR TO DRIVE.
2459 1743 64001641           BR : STROB
```

```
2460                          SUBROUTINES:  SBTCK, USTAT, CONTROLLER ERROR STATUS SIEVE
2461                 /
2462                 /
2463                 /
2464                 /
2465                 /  SUBROUTINE SBTCK IS CALLED FOR DISC ADDRESS MISCOMPARES.  IT
2466                 /  CHECKS WHETHER THE MISCOMPARE IS DUE TO TRYING TO DIRECTLY
2467                 /  ADDRESS A SPARE TRACK (S-BIT SET).  IF SO, WE BRANCH DIRECTLY
2468                 /  TO SPER (SPARE TRACK ERROR), OTHERWISE WE RETURN.
2469                 /
2470 1744 37724064 SBTCK  HSAD =       PASS BUF1 ,L     / RESTORE SECTOR SAVED BY VRFY.
2471 1745 27402600       DUMMY = 200   AND  STATR,U     / CHECK IF S-BIT SET.
2472 1746 44400000          CC = TNZRO
2473 1747 64001764          BR ! SPER                    / YES, REPORT SPARE TRACK ERROR.
2474 1750 72000000          RT : 0                       / NO, RETURN.
2475                 /
2476                 /
2477                 /  SUBROUTINE USTAT SETS THE CURRENT UNIT NUMBER (BUT NOT THE
2478                 /  HOLD BIT) IN THE LOWER BYTE OF STATR.  IT THEN CHECKS THE
2479                 /  USER-REQUESTED DRIVE NUMBER TO SEE THAT IT IS WITHIN THE LIMITS
2480                 /  OF THE CONTROLLER.  IF NOT, A "DRIVE UNAVAILABLE" ERROR RETURN
2481                 /  IS TAKEN.
2482                 /
2483 1751 34032074 USTAT STATR =       CMP  CONWD,L     / PUT UNIT # IN STATR,L.
2484 1752 27432417       STATR = 17    AND  STATR,L
2485 1753 22222765       DUMMY = MXDRV PLUS STATR,L     / MUST BE 10 OR LESS (OPERATING
2486 1754 44140000          CC = LOVER           ,RS    /   SYSTEM MAY IMPOSE A LESSER
2487                                                    /   MAXIMUM).
2488 1755 72000000          RT : 0                       / DRIVE NUMBER O.K., RETURN.
2489                                                    / NO GOOD, FALL INTO BSYER.
2490                 /
2491                 /  THIS SECTION SETS UP CONTROLLER ERROR STATUS IN STATR AND
2492                 /  REQUESTS AN INTERRUPT.
2493                 /
2494 1756 66001614 BSYER     CA : FLUSH                  / CLR CMND BUFR BEFORE REPORTING
2495 1757 22212401        STATR = 1     PLUS STATR,U     /   UNIT UNAVAILABLE ERROR    (27).
2496 1760 22212403 WPER   STATR = 3     PLUS STATR,U     / WRITE ON PROT-DEF CYLINDER  (26).
2497 1761 22212401 STER   STATR = 1     PLUS STATR,U     / STATUS-2 ERROR              (23).
2498 1762 22212401 ACRER  STATR = 1     PLUS STATR,U     / ACCESS READY ERROR          (22).
2499 1763 22212401 DEFER  STATR = 1     PLUS STATR,U     / DEFECTIVE TRACK             (21).
2500 1764 22212404 SPER   STATR = 4     PLUS STATR,U     / SPARE TRACK                 (20).
2501 1765 22212402 EOCER  STATR = 2     PLUS STATR,U     / END-OF-CYLINDER ERROR       (14).
2502 1766 22212401 IOPER  STATR = 1     PLUS STATR,U     / I-O PROG ERROR (3000 ONLY)  (12).
2503 1767 22212402 HSER1  STATR = 2     PLUS STATR,U     / HEAD-SECTOR COMPARE ERROR   (11).
2504 1770 22212406 CYLR2  STATR = 6     PLUS STATR,U     / CYLINDER COMPARE ERROR      (07).
2505 1771 22212401 OPCER  STATR = 1     PLUS STATR,U     / ILLEGAL OPCODE ERROR        (01).
2506 1772 66001702          CA : TRNOF                  / CLOCKS OFF, WRT TIO, CLEAR BUSY.
2507 1773 07622677          DO : IFNBF  ←   STINT,L     / NOW INTERRUPT THE CPU.
2508 1774 66001630          CA : ICLCK
2509 1775 64000444          BR : NRTER                  / CLEAN UP, WAIT FOR NEXT COMMAND.

                          ORG = 1777
     1777 06026505    CHECKSUM = 06026505

   0 ERRORS, LAST ADDRESS USED = 1775
```

APPENDIX I
MICROASSEMBLER INSTRUCTION DESCRIPTION

Described below is each type of instruction used by the 13037 firmware. These instructions are categorized as to type, and reference should be made to the instruction format of figure 1. The actual listing is organized as follows:

332	0001	17620405	CLHLD	DO : RAMAD ← COUNT,L	/
listing line number	*ROM address*	*microcode (octal)*	*label*	*microcode instruction*	*comment field*

ARITHMETIC INSTRUCTIONS

FORM:	TARGET REGISTER	=	REG	OPERATION	REG, BYTE, SHIFTER OPERATION
EXAMPLES:	HSAD	=	HSAD	AND	BUF3, U *(shifter defaults to PASS mode)*
	HSAD	=		PASS	REG0, SL *(second register is not needed so it is left unspecified; operation is full word so no byte need be specified; shifter will shift left one bit)*
	DUMMY	=	REG0	XOR	BUF3, L *(no target register; operation meant solely to update T register and process status register for subsequent indirect branch or conditional test)*
FORM:	TARGET REGISTER	=	IMMEDIATE OPERAND	OPERATION	BREG, BYTE, SHIFTER OPERATION
EXAMPLES:	COMWD	=	377	PASS	,L *(immediate operand from firmware written directly into lower byte of register COMWD)*
	DUMMY	=	351	PLUS	COMWD, L *(T register and process status register updated with masked bits of register COMWD; immediate operand always replaces an A register)*

I/O (INPUT/OUTPUT) INSTRUCTIONS

FORM: DO : HARDWARE ← REGISTER, BYTE, COMP

(data output form; register can be replaced by an immediate operand)

EXAMPLES: DO : INITL ← FALSE, L

(reset INITL flip-flop on the device controller; note that FALSE is a lower byte immediate operand)

DO : RAM ← IFCNO, L

(RAM is updated with lower byte contents of register IFCNO)

DO : IDTBF ← BUF3, L, C

(IBUS data buffer is updated with the complement of the lower byte of register BUF3)

FORM: DO : HARDWARE ← IMMEDIATE OPERAND, BYTE

EXAMPLES: DO : TIMER ← START, U

(timer hardware receives the immediate operand pattern START over the upper byte of the MIO bus to enable the timer to count)

DO : CBUF ← CFST, L

(control bus output buffer is updated with immediate operand pattern CFST over the lower MIO bus byte to cause the drive to clear first status)

FORM: DI : REGISTER ← HARDWARE, BYTE

(no shifter field is available as this hardware is bypassed during all data input operations)

EXAMPLES: DI : TSTTN ← TIMER, U

(timeout circuitry is monitored by inputting status via the MIO bus; register TSTTM is updated with these contents)

SET CC (CONDITIONAL) INSTRUCTIONS

FORM: CC = CONDITION SELECT , REVERSE SENSE

EXAMPLES: CC = LOVER

(condition code bit is set with the state of the lower byte overflow bit of the process status register)

CC = CMRDY ,RS

(condition code bit is set with the complement of the state of the command ready flag from the interface)

BRANCH INSTRUCTIONS

FORM:	BR	:	ADDRESS	
EXAMPLES:	BR	:	ATTN3	*(branch to address labeled ATTN3; the actual address can be found in the right-most four digits of the microcode; as with all branches, its success depends on the condition of the CCB)*
	BR	:	I	*(a previous arithmetic instruction computes a twelve bit address used to replace the normal ROM address field for branching)*
FORM:	CA	:	ADDRESS	
EXAMPLES:	CA	:	TRNOF	*(a subroutine call is made to the address labeled TRNOF)*
	CA	:	I	*(a previous arithmetic instruction computes a twelve bit address used to replace the normal ROM address field for subroutine calls)*
FORM:	RT	:	O	*(this is the only form of a subroutine return)*

NOP INSTRUCTIONS

FORM:	NO	:	REGISTER	,BYTE ,L
EXAMPLE:	NO	:	BUF3	*(register BUF3, lower byte, is complemented)*
FORM:	NO	:		
EXAMPLE:	NO	:		*(no operation is performed except to delay subsequent instruction execution by 200 nanoseconds)*

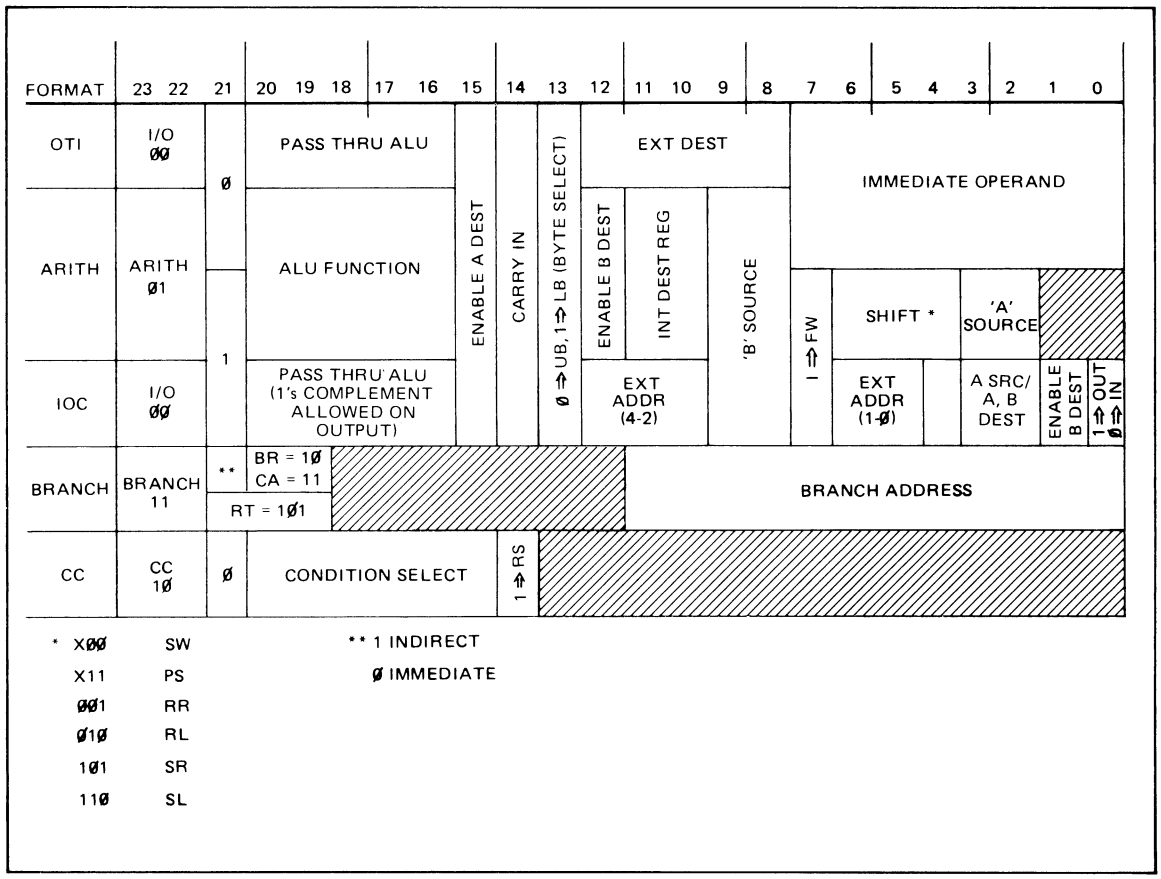


Figure 1. MICROCODE INSTRUCTION FORMAT

APPENDIX II
LABEL CALL ADDRESSES

ACMOD	(1200)	1130,1212
ACRER	(1762)	0542,0611,0626,0720,1050
ADJX	(1250)	1246
ATTN1	(0076)	0101
ATTN2	(0102)	0106
ATTN3	(0131)	0124
ATTN4	(0140)	0453
BOOT	(0215)	0164
BSYCL	(1704)	0041,1104
BSYER	(1756)	1755*
BSYST	(1611)	1106,1266,1341
CDER1	(1140)	1236
CDTER	(0435)	0434*
CKXFR	(0424)	0056,0121,0143,0266,1311,1662
CKWRT	(0775)	0644,1034
CLER	(1441)	1411
CLHLD	(0011)	0023
CLOOP	(1122)	1177
CLR	(1262)	0176
CLRAM	(0026)	0032
CLROF	(0454)	0445,0502
CLRST	(1651)	0044,0157,0313,0346,0426,1317,1610
CLR1	(0035)	0024
CMDWT	(0141)	0271,1105,1261
CMDW1	(0143)	0145
COMP1	(1171)	1167
COMP2	(1172)	1161
COMP3	(1230)	1225
CST	(1606)	0316
CVECT	(0164)	

(*) denotes a call to the corresponding label as a result of "falling through" the previous routine.

```
CYLER   (1441)   1411
CYLRT   (1457)   1442
CYLR1   (1443)   1436
CYLR2   (1770)   1460

DECS    (1513)   1707,1714
DECS1   (1527)   1515
DECS2   (1530)   1504
DECS3   (1542)   1537
DEFER   (1763)   1421,1427
DFLST   (1315)   0463,0505,1315,1654
DFTRK   (1420)   1403
DFTR1   (1430)   1424
DISCO   (1640)   0042,0141,1342
DRVOK   (1333)   1327
DSTRT   (0422)   0411,0473,0615
DSTR1   (0423)   0663,1006,1036
DTERR   (1360)   1374
DTER2   (1416)   1364
DVECT   (0343)

EOCER   (1765)   1531
EWRFS   (0770)   0637,1074
EWRF1   (0771)   0413,1010
EXEC    (0147)   0060,0146*
EXEC1   (0150)   0130
EXEC2   (0161)   0155

FLUSH   (1614)   0037,0156,1756

GDTYP   (0320)   0257,0362,0660,1367,1451,1461,1513,1717
GET     (1654)   0274,0276,0461,0510,1101
GET1    (1656)   0247,1657
GSTAT   (1551)   0304,0447,1351
GTPAT   (1205)   1173
```

(*) denotes a call to the corresponding label as a result of "falling through" the previous routine.

```
HSER    (1435)   1406
HSER1   (1767)   1440

ICLCK   (1630)   0231,0442,1256,1316,1613,1624,1626,1627*,1706,1774
ICLK1   (1631)   0423,1310,1615,1661,1667,1676
ICLK2   (1633)   0136
ICLK3   (1636)   1622,1673
IFDSC   (1625)   0001,0043,0061
IFSEL   (1627)   0055,0116
ILLGL   (0156)   0204,0205
INCHS   (1461)   0475,0515,0673,0771,1150
INCS    (1717)   1511,1713,1716*
INCS1   (1720)   1370
INIT    (0642)   0177
INIT1   (0662)   0654
INIT2   (0664)   0704
INIT3   (0665)   0701
INIT4   (0700)   0675
INTRP   (1257)   0214,1265
IOPER   (1766)   0427

LAST    (1723)
LTIO    (1100)   0207

MOREC   (1120)   1133
MOREP   (1162)   1175

NOERR   (0311)   0307
NRTER   (0444)   1775
NTEQL   (0021)   0016
NTLST   (1724)   1464,1722
NXTAD   (0001)   0000

OPCER   (1771)   0160
OUTAD   (1726)   1477,1502
```

(*) denotes a call to the corresponding label as a result of "falling through" the previous routine.

```
OUTA1   (1727)   0263
OUTA2   (1742)   1732,1740
OUTSC   (1725)   0661,1453,1523,1526
OVERR   (0436)   0606

PLOOP   (1152)   1121
POLL    (0037)   0036*,0211
POLL1   (0040)   0135,0240,0246
POLL2   (0044)   0067
POLL3   (0061)   1332
PRFCH   (1666)   0125,0147
PUT     (1304)   0303,0312,0370,1136,1141,1143,1145,1147,1206,1275
PUT1    (1305)   1306
PWRON   (0025)   0005

RBOOT   (0410)   0232
RCORD   (0272)   0242,0265
READ    (0407)   0171
READ1   (0411)   0415,0421
READ2   (0412)
READ3   (0530)   0430
RECAL   (0233)   0165
RECRD   (0264)   0200
REDE1   (0430)   0412,0474
RFDAT   (0627)   0630,0636
RFS     (0613)   0172
RFULL   (0616)   0641
RNVFY   (0416)   0206
ROFST   (0460)   0202
ROFS1   (0473)   0501
ROFS2   (0502)   0477
ROFS3   (0504)   0520,0667,0773
RQAD    (1266)   0210
RQAD1   (1271)   1137,1207,1267
RQAD2   (1275)   1273
RQAD3   (1303)   1301
```

(*) denotes a call to the corresponding label as a result of "falling through" the previous routine.

```
RQSCT   (0346)   0170
RQSC1   (0366)   0361
RQSC2   (0371)   0317,1151,1270
RQSC3   (0375)   0354,0655
RQSC4   (0400)   0404
RQSC5   (0372)   0373
RQSYN   (1106)   0201

SBTCK   (1744)   1437,1457
SECTR   (1503)   0414,0500,0521,0640,0702,1011,1075
SEEK    (0241)   0166
SEEK1   (0247)   0225,0243
SEEK2   (0260)   1546
SEEK3   (0261)   1452
SETUP   (1610)   0264,1100,1252,1262
SFMSK   (1252)   0203
SHFTX   (1244)   1242
SPARE   (1372)   1456,1712
SPAR1   (1447)   1434
SPER    (1764)   1747
SRQ     (1623)   0245,0270,0443,0462,1103
STATS   (0301)   0167
STER    (1761)   0252,0353,1000,1030,1572
STPCL   (1620)   0040,1702
STPC1   (1621)   1314,1665,1701
STRB1   (1642)   0063,0376,0617
STRB2   (1647)   0406
STROB   (1641)   0034,0236,0255,0457,0470,1264,1350,1450,1545,1607,
                 1640*,1743
STRT1   (1707)   0410,0472,0512,1003
STRT2   (1714)   0417,0614,0643,1033
STRT3   (1711)   1512
SURFC   (1501)   1470
SVRAM   (1340)   1336
SWAIT   (1547)   0234,0464,0471,0503,1454,1546*,1710,1715
SWAT1   (1552)   1550,1575
SWAT2   (1570)   1566
SWAT3   (1604)   0140,0444
```

(*) denotes a call to the corresponding label as a result of "falling through" the previous routine.

```
TERM    (0266)   0374,0506
TRNOF   (1702)   0132,0267,0440,1257,1702

UDTER   (0437)   1417
UNIT    (1317)   0212,0217,0233,0241,0407,0416,0460,0507,0613,0642,
                 1002,1032
UNIT1   (1341)   0301,0350
UNIT2   (1347)   0120
USTAT   (1751)   0314,0347,1751

VCHK    (0431)   0430*,0514
VCYLD   (0547)   0550
VDATA   (0557)   0560,0566
VDTA1   (0567)   0570
VERWD   (0577)   0600,0604
VHSAD   (0553)   0554
VLOOP   (0513)   0522
VRFCM   (0507)   0173
VRFY    (0523)   1711
VRFY1   (0527)   0513,1371
VRFY2   (0524)   1455
VRFY3   (0605)   0767,0770
VSYNC   (0543)   0544

WAKUP   (0212)   0211*
WCYLD   (0733)   0734
WDATA   (0741)   0742,0750
WDTA2   (0751)   0752
WFDAT   (1057)   1060,1066
WFS     (1032)   0175
WFSYN   (1051)   1052,1056
WFULL   (1037)   1077
WFWAT   (1067)   1070
WFWA2   (1072)   1073
WHSAD   (0736)   0737
WLOOP   (1234)   1251
WPER    (1760)   1023
```

(*) denotes a call to the corresponding label as a result of "falling through" the previous routine.

```
WRITE   (1002)   0174
WRLWD   (0765)   0766
WRONE   (0705)   0670,1007
WRTOK   (1007)   1013
WSTA1   (1014)   1004,1012
WSTA2   (1024)   0703,0775,1016,1023*,1076
WSYNC   (0721)   0722,0726
WSYN1   (0727)   0730
WTATN   (0447)   0452
WTEWD   (0757)   0760,0764
WTIO    (1674)   0237,0244,1703
WTIO1   (1675)   1102
```

(*) denotes a call to the corresponding label as a result of "falling through" the previous routine.

CONTENTS

MAJOR 13037
DISC CONTROLLER FEATURES

1.0 INTRODUCTION

The 13037 Disc Controller for the 79XX family of disc drives is designed to interface up to eight disc drives with up to eight CPU's. The 13037 may interface with either 3000 or 21XX CPU's (or both) with provisions for future CPU designs.

One of the primary design goals of the 13037 was to relieve the CPU of much of the busy work associated with data transfer. In previous designs, the CPU would have to interact with the disc drive on long data transfers whenever head and/or track switching was required, or whenever an additional seek to a spare track was required. All these functions can now be handled automatically by the 13037 (although it may be programmed as desired to turn these functions on and off).

In addition to these features, the 13037 is capable of operating in two addressing modes: cylinder mode and surface mode. In cylinder mode (figure 1A), once the heads have been positioned over a particular track, data is recorded starting with the lowest numbered head (0) and continuing to the highest numbered head before the heads are moved to a different track. In this way, a "cylinder" of data is generated. In surface mode (figure 1B), only one particular head is used. When the end of the track is reached, a new track is begun using the same head. Thus, data is recorded only on one surface of the disc.

Another time-saving feature of the controller is the mapping of the bit-per-line drive attention request into a drive unit number before transmitting the attention interrupt to the CPU.

2.0 MULTIPLE CPU DISC DRIVE OPERATION

As mentioned in the first paragraph, the 13037 serves as a clearing house for up to eight CPU interfaces and up to eight disc drives. This requires some special design considerations, which will now be described.

Interface selection: interfaces are numbered from 0 to 7 and are polled by (attached to) the controller serially from 0 to 7 and around again. Any interface which has a command ready when it is attached to the controller will remain connected and have its command serviced.

Attention request: after a SEEK or RECALIBRATE command is put out to the drive, the controller disconnects both the interface and that drive and returns to the interface selection (polling) loop to service other interfaces while waiting for the SEEK or RECALIBRATE to complete. When the drive completes its operation, it sets up an attention request to the controller. The controller checks for these requests in between polls to the interfaces, and connects the next drive in sequence that requests attention to the interface which last accessed that drive.

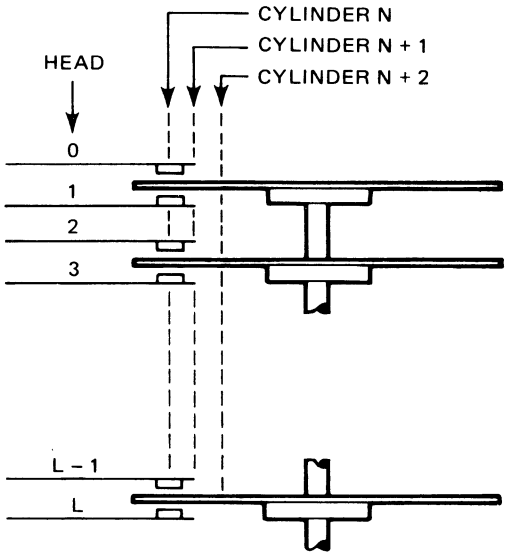


FIGURE 1A. CYLINDER MODE

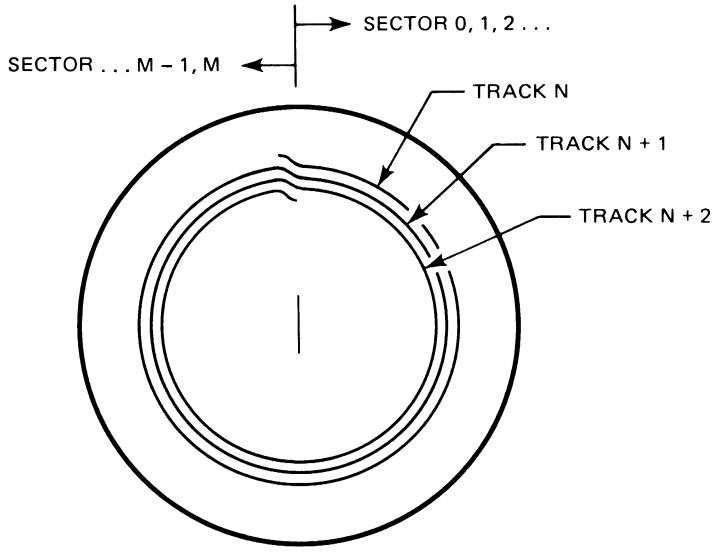


FIGURE 1B. SURFACE MODE

Hold bit: since the controller may service other interfaces between a SEEK (or RECALIBRATE) command from one interface to a given drive and the attention request signaling the end of that operation, what is to stop another interface from accessing that same drive, unknown to the first interface? To prevent this, each drive has a hold bit associated with it. Each command to the controller which references a drive includes a one-bit "hold" field which the programmer may set or clear and which is retained by the controller. While a hold bit is set for a particular drive, no other interface may access it. An attempt to access a held drive will cause the controller to resume polling until that function can be accomplished.

The disc-drive-to-interface mapping is implemented in a small read/write memory on the controller (see figure 2). Each word address (from 0 to 7) in the RAM corresponds to a disc drive, while the contents correspond to the interface which last accessed the drive (left-most three bits) and the hold bit. Figure 2 details what functions the other RAM addresses are assigned.

The entire poll/attention request servicing sequence is shown in the flow-chart of figure 3.

Another consideration of multiple-interface operation is when to disconnect an interface from the controller. In general, the interface cannot be disconnected at the completion of all commands, since abnormal completion status (as well as relevant temporary quantities such as the file mask and disc address registers) would disappear if another interface were allowed to begin a command. Yet, if an interface were always to remain connected, no interface polling could take place.

The 13037 handles this situation by disconnecting from a particular interface only during SEEK, END, or RECALIBRATE instructions, or after a timeout occurs. Note that SEEK and RECALIBRATE disconnects allow holding onto a particular drive for subsequent interface use, while the END command is unconditional.

3.0 THE FILE MASK

The file mask is a four-bit register (bits 3 – 0 of figure 4) which may be set to any desired combination by the programmer. Bits 7 – 4 in the same command word are a retry counter used by the 3000 interface and selector channel to retry data operations in case of an error.

Bit 0, when set, allows the controller to automatically seek to the next higher or lower cylinder address (depending on bit 3) when a data operation continues through the end of the current logical cylinder. The new track is subjected to disc address and track status verification before any further operation is performed. When bit 0 is clear, the option is prohibited, and any data operation which exceeds the end of the current cylinder is aborted with an end-of-cylinder error status.

Bit 1, when set, indicates that the controller is operating in the cylinder mode. In this mode, the first head address is 0 (the top-most head) and head switching occurs automatically following the last data sector of each track. The head address increases incrementally from 0 to the last head address of the particular disc drive. As each new head is switched in, the verification described in the previous paragraph is performed. The last data sector of the last head address of the current track address is the end of the logical cylinder.

When bit 1 is clear, the controller is operating in the surface mode, where the head address never changes. The last sector of the current track address is the end of cylinder. If bit 0 is set (incremental/decremental seek allowed), a data operation which exceeds the end of the cylinder will continue at the next higher/lower cylinder address (subject to address and track status verification).

The following diagram is the layout of the 64-bit (16 x 4) read/write memory which maps the disc unit number (same as the R/W memory address for addresses 0 – 7) into the number of the interface which last accessed that drive. Word 15 contains the number of the interface last polled to see if a command was pending. Word 14 contains the number of the interface currently connected to the controller. Word 13 is a sector counter used by the verification algorithm. Word 12 holds the drive type, returned in bits 12-9 of the drive status word. This word determines how many heads per drive and sectors per track the drive has. If bit 3 of word 11 is set, the platter protect switch for the current surface (7920, 7925) is on, locking out writing of any kind. If bit 2 is on, the drive's Format switch is set to override. Word 8 holds the number of the last drive for which the controller processed an attention request. Command words should not refer to unit 8 or errors may occur.

BIT 3	2	1	0	WORD
X	I	I	I	15
X	I	I	I	14
C	C	C	C	13
D	R V	T	Y P	12
R	D	O N	L Y	11
I	I	I	H	10
I	I	I	H	9
P	R V	A	T N	8
I	I	I	H	7
.	.	.	.	
.	.	.	.	
.	.	.	.	
I	I	I	H	1
I	I	I	H	0

WHERE I is a 3-bit number (0 – 7) representing the interface to which the associated drive (word number) was last connected. For words 15 and 14 only, the I-field is as described above the diagram. Words 15 and 14 are the same except when the controller is responding to a drive attention bit. There can be no disc drive 8 – 15.

H if set means that the interface has reserved (held) the drive for a series of operations. While H is set, no other interface may access that drive.

C is a sector counter used by subroutine "spare".

X is not used.

Figure 2. READ/WRITE Memory Layout

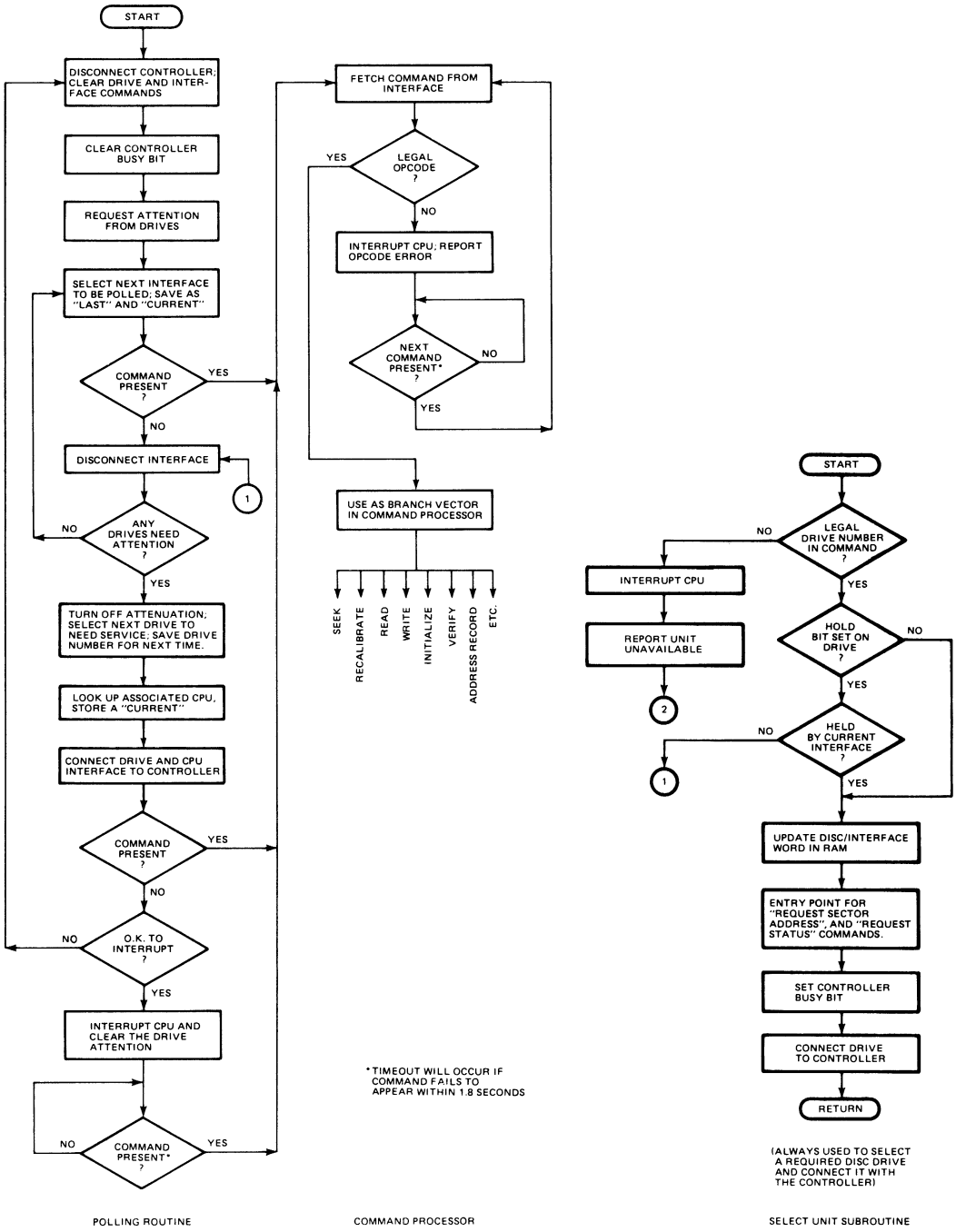


Figure 3. Interface Poll Loop, Command Processor, Select Unit Subroutine

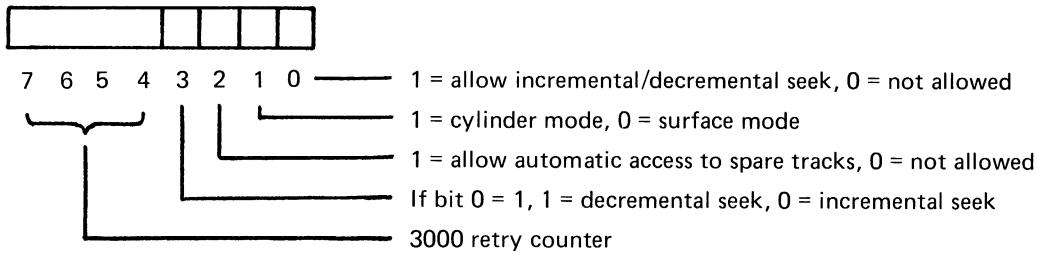


Figure 4. File Mask and Retry Counter

When bit 2 is set, the controller is allowed to automatically seek to previously initialized spare tracks when a defective track (that is, one which has been initialized defective) is encountered. (The entire spare track algorithm will be described later.) If bit 2 is clear, the automatic sparing will not occur and a defective track error status will be reported to the interface.

Bit 3 operates in conjunction with bit 0 and determines whether an automatic end-of-cylinder seek will be to the next higher or lower track address. If bit 3 is 0, the next higher track is selected; if it is 1, the next lower track is selected. Note: If bit 0 is 0, bit 3 is not examined.

4.0 SPARE TRACKS AND THE 13037

4.1 WHY SPARE TRACKS?

Why spare anything? In a disc drive the heads do not contact the magnetic surface (floppy discs excepted) but fly a small distance above it. Even this small distance compounds the complexity of accurate data recovery. Errors, while not common, do occur (for example when a microscopic particle intervenes between the disc surface and the head). The sparing process, used on most (if not all) high-quality disc systems, has two objectives: to recover as much of the data from the bad area as possible, and to flag the bad area so that it will not be used again.

Although the smallest addressable space on the disc is the sector, most files take up many times this space. It would be impractical to maintain the system overhead (including increased access times) necessary to assign and remember spare areas on a sector-by-sector basis. A large part of this overhead includes an address and sector status field which must by physically separate from the data field of the sector. Without this, the address and status could not be verified on the same rotation as the data transfer. The gap between the two fields would be wasted space, thus lowering the overall capacity of the disc.

On the other hand, if an error caused an entire cylinder to be spared and marked defective because of a microscopic area which faulted one sector, vast areas of the disc might soon be marked defective and similarly large spare areas might be required.

As in most other situations, a compromise has been achieved, whereby one track is made to answer for the sins of a dust particle. A track is defined as the area accessible by one head with the head access mechanism positioned at a given cylinder. In the 13037, spare areas are assigned on a track-for-track basis. Although the spare track need not be on the same surface as the defective track, a systematic procedure still must be adhered to by the operating system when assigning and generating the spare track in order for the automatic spare track access algorithm to work properly.

Briefly, when a defective track is first discovered (via a disc parity error), the operating system must assign and initialize a spare track and initialize the bad track as defective. If this is done properly (and automatic track sparing is enabled in the file mask), the system can forget about the spare track from then on; the controller will take care of the extra access required to get to the spare track whose logical address is that of the defective track.

So first we'll examine how to generate a spare track, and then how the 13037 accesses it.

4.2 HOW TO GENERATE A SPARE TRACK

Suppose the operating system, while reading cylinder 100, head 0, sector 0 (abbreviated (100,0,0) from now on), as in figure 5A, determines that a permanent data error exists, and that a spare track must be generated. The system looks up the next available spare track (e.g., (406,1)) in an appropriate table.

The system initializes each sector of the spare track with the disc address of the defective track (note the difference in addresses) with sectors of the spare and defective tracks corresponding one-for-one. (The detailed steps of how to do this are shown below.) The S-bit (spare bit) is set to 1. See figure 5B. Optionally, data may be copied from the defective track to the spare track during the initialization process by reading one sector at a time, ignoring all error status, and then initializing the corresponding sector of the spare track with the data from the defective track. This process risks copying a data error, but preserves the remainder of the valid data in a track if only one sector is defective.

Following the initialization of the spare track, the system then initializes the defective track with the disc address of the spare track (again, note the difference in addresses). This time the D-bit (defective bit) is set to 1. See figure 5C. The system then deletes the just-assigned spare track from its table of available tracks and then forgets about the track completely.

The command steps necessary to generate a spare track are:

1. Enable automatic track sparing.
2. Seek to the first sector of the defective track.
3. Read the sector, ignoring all error status.
4. Seek to the same sector of the spare track.
5. Perform an ADDRESS RECORD command with the defective track address. This sets controller registers to the desired disc address but does not talk to the drive.
6. Perform an INITIALIZE command, transferring the data from the defective sector to the spare sector. Be sure the S-bit is set in the INITIALIZE command word.
7. Repeat steps 2 through 6 for each sector of the track.
8. Seek to the defective track.
9. Verify one sector. This will get the controller to any spare track which may exist as the result of a previous spare track generation.
10. Perform an ADDRESS RECORD command with the cylinder address set to 177777 (octal) (i.e., garbage).
11. Initialize the entire track without intervening seeks. Make sure the S- and D-bits are clear.
12. Seek to the defective track. Do NOT verify, as was done in step 9.
13. Perform an ADDRESS RECORD command with the spare track address.
14. Initialize as many sectors at a time as feasible (the data field is irrelevant). Be sure the D-bit is set in the INITIALIZE command word.
15. Repeat steps 13 and 14 until the entire track has been initialized defective.

Figure 5. Generating a Spare Track

Although it seems that the defective track is being initialized twice (true in most cases), this sequence is required in case a track previously initialized spare fails. This sequence assures that the spare track which failed is given a garbage address so that it can never be accessed during normal operation.

If data is not being copied to the spare track, the spare track can be initialized spare as many sectors at a time as is feasible, just as steps 13 and 14 were used above to initialize the defective track.

4.3 HOW A SPARE TRACK IS ACCESSED

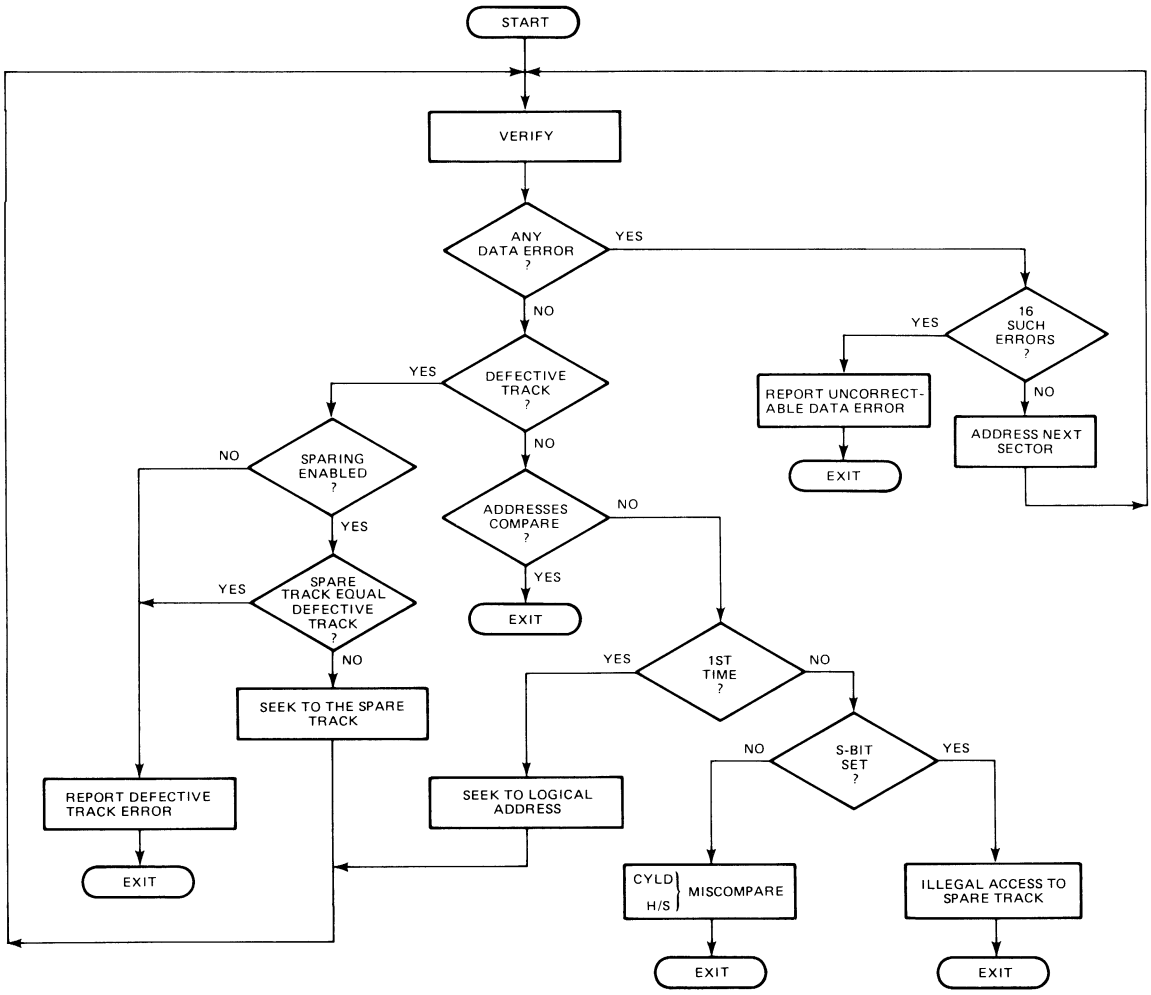
To assure that data is transferred to/from the proper area of the disc, the desired address (in CYLAD and HSAD) is periodically compared to that in the address field of the sector on the disc. The comparisons are made before each transfer and whenever a track boundary is crossed for a VERIFY command and all data transfer commands except INIT, WFS, and RFS. For RNVFY, the comparison is omitted at the start of transfer only.

In addition to address verification, a spare track access algorithm (figure 6) provides automatic 13037 accessing of spare tracks if the feature is enabled in the file mask. With this feature, a computer system need only assign a spare track when a regular track proves defective. It can then forget the spare track and make all accesses through the original logical track address, even though the physical track has a defect. Both spare track accesses and address verification are implemented in subroutine SPARE.

Subroutine SPARE is called only after a verify operation has read the address area of a sector into temporary storage. The subroutine's algorithm is shown below:

```
SPARE:    WHILE DATA'ERROR DO
              BEGIN IF NUMBER'OF'ERRORS >= 16 THEN
                GO TO UNCORRECTABLE'DATA'ERROR;
              INCS;   & NOT AT LIMIT YET, TRY NEXT SECTOR.
              END OF DATA'ERROR;
          IF D'BIT THEN
              BEGIN IF NOT SPARING'ENABLED THEN GO TO
                DEFECTIVE'TRACK'ERROR;
              IF CYLAD = TCYLD AND HSAD'UPPER = THSAD'UPPER
                THEN GO TO DEFECTIVE'TRACK'ERROR;
              SEEK'TO'TCYLD'AND'THSAD;
              VERIFY'SECTOR;
              GO TO SPARE;
              END OF D'BIT;
          IF CYLAD <> TCYLD OR HSAD <> THSAD THEN
            IF ADDRESS'ERROR'FLAG
              THEN GO TO ADDRESS'ERROR
              ELSE BEGIN ADDRESS'ERROR'FLAG  := TRUE;
                   SEEK'TO'CYLAD'AND'HSAD;
                   VERIFY'SECTOR;
                   GO TO SPARE;
                   END OF ADDRESS'ERROR;
          ADDRESS'ERROR'FLAG  := FALSE;
          END OF SPARE;
```

The first thing we must do is check for a read error. If there is one, we can't trust the address area we just read. So we remember our original target sector (done externally), increment to the next sector, and try again. To avoid looping indefinitely, we limit the number of re-reads to 15 (the highest count we can store in word 13 of our scratch-pad RAM.



NOTE THAT ON POSSIBLE ADDRESS MISCOMPARES, THE CONTROLLER WILL SEEK TO THE TARGET ADDRESS IT DESIRES IN AN ATTEMPT TO CORRECT THE MISCOMPARE. IN MULTIPLE CPU ENVIRONMENTS, THIS CAPABILITY CAN CAUSE TRANSFERS ON UNWANTED AREAS OF THE DISC; FOLLOWING EACH "SEEK" OR "RECALIBRATE" INSTRUCTION BY AN ADDRESS RECORD COMMAND WILL PREVENT SUCH OCCURRENCES.

Figure 6. Basic Sparing Subroutine

Once we have a valid address area, we check the defective bit (D-bit). If it's set, we're on a track which was flagged (initialized) defective earlier. If we're allowed to access the spare track, and the defective and spare track addresses are not the same, we seek to the spare track on our own, read (verify) a sector, then start over again. Thus we can go through as many defective tracks as required to get to the spare, as long as we don't get back to where we started (else we would loop endlessly).

If the D-bit isn't set, we compare the address (cylinder, head, and sector) read from the disc (TCYLD and THSAD) with that in CYLAD and HSAD. If they match, we're done. If not, we set a flag, seek on our own to CYLAD and HSAD, and try once more. If the comparison fails again, we report an address error of some kind (cylinder miscompare, head/sector miscompare, or illegal access to spare track).

An example will show the required disc format and how it is used to access the spare track. Consider the two sectors shown (address field only) and the following sequence of events:

PHYSICAL ADDRESS			DISC ADDRESS FIELD					
CYL	HEAD	SECTR	CYL	S	P	D	HEAD	SECTR
100	0	23	406			1	1	23
406	1	23	100	1			0	23

1. 13037 seeks to cylinder 100, head 0, sector 24 (100,0,24).
2. Back up one sector, verify 100,0,23.
3. No errors, but D-bit set.
4. Logical address (100,0,23) ≠ disc address (406,1,23), so seek to spare (disc) address (406,1,23), then forget it. Retain logical address (100,0,23) only.
5. Verify (read) physical sector 406,1,23.
6. Assume no error. Also no D-bit, so compare addresses.
7. Logical address (100,0,23) matches disc address (100,0,23), so bump to sector 24 and begin transfer.
8. At end of track, assume cylinder mode. 13037 will bump its address from 100,0,47 to 100,1,47, while drive will be addressed to 406,2,47 (head carriage doesn't move here, only head and sector are addressed).
9. Because of track crossing, a verify is made and the disc address (406,2,47) is found to differ from the controller target address (100,1,47).
10. A controller generated seek is issued to the target address (100,1,47) and a subsequent verify is made.
11. Disc address read from verify (100,1,47) now matches target address (100,1,47), so the transfer begins with the next sector (100,1,0). (If the verify had produced a set D-bit, sparing would again take place.)

Note that by using this scheme, we can detect a defective track, get to its spare track, transfer data, return to the track following the defective track, and continue transferring data, all without CPU or channel intervention.

4.4 DIRECTLY ACCESSING A SPARE TRACK

What if the channel causes a seek directly to a spare track? The following sequence then takes place (using our earlier address):

1. 13037 seeks to 406,1,24.
2. Back up one, verify 406,1,23.
3. No errors, no D-bit, 406,1,23 ≠ 100,0,23,
4. Set address error flag, seek to logical address 406,1,23, verify.
5. No errors, no D-bit, 406,1,23 ≠ 100,0,23.
6. Flag set, address miscompare established.
7. S-bit set, so report attempt to directly access spare track, rather than address miscompare.

Thus a spare track can only be accessed through its corresponding defective track (except , of course, for commands such as READ FULL SECTOR, which do not perform address comparisons or track status verification).

5.0 AUTOMATIC HEAD AND/OR TRACK SWITCHING

Earlier controllers required CPU intervention to change the head and track addresses whenever a multiple-sector operation (READ, WRITE, VERIFY, INITIALIZE) continued beyond the end of a disc track. The 13037 eliminates the need for this handshake (if automatic head and/or track switching is enabled in the file mask), thus leaving the CPU free until the end of the transfer. The basic assumption made is that logical sector 0 of the next track follows the last logical sector of the current track (whether in cylinder or surface mode).

Consider the hypothetical situation shown in figure 7A, with the controller reading while in cylinder mode. The controller finishes (45,0,47), discovers it has more data to transfer, and is allowed to switch heads. It switches to head 1, but it must verify the disc address before continuing the data transfer (the reason for this will be clear shortly). It must also verify (for WRITE operation only) that the upcoming track has not been initialized protected. The controller could verify sector 0 of the new track to assure itself of proper address and track status. Normally these fields would check, but sector 0 would then have passed and we would have to wait out one rotation of the disc to continue the data transfer.

To prevent this, logical sectors are "staggered" as tracks progress downward on the cylinder (see figure 7B). Thus, as the disc rotates, sector 47 of the next head follows sector 47 of the current head. Address and track status verification can be performed on sector 47 of the new track, then (normally) data transfer can continue immediately with sector 0. Note that the physical sector address (the one the drive itself sees) does not change from track to track. The mapping from logical sector to physical sector is done at only one point in the microcode (at the moment of truth when the address is transmitted to the drive); therefore the controller deals only with logical sectors.

The only other time the difference arises is during the "REQUEST SECTOR" command. The drive returns the physical sector address, which must be inverse-mapped into the logical address before transmitting it to the interface.

The mapping itself is straightforward, and consists of adding the head address to the logical sector address and taking the result Mod (# Sectors/Track). This is the physical sector address sent to the drive (see figure 7B).

Note that all the spare track generation and access algorithms still work, because the controller does all its operations on logical sectors.

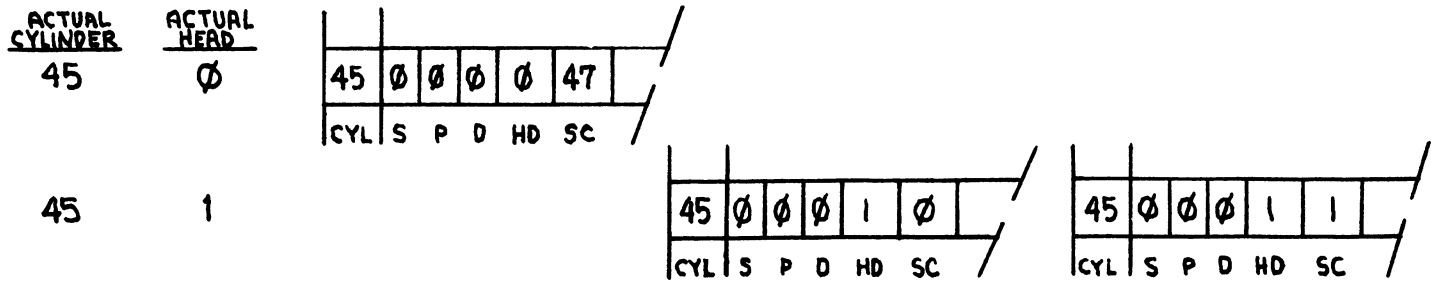


Figure 7A. Hypothetical Disc Layout, Sectors Not Staggered

HEAD	SECTORS				
Ø	47	Ø	1	2	LOGICAL
	47	Ø	1	2	PHYSICAL
1	46	47	Ø	1	LOGICAL
	47	Ø	1	2	PHYSICAL
2	45	46	47	Ø	LOGICAL
	47	Ø	1	2	PHYSICAL

Figure 7B. Logical vs. Physical Sectors

Now why is address verification required at all? (We've already seen why the P-bit must be checked.) Because of the possibility of automatic track sparing. Consider a similar example to the above one (figure 8). The controller has read logical (100,0,47) (spared to physical track and head (406,1), logic sector 47) and has more to do. In cylinder mode it will switch to (406,2,47), another spare track. If there were no verification, the controller would transfer logical (200,1,47) (actually (200,1,0), since without verification no sector offset would be required). This is wrong, since the system expects logical (100,1,0).

What actually happens is as follows: when the controller seeks to the spare track (as described earlier), it immediately forgets the physical cylinder and head address, retaining only the logical address (100,0). When it finishes sector 47, it switches to (it thinks) logical address (100,1,47) (last logical sector, head 1), then attempts to verify that sector. Since it actually switched to (406,2,47) it will read logical address (200,1,47) which fails when compared to (100,1,47).

So even with verification, addresses may not compare. What then? The controller has retained the logical next address (100,1,47), so just in case the address mis-comparison resulted from such a situation, it automatically issues a seek (with wait) to the logical address, then tries once more to verify the sector at that address. If it is successful, the transfer continues. If not, and the track has not been initialized defective, the address error is assumed to really exist and is reported as such to the system. (If it has been initialized defective, the address field contains a spare track address, and a spare track access operation begins anew.)

For an overview of the track verification and sparing operation, see the flowchart, figure 6.

6.0 SOME ROUTINE OVERVIEWS

The following set of routines, DECS, INCHS, INCS, and SECTR, are all used to send the proper address to the disc before all data transfers and between sectors of a multiple-sector transfer. Each routine is described in detail adjacent to its coding. These comments are an overview of their group function.

The user need not re-address the 13037 or disc drive (via an ADDRESS RECORD or SEEK command) between data transfers which progress in either cylinder mode or surface mode. (The user must set the mode using the SET FILE MASK command before starting the transfer.) These terms are defined as follows:

1. CYLINDER MODE — Data is transferred in ascending order of sectors, then in ascending order of heads, and finally in ascending or descending order of cylinders. All heads at a given head carriage position are accessed before the heads are moved. Algorithmically speaking, cylinder mode can be written:

```
FOR CYLINDER  := CYL1 STEP +/-1 UNTIL CYL2 DO
  FOR HEAD  := 0 10 M--1 DO        & M HEADS IN THE DRIVE.
    FOR SECTOR  := 0 TO N--1 DO        & N SECTORS PER TRACK.
      TRANSFER (CYLINDER, HEAD, SECTOR);
```

2. SURFACE MODE — For a given head, data is transferred in ascending order of sectors, then in ascending or descending order of cylinders. Only one head is used, then the heads must be moved to continue the transfer. Algorithmicly:

```
HEAD  := 1;        & 0 <= 1 <= M-1
FOR CYLINDER  := CYL1 STEP +/-1 UNTIL CYL2 DO
  FOR SECTOR  := 0 TO N-1 DO TRANSFER (CYLINDER, HEAD, SECTOR);
```

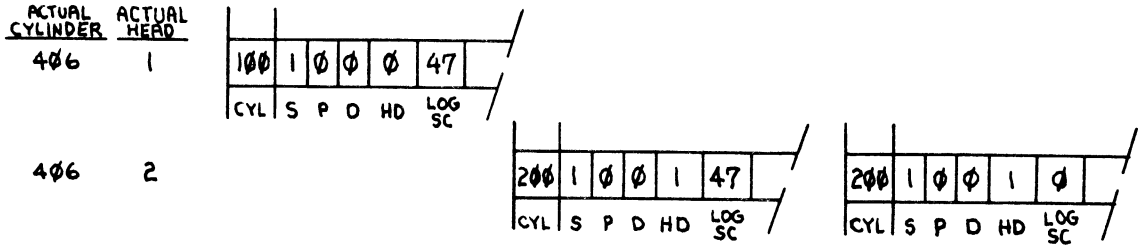


Figure 8. Head Switching at End of Spare Track

The end of logical cylinder occurs whenever the heads must be moved to continue a transfer (whenever the "FOR CYLINDER . . ." statement is executed) in either cylinder or surface mode. Thus, in surface mode, a logical cylinder and a track are equivalent.

The 13037 will automatically maintain the next addressed to be transferred:

1. When a transfer completes normally (no errors), the address in the 13037 is that of the next sector to be transferred. Thus a new transfer request will continue the transfer.
2. When an error occurs, the address in the 13037 is that of the erroneous sector. The system may then retry the transfer, starting with the bad sector.

In either case, the REQUEST DISC ADDRESS (REQAD) command will return the address pointed to by the 13037 (current sector if error, next sector if no error).

By clearing bit 0 of the file mask, the user may prevent the automatic seek at the end of logical cylinder. Transfers which try to continue beyond the end of logical cylinder are aborted with an end-of-cylinder (EOC) error status, although a REQAD command will return the address of the first sector of the new cylinder.

This conflicts with our earlier statement. If a transfer ends without error at the end of a logical cyliner, the 13037 should point to an address on the next cylinder, implying that a seek has occurred. But if the automatic seek bit in the file mask is clear, the seek attempt will cause an EOC error. The user will be puzzled because the data transfer didn't cross a cylinder boundary.

To avoid a possible end-of-cylinder (EOC) error when a transfer ends normally at the end of a logical cylinder, the address update is performed in three steps:

1. The next head and sector address are set by subroutine INCHS (increment head and sector), based on the current address and on cylinder or surface mode. The algorithm is:

```
INCHS:      IF NOT LAST'SECTOR
              THEN SECTOR  := SECTOR + 1
              ELSE BEGIN NEW'TRACK'FLAG  := TRUE;
                   IF CYLINDER'MODE THEN
                     IF NOT LAST'HEAD
                       THEN BEGIN HEAD  := HEAD + 1;
                                  GO EXIT;
                                  END
                       ELSE HEAD  := 0;
                   END'OF'CYLINDER'FLAG  := TRUE;
                   END OF LAST SECTOR OF TRACK;
EXIT: END OF INCHS;
```

Note that 1) a true end-of-cylinder flag (EOCF) implies that the new track flag (NWTRF) is also true, 2) we haven't tried to perform any seek (which could generate an EOC error), we've just left EOCF set to tell us it needs doing if we go on, and 3) if NWTRF is set, we haven't changed sector. This is because we just finished sector N−1, and (usually) we next verify sector N−1 of the next track.

2. The data transfer is checked for completion (end of transmission) in the individual command processors. If so, the 13037 completes command processing leaving EOCF and NWTRF as they were set (or cleared) by INCHS. A REQUEST DISC ADDRESS (REQAD) command issued here will examine EOCF and NWTRF to report the proper next address, not the in-between one actually contained in the 13037.
3. If the transfer is not complete, the user is committed to crossing a cylinder boundary if required, and we can now check the auto seek bit, perform the seek, and generally finish the address updating process. This is done in subroutine SECTR, using the EOCF and NWTRF conveniently left lying around by INCHS:

```
SECTR:      IF END'OF'CYLINER'FLAG THEN     & NEXT CODE IS SUBR DECS2.
                BEGIN IF NOT AUTO'SEEK'ALLOWED THEN GO TO
                  END'OF'CYLINDER'ERROR;
                IF SEEK'DIRECTION = 1
                  THEN CYLINDER  := CYLINDER - 1
                  ELSE CYLINDER  := CYLINDER + 1;
                SEEK2;    & RESETS EOCF, NOT NWTRF.
                SWAIT;    & WAIT FOR SEEK TO COMPLETE.
                END OF DECS2;
            IF NEW'TRACK'FLAG THEN    & WE'RE STARTING A NEW TRACK.
                BEGIN NEW'TRACK'FLAG  := FALSE;
                IF INITIALIZE'GLAG
                  THEN GO TO INCS   & INIT, WFS OR RFS, NO VERIFY.
                  THEN GO TO STRT3;    & MUST VERIFY SECTOR.
                END OF NEW'TRACK'FLAG;
            END OF SECTR;
```

This takes care of multiple-sector transfers, including those which end on a track or cylinder boundary. Now what about a new transfer without (re-)addressing the drive. We generally want to back up one sector, verify one, restore the original sector address, then begin the transfer. This can be done with the following sequence:

```
JSB DECS
JSB SWAIT
JSB VRFY
JSB SPARE
JSB INCS
```

. . . as long as DECS can cope with EOCF and NWTRF if either or both are set. In these cases, DECS functions in much the same way as SECTR. If neither EOCF nor NWTRF is set, we merely decrement the sector:

```
DECS:       IF END'OF'CYLINDER'FLAG    & THEN IT MUST BE A CONTINUATION.
              THEN BEGIN NEW'TRACK'FLAG  := FALSE;
                   DECS2;    & SEE EXPANSION IN SECTR.
                   END
              ELSE IF NEW'TRACK'FLAG    & THEN IT'S ALSO A CONTINUATION.
                     THEN NEW'TRACK'FLAG  := FALSE
                     ELSE BEGIN SECTOR  := SECTOR - 1;
                          IF SECTOR < 0 THEN SECTOR  := MAX'SECTOR;
                          END OF NORMAL DECREMENT;
            END OF DECS;
```

For an INIT, WFS, or RFS command, we don't verify a sector preceding transfer, but we must still call DECS and INCS in case either EOCF or NWTRF is set, so as to treat continuations properly.

Note that a SEEK, ADDRESS RECORD, or RECALIBRATE command resets both EOCF and NWTRF (because the drive and 13037 have been re-addressed). Also, an error during a sector is detected and handled before INCHS is called; therefore neither EOCF nor NWTRF will be set.

CONTENTS

Page

PROGRAMMING INFORMATION

COMMAND WORDS

To initiate an operation, the controller accepts a 16-bit command word from the CPU interface, decodes, and executes the operation specified. The command may pertain to the controller only, a selected disc drive only, or both. Some commands require additional information from the interface for proper execution. Each command is discussed separately in the following paragraphs. The command word format relating to a specific command is shown at the beginning of each paragraph. Bits 8 through 12 of the command are used for the opcode (that is, the octal representation that identifies each command). The U referred to in the command format is the disc drive unit number. (The crosshatched bits indicated in some of the command words are not processed by the controller.) Table 10-1 is a listing of command codes and commands. The commands are listed functionally as control, sense, read, and write groups. They are also discussed in the following paragraphs in functional groups. The timeout function referred to in the discussion is approximately 1.8 seconds, and is initiated when the controller must wait more than that time for a subsequent CPU command.

Table 10-1. Command Codes

CODE (OCTAL)	COMMAND AND GROUP			
	CONTROL	SENSE	READ	WRITE
00			Cold Load Read	
01	Recalibrate			
02	Seek			
03		Request Status		
04		Request Sector Address		
05			Read	
06			Read Full Sector	
07			Verify	
10				Write
11				Write Full Sector
12	Clear			
13				Initialize
14	Address Record			
15		Request Syndrome		
16			Read with Offset	
17	Set File Mask			
22			Read without Verify	
23		Load TIO Register		
24		Request Disc Address		
25	End			
26	Wakeup			

Each disc drive has a "hold" bit associated with it to prevent two or more CPU interfaces from accessing the same disc drive at the same time. Each command to the controller that references a disc drive (except REQUEST STATUS and REQUEST SECTOR ADDRESS) includes a one-bit hold field which is retained by the controller. While a hold bit is set for a particular disc drive, no other CPU interface may access it with a command that could modify the disc drive status. An attempt to access a held disc drive will cause the controller to leave the command pending on the interface until the desired drive becomes available.

COMMAND FORMATS

```
1 1 1 1 1 1
5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0      BIT (2100 NUMBERING)
┌─────────────────────────────────┐
│       C C C C C                 │  ADREC, CLEAR, END,
│                                 │    LTIO, REQAD, RQSYN
│       C C C C C           U U U U │  RQSCT, STAT
│       C C C C C H D S E C T O R │  COLD LOAD
│       C C C C C R T R Y M A S K │  SET FILE MASK
│       C C C C C H         U U U U │  SEEK, RECALIBRATE*
│ S P D C C C C C H         U U U U │  WRITE ADDRESS (INITIALIZE)**
│       C C C C C H         U U U U │  (ALL OTHER COMMANDS)**
└─────────────────────────────────┘
```

WHERE C = FIVE-BIT COMMAND OPCODE:

AS SHOWN	OCTAL	DESCRIPTION
0	0	COLD LOAD READ (BOOT)
400	1	RECALIBRATE
1000	2	SEEK
1400	3	REQUEST STATUS
2000	4	REQUEST SECTOR ADDRESS
2400	5	READ
3000	6	READ FULL SECTOR
3400	7	VERIFY
4000	10	WRITE
4400	11	WRITE FULL SECTOR
5000	12	CLEAR
5400	13	WRITE ADDRESS (INITIALIZE)
6000	14	ADDRESS RECORD
6400	15	REQUEST SYNDROME
7000	16	READ WITH OFFSET
7400	17	SET FILE MASK
10000	20	(NOT AVAILABLE IN 13037 INSTRUCTION SET)
10400	21	(NOT AVAILABLE IN 13037 INSTRUCTION SET)
11000	22	READ WITHOUT VERIFY
11400	23	LOAD TIO REGISTER (3000 SYSTEM)
12000	24	REQUEST DISC ADDRESS
12400	25	END (RETURNS CONTROLLER TO POLL LOOP)
13000	26	WAKEUP

S = FLAG AS SPARE TRACK.
P = FLAG AS PROTECTED TRACK (SUBSEQUENT WRITE COMMAND MAY FAIL).
D = FLAG AS DEFECTIVE TRACK.
HD = HEAD ADDRESS FOR COLD LOAD READ COMMAND.
H = HOLD BIT OF DRIVE
RTRY = DATA TRANSFER RETRY COUNTER (3000 ENVIRONMENTS).
MASK = FILE MASK BITS (SEE DESCRIPTION IN SFMSK PROCESSOR).
U = DISC DRIVE (UNIT) ADDRESSED BY THE COMMAND. ERRORS MAY OCCUR IF U = 10 (OCTAL).
BIT 7 = HOLD BIT. IF SET, RESERVES DRIVE "U" FOR INTERFACE ISSUING THE COMMAND. COMMANDS FROM OTHER INTERFACES TRYING TO ACCESS THIS DRIVE WILL BE DEFERRED UNTIL THE DRIVE IS AVAILABLE.

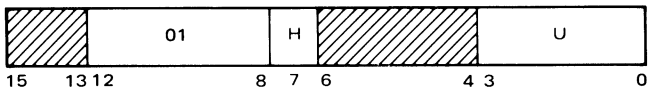
*HOLD BIT (BIT 7) USUALLY SET TO 1
**HOLD BIT (BIT 7) USUALLY SET TO 0

CONTROL COMMANDS

Control commands start operations not involving a transfer of data sectors between the controller and computer. Instead these commands are used to establish certain known conditions within the controller or disc drive. In multi-CPU environments, the ordering and inclusion of certain of these commands warrant special considerations if software is to function as desired; these considerations will be pointed out as they apply to the various commands.

It should be noted that when a command is accepted by the controller, the busy bit of the calling interface is set (and remains set until command completion). In addition, RQSRV is issued after every transfer of control information other than the command word, and after every data transfer. The characteristics of the busy bit and RQSRV issuance apply to all commands of each functional group.

RECALIBRATE



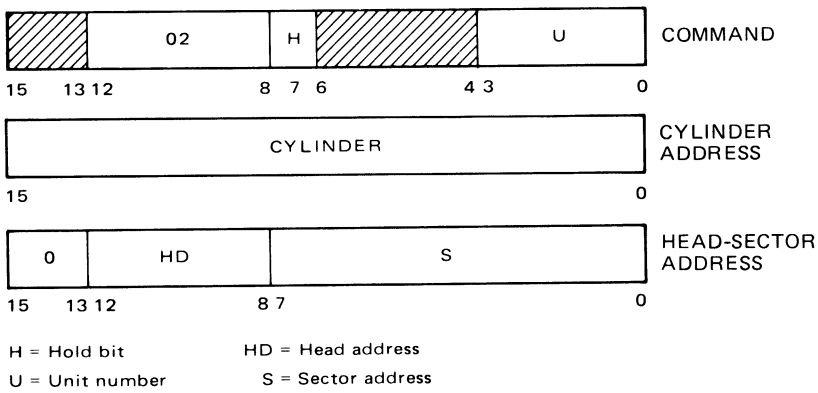
H = Hold bit
U = Unit number

The RECALIBRATE (SEEK HOME) command provides the operating system with a means of restoring the disc drive's heads over a known cylinder (always 0) after that drive is suspected of having lost track of its exact position. To perform this command, only the command word is necessary. The word contains only the opcode, the hold bit, and unit number. If the addressed unit is available and ready, the controller issues a RECALIBRATE command and returns to the poll loop. The disc drive positions its heads over cylinder 0 and clears its "current cylinder address" register.

If the desired drive is found to be "held" by another interface, the controller will leave the RECALIBRATE command pending at the current interface and resume polling. Further, if a unit number greater than 10 is requested, the CPU will be interrupted with a "unit unavailable" error. If the drive unit is available but will not respond, STINT is sent to the interface (and the appropriate Status 2 word is generated). The controller then waits for a new command (or a timeout to occur).

As with the SEEK command, the mechanical process of moving the heads consumes enormous amounts of time relative to execution of CPU instructions; thus the controller will disconnect the current interface and return to polling. In this way, the controller can service other CPU's while the current CPU waits for the desired drive to finish its seek. If another CPU has an available command which can be executed (that is, does not require the use of a drive held by another CPU), the controller will service that command. Eventually the original drive indicates to the controller that its heads are positioned by setting its attention bit. The controller recognizes the attention bit whenever it is polling. (If another CPU has been serviced during the seek, it should eventually issue an END command to cause the controller to resume polling.) The controller reconnects the CPU and the drive. Sometimes, however, the CPU will have no command pending (the CPU, too, may elect to perform other duties while waiting for the drive to complete its mechanical seek); in this case, the controller will determine if it may interrupt the CPU by interrogating the INTOK (Interrupt O.K.) flag. If true, STDFL will be flagged to the CPU; if false, the controller will continue to poll until a command becomes available or an interrupt is allowed.

SEEK



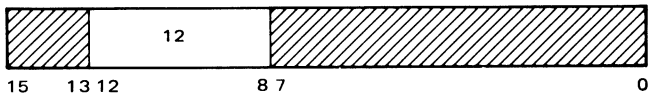
The SEEK command will cause positioning of the disc drive's heads over the desired cylinder; at the same time the controller and drive will store the transmitted cylinder, head, and sector address in their internal registers to allow subsequent verification of the actual address transmitted from the drive prior to any data transfer. This instruction is initiated by issuing the command word followed by two additional words in the order shown. The first contains the cylinder address and the second contains the head and sector address. The controller requests each of these words by transmitting STDFL to the interface. The controller then transmits them to the disc drive.

If the addressed unit is available and ready, the controller accepts the two address words and correspondingly issues the appropriate information to the disc drive to initiate the seek operation. The controller then returns to the poll loop. When the drive completes its seek, an attention bit is set to notify the controller that the drive is ready. The controller, upon recognizing this bit (see corresponding description under the RECALIBRATE command), reconnects the CPU to the drive. If no command is pending, the CPU generated INTOK flag is checked to see if a subsequent interrupt is allowed. If true, STDFL is sent to the CPU and the controller waits for the next command. If INTOK is false, the controller will continue to poll until a command becomes available or an interrupt is allowed.

If the desired drive is found to be "held" by another interface, the controller will leave the SEEK command pending at the current interface and resume polling (the two address words will not be requested at this time). Further, if a unit number greater than 10 is requested, the CPU will be interrupted with a "unit unavailable" error. If the drive unit is available but not ready, STINT is sent to the interface after acceptance of the address words. The controller then waits for a new command (or a timeout to occur).

Note that the target head, sector, and cylinder address are retained in the controller for future data operations, but will be overwritten by a subsequent ADDRESS RECORD or SEEK command. Since the controller polls other interfaces before reporting a seek completion, the programmer should issue an ADDRESS RECORD command immediately preceding a disc drive transfer to refresh the controller's memory as to the current position of that drive.

CLEAR

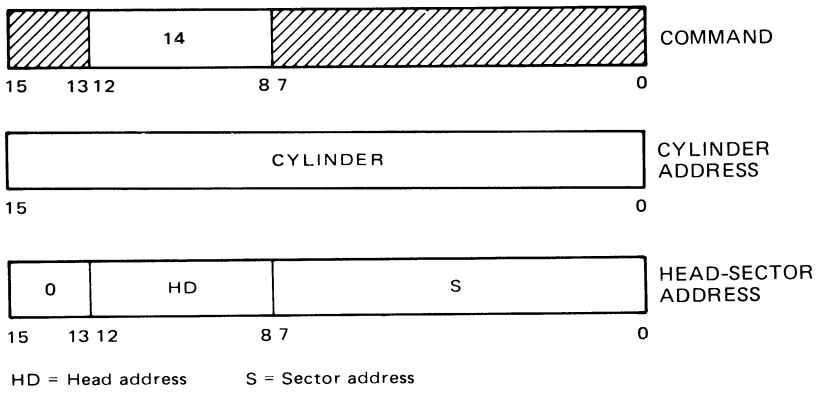


Upon receipt, the controller will issue a CLEAR to all drives, clear status, clear any clock or head offset, turn off the timer, clear the interface busy bit, disconnect all disc drives, transmit Set Data Flag (STDFL) to the interface, and wait for the next command (or a timeout). A unit number is not required.

Because the CLEAR command issues a clear to all disc drives simultaneously (regardless as to whether they are selected), this command should be used with caution in multi-CPU applications.

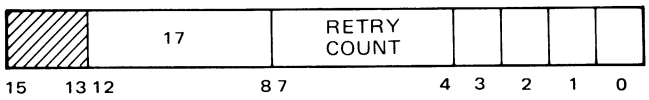
Note that the software clear command differs from the "hard clear" sent to the controller by the interface. The latter unconditionally resets the controller to its power-on state (i.e., all CPU/drive mapping is reset, polling returns to its initial state, current command processing is lost, etc.). For this reason, only one CPU interface should be enabled to issue a "hard clear", such that conflicts will not arise where more than one CPU can unconditionally reset the controller.

ADDRESS RECORD



This command is used to set a logical address in the controller without transmitting it to the disc drive. The issuance of the command word is followed by two words in the order shown. The first contains the cylinder address and the second contains the head and sector address. The controller requests each of these words by transmitting STDFL to the interface. Upon receipt, the controller transmits Request Service (RQSRV) to the interface and waits until either a command is received or a timeout occurs. At completion of this command, the controller waits for a new command (or a timeout). If used in a multiple computer environment, an ADDRESS RECORD should follow each SEEK or RECALIBRATE command, or should precede a data transfer operation if the SEEK or RECALIBRATE is not used. Single CPU environments should take this same precaution if more than one SEEK or RECALIBRATE command is issued without intervening data transfers. Note that reading or writing on unintended portions of the disc can occur if such precautions are not taken.

SET FILE MASK



Bit 0, when set, allows the controller to automatically seek to the next higher or lower cylinder address (depending on bit 3) when a data operation continues through the end of the current logical cylinder. The new track is subjected to disc address and track status verification before any further operation is performed. When bit 0 is clear, the option is prohibited, and any data operation which exceeds the end of the current cylinder is aborted with an end-of-cylinder error status.

Bit 1, when set, indicates that the controller is operating in the cylinder mode. In this mode, the first head address is 0 (the top-most head) and head switching occurs automatically following the last data sector of each track. The head address increases incrementally from 0 to the last head address of the particular disc drive. As each new head is switched in, the verification described in the previous paragraph is performed. The last data sector of the last head address of the current track address is the end of the logical cylinder.

When bit 1 is clear, the controller is operating in the surface mode, where the head address never changes. The last sector of the current track address is the end of cylinder. If bit 0 is set (incremental/decremental seek allowed), a data operation which exceeds the end of the cylinder will continue at the next higher/lower cylinder address (subject to address and track status verification).

When bit 2 is set, the controller is allowed to automatically seek to previously initialized spare tracks when a defective track (that is, one which has been initialized defective) is encountered. (The entire spare track algorithm will be described later.) If bit 2 is clear, the automatic sparing will not occur and a defective track error status will be reported to the interface.

Bit 3 operates in conjunction with bit 0 and determines whether an automatic end-of-cylinder seek will be to the next higher or lower track address. If bit 3 is 0, the next higher track is selected; if it is 1, the next lower track is selected. Note: If bit 0 is 0, bit 3 is not examined.

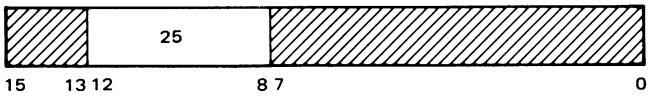
Bits 4 through 7 indicate how many retries are allowed.

The mask sets the mode of operation for the controller. The controller transfers bits 0 – 3 into its "mask" register which is used to control the action taken while transferring data. The entire word is transmitted to the interface via a Set Retry Counter (SRTRY) order. For computers with a data channel separate from the CPU, the retry counter allows the channel program to retry a data transfer in case of an error. The programmer sets the counter with bits 7 through 14 of this command. After the specified number of retries, the interface interrupts the CPU. After sending SRTRY, the controller transmits Set Data Flag (STDFL) to the interface. The controller takes no further action until a command is received or a timeout occurs. At power up, the mask is set to 0 (no-automatic-seek, surface mode, and no sparing).

Because the file mask affects both sparing operations and data transfers that require crossing over the last sector of a particular cylinder, the condition of the file mask always must be known by the operating system. A timeout error will reset the file mask to the "no automatic seek, surface mode, no sparing" state. Also, in multiple CPU environments, one

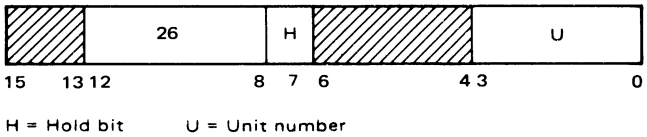
CPU is not given any direct indication that another CPU has changed the file mask (or timed out, since the effect of a timeout error is to disconnect the offending interface and return to polling). Thus it is suggested that directly prior to any data transfer, the SET FILE MASK instruction (as well as ADDRESS RECORD, mentioned previously) be issued to put the file mask in the desired state.

END



This command is used to avoid a timeout if no other command is expected to be outputted from an interface and the controller is waiting for a command from that interface. When the controller accepts this command from an interface, no action is taken except to resume polling. This command should follow any string of commands when they have been completed.

WAKEUP



This command checks if the specified unit is available (attached to current interface or hold bit clear). If not, the command is left pending on the interface and the controller resumes polling of other interfaces.

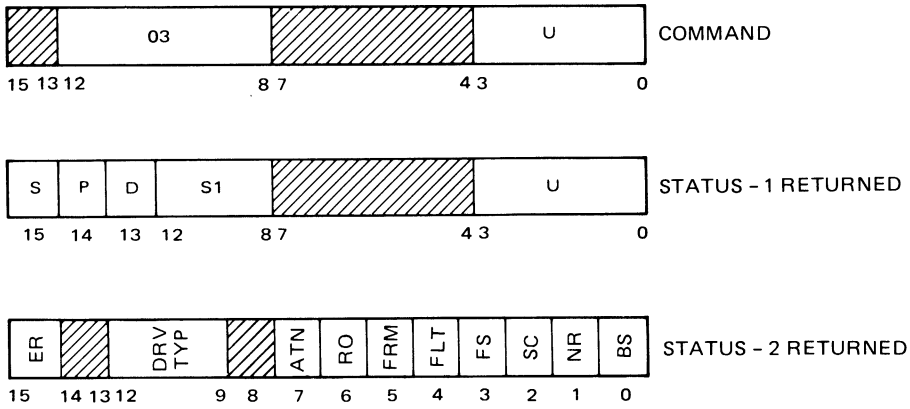
If the unit is available, the status register is set to indicate "unit available" and STDFL is transmitted to the interface. The controller then waits until a command is received or until timeout occurs.

Because the WAKEUP routine is incorporated in all commands that access a disc drive, this command is no longer necessary in multi-CPU systems (if a different CPU issues a command that attempts to access a "held" disc drive, the command will remain pending at the interface until that drive becomes available). It is suggested that if the WAKEUP command is used, a REQUEST STATUS command should follow to determine the condition of the accessed drive prior to its use.

SENSE COMMANDS

Sense commands are included as a means of providing feedback to the operating system as to the condition of the controller, disc drive, or various operations. Included in this group are diagnostic and data recovery commands that expand the versatility of the 13037 controller and enhance the reliability of data transfers. Returned information is accompanied by a STDFL, telling the interface that data has been returned by the controller.

REQUEST STATUS



U — Unit number to which status on last operation applies; if S1 is 37 (drive attention), U is number of interrupting drive. If no unit number is appropriate, U will be zero.

S1 — Encoded termination status (octal).

After receipt of the command, the controller returns two status words to the interface. The first (Status-1) contains information relating to the last operation performed, and the second (Status-2) contains information pertaining to the disc drive addressed in the command word. Any hold bit is ignored, and the First Status bit in Status-2 is cleared by this command after it has been returned. The controller then clears Status-1 and waits for a command from the same interface or a timeout to occur. Note that the unit field of the command word determines the drive that the returned Status-2 word will describe, while the unit field of the Status-1 word relates to that associated with the last operation performed.

Note: The Status-1 word is also transmitted to the interface at the completion of all commands or at an error interrupt (the WRTIO function bus command is used).

The various encoded termination status values in the S1 field are explained below.

00 — NORMAL COMPLETION. This status is transmitted in one of two situations:

a. When command has been fully executed without error.

b. At completion of a REQUEST STATUS command whenever the command immediately follows another REQUEST STATUS command or when it is the first command issued after interface is connected to controller during a polling sequence. For the latter case, the U field will be zero.

01 — ILLEGAL OPCODE. A command word has been received by the controller of which bits 12-8 contain a command code which is not one of controller's command set.

02 — UNIT AVAILABLE. Controller transmits this status after interface has put out a WAKEUP command for a specific drive and that drive has become available.

07 — CYLINDER COMPARE ERROR. During verification of address of sector previous to first sector to be read from or written to, the contents of cylinder address field of that sector do not match contents of controller's cylinder address register. This status is transmitted only after the sequence of events listed below.

a. Addresses do not compare as described above.

b. Controller generates a seek to address in its cylinder address register and head sector address register.

c. Controller again attempts to verify a sector.

d. Addresses still do not compare.

e. The S bit is not set at new track address.

When this status is received, the system should issue a RECALIBRATE command and then retry data transfer sequence.

10 — UNCORRECTABLE DATA ERROR. This status is generated by the error correction circuits and is transmitted in one of three cases:

a. Immediately following a data transfer (or VERIFY) command if error is uncorrectable.

b. In response to a REQUEST SYNDROME command whenever a Possibly Correctable Data Error has proved uncorrectable.

c. During verification of address of sector previous to first sector to be read from or written to, controller is unable to read (verify) any of 16 consecutive sectors without error.

11 — HEAD-SECTOR COMPARE ERROR. Similar to Cylinder Compare Error, including controller's recovery attempt sequence described for that status, except that head and/or sector address field of disc sector does not compare with corresponding field in controller's head sector address register. The system need not issue a RECALIBRATE command when this status is received.

12 — I/O PROGRAM ERROR. Systems containing a programmable data channel separate from CPU may have their interface detect abnormal channel operations and notify controller. At that time, controller will interrupt CPU with this status. An example of such an error might be an inconsistent direction of data transfer (a read command has been transmitted to controller, but channel has been programmed to write).

14 — END OF CYLINDER. A multiple-sector data transfer must continue beyond end-of-logical-cylinder, but file mask will not allow controller to automatically seek to next logical cylinder and continue.

16 — OVERRUN. Detected by interface (read) or controller (write) whenever instantaneous data rate of controller exceeds that of CPU-interface combination. The overrun is reported at end of sector in which it occurred. The contents of that sector, either on disc (write) or in I/O buffer (read) should be considered invalid.

Note: The controller always transfers complete sectors between itself and the disc. If the CPU or data channel wishes to transfer less than a complete sector, it must notify interface (or controller) when the transfer is complete so that subsequent controller requests for data transfer do not cause an Overrun error.

17 — POSSIBLY CORRECTABLE DATA ERROR. This status is generated by the error correction circuits and is transmitted in one of two cases:

a. Immediately following a data transfer (or VERIFY) command if error is possibly correctable.

b. In response to a REQUEST SYNDROME command if error is in fact correctable. In this case, proceed as described in REQUEST SYNDROME command.

20 — ILLEGAL ACCESS TO SPARE TRACK. The same conditions and sequence of events described for a Cylinder Compare Error or Head-Sector Compare Error have occurred, except that S bit is set at new track address. This error usually results from trying to directly access (via a SEEK command) a spare track in active use. The addresses will not compare because of the way in which spare tracks are set up and this status merely differentiates between this situation and other address errors.

21 — DEFECTIVE TRACK. This status is transmitted in one of two situations:

a. During verification of track status of sector previous to first sector to be read from or written to, the D bit is found to be set but File Mask will not allow automatic seeking to a spare track.

b. In the same situation, the File Mask does allow automatic seeking but the address field contains the defective track address (in other words, the D bit was set without assigning a spare track). See the section on SPARE TRACKS and ADDRESS VERIFICATION.

22 — ACCESS NOT READY DURING DATA OPERATION. While in process of transferring data to or from disc, the track center detector in drive detected head motion. The transfer should be retried.

Note: If heads moved during a write operation, the drive will fault and must be manually reset.

23 — STATUS-2 ERROR. The controller is unable to complete a command due to some condition in disc drive. The Status-2 word may be examined for reason. Examples of Status-2 Errors are:

a. An Initialize command, but Format switch is off or Read Only switch is on.

b. A command is issued to a drive which is Not Ready (heads unloaded) or for which a Drive Fault has occurred.

26 — ATTEMPT TO WRITE ON PROTECTED TRACK. During verification of track status of sector previous to first sector to be written to using a Write command, the P bit is found to be set and the Format switch is off.

27 — UNIT UNAVAILABLE. This status is returned whenever the U field of the command word is greater than 12 (octal).

Note: The interface busy bit is false whenever this status is returned.

37 — DRIVE ATTENTION. Controller generates an interrupt (issues STINT) to interface which last accessed drive that is requesting attention (or interface 0 if this is first attention after power-on or hard clear) whenever:

a. Drive is requesting attention.

b. Interface does not have a subsequent command pending in its command buffer excepting Wakeup, which is ignored here.

c. Interface flag INTOK (Interrupt O.K.) is set, thereby allowing attention interrupts.

Briefly, conditions causing a drive to request attention are:

a. Seek completion.

b. Drive becomes ready (heads load).

c. Drive becomes not ready (heads unload).

d. Seek check.

e. Drive Fault.

S — Spare track

P — Protected track

D — Defective track

S2 — Unit status

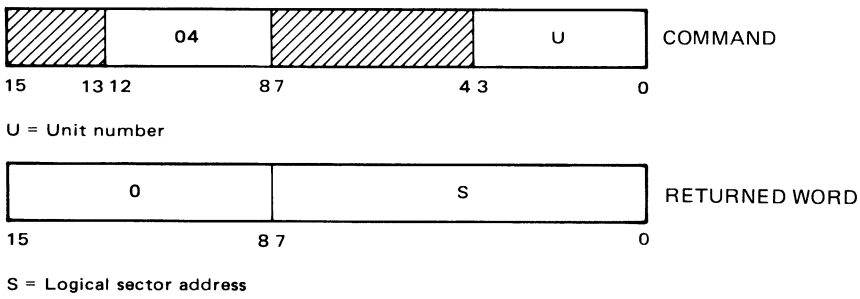
Bit

0* — Drive busy
1* — Drive not ready
2* — Seek check
3 — First status
4* — Fault
5 — Format
6 — Protected (drive cannot write)
7 — Attention
8 — (Reserved)
9-12 — Encoded drive type (used by controller to determine last available head and sector) as described below.
15 — Status-2 error (true if any bit marked * is true)

Drive types:

0 – 7906 (4 heads, 48 sectors)
1 – 7920 (5 heads, 48 sectors)
2 – 7905 (3 heads, 48 sectors)
3 – 7925 (9 heads, 63 sectors)

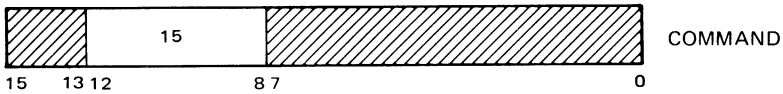
REQUEST SECTOR ADDRESS



After receipt of the command, the controller returns the logical address of the sector currently passing under the heads of the specified unit. The existing hold bit is not checked or altered. At completion, the controller waits for the next command (or a timeout).

Besides use in wraparound read operations, this command also can be used to allow the CPU to perform other operations while waiting for the desired sector to pass underneath the accessed disc drive's heads (348 microseconds are required for each sector to pass by the heads, allowing as much as 16.3 milliseconds (21.9 milliseconds for the 7925 disc drive)for the CPU to do other operations). However, users are cautioned in that improper use of this command can create timing conflicts that unnecessarily delay access to the desired sector for subsequent data transfer.

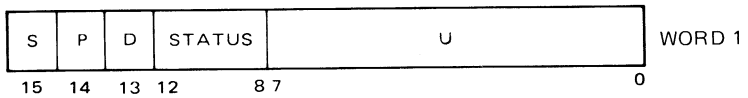
REQUEST SYNDROME



To provide greater reliability in data transfer through the 13037 controller, an error correction process (involving both hardware and firmware) is included that can be invoked by the REQUEST SYNDROME command. This command is used when errors have been detected by the controller after transferring data from the disc drive to the computer. The command may be issued after a READ, COLD LOAD READ, READ WITH OFFSET, VERIFY, or READ WITHOUT VERIFY command which terminates with a Status-1 word indicating "Possibly Correctable". The software correction routine should not try to correct the preamble or postamble since these areas have not been transmitted to the data buffer in the computer (if used following a VERIFY command, no attempt should be made to correct data, since none has been transmitted).

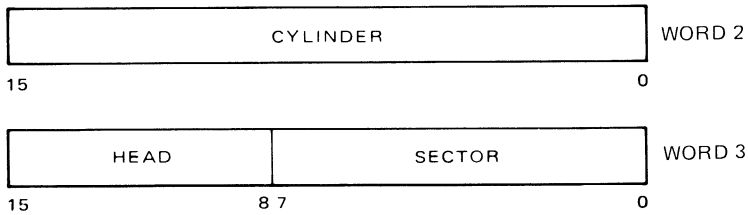
This command should only be issued if the Status-1 field immediately following a data transfer (or VERIFY command) contains Possibly Correctable Error status. False results will be obtained if this command is issued at any other time (including after a previous REQUEST SYNDROME command). Figure 10-2 outlines a suggested use of the REQUEST SYNDROME command (the retry counter indicated on the flowchart can be that of the interface, or a scratch pad register within the operating system).

This error correction routine will provide all of the information needed to correct a "correctable" data error. However, the operating system must use the returned displacement to align the correction mask properly with the appropriate section of the erroneous data. Seven words are returned, with the following significance:

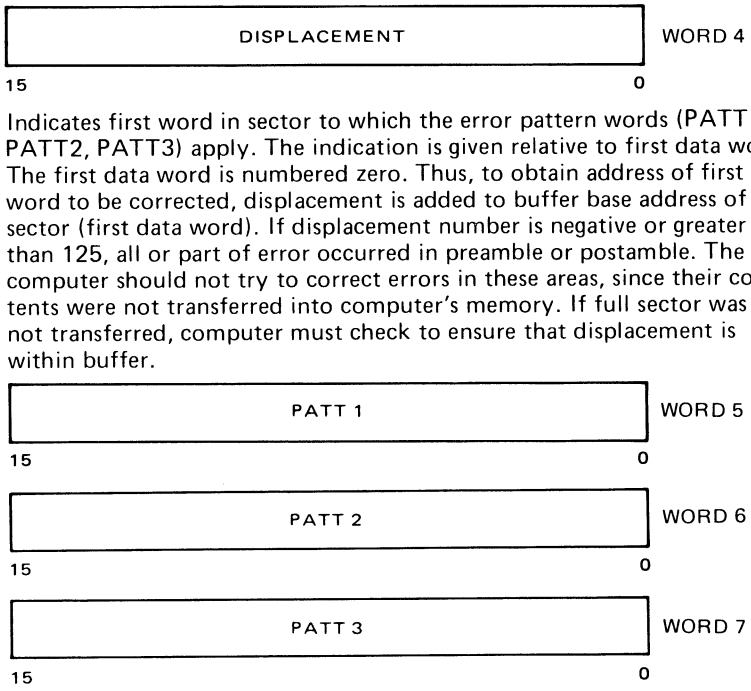


S = Spare track
P = Protected track
D = Defective track
U = Unit number

Status – May be "Correctable" or "Uncorrectable" (Ø17 or Ø1Ø octal) at this point. If uncorrectable, ignore displacement and pattern words. If correctable (Ø17), correction data is available in displacement and pattern words.



Cylinder, Head, Sector – The logical address of sector in which error occurred.



Displacement – Indicates first word in sector to which the error pattern words (PATT1, PATT2, PATT3) apply. The indication is given relative to first data word. The first data word is numbered zero. Thus, to obtain address of first word to be corrected, displacement is added to buffer base address of sector (first data word). If displacement number is negative or greater than 125, all or part of error occurred in preamble or postamble. The computer should not try to correct errors in these areas, since their contents were not transferred into computer's memory. If full sector was not transferred, computer must check to ensure that displacement is within buffer.

Three words, PATT1, PATT2, and PATT3, are generated by the controller for error correction. These words are "exclusive-or'ed" with the data word errors, beginning with the word indicated by the displacement number. Be sure that the displacement refers to words actually present in the data buffer (see above description entitled "Displacement"). Performance of the "exclusive-or" function of the computer corrects the errors. Figure 10-3 outlines a suggested method of error correction.

The command will complete in less than 700 microseconds. When the process is complete, the controller addresses the next logical sector and waits for a command or until timeout occurs. If the transfer was not complete, the interface may issue a READ without an intervening SEEK or ADDRESS RECORD to continue the transfer. Polling is not resumed when REQUEST SYNDROME is completed; after command completion, the controller waits for a new command (or a timeout).

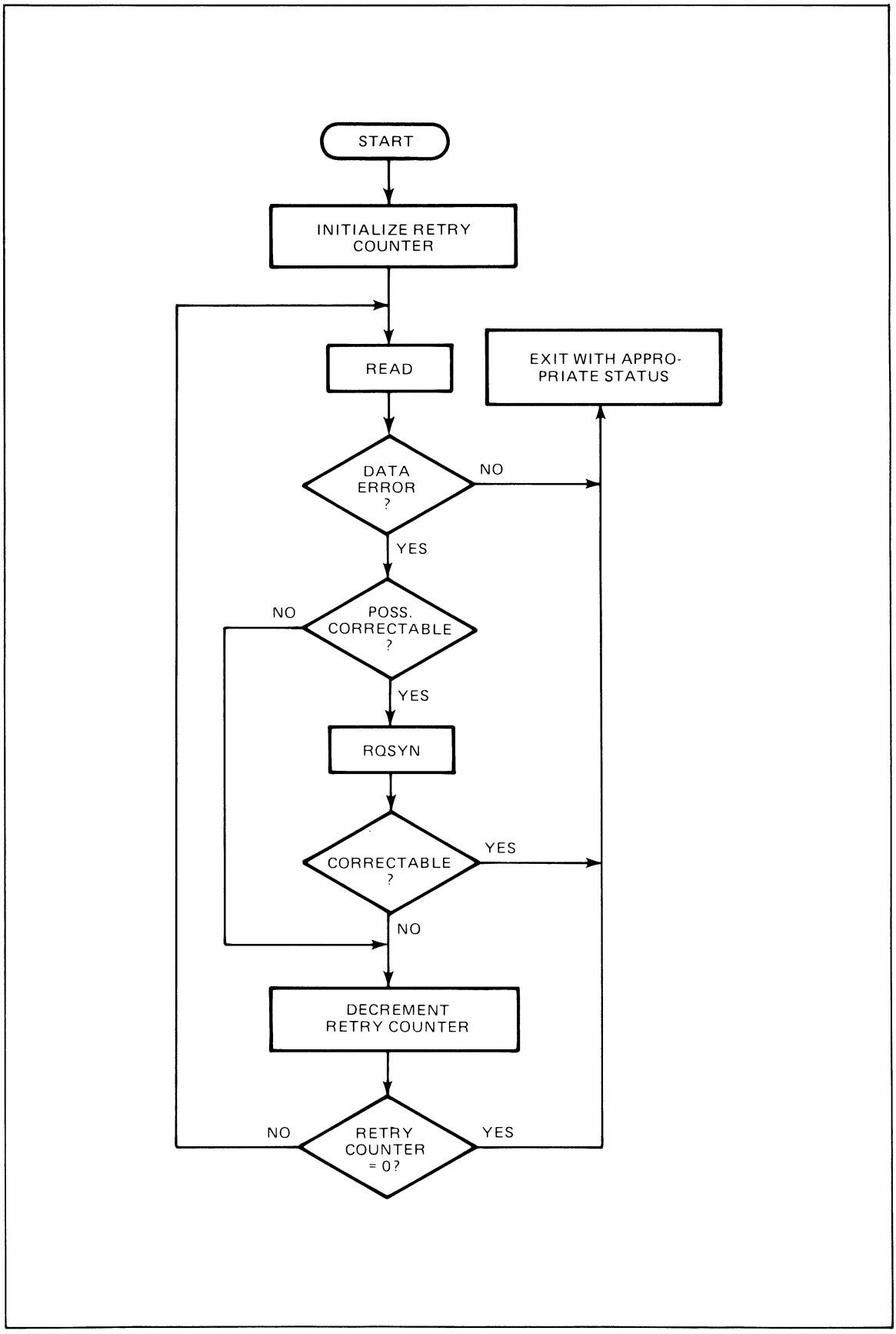


Figure 10-2. Suggested Request Syndrome Routine

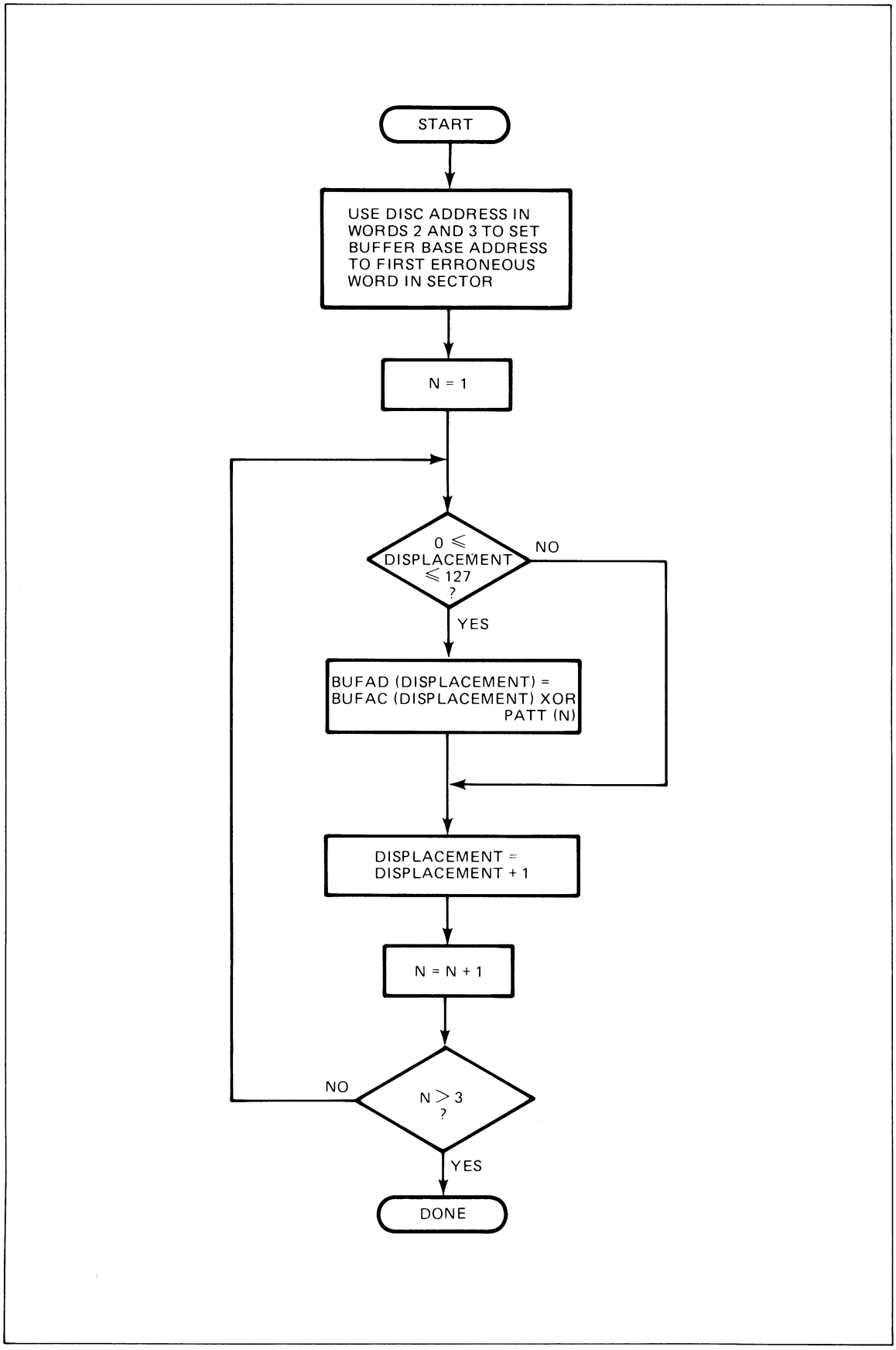
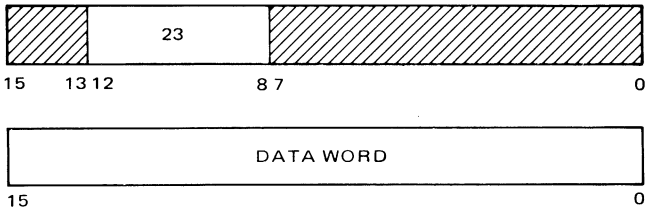


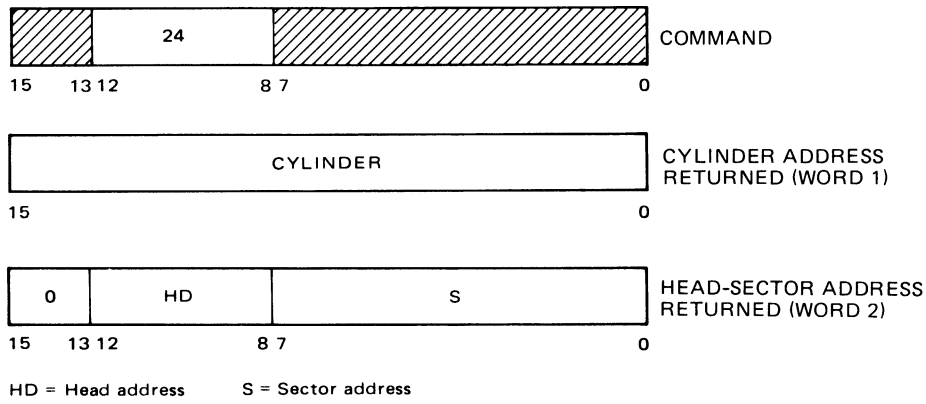
Figure 10-3. Suggested Error Correction Routine

LOAD TIO REGISTER



Included as another diagnostic tool, the LOAD TIO REGISTER command allows the CPU to accept information and write it into the interface's TIO register much as the Status-1 word is automatically written into that register after every controller operation. The controller requests a data word by issuing STDFL. This data word is then written into the TIO (status) register on the interface by a Write Interface Status Register (WRTIO) order. The controller then transmits Request Service (RQSRV) to the interface and waits for a new command (or timeout).

REQUEST DISC ADDRESS



The controller returns two words in the order and format of the address words passed in a SEEK command. The command may be used at any time, but is most useful following a multiple-sector I/O or VERIFY command which aborted with an error. This command allows the operating system to determine where the abort occurred. After completion, the controller then transmits Request Service (RQSRV) to the interface and waits for a command or until timeout occurs.

The returned address is that of the current disc address stored in the controller. This will be the address of the current sector if a data error occurred. or the address of the next logical sector if no data error occurred.

Note that if either word is not taken by the CPU within 1.8 seconds after the command is accepted, a timeout will occur.

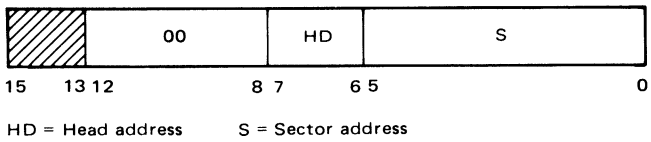
READ COMMANDS

Read commands transfer information from a disc drive to the computer. On all read commands (except READ FULL SECTOR), the controller examines correction and cyclic code words to check the validity of each record area. Previous to all operations (except READ FULL SECTOR and READ WITHOUT VERIFY), the sector prior to that in which the transfer begins is checked to see if the written address (on the disc) agrees with that in the controller register. On multiple sector data transfers, if a track boundary is crossed another address verification takes place. Such address verification is used by the controller to determine if sparing operations need to be performed. Sparing will occur, if enabled and necessary, only for COLD LOAD READ, VERIFY, READ, READ WITH OFFSET, and (only when a track boundary is crossed) READ WITHOUT VERIFY commands.

Because the controller always transfers complete sectors (128 words normally, 138 words for READ FULL SECTOR), any error indications will occur at the end of that sector. Note that EOD can be asserted at any time during a sector if only part of that sector is to be read.

Normal read transfers result in STDFL and RQSRV being sent to the interface. Data errors cause DVEND and RQSRV to be sent to the interface. STINT is sent to the interface on any other error condition. Because some interfaces combine DVEND, STDFL, and/or STINT, the operating system should inspect status on apparent completion to check the state of the preceding data transfer.

COLD LOAD READ



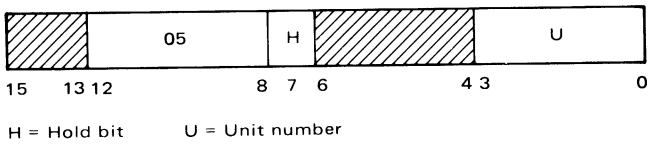
The controller first checks that unit 0 is available (hold bit clear) or attached to the current interface. If not, it resumes polling until unit 0 is available, then executes the command.

After issuing a SEEK to unit 0, cylinder 0 and the head and sector specified in the command word, and waiting for seek completion, the controller sets the FILE MASK to Sparing Enabled, Incremental Seek Not Allowed, and Surface Mode. The controller also transmits a 0 retry count to the interface and then begins reading, starting with the sector and head addressed in the command word.

The controller continues to transfer 128 words per sector until the interface sets End of Data (EOD) or the end-of-logical-cylinder is reached. The last sector does not have to be completely transferred. At the end of each sector, checks are made for overrun (detected by the interface) and data errors. If any errors are detected, the operation will be aborted regardless of End of Data (EOD). Sparing will occur if it is necessary.

On normal completion, Set Data Flag (STDFL) and Request Service (RQSRV) are sent to the interface. On an overrun or data error, Device End (DVEND) and Request Service (RQSRV) are transmitted. Any other error causes the controller to transmit STINT to the interface. The controller then waits for a command from the same interface or a timeout to occur.

READ



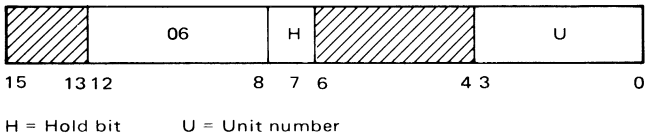
The controller first checks that the specified unit is available (if the hold bit is set, the controller will resume polling, leaving the command pending at the interface). The following description assumes the unit is available.

After sparing (if necessary and enabled) and waiting for seek completion, the controller begins reading from (1) the sector last addressed by a SEEK command; (2) the sector last addressed by an ADDRESS RECORD command; or (3) the sector following the last one transferred, whichever occurred most recently.

The controller continues to transfer 128 words per sector until the interface sets End of Data (EOD). The last sector does not have to be completely transferred. At the end of each sector, checks are made for overrun (detected by the interface) and data errors. If any errors are detected, the operation will be aborted regardless of End of Data (EOD). Sparing will occur at the start of any track if necessary and enabled.

On normal completion, Set Data Flag (STDFL) and Request Service (RQSRV) are sent to the interface. On an overrun or data error, Device End (DVEND) and Request Service (RQSRV) are transmitted. Any other error causes the controller to transmit STINT to the interface. The controller then waits for a command from the same interface or a timeout to occur.

READ FULL SECTOR



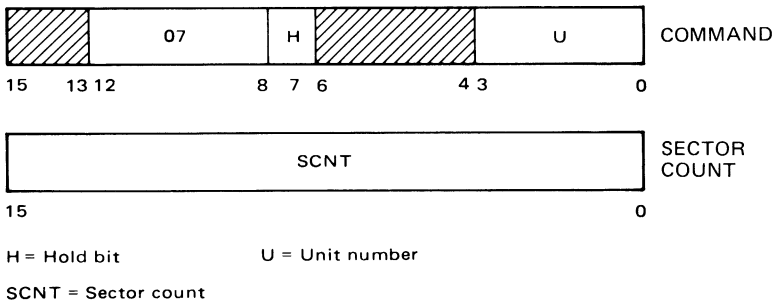
The controller first checks that the specified unit is available (if the hold bit is set, the controller will resume polling, leaving the command pending at the interface). The following description assumes the unit is available.

After waiting for seek completion (without sparing), the controller begins reading from (1) the cylinder, head and sector last addressed by a SEEK command, (2) the cylinder and head last addressed by a SEEK command and the sector last addressed by an ADDRESS RECORD command, or (3) the sector following the last sector transferred, whichever occurred most recently.

The controller then transfers 138 words per sector (sync word, cylinder address, head-sector address, 128 data words, CRC word, and 6 ECC words) to the computer until the interface sets End of Data (EOD). The last sector does not have to be completely transferred. At the end of each sector, a check is made for overrun (detected by the interface). If an error is detected, the operation will be aborted regardless of End of Data (EOD).

On normal completion, Set Data Flag (STDFL) and Request Service (RQSRV) are sent to the interface. On an overrun error, Device End (DVEND) and Request Service (RQSRV) are sent to the interface. Any other error causes the controller to transmit STINT to the interface. The controller then waits for a command from the same interface or a timeout to occur.

VERIFY



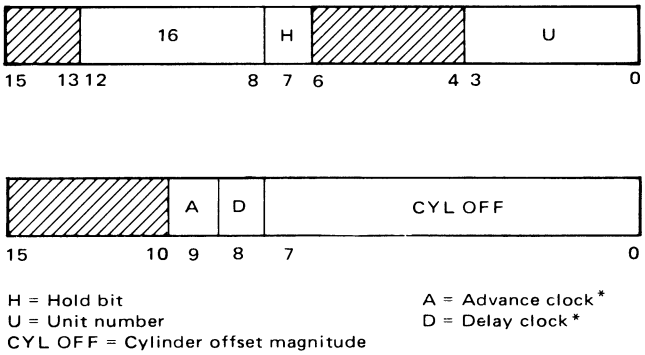
The controller first checks that the specified unit is available (if the hold bit is set, the controller will resume polling, leaving the command pending at the interface). The following description assumes the unit is available.

After acceptance of the command word, the controller requests the sector count (SCNT) by issuing STDFL to the interface. Starting with the currently addressed sector, the controller reads from the disc without passing data to the computer. At the end of each sector, a check is made for data errors. If any errors are detected, the operation will be aborted regardless of sector count word. The controller continues to scan sectors until SCNT has decremented to zero. The SCNT designates how many sectors are to be verified. If SCNT is zero, the interpretation is 65,536 (decimal) sectors.

After sparing (if necessary and enabled) and waiting for seek completion, the controller begins reading (without transmitting to the interface) from (1) the sector last addressed by a SEEK command; (2) the sector last addressed by an ADDRESS RECORD command; or (3) the sector following the sector last transferred, whichever occurred most recently. Sparing will occur at the start of any track if necessary and enabled.

On normal completion, Set Data Flag (STDFL) and Request Service (RQSRV) are sent to the interface. On data error, Device End (DVEND) and Request Service (RQSRV) are transmitted. Any other error causes the controller to transmit STINT to the interface. The controller then waits for a command from the same interface or a timeout to occur.

READ WITH OFFSET



H = Hold bit
U = Unit number
CYL OFF = Cylinder offset magnitude
A = Advance clock*
D = Delay clock*

* 13037B Disc Controllers containing 13037-60028 Device Controller PCA's ignore these bits.

The controller first checks that the specified unit is available (if the hold bit is set, the controller will resume polling, leaving the command pending at the interface). The following description assumes the unit is available.

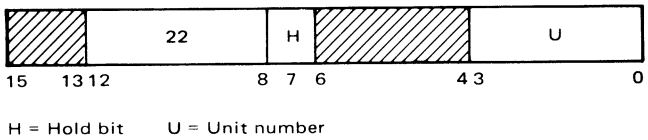
This command performs like a normal read (including termination sequences) except that the heads are moved off of track center by an amount proportional to an offset parameter word requested from the computer before data transfer begins. The controller requests this parameter by sending STDFL to the interface. Upon receipt, RQSRV is sent. Direct memory access (DMA) should not be activated for the data transfer portion of the command until one or both of these signals are received on the interface.

The controller transmits the cylinder offset to the selected disc drive, and may (for 13037 Disc Controllers having serial prefixes before 1815) also advance or delay the separator clock 10 nanoseconds with respect to the data. A cylinder offset of zero is on the track center. Offset parameters in the range of +63 to −63 move the heads off of track center by incremental amounts characteristic of the unit specified. (A negative parameter is in sign magnitude form.) All offsets are removed after normal command completion or error terminations.

This command is usually used only in error recovery situations when normal re-reads and REQUEST SYNDROME commands have failed. Since each offset operation requires a minimum of 1.5 milliseconds to complete, the target sector will have passed under the heads by the time the offset is complete, thus requiring an extra rotation of the disc. A similar amount of time is required at the end of the command to restore the heads to track center.

Spare tracks cannot be read with offset. The process of seeking to the spare track causes the disc drive to cancel the offset, and the controller cannot restore it. A similar situation occurs if a transfer requires an automatic incremental or decremental seek at the end of a logical cylinder.

READ WITHOUT VERIFY



The controller first checks that the specified unit is available (if the hold bit is set, the controller will resume polling, leaving the command pending at the interface). The following description assumes the unit is available.

After waiting for seek completion (without sparing), the controller begins reading from (1) the cylinder, head and sector last addressed by a SEEK command, (2) the cylinder and head last addressed by a SEEK command and the sector last addressed by an ADDRESS RECORD command, or (3) the sector following the last sector transferred, whichever occurred most recently.

This command performs like a normal read, but does not verify the preceding sector. Therefore, no address checking or sparing operations occur unless a track boundary is crossed during the operation. This command is provided for error recovery or, in conjunction with the REQUEST DISC ADDRESS command, wraparound read operations (the latter is discouraged unless extremely fast CPU's are being used).

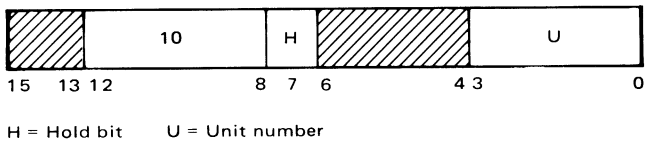
WRITE COMMANDS

Write commands transfer information from the computer to the controller for subsequent writing onto the disc. Except for the WRITE FULL SECTOR command, the controller will append the preamble and postamble to the data field (the computer must supply 138 words of information — the preamble, 128 data words, and the postamble — during WRITE FULL SECTOR commands; the 12 word zero sync field, however, is supplied by the controller).

128 words per sector (138 words per sector for WRITE FULL SECTOR commands) are transferred until the interface sets EOD or until the controller is unable to continue the transfer due to a drive error, interface error, or condition set in the file mask. Because the controller always processes complete sectors, normal completion or any errors that occur will be reported at the end of the current sector. In the event that EOD is asserted prior to the end of a sector, the controller repeatedly will write the last transmitted word into the remainder of the sector data field (and through the postamble during WRITE FULL SECTOR commands).

STDFL and RQSRV are sent to the interface on normal completion of write commands. Overrun errors cause DVEND and RQSRV to be sent. Other errors will cause STINT to be sent. In all cases, the controller waits for a subsequent command or a timeout to occur.

WRITE



The controller accepts data from the computer and writes the addressed sector (preamble, data, and postamble) after verifying the previous sector. The command will not be executed if the surface is protected. If the P-bit is set, the Format switch must also be on.

The controller first checks that the specified unit is available. If the hold bit is set, the command is left pending on the interface and the controller resumes polling. The following description assumes the unit is available.

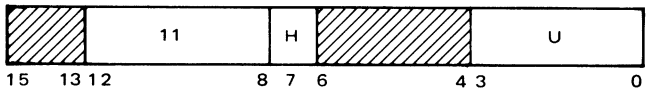
After sparing (if necessary and enabled) and waiting for seek completion, the controller begins writing on (1) the sector last addressed by a SEEK command; (2) the sector last addressed by an ADDRESS RECORD command; or (3) the sector following the last sector transferred, whichever occurred most recently.

On all data transfers out of the computer, the controller will request 128 words of data. If the computer transfers less, the controller will repeatedly write the last transmitted word into the remainder of the sector data field. If the controller fills the sector in this manner and then finds that the computer has more data available, it will flag an overrun error and terminate the operation. (Overrun is detected by the controller.) Normally, the controller continues to transfer 128 data words per sector until the interface sets End of Data (EOD). At the end of each sector, a check is made for overrun or a drive error. If an error is detected, the operation will be aborted regardless of End of Data (EOD). Sparing will occur at the end of any track if necessary and enabled.

On normal completion, Set Data Flag (STDFL) and Request Service (RQSRV) are sent to the interface. On an overrun error, Device End (DVEND) and Request Service (RQSRV) are transmitted. Any other error causes the controller to transmit STINT to the interface. The controller then waits for a command from the same interface or a timeout to occur.

The operation will be aborted whenever a track is encountered which has been flagged protected (unless the Format switch is on), or if the surface is protected by the disc drive PROTECT (READ ONLY) switch(es).

WRITE FULL SECTOR



H = Hold bit U = Unit number

This command is like a write except that the computer must pass the sync word, address words, and postamble to the controller in addition to the data field.

The controller first checks that the specified unit is available. If the hold bit is set, the command is left pending at the interface and the controller resumes polling. The following description assumes the unit is available.

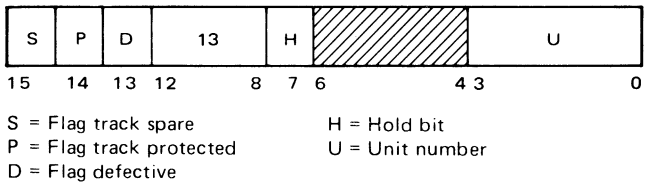
After waiting for seek completion (without sparing), the controller begins writing on (1) the cylinder, head and sector last addressed by a SEEK command; (2) the cylinder and head last addressed by a SEEK command and the sector last addressed by an ADDRESS RECORD command; or (3) the sector following the last sector transferred, whichever occurred most recently.

The controller will request 138 words per sector (sync word, cylinder address, head-sector address, 128 data words, CRC word, and 6 ECC words). If the computer transfers less, the controller will repeatedly write the last transmitted word into the remainder of the sector. If the controller fills the sector in this manner and then finds that the computer has more data available, it will flag an overrun error and terminate the operation. (Overrun is detected by the controller.) Normally, the controller continues to transfer 138 words per sector until the interface sets End of Data (EOD). At the end of each sector, a check is made for overrun or a drive error. If an error is detected, the operation will be aborted regardless of End of Data (EOD).

On normal completion, set Data Flag (STDFL) and Request Service (RQSRV) are sent to the interface. On an overrun error, Device End (DVEND) and Request Service (RQSRV) are transmitted. Any other error causes the controller to transmit STINT to the interface. The controller then waits for a command from the same interface or a timeout to occur.

The operation will be aborted if the FORMAT switch on the disc drive is off, or if the surface is protected by the disc drive PROTECT (READ ONLY) switch(es).

This command is intended only as a diagnostic tool. In any case, it is strongly recommended that any multiple sector transfers not cross a track boundary since all address verification and track status checks are off.



On receipt of the command word, the controller checks that the surface is not protected by the disc drive PROTECT (READ ONLY) switch(es) and that the FORMAT switch on the drive is set to ON before executing the command. The controller then begins requesting data from the computer and writes 128 data words on the addressed sector without verifying the preceding sector. Data transfer continues until the interface sets End of Data (EOD).

The controller first checks that the specified unit is available. If the hold bit is set, the command is left pending at the interface and the controller resumes polling. The following description assumes the unit is available.

After waiting for seek completion (without sparing), the controller begins writing on (1) the cylinder, head and sector last addressed by a SEEK command; (2) the cylinder and head last addressed by a SEEK command and the sector last addressed by an ADDRESS RECORD command; or (3) the sector following the last sector transferred, whichever occurred most recently.

On all data transfers out of the computer, the controller will request 128 words of data. If the computer transfers less, the controller will repeatedly write the last transmitted word into the remainder of the sector data field. If the controller fills the sector in this manner and then finds that the computer has more data available, it will flag an overrun error and terminate the operation. (Overrun is detected by the controller.) Normally, the controller continues to transfer 128 data words per sector until the interface sets End of Data (EOD). At the end of each sector, a check is made for overrun or a drive error. If an error is detected, the operation will be aborted regardless of End of Data (EOD).

On normal completion, Set Data Flag (STDFL) and Request Service (RQSRV) are sent to the interface. On an overrun error, Device End (DVEND) and Request Service (RQSRV) are transmitted. Any other error causes the controller to transmit STINT to the interface. The controller then waits for a command from the same interface or a timeout to occur.

Bits 13, 14, and 15 of the command word are used to flag the track defective, protected, or spare. Subsequently, when a data transfer (or VERIFY) operation is attempted on a track so marked, the status of the track last accessed by that operation will appear in bits 13-15 of STATUS-1 word. It is the responsibility of the system to ensure that all sectors on a track have the same status.

If the D bit is set, the track is considered defective and sparing will occur if required and enabled. The only drive operations allowed on a track so marked are READ FULL SECTOR, READ WITHOUT VERIFY, INITIALIZE, and WRITE FULL SECTOR. The latter two will not preserve track status unless specifically directed to do so.

If the P bit is set, a write operation will only be allowed on that track if the FORMAT switch is on; track status is preserved. INITIALIZE or WRITE FULL SECTOR will also be allowed if the FORMAT switch is on, but will not preserve track status unless specifically directed to do so.

If the S bit is set on a track, a spare track is in active use. Data operations including VERIFY but excepting INITIALIZE, WRITE FULL SECTOR, READ FULL SECTOR, and READ WITHOUT VERIFY will not be allowed on a track so marked unless access is made through its corresponding defective track. INITIALIZE and WRITE FULL SECTOR will not preserve track status unless directed to do so.

It is strongly recommended that any multiple sector transfers using the INITIALIZE command not cross a track boundary, since all address verification and status checks are off. In addition, a track boundary *must* not be crossed during a multiple sector transfer whenever tracks are being flagged spare or defective since the automatic track sparing algorithm may not work.

SALES OFFICES

Arranged alphabetically by country

ANGOLA
Telectra
Empresa Técnica de Equipamentos Eléctricos, S.A.R.L.
R. Barbosa Rodrigues,
41-1° DT.°
Caixa Postal, 6487
Luanda
Tel: 35515/6

ARGENTINA
Hewlett-Packard Argentina S.A.
Santa Fe 2035, Martinez
6140 **Buenos Aires**
Tel: 792-1239, 798-6086
Telex: 122443 AR CIGY

Biotron S.A.C.I.y M.
Avda. Paseo Colon 221
9 piso
1399 **Buenos Aires**
Tel: 30-4846/1851/8384
34-9356/0460/4551
Telex: (33) 17595 BIO AR

AUSTRALIA

AUSTRALIA CAPITAL TERR.
Hewlett-Packard Australia Pty. Ltd.
121 Wollongong Street
Fyshwick, 2609
Tel: 804244
Telex: 62650

NEW SOUTH WALES
Hewlett-Packard Australia Pty. Ltd.
31 Bridge Street
Pymble, 2073
Tel: 4496566
Telex: 21561

QUEENSLAND
Hewlett Packard Australia Pty. Ltd.
5th Floor
Teachers Union Building
495-499 Boundary Street
Spring Hill, 4000
Tel: 2291544

SOUTH AUSTRALIA
Hewlett-Packard Australia Pty. Ltd.
153 Greenhill Road
Parkside, 5063
Tel: 2725911
Telex: 82536

VICTORIA
Hewlett-Packard Australia Pty. Ltd.
31-41 Joseph Street
Blackburn, 3130
Tel: 89-6351
Telex: 31024 MELB

WESTERN AUSTRALIA
Hewlett-Packard Australia Pty. Ltd.
141 Stirling Highway
Nedlands, 6009
Tel: 3865455
Telex: 93859

AUSTRIA
Hewlett-Packard Ges.m.b.H.
Wehlistrasse 29
P.O. Box 7
A-1205 **Vienna**
Tel: 35-16-21-0
Telex: 13582/135066

BAHRAIN
Medical Only
Wael Pharmacy
P.O. Box 648
Bahrain
Tel: 54886, 56123
Telex: 8550 WAEL GJ

Al Hamidiya Trading and Contracting
P.O. Box 20074
Manama
Tel: 259978, 259958
Telex: 8895 KALDIA GJ

BANGLADESH
The General Electric Co. of Bangladesh Ltd.
Magnet House 72
Dilkusha Commercial Area
Motijheil, Dacca 2
Tel: 252415, 252419
Telex: 734

BELGIUM
Hewlett-Packard Benelux S.A./N.V.
Avenue du Col-Vert, 1,
(Groenkraaglaan)
B-1170 **Brussels**
Tel: (02) 660 50 50
Telex: 23-494 paloben bru

BRAZIL
Hewlett-Packard do Brasil I.e.C. Ltda.
Alameda Rio Negro, 750
Alphaville
06400 **Barueri** SP
Tel: 429-3222

Hewlett-Packard do Brasil I.e.C. Ltda.
Rua Padre Chagas, 32
90000-**Pôrto Alegre**-RS
Tel: 22-2998, 22-5621

Hewlett-Packard do Brasil I.e.C. Ltda.
Av. Epitacio Pessoa, 4664
22471-**Rio de Janeiro**-RJ
Tel: 286-0237
Telex: 021-21905 HPBR-BR

CANADA

ALBERTA
Hewlett-Packard (Canada) Ltd.
11620A - 168th Street
Edmonton T5M 3T9
Tel: (403) 452-3670
TWX: 610-831-2431

Hewlett-Packard (Canada) Ltd.
210, 7220 Fisher St. S.E.
Calgary T2H 2H8
Tel: (403) 253-2713
TWX: 610-821-6141

BRITISH COLUMBIA
Hewlett-Packard (Canada) Ltd.
10691 Shellbridge Way
Richmond V6X 2W7
Tel: (604) 270-2277
TWX: 610-925-5059

MANITOBA
Hewlett-Packard (Canada) Ltd.
380-550 Century St.
St. James,
Winnipeg R3H 0Y1
Tel: (204) 786-6701
TWX: 610-671-3531

NOVA SCOTIA
Hewlett-Packard (Canada) Ltd.
P.O. Box 931
800 Windmill Road
Dartmouth B3B 1L1
Tel: (902) 469-7820
TWX: 610-271-4482

ONTARIO
Hewlett-Packard (Canada) Ltd.
1020 Morrison Dr.
Ottawa K2H 8K7
Tel: (613) 820-6483
TWX: 610-563-1636

Hewlett-Packard (Canada) Ltd.
6877 Goreway Drive
Mississauga L4V 1M8
Tel: (416) 678-9430
TWX: 610-492-4246

Hewlett-Packard (Canada) Ltd.
552 Newbold Street
London N6E 2S5
Tel: (519) 686-9181
TWX: 610-352-1201

QUEBEC
Hewlett-Packard (Canada) Ltd.
275 Hymus Blvd.
Pointe Claire H9R 1G7
Tel: (514) 697-4232
TWX: 610-422-3022

FOR CANADIAN AREAS NOT LISTED:
Contact Hewlett-Packard (Canada) Ltd. in Mississauga.

CHILE
Jorge Calcagni y Cia. Ltda.
Arturo Burhle 065
Casilla 16475
Correo 9, **Santiago**
Tel: 220222
Telex: JCALCAGNI

COLOMBIA
Instrumentación
Henrik A. Langebaek & Kier S.A.
Carrera 7 No. 48-75
Apartado Aéreo 6287
Bogotá, 1 D.E.
Tel: 269-8877
Telex: 44400

Instrumentación
H.A. Langebaek & Kier S.A.
Carrera 63 No. 49-A-31
Apartado 54098
Medellin
Tel: 304475

COSTA RICA
Cientifica Costarricense S.A.
Avenida 2, Calle 5
San Pedro de Montes de Oca
Apartado 10159
San José
Tel: 24-38-20, 24-08-19
Telex: 2367 GALGUR CR

CYPRUS
Kypronics
19 Gregorios Xenopoulos Street
P.O. Box 1152
Nicosia
Tel: 45628/29
Telex: 3018

CZECHOSLOVAKIA
Hewlett-Packard
Obchodni zastupitelstvi v CSSR
Pisemny styk
Post. schranka 27
CS 118 01 **Praha** 011
CSSR

Vyvojova a Provozni Zakladna
Vyzkumnych Ustavu v Bechovicich
CSSR-25097 **Bechovice u Prahy**
Tel: 89 93 41
Telex: 12133

Institute of Medical Bionics
Vyskumny Ustav Lekarskej Bioniky
Jedlova 6
CS-88346 **Bratislava-Kramare**
Tel: 44-551
Telex: 93229

DENMARK
Hewlett-Packard A/S
Datavej 52
DK-3460 **Birkerod**
Tel: (02) 81 66 40
Telex: 37409 hpas dk

Hewlett-Packard A/S
Navervej 1
DK-8600 **Silkeborg**
Tel: (06) 82 71 66
Telex: 37409 hpas dk

ECUADOR
CYEDE Cia. Ltda.
P.O. Box 6423 CCI
Av. Eloy Alfaro 1749
Quito
Tel: 450-975, 243-052
Telex: 2548 CYEDE ED

Medical Only
Hospitalar S.A.
Casilla 3590
Robles 625
Quito
Tel: 545-250

EGYPT
I.E.A.
International Engineering Associates
24 Hussein Hegazi Street
Kasr-el-Aini
Cairo
Tel: 23 829
Telex: 93830

SAMITRO
Sami Amin Trading Office
18 Abdel Aziz Gawish
Abdine-Cairo
Tel: 24932

EL SALVADOR
IPESA
Bulevar de los Heroes 11-48
Edificio Sarah 1148
San Salvador
Tel: 252787

ETHIOPIA
Abdella Abdulmalik
P.O. Box 2635
Addis Ababa
Tel: 11 93 40

FINLAND
Hewlett-Packard Oy
Revontulentie, 7
SF-02100 **Espoo** 10
Tel: (90) 455 0211
Telex: 121563 hewpa sf

FRANCE
Hewlett-Packard France
Zone d'activites de Courtaboeuf
Avenue des Tropiques
Boite Postale 6
91401 **Orsay**-Cédex
Tel: (1) 907 78 25
TWX: 600048F

Hewlett-Packard France
Chemin des Mouilles
B.P. 162
69130 **Ecully**
Tel: (78) 33 81 25
TWX: 310617F

Hewlett-Packard France
20, Chemin de La Cépière
31081 **Toulouse Le Mirail**-Cédex
Tel: (61) 40 11 12

Hewlett-Packard France
Le Ligoures
Place Romée de Villeneuve
13100 **Aix-en-Provence**
Tel: (42) 59 41 02
TWX: 410770F

Hewlett-Packard France
2, Allee de la Bourgonette
35100 **Rennes**
Tel: (99) 51 42 44
TWX: 740912F

Hewlett-Packard France
18, rue du Canal de la Marne
67300 **Schiltigheim**
Tel: (88) 83 08 10
TWX: 890141F

Hewlett-Packard France
Immeuble péricentre
rue van Gogh
59650 **Villeneuve** D'Ascq
Tel: (20) 91 41 25
TWX: 160124F

Hewlett-Packard France
Bâtiment Ampère
Rue de la Commune de Paris
B.P. 300
93153 **Le Blanc Mesnil**-Cédex
Tel: (01) 931 88 50
Telex: 211032F

Hewlett-Packard France
Av. du Pdt. Kennedy
33700 **Merignac**
Tel: (56) 97 01 81

Hewlett-Packard France
Immeuble Lorraine
Boulevard de France
91035 **Evry**-Cédex
Tel: 077 96 60
Telex: 692315F

Hewlett-Packard France
23 Rue Lothaire
57000 **Metz**
Tel: (87) 65 53 50

GERMAN FEDERAL REPUBLIC
Hewlett-Packard GmbH
Vertriebszentrale Frankfurt
Berner Strasse 117
Postfach 560 140
D-6000 **Frankfurt** 56
Tel: (06011) 50041
Telex: 04 13249 hpffm d

Hewlett-Packard GmbH
Technisches Büro Böblingen
Herrenberger Strasse 110
D-7030 **Böblingen,**
Württemberg
Tel: (07031) 667-1
Telex: 07265739 bbn

Hewlett-Packard GmbH
Technisches Büro Düsseldorf
Emanuel-Leutze-Str. 1 (Seestern)
D-4000 **Düsseldorf**
Tel: (0211) 5971-1
Telex: 085/86 533 hpdd d

Hewlett-Packard GmbH
Technisches Büro Hamburg
Kapstadtring 5
D-2000 **Hamburg** 60
Tel: (040) 63804-1
Telex: 21 63 032 hphh d

Hewlett-Packard GmbH
Technisches Büro Hannover
Am Grossmarkt 6
D-3000 **Hannover** 91
Tel: (0511) 46 60 01
Telex: 092 3259

Hewlett-Packard GmbH
Technisches Büro Nürnberg
Neumeyerstrasse 90
D-8500 **Nürnberg**
Tel: (0911) 52 20 83
Telex: 0623 860

Hewlett-Packard GmbH
Technisches Büro München
Eschenstrasse 5
D-8021 **Taufkirchen**
Tel: (089) 6117-1
Telex: 0524985

Hewlett-Packard GmbH
Technisches Büro Berlin
Kaithstrasse 2-4
D-1000 **Berlin** 30
Tel: (030) 24 90 86
Telex: 018 3405 hpbln d

GREECE
Kostas Karayannis
8 Omirou Street
Athens 133
Tel: 32 30 303/32/37 731
Telex: 21 59 62 RKAR GR

GUAM
Guam Medical Supply, Inc.
Suite C, Airport Plaza
P.O. Box 8947
Tamuning 96911
Tel: 646-4513

GUATEMALA
IPESA
Avenida Reforma 3-48
Zona 9
Guatemala City
Tel: 316627, 314786, 66471-5, ext. 9
Telex: 4192 Teletro Gu

SALES OFFICES

Arranged alphabetically by country (cont.)

HONG KONG
Hewlett-Packard Hong Kong Ltd.
11th Floor, Four Seas Bldg.
212 Nathan Rd.
Kowloon
Tel: 3-697446 (5 lines)
Telex: 36678 HX

Medical/Analytical Only
Schmidt & Co. (Hong Kong) Ltd.
Wing On Centre, 28th Floor
Connaught Road, C.
Hong Kong
Tel: 5-455644
Telex: 74766 SCHMX HX

INDIA
Blue Star Ltd.
Sahas
414/2 Vir Savarkar Marg
Prabhadevi
Bombay 400 025
Tel: 45 78 87
Telex: 011-4093

Blue Star Ltd.
Band Box House
Prabhadevi
Bombay 400 025
Tel: 45 73 01
Telex: 011-3751

Blue Star Ltd.
Bhavdeep
Stadium Road
Ahmedabad 380 014
Tel: 43922
Telex: 012-234

Blue Star Ltd.
7 Hare Street
Calcutta 700 001
Tel: 23-0131
Telex: 021-7655

Blue Star Ltd.
Bhandari House
91 Nehru Place
New Delhi 110 024
Tel: 682547
Telex: 031-2463

Blue Star Ltd.
T.C. 7/603 'Poornima'
Maruthankuzhi
Trivandrum 695 013
Tel: 65799
Telex: 0884-259

Blue Star Ltd.
11 Magarath Road
Bangalore 560 025
Tel: 55668
Telex: 0845-430

Blue Star Ltd.
Meeakshi Mandiram
XXXXV/1379-2 Mahatma
Gandhi Rd.
Cochin 682 016
Tel: 32069
Telex: 085-514

Blue Star Ltd.
1-1-117/1 Sarojini Devi Road
Secunderabad 500 033
Tel: 70126
Telex: 0155-459

Blue Star Ltd.
133 Kodambakkam High Road
Madras 600 034
Tel: 82057
Telex: 041-379

ICELAND
Medical Only
Eliding Trading Company Inc.
Hafnarnvoli - Tryggvagötu
P.O. Box 895
IS-**Reykjavik**
Tel: 1 58 20/1 63 03

INDONESIA
BERCA Indonesia P.T.
P.O. Box 496/Jkt.
Jln. Abdul Muis 62
Jakarta
Tel: 349255, 349886
Telex: 46748 BERSIL IA

BERCA Indonesia P.T.
P.O. Box 174/Sby.
23 Jln. Jimerto
Surabaya
Tel: 42027

IRELAND
Hewlett-Packard Ltd.
Kestrel House
Clanwilliam Place
Lower Mount Street
Dublin 2, Eire

Hewlett-Packard Ltd.
2C Avongberg Ind. Est.
Long Mile Road
Dublin 12
Tel: 514322/514224
Telex: 30439

Medical Only
Cardiac Services (Ireland) Ltd.
Kilmore Road
Artane
Dublin 5, Eire
Tel: (01) 315820

ISRAEL
Electronics Engineering Div.
of Motorola Israel Ltd.
16, Kremenetski Street
P.O. Box 25016
Tel-Aviv
Tel: 38973
Telex: 33569, 34164

ITALY
Hewlett-Packard Italiana S.p.A.
Via G. Di Vittorio, 9
20063 **Cernusco Sul Naviglio** (MI)
Tel: (2) 903691
Telex: 334632 HEWPACKIT

Hewlett-Packard Italiana S.p.A.
Via Turazza, 14
35100 **Padova**
Tel: (49) 664888
Telex: 430315 HEWPACKI

Hewlett-Packard Italiana S.p.A.
Via G. Armellini 10
1-00143 **Roma**
Tel: (06) 54 69 61
Telex: 610514

Hewlett-Packard Italiana S.p.A.
Corso Giovanni Lanza 94
I-10133 **Torino**
Tel: (011) 659308
Telex: 221079

Hewlett-Packard Italiana S.p.A.
Via Principe Nicola 43 G/C
I-95126 **Catania**
Tel: (095) 37 05 04
Telex: 970291

Hewlett-Packard Italiana S.p.A.
Via Nuova san Rocco A
Capadimonte, 62A
80131 **Napoli**
Tel: (081) 710698

Hewlett-Packard Italiana S.p.A.
Via Martin Luther King, 38/111
I-40132 **Bologna**
Tel: (051) 402394
Telex: 511630

JAPAN
Yokogawa-Hewlett-Packard Ltd.
29-21, Takaido-Higashi 3-chome
Suginami-ku, **Tokyo** 168
Tel: 03-331-6111
Telex: 232-2024 YHP-Tokyo

Yokogawa-Hewlett-Packard Ltd.
Chuo Bldg., 4th Floor
4-20, Nishinakajima 5-chome
Yodogawa-ku, Osaka-shi
Osaka, 532
Tel: 06-304-6021
Telex: 523-3624

Yokogawa-Hewlett-Packard Ltd.
Sunitomo Seimei Nagaya Bldg.
11-2 Shimosasajima-cho,
Nakamura-ku, **Nagoya,** 450
Tel: 052 571-5171

Yokogawa-Hewlett-Packard Ltd.
Tanigawa Building
2-24-1 Tsuruya-cho
Kanagawa-ku
Yokohama, 221
Tel: 045-312-1252
Telex: 382-3204 YHP YOK

Yokogawa-Hewlett-Packard Ltd.
Mito Mitsui Building
105, 1-chome, San-no-maru
Mito, Ibaragi 310
Tel: 0292-25-7470

Yokogawa-Hewlett-Packard Ltd.
Inoue Building
1348-3, Asahi-cho, 1-chome
Atsugi, Kanagawa 243
Tel: 0462-24-0452

Yokogawa-Hewlett-Packard Ltd.
Kumagaya Asahi
Hachijuni Building
4th Floor
3-4, Tsukuba
Kumagaya, Saitama 360
Tel: 0485-24-6563

JORDAN
Mouasher Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1456

KENYA
ADCOM Ltd., Inc.
P.O. Box 30070
Nairobi
Tel: 331955
Telex: 22639

Medical Only
International Aeradio (E.A.) Ltd.
P.O. Box 19012
Nairobi Airport
Nairobi
Tel: 336055/56
Telex: 22201/22301

Medical Only
International Aeradio (E.A.) Ltd.
P.O. Box 95221
Mombasa

KOREA
Samsung Electronics Co., Ltd.
4759 Shingil-6-Dong
Yeong Deung POU
Seoul
Tel: 833-4122, 4121
Telex: SAMSAN 27364

KUWAIT
Al-Khaldiya Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 4910/41 1726
Telex: 2481 Areeg kt

LUXEMBURG
Hewlett-Packard Beneluz S.A./N.V.
Avenue du Col-Vert, 1
(Groenkraaglaan)
B-1170 **Brussels**
Tel: (02) 660 5050
Telex: 23 494

MALAYSIA
Hewlett-Packard Sales
(Malaysia) Sdn. Bhd.
Suite 2.21/2.22
Bangunan Angkasa Raya
Jalan Ampang
Kuala Lumpur
Tel: 483680, 485653

Protel Engineering
P.O. Box 1917
Lot 259, Satok Road
Kuching, **Sarawak**
Tel: 53544

MEXICO
Hewlett-Packard Mexicana,
S.A. de C.V.
Av. Periférico Sur No. 6501
Tepepan, Xochimilco
Mexico 23, D.F.
Tel: 905-676-4600
Telex: 017-74-507

Hewlett-Packard Mexicana,
S.A. de C.V.
Rio Volga #600
Col. Del Valle
Monterrey, N.L.
Tel: 78-32-10

MOROCCO
Dolbeau
81 rue Karatchi
Casablanca
Tel: 3041 82
Telex: 23051/22822

Gerep
2, rue d'Agadir
Boite Postal 156
Casablanca
Tel: 272093/5
Telex: 23 739

MOZAMBIQUE
A.N. Goncalves, Ltd.
162, 1° Apt. 14 Av. D. Luis
Caixa Postal 107
Maputo
Tel: 27091, 27114
Telex: 6-203 NEGON Mo

NETHERLANDS
Hewlett-Packard Benelux N.V.
Van Heuven Goedhartlaan 121
P.O. Box 667
1181KK **Amstelveen**
Tel: (20) 47 20 21
Telex: 13 216

NEW ZEALAND
Hewlett-Packard (N.Z.) Ltd.
4-12 Cruickshank Street
Kilbirnie, Wellington 3
P.O. Box 9443
Courtney Place
Wellington
Tel: 877-199

Hewlett-Packard (N.Z.) Ltd.
P.O. Box 26-189
169 Manukau Road
Epsom, **Auckland**
Tel: 687-159

Analytical/Medical Only
Northrop Instruments &
Systems Ltd.,
Sturdee House
85-87 Ghuznee Street
P.O. Box 2406
Wellington
Tel: 850-091
Telex: NZ 31291

Northrup Instruments &
Systems Ltd.
Eden House, 44 Khyber Pass Rd.
P.O. Box 9682, Newmarket
Auckland 1
Tel: 794-091

Northrup Instruments &
Systems Ltd.
Terrace House, 4 Oxford Terrace
P.O. Box 8388
Christchurch
Tel: 64-165

NIGERIA
The Electronics
Instrumentations Ltd.
N6B/770 Oyo Road
Oluseun House
P.M.B. 5402
Ibadan
Tel: 461577
Telex: 31231 TEIL NG

The Electronics
Instrumentations Ltd.
144 Agege Motor Road, Mushin
P.O. Box 481
Mushin, **Lagos**

NORWAY
Hewlett-Packard Norge A/S
Ostendalen 18
P.O. Box 34
1345 **Osteraas**
Tel: (02) 1711 80
Telex: 16621 hpnas n

Hewlett-Packard Norge A/S
Nygaardsgaten 114
P.O. Box 4210
5013 Nygaardsgaten,
Bergen
Tel: (05) 21 97 33

PANAMA
Electrónico Balboa, S.A.
Aparatado 4929
Panama 5
Calle Samuel Lewis
Edificio "Alfa," No. 2
Ciudad de Panama
Tel: 64-2700
Telex: 3483103 Curundu,
Canal Zone

PERU
Compañía Electro Médica S.A.
Los Flamencos 145
San Isidro Casilla 1030
Lima 1
Tel: 41-4325
Telex: Pub. Booth 25424 SISIDRO

PAKISTAN
Mushko & Company Ltd.
Oosman Chambers
Abdullah Haroon Road
Karachi-3
Tel: 511027, 512927
Telex: 2894

Mushko & Company, Ltd.
10, Bazar Rd.
Sector G-6/4
Islamabad
Tel: 28264

PHILIPPINES
The Online Advanced Systems
Corporation
Rico House
Amorsolo cor. Herrera Str.
Legaspi Village, Makati
P.O. Box 1510
Metro **Manila**
Tel: 85-35-81, 85-34-91,
85-32-21
Telex: 3274 ONLINE

RHODESIA
Field Technical Sales
45 Kelvin Road North
P.O. Box 3458
Salisbury
Tel: 705231 (5 lines)
Telex: RH 4122

POLAND
Biuro Informacji Technicznej
Hewlett-Packard
Ul Stawki 2, 6P
PL00-950 **Warszawa**
Tel: 39 59 62, 39 51 87
Telex: 81 24 53

PORTUGAL
Telectra-Empresa Técnica de
Equipamentos Eléctricos S.a.r.l.
Rua Rodrigo da Fonseca 103
P.O. Box 2531
P-**Lisbon** 1
Tel: (19) 68 60 72
Telex: 12598

Medical Only
Mundinter
Intercambio Mundial de Comércio
S.a.r.l.
P.O. Box 2761
Avenida Antonio Augusto
de Aguiar 138
P-**Lisbon**
Tel: (19) 53 21 31/7
Telex: 16691 munter p

PUERTO RICO
Hewlett-Packard Inter- Americas
Puerto Rico Branch Office
Calle 272,
#203 Urb. Country Club
Carolina 00630
Tel: (809) 762-7255
Telex: 345 0514

QATAR
Nasser Trading & Contracting
P.O. Box 1563
Doha
Tel: 22170
Telex: 4439 NASSER

ROMANIA
Hewlett-Packard Reprezentanta
Bd.n. Balcescu 16
Bucuresti
Tel: 15 80 23/13 88 85
Telex: 10440

SAUDI ARABIA
Modern Electronic
Establishment (Head Office)
P.O. Box 1228, Baghdadiah Street
Jeddah
Tel: 27 798
Telex: 40035
Cable: ELECTA JEDDAH

Modern Electronic Establishment
(Branch)
P.O. Box 2728
Riyadh
Tel: 62596/66232
Telex: 202049

Modern Electronic Establishment
(Branch)
P.O. Box 193
Al-Khobar
Tel: 44678-44813
Telex: 670136
Cable: ELECTA AL-KHOBAR

SINGAPORE
Hewlett-Packard Singapore
(Pte.) Ltd.
6th Floor, Inchcape House
450-452 Alexandra Road
P.O. Box 58
Alexandra Post Office
Singapore 9115
Tel: 631788
Telex: HPSG RS 21486

SOUTH AFRICA
Hewlett-Packard South Africa
(Pty.), Ltd.
Private Bag Wendywood,
Sandton, Transvaal, 2144
Hewlett-Packard Centre
Daphne Street, Wendywood,
Sandton, 2144
Tel: 802-5111/25
Telex: 8-4782

Hewlett-Packard South Africa
(Pty.), Ltd.
P.O. Box 120
Howard Place,
Cape Province, 7450
Pine Park Centre, Forest Drive,
Pinelands,
Cape Province, 7405
Tel: 53-7955 thru 9
Telex: 57-0006

SPAIN
Hewlett-Packard Española, S.A.
Calle Jerez 3
E-**Madrid** 16
Tel: (1) 458 26 00 (10 lines)
Telex: 23515 hpe

Hewlett-Packard Española S.A.
Colonia Mirasierra
Edificio Juban
c/o Costa Brava, 13
Madrid 34

Hewlett-Packard Española, S.A.
Milanesado 21-23
E-**Barcelona** 17
Tel: (3) 203 6200 (5 lines)
Telex: 52603 hpbe e

Hewlett-Packard Española, S.A.
Av Ramón y Cajal, 1
Edificio Sevilla, planta 9°
E-**Sevilla** 5
Tel: 64 44 54/58

Hewlett-Packard Española S.A.
Edificio Albia II 7° B
E-**Bilbao** 1
Tel: 23 83 06/23 82 06

Hewlett-Packard Española S.A.
C/Ramon Gordillo 1 (Entlo.)
E-**Valencia** 10
Tel: 96-361.13.54/361.13.58

SRI LANKA
Metropolitan Agencies Ltd.
209/9 Union Place
Colombo 2
Tel: 35947
Telex: 1377METROLTD CE

SUDAN
Radison Trade
P.O. Box 921
Khartoum
Tel: 44048
Telex: 375

SURINAM
Surtel Radio Holland N.V.
Grote Hofstr. 3-5
P.O. Box 155
Paramaribo
Tel: 72118, 77880

SWEDEN
Hewlett-Packard Sverige AB
Enighetsvägen 3, Fack
S-161 **Bromma** 20
Tel: (08) 730 05 50
Telex: 10721
Cable: MEASUREMENTS
Stockholm

Hewlett-Packard Sverige AB
Frötallsgatan 30
S-421 32 **Västra**
Frölunda
Tel: (031) 49 09 50
Telex: 10721 via Bromma office

SWITZERLAND
Hewlett-Packard (Schweiz) AG
Zürcherstrasse 20
P.O. Box 307
CH-8952 **Schlieren-**
Zürich
Tel: (01) 7305240
Telex: 53933 hpag ch
Cable: HPAG CH

Hewlett-Packard (Schweiz) AG
Château Bloc 19
CH-1219 **Le Lignon-**
Geneva
Tel: (022) 96 03 22
Telex: 27333 hpag ch
Cable: HEWPACKAG Geneva

SYRIA
General Electronic Inc.
Nuri Basha-Ahnaf Ebn Kays Street
P.O. Box 5781
Damascus
Tel: 33 24 87
Telex: 11215 ITIKAL
Cable: ELECTROBOR DAMASCUS

Medical only
Sawah & Co.
Place Azmé
B.P. 2308
Damascus
Tel: 16 367-19 697-14 268
Telex: 11304 SATACO SY
Cable: SAWAH, DAMASCUS

Suleiman Hilal El Mlawi
P.O. Box 2528
Mamoun Bitar Street, 56-58
Damascus
Tel: 11 46 63
Telex: 11270
Cable: HILAL DAMASCUS

TAIWAN
Hewlett-Packard Far East Ltd.
Taiwan Branch
Bank Tower, 5th Floor
205 Tun Hau North Road
Taipei
Tel: (02) 751-0404 (15 lines)

Hewlett-Packard Far East Ltd.
Taiwan Branch
68-2, Chung Cheng 3rd. Road
Kaohsiung
Tel: (07) 242318-Kaohsiung

Analytical Only
San Kwang Instruments Co., Ltd.
20 Yung Sui Road
Taipei
Tel: 3615446-9 (4 lines)
Telex: 22894 SANKWANG

TANZANIA
Medical Only
International Aeradio (E.A.), Ltd.
P.O. Box 861
Dar es Salaam
Tel: 21251 Ext. 265
Telex: 41030

THAILAND
UNIMESA Co. Ltd.
Elcom Research Building
2538 Sukumvit Ave.
Bangchak, Bangkok
Tel: 39-32-387, 39-30-338

TRINIDAD & TOBAGO
CARTEL
Caribbean Telecoms Ltd.
P.O. Box 732
69 Frederick Street
Port-of-Spain
Tel: 62-53068

TUNISIA
Tunisie Electronique
31 Avenue de la Liberte
Tunis
Tel: 280 144

Corema
1 ter. Av. de Carthage
Tunis
Tel: 253 821
Telex: 12319 CABAM TN

TURKEY
TEKNIM Company Ltd.
Riza Sah Pehlevi
Caddesi No. 7
Kavaklidere, **Ankara**
Tel: 275800
Telex: 42155

Teknim Com., Ltd.
Barbaros Bulvari 55/12
Besikyas, **Istanbul**
Tel: 613 546
Telex: 23540

E.M.A.
Muhendislik Kollektif Sirketi
Mediha Eldem Sokak 41/6
Yüksel Caddesi
Ankara
Tel: 17 56 22

Yilmaz Ozyurek
Milli Mudafaa Cad 16/6
Kizilay
Ankara
Tel: 25 03 09 - 17 80 26
Telex: 42576 OZEK TR

UNITED ARAB EMIRATES
Emitac Ltd. (Head Office)
P.O. Box 1641
Sharjah
Tel: 354121/3
Telex: 8136

Emitac Ltd. (Branch Office)
P.O. Box 2711
Abu Dhabi
Tel: 331370/1

UNITED KINGDOM
Hewlett-Packard Ltd.
King Street Lane
Winnersh, Wokingham
Berkshire RG11 5AR
GB-England
Tel: (0734) 784774
Telex: 84 71 78/9

Hewlett-Packard Ltd.
Fourier House,
257-263 High Street
London Colney
St. Albans, Herts
GB-England
Tel: (0727) 24400
Telex: 1-8952716

Hewlett-Packard Ltd.
Trafalgar House
Navigation Road
Altrincham
Cheshire WA14 1NU
GB-England
Tel: (061) 928 6422
Telex: 668068

Hewlett-Packard Ltd.
Lygon Court
Hereward Rise
Dudley Road
Halesowen,
West Midlands, B62 8SD
GB-England
Tel: (021) 501 1221
Telex: 339105

Hewlett-Packard Ltd.
Wedge House
799, London Road
Thornton Heath
Surrey, CR4 6XL
GB-England
Tel: (01) 684-0103/8
Telex: 946825

Hewlett-Packard Ltd.
14 Wesley St
Castleford
Yorks WF10 1AE
Tel: (0977) 550016
TWX: 5557335

Hewlett-Packard Ltd.
Tradax House
St. Mary's Walk
Maidenhead
Berkshire, SL6 1ST
GB-England

Hewlett-Packard Ltd.
Morley Road
Staplehill
Bristol, BS16 4QT
GB-England

Medical Only
Cardiac Services Co.
95A Finaghy Rd. South
Belfast BT10 0BY
GB-Northern Ireland
Tel: (0232) 625566
Telex: 747626

Hewlett-Packard Ltd.
South Queensferry
West Lothian, EH30 9TG
GB-Scotland
Tel: (031) 331 1188
Telex: 72682

UNITED STATES

ALABAMA
700 Century Park South, Suite
128
Birmingham 35226
Tel: (205) 822-6802

P.O. Box 4207
8290 Whitesburg Dr.
Huntsville 35802
Tel: (205) 881-4591

ARIZONA
2336 E. Magnolia St.
Phoenix 85034
Tel: (602) 273-8000

2424 East Aragon Rd.
Tucson 85706
Tel: (602) 273-8000

***ARKANSAS**
Medical Service Only
P.O. Box 5646
Brady Station
Little Rock 72215
Tel: (501) 376-1844

CALIFORNIA
1579 W. Shaw Ave.
Fresno 93771
Tel: (209) 224-0582

1430 East Orangethorpe Ave.
Fullerton 92631
Tel: (714) 870-1000

5400 West Rosecrans Blvd.
P.O. Box 92105
World Way Postal Center
Los Angeles 90009
Tel: (213) 970-7500
TWX: 910-325-6608

3939 Lankershim Boulevard
North Hollywood 91604
Tel: (213) 877-1282
TWX: 910-499-2671

3200 Hillview Av
Palo Alto, CA 94304
Tel: (408) 988-7000

646 W. North Market Blvd.
Sacramento 95834
Tel: (916) 929-7222

9606 Aero Drive
P.O. Box 23333
San Diego 92123
Tel: (714) 279-3200

363 Brookhollow Dr.
Santa Ana, CA 92705
Tel: (714) 641-0977

3003 Scott Boulevard
Santa Clara 95050
Tel: (408) 988-7000
TWX: 910-338-0518

454 Carlton Court
So. San Francisco
94080
Tel: (415) 877-0772

***Tarzana**
Tel: (213) 705-3344

COLORADO
5600 DTC Parkway
Englewood 80110
Tel: (303) 771-3455

SALES OFFICES

Arranged alphabetically by country (cont.)

CONNECTICUT
47 Barnes Industrial Road
Barnes Park South
Wallingford 06492
Tel: (203) 265-7801

FLORIDA
P.O. Box 24210
2727 N.W. 62nd Street
Ft. Lauderdale 33309
Tel: (305) 973-2600

4080 Woodcock Drive #132
Brownett Building
Jacksonville 32207
Tel: (904) 398-0663

P.O. Box 13910
6177 Lake Ellenor Dr.
Orlando 32809
Tel: (305) 859-2900

P.O. Box 12826
Suite 5, Bldg. 1
Office Park North
Pensacola 32575
Tel: (904) 476-8422
110 South Hoover Blvd.
Suite 120
Tampa 33609
Tel: (813) 872-0900

GEORGIA
P.O. Box 105005
450 Interstate North Parkway
Atlanta 30348
Tel: (404) 955-1500
TWX: 810-766-4890

Medical Service Only
*__Augusta__ 30903
Tel: (404) 736-0592

P.O. Box 2103
1172 N. Davis Drive
Warner Robins 31098
Tel: (912) 922-0449

HAWAII
2875 So. King Street
Honolulu 96826
Tel: (808) 955-4455

ILLINOIS
211 Prospect Rd.
Bloomington 61701
Tel: (309) 663-0383

5201 Tollview Dr.
Rolling Meadows
60008
Tel: (312) 255-9800
TWX: 910-687-2260

INDIANA
7301 North Shadeland Ave.
Indianapolis 46250
Tel: (317) 842-1000
TWX: 810-260-1797

IOWA
2415 Heinz Road
Iowa City 52240
Tel: (319) 351-1020

KENTUCKY
10170 Linn Station Road
Suite 525
Louisville 40223
Tel: (502) 426-0100

LOUISIANA
P.O. Box 1449
3229-39 Williams Boulevard
Kenner 70062
Tel: (504) 443-6201

MARYLAND
7121 Standard Drive
Parkway Industrial Center
Hanover 21076
Tel: (301) 796-7700
TWX: 710-862-1943

2 Choke Cherry Road
Rockville 20850
Tel: (301) 948-6370
TWX: 710-828-9684

MASSACHUSETTS
32 Hartwell Ave.
Lexington 02173
Tel: (617) 861-8960
TWX: 710-326-6904

MICHIGAN
23855 Research Drive
Farmington Hills 48024
Tel: (313) 476-6400

724 West Centre Ave.
Kalamazoo 49002
Tel: (616) 323-8362

MINNESOTA
2400 N. Prior Ave.
St. Paul 55113
Tel: (612) 636-0700

MISSISSIPPI
322 N. Mart Plaza
Jackson 39206
Tel: (601) 982-9363

MISSOURI
11131 Colorado Ave.
Kansas City 64137
Tel: (816) 763-8000
TWX: 910-771-2087

1024 Executive Parkway
St. Louis 63141
Tel: (314) 878-0200

NEBRASKA
Medical Only
7101 Mercy Road
Suite 101
Omaha 68106
Tel: (402) 392-0948

NEVADA
*__Las Vegas__
Tel: (702) 736-6610

NEW JERSEY
Crystal Brook Professional Building
Route 35
Eatontown 07724
Tel: (201) 542-1384

W. 120 Century Rd.
Paramus 07652
Tel: (201) 265-5000
TWX: 710-990-4951

NEW MEXICO
P.O. Box 11634
Station E
11300 Lomas Blvd., N.E.
Albuquerque 87123
Tel: (505) 292-1330
TWX: 910-989-1185

156 Wyatt Drive
Las Cruces 88001
Tel: (505) 526-2484
TWX: 910-9983-0550

NEW YORK
6 Automation Lane
Computer Park
Albany 12205
Tel: (518) 458-1550
TWX: 710-444-4961

650 Perinton Hill Office Park
Fairport 14450
Tel: (716) 223-9950
TWX: 510-253-0092

No. 1 Pennsylvania Plaza
55th Floor
34th Street & 8th Avenue
New York 10001
Tel: (212) 971-0800

5858 East Molloy Road
Syracuse 13211
Tel: (315) 455-2486

1 Crossways Park West
Woodbury 11797
Tel: (516) 921-0300
TWX: 510-221-2183
Tel: (513) 671-7400

NORTH CAROLINA
5605 Roanne Way
Greensboro 27409
Tel: (919) 852-1800

OHIO
Medical/Computer Only
9920 Carver Road
Cincinnati 45242
Tel: (513) 891-9870
16500 Sprague Road
Cleveland 44130
Tel: (216) 243-7300
TWX: 810-423-9430

962 Crupper Ave.
Columbus 43229
Tel: (614) 436-1041

330 Progress Rd.
Dayton 45449
Tel: (513) 859-8202

OKLAHOMA
P.O. Box 32008
6301 N. Meridan Avenue
Oklahoma City 73112
Tel: (405) 721-0200

9920 E. 42nd Street
Suite 121
Tulsa 74145
Tel: (918) 665-3300

OREGON
17890 S.W. Lower Boones Ferry
Road
Tualatin 97062
Tel: (503) 620-3350

PENNSYLVANIA
1021 8th Avenue
King of Prussia Industrial Park
King of Prussia 19406
Tel: (215) 265-7000
TWX: 510-660-2670

111 Zeta Drive
Pittsburgh 15238
Tel: (412) 782-0400

SOUTH CAROLINA
P.O. Box 6442
6941-0 N. Trenholm Road
Columbia 29206
Tel: (803) 782-6493

TENNESSEE
8906 Kingston Pike
Knoxville 37919
Tel: (615) 691-2371

3070 Directors Row
Directors Square
Memphis 38131
Tel: (901) 346-8370

*__Nashville__
Medical Service Only
Tel: (615) 244-5448

TEXAS
4171 North Mesa
Suite C110
El Paso 79902
Tel: (915) 533-3555

P.O. Box 42816
10535 Harwin St.
Houston 77036
Tel: (713) 776-6400

*__Lubbock__
Medical Service Only
Tel: (806) 799-4472

P.O. Box 1270
201 E. Arapaho Rd.
Richardson 75081
Tel: (214) 231-6101

205 Billy Mitchell Road
San Antonio 78226
Tel: (512) 434-8241

UTAH
2160 South 3270 West Street
Salt Lake City 84119
Tel: (801) 972-4711

VIRGINIA
P.O. Box 9669
2914 Hungary Spring Road
Richmond 23228
Tel: (804) 285-3431

Computer Systems/Medical Only
Airport Executive Center
Suite 302
5700 Thurston Avenue
Virginia Beach 23455
Tel: (804) 460-2471

WASHINGTON
Bellefield Office Pk.
1203 - 114th Ave. S.E.
Bellevue 98004
Tel: (206) 454-3971
TWX: 910-443-2446

P.O. Box 4010
Spokane 99202
Tel: (509) 535-0864

*__WEST VIRGINIA__
Medical/Analytical Only
4604 Mac Corkle Ave., S.E.
Charleston 25304
Tel: (304) 925-0492

WISCONSIN
150 South Sunny Slope Road
Brookfield 53005
Tel: (414) 784-8800

FOR U.S. AREAS NOT LISTED:
Contact the regional office nearest you:
Atlanta, Georgia...**Los Angeles,** California...**Paramus,** New Jersey,...**Rolling Meadows**, Illinois. Their complete addresses are listed above.

USSR
Hewlett-Packard
Representative Office
USSR
Pokrovsky Boulevard 4/17-kw 12
Moscow 101000
Tel: 294.20.24
Telex: 7825 hewpak su

URUGUAY
Pablo Ferrando S.A.C.el.
Avenida Italia 2877
Casilla de Correo 370
Montevideo
Tel: 40-3102
Telex: 702 Public Booth
Para Pablo Ferrando

VENEZUELA
Hewlett-Packard de Venezuela C.A.
P.O. Box 50933
Caracas 105
Los Ruices Norte
3a Transversal
Edificio Segre
Caracas 107
Tel: 239-4133 (20 lines)
Telex: 25146 HEWPACK

YUGOSLAVIA
Iskra Commerce, n.sol.o.
Zastopstvo Hewlett-Packard
Obilicev Venac 26
YU 11000 **Beograd**
Tel: 636-955
Telex: 11530

Iskra Commerce, n.sol.o.
Zastopstvo Hewlett-Packard
Miklosiceva 38/VII
YU-61000 **Ljubljana**
Tel: 321-674, 315-879
Telex: 31583

ZAMBIA
R.J. Tilbury (Zambia) Ltd.
P.O. Box 2792
Lusaka
Tel: 73793

MEDITERRANEAN AND MIDDLE EAST COUNTRIES NOT SHOWN, PLEASE CONTACT:
Hewlett-Packard S.A.
Mediterranean and Middle East
Operations
35, Kolokotroni Street
Platia Kefallariou
GR-Kifissia-**Athens,** Greece
Tel: 8080359/429
Telex: 21-6588
Cable: HEWPACKSA Athens

SOCIALIST COUNTRIES NOT SHOWN, PLEASE CONTACT:
Hewlett-Packard Ges.m.b.H.
Handelskai 52
P.O. Box 7
A-1205 **Vienna,** Austria
Tel: (0222) 35 16 21 to 27
Cable: HEWPAK Vienna
Telex: 75923 hewpak a

OTHER AREAS NOT LISTED, CONTACT:
Hewlett-Packard Intercontinental
3495 Deer Creek Road
Palo Alto, California 94304
Tel: (415) 856-1501
TWX: 910-373-1267
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

Hewlett-Packard S.A.
7, rue du Bois-du-Lan
P.O. Box
CH-1217 Meyrin 2 - **Geneva**
Switzerland
Tel: (022) 82 70 00
Cable: HEWPACKSA Geneva
Telex: 2 24 86

*__Service Only__
3-10-80