Customer Service Training Department
John Fluke Mfg. Co., Inc.

This training course has been prepared by the Customer Service Department of the John Fluke Mfg. Co., Inc. to provide a useful learning tool which will increase technical troubleshooting skills. The Workbook approach allows the technician to gain an understanding of the instrument by performing exercises with a working instrument to isolate problems using a system approach. Should you have any questions, comments or suggestions about this course of instruction please feel free to communicate them to the Customer Service Training Department by letter or phone.

P. O. Box C9090 Everett, WA 98206

(206) 356-5529

9010A TRAINING WORKBOOK

TABLE OF CONTENTS

SECTION 4 - SUPPORT CIRCUITRY TROUBLESHOOTING Page

SECTION 5 - THE TEST

CRASH COURSE REVIEW TEST

The purpose of this test is to ensure that a certain level of knowledge has been obtained before continuing 9010A training.

Answer the following questions in the space provided. When finished with the last question turn to the review section which follows immediately after the test.

The review section supplies the answers to the test questions and at the same time explains those answers. If questions about the answers still remain, they will be answered in a group discussion when everyone has completed the review.

1. Convert the binary number 100001101 to decimal 269. ✓

2. The hex equivalent of 0011100101101100 is 396C. ✓

3. Convert the hex number A7BF to binary 1010 0111 1011 1111. ✓

4. A binary word contains 16 bits. How many bytes does it contain? 2 ✓

5. a) The largest number that can be represented by a 12-bit binary word is? 4095. ✓

× b) The maximum number of bits that can be represented by a 12-bit binary word is? 12 X 4096.

6. The two main types of semiconductor memories are Static ✓ and Dynamic. ✓

7. The two types of information stored in memory are Programs ✓ and Data. ✓.

× 8. The basic static memory element is called a Flip-Flop ✓.

9. A dynamic storage cell is basically a Capacitor ✓.

10. Periodically recharging a dynamic cell capacitance is called refreshing. ✓.

11. Both static and dynamic cells are volatile ✓ since they will lose data if the power is removed.

12. A 2K X 8 memory contains a total of 16384 ✓ storage cells.

13. With 12-bits, 4096 ✓ memory locations can be addressed.

14. The type of memory that cannot perform a write operation is called a ROM ✓.

15. Which type of semiconductor memory is non-volatile? ROM ✓

16. When an instruction is fetched from RAM or ROM, it passes over the Data bus.

17. Moving all bits of a word from the memory to a register or from one register to another at the same time is referred to as a Parallel data transfer.

18. The 16-bit output of the MAR forms the Address Bus.

Memory Address Register.

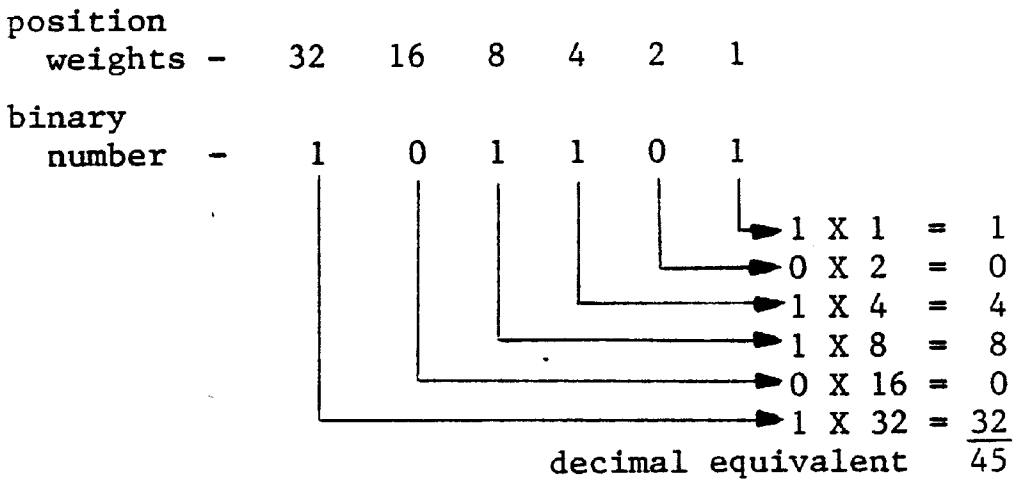
TEST REVIEW

1. The binary system is similar to the decimal system in that the position of a digit in a number determines its weight. The position weights of a binary number are also powers of the number system base. In the binary system, each bit position carries a weight that is some power of 2. These are:

$2^0 = 1$	$2^1 = 2$	$2^2 = 4$	$2^3 = 8$
$2^4 = 16$	$2^5 = 32$	$2^6 = 64$	$2^7 = 128$

By studying the above example it can be seen that each successive higher bit position is 2 times the weight of the previous position.

To convert a binary number to decimal we can simply multiply each bit by its position weight and sum the values to get the decimal equivalent.



Convert the binary number 100001101 to decimal (269).

2. The hexadecimal or hex system uses 16 symbols to represent quantities. These are the numbers 0 through 9 and the letters A through F.

The binary and hex systems are related in that the hex system base is a power of 2. (2^4=16). With 4 bits, 16 different numbers can be defined ($2^N=2^4=16$). The binary and hex equivalents are given below.

Hex and Binary Equivalents

Hex	Binary	Hex	Binary
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

To use the hex system to simplify the representation of binary numbers, you divide the binary number into 4-bit groups starting with the LSB. Then replace each 4-bit group with its hex equivalent.

EXAMPLES:

```
10100101 = 1010/0101
             A    5

1000100110101101 = 1000/1001/1010/1101
                     8    9    A    D
```

The hex equivalent of 0011100101101100 is = (396C) .

3. The letters within a number are a bit weird, but you have to admit that a 4-digit hex number is easier to remember than a 16-bit binary number.

To convert hex numbers into binary, the substitution process is reversed. Each hex digit is replaced by its 4-bit binary equivalent. To convert 4C7E to binary, replace each hex digit with its 4-bit equivalent.

EXAMPLE:

```
   4    C    7    E
0100/1100/0111/1110
```

Convert A7BF to binary. (1010011110111111) .

4. All microcomputers work with a fixed-length binary word. That is, the data words in the computer have a specific number of bits. The most common binary word length in microcomputers is 8 bits. All data storage, processing, manipulation, and transmission is carried out in 8-bit groups. A common characteristic or specification of any computer or microcomputer is its word length. Microprocessors are available in word lengths of 4, 8, 12, and 16 bits.

An 8-bit word is called a byte. A 4-bit word is called a nibble. The number of bits in a binary word determines the maximum decimal value that can be represented by that word. This maximum value is determined with the simple formula:

$$M = 2^N - 1$$

M is the maximum decimal value while N is the number of bits in the word.

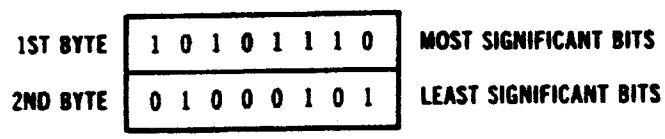
Example: What is the largest decimal number that can be represented with 4-bits?

$M=2^N-1$
$M=2^4-1$ $(2^4=2X2X2X2=16)$

$M=16-1 = 15$

With 4 bits, the maximum possible number is binary 1111 or 15. The maximum decimal number that can be represented with one byte is 255.

An 8-bit word greatly restricts the range of numbers that can be accommodated in a computer. But this is usually overcome by using more than one word to represent a number. For example, two bytes can be used to form a single 16-bit word. The eight most significant bits are contained in one byte and the eight least significant bits in the other byte. This is illustrated below:



Two 8-bit bytes form a single 16-bit word.

A binary word contains 16 bits. How many bytes does it contain? (2).

5. The formula $M=2^N-1$ determines the maximum decimal quantity (M) that can be represented with a binary word of N bits. This value is one less than the maximum number of values that can be represented. The maximum number of values that can be represented (Q) is determined by the formula $Q=2^N$. Again N is the number of bits.

Example: 4 bits

$$Q = 2^4 = 16$$

With 4 bits, 16 values can be represented. These 16 values are 0 through 15 where 15 is the maximum number ($2^4-1=15$). Remember that zero is a valid value.

Maximum Number of States for a Given Number of Bits

Number of Bits (N)	Maximum States (2^N)	Number of Bits (N)	Maximum States (2^N)
1	2	9	512
2	4	10	1024
3	8	11	2048
4	16	12	4096
5	32	13	8192
6	64	14	16,384
7	128	15	32,768
8	256	16	65,536

a) The largest number that can be represented by a 12-bit binary word is? (4095).

b) The maximum number of bits that can be represented by a 12-bit binary word is? (4096).

6. Most computer memories are semiconductor circuits that are used to store the data and instructions in binary form. Semiconductor memories are made up of many individual storage elements or cells, each capable of remembering one bit of information. Each bit can assume either of two states.

   A bit can either be a 1 or a 0. A digital memory circuit capable of assuming two states is used to store each bit.

   There are two basic types of semiconductor storage elements: static and dynamic. In both types of memories, the main component is the metal-oxide silicon field-effect transistor (MOSFET).

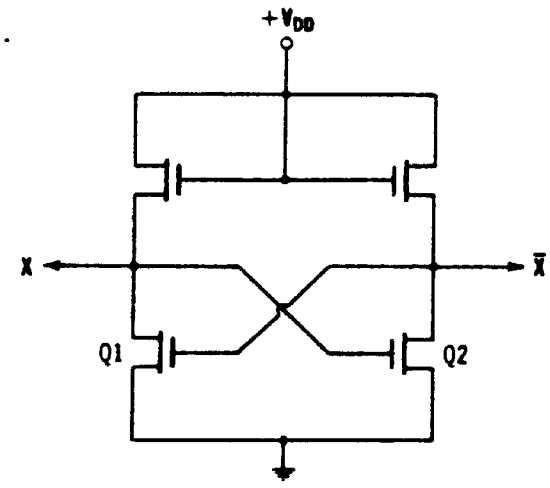
   The two main types of semiconductor memories are (static) and (dynamic).

7. The memory stores data and instructions. The microprocessor uses the information stored in memory for issuing control signals and process data or control external devices.

   The two types of information stored in memory are (instructions) and (data).

8. Static memories use a flip-flop as the storage element. A flip-flop is a digital circuit that can assume either of two states, set or reset. When the flip-flop is set, it is said to be storing a binary 1. When it is reset, it is storing a binary 0.

   A typical static memory element is shown below.



MOSFET flip-flop static storage element.

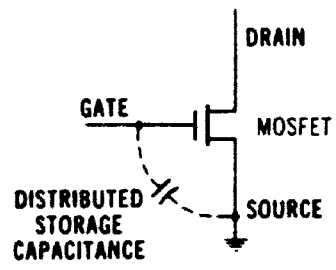
The flip-flop latches into the set or reset state and remains there storing a 1 or 0. This state is static or fixed unless the power is removed or external signals are applied to change it. The bit stored in the flip-flop can be monitored and used elsewhere in the microprocessor.

The basic static memory element is called a (flip-flop).

9. Another type of semiconductor storage element is called a dynamic memory cell. For all practical purposes, the dynamic memory cell is a capacitor. In some dynamic memory cells, the distributed capacitance between the gate and source elements of a MOS field-effect transistor (MOSFET) is used.

   A dynamic storage cell is basically a (capacitor).

10. A typical dynamic memory storage cell is shown below.



MOSFET dynamic memory storage cell.

   When the capacitance is charged, one binary state is stored. When the capacitor is discharged, the other binary state is stored.

   In the above circuit the charge on the capacitor affects the state of the MOSFET. When the capacitance is charged, the MOSFET conducts. When the capacitance is discharged, the MOSFET is cut off.

   Because the capacitance in a dynamic cell is very small, any charge on it will quickly leak off. Of course, this is undesirable since the state of the cell will change and data stored will be lost. To overcome this problem, the charge on the cell must be "refreshed". This refresh operation occurs approximately every two milliseconds.

   Periodically recharging a dynamic cell capacitance is called (refresh).

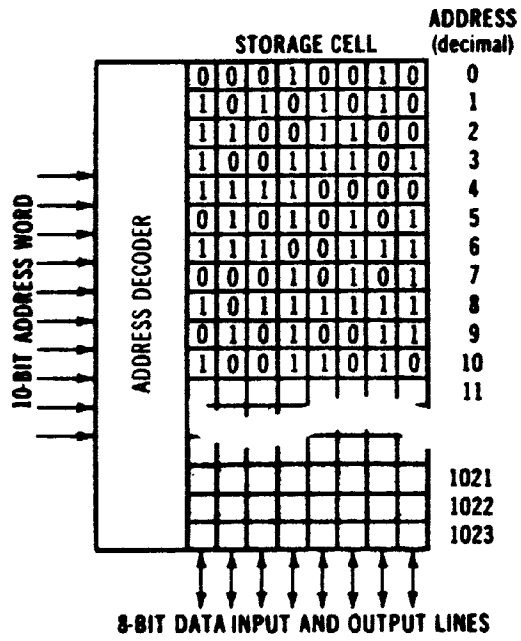
11. An important point to remember is that any stored data is lost if power is removed from a memory cell. Power must be applied if the cell is to retain the data. This applies to both static and dynamic cells. Memory cells with this characteristic are said to be volatile. A nonvolatile memory cell is one that retains the data even if power is removed.

   Both static and dynamic memory cells are (volatile) since they will lose data if the power is removed.

12. Regardless of the type of memory, the cells are arranged in various configurations. For example, a 1024-bit memory can be organized as 1024 1-bit words, (1024X1) or as 256 4-bit words (256X4). 16K-bit memories are also available in several arrangements: 16,384X1, 4096X4, and 2048X8. Note that they all have the same number of bits (16,384), but each is organized differently.

   A 2Kx8 memory contains a total of (16,384) storage cells.

13. A block diagram of a typical microcomputer memory is shown below.



Block diagram of a random-access memory.

Note that with a 10-bit address word, a total of 2^{10}=1024 words can be addressed. The memory locations are numbered 0 through 1023. An 8-bit word can be stored in each location. Eight data lines are used to store or retrieve a word from memory. Note that the address is applied to an address decoder. This is a circuit that looks at the 10-bit binary address, then identifies and enables the single location corresponding to that address.

With 12-bits (4096) memory locations can be addressed.

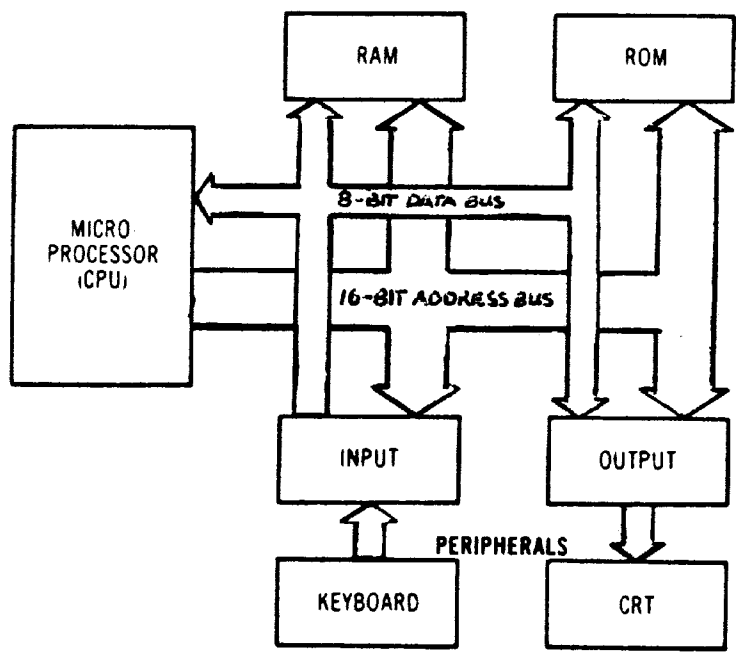
14. Another kind of memory used in microcomputers is the read-only memory or ROM. Data can only be retrieved or read from a ROM. Data cannot be stored in the memory under the control of the CPU. The data and instruction words are permanently stored in the ROM when it is manufactured.

The type of memory that cannot perform a write operation is called a (ROM).

15. The major advantage of a ROM over a RAM or read/write memory is that the ROM is nonvolatile. When power is removed from a ROM, its contents remain undisturbed.

Which type of semiconductor memory is non-volatile? (ROM)

16. Microprocessors usually have two major buses, a data bus and an address bus. These are made available to external circuits. A typical 8-bit CPU has an 8-bit data bus and a 16-bit address bus as shown below.

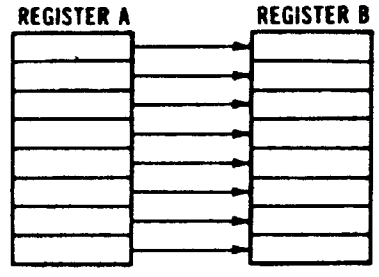


Block diagram of a microcomputer showing data and address buses.

The data bus sends data to and from the CPU, RAM, ROM, and I/O sections. All data transfers between the CPU and memory or I/O sections take place over the data bus. The address bus drives all of the memory and I/O devices.

When an instruction is fetched from RAM or ROM, it passes over the (data) bus.

17. Data transfers in a microprocessor take place in parallel. This means that all bits in a word are transferred simultaneously from one place to another.



A parallel data transfer from register A to register B.

It takes only microseconds or even nanoseconds for all data bits in one register to be moved to another register.

The parallel data transfers take place over a data bus. A bus is simply multiple electrical connections from a source or destination. The figure below illustrates a typical 8-bit data bus in a microprocessor. The eight parallel lines are usually represented by a single wide path.

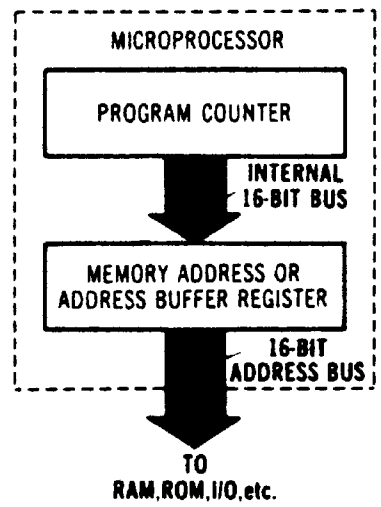
Data bus representation.

Moving all bits of a word from the memory to a register or from one register to another at the same time is referred to as a (parallel) data transfer.

18. The address bus is a unidirectional bus. It handles data transfers in one direction only. That direction is from the CPU to all external circuits.

Address words are produced in the CPU. The program counter generates the address that points to the instruction to be fetched. The content of the PC is transferred over a parallel 16-bit internal address bus to the memory address register or address buffer. The output of the MAR or address buffer is the address bus.

The 16-bit output of the MAR forms the (address bus).



Generation of the address bus in the 8080 microcomputer.

INSTRUMENT DESCRIPTION

The 9010A Microsystem Troubleshooter is a portable service instrument for testing and troubleshooting microprocessor-based equipment. The 9010A provides the following features:

* Keyboard selection of functions and operating modes.

* 32-character display for presentation of test results, operator messages, and prompts.

* Single-keystroke validation of electrical integrity of uP bus.

* LEARN function for mapping UUT address space and identifying RAM, ROM, and I/O.

* Comprehensive, functional testing of RAM, ROM, and I/O.

* Nine troubleshooting functions for troubleshooting on or off the bus.

* On-line programming for development of system test and fault isolation programs.

* Consistent prompts and defaults for easy selection and specification of functions.

* Detailed error messages for locating UUT failures.

* Dual-function stimulus/response probe for generating bus-synchronized pulses or gathering signatures, counting transistions, and detecting logic levels.

* Hexadecimal keyboard for data entry.

* Sixteen 32-bit internal registers for storage and manipulation of data.

* Built-in cassette recorder for nonvolatile storage and transfer of test programs and data on minicassettes.

* Optional RS-232 port for remote communication.

* Optional interface pods for interfacing with the following microprocessors: 8080, 8085, Z80, 6800, 6502, and 9900 - with more interface pods to come.

* UUT (unit under test) microprocessor emulation for execution of UUT program code.

* Rear-panel scope trigger output that is synchronized to UUT microprocessor bus events.

The 9010A consists of the main instrument, the probe, and one of the interface pod options. The probe is included with the purchase of the 9010A, and the interface pod option is purchased separately. The stimulus/response probe and interface pod attach to the main instrument by means of cables. The interface pod contains the microprocessor (uP)-dependent circuitry which allows the 9010A to interface with a particular uP-based UUT. The interface pod allows the 9010A to gain access to the UUT (unit under test) by plugging into the uP socket on the UUT.

Through the uP socket, the 9010A gains access to all hardware connected to the UUT uP bus, including the address, data, status, and control lines. The 9010A can test or exercise all hardware connected to the UUT uP bus. In addition, the 9010A can emulate the UUT uP.

INTRODUCTION

The interaction between the 9010A and the UUT is based on the capability of a uP to read or write data at an address. Unlike other digital troubleshooting or testing equipment, such as logic analyzers or signature analyzers, the 9010A does not gather information in the time-domain. Through the UUT connection at the UUT uP socket, the 9010A actually takes control of the UUT uP bus and allows the operator to specify read and write operations anywhere in the UUT address space. All of the 9010A operations are derived from this fundamental ability to manipulate data at an address.

The 9010A uniformly views all microprocessors that have up to 32 bits of address and data. In addition to the 32 bits of address and data, the 9010A views the UUT uP as having the following:

* Up to 16 status lines that may be read by the operator. Status lines are defined as inputs to the uP, such as reset or interrupt lines.

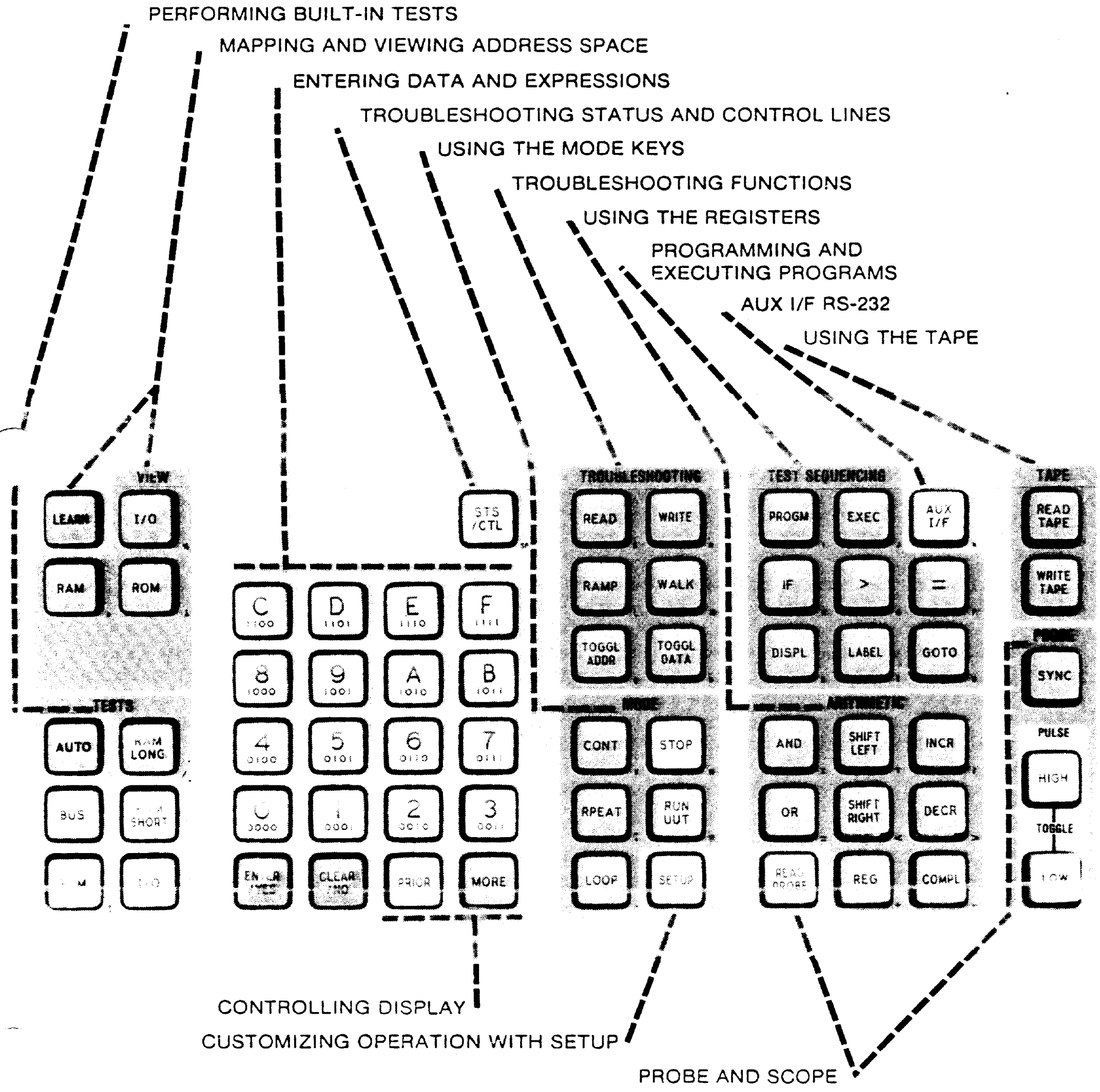
* Up to eight control lines that may be explicitly written by the operator and up to eight more that may not be explicitly written by the operator. Control lines are defined as outputs from the uP to the uP bus, such as read enable or write enable lines.

Information about the actual addresses, data size, status lines, and control lines for each uP is supplied to the 9010A main instrument by the interface pod, and is documented in the appropriate interface pod manual. Note that regardless of the nomenclature used by the manufacturer of the UUT uP, the 9010A regards bit 0 as the low order bit (Least Significant Bit) and bit 31 as the high order bit (Most Significant Bit).

Communication Between The 9010A And The UUT

Access to the UUT is provided by the interface pod and the probe. The probe may be applied to the UUT to stimulate or read data from logic nodes on the UUT.

The main instrument is designed to be used with any uP. The uP-dependent features of the 9010A are provided by the interface pod that is designed for each particular uP. The interface pod contains a uP of the same type that it replaces in the UUT.


PERFORMING BUILT-IN TESTS
MAPPING AND VIEWING ADDRESS SPACE
ENTERING DATA AND EXPRESSIONS
TROUBLESHOOTING STATUS AND CONTROL LINES
USING THE MODE KEYS
TROUBLESHOOTING FUNCTIONS
USING THE REGISTERS
PROGRAMMING AND EXECUTING PROGRAMS
AUX I/F RS-232
USING THE TAPE
VIEW
LEARN
I/O
RAM
ROM
TESTS
AUTO
BUS
STS /CTL
C
D
E
F
8
9
A
B
4
5
6
7
2
3
PRIOR
MORE
TROUBLESHOOTING
READ
WRITE
RAMP
WALK
TOGGL ADDR
TOGGL DATA
MODE
CONT
STOP
RPEAT
LOOP
SETUP
TEST SEQUENCING
PROGM
EXEC
AUX I/F
IF
>
=
DISPL
LABEL
GOTO
ARITHMETIC
AND
SHIFT LEFT
INCR
OR
SHIFT RIGHT
DECR
REG
COMPL
TAPE
READ TAPE
WRITE TAPE
PROBE
SYNC
PULSE
HIGH
TOGGLE
CONTROLLING DISPLAY
CUSTOMIZING OPERATION WITH SETUP
PROBE AND SCOPE

When the main instrument and the interface pod are connected to the UUT, the digital activity provided by the UUT uP is provided by the 9010A. The 9010A can take control of the UUT uP bus to exercise and test ROM, RAM, I/O, and all other circuits interfaced to the bus. The 9010A can also emulate the UUT uP and execute the program code from the UUT memory.

The 9010A can obtain and store information classifying the UUT uP address space into ROM addresses, RAM addresses, and I/O register addresses. The 9010A can determine and store information about the read-write capability of bits in I/O registers, and can compute and store characteristic ROM signatures for blocks of ROM. A ROM signature is a four-digit hexadecimal number that is shorthand representation of the data contained in an area of ROM memory. In addition to computing ROM signatures, the 9010A is also able to compare UUT ROM signatures with expected ROM signatures and report discrepancies.

The UUT clock is supplied to the interface pod uP, which allows the interface pod uP to execute operations on the UUT uP bus at the speed intended by UUT design. The UUT power supplied at the UUT uP socket does not provide operating power for the interface pod uP. The entire interface pod, including the uP, receives operating power from the 9010A main instrument. However, the interface pod monitors the UUT power supplied at the UUT uP socket and reports to the main instrument if the UUT power fails.

In addition to the address, data, status, and control lines, the 9010A also tests the UUT timing and bus handshake lines. The timing and bus handshake lines are tested to ensure that they can be driven by the uP, even though the operator may not explicitly write to these lines.

MAIN INSTRUMENT

The main instrument contains the keyboard and display along with the master logic, control circuitry, and internal memory.

A built-in cassette recorder allows nonvolatile storage and transfer of test programs and other data on minicassettes. A rear panel TRIGGER OUTPUT enables the operator to synchronize an oscilloscope with UUT bus events while testing or troubleshooting with the 9010A. An optional rear panel RS-232 port allows the 9010A to communicate with remote devices.

Operation and programming of the 9010A is accomplished using the front panel keyboard.

The display is capable of displaying up to 32 characters. If single line messages are longer than 32 characters, the first 32 characters of the message are displayed. The remainder of the message may be scrolled horizontally onto the screen using the MORE and PRIOR keys as described later. Multi-line messages may also occur during 9010A operation, and are brought to the display using the same keys.

Five LED annunciators are located to the right of the display. They provide information about the 9010A display and operating modes. When flashing, they indicate the following:

* MORE. Additional message lines exist and may be summoned with the MORE key. The MORE LED is not turned on when the visible line requires scrolling, since the truncated text on the display indicates more text is available.

* LOOPING. The 9010A is recurrently executing some action.

* STOPPED. The STOP key has been pressed and an operation that was being executed has been discontinued.

* PROGMING. The 9010A is in the Programming Mode.

* EXECUTING. The 9010A is in the Executing Mode.

The LOOPING and STOPPED states are mutually exclusive, so the LOOPING and STOPPED annunciators are never flashing simultaneously. Similarly, the Programming and Executing operating modes are mutually exclusive, and the PROGMING and EXECUTING annunciators are never flashing simultaneously.

Interface POD

The main instrument is designed for use with any uP. The uP-dependent features are provided by the interface pod that is designed for each particular uP or uP family.

The interface pod attaches to the main instrument with a 6-foot cable and to the UUT with a short ribbon cable. A plug at the end of the ribbon cable plugs into the uP socket on the UUT. Each interface pod has a decal which provides information about the uP address assignments, pin assignments, and status and control line functions.

The interface pod has a self-test capability. The plug on the end of the interface pod ribbon cable may be inserted into a self-test socket on the interface pod. When the BUS TEST key is pressed, the result of the self test appears on the display. The self-test socket also provides a convenient place to carry and protect the interface pod plug when not installed in a UUT.

Probe

The probe attaches to the main instrument with a 7-foot cable. The probe shell contains two indicator lights, one red and one green, which indicate logic levels and events counted. Both alligator and pin-grabber ground clips are provided with the probe. The ground clip screws into the probe shell and should be used to connect the probe to UUT ground during use.

CAUTION

The probe ground clip MUST be connected to UUT ground when the probe is used as a pulser and connection to UUT ground is recommended at other times.

The probe is used to locate faults on the UUT. It provides both stimulus and response functions that may be selected by the operator. Stimulus functions include the generation of high or low pulses for stimulating particular nodes on or off the UUT uP bus. Response functions include logic level detection, event counting, and signature computation.

The timing of probe stimulus or response functions may be free-running or synchronized with the address-valid or data-valid time periods of the UUT uP. This synchronization capability is especially important while troubleshooting uP systems with multiplexed addresses and data.

9010A Internal Memory

The 9010A provides two areas of internal memory for storage of information useful to the operator. The memory storage is volatile; stored information is lost when power is removed from the 9010A.

Tape-Transferable Memory

The first area of memory consists of 12K bytes of tape-transferable memory; (note that in this manual, a byte is defined as equaling eight bits). Information may be written to or read from this area on minicassette tapes. This information may also be sent to remote devices or received from another 9010A. There are three kinds of information that may be stored in this area:

1. Programs that may be executed by the 9010A.

2. UUT address space descriptors obtained through the LEARN operation or entered by the operator with the VIEW keys.

3. SETUP parameters.

Internal Registers

The second area of memory consists of sixteen 32-bit registers. The registers are labeled 0-9 and A-F. Seven of the registers are dedicated for use by the 9010A software (any may also be used by the operator). Nine of the registers are not used by the 9010A software (non-dedicated), and are available for the sole use of the operator.

Register 0 is a dedicated register that is used by the 9010A software to store probe response data obtained during the Read Probe operation.

Registers A-F are dedicated registers used by the 9010A software to store operating parameters. These registers contain default values supplied by the 9010A when needed during operation.

Registers 1-9 are non-dedicated registers. The operator may use these registers for temporary storage and manipulation of data during operation. The programmer may incorporate the use of these registers into programs as described in the 9010A Programming Manual.

CONNECTING THE INTERFACE POD TO THE 9010A

The interface pod cable plugs into connectors located on the front base of the main instrument under the keyboard. The cable plug has a sliding metal collar which slides over small metal posts on the connector to lock the plug firmly in place.

To connect the interface pod, follow these steps.

1. Tip up the front of the main instrument to allow access to the connector.

2. Slide the metal collar on the cable plug to the left.

3. Plug the cable into the connector and slide the collar to the right so that it locks over the small metal posts on the connector. The plug should now be firmly locked in place.

NOTE

Whenever an interface pod is first connected to the 9010A or after changing pods, it is recommended that the BUS TEST key be pressed. This forces a reset on the interface pod and ensures that necessary initial information is sent from the pod to the 9010A. It is also recommended that the Interface Pod Self Test be performed to ensure that the interface pod is operating properly.

Turn-On Procedure

To turn on the instrument, press the green power switch on the front left corner of the instrument. No calibration or warmup time is required.

When the 9010A is first turned on, the internal volatile memory contains no programs and no address descriptors. Setup parameters assume power-on values. When the 9010A is turned off, any stored programs or UUT address decriptors are lost.

9010A Self Test

When the power is first applied to the 9010A, the following message is displayed:

FLUKE 9000 POWER-UP SELF TEST

The 9010A then performs a self test to verify proper internal operation. If the 9010A is operating properly, the following message is displayed:

FLUKE 9000 POWER-UP OK VER-nn

POWER UP/RESTART ERROR CODES

If the 9010A fails the self test that is initiated on power on the following code will be displayed.

	Hex Code (mm)	Meaning
ERROR	00	No Error
	01	Restart Error
	02	RAM will not accept a Write 1
	04	RAM will not accept a Write Ø
	08	ROM Check Sum Error

The error may also be combinations of these indicating more than one (1) failure.

The letters nn represent a number which corresponds to the version of software that is in the 9010A.

If the 9010A is not operating properly, after a brief interval, the original self-test message is replaced with the following message:

FLUKE 9000 POWER-UP FAIL mm

where the letters mm represent a failure code describing the failure.

The self test may also occur at times other than power-on. The 9010A has a 'watchdog timer' circuit which monitors routine operations in the 9010A and initiates the self test sequence if something appears to be wrong. For example, if a momentary drop in the line power causes the 9010A uP to temporarily malfunction, the monitoring circuit initiates the self test. The self test is identical to the self test performed at power-on, but the messages are slightly different.

The first message displayed is:

FLUKE 9000 RESTARTED SELF TEST

followed by either of these two messages:

FLUKE 9000 RESTARTED OK VER-nn

FLUKE 9000 RESTARTED FAIL mm

After the 9010A is restarted, the memory is initialized as at power-on, and any stored programs or address descriptors are lost.

Interface Pod Self Test

The interface pod also has a self test. Since the interface pods may operate in an electrically hostile environment, it is recommended that the self test be performed whenever an interface pod is first connected to the 9010A to ensure the pod is operating properly. To perform the pod self test, do the following:

1. Insert the plug at the end of the pod ribbon cable into the self test socket on the interface pod, and turn the thumbwheel to lock the plug in the socket.

2. Press the BUS TEST key. The 9010A displays one of the following messages:

   POD SELF-TEST xxxx OK

   POD SELF-TEST xxxx FAIL nn

   The letters xxxx represent the name of the particular interface pod that is in use (for example, 8080 or 9900). When present, the letters nn represent a failure code describing the failure. Refer to the appropriate interface pod manual for an explanation of the failure codes.

UUT PREPARATION

This section describes how to avoid potential operator and instrument hazards when interfacing and operating the 9010A with a UUT.

Avoiding Hazards to the Operator, the UUT, and/or UUT Peripheral Devices

Hazards to the operator caused by the UUT should be minimized by the operator's knowledge of the UUT.

WARNING

TO AVOID ELECTRIC SHOCK AND OTHER HAZARDS TO THE OPERATOR OR DAMAGE TO THE UNIT UNDER TEST (UUT) DO THE FOLLOWING: 1) OBSERVE NORMAL SAFETY PRECAUTIONS FOR OPERATING THE UNIT UNDER TEST. 2) REMOVE POWER FROM THE UNIT UNDER TEST AND DISCONNECT HAZARDOUS UUT ANALOG OUTPUTS OR UUT PERIPHERAL DEVICES BEFORE INSTALLING THE INTERFACE POD IN THE UUT; ALL MOTORS, MECHANICAL ACTUATORS, OR THERMAL PRINTHEADS CONTROLLED BY THE UUT MICROPROCESSOR SHOULD BE DISCONNECTED FROM THE UUT. 3) BE SURE THAT THE UUT DOES NOT FLOAT LOGIC COMMON GROUND MORE THAN +/-30 VOLTS PEAK FROM EARTH GROUND.

When the UUT uP is replaced by the 9010A, the operator is capable of causing the 9010A to perform a write operation at any address on the UUT uP bus. While this is a tremendous asset when testing and troubleshooting the UUT, the operator must take proper precautions to insure that voltage, heat, mechanical equipment or emanations from the UUT will not create a hazardous condition.

For an example of a potential operator hazard, consider a programmable power supply. If the operator selects and specifies the 9010A Learn operation for the entire UUT uP bus, the 9010A performs write and read operations at every location on the bus. If the digital-to-analog converter for the output voltages is connected to the bus, a write operation at the d/a converter address could produce unexpected and dangerously high voltages at the output terminal.

For an example of the potential hazard to a UUT or UUT peripheral device, consider a uP-controlled disk drive. If the operator causes the 9010A to perform write operations on the UUT uP bus without removing the output to the disk drive, the 9010A could destroy information stored on the disk by writing random data on the disk.

9010A Overload Protection

The 9010A is designed to operate in electrically hostile UUT environments and to withstand assaults commonly encountered by test equipment, such as electromagnetic interference. The only direct electrical connection between the 9010A and the UUT (except for the probe) is the interface pod which plugs into the UUT uP socket. The interface pod inputs are protected to withstand +12V to -7V,

which is more than adequate for potential voltages at the UUT uP socket. This protection generally prevents damage to the UUT or interface pod due to improper insertion of the interface pod plug in the UUT, or installation of the wrong type of interface pod. The probe can withstand inputs of +/-30V dc.

Installation of the Pod Into the UUT

To install the interface pod into the UUT, use the following procedure:

1. Remove power from the UUT. (The 9010A power may be left on.)

2. Disconnect UUT analog outputs or potentially hazardous UUT peripheral devices.

3. Gain access to the UUT uP socket and remove the UUT uP. For complex instruments or machinery, this may require mounting the printed circuit board on an extender board or removing the printed circuit board and placing it on a table with connections to a power supply.

4. Turn the pod self-test socket thumbwheel to release the pod plug, and remove the pod plug from the self-test socket.

5. Align the pod plug properly with the UUT uP socket and insert the pod plug into the uP socket. The slanted corner of the pod plug should be aligned with pin 1 of the uP socket.

CAUTION

Be sure that 9010A power is on before turning on UUT power in order to activate input protection circuits within the pod.

6. Apply power to the UUT.

OPERATING MODES

During operation, the 9010A may be in any one of three operating modes:

1. Immediate Mode

2. Programming Mode

3. Executing Mode

The 9010A may operate in only one of the three operating modes at any time. Operation in the Programming Mode or Executing Mode is indicated by the flashing PROGMING or EXECUTING annunciator at the far right of the display. When the 9010A is operating and neither the PROGMING nor the EXECUTING annunciator is on, the 9010A is in the Immediate Mode.

Immediate Mode

The operator selects a test, operation, or troubleshooting function by pressing the appropriate key, such as RAM SHORT, LEARN, or READ. Pressing the key initiates the specification process, in which the operator supplies the additional parameters required by the 9010A before the test, operation, or function can be performed. Parameters required during specification may include such things as addresses, data, ROM signatures, or bit numbers. For example, if the Read function is selected, the address where the Read function is to be performed must be specified.

In the Immediate Mode after the specification is complete, the 9010A performs the test, operation, or function as specified. At any point during a specification the operator may abort the current specification and initiate a new specification by pressing the appropriate key, such as RAM SHORT or LEARN.

Unless otherwise specified, the operation described in this manual takes place in the Immediate Mode.

Programming Mode

The 9010A operates in this mode during the creation or editing of programs. The programmer causes the 9010A to enter the Programming Mode by creating or opening a program. Notice that in contrast to the Immediate Mode, the selection and specification of a test, operation, or troubleshooting function is not performed after the specification is completed. Instead, the specification is stored as a step in a program. The 9010A does not perform the specified action until the program is executed in the Executing Mode.

When the program is closed by the programmer, the 9010A returns to the Immediate Mode.

Executing Mode

The 9010A operates in this mode during the execution of programs and performs actions as specified by the program.

The operator causes the 9010A to enter the Executing Mode by specifying the execution of a program. After the program execution is complete, the 9010A returns to the Immediate Mode.

PROMPTS

One feature which makes the 9010A easy to operate is the system of clear, consistent, and understandable prompts. Prompts are requests by the 9010A for more information from the operator. This section describes the information requested by prompts and describes when prompts are encountered during operation.

Information Requested By Prompts

Information requested by prompts may be data, such as an address or numerical expression, or it may be a YES or NO response to a question.

Prompts for data are indicated by a blinking cursor that appears on the display next to the function selected. For example, when the Read troubleshooting function is selected, the 9010A displays the message READ @ _. The underscore (_) is the blinking cursor that serves as a prompt for the operator to define the address location where the Read operation is to take place.

Prompts for a YES or NO response include a question and end with a question mark. For example, if the operator specifies the Read function at address 4D77 and the 9010A detects a bad power supply at the UUT uP socket while performing the Read operation, it displays the following message:

BAD PWR SUPPLY @ 4D77-LOOP?

This message asks the operator if the 9010A is to "loop on the error". Pressing the YES key or the LOOP key causes the 9010A to loop on the error. Pressing the NO key or the CONT key causes the 9010A to ignore the detected error.

Prompts During Specification

When a 9010A function is selected by pressing the appropriate key, the 9010A usually requires the operator to supply additional operating parameters or specifications for the function before the 9010A can perform the operation on the UUT. The 9010A prompts the operator for the specifications in the order the information is required. For example, when the ROM Test is selected, three specifications must be provided:

1. The first address of the block of addresses on which the ROM Test is to be performed.

2. The last address of the block of addresses on which the ROM Test is to be performed.

3. The ROM signature that is to be compared with the ROM signature computed from the address block.

First, the 9010A prompts for the first address. When an address has been entered by the operator, the 9010A prompts for the second address. When the second address has been entered by the operator, the 9010A prompts for the ROM signature. When the ROM signature has been entered by the operator, the 9010A performs the operation as specified.

The following exercise shows the steps necessary for specifying the ROM Test over the address block 2000-3FFF with ROM signature 895A, and the associated displays and prompts.

Press	Display	Comment
ROM TEST	ROM TEST @ _	Prompt for first address.
2000 ENTER	ROM TEST @ 2000-_	First address has been entered, now prompting for second address.
3FFF ENTER	ROM TEST @ 2000-3FFF SIG_	Second address has been entered, now prompting for ROM signature.
895A ENTER	ROM TEST @ 2000-3FFF SIG 895A	Specification complete.

For another example, consider the Write function. When the Write function is selected, the 9010A prompts for two parameters:

1. The address where data is to be written.

2. The data that is to be written.

The following exercise shows the steps necessary to select the Write function and specify the data AA to be written at the address CØ7F.

Press	Display	Comment
WRITE	WRITE @ _	Prompt for address.
CØ7F ENTER	WRITE @ CØ7F = _	Address has been entered, now prompting for data.
AA ENTER	WRITE @ CØ7F = AA	Specification complete.

Understanding "Wait" After Specification Is Complete

After the operator completes the specification of the 9010A action, the 9010A appends the word WAIT to the final specification on the display and begins performing the action as specified. The word WAIT assures the operator that the 9010A is performing the action. Most of the troubleshooting functions (such as Read or Write) are performed so quickly that the final specification (including WAIT) is only seen briefly.

Audible Feedback

The 9010A also provides audible feedback in the form of a beep when the operator presses a key at a time when it is not allowed. For example, if the 9010A is in the Immediate Mode and the GOTO key is pressed, the 9010A emits a beep which tells the operator that the function cannot be selected. The GOTO key may only be pressed while the 9010A is in the Programming Mode. The audible beep may also be selected by the programmer for insertion in programs.

CUSTOMIZING OPERATION WITH SET-UP

Although many of the uP-dependent requirements for each microsystem are provided by the design of the interface pod, each UUT may have particular operating requirements beyond those met by the interface pod design. The Setup function allows the operator to select specific operating features and parameters to meet particular UUT requirements.

The Setup Menu

The Setup function does not directly cause any actions to be performed on the UUT that are normally visible to the operator. Pressing the SETUP key invokes a menu comprised of a sequence of one-line messages. The Setup messages are listed in the order in which they are scrolled with the MORE key. When the Setup menu is first invoked by pressing the SETUP key, the 9010A enters the Setup menu at the point where it last exited. Whenever a Setup message is present on the display, pressing SETUP brings the first message in the table to the display. The circular Setup menu may be scrolled forward or backward with the MORE and PRIOR keys.

There are two types of Setup messages. One type involves the reporting of UUT errors or the enabling of uP lines. This type of message ends with the word YES or NO, and may be changed by pressing the YES or NO key.

The other type of Setup message involves the specification of operating parameters. This type of message ends as follows: CHANGE?

The following exercise shows the Setup messages that occur. The table lists the messages and a description of these messages.

PRESS	COMMENTS
SETUP	Observe the display
MORE	Continue pressing MORE so as to view all Setup messages

The actual number of Setup messages varies, depending on the type of interface pod that is connected to the main instrument. Each uP may have up to eight lines that may be enabled or disabled with a Setup parameter. Typically the lines are such things as Ready, Hold, Wait, or Bus Request. Each line that may be enabled or disabled has a separate message in the Setup menu.

SETUP MESSAGES

DISPLAY MESSAGE	DESCRIPTION
SET-TRAP BAD PWR SUPPLY? YES SET-TRAP ILLEGAL ADDRESS? YES SET-TRAP ACTIVE INTERRUPT? YES SET-TRAP ACTIVE FORCE LINE? YES SET-TRAP CTL ERR? YES SET-TRAP ADDR ERR? YES SET-TRAP DATA ERR? YES	The first seven messages correspond to the seven UUT system errors. If "YES" is selected, the UUT system error is reported to the operator. If "NO" is selected, the UUT system error is not reported to the operator.
SET-ENABLE xxxx? Yes SET-ENABLE WAIT? YES	This message appears for each uP line that may be anabled or disabled. The letters xxxx correspond to the name of the line, such as ready or wait. The actual names and number of lines are documented in the appropriate interface pod manual.
SET-BUS TEST @ FFFF-CHANGE?	When the BUS TEST is performed, testing of data lines occurs at the address listed.
SET-RUN UUT @ 0000-CHANGE?	When the address for the Run UUT operation is allowed to default, this address is used.
SET-TIMEOUT 100-CHANGE?	The parameter following the word TIMEOUT represents a count of how long the 9010A waits before timing out on an interface pod operation. The parameter may be any decimal number between 0 and 60,000.

SET-EXERCISE ERRORS? YES	If "YES" is selected, the 9010A displays detected error messages and prompts the operator to loop on the errors. If "NO" is selected, the errors are not reported to the operator, but error messages are transmitted to the RS-232-C if it is connected (without the -LOOP? portion of the message).
SET-BEEP ON ERR TRANSITION? YES	The "YES" enables the audible beep which sounds whenever an error is detected and reported. The beep also sounds whenever the error is removed.

The following Setup parameters relate to the operation of the options AUX I/F. If the 9010A detects at power-on that the optional AUX I/F is not installed, the following Setup parameters do not appear in the Setup menu. For a complete description of the following Setup parameters and the AUX I/F, refer to Section 5 of the operator's manual.

SET-STALL 13-CHANGE?	Any hexadecimal value from 0 to FF may be entered. The corresponding ASCII character is the stall character.
SET-UNSTALL 11-CHANGE?	Any hexadecimal value from 0 to FF may be entered. The corresponding ASCII character is the stall character.
SET-NEWLINE 0000000A-CHANGE?	This is the terminator sequence and timing delay between lines.
SET-LINESIZE 79-CHANGE?	This is the maximum line length for data transmission from the 9010A. The length may be any decimal value from 10 to 255.

VIEW KEYS

Troubleshooting any microsystem requires information about the UUT address space. This section describes how the Learn operation may be used to obtain UUT address space information and how the View operations may be used to examine the information.

Learn Mapping Key not a view function.

If UUT address space information is not known or is incomplete, the Learn operation may be used to explore and map the UUT memory. The Learn operation is typically performed on a properly operating UUT to obtain valid reference data for use in testing a suspect UUT. For the experienced operator, the Learn operation may also provide clues about the cause of faults when performed on a failing UUT.

The Learn operation tests the UUT address locations and identifies the addresses of blocks of ROM, RAM, and I/O registers. For each block of ROM, the operation computes a ROM signature. For each block of I/O registers, the operation computes an I/O bit mask which indicates which bits in I/O have read/write capability.

During the Learn operation, the 9010A writes and reads data patterns to 64-byte blocks (1 byte = 8 bits) for the entire address range of the uP. Based on the data read, the 9010A uses the following criterion to determine whether addresses are ROM, RAM, I/O or unassigned:

* ROM: The addresses cannot be written to, and all address bits are fully decoded.

* RAM: All bits of all addresses for the 64-byte block are write-readable, and all address bits are fully decoded.

* I/O: At least one bit of an address is write-readable, but any other condition for determining RAM fails.

* Unassigned: Anything not identified as ROM, RAM, or I/O.

Note that although the Learn algorithm which obtains the descriptors is very powerful, it does have some limitations. For example, it cannot identify bank-switched RAM or I/O which does not have at least one bit that is read-writable. The Learn operation is not intended to be a replacement for schematics or other existent UUT documentation. For this reason, it is recommended that the operator enter address descriptors with the VIEW keys if the descriptors are known.

Specifying the Operation

The only specification required for the Learn operation is the UUT address space on which the operation is to be performed. The operation may be performed on all or part of the UUT address space. Note that the RPEAT and LOOP keys do not affect the operation.

To specify the entire uP address space for the Learn operation, we would do the following. The 9010A would take approximately 1.5 hours for a complete LEARN. Because of this, the example below will not be performed.

1. Press LEARN. The 9010A displays a prompt for the first address by displaying the following message:

   LEARN @ _

2. Press ENTER. The 9010A begins execution on the entire UUT address space specified by the interface pod after displaying the following message:

   LEARN

To do a LEARN on known address space perform the following:

1. Press LEARN. The 9010A displays a prompt for the first address by displaying the following message:

   LEARN @ _

2. Key in the first address, ØØØØ, and press ENTER. The 9010A displays the first address and prompts for the second address with the following message:

   LEARN @ ØØØØ-_

3. Key in the second address, FFFF, and press ENTER. The 9010A begins execution on the specified address space after displaying the following message:

   LEARN @ ØØØØ-FFFF

After the address specification is complete, execution of the Learn operation proceeds. Any presently stored address space descriptors whose addresses are within the address range of the Learn operation are deleted from 9010A memory.

Because of the tremendous number of operations involved, the execution of the operation may take a considerable length of time (typically 10-100 minutes for a common 8-bit uP in a typical UUT). The length of time varies considerably depending on the relative proportion of RAM, ROM, I/O and unassigned memory, as well as the size of the specified address block. To provide assurance that the operation is executing properly, the 9010A appends the following message to the test on the display:

NOW aa00

The digits represented by aa00 are the hexadecimal digits for the address where the operation is presently taking place. The display is updated every 256 locations.

After the Learn operation has explored all of the addresses specified, the following message is briefly displayed:

LEARN @ aaaa-aaaa WAIT

While the WAIT message is on the display, the operation compiles the information that has been obtained and composes the descriptors for the address space that has been explored. After the information has been compiled, the 9010A replaces the word WAIT with the words OK or FAIL.

The operator may terminate execution of the Learn operation by pressing the STOP key. Note that if the operator terminates execution before the operation is complete, address space information is incomplete. For example, ROM signatures are not compiled until the WAIT message is displayed.

View

The three VIEW keys (VIEW RAM, VIEW ROM, and VIEW I/O) allow the operator to view and edit the UUT memory map that was obtained during the LEARN.

Viewing The Memory Map

The address space descriptors for RAM, ROM, or I/O are placed on the display when the operator presses the appropriate VIEW key. Each key places the first descriptor for that type of address block on the display. The flashing MORE annunciator indicates that more descriptors of the type being viewed are available. Other descriptors of the same type may be brought to the display with the MORE and PRIOR keys.

EXERCISE: Press the ROM VIEW key and fill in the below space with ROM address and SIG data.

Press	Display
ROM VIEW	ROM @ 0000-BEFF SIG 0679
	ROM @ BFC0-BFFF SIG 8100

The first descriptor in the list appears on the display. The subsequent descriptors are brought to the display with the MORE key, and preceding descriptors with the PRIOR key. The RAM VIEW and I/O VIEW keys operate in a similar manner.

EXERCISE: Press the RAM VIEW key and record the information in the same manner as ROM.

RAM @ C000-FFFF

If no address descriptors for the type requested (ROM, RAM, I/O) are present in 9010A memory, the 9010A provides the appropriate message, such as the following:

NO ROM INFO

EXERCISE: Press the I/O VIEW key and record the information in the same manner as ROM.

NO IO INFO between. 0000-FFFF

TEST KEYS

There are five built-in tests in the 9010A to automatically test the electrical integrity of the UUT uP bus, the read-write capability of I/O registers, the data in ROM, and the functionality of RAM. These tests are Bus Test, ROM Test, I/O Test, RAM Short, and RAM Long. In addition, Auto Test provides a combination of four of the other tests.

Bus Test

Bus Test is a test of the electrical integrity of UUT control, address, and data buses. Bus Test identifies control lines that are not drivable, as well as address lines that are tied high, low, or tied together, and data lines that are tied high, low, or tied together.

Specification and Performance of Bus Test

To specify the Bus Test, press the BUS TEST key. No other entries are required.

As soon as the BUS TEST key is pressed, the 9010A begins performing the test and displays the following message.

BUS TEST WAIT

After the Bus Test is completed, the 9010A replaces WAIT with an OK or FAIL.

If a Bus Test Error occurs refer to section 4H-4 of the operators manual for the definition of that error. Listed below are types of errors the Bus Test will detect.

1. CONTROL LINES NOT DRIVABLE.
2. ADDRESS LINES TIED HIGH OR LOW.
3. ADDRESS LINES TIED TOGETHER
4. DATA LINES TIED HIGH OR LOW.
5. DATA LINES TIED TOGETHER.

Perform a BUS TEST on the UUT and observe the operation.

Specifying The Addresses For Performing A Test

When any one of four of the built-in tests (I/O Test, ROM Test, RAM Short, and RAM Long) are selected by the operator, the operator may specify that the test be performed at a single address or a block of addresses. When the operator specifies a block of addresses, the first (lower) address is specified, and then the second (upper) address is specified.

ROM Test

Specification of ROM Test

To select the ROM Test, do the following:

1. Key in the first and second addresses.

   The 9010A accepts the addresses and prompts for the ROM signature for the address block with the following message:

   ROM TEST @ aaaa-aaaa SIG _

2. Key in the ROM signature nnnn and press ENTER.

A ROM signature is entered in hexadecimal and must be less than or equal to FFFF. The 9010A accepts the ROM signature, displays the following message, and begins performing the test. Note that the first word in the message is truncated.

M TEST @ aaaa-aaaa SIG nnnn WAIT

The following exercise shows the proper steps for specifying the ROM Test at addresses 4000-5FFF with ROM signature F3AA and the corresponding displays. Perform the exercise and verify that you obtain the same results as listed below.

Press	Display
ROM TEST	ROM TEST @ _
4000 ENTER	ROM TEST @ 4000-_
5FFF ENTER	ROM TEST @ 4000-5FFF SIG _
F3AA ENTER	M TEST @ 4000-5FFF SIG F3AA WAIT

Performance Of ROM Test

After the specifications are complete, the 9010A begins performing the ROM Test. After the ROM Test is completed, the 9010A appends an OK or FAIL to the message on the display.

The 9010A computes the ROM signature for the address block specified and compares it with the specified ROM signature. If the computed and specified ROM signatures are not equal, the 9010A places the first line of the following two-line message on the display. The message may be scrolled using the MORE and PRIOR keys.

ROM ERR @ aaaa-aaaa-LOOP?
SIG WAS mmmm NOT nnnn-LOOP?

The second line in the display shows both the computed and specified ROM signatures which the 9010A found unequal. If the operator exercises the error by pressing the YES or LOOP keys, the 9010A displays the following message:

ROM ERR @ aaaa-aaaa

While the error is exercised, the 9010A continues to compute the ROM signature for the address block specified and compare it with the specified signature. If the cause of the error is removed and the error is no longer detected, the following message is displayed:

ROM OK @ aaaa-aaaa

Example Of ROM Test Error

Perform the following steps:

The operator selects the ROM TEST and specified addresses 4000-43FF with ROM signature 2D73. The 9010A begins execution and displays the following message:

M TEST @ 4000-43FF SIG 2D73 WAIT

After the 9010A computes the ROM signature, it detects an error and places the first line of the following two-line message on the display:

ROM ERR @ 4000-43FF-LOOP?
SIG WAS AAFF NOT 2D73-LOOP?

The operator chooses to loop on the error and presses the YES key. The 9010A begins looping and displays the following message:

ROM ERR @ 4000-43FF

The operator manipulates the UUT in some way (for example, by cleaning or applying pressure to chips), and the error is no longer detected. The 9010A displays the following message:

ROM OK @ 4000-43FF

If the operator chooses to press the NO key the 9010A will essentially ignore the error and go to the Immediate mode.

RAM Short

To ensure that all RAM failures are identified and yet optimize test times, the 9010A provides two tests for RAM, RAM Short and RAM Long. As the name implies, RAM Short is a shorter, faster test than RAM Long. RAM Short is designed to quickly identify common RAM failures such as address decoding errors or bits that are not read-writable.

The specification and operation of the RAM SHORT will be descirbed now. ROM LONG will not be exercised in the operation section.

Specification of RAM Short

To select the RAM Short test, key in the first and second addresses. The 9010A accepts the addresses and displays the following message:

RAM SHORT @ aaaa-aaaa WAIT

The following exercise shows the proper steps for specifying RAM Short at addresses CØØØ-BFFF and the corresponding displays. Perform the exercise.

Press	Display
RAM SHORT	RAM SHORT @ _
CØØØ ENTER	RAM SHORT @ CØØØ-_
BFFF ENTER	RAM SHORT @ CØØØ-BFFF WAIT

Performance of RAM Short

After the specifications are complete, the 9010A begins performing RAM Short. RAM Short is executed on each address block specified. After RAM Short is completed, the 9010A replaces the word WAIT in the message on the display with OK or FAIL.

Performance of RAM Short consists of three phases. Each phase performs unique operations during testing and looping on errors and has unique error messages. The operations and messages for each phase are as follows. Again, for the definitions to the error codes refer to the operators manual section 4H-16.

1. A test of the read-write capability of every data bit of every address location in the address block.

2. A test for data lines tied together.

3. A test for address decoding errors within the address block.

Auto Test

The Auto Test is a combination of four other tests: Bus Test, ROM Test, RAM Short, and I/O Test. Auto Test is selected by pressing the AUTO TEST key. During execution, the 9010A displays the following message (unless there are error messages):

AUTO TEST WAIT

After the test is completed, the 9010A replaces WAIT with an OK or FAIL. The actions performed by the 9010A are identical to the actions specified previously for the sequence of four tests that comprise Auto Test. The tests are performed in the order listed. The specifications for ROM Test, RAM Short, and I/O Test are the default specifications supplied by the UUT address descriptors. The error messages, the actions taken while looping on errors, and the associated messages are identical to those in the individual tests. Note that the operation of the RPEAT, LOOP, and STOP keys applies to the entire sequence of 9010A actions specified for Auto Test.

TROUBLESHOOTING FUNCTIONS

The troubleshooting functions allow the operator to concentrate the scope of troubleshooting activity to the stimulation or monitoring of particular address locations or bits on the UUT uP bus.

Six of the functions are selected by single keystrokes. The keys are listed as follows: READ, WRITE, RAMP, WALK, TOGGL, ADDR and TOGGL DATA. Three of the troubleshooting functions are selected by the combined use of the STS/CTL (Status/Control) key and three of the function keys. The STS/CTL troubleshooting functions are listed as follows: READ STS, WRITE CTL, and TOGGL DATA CTL.

The specification and execution of all the troubleshooting functions are described in the following paragraphs. There are no error messages unique to the troubleshooting functions, although the timeout and UUT system errors may be detected and reported as usual. (The one exception to this is Read STS, during which only timeout errors may be reported, and not UUT system errors.) Note that the RPEAT, LOOP, and STOP keys may be used with any of the functions.

Read

The Read function causes the 9010A to read the data at an operator-specified location and display the contents.

To select the Read function, do the following:

1. Press READ. The 9010A prompts for the address where the read operation is to take place by displaying the following message:

   READ @ _

2. Key in the address CØØØ and press ENTER. The 9010A displays the following meassage:

   READ @ CØØØ WAIT

After the specification is complete, the 9010A reads the data at the specified address and displays the data hh (in hexidecimal) along with an OK or FAIL as follows: (hh corresponds to the data that is in the address at that time)

```
READ @ CØØØ = hh OK
READ @ CØØØ = hh FAIL
```

Read STS

The Read STS function causes the 9010A to read the status lines on the uP and display their values.

To select the Read STS function, press the Read key and then the STS/CTL key. The 9010A displays the following message:

```
READ @ STS WAIT
```

The 9010A reads the status lines and places the following message on the display:

```
D @ STS = nnnn nnnn nnnn nnnn OK
```

Note in the message listed that the 9010A truncates the message on the left so that the pertinent portion of the message is displayed. The binary string nnnn nnnn nnnn nnnn represents the 16 possible status lines. A one corresponds to lines that are detected high, and a zero corresponds to lines that are detected low. A microsystem may have 16 or fewer status lines. The status lines for each particular microsystem are documented in the appropriate interface pod manual. The binary strings representing the status lines are always displayed in groups of four. If a uP has 9-12 meaningful status lines, only three groups of four digits are displayed. If a uP has eight or fewer meaningful status lines, only two groups of four digits are displayed. Refer to the appropriate interface pod manual for the meaning of each bit in the string.

Write

The Write function causes the 9010A to write operator-specified data to an operator-specified address.

To select the Write function, do the following:

1. Press the WRITE key. The 9010A prompts for the address to be written to by displaying the following message:

   ```
   WRITE @ _
   ```

2. Key in the address CØØØ and press ENTER. The 9010A prompts for the data to be written by displaying the following message:

   ```
   WRITE @ CØØØ = _
   ```

3. Key in the data FF and press ENTER. The 9010A displays the following message:

   ```
   WRITE @ CØØØ = FF WAIT
   ```

After the specification is complete the 9010A writes the specified data to the specified address. When the operation is complete, the 9010A replaces WAIT with an OK or FAIL.

WRITE CTL

The Write CTL function causes the 9010A to write operator-specified control lines to the operator-specified logic levels.

To select the Write CTL function, do the following:

1. Press the WRITE key and then the STS/CTL key. The 9010A prompts for the binary string specifying the values to write to the control lines by displaying the following message:

   WRITE @ CTL = _

2. Key in the desired control information in binary and press ENTER. The 9010A accepts any binary value from 0 to 11111111. The 9010A displays the following message:

   WRITE @ CTL = bbbbbbbb WAIT

The binary string bbbbbbbb corresponds to the eight possible UUT control lines. The 9010A forces control lines represented by a one high, and forces control lines represented by a zero low. The control lines for each microsystem are documented in the appropriate interface pod manual. Note that the 9010A supplies the value zero for any lines not specified by the operator. Values that are specified for nonexistent control lines are ignored by the 9010A when the function is performed.

After the specification is complete, the 9010A writes the UUT control lines as specified and replaces WAIT with an OK or FAIL.

Ramp

The Ramp function causes the 9010A to perform a series of write operations, beginning with all data bits equal to zero, and increasing until all data bits equal one.

To select the Ramp function, do the following:

1. Press the RAMP key. The 9010A prompts for the address where the operation is to be performed by displaying the following message:

   RAMP @ _

2. Key in the address 1ØØ8Ø and press ENTER. The 9010A displays the following message:

   RAMP @ 1ØØ8Ø WAIT

After the specification is complete, the 9010A performs a series of write operations at address 1ØØ8Ø. The write operations begin with all data bits equal to zero. The value of the data for each successive write operation increases by one until all data bits are equal to one.

Walk

The Walk function causes the 9010A to perform a series of write operations at an operator-specified address. First the 9010A writes operator-specified data, then rotates the data one bit, and writes the new data. This process continues until the data is rotated around completely.

To select the Walk function, do the following:

1. Press the WALK key. The 9010A prompts for the address where the operation is to be performed by displaying the following message:

   WALK @ _

2. Key in the address 1ØØ8Ø and press ENTER. The 9010A prompts for the data for the Walk function by displaying the following message:

   WALK @ 1ØØ8Ø = _

3. Key in the data AA in hexadecimal and press ENTER. The 9010A accepts hexadecimal values for the data specification. The 9010A displays the following message:

   WALK @ 1ØØ8Ø = AA WAIT

After the specification is complete, the 9010A performs the write operations associated with the Walk function at the address 1ØØ8Ø. First the 9010A writes AA. Then it rotates the data to the right, wrapping the lowest bit around to the highest bit position, and writes the resulting data. This rotate and write process continues until the bits have been rotated around to their original position.

Toggl Addr

The Toggle Address function causes the 9010A to toggle an operator-specified address bit from one logic state to the other. To select the Toggle Address function, do the following:

1. Press the TOGGL ADDR key. The 9010A prompts for the first address where the operation is to be performed by displaying the following message:

   ATOG @ _

2. Key in the address and press ENTER. The 9010A prompts for the address bit that is to be toggled by displaying the following message:

   ATOG @ 1FFF BIT _

3. Key in the address bit 2 that is to be toggled. The 9010A accepts any decimal value from 0 to 31 for the address bit. The 9010A displays the following message:

   ATOG @ 1FFF BIT 2 WAIT

When the specification is complete, the 9010A performs two read operations, each at a different address. First the 9010A reads the data at the address specified. Then the 9010A toggles the address bit specified, changing it to the opposite logic level, and reads the data at the resultant address. Note that the data that is read is not presented on the display. While the 9010A performs the two read operations, the 9010A displays the following message:

```
ATOG @ 1FFF BIT 2 WAIT
```

When the two read operations are completed, the 9010A replaces WAIT with an OK or FAIL.

The following example shows the proper steps for specifying TOGGL ADDR at address 1A00 with address bit 3 toggle, and the corresponding display message:

Press	Display
TOGGL ADDR	ATOG @ _
1A00 ENTER	ATOG @ 1A00 BIT _
3 ENTER	ATOG @ 1A00 BIT 3 WAIT

At this point the 9010A reads the data at address 1A00. Then the 9010A toggles address bit 3, creating the new address 1A08. This is illustrated in binary as follows:

```
1A00 = 0001 1010 0000 0000
1A08 = 0001 1010 0000 1000
                      ^
                      bit 3
```

After creating the address 1A08, the 9010A reads the data at 1A08.

✶ Note that the Toggle Address Function is particularly useful in tracing and troubleshooting address decode problems.

Toggl Data

The Toggle Data function causes the 9010A to toggle an operator-specified data bit from one logic state to the other.

To select the Toggl Data function, do the following:

1. Press the TOGGL DATA key. The 9010A prompts for the address where the operation is to be performed by displaying the following message:

   ```
   DTOG @ _
   ```

2. Key in the address and press ENTER. The 9010A prompts for the data that is to be written by displaying the following message:

   ```
   DTOG @ 1ØØ8Ø = _
   ```

3. Key in the data and press ENTER. The 9010A prompts for the bit number of the data bit that is to be toggled by displaying the following message:

   DTOG @ 1ØØ8Ø = EC BIT _

4. Key in the data bit dd in decimal and press ENTER. The 9010A accepts any decimal value from 0 to n-1 where n equals the number of data lines for the uP. The 9010A displays the following message:

   DTOG @ 1ØØ8Ø = EC BIT 1 WAIT

After the specification is complete, the 9010A performs two write operations at the same address. First the 9010A writes the specified data to the specified address. Then the 9010A toggles the specified data bit, changing its logic level to the opposite value, and writes the resulting data to the specified address. While the 9010A performs the two write operations, the following message is displayed:

DTOG @ aaaa = hhhh BIT dd WAIT

When the two write operations are completed, the 9010A replaces WAIT with an OK or FAIL. After the Toggle Data function is performed, the toggled data (from the second write operation) remains in the address where the function is performed.

The following example shows the proper steps for the specification of Toggl Data for a 8-bit uP, and the corresponding display messages. TOGGL DATA is specified with address D407, data 10, and data bit 5 to be toggled:

Press	Display
TOGGL DATA	DTOG @ _
D407 ENTER	DTOG @ D407 - _
10 ENTER	DTOG @ D407 = 10 BIT _
5 ENTER	DTOG @ D407 = 10 BIT 5 WAIT

At this point the 9010A writes 10 to address D407. Then the 9010A toggles data bit 5, creating the new data value 30. This is illustrated in binary as follows:

```
10 = 0001 0000
30 = 0011 0000
       ^
       bit 5
```

The 9010A writes 30 to address D407.

✶ Note that the Toggle Data function is particularly useful in tracing and troubleshooting data lines up to and beyond peripheral IC's.

Toggl Data Ctl

The Toggle Data Control function causes the 9010A to toggle an operator-specified control line from one logic state to the other.

To select the Toggle Data Control function, do the following:

1. Press the TOGGL DATA key and then the STS/CTL key. The 9010A prompts for the binary string specifying the values to write to the control lines by displaying the following message:

   DTOG @ CTL = _

2. Key in the desired control information in binary and press ENTER.

The 9010A accepts any binary value from 0 to 11111111. The binary values entered correspond to the UUT control lines. The 9010A forces control lines represented by a one high, and forces control lines represented by a zero low. The control lines for each microsystem are documented in the appropriate interface pod manual. Note that the 9010A supplies the value zero for any lines not specified by the operator.

The 9010A accepts the binary string bbbbbbbb and prompts for the bit number of the control line that is to be toggled by displaying the following message:

   DTOG @ CTL = bbbbbbbb BIT _

3. Key in the bit number d in decimal. The 9010A accepts any decimal value from 0 to 7 from the bit number of the control line. The 9010A displays the following message:

   DTOG @ CTL = bbbbbbbb BIT d WAIT

When the specification is complete, the 9010A performs two write operations. First the 9010A writes the control lines as specified. Then the 9010A toggles the specified bit, changing its logic level to the opposite value. Then the 9010A again writes the control lines, with the line corresponding to the toggles bit written to the opposite value. After the write operations are completed, the 9010A replaces WAIT with an OK or FAIL.

USING THE MODE KEYS

There are five MODE keys. Four of the modes are functions that modify the performance of actions which have been specified: the Continue (CONT), Stop (STOP), Repeat (RPEAT), and Loop (LOOP) mode functions. These four functions are related and will be discussed together. The fifth mode, RUN UUT, will be discussed in a separate section.

Stop, Continue, Repeat, and Loop Functions

The effect that the Stop, Continue, Repeat, and Loop functions have on the activity of the 9010A depends on the activity that is taking place at the time the mode function key is pressed. The best method to describe and understand the operation of the Mode Keys is an exercise. The exercise will demonstrate the use of the Mode Keys by addressing and observing the UUT's display.

1. RAMP @ 1ØØ8Ø.

2. Observe the UUT display.

3. If Repeat were to be pushed the RAMP would cycle through one more time.

4. Press RPEAT and observe the display.

5. To continuously RAMP at the display what key would be pressed?

6. The LOOPING key will cause the 9010A to continuously RAMP @ 1ØØ8Ø. Now press the LOOP key and observe the display.

7. The STOP key will stop the loop @ 1ØØ8Ø.

8. Press the CONT key.

   Why was there a BEEP?

The only time that the CONT Key has an effect is when either an error is involved or during program execution.

CASSETTE TAPE OPERATION

This section describes the operation of the cassette tape. It includes information about cassette loading, the Read Tape and Write Tape operations, and error detection.

Tape-Transferable Data

Tape-transferable data may consist of test programs, UUT memory map descriptors, and Setup parameters. No other type of data may be written to or read from the 9010A cassette tape. The storage capacity of the 9010A memory which stores the tape-transferable data is 12K bytes.

Loading the cassette into the 9010A

To load a cassette into the 9010A, follow these steps.

1. Press the eject button at the left rear of the cassette enclosure to release the cassette enclosure.

2. Select the desired side of the cassette, turn it face up with the open side of the cassette facing the rear, insert the cassette, and close the enclosure.

Read Tape

To select the Read Tape operation, press the READ TAPE key. The 9010A displays the following messages:

READ TAPE - ARE YOU SURE?

If the operator presses the NO key, the 9010A displays the following message:

READ TAPE ABORTED

If the operator presses the READ TAPE key and then the YES key, the 9010A displays the following message:

READ TAPE WAIT

The 9010A rewinds the tape (if necessary) and begins reading information from the tape. If the 9010A does not detect any errors during the operation, the 9010A completes the operation and displays the following message:

READ TAPE OK

Errors that may be detected with the Read Tape operation are described in the operators manual Table 4P-1. Note that if the Read Tape operation fails, the 9010A memory is restored to power-on values.

The Read Tape operation may not be interrupted by the operator and then continued. It is possible to abort the operation by pressing the key for another function, such as STOP or a troubleshooting function. Aborting the Read Tape operation before it is completed removes any data stored by the operator in the 9010A internal memory.

Write Tape

To select the Write Tape operation, press the WRITE TAPE key. The 9010A displays the following message:

WRITE TAPE - ARE YOU SURE?

If the operator presses the NO key, the 9010A displays the following message:

WRITE TAPE ABORTED

If the operator presses the WRITE TAPE key and then the YES key, the 9010A displays the following message:

WRITE TAPE WAIT

The 9010A rewinds the tape (if necessary) and begins writing information on the tape. When all the data has been transferred from internal memory, the 9010A again rewinds the tape. Then the 9010A performs a byte-for-byte comparison of the data on the tape and the data stored in memory. If no errors are detected, the 9010A displays the following message:

WRITE TAPE OK

Errors that may be detected with the Write Tape operation are described in the operators manual Table 4P-1.

The Write Tape operation may not be interrupted by the operator and then continued. It is possible to abort the operation by pressing the key for another function, such as STOP or a troubleshooting function. Aborting the Write Tape operation does not affect the internal memory of the 9010A.

Write Tape Protection

Later models of the 9010A have a Write Tape protection which functions as follows. The cassettes have a plastic punchout tab which may be removed to disable the Write Tape operation. This protects data already written on the tape. A plastic tab is located on each side of the cassette, and the removal of the tab affects only the side on which it is located. Removal of the tab does not affect the Read Tape operation.

If a cassette is inserted which has the tab removed and the WRITE TAPE key is pressed, the 9010A detects that the tape is Write Tape protected and presents the following message:

WRITE TAPE - WRITE PROTECTED

TAPE OPERATION ERROR MESSAGES

ERROR MESSAGE	DESCRIPTION
READ TAPE-NO CASSETTE WRITE TAPE-NO CASSETTE	These messages may be presented during the selection of the tape operations. They indicate that no cassette is loaded in the 9010A.
READ TAPE FAIL	There are a variety of possible causes: Cassette enclosure open. Loss of synchronization or overruns during transfer of data. Unmatching checksums. When the 9010A transfers data from internal memory to the tape, it generates a checksum from the data and stores the checksum with the data. When the Read Tape operation occurs, the 9010A generates a checksum from the data that is read from the tape and compares the new checksum with the checksum stored on tape. If the checksums do not agree, an error is reported.
WRITE TAPE FAIL	There are a variety of possible causes: Cassette enclosure open. Loss of synchronization or overruns during ransfer of data. Failure of the byte-for-byte data comparison test.

DEFAULTS

Defaults are parameters, such as addresses or ROM signatures, that are stored in memory and supplied by the 9010A if the prompted specification is not supplied by the operator. This section describes how the operator may use defaults during the specification of a 9010A action.

Defaults During Specification

The operator may invoke a default value by pressing the ENTER key when the 9010A prompts for a specification. For example, when the READ key is pressed, the 9010A prompts for the address where the Read operation is to be performed by displaying the following message:

```
READ @ _
```

The operator may default the address specification by pressing ENTER. The 9010A supplies the contents of Register F, which is generally the last address that was specified or used during an operation, and performs the operation.

The following examples compare the manual entry with the default entry for a troubleshooting function.

Press	Display	Comment
READ	READ @ _	Prompt for address.
40FF ENTER	READ @ 40FF	Operator enters the address and the 9010A performs the operation.
READ	READ @ _	Prompt for address.
ENTER	READ @ 40FF	Operator defaults address specification by pressing ENTER. The 9010A supplies the address 40FF and performs the operation.

Defaults are primarily useful in saving the operator time and effort. Defaults allow the operator to enter parameters from a previous operation. The parameters that are to be used again can be entered with one keystroke (the ENTER key) instead of having to be completely reentered.

For example, if the operator wants to write the same data (7AB3) to two addresses (20F0,38C6), the data specification may be defaulted as follows:

Press	Display	Comment
WRITE	WRITE @ _	Prompt for address.
20F0 ENTER	WRITE @ 20F0 = _	Prompt for data.
7AB3 ENTER	WRITE @ 20F0 = 7AB3	9010A performs the operation as specified.

WRITE	WRITE @ _	Prompt for address.
38C6 ENTER	WRITE @ 38C6 = _	Prompt for data.
ENTER	WRITE @ 38C6 = 7AB3	Operator defaults data specification by pressing ENTER. 9010A supplies data and performs the operation.

Defaults may also be provided during the specification of the functional tests (ROM, RAM, and I/O tests). For example, when the ROM TEST key is pressed, the 9010A prompts for the address and ROM signature specifications. Rather than manually entering the prompted values, the operator may press the ENTER key when prompted for the first address. This causes the 9010A to supply the UUT ROM descriptors obtained during the Learn operation or entered through the VIEW keys.

DETECTION OF UUT ERRORS

Detecting and reporting errors involving the UUT is critical, since errors in UUT actions are the symptoms available to the operator in diagnosing and repairing the UUT. The 9010A provides a consistent, readily understandable system for handling UUT errors. The 9010A error handling system is described in this section.

Types of UUT Errors

There are three types of errors involving the UUT that can occur: timeout errors, UUT system errors, and test errors. They are described in the following paragraphs.

1. TIMEOUT ERROR.

   To perform each action on the UUT, the 9010A must successfully transmit a command consisting of one or more bytes to the interface pod, and the 9010A must successfully receive a response of one or more bytes from the interface pod. Communication of these bytes involves handshaking between the interface pod and the 9010A. If the handshake is not successfully completed and the 9010A is unable to get a response from the interface pod, the 9010A "times out". Any of three timeout error conditions can be detected, and the appropriate timeout error message is displayed.

   The three possible one-line timeout error messages are as follows:

   ```
   POD TIMEOUT          -ATTEMPTING RESET
   UUT POWER FAIL       -ATTEMPTING RESET
   POD RESET ERROR      -ATTEMPTING RESET
   ```

   The first message is a non-specific message that indicates a variety of possible causes of the error. These may include a missing UUT clock or a faulty connection between the interface pod and the 9010A.

The second message indicates that the UUT power supply may be out of tolerance, and the UUT clock may be faulty.

The third message indicates that the interface pod may be the cause of the error, and the operator should perform the interface pod self test. When a timeout error occurs, the 9010A attempts to reset the interface pod. If the reset fails, the 9010A continues to attempt the reset until it succeeds or the operator initiates some other action.

2. UUT SYSTEM ERROR.

The interface pod continually monitors several UUT conditions while the 9010A is in operation. For example, whenever the interface pod writes to a line, the pod monitors the actual logic level on the line. If the value written and the value monitored disagree, a "drivability" error is considered to have occurred. Driveability errors may be detected on control, address, or data lines. These errors typically occur if another bus device is holding onto a data or address line or if there are short circuits associated with these lines.

Other UUT conditions monitored during operations include the power supply voltage at the uP socket and interrupt requests. Errors associated with these conditions are called UUT system errors. If an error is detected, the 9010A displays one of the messages listed and described in table provided.

Note that four of the UUT system error messages listed in the table have parentheses around the address portion of the message, (@ aaaa). This is because the address is included in the message only if the operator specifies an address during selection of the operation or if the 9010A used internally generated addresses for the operation that reported the error. For example, the Write Control function requires no address, and no addresses are included in associated UUT system errors. Similarly, the UUT system address or data errors are not reported at all for operations that do not require an address, such as Write Control.

Sometimes it may be useful to disable the detection of a particular UUT system error if the reporting of such an error interferes with testing or troubleshooting. UUT system error detection may be disabled by the Setup function.

3. TEST ERROR.

The largest category of UUT errors are associated with the functional tests. The error messages are listed and described with the tests and are summarized in the tables provided.

Hierarchy In Detection Of UUT Errors

The three types of UUT errors are arranged according to priority. The hierarchy in order of priority is as follows.

1. Timeout Error — Highest Priority
2. UUT System Error
3. Test Error — Lowest priority

This hierarchy means that the 9010A detects and reports higher priority errors before lower priority errors. Higher priority errors must be removed or their detection disabled (if possible) before the 9010A detects and reports lower priority errors.

Looping On Errors With The Loop Function

One of the most powerful troubleshooting features of the 9010A is its capacity to continually monitor and report the appearance or disappearance of a UUT failure while the operator physically manipulates the UUT. This monitoring process is accomplished with the Loop function.

When the 9010A detects a UUT error, the 9010A suspends execution of whatever operation is in progress and reports the error. A prompt for the Loop function key is included in many of the error messages. The operator may press the Loop key to cause the 9010A to continuously perform the operation. The operator stimulates or physically manipulates the UUT to try to remove the cause of the error, while observing the 9010A display or listening for the audible beep to determine if the error is still reported. This process, called "looping on errors", helps the operator to see the effect of his actions and quickly locate the cause of the UUT error.

To illustrate the process of looping on an error, consider the following 9010A display message which reports a UUT system error.

```
BAD PWR SUPPLY @ aaaa-LOOP?
```

This message reports that the UUT is not supplying the proper voltage levels at the uP socket. Note that the 9010A beeps to attract the operator's attention when the error is first reported. If the operator presses the LOOP key or the YES key, the 9010A continuously performs the same operation during which the error was reported. If the error condition is corrected during the Loop function, the 9010A continues looping and displays the following message:

```
GOOD PWR SUPPLY @ aaaa
```

Note that while a prompt for a loop is displayed, the STOPPED annunciator flashes. It discontinues flashing only when the operator initiates some kind of action, such as pressing the LOOP or CONT key. Note also that while the 9010A is looping on an error, the LOOPING annunciator flashes.

Note that the use of the Loop function is not restricted to use with errors. The Loop function may also be used with some of the 9010A tests, functions, and operations.

Understanding OK or FAIL

When the 9010A completes the execution of an operator-specified operation, test, or function, such as Learn, ROM Test, or Ramp, the message displayed ends with either the word OK or the word Fail. For example, if the Read function is specified to be performed at address aaaa, when the 9010A completes the operation, one of the following messages is displayed:

```
READ @ aaaa = hhhh OK
READ @ aaaa = hhhh FAIL
```

OK at the end of a display message indicates that during the execution of the operation, either no errors occurred, or else all errors that occurred were corrected by the operator before proceeding.

FAIL at the end of a display message indicates that during the execution of the operation, one or more UUT errors were reported but not corrected by the operator. FAIL is a warning to the operator that something in the operation was amiss, and the results may not be reliable.

To illustrate the meaning of FAIL, assume the operator selects and specifies a ROM Test. During the course of the operation, the following message appears:

```
ROM ERR @ 1000-1FFF-LOOP?
```

Note that this message is the first line of a two-line message. The operator chooses to ignore the error message and presses the CONT key. The 9010A completes the operation and displays the following message.

```
ROM TEST FAIL
```

The message, ROM TEST FAIL, warns the operator that even though the operation was completed, the results are not reliable because an error was detected, reported, and not corrected.

UUT ERROR TYPES AND MESSAGES

TYPE	MESSAGE	DESCRIPTION
	POD TIMEOUT -ATTEMPTING RESET	General error message caused by such things as no UUT clock or a faulty connection between the pod & 9010A.
TIMEOUT ERROR	UUT POWER FAIL -ATTEMPTING RESET	UUT power supply possibly at fault.
	POD RESET ERR -ATTEMPTING RESET	Interface pod possibly failing. Operator should perform interface pod self test.
	BAD PWR SUPPLY (@ aaaa)-LOOP?*	Improper power supply voltages at UUT uP socket.
	ILLEGAL ADDR @ aaaa-LOOP?	This message appears if the operator keys in an address that is not within the valid uP address space.
	ACTIVE FORCING LINE (@ aaaa)-LOOP?* STS BTS bbbb bbbb bbbb bbbb	These two-line messages are displayed if any forcing lines or interrupts are detected to be active during an operation. The binary string in the second line corresponds to the status lines documented in the appropriate pod manual. A one corresponds to an active line, and a zero corresponds to an inactive line. The binary string is shown in groups of four, with up to 16 digits in all.
	ACTIVE INTERRUPT (@ aaaa)-LOOP?* STS BTS bbbb bbbb bbbb bbbb	

UUT SYSTEM ERROR	CTRL ERR (@ aaaa)-LOOP?* BAD CTRL BTS bbbbbbbb-LOOP? ADDR ERR @ aaaa-LOOP? BAD ADDR BTS bbbbbbbb-LOOP? DATA ERR @ aaaa-LOOP? BAD DATA BTS bbbbbbbb-LOOP?	These two-line messages are displayed when any control, address, or data lines are detected to be not drivable. The string bbbbbbbb is a mask of 1's & 0's where a 1 corresponds to a line that is not drivable. Address errors or data errors may only be detected if an address is included in the specification of an operation.

TEST ERROR	Test error messages are listed and described with the tests in which they occur in subsequent sections of this manual.

---

*Note that the address is included in the error message only if an address is included in the specification of an operation.

BUS TEST Error Messages

	ERROR TYPE	ERROR MESSAGE	NOTE
1.	Control Lines Not Drivable	CTL ERR bbbbbbbb bbbbbbbb-LOOP?	1
2.	Address Lines Tied High or Low	ADDR BIT aa TIED HIGH-LOOP?	2
		ADDR BIT aa TIED LOW-LOOP?	2
3.	Address Lines Tied Together	ADDR BITS a1 and a2 TIED-LOOP?	3
4.	Data Lines Tied High or Low	DATA BIT dd TIED HIGH-LOOP?	4
		DATA BIT dd TIED LOW-LOOP?	4
5.	Data Lines Tied Together	DATA BITS d1 AND d2 TIED-LOOP?	5

NOTES:

1. In the binary 16-bit string bbbbbbbb bbbbbbbb, a 1 corresponds to undrivable lines, and a 0 corresponds to drivable lines. Refer to the appropriate interface pod manual for the control lines that correspond to the bit numbers. Extra bits are reported as 0.

2. The "aa" is a decimal number corresponding to the bit number of the bit tied high or low. The range of values depends on the number of address lines.

3. The "a1" and "a2" are decimal numbers corresponding to the bit numbers of the bits tied together, with a1< a2.

4. The "dd" is a decimal number corresponding to the bit number of the bit tied high or low. The range of values depends on the number of data lines.

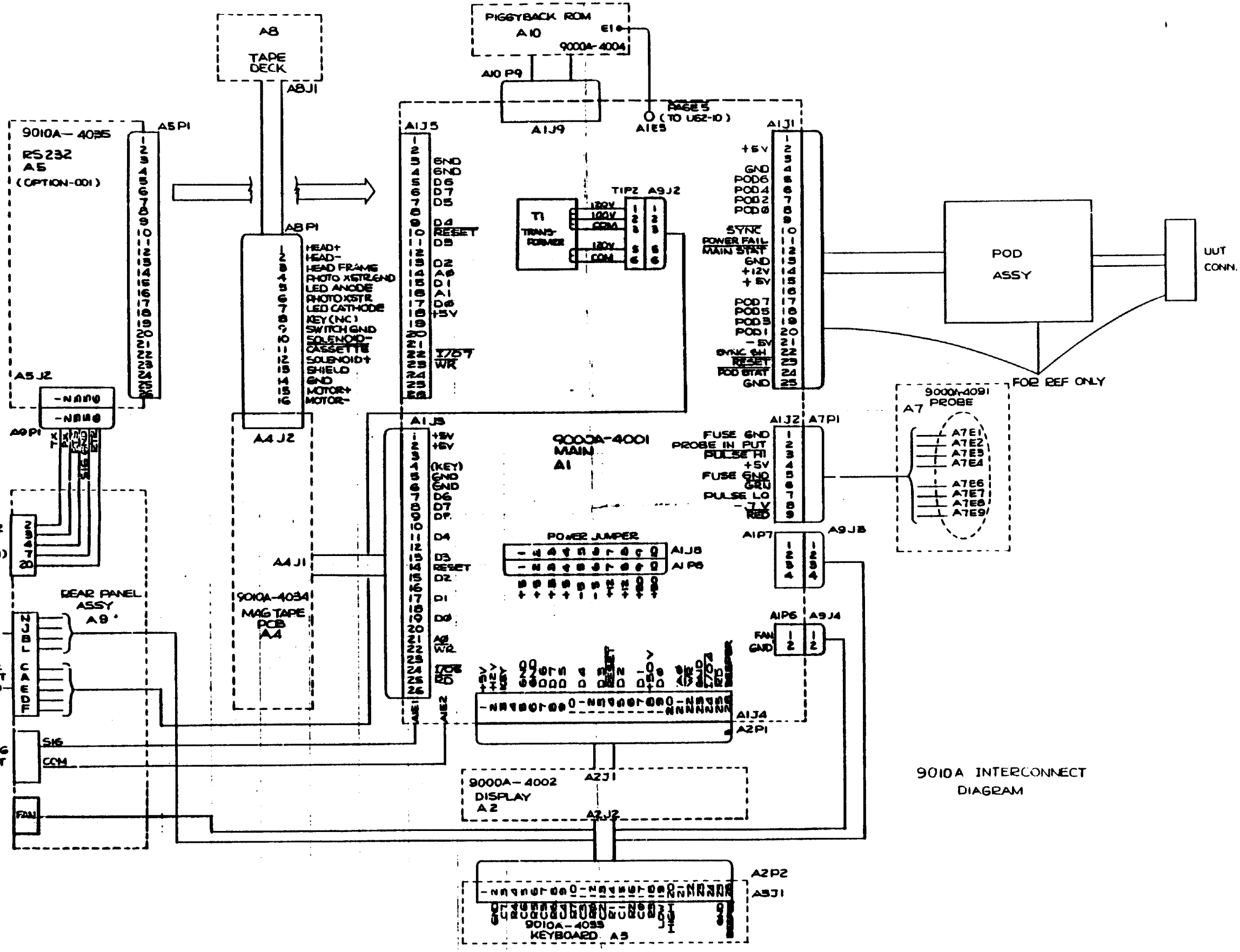
5. The "d1" and "d2" are decimal numbers corresponding to the bit numbers of the bits tied together, with d1< d2.

Functional Test Error Messages

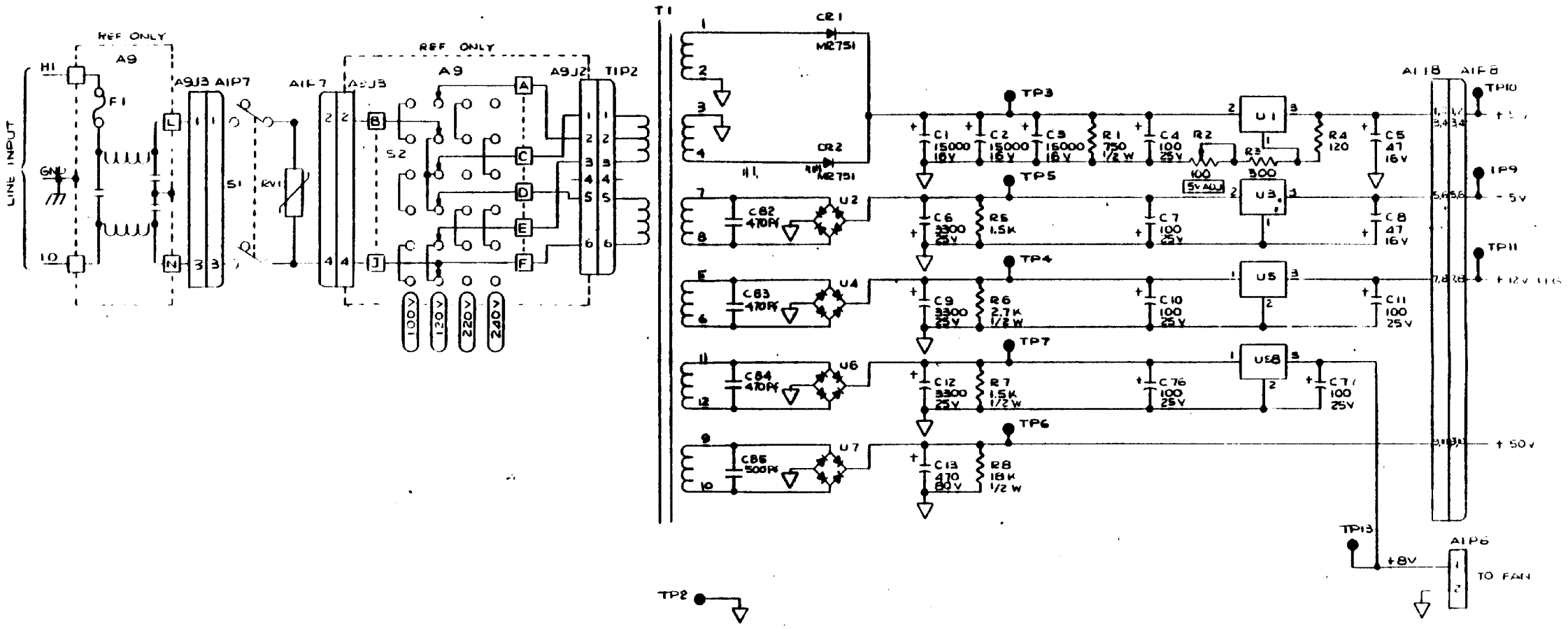
I/O TEST	R/W ERR @ aaaa BTS bbbbbbbb-LOOP?	Binary string bbbbbbbb is bit mask where 1=bit specified read/writable and detected not read/writable, 0=bit not specified.
ROM TEST	ROM ERR @ aaaa-aaaa-LOOP? SIG WAS mmmm NOT nnnn-LOOP?	Two-line message. Hex string mmmm=signature computed during test. Hex string nnnn=signature specified before test.
RAM SHORT	R/W ERR @ aaaa BTS bbbbbbbb-LOOP?	Binary string bbbbbbbb, 1=bit identified not read/writable, 0=bit identified read/writable.
RAM SHORT	RAM BITS d1 AND d2 TIED-LOOP?	Decimal numbers d1 and d2 are the tied bits.
RAM SHORT	RAM DCD ERR @ aaaa BIT dd-LOOP?	Decimal number dd is the bit that is not decoding properly.
RAM LONG	All error messages associated with RAM SHORT also apply RAM LONG.	
RAM LONG	RAM PATT ERR @ aaaa-LOOP? DATA WAS xxxx NOT hhhh-LOOP?	Two-line message. Hexadecimal number xxxx is actual data, hexadecimal number hhhh is expected data.
BUS TEST	BUST TEST error messages are summarized in operator's manual Table 4H-1.	
AUTO TEST	There are no error messages unique to AUTO TEST. Any error messages that might be encountered are described with the tests that AUTO TEST invokes.	

Fatal Error Messages

ERROR MESSAGE	DESCRIPTION
FATAL-MEMORY EXCEEDED FOR LEARN	The address descriptors obtained during a Learn operation have exceeded the 9010A internal memory.
FATAL-ATTEMPTED RECURSION FATAL-DEPTH EXCEEDED FATAL-ILLEGAL EXPRESSION FATAL-PROG NOT FOUND	These messages indicate something is wrong with the construction of the program, and can only be corrected by modifying the program. For an explanation of what specific corrections are needed, refer to the 9010A Programming Manual.



9010A INTERCONNECT DIAGRAM



SCHEMATIC—
MAIN, PCB A1

SH 1 OF 4

POWER SUPPLY NORMAL AND UNLOADED MEASUREMENTS

Supply unloaded. 132VAC +1V, -0V AC input.

Measure the filtered DC (at the filter caps). Minimum DC values:

+5V Supply:	13.2VDC
+8V Supply:	17.4VDC
+12V Supply:	24.0VDC
+50V Supply:	57.8VDC
-5V Supply:	-14.7VDC

If the polarity is reversed, terminate the test immediately. A diode bridge is installed backwards. If the filtered DC falls below these minimums, a filter cap is installed backwards, or the transformer is defective. Terminate the test and repair the unit.

Set the +5V output to 5.05VDC +-100MVDC. Save the reading. Measure the +8V, +12V, -5V supplies, and save the reading. Load all the supplies to the following values:

+5V Supply:	1.5A
+8V Supply:	230mA
+12V Supply:	150mA
+50V Supply:	40mA
-5V Supply:	150mA

Measure the dc output voltage for each supply at 105vac +0v -1v ac input. Max. change from unloaded:

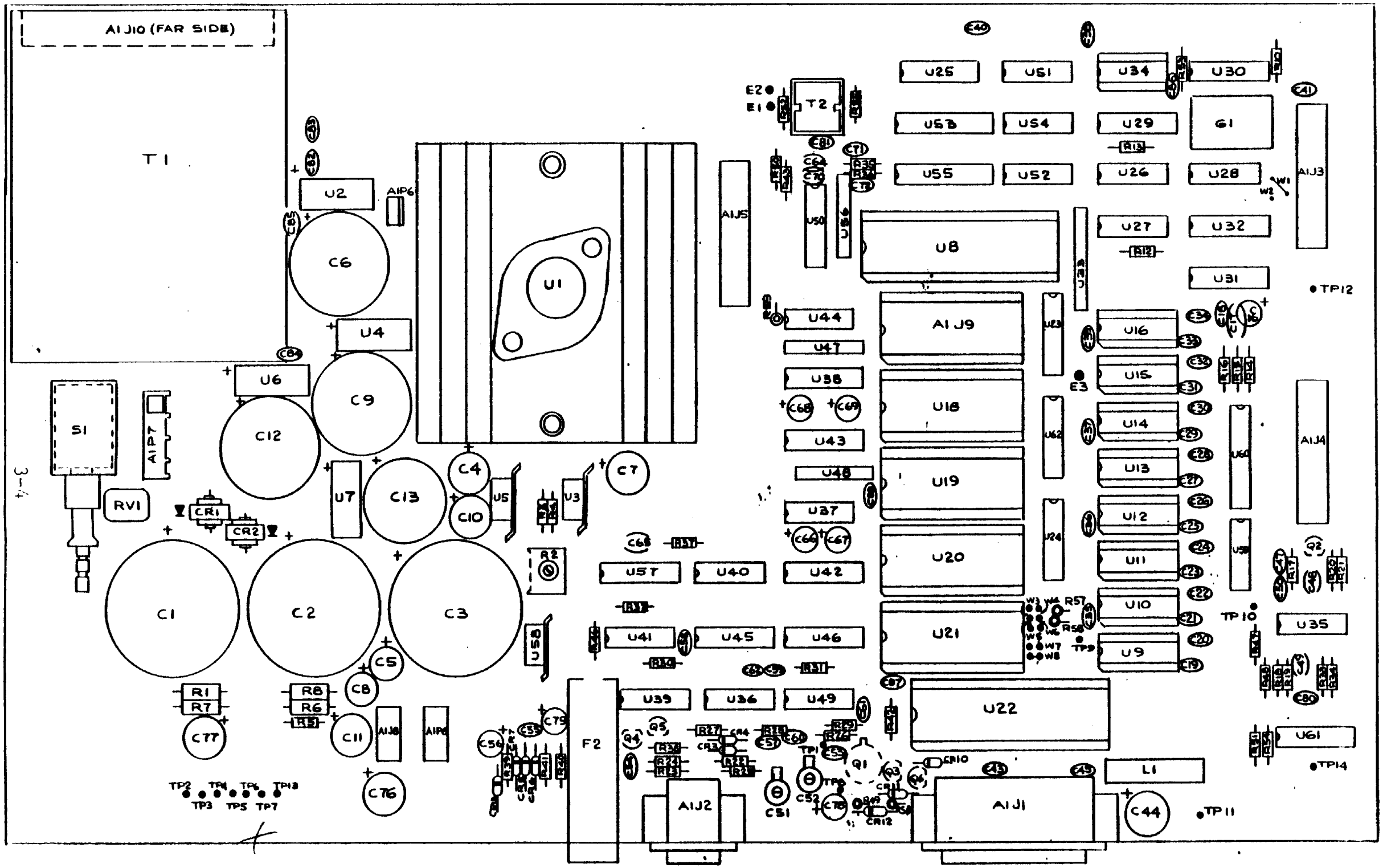
+5V Supply:	70mVDC
+8V Supply:	150mVDC
+12V Supply:	100mVDC
-5V Supply:	110mVDC

All supplies should be within the following limits:

+5V Supply:	4.8VDC to 5.1VDC
+8V Supply:	7.7VDC to 8.3VDC
+12V Supply:	11.5VDC to 12.5VDC
-5V Supply:	-4.8VDC to -5.2VDC

Measure the AC ripple on the DC outputs:

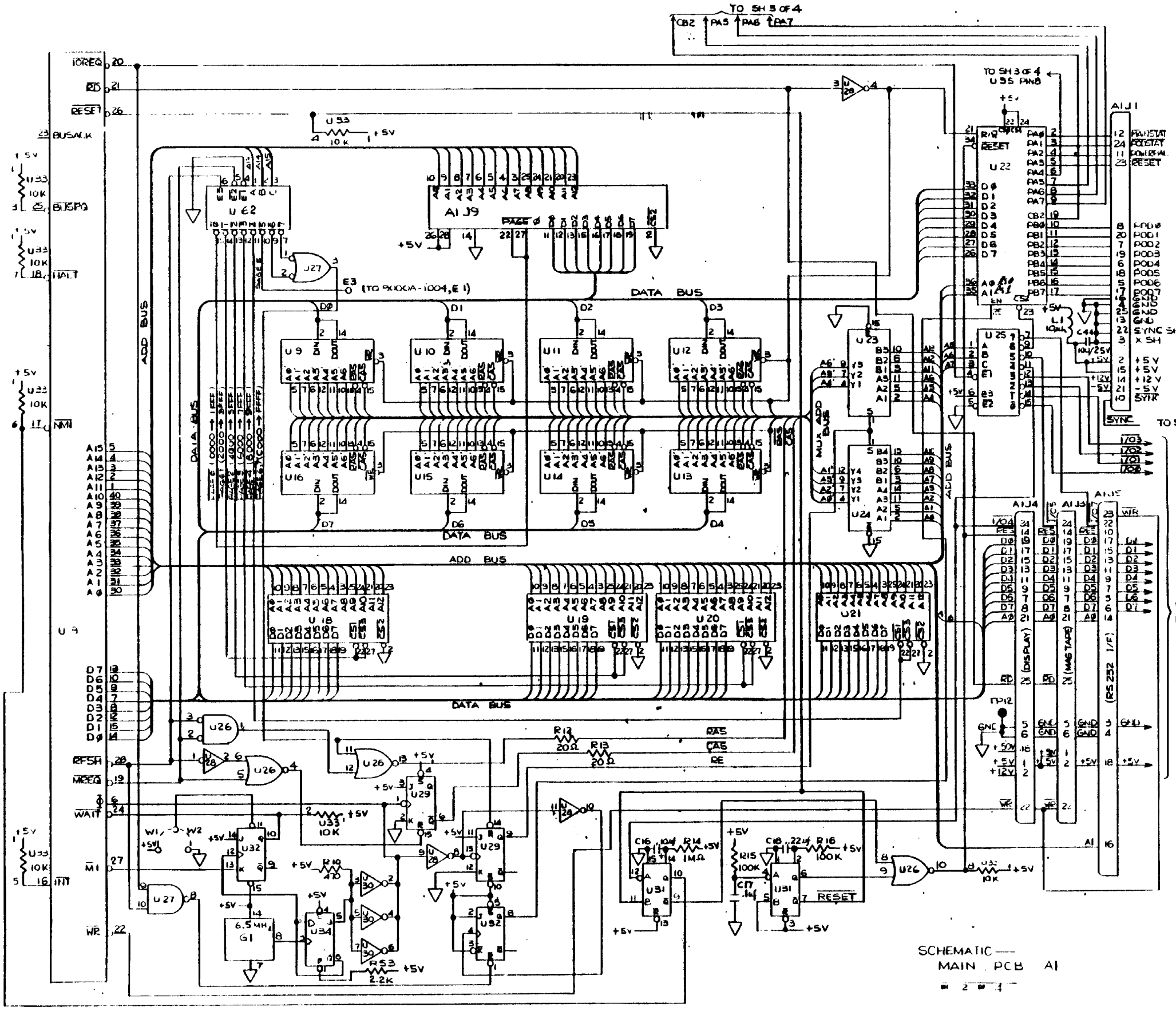
All supplies except the +50V Supply: 12MVRMS Max.

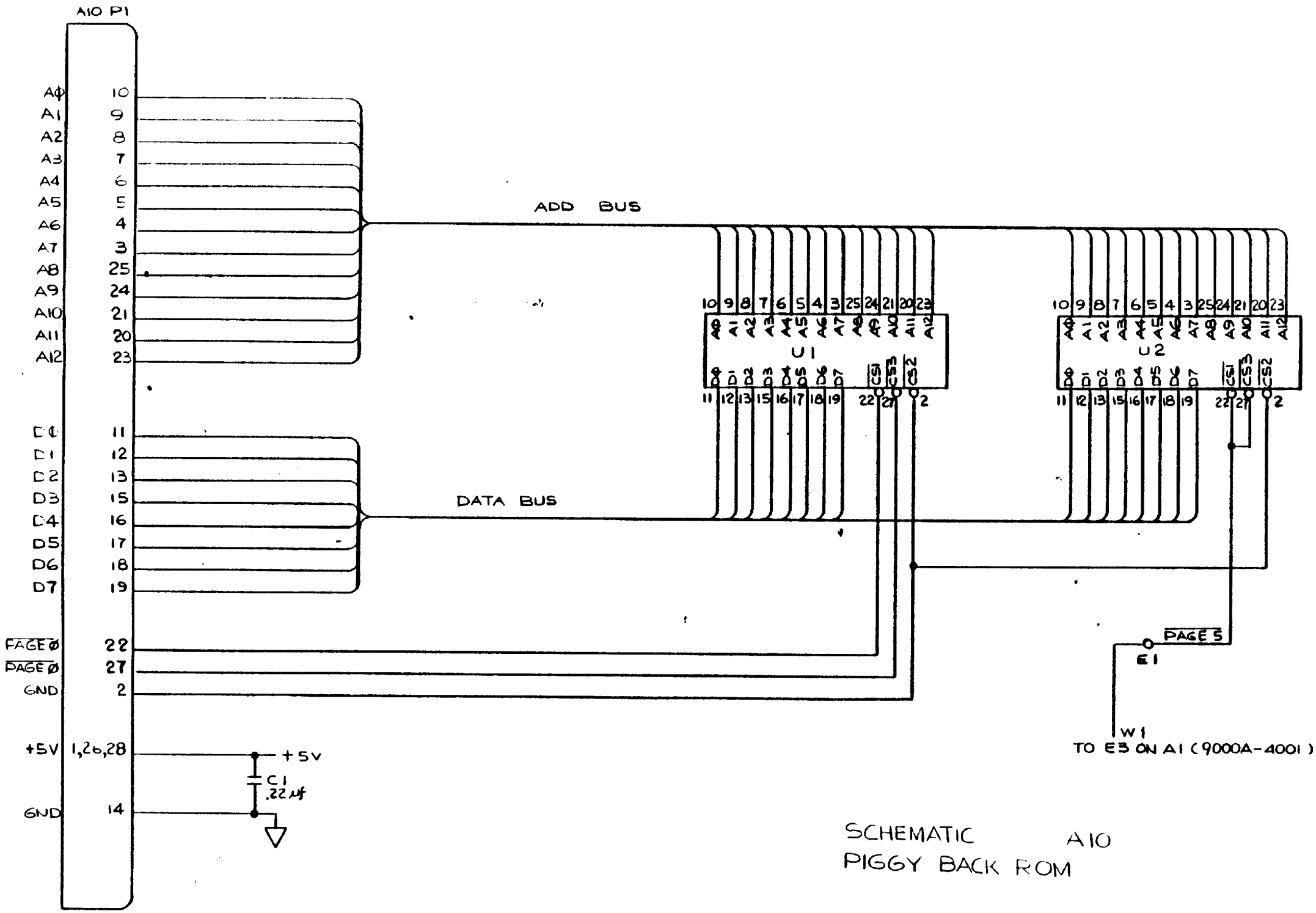


3-4

REFERENCE DESIGNATION

MAIN BOARD A1


TO SH 5 OF 4
CB2
PA5
PA6
PA7
TO SH 3 OF 4
U 35 PIN 8
IORQ
RD
RESET
BUSAK
BUSRQ
HALT
NMI
INT
WAIT
MREQ
RFSH
M1
WR
ADD BUS
DATA BUS
U 33
10K
+5V
U 62
U 27
E3
(TO 9000A-1004, E1)
A1 J9
PAGE Ø
U 9
U 10
U 11
U 12
U 16
U 15
U 14
U 13
U 18
U 19
U 20
U 21
U 22
U 23
U 24
U 25
U 26
U 27
U 28
U 29
U 30
U 31
U 32
U 34
A1 J1
A1 J2
A1 J3
A1 J4
A1 J5
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
CB2
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
PA0STAT
PB0STAT
CONTROL
RESET
POD0
POD1
POD2
POD3
POD4
POD5
POD6
POD7
GND
SYNC SH
X SH
+5 V
+12 V
-5V
SYNC
TO SH 3 OF 4
TO SH 4
(DISPLAY)
(MASS TAPE)
(RS 232 I/F)
WR
RD
RAS
CAS
RE
RESET
SCHEMATIC —
MAIN PCB A1



SCHEMATIC A10
PIGGY BACK ROM

SETUP INFORMATION

POD - Z80
ENABLE BUSRQ-YES
ENABLE WAIT-YES

TRAP BAD POWER SUPPLY-YES
TRAP ILLEGAL ADDRESS-YES
TRAP ACTIVE INTERRUPT-NO
TRAP ACTIVE FORCE LINE-YES
TRAP CONTROL ERROR-YES
TRAP ADDRESS ERROR-YES
TRAP DATA ERROR-YES

EXERCISE ERRORS-YES
BEEP ON ERR TRANSITION-YES

BUS TEST @ FFFF
RUN UUT @ 0000
TIMEOUT 100
STALL 13
UNSTALL 11
NEWLINE 00000D0A
LINESIZE 7920

ADDRESS SPACE INFORMATION

RAM @ C000-FFFF

ROM @ 0000-1FFF SIG 8345
ROM @ 2000-3FFF SIG 895A
ROM @ 4000-5FFF SIG F3AA
ROM @ 6000-7FFF SIG 9BE4
ROM @ 8000-9FFF SIG 2140
ROM @ A000-BEFF SIG 5583
ROM @ BFC0-BFFF SIG 810D

ROM TEST

To understand ROM addressing and how U62 works let's review binary to hex conversion.

Convert the following binary number to hex. 0001111111111111.
1 F F F.

With the Z-80 we have 16-address lines. The ROM's that are used (2564's), use 13 of those address lines (AØ-A12). The 9010A uses the other three address lines (A13-A15) as an input to a decoder (U62) for the chip select.

The chip select, from the decoder, will enable a specific ROM depending on the state of A13-A15.

The 9010A schematic maps out the address for each ROM and RAM. For the time being we won't discuss the RAM address map. To identify this map take out the Main PCB schematic (A1 sheet 2 of 4) and look at the output side of U62, (located in the upper left corner), pin 15, 14, 13 etc. Follow the lines from those pins and locate the writing, PAGE Ø (ØØØØ-1FFF). This is the chip select line (CS), which when active low ($\overline{CS}$), enables a specific ROM. This line indicates the specific address space in that ROM.

An example of this is U62 pin 15. This is the, $\overline{\text{PAGE}}$ Ø (ØØØØ-1FFF), address enable line for A1OU1. Converting the address to binary form is illustrated below.

	A15	A14	A13	A12		A11	A1Ø	A9	A8		A7	A6	A5	A4		A3	A2	A1	AØ
ØØØØ =	0	0	0	0	/	0	0	0	0	/	0	0	0	0	/	0	0	0	0
1FFF =	0	0	0	1	/	1	1	1	1	/	1	1	1	1	/	1	1	1	1

There is a (CS) line for each and every ROM. That means that the memory map for ROM has 6 pages with addressing from (ØØØØ-BFFF). By looking at the address map for each ROM we can construct a truth table for U62.

There will be 2 exercises to the ROM TEST. The first is to construct the truth table for U62 and the second will be verification of that truth table using the 9010A probe.

In the previous example we found that A1OU1 was enabled when A13, A14 and A15 equaled zero. An important thing to remember about U62 is it must be enabled by U8. This is accomplished with the $\overline{\text{RFSH}}$ and $\overline{\text{MREQ}}$ lines. To enable U62 E3 must be a logic 1, E2 and E1 must be logic 0's.

EXERCISE #1

Complete a truth table for U62. In the previous example we found A10U1 was enabled when A13, A14 and A15 equaled zero. A close examination of PAGE 3 (6ØØØ-7FFF) will illustrate the condition of A13-A15 and what ROM will be addressed.

	A15	A14	A13	A12	/	A11	A10	A9	A8	/	A7	A6	A5	A4	/	A3	A2	A1	AØ
6ØØØ =	0	1	1	0	/	0	0	0	0	/	0	0	0	0	/	0	0	0	0
7FFF =	0	1	1	1	/	1	1	1	1	/	1	1	1	1	/	1	1	1	1

PAGE 3 will enable U20; A14 & A13 = logic HI.

Using the examples given, complete the following truth table.

ADDRESS	U62 PIN # →	Ø 15	1 14	2 13	3 12	4 11	5 10	A15 3	A14 2	A13 1	E3 6	E2 5	E1 4	ROM ENABLED
1FFF		0	1	1	1	1	1	0	0	0	1	1	0	U1 U10
3FFF		1	0	1	1	1	1	0	0	1	1	0	0	U18
5FFF		1	1	0	1	1	1	0	1	0	1	0	0	U19
7FFF		1	1	1	0	1	1	0	1	1	1	0	0	U20
9FFF		1	1	1	1	0	1	1	0	0	1	0	0	U21
BFFF		1	1	1	1	1	0	1	0	1	1	0	0	U2 U10

EXERCISE #2

Now that we've completed the truth table for U62, let's verify it with the 9010A probe. This exercise can be accomplished by addressing each individual ROM in the immediate mode.

1. To view address information with the probe, the first step is to SYNC to address information.

KEYBOARD ENTRY	9010A DISPLAY
SYNC	SYNC MODE <Ø-F>
A	SYNC MODE <Ø-F> ADDRESS OK

A - Address ONLY possible
D - Data
F - Free Run

2. Connect the probe ground to TP2.

3.

KEYBOARD ENTRY	9010A DISPLAY
READ	READ @
1FFF	READ @ 1FFF
ENTER	READ @ 1FFF = 23 OK
LOOP	LOOPING light comes on

4. The 9010A is now ready to verify the first step in your truth table. What we have done with the above keystrokes is continuously address 1FFF. This will allow us to check the states of the ($\overline{CS}$) lines from U62.

   a. Place the probe on U62 pin 15. What light is lit on the probe?

   Green

   Does this light indicate a logic HI or LOW?

   low

   b. Place the probe on U62 pin 14. What light is lit on the probe?

   Red

   Does this light indicate a logic HI or LOW?

   Hi

5. Now that we've seen how the probe lights operate let's change the ROM address.

KEYBOARD ENTRY	9010A DISPLAY
READ	READ @
2ØØØ	READ @ 2ØØØ
ENTER	READ @ 2ØØØ = DD OK
LOOP	LOOPING light comes on

The 9010A is now continuously addressing ROM space 2ØØØ.

   a. Place the probe on U62 pin 15. What light is lit on the probe?

   Red

   Does this light indicate a logic HI or LOW?

   HI

   b. Place the probe on U62 pin 14. What light is lit on the probe?

 Green

Does this light indicate a logic HI or LOW?

Low

c. What is the difference between the indications found in question 5 and question 4?

Different Rows selected

6. By using the proper keystroke actions and probe, fill in the following table as to logic states of U62.

ADDRESS	U62 PIN # →	$\overline{0}$ 15	$\overline{1}$ 14	$\overline{2}$ 13	$\overline{3}$ 12	$\overline{4}$ 11	$\overline{5}$ 10	A15 3	A14 2	A13 1	E3 6	E2 5	E1 4	ROM ENABLED
1FFF		0	1	1	1	1	1	0	0	0	1	0	0	4 1 4 10
3FFF		1	0	1	1	1	1	0	0	1	1	0	0	
5FFF		1	1	0	1	1	1	0	1	0		1		
7FFF		1	1	1	0	1	1	0	1	1				
9FFF		1	1	1	1	0	1	1	0	0				
BFFF		1	1	1	1	1	0	1	0	1				

7. Now compare the table that you've just completed in question 6 to the truth table that was made in exercise #1. Do they match exactly? If not, why?

Yes.

8. Now that we have seen how the probe can be used and how the 9010A can read any address within range of the microprocessor being used, do the following:

   a. Read @ BFBØ.

   b. What data is read?

   FF

   c. Read above and below BFBØ always remaining between BEFF and BFCØ.

   d. What data is read?

   FF.

   e. Why?

   unprogramed Prom Data.

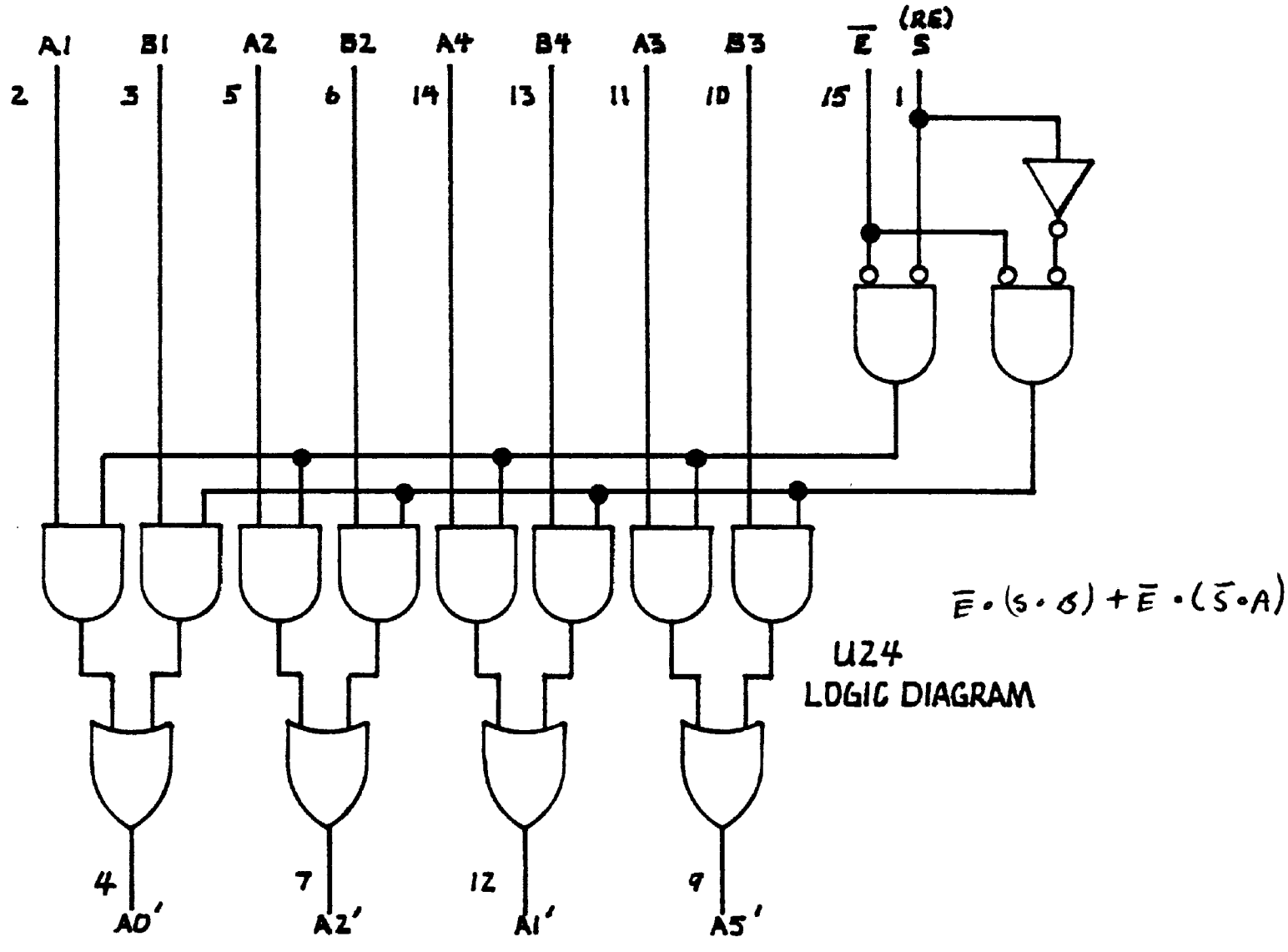
NOTE: The memory map found during a learn of the 9010A shows this to be an open area in the memory. In most systems the Data Bus is pulled-up. Therefore, when reading an empty location all Data lines will be high.

RAM TEST

The RAM for the 9010A can be found on the MAIN PCB schematic A1 sheet 2 of 4 (U9 thru U16). Dynamic RAM is set-up in Row and Column format. Thus making AØ-A6 on the RAM equivalent to 14 bits of addressing which gives 16K bits of memory space.

Remember that for a microprocessor to address 16K bits of memory, 14 address lines are required. The 9010A uses AØ-A13 as an input to Row and Column multiplexers (U23 and U24).

The Logic Diagram for U24 is illustrated below. Refer to that diagram for the discussion that follows.



U24
LOGIC DIAGRAM

Pin 1 is the stobe line for the multiplexer. A close look at that line reveals that the stobe is initiated by U29, Row Enable (RE). Pin 15, Enable ($\overline{E}$) is tied to low.

During stobe HIGH to U23 & U24 the address signals on the "B" inputs are passed to the output. The "A" inputs are passed to the output during stobe LOW. The "A" signals are used to set-up the column address so that the outputs of U23 and U24 during $\overline{CAS}$ is approximately 100ns wide.

Due to the operating characteristics of the circuitry, the best trouble-shooting approach for the MUX is to separate their operating modes. The ROW ADDRESS' come through the "B" input and are easily viewable by addressing RAM space at A7-A13 (ADDRESS>CØ7F).

The RE strobe must be LOW to verify the "A" input. The easiest way to accomplish this is to address something other then RAM space. Because something other than RAM space is addressed RE is low and the "A" inputs will be passed to the output of the multiplexers. The RAM's will not be enabled because $\overline{RAS}$ and $\overline{CAS}$ are not present.

EXERCISE #1

Knowing the following, come up with a method to check U23 and U24.

1. Row address information is present on A7-A13.

2. Column address information is present on AØ-A6.

3. RE HIGH enables U23 and U24 "B" inputs.

4. The pulse on the $\overline{CAS}$ is ≈ 100ns and may not be captured by the Test 9010A probe when Sync'd to ADDRESS.

5. RAM address starts at CØØØ and ends at FFFF.

With the 9010A there are 2 methods to test RAM. RAM SHORT and RAM LONG. RAM LONG will Read and Write to every RAM location. This test takes approximately 1 hour and 7 minutes. RAM SHORT takes approximately 4 minutes and will detect most RAM problems.

OK

EXERCISE #2

1. Perform a RAM SHORT test.

There is a third way to check RAM. This is done with PROGRAM Ø. The program checks each RAM by writing AA and 55 to each RAM location. If an error occurs, it will identify which DATA BIT is inoperative. It takes slightly over 1 hour to run the complete program. When an error occurs the instrument will BEEP.

If a RAM is totally inoperative the program will find the bad DATA LINE faster than RAM SHORT.

EXERCISE #3

1. Explain why the program writes to the RAM AA and 55.

IF Two Data lines were tied together this would not work because the data is alternating 1's and "0"s and the data lines are adjacent. IF a short between adjacent bits occured the pattern would be upset.

To verify the $\overline{\text{RAS}}$ (Row Address Strobe) and $\overline{\text{CAS}}$ (Column Address Strobe) lines, use the Test 9010A Probe Sync'd to ADDRESS and trace the logic back through U29, U26, and U27.

Higher order.

RAS CAS

[illegible]

for B inputs address Read at.

C080
C100
C200
C400
C800
D000
E000

for A address

0001
0002
0004
0008
0010
0020
0040

look at B inputs then outputs with Address sync.

```
PROGRAM 0    334 BYTES

    REG1 = C000
    REG2 = 1
    REG3 = 0
    DPY-BAD RAM IDENTIFIER
    STOP
0:  LABEL 0
    IF REG1 > FFFF GOTO 6
    WRITE @ REG1 = AA
    READ @ REG1
    DPY-ADDRESS = $F     DATA = $E
•   IF REGE = AA GOTO 3
1:  LABEL 1
    IF REGE AND REG2 > 0 GOTO 5
    INC REG3
    SHR REGE
    IF REGE AND REG2 > REG2 GOTO 5
    IF REG2 > REGE AND REG2 GOTO 5
    INC REG3
    IF REG3 > 80 GOTO 0
    SHR REGE
    GOTO 1
3:  LABEL 3
    WRITE @ REG1 = 55
    READ @ REG1
    DPY-ADDRESS = $F     DATA = $E
    INC REG1
    IF REGE = 55 GOTO 0
4:  LABEL 4
    IF REGE AND REG2 > REG2 GOTO 5
    IF REG2 > REGE AND REG2 GOTO 5
    INC REG3
    SHR REGE
    IF REGE AND REG2 > 0 GOTO 5
    INC REG3
    IF REG3 > 80 GOTO 3
    SHR REGE
    GOTO 4
5:  LABEL 5
    DPY-#
    DPY-CHANGE RAM CONNECTED TO D$3
    STOP
6:  LABEL 6
    DPY-RERUN RAM SHORT OR RAM LONG
```

POD INTERFACE TEST

The 9010A utilizes a PERIPHERAL INTERFACE ADAPTER (PIA) to interface the main frame to the POD and probe circuitry. The main focus for this section will be on the PIA device. The POD INTERFACE circuitry consists of a decoder U25 (the same type used in the ROM circuitry), a reset timer U31, and the PIA U22. All of these components may be found on the Main PCB schematic sheet 2 of 4.

The 9010A uses a Rockwell 6520. The 6520 Peripheral Interface Adapter provides the universal means of interfacing peripheral equipment to 9 Microprocessing Units (MPU). This device is capable of interfacing the MPU to peripherals through two 8-bit bidirectional peripheral data buses and four control lines. No external logic is required for interfacing to most peripheral devices.

The functional configuration of the PIA is programmed by the MPU during system initialization. Each of the peripheral data lines can be programmed to act as an input or output, and each of the four control/interrupt lines may be programmed for one of several control modes. This allows a high degree of flexibility in the over-all operation of the interface.

CHARACTERISTICS OF A PIA

* 8-bit bidirectional data bus for communication with the MPU.
* Two bidirectional 8-bit buses for interface to peripherals.
* Two programmable control registers.
* Four individually-controlled interrupt input lines; two usable as peripheral control outputs.
* Handshake control logic for input and output peripheral operation.
* High-impedance 3-state and direct transistor drive peripheral lines.
* Program controlled interrupt and interrupt disable capability.
* CMOS drive capability on side A peripheral lines.
* Two TTL drive capability on all A and B side buffers.
* TTL-compatible.
* Static operation.

Port A (PAØ-PA3) and Port B (PBØ-PB7) are used to interface to the POD. The remaining bits of Port A and CB2 are used to interface to the probe circuitry.

To check U22 we will start by checking the control lines using the 9010A in the immediate mode. The TIMER CIRCUIT, U31 must be checked first.

EXERCISE #1

Remove Jumper

Perform the following operation to check U31.

1. SYNC the probe to FREE-RUN.

2. Place the probe on U22 pin 34.

3. Read @ 1ØØ8Ø.

4. Verify the probe goes HIGH then LOW.

5. LOOP the 9010A with the READ @ 1ØØ8Ø.

6. Verify the probe stays HIGH.

The TIMER CIRCUIT (U31) serves two functions.

1. Power on RESET - At power on U31 pins 4, 5, etc. times out applying a logic LOW to U22 pin 34.

2. Reset U22 - If U31 pins 11, 12, etc. is <u>not</u> addressed, 1ØØ8Ø via U25, the circuitry will time out approximately every 2 sec. resetting U22. This is done to prevent the data lines from the POD hanging-up the 9010A.

EXERCISE #2

The next step to verify U22 is to ensure that the control lines are in the proper state. U25 is the same decoder used in the ROM TEST so time won't be spent to verify its operation.

1. Disable the Timer Circuit. To accomplish this jumper U31 pin 14 to U31 pin 15.

2. SYNC probe to ADDRESS.

3. Write @ 1ØØC1=Ø.

4. LOOP.

5. What logic level should $\overline{CS2}$ be to enable U22?

Low

Verify your answer by using the probe at U22 pin 23.

Ok

6. What logic level should EN be to enable U22?

High

Verify your answer by using the probe at U22 pin 25. OK

7. What logic level should R/$\overline{W}$ be to enable U22 during a write operation?

low.

Verify your answer by using the probe at U22 pin 21. OK

8. Read @ 1ØØC1.

9. What logic level should R/$\overline{W}$ be to enable U22 during a read operation?

High

Verify your answer by using the probe at U22 pin 21.

OK

EXERCISE #3

To verify the actual operation of U22 a program has been written. Each port will be enabled for an output and then AA will be written to the port. The probe will be used to verify each data line. After AA has been written and verified, 55 will be written and verified.

1. Why is AA and 55 used as data? (HINT: note the bit pattern.)

every other bits are opposite this will catch adjacent lines shorted.

After verification that each port is Writable we will then verify that each port is Readable using the pulse capabilities of the probe and read port.

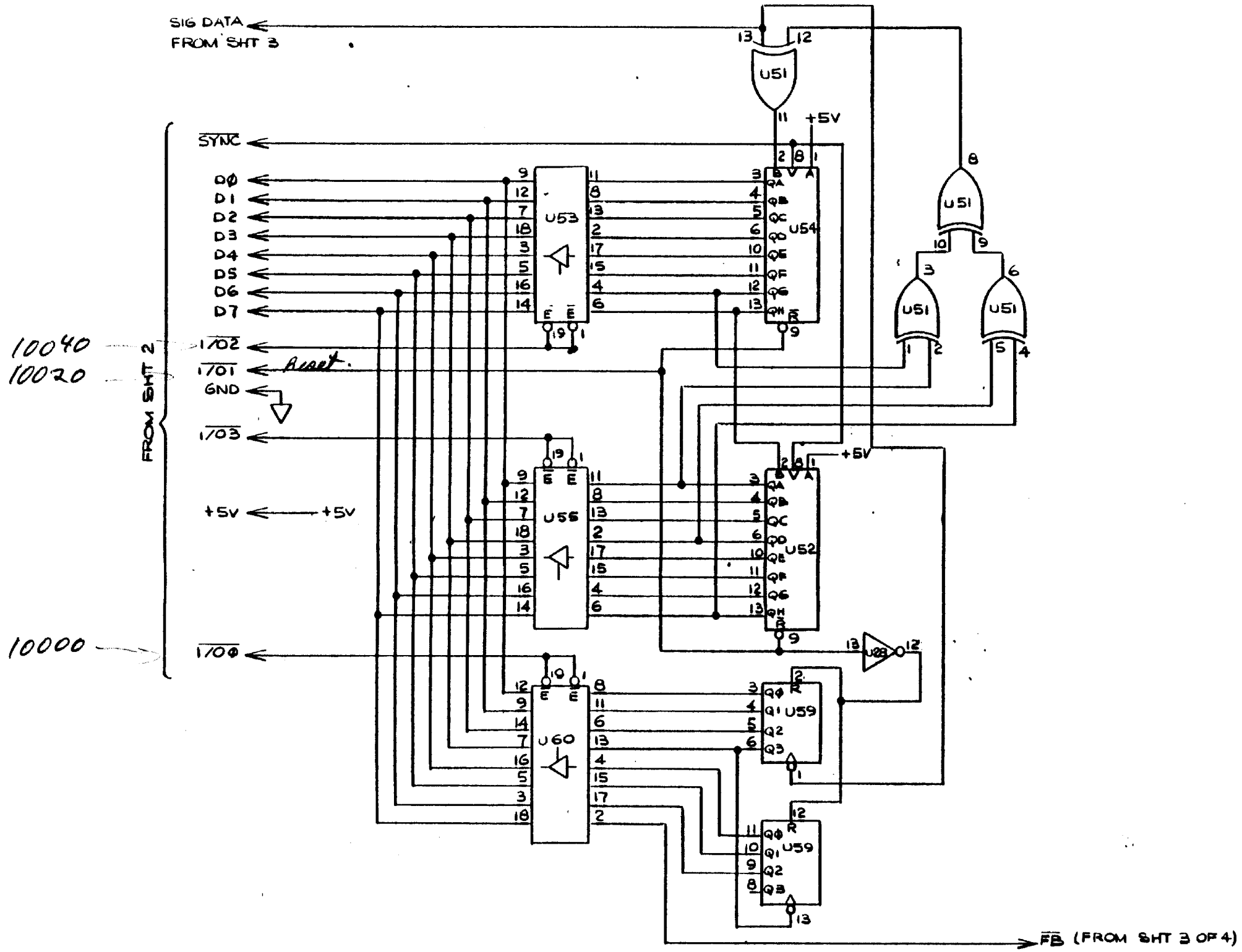
2. Execute Program 1.

```
PROGRAM 1    1077 BYTES

    SYNC FREE-RUN
    REG1 = 0
    REG2 = 1000000
    REG3 = 4000000
    DPY-PIA <U22> CHECK
    STOP
    DPY-CHECK OUTPUT AT PA0-PA7
    STOP
    DPY-ATTATCH FIXTURE TO -A1J1-
    STOP
    WRITE @ 100C1 = 0
    WRITE @ 100C0 = FF
    WRITE @ 100C1 = 4
    WRITE @ 100C0 = AA
0:  LABEL 0
    IF REG1 > 7 GOTO 1
    DPY-PLACE THE PROBE ON PA$1
    STOP
    READ PROBE
    READ PROBE
    REG0 = REG0 AND 7000000
    INC REG1
    IF REG3 > REG0 GOTO A
    DPY-PLACE THE PROBE ON PA$1
    STOP
    READ PROBE
    READ PROBE
    REG0 = REG0 AND 7000000
    INC REG1
    IF REG0 > REG2 GOTO A
    GOTO 0
1:  LABEL 1
    REG1 = 0
    WRITE @ 100C0 = 55
2:  LABEL 2
    IF REG1 > 7 GOTO 3
    DPY-PLACE THE PROBE ON PA$1
    STOP
    READ PROBE
    READ PROBE
    REG0 = REG0 AND 7000000
    INC REG1
    IF REG0 > REG2 GOTO A
    DPY-PLACE THE PROBE ON PA$1
    STOP
    READ PROBE
    READ PROBE
    REG0 = REG0 AND 7000000
    INC REG1
    IF REG3 > REG0 GOTO A
    GOTO 2
3:  LABEL 3
    REG1 = 0
    DPY-CHECK OUTPUT AT PB0-PB7
    STOP
```

Port B output

```
    WRITE @ 100C3 = 0
    WRITE @ 100C2 = FF
    WRITE @ 100C3 = 4
    WRITE @ 100C2 = AA
4:  LABEL 4
    IF REG1 > 7 GOTO 5
    DPY-PLACE THE PROBE ON PB$1
    STOP
    READ PROBE
    READ PROBE
    REG0 = REG0 AND 7000000
    INC REG1
    IF REG3 > REG0 GOTO A
    DPY-PLACE THE PROBE ON PB$1
    STOP
    READ PROBE
    READ PROBE
    REG0 = REG0 AND 7000000
    INC REG1
    IF REG0 > REG2 GOTO A
    GOTO 4
5:  LABEL 5
    REG1 = 0
    WRITE @ 100C2 = 55
6:  LABEL 6
    IF REG1 > 7 GOTO 7
    DPY-PLACE THE PROBE ON PB$1
    STOP
    READ PROBE
    READ PROBE
    REG0 = REG0 AND 7000000
    INC REG1
    IF REG0 > REG2 GOTO A
    DPY-PLACE THE PROBE ON PB$1
    STOP
    READ PROBE
    READ PROBE
    REG0 = REG0 AND 7000000
    INC REG1
    IF REG3 > REG0 GOTO A
    GOTO 6
7:  LABEL 7
    SYNC ADDRESS
    REG1 = 0
    REG4 = 1
    WRITE @ 100C1 = 0
    WRITE @ 100C3 = 0
    WRITE @ 100C0 = 0
    WRITE @ 100C2 = 0
    WRITE @ 100C1 = 4
    WRITE @ 100C3 = 4
    DPY-CHECK INPUT AT PA0-PA7
    STOP
    DPY-SET PROBE TO <PULSE-LOW>
    STOP
8:  LABEL 8
    IF REG1 > 7 GOTO B
    DPY-PLACE THE PROBE ON PA$1
```



SCHEMATIC
MAIN PCB A1

SH 4 OF 4

PROBE DATA CIRCUIT CHECK

The purpose of this check is to verify the proper operation of the shift registers and tristate buffers making up the Probe Data Circuit (U52, U53, U54, and U55).

The design of the circuit is to provide a 16-bit word that is unique to the operation being performed. This is accomplished by combining pulses from the probe (SIG DATA) and the SYNC information from the POD in the shift registers.

EXERCISE #1

Perform the following steps to verify proper operation of the Probe Data Circuit.

SYNC. A.

1. Execute Program 5 - Probe Circuit Reset.

2. Connect the UUT Probe to +5V (TP1Ø).

3. Connect the Test 9010A probe to the UUT SYNC (located on the U41 pin 9 side of R32).

4. Read @ 1ØØ2Ø - Reset (Ignore Response.)

5. Select PULSE LOW on the Test 9010A.

6. Read @ 1ØØ4Ø .

   a. What is the response?

   ØØ

   b. If your response was not Ø you most likely have a problem in the circuitry involving U53 and U54. Disconnect the Test probe from the UUT and SYNC the Test probe to FREE-RUN. Your Test probe now becomes a Logic Probe. This now allows you to check the data from U53 and U54.

   c. If your response was zero, continue to step 7.

7. Press REPEAT.

   What is the response?

8. Continue to Press REPEAT and watch the Display value increase with each READ.

   Why doesn't the Read value increase in a 1, 2, 3 ... fashion?

   00 0000
   00 0001
   00 0011
   00 0111
   00 1111 etc
   01 1111

9. When the display reads FE perform a Read @ 10060.

10. After initiating a Read @ 10060, repeat the process performed with steps 7 and 8.

    0001 0041

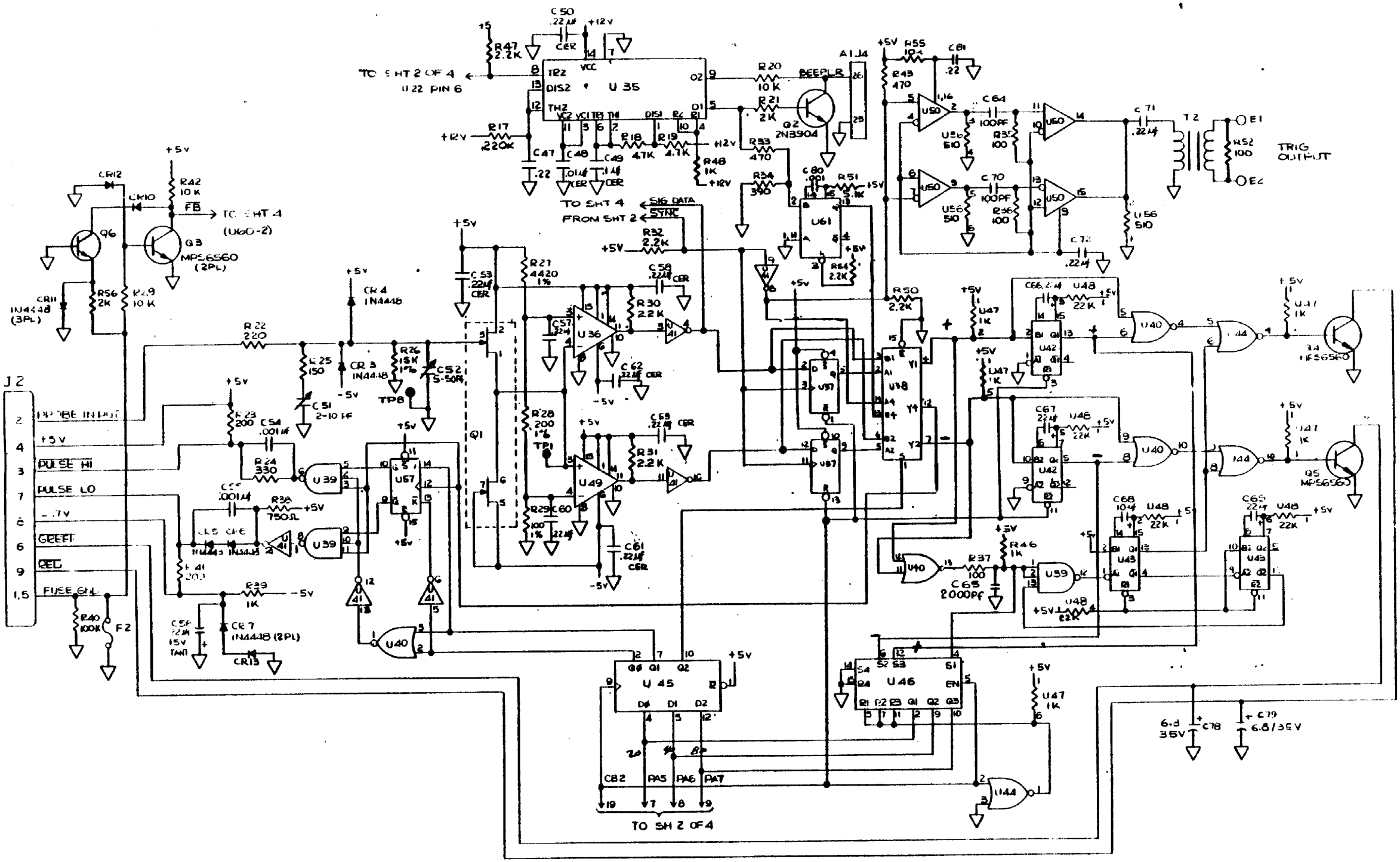
11. Any response other than expected should be checked as in step 6.

PROBE COUNT CIRCUIT

To verify the operation of the Probe Count Circuit, U59 and U60, you will be asked to write a program. You will be asked to accomplish this feat at the end of the session.

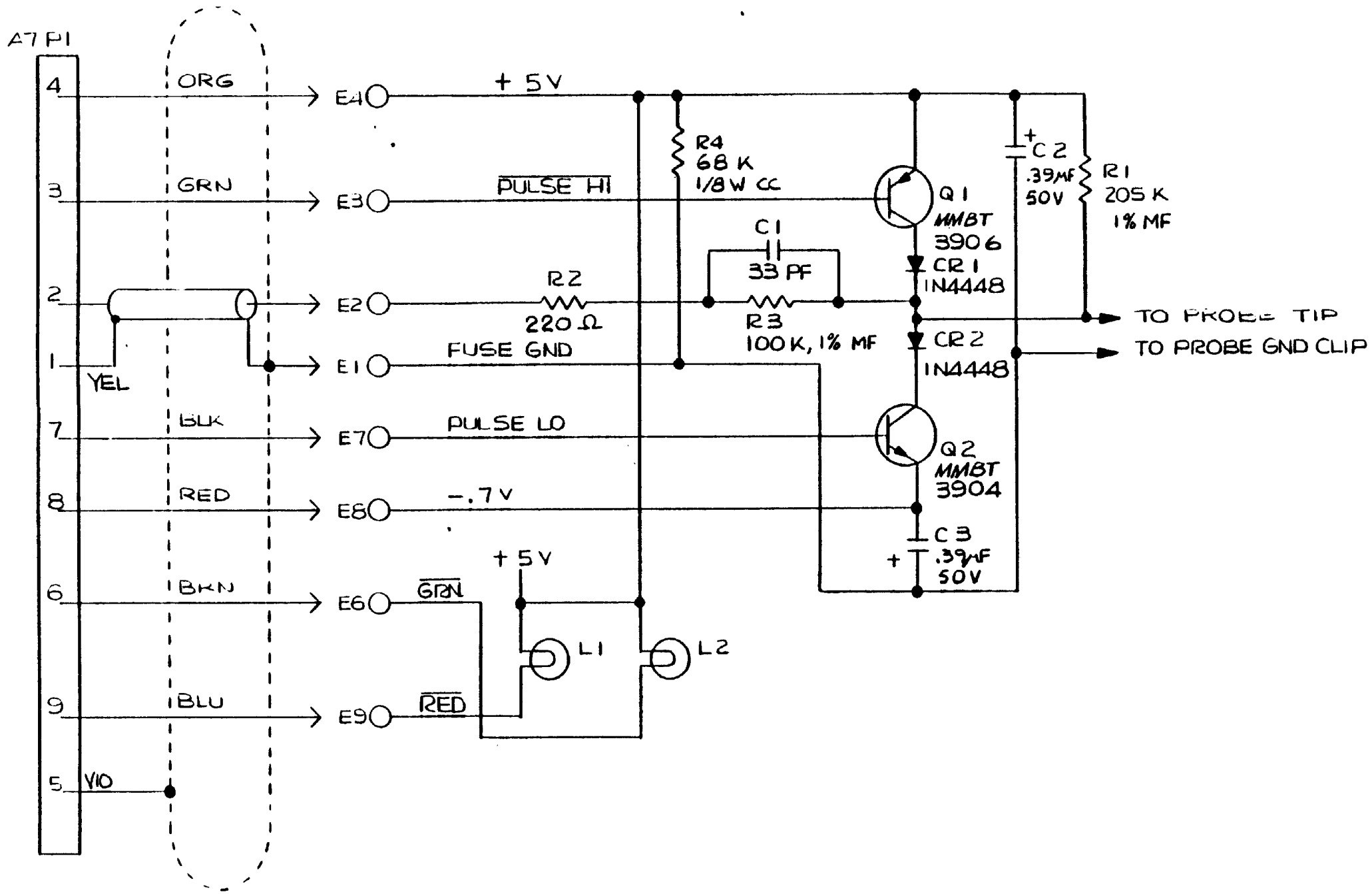
```
PROGRAM 5    111 BYTES

    DPY-PROBE CIRCUIT RESET
    STOP
    WRITE @ 100C1 = 0
    WRITE @ 100C0 = FF
    WRITE @ 100C1 = 4
    WRITE @ 100C0 = 80
    WRITE @ 100C3 = 38
    WRITE @ 100C3 = 30
    READ @ 10020
    DPY-PROBE RESET COMPLETE
```



SCHEMATIC — PCB, MAIN A1

SH 3 OF 4



SCHEMATIC —
PROBE ASSY A7

PROBE CIRCUITRY TESTS

The Probe Circuit consists of an analog input and digital routing circuitry. This may be found on the MAIN PCB (A1) schematic sheet 3 of 4. The Probe Circuitry also consists of a Read Probe Data circuit on sheet 4 of 4 on the MAIN PCB (A1) schematic.

The Probe will operate in either FREE-RUN or SYNC mode to input or output probe data. Testing of the Probe Circuitry will be accomplished with 3 exercises all under program control: Free-Run Probe Input Check, Sync'd Probe Input Check, and Pulse Output Check.

EXERCISE #1

Free-Run Probe Input Check - PROGRAM 2

1. This program will set-up the PIA to control U45 and through it U38. The truth table for U38, a Quad 2 - Input Multiplexer is $\overline{E}\cdot(B\cdot S+A\cdot S)$.

2. The program also allows the Test 9010A to read the information received from the probe by enabling port A of U22 to input information from U46.

3. The use of the program requires the UUT Probe to be connected for information to be inputed to the Probe Circuitry. When the program is running the UUT Probe may be connected to TP2 or to +5V on the UUT for LOW and HIGH inputs respectively.

4. Because the program is a continuous loop, the probe may be used as a Logic Probe when an incorrect condition is seen.

5. Execute Program 2.

EXERCISE #2

Sync'd Probe Input Check - PROGRAM 3

1. Program 3 will be used to verify the operation of the circuitry that allows the 9010A user to monitor activity as it relates to the UUT's Microprocessor Address or Data Bus. The program is a continuous loop which requires the following connections for proper operation.

2. Connect the Test 9010A ~~Trig Output~~ Probe to SYNC on the UUT (~~Center conductor~~ Gnd clip of probe to TP2 GND and the ~~shield~~ Probe tip to the U41 side of R32).

~~3. Connect the Test probe to the UUT probe.~~ Execute Program #3

4. Set the Test probe to pulse ~~High~~ LOW and record your observations.

5. Touch the unit Probe to Hi/lo and lows and record results.

5. Set the ~~Test probe to pulse Low and record your observations~~.

You see what data is at the Probe. as a type pulse arrives.

6. ~~Set the Test~~ probe to no pulse and record your observations.

UUT Probe is latched and last data it had while the Test 9010A was pulsing

What condition does this indicate?

Tests 4 thru 6 verify the operation of U37 and U38.

7. Execute Program 3.

EXERCISE #3

Pulse Output Check - PROGRAM 4

1. This program verifies the operation of the Pulse circuitry. The program sets up the PIA for output to control U45 which in turn enables U57, U37, U40 and U41. It also reads the output from the probe circuit and displays the pulse level on the Test 9010.

2. To verify proper probe output, connect the Test probe to the UUT probe and observe the lights during the time the program is running.

3. Execute Program 4.

PROBE INDICATOR LIGHT ACTIVITY

CONDITION	DESCRIPTION
Green on continuously, red off	Indicates a steady dc low.
Red on continuously, green off	Indicates a steady dc high.
Both lights off	Indicates the line remains in the tristate logic continuously.
Both red and green on continuously	Indicates that a line is toggling between high and low, but is staying in the tristate area for less than 100 ns. An example of this would be a clearly defined square wave.
Green flashing, red off	Indicates the line is toggling in between logic low and tristate.
Red flashing, green off	Indicates the line is toggling in between logic high and tristate.
Both lights flashing	Indicates the line is toggling between all three logic states.

PROGRAM 2 257 BYTES

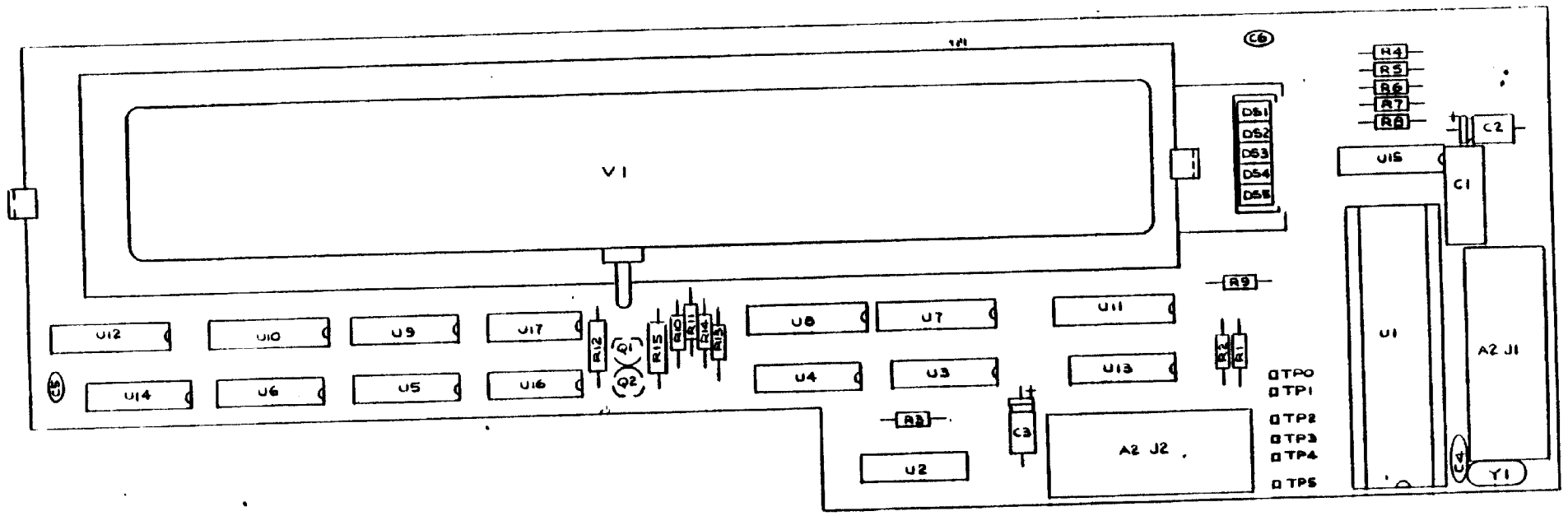
```
   DPY-FREE-RUN PROBE CHECK
0: LABEL 0
   WRITE @ 100C3 = 30
   WRITE @ 100C1 = 0
   WRITE @ 100C0 = FF
   WRITE @ 100C1 = 4
   WRITE @ 100C0 = 80
   WRITE @ 100C3 = 38
   WRITE @ 100C1 = 0
   WRITE @ 100C0 = 0
   WRITE @ 100C1 = 4
   READ @ 100C0
   REGE = REGE AND E0
   IF REGE = 20 GOTO 1
   IF REGE = 40 GOTO 2
   IF REGE = 80 GOTO 3
   GOTO 0
1: LABEL 1
   DPY-TRISTATE INPUT <LIGHTS OFF>
   GOTO 0
2: LABEL 2
   DPY-NEGATIVE INPUT <GREEN ON>
   GOTO 0
3: LABEL 3
   DPY-POSITIVE INPUT <RED ON>
   GOTO 0
```

PROGRAM 3 86 BYTES

```
   DPY-SYNC PROBE CHECK
   SYNC FREE-RUN
   WRITE @ 100C3 = 30
0: LABEL 0
   WRITE @ 100C1 = 0
   WRITE @ 100C0 = FF
   WRITE @ 100C1 = 4
   WRITE @ 100C0 = [redacted]0
   WRITE @ 100C3 = 38
   GOTO 0
```

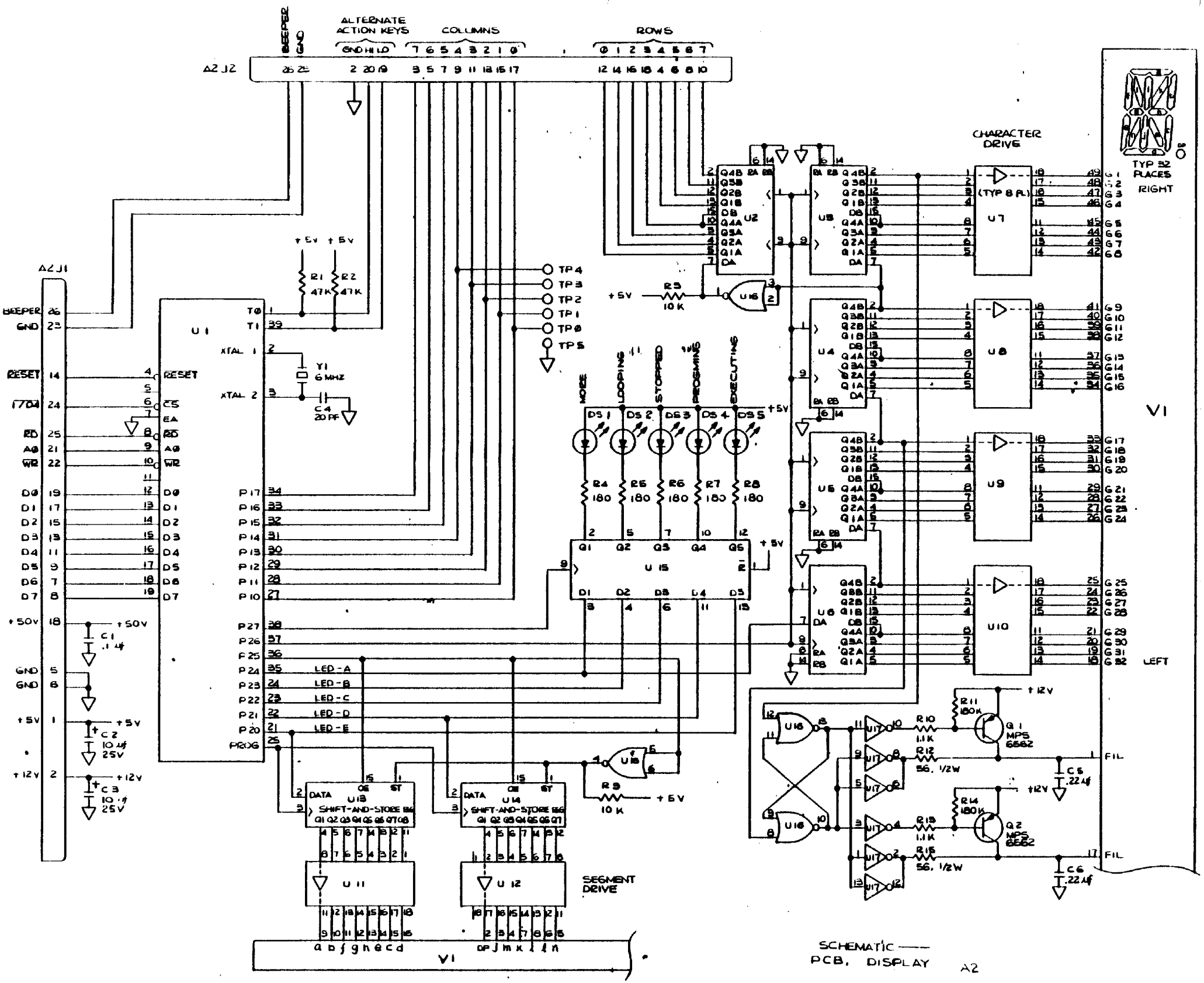
PROGRAM 4 437 BYTES

```
   DPY-PULSE OUTPUT CHECK
   STOP
0: LABEL 0
   REG1 = 0
   WRITE @ 100C3 = 30
   WRITE @ 100C1 = 0
   WRITE @ 100C0 = FF
   WRITE @ 100C1 = 4
   WRITE @ 100C0 = 10
   DPY-#
   WRITE @ 100C0 = A0
   WRITE @ 100C3 = 38
   GOTO 3
1: LABEL 1
   WRITE @ 100C3 = 30
   WRITE @ 100C1 = 0
   WRITE @ 100C0 = FF
   WRITE @ 100C1 = 4
   WRITE @ 100C0 = C0
   WRITE @ 100C3 = 38
   GOTO 3
2: LABEL 2
   WRITE @ 100C3 = 30
   WRITE @ 100C1 = 0
   WRITE @ 100C0 = FF
   WRITE @ 100C1 = 4
   WRITE @ 100C0 = E0
   WRITE @ 100C3 = 38
   GOTO 3
3: LABEL 3
   WRITE @ 100C1 = 0
   WRITE @ 100C0 = 0
   WRITE @ 100C1 = 4
   READ @ 100C0
   INC REG1
   REGE = REGE AND E0
   IF REGE = 60 GOTO 4
   IF REGE = A0 GOTO 5
   IF REGE = E0 GOTO 6
   GOTO 0
4: LABEL 4
   DPY-PULSE LOW <GREEN>
   IF REG1 = 1 GOTO 1
   IF REG1 = 2 GOTO 2
   GOTO 0
5: LABEL 5
   DPY-PULSE HIGH <RED>
   IF REG1 = 1 GOTO 1
   IF REG1 = 2 GOTO 2
   GOTO 0
6: LABEL 6
   DPY-PULSE HIGH AND LOW <BOTH>
   IF REG1 = 1 GOTO 1
   IF REG1 = 2 GOTO 2
   GOTO 0
```

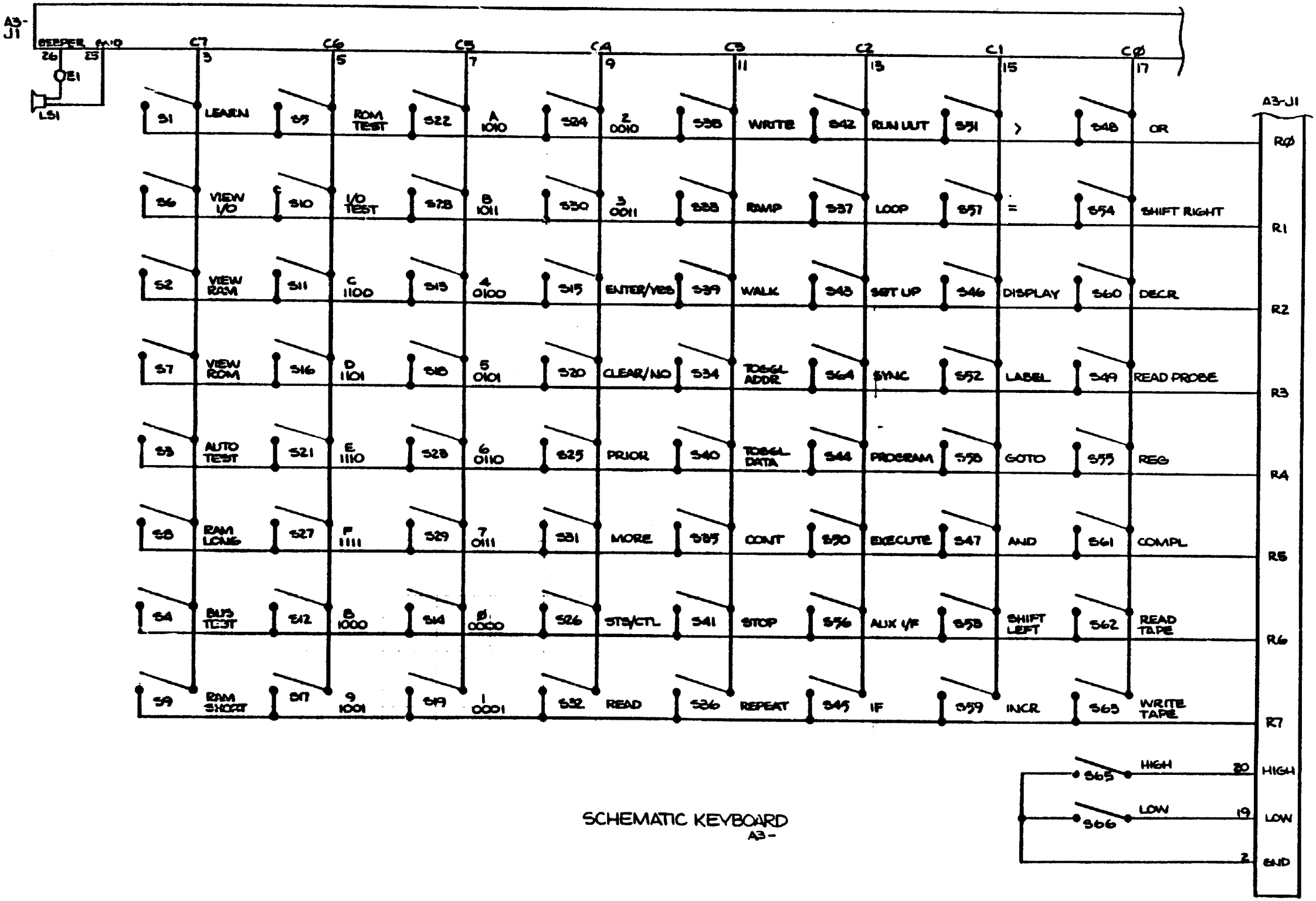


REFERENCE DESIGNATION

DISPLAY A2



SCHEMATIC —— PCB, DISPLAY A2



SCHEMATIC KEYBOARD
A3-

DISPLAY/KEYBOARD TEST

The Display and Keyboard have on-board test programs which provide the unique opportunity to completely isolate the two sub-sections from the mainframe. There are 5 test programs which are easily accessible by jumpering TP5 (LOW) to the following listed Test Points.

*TPØ - Display Segment Test - Each display segment will light in turn. The test can be used with the Test 9010A probe to verify Segment Data is getting to the display from U11, 12, 13, and U14. The Test 9010A probe should be Sync'd to FREE-RUN.

1. Short TPØ to TP5 and verify that the Display Segment Test does occur.

*TP1 - Digit Drivers Test - The test initiates a walking character across the display. Use the 9010A Test Probe to verify the operation of Digit Drivers U3, 4, 5, 6, 7, 8, 9, and U10.

1. Short TP1 to TP5 and observe the process that occurs.

*TP2 - LED Test - This test will cause the MORE, STOPPED, and EXECUTING LED's to come on when the PULSE HIGH is pressed. The LOOPING and PROGMING LED's to come on when PULSE LOW is pressed. U15 may be checked by using the Test 9010A probe.

1. Short TP2 to TP5 and observe the process that occurs.

*TP3 - Keyboard Test - When the test is initiated a test pattern is displayed which is illustrated on 4-6. When a KEY is pressed a portion of the display will go out. When all keys have been pressed the display will be blank.

If a key is shorted, a part of the pattern corresponding to that key will be missing on initialization. If 2 or more keys are shorted together, 2 or more pieces of the pattern will go out when 1 key is pressed.

1. The illustration of the Display test pattern has two components left out: Key and Read Data.

2. Initiate the Keyboard Test. In the appropriate space next to a specific test pattern segment, write the name of the key that makes that segment go out.

3. When step 2 is completed, use the Test 9010A to Read @ the display address (1ØØ8Ø). LOOP the Read. Now by pressing a key, a HEX number will appear on the Test 9010A display. This is the Read Data. Fill in the blanks next to the appropriate key.

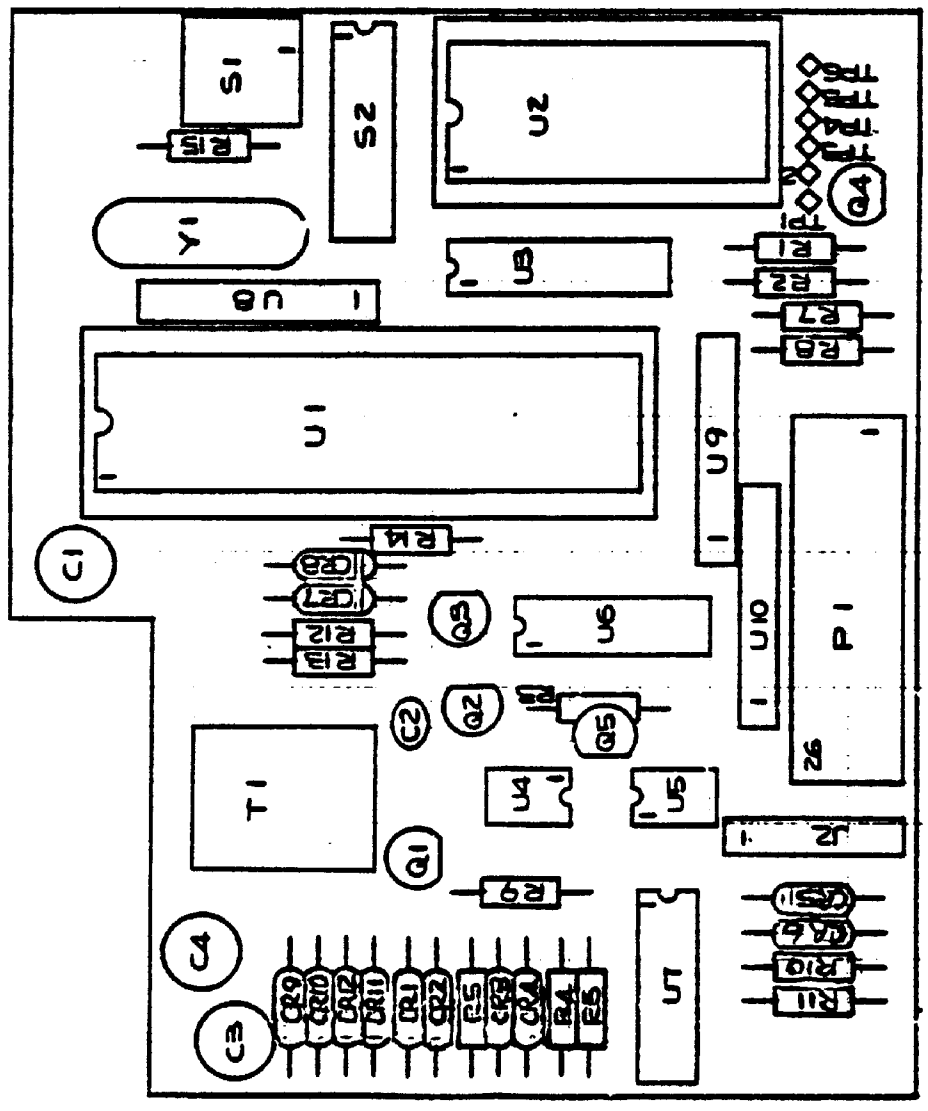
*TP4 - Reset - Will reset the display.

To verify the communication between the main microprocessor and the display processor, execute PROGRAM 7.

READ DATA	KEY	LEFT SIDE OF DISPLAY		KEY	READ DATA
10	Learn	S1 →	← S6	I/O View.	12
11	RAM View	S2 →	← S7	ROM View	13
14	Auto Test	S3 →	← S8	RAM Long	17
15	Bus Test	S4 →	← S9	RAM Short.	18
16	Rom Test	S5 →	← S10	I/O Test	19
0C	C	S11 →	← S16	D	0D
0E	E	S21 →	← S27	F	0F
08	8	S12 →	← S17	9	09
0A	A	S22 →	← S28	B	0B
04	4	S13 →	← S18	5	05
06	6	S23 →	← S29	7	07
00	0	S14 →	← S19	1	01
02	2	S24 →	← S30	3	03
1C	Enter	S15 →	← S20	Clear	1D
1A.	Prior	S25 →	← S31	More	1B
1E	STS/CTL	S26 →	← S32	Read	1F
20	WRITE	S38 →	← S33	RAMP	21
22	WALK.	S39 →	← S34	TOGGLE Address	23
24	TOGGLE Data	S40 →	← S35	Cont	25
28	STOP	S41 →	← S36	Repeat.	26
29	Run UUT.	S42 →	← S37	Loop	27
30	SETup.	S43 →	← S64	SYNC	3C
37	Execute	S50 →	← S44	Program.	2A.
2D	=F	S45 →	← S56	Aux IF.	3F
2F	=	S57 →	← S51	>	2E
2B	Lable	S52 →	← S46	Display	3E
30	AND	S47 →	← S58	Goto	2C
34	INCR.	S59 →	← S53	Shift Left.	32
33	Shift. Right	S54 →	← S48	OR	31
39	Read Probe	S49 →	← S60	DECR.	35
36	COMPL	S61 →	← S55	REG	38
3B	WRITE TAPE.	S63 →	← S62	Read tape	3A.

PROGRAM 7 152 BYTES

```
    REG2 = 0
    DPY-DISPLAY TEST
    STOP
0:  LABEL 0
    REG1 = 0
    WRITE @ 10080 = BA
    INC REG2
    IF REG2 = 20 GOTO 2
1:  LABEL 1
    WRITE @ REGF = 20
    INC REG1
    IF REG1 = 20 GOTO 0
    GOTO 1
2:  LABEL 2
    REG2 = 0
3:  LABEL 3
    WRITE @ 10080 = 20
    INC REG1
    IF REG1 = 1E GOTO 4
    GOTO 3
4:  LABEL 4
    WRITE @ 10080 = BA
    REG1 = 0
    INC REG2
    IF REG2 = 1F GOTO 5
    GOTO 3
5:  LABEL 5
    REG2 = 0
    GOTO 0
```



REFERENCE DESIGNATIONS
RS-232 A5

S2 POSITION	BAUD RATE
0	110
1	150
2	300
3	600
4	1200
5	2400
6	4800
7	9600
8	110
9	150

SCHEMATIC — RS 232

RS232C TEST

The RS232C interface may be tested by looping the SERIAL OUT to the SERIAL IN, then Writing and Reading to the interface using the Test 9010A. Data for the output will be written at address 1ØØE3 and read at 1ØØE1. The status of the UART can be read at 1ØØE2. PROGRAM 8 will do this and check for errors. If a data error occurs, the data may be traced from the SERIAL OUTPUT port to the SERIAL IN port using a scope. The tracing of the data with a scope can be accomplished by doing a looping WRITE @ 1ØØE3=55.

The BAUD RATE switch (S2), BAUD RATE GENERATOR (U2) and CONTROL LOGIC (U3) should also be checked with a scope.

The frequency affected by S2 should be checked with a scope at TP6.

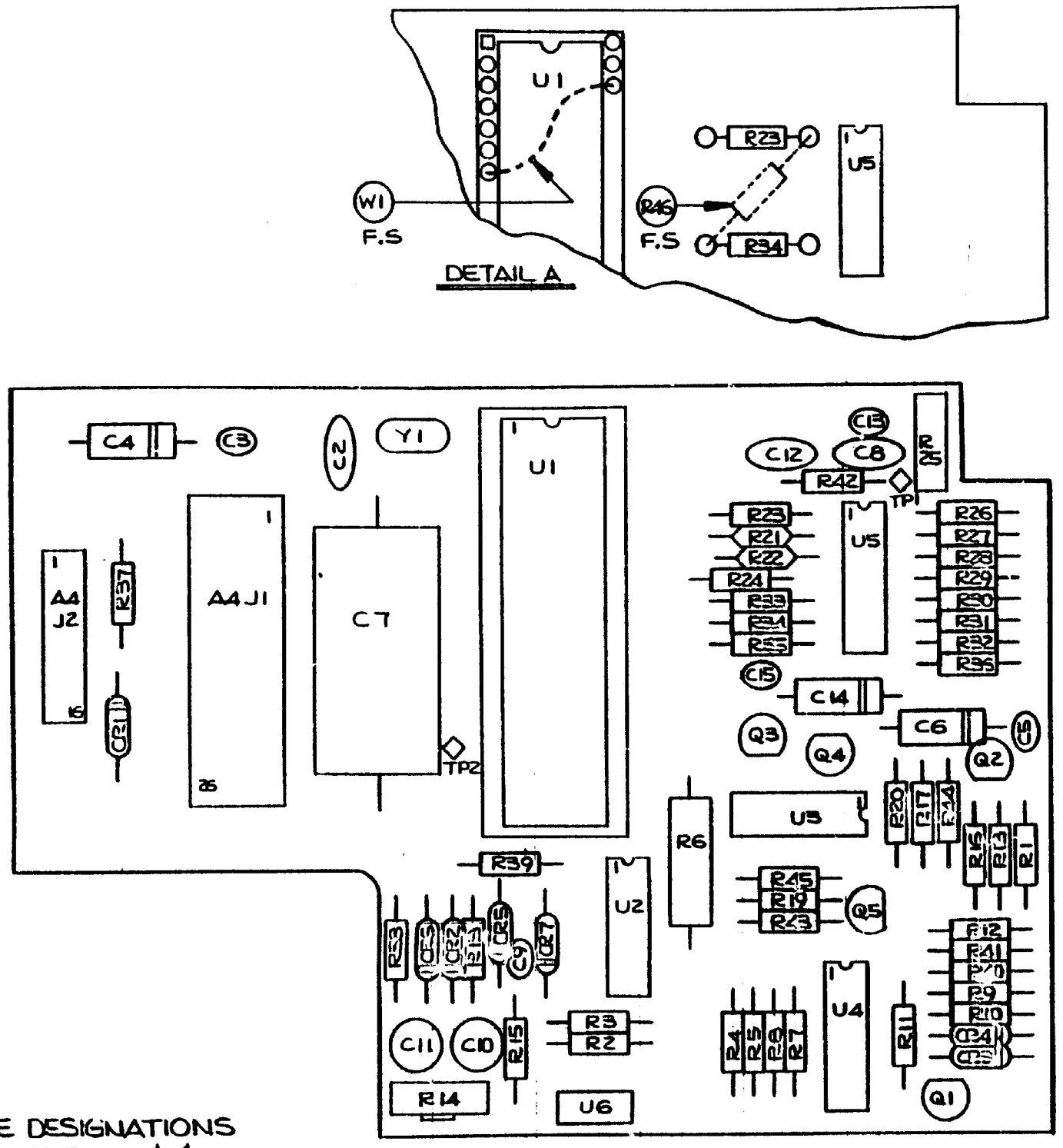
NOTE: DO NOT CROSS THE GUARD CIRCUIT USING THE TEST 9010A PROBE. THIS WILL DAMAGE U1.

1. Execute PROGRAM 8.

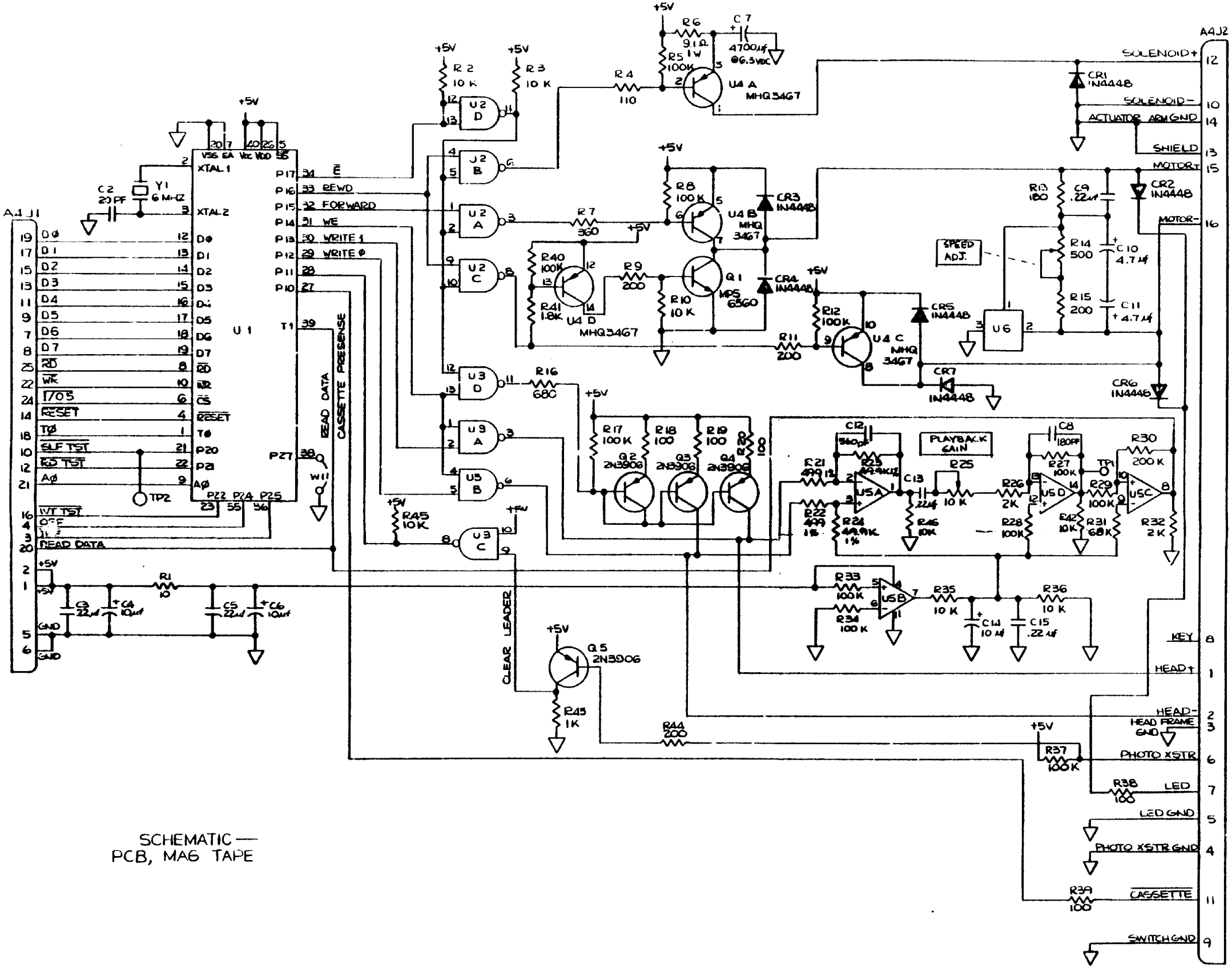
Baud	Freq
110	1.76 KHz.
150	2.4
300	4.8
600	9.6
1200	19.2
2400	38.4
4800	76.8
9600	~~96.15~~ 153.6

```
PROGRAM 8    884 BYTES

    DPY-RS-232 TEST
    STOP
    DPY-SET PARITY ON
    STOP
    DPY-SET STOP BITS TO 1
    STOP
    DPY-SET 7 OR 8 BIT TO 8 BIT
    STOP
    DPY-SET EVEN OR ODD TO ODD
    STOP
    DPY-SET BAUD RATE TO 9600
    STOP
    WRITE @ 100E3 = 0
    READ @ 100E1
    WRITE @ 100E3 = 0
    READ @ 100E1
    REG1 = 18
    REG2 = 7
    REG3 = 55
0:  LABEL 0
    READ @ 100E2
    IF REG2 AND REGE > 0 GOTO 6
    IF REG1 AND REGE = 10 GOTO 1
    IF REG1 AND REGE = 0 GOTO 7
    IF REG1 AND REGE = 18 GOTO 2
    IF REG1 AND REGE = 8 GOTO 8
1:  LABEL 1
    WRITE @ 100E3 = REG3
    READ @ 100E2
    IF REG1 AND REGE = 10 GOTO 5
    GOTO 0
2:  LABEL 2
    READ @ 100E1
    DPY-DATA RECEIVED = $E
    IF REGE AND REG3 = 55 GOTO 3
    IF REGE AND REG3 = AA GOTO 4
    DPY-DATA RECEIVED BAD
    STOP
    GOTO 5
3:  LABEL 3
    REG3 = AA
    READ @ 100E2
    IF REG1 AND REGE = 18 GOTO 5
    GOTO 0
4:  LABEL 4
    REG3 = 55
    READ @ 100E2
    IF REG1 AND REGE = 18 GOTO 5
    GOTO 0
5:  LABEL 5
    REG4 = 7000000
    SYNC ADDRESS
    DPY-CONNECT PROBE TO U1-23
    STOP
    READ PROBE
    READ PROBE
    WRITE @ 100E3 = 0
    READ PROBE
    IF REG4 AND REG0 > 4000000 GOTO 9
    IF 4000000 > REG4 AND REG0 GOTO 9
    READ PROBE
    READ @ 100E1
    READ PROBE
    IF REG4 AND REG0 > 1000000 GOTO 9
    DPY-CONNECT PROBE TO U1-4
    STOP
    READ PROBE
    READ PROBE
    WRITE @ 100E3 = 0
    READ PROBE
    IF REG4 AND REG0 > 1000000 GOTO 9
    READ PROBE
    READ @ 100E1
    READ PROBE
    IF REG4 AND REG0 > 4000000 GOTO 9
    IF 4000000 > REG4 AND REG0 GOTO 9
    DPY-CONNECT PROBE TO U1-16
    STOP
    READ PROBE
    READ PROBE
    WRITE @ 100E3 = 0
    READ PROBE
    IF REG4 AND REG0 > 1000000 GOTO 9
    READ PROBE
    READ @ 100E2
    READ PROBE
    IF REG4 AND REG0 > 4000000 GOTO 9
    IF 4000000 > REG4 AND REG0 GOTO 9
    GOTO A
6:  LABEL 6
    DPY-UART ERROR-REPLACE
    GOTO B
7:  LABEL 7
    DPY-TBR FULL - NO DATA REC.
    STOP
    DPY-REPLACE UART
    GOTO B
8:  LABEL 8
    DPY-TBR FULL - DATA REC.
    STOP
    DPY-REPLACE UART
    GOTO B
9:  LABEL 9
    DPY-REPLACE U6
    GOTO B
A:  LABEL A
    DPY-TRACE SERIAL DATA ON SCOPE
B:  LABEL B
```



REFERENCE DESIGNATIONS
MAG TAPE, A4



SCHEMATIC — PCB, MAG TAPE

TAPE DECK TEST

All TAPE DECK TESTS are accomplished by jumpering U31-14 to U31-15. The Tape Deck has three built-in test routines available. These tests, which are listed and defined below, are used for setting up and troubleshooting the deck.

TEST ROUTINES

SELF TEST - Will first rewind, then write one thousand words of known data onto the tape. Rewinds, then reads the tape. Compares read data with known write data, then rewinds and repeats as long as there are no errors. Will stop the tape and drops dead on any error.

READ TEST - Attempts rewind first. Then puts the deck into a forward read mode. If a cassette is installed, will rewind again and continue when EOT is sensed. If no cassette is installed, it will continue in forward read mode until a hardware reset is hit.

WRITE TEST - Same as READ TEST except continually writes the hex word 'CA' in ratio format with IWH's between words.

The deck requires two adjustments be made for proper operation: Speed Adjustment (R14), and Playback Gain Adjust (R25).

To perform the Speed Adjustment a strobe disk is required. Two disks are available: 50Hz, and 60Hz.

1. Place probe disk on the left spindle of the deck.

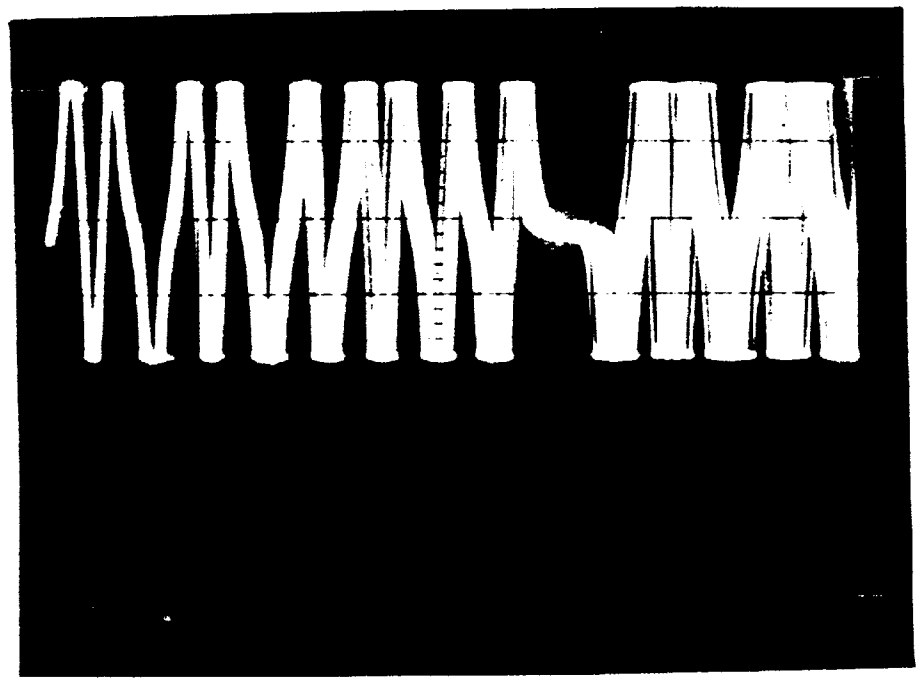
2. Momentarily connect a jumper between U1-22 (READ TEST) and ground (negative side of C7).

3. Observe the disk and adjust R14 so that it appears as close to motionless as possible.

   NOTE: There may be some instability, however, it should be possible to adjust R14 so that the disk appears motionless.

There are two methods to set the Playback Gain. One is to initiate the Self Test by grounding TP2. This will write a small amount of data, then read back the data which may be checked at TP1.

1. Using a scope, adjust the gain (R25) as illustrated.



The second method is to initiate a Write Test by jumpering U1-23 to ground and wait for the tape to fill with data. When the tape starts to rewind initiate a reset at U1-4 by pulling it low. Initiate a Read Test (U1-22 to ground) and adjust the gain using a scope at TP1 and R25.

Other Tape Deck Tests

1. Use the Write Tests and Read Tests to trace signals to and from the tape head.

2. Use the Self Test to verify the Tape Read and Write operations. A failure will cause the test to stop.

3. To check communication to the MAIN PCB, perform the following operations in Immediate Mode and note the response of the Tape Deck.

READ @ 1ØØA1	This produces a Status Message. Ignore the message.
WRITE @ 1ØØA1=11	Tape rewinds then moves forward. write tape
WRITE @ 1ØØA1=12	Tape rewinds then moves forward. read tape
READ @ 1ØØAØ	Read any data put into buffer. Ignore data.

THE TEST

Together U59 and U60 form a counter circuit.

With the following information write a program that verifies the operation of U59 and U60.

1. Sig Data required to clock counter pulse HIGH.

2. Reset Counter Address is 1ØØ2Ø.

3. To Read Data Address 1ØØØØ.

PROGRAM 6, which is located on page 5-2, is a sample program that verifies U59 and U60.

```
PROGRAM 6   240 BYTES

   REG1 = 80
   DPY-PROBE COUNT CIRCUIT CHECK
   STOP
   DPY-CONNECT TEST PROBE TO U59-1
   STOP
   DPY-SET PROBE TO PULSE HIGH
   STOP
   READ @ 10020
0: LABEL 0
   READ @ 10000
   DPY-DATA READ = $E
   INC REG1
   IF REG1 = FF GOTO 1
   IF REGE = REG1 GOTO 0
   DPY-DATA = $E SHOULD BE $1
   STOP
   DPY-CHECK U59 AND U60
   STOP
   DPY-USE TEST PROBE
   STOP
1: LABEL 1
   DPY-END OF PROBE COUNTER CHECK
```