# µscope 820

MICROPROCESSOR SYSTEM CONSOLE

# operator's handbook

MANUAL ORDER NO. 9800526B

intel®

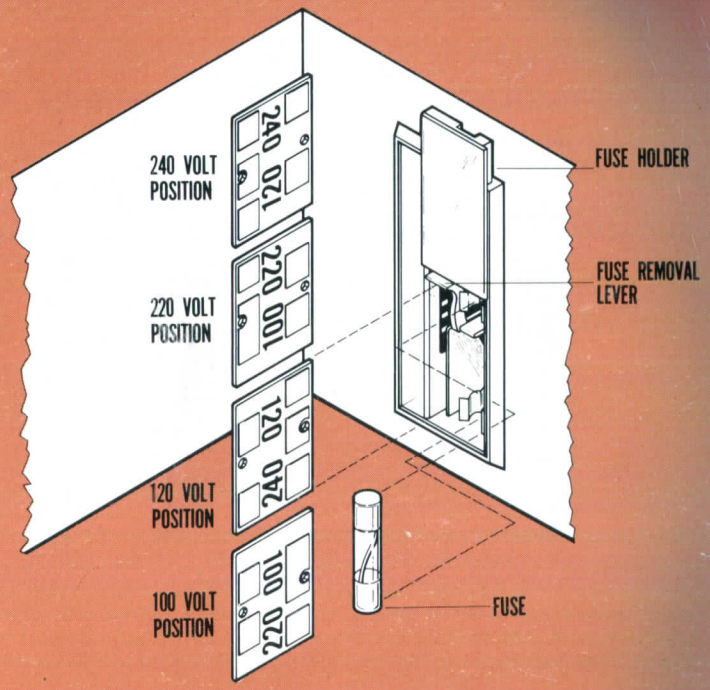
# Caution

The information stored in EPROMs may be damaged if subjected to X-ray exposure. If the unit is to be inspected as part of airport or other security measures, insist that it be inspected visually.

## Line Voltage Selection

The μSCOPE 820 Microprocessor System Console can operate with power inputs of 100, 120, 220, and 240 V, 48 to 63 Hz. Power selection is accomplished by positioning a 4-way printed-circuit card located in the line connector module. An additional fuse for 220 and 240 V is supplied with the unit. Prior to connecting the μSCOPE 820 to facility power, make sure the unit is equipped with the proper fuse and is compatible with facility power by performing the following procedures:

- Open storage compartment door and remove ribbon cable and personality module.
- Slide fuse holder door up to gain access to fuse and power selection printed circuit card.
- Lift fuse removal lever and remove fuse.
- Using a hook or angle needle-nose pliers, remove power selection printed circuit card. Fuse removal lever must be held in the up position.
- Position the power selection printed circuit as shown in the following figure to match your facility power and reinsert into position.
- Lower the fuse removal lever and install the proper fuse.
- Use the 1.25 ampere fuse for 100 or 120 volt facility power or the 0.70 ampere fuse for 220 or 240 volt facility power.




**WARNING**

For safety reasons the μSCOPE 820 is equipped with a 3-prong grounded power connector. Do not defeat this safety feature.

# µscope 820

**MICROPROCESSOR SYSTEM CONSOLE**

# operator's handbook

**MANUAL ORDER NO. 9800526B**

# Preface

This operator's handbook has been prepared for a wide range of operators who have varying degrees of experience with microprocessors and microprocessor- based systems. The information in this manual is organized in a step-by-step way to familiarize you with the operation of the μSCOPE 820 and some of its many applications. No doubt many of you will find additional applications for this powerful analytical tool.

Operators should be familiar with the contents of this handbook and the operation of the system being tested. However, those of you who are not microprocessor experts will also find the μSCOPE 820 an easy tool to use.

This handbook is organized into three chapters. Chapter 1 introduces the μSCOPE 820 and describes what it is, what it does and, briefly, how it does it. Chapter 2 contains operating information that describes how to connect the μSCOPE 820 to a system, the operating controls and indicators, a brief operating exercise, and how to perform many functions. Chapter 3 describes the μSCOPE 820 front panel EPROM/ROM format to aid in the writing of diagnostic subroutines.

This manual assumes the System Under Test contains an Intel 8080A microprocessor. Information required for other microprocessors will be provided in manual supplements. Information required to service and maintain the μSCOPE 820 will be provided by a service manual.

# Table of Contents



# List of Illustrations


intel
iscope 820
intel
8080A
iscope
probe

# Chapter 1
# Introducing The μSCOPE 820

## What Is The μSCOPE 820?

The μSCOPE 820 Microprocessor System Console is a portable, self-contained test instrument that helps you control, monitor and interact with an 8-bit microprocessor-based system. Since it is packaged in a standard-size briefcase that provides storage for all required accessories, the μSCOPE 820 can be easily carried to any location. Upon opening the case, you will find everything necessary to quickly and effectively evaluate the performance of an 8-bit microprocessor-based system.

The μSCOPE 820 is most easily described as three instruments combined into a single convenient test instrument.

- A minicomputer front panel
- A microprocessor analyzer
- A substitute program memory

The μSCOPE 820 front panel provides many of the controls and indicators typically found on a minicomputer front panel. With these controls and indicators, you can fully control the System Under Test, including the operating mode of the CPU. You can halt the CPU, single step it, run in real time, or run with system data periodically collected and displayed or modified. You can examine any of the System Under Test memory locations. If memory, or any portion of memory, is resident in system RAM, this memory can also be examined and modified. In addition, you can examine or modify any system I/O port or CPU working register.

The μSCOPE 820 provides a 32-bit breakpoint function, a 256-machine-cycle trace memory and a substitute program memory for the System Under Test. When used in conjunction with the substitute memory, these diagnostic capabilities can modify normal system program execution or record historical data during execution. The substitute memory can also be used to execute special diagnostic or maintenance routines without altering the System Under Test program.

## What Does The μSCOPE 820 Do?

The μSCOPE 820 is a powerful, flexible tool that supports various microprocessors, microprocessor-based systems and microprocessor-based end products. It can be used to maintain and troubleshoot microprocessor-based equipment in the laboratory, on the factory floor, in field depots or at customer sites—anywhere a microprocessor is found.

The μSCOPE 820 contains its own microprocessor, so sophisticated troubleshooting techniques are easy to use. This versatile test instrument can support a multitude of tasks associated with microprocessors.

- Basic product engineering, troubleshooting, debugging and design verification
- Incoming inspection, source inspection and functional testing
- Manufacturing final testing
- Assistance during field installation
- Troubleshooting hardware, firmware and software field-reliability problems
- Status of operational end products (periodic preventive maintenance)
- Verification of field modifications
- Verification of special customer options or modifications

As you use the μSCOPE 820 and experience its control over in-system CPUs, you will find many more applications and uses for this sophisticated and easy to use tool.

## How Does The μSCOPE 820 Function?

The μSCOPE 820 contains an 8085 microprocessor and its associated keyboard, display, RAM, ROM, and I/O. The 8085 and these circuits are the master control of the μSCOPE 820. The System Under Test CPU operates as a slave to the μSCOPE 820 master CPU. Firmware in

the personality module programs the 8085 of the μSCOPE 820 to configure the console for operation with a particular 8-bit microprocessor. Figure 1 is a functional block diagram of the μSCOPE 820.

During operation, the 8085 master CPU loads instructions into the μSCOPE 2-port RAM. The master CPU then uses the control logic of the μSCOPE probe to force the System Under Test slave CPU to execute these instructions. The remaining circuits consist of the following:

- Data transceivers
- Address receivers
- Control receivers and drivers
- Breakpoint and pass-count circuits
- Trace-memory circuits
- Data and address multiplexing circuits
- 2-port RAM
- Front panel PROM

All circuits with the exceptions of the front panel PROM and interface circuits are under direct control of the 8085.

The breakpoint circuits consist of a 32-bit breakpoint condition register, a 32-bit breakpoint mask register and an 8-bit pass-count register. When establishing a breakpoint, the master CPU sets the registers and when a breakpoint match occurs, the master CPU suspends the System Under Test CPU or forces the CPU to run a routine established by the master. This routine can reside in the μSCOPE 820 or in System Under Test memory.

The trace-memory circuits consist of a 256 × 32 RAM that records 256 machine cycles. The trace memory can record all cycles or only those cycles of a breakpoint match.

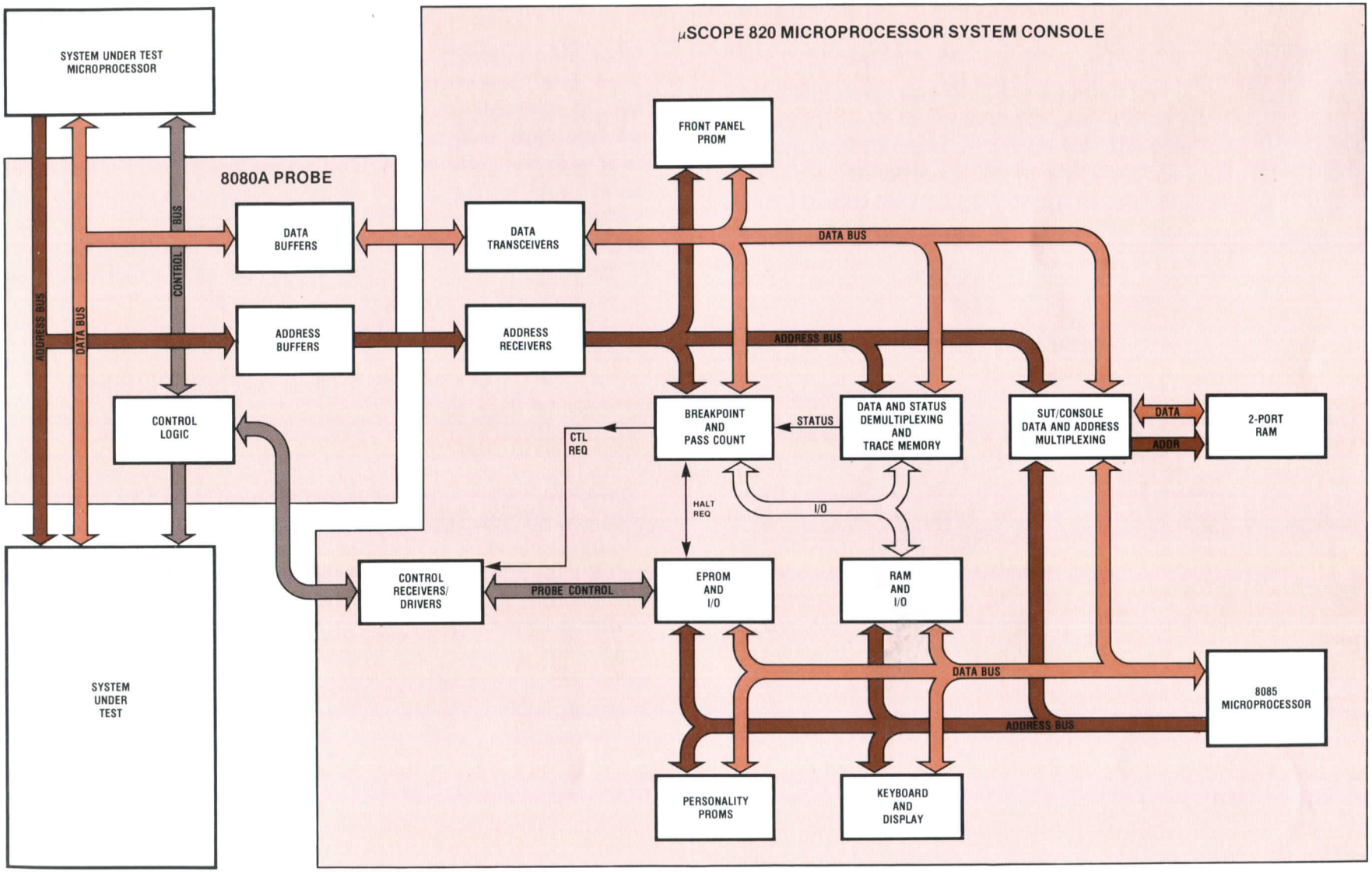



Fig. 1. μSCOPE 820 Functional Block Diagram

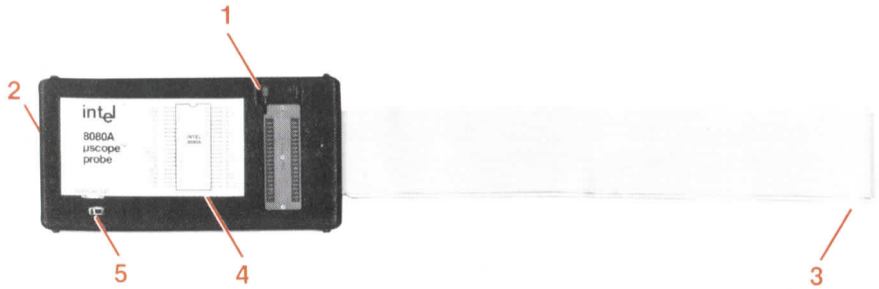
# Chapter 2
# Operating Information

This chapter contains installation procedures, description of controls and indicators, control panel familiarization, and basic operating instructions for the μSCOPE 820 Microprocessor System Console. Throughout this handbook, the references made and examples shown assume the μSCOPE 820 is operating with an Intel 8080A microprocessor-based system. Information for other microprocessors will be provided in handbook supplements. Before you connect the μSCOPE 820 to a system, it is important to become familiar with the parts of the μSCOPE 820.

## Getting Familiar With Your μSCOPE 820

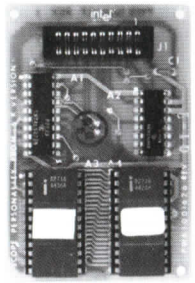
### The Probe

Each probe is labeled as to the applicable microprocessor, for example 8080A μSCOPE probe. Release the zero insertion force socket lever **1**, remove the cable connector, and lay the cable out as shown. On the left edge of the probe buffer box is a 50-pin connector **2** that connects with the 4-foot cable from the μSCOPE 820 console. The other end of the buffer box has a short captive cable with a 40-pin IC DIP connector **3**. The pin arrangement of this connector is the same as the pin arrangement shown on the label of the buffer box **4**. When viewed as shown, pin 1 is the upper left pin. A label on the connector shows this. The 2-position switch **5** selects 1K or 2K overlay memory size.



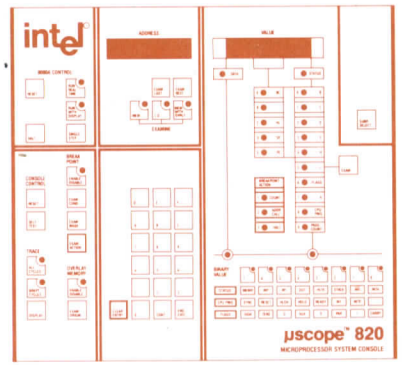

### The Personality Module

The module has two 16K (2K × 8) ROMs that contain firmware to configure the μSCOPE 820 for the microprocessor of the System Under Test. This module has a 20-pin edge connector (J1) that mates with a connector located in the storage compartment of the μSCOPE 820 console.



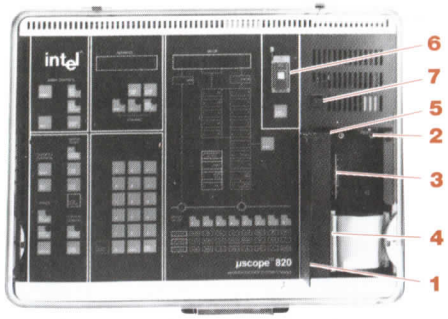
### The Panel Keyboard Overlay

This overlay is also unique to the microprocessor of the System Under Test. The panel overlay fits over the control panel of the μSCOPE 820 console and provides the necessary labels for switches, indicators, and displays of the μSCOPE 820. Two plastic captive pins align the overlay and hold it in position.




These three items (probe, personality module and panel overlay) are all identified for the appropriate microprocessor of the System Under Test. The remaining items are common to any microprocessor compatible with the μSCOPE 820. The 4-foot ribbon cable connects the μSCOPE 820 console to the personality probe. This cable is normally rolled and stowed within the μSCOPE console. The cable has two different 50-pin connectors. The large connector attaches to the console; the other connects to the probe.

## The Console




The μSCOPE 820 console is a stand-alone, self-contained unit with storage for probe, personality module and cables. Open the storage compartment door 1 and locate the power connector/fuseholder 2. This assembly also contains a printed-circuit card that conditions the machine for the facility line voltage. In the upper compartment is a 20-pin edge connector 3 for the personality module. The 50-pin edge connector 4 located in the lower left of the compartment connects the 4-foot ribbon cable to the probe buffer box. An access hole cut in the upper left corner of the storage compartment 5 exposes a pin on the printed-circuit board. This pin provides a sync signal to trigger an oscilloscope when a breakpoint match occurs.

On the upper right of the panel is a 24-pin, zero insertion force socket 6 for a pre-programmed overlay EPROM/ROM. Below the socket and to the right is the power on/off switch 7.

## How To Install The μSCOPE 820

Study the illustrations shown before you attempt to connect the μSCOPE 820 to a system. This might save you time and prevent a lot of frustration later.

Select the appropriate panel keyboard overlay for the microprocessor of the System Under Test. Place the overlay in position over the μSCOPE 820 console control panel. The overlay fits under the aluminum trim at the top and bottom edges of the console. Locate the holes for the overlay alignment pins and press the pins into position.

Have you checked to make sure the μSCOPE 820 console is compatible with facility power? If not, the line voltage selection guide inside the front cover describes how to check and change the power supply to operate with your facility power.



### CAUTION

The μSCOPE 820 will operate with one of several input voltages. Make sure the voltage selected is compatible with facility power. Serious damage to the μSCOPE 820 circuits may occur if the unit is connected to voltages that it is not conditioned for.

Select the appropriate personality module for the microprocessor in the System Under Test. Plug the module into the 20-pin edge connector of the μSCOPE 820 console. Orient the module so that the storage door can close when the module is in position.

Unroll the 4-foot ribbon cable and connect the large connector to the 50-pin board edge connector of the μSCOPE 820 console. Note that the top of the cable is labeled.

Select the proper μSCOPE 820 probe and connect it to the 4-foot cable. Note that the top of the cable connector is labeled.

Connect the power cord to the console power connector.

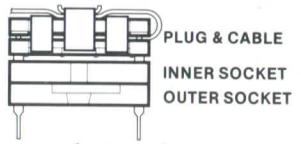
### NOTE

Before you power down the System Under Test to connect the μSCOPE 820 to the system, familiarize yourself with the operating controls and indicators of the μSCOPE 820 console by reading pages 8 to 13. These pages describe the controls and indicators and also provide you with an exercise to increase your confidence in the μSCOPE 820.

Power down the System Under Test and gain access to the microprocessor. Note the pin locations and carefully remove the microprocessor. Connect the IC DIP connector of the μSCOPE 820 probe into the System Under Test microprocessor socket. Insert the microprocessor into the zero insertion force socket of the μSCOPE 820 probe and tighten the pins by lowering the socket lever. Connect the μSCOPE 820 power cable to facility power, power up the μSCOPE 820, and power up the System Under Test. You are ready to control, monitor and interact with the System Under Test.

## CAUTION

Inserting the IC DIP connector backwards into the SUT CPU socket allows damaging current to flow into the probe buffer box. Be sure to insert pin 1 of the connector into pin 1 of the CPU socket.




If you break a pin on the IC DIP connector, carefully remove the outer socket and replace it. Never use the connector without the outer socket; if you damage the inner socket and attempt to remove it, you may cause intermittent failures in the cable. If the inner socket is damaged, the entire cable and connector assembly must be replaced.

**The panel overlay provides a graphic legend for the keyboard membrane switches and LED indicators. This figure shows the layout of the console controls with an Intel 8080A panel keyboard overlay. Other CPU panel keyboard overlays will be shown and described in handbook supplements.**

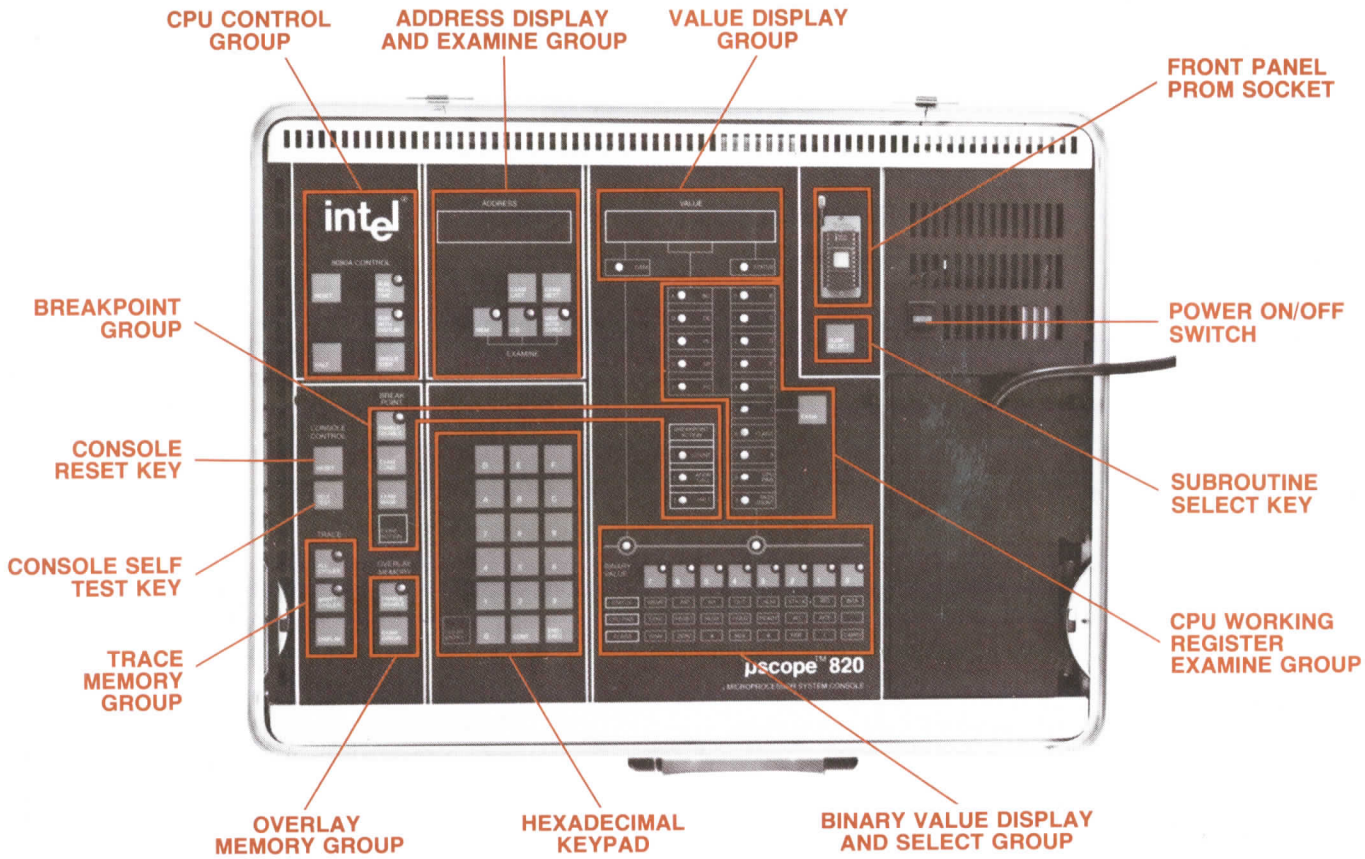
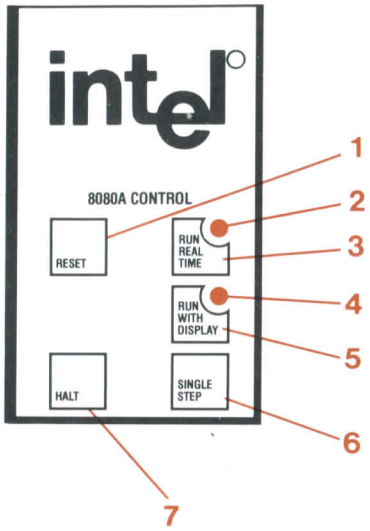



Fig. 2 μSCOPE 820 Console Controls and Indicators (With Intel 8080A Panel Keyboard Overlay)

# Console Controls and Indicators

## CPU Control Group




The CPU control group provides complete control over the operation of the CPU in the System Under Test. You can force the system CPU to halt, single step, reset, run real time or run with display.

Pressing the RESET key **1** resets the CPU only and does not reset the System Under Test.

The RUN REAL TIME key **3** configures the μSCOPE 820 console so panel commands do not alter program flow. The LED **2** illuminates to indicate that the run real time mode has been selected. The LED flashes to indicate lack of CPU activity.

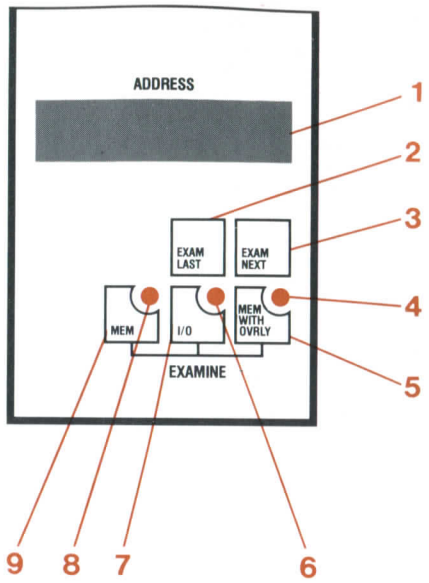
Pressing RUN WITH DISPLAY key **5** configures the μSCOPE 820 console so you can examine and alter program flow. In this mode, the μSCOPE 820 can alter System Under Test timing. The LED **4** illuminates to indicate that the run with display mode has been selected. The LED flashes to indicate lack of CPU activity.

The SINGLE STEP key **6** is used with the HALT key **7**. During system operation the halt mode places the CPU in suspension. Pressing the SINGLE STEP key steps the program one instruction cycle at a time.

## Address Display And Examine Group

The address display and examine group consists of a 9-segment, 4- digit LED readout **1**, five keys, and three LED indicators. The LED readout displays the 4-digit hexadecimal value of a 16-bit address or 2-digit hexadecimal value of an 8-bit I/O port number. The readout also prompts you in performing sequential procedures and informs you of procedural errors.

The EXAM LAST key **2** decrements the μSCOPE 820 to the I/O port number or memory address prior to the one selected and displayed. The EXAM NEXT key **3** increments the μSCOPE 820 to the next memory location or I/O port number.

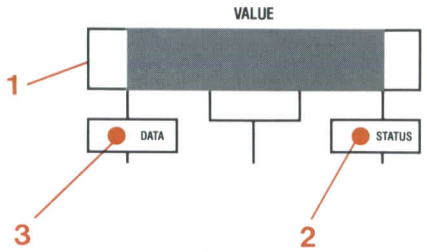



The MEM WITH OVRLY key **5** initiates a sequence allowing you to select, examine, and alter contents of memory including the overlay memory provided by the μSCOPE 820. LED **4** lights to indicate that you can select an address or change the address displayed.

The I/O key **7** initiates a sequence so you can select, examine, and alter I/O port values. The LED **6** flashes to indicate that you can enter an I/O port number during a breakpoint-condition or mask-examine sequence.

The MEM key **9** initiates a sequence that permits you to select, examine, and alter the contents of System Under Test memory locations only. The LED **8** flashes to indicate that you can enter an address during a breakpoint-condition or mask-examine sequence.

## Value Display Group




The value display group consists of a 9-segment, 4-digit LED readout **1**, a STATUS LED indicator **2**, and a DATA LED indicator **3**.

The value readout displays the 8-bit hexadecimal value of I/O or memory data as well as the 8-bit hexadecimal value of status. The readout will also display the 8- or 16-bit value of register contents. The graphic arrangement of the readout and related LED indicators help you evaluate the information displayed.

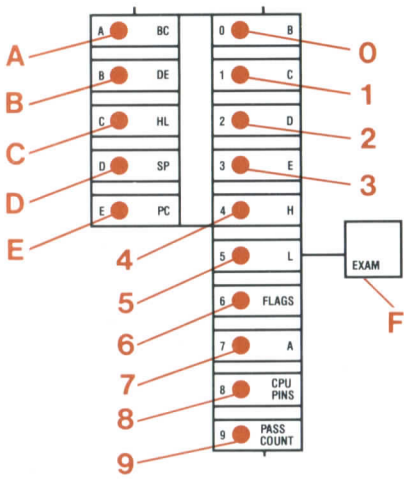
The STATUS LED **2** illuminates to show that you are looking at the value of an 8-bit status byte. The LED flashes to indicate that the status byte is alterable during breakpoint sequences.

The DATA LED **3** illuminates to indicate that you are examining the value of an 8-bit data byte. The LED flashes to show that you are able to modify or alter the data byte.

### Front Panel PROM Socket/Subroutine Select Key

The 24-pin, zero insertion force socket allows you to use preprogrammed (canned) routines through the μSCOPE 820. The socket is oriented with pin 1 at the upper left. The key lets you execute a particular routine from the preprogrammed PROM/ROM or a routine resident in the System Under Test.

### CPU Working-Register Examine Group




The CPU working-register examine group consists of an EXAM key **F** and two groups of LED indicators. The first group of LED indicators **0** through **8** represent the single-byte registers, the register for the flags, accumulator, and CPU pins. The remaining indicator of this first group **9** represents the contents of the μSCOPE 820 pass-count register. The second group of LED indicators **A** through **E** represent the double-byte registers, stack pointer, and program counter of the CPU.

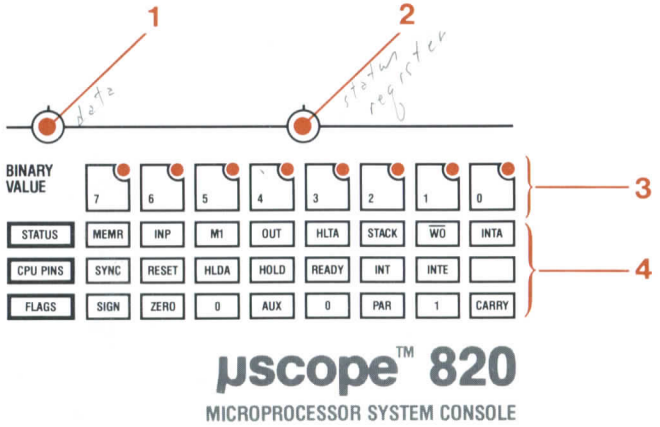
Pressing the EXAM key **F** starts a sequence so you can select any one of the 15 registers. (The term register is used loosely because the CPU pins grouping is not a register.) The letter or number to the left of the LED corresponds to a key of the hexadecimal key pad used to display the register contents. The nomenclature to the right of the LED corresponds to the single- or double-byte registers, CPU pins, accumulator, flags, or μSCOPE 820 pass counter.

The LED of the selected register illuminates to indicate that you are examining the contents of that register. When the LED flashes, you are able to alter the contents of that register.

### Binary Value Display And Select Group

The binary value display and select group consists of a left BINARY VALUE LED **1**, a right BINARY VALUE LED **2**, eight binary keys and LED indicators **3**, and a bit-assignment reference table **4**. The left BINARY VALUE LED **1** illuminates to show that the eight LED indicators represent an 8-bit data byte. The eight keys **3** allow you to change each bit of the data byte individually.

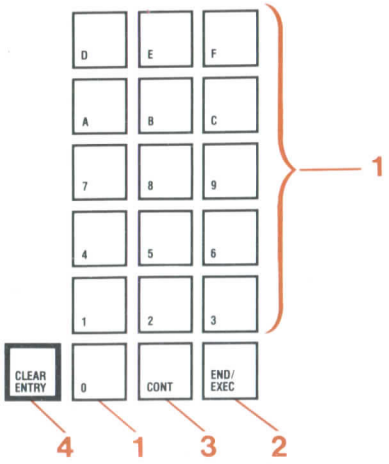
The right BINARY VALUE LED **2** illuminates to show that the eight LED indicators represent an 8-bit status byte or register contents. The eight binary keys **3** allow you to change each bit of the status byte. You can also alter the contents of any single-byte CPU register, the flags, or the CPU accumulator.




### Hexadecimal Keypad

The hexadecimal keypad consists of a 16 hexadecimal value keys **1**, an END/EXEC key **2**, a CONT key **3**, and a CLEAR ENTRY key **4**.

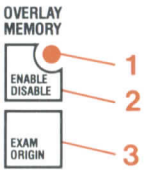
The hexadecimal keys are used to select or alter memory, I/O port numbers, data bytes, status bytes, or register contents. The alphanumerics on the keys also correspond to the μSCOPE 820 value




examine and breakpoint action functions.

The END/EXEC key **2** terminates a sequence or executes an action. The CONT key **3** ends one portion of a sequence and begins another or continues a series of sequences. The CLEAR ENTRY key **4** cancels any numeric selections made during a sequence step and reverts the console to the original value.

## Overlay Memory Group

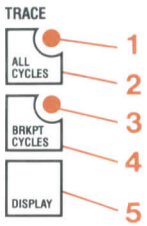



The overlay memory group consists of the ENABLE/DISABLE LED **1**, the ENABLE/DISABLE key **2**, and the EXAM ORIGIN key **3**.

The ENABLE/DISABLE LED **1** illuminates to indicate that μSCOPE 820 overlay memory is inserted over the System Under Test memory space. The ENABLE/DISABLE key **2** enables or disables the insertion of the μSCOPE 820 overlay memory into the System Under Test memory space.

The EXAM ORIGIN key **3** initiates a sequence so you can assign the address of the first byte of a 1K or 2K overlay memory block provided by the μSCOPE 820. The size of the block is assigned by a switch on the probe.

## Trace-Memory Group




The trace-memory group controls the passive trace memory modes of the μSCOPE 820. LED indicators illuminate to show selected modes.

The ALL CYCLES key **2** configures the trace memory circuits to record all machine cycles of CPU operation. The trace recording is constantly refreshed with the most recent 256 cycles of CPU operation.

The ALL CYCLES LED **1** illuminates to indicate that the μSCOPE 820 is recording all CPU machine cycles.

The BRKPT CYCLES key **4** selects a mode of trace recording where a record is kept of only those cycles satisfying a breakpoint match. A lighted BRKPT CYCLES LED **3** indicates the μSCOPE 820 is recording only the cycles that match the breakpoint value.

The DISPLAY key **5** initiates a trace-memory display sequence and disables the mode of tracing that was previously selected. The CPU machine cycles recorded by the trace memory are then sequentially shown in the ADDRESS and VALUE readouts. The values displayed in the ADDRESS and VALUE readouts are not alterable.

## Console Self Test Key

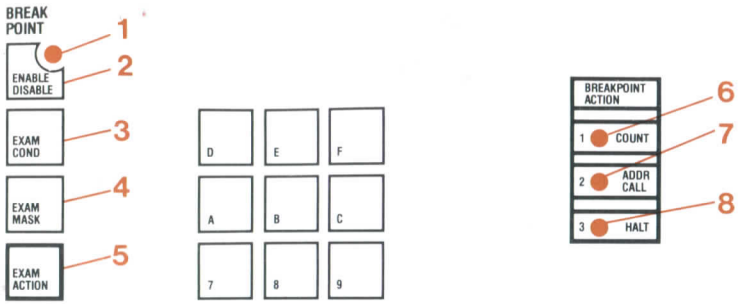

The SELF TEST key starts a series of firmware routines to verify the operation of the μSCOPE 820 console circuits. A total of 13 tests are performed sequentially. Should the console fail a particular test, the ADDRESS readout will display FAIL and the VALUE readout will display a hexadecimal value. This value represents the number of test failed. A list of these tests and their hexadecimal numbers are shown in Appendix A of this handbook. Self test must be performed with the System Under Test powered down and the console reset.

## Console Reset Key


The RESET key controls a hardware reset of the μSCOPE 820 console circuits. When you press the reset key, the μSCOPE 820 console resets all internal circuits and is configured in an initialized mode. The System Under Test CPU is also reset.

## Breakpoint Group

The breakpoint group consists of four keys, an ENABLE/DISABLE LED **1** and three BREAKPOINT ACTION LED indicators. The following paragraphs describes the controls and indicators in the order that you will use them.




The EXAM COND key **3** starts a sequence that permits you to examine and select the address, data, and status bytes at which a breakpoint is to occur.

The EXAM MASK key **4** starts a sequence that allows you to examine and establish a mask over the address, data, and/or status bytes.

The EXAM ACTION key **5** begins a sequence so you can select the action the μSCOPE 820 is to take when a breakpoint match occurs.

The BREAKPOINT ACTION COUNT LED **6** illuminates to show you have selected the pass-count mode. You can enter a value into the μSCOPE 820 pass-count register and, after a breakpoint match has occurred the specified number of times, the μSCOPE 820 halts (suspends) the CPU. The numeral 1 to the left of the LED corresponds to the numeral 1 that is found on the hexadecimal keypad.

The BREAKPOINT ACTION ADDR CALL LED **7** illuminates when you select a subroutine mode. You enter the address of a subroutine and, when a breakpoint match occurs, the system program is diverted to the selected routine. The numeral 2 corresponds to numeral 2 of the hex keypad.

The BREAKPOINT ACTION HALT LED **8** illuminates when you select the halt mode. With this action selected, the μSCOPE 820 halts (suspends) the CPU whenever a breakpoint match occurs.

The BREAKPOINT ENABLE/DISABLE LED **1** illuminates to indicate the breakpoint function is enabled. The ENABLE/DISABLE key **2** activates or deactivates the breakpoint circuits. When using trace at breakpoint, a breakpoint can be established and tracing of breakpoint cycles occurs regardless of the condition of the breakpoint function (enabled or disabled).

## Control Panel Familiarization

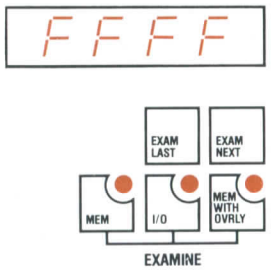
Now that you are familiar with the controls and indicators of the μSCOPE 820, let's get familiar with the operation of the console. It's a good idea to learn a few sequences and gain confidence in the μSCOPE 820 before you connect it to a system. But first, have you checked to make sure that the unit is compatible with facility power? Have you installed a personality module that is appropriate for the System Under Test?


Connect the μSCOPE 820 console to facility power and power up the unit by pressing the power switch. The power switch is not labeled; however, you will know when power is on since the ADDRESS and VALUE readouts display "Init 8080." Note that the VALUE readout displays the microprocessor that the personality module is configured for.

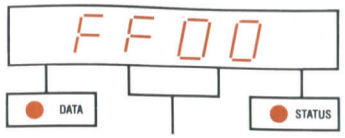
Self test the μSCOPE 820 by pressing the CONSOLE CONTROL SELF TEST [SELF TEST] key. All LED indicators flicker for an instant and then all segments of the ADDRESS and VALUE displays illuminate. Press the CONT [CONT] key of the hexadecimal keypad. The readouts go out and all LED indicators illuminate. Pressing the CONT [CONT] key at this point toggles the lamp-test mode between the readouts and the LED indicators. Now press the END/EXEC [END/EXEC] key. You have ended the self-test mode and the console is back to the initialized state.

So that you will recognize an error, press any key of the hexadecimal keypad. Notice that the ADDRESS readout displays "Err". You have made an operating error and the console will not accept the command given. No harm has been done. If you get an error indication during an operating sequence, you will have to restart that particular sequence. Let's try again. Press the RUN WITH DISPLAY [RUN WITH DISPLAY] key. The error indication is gone and the RUN WITH DISPLAY LED illuminates. Because the console is not connected to a system, the RUN WITH DISPLAY indicator blinks. This has no effect on what you are doing at this point, so pay no attention to the RUN WITH DISPLAY indicator.

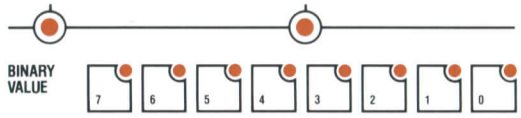
As an exercise, let's set up a breakpoint condition, establish a breakpoint mask, and select a breakpoint action when a breakpoint match occurs. Start by pressing the BREAKPOINT EXAM COND [EXAM COND] key. The ADDRESS and VALUE readouts display a hexadecimal value (0000-0000) and the MEM [MEM] indicator flashes. The flashing LED indicates that you are looking at a memory address and that this address can be changed. Using the hexadecimal keypad, select a 4-digit address, for example FFFF. The ADDRESS readout now displays the selected address. Press the CLEAR ENTRY [CLEAR ENTRY] key on the keypad. Note that the readout reverts to the original address (0000). Select a new address (try ABCD), and then press the CONT [CONT] key of the keypad. The indicator stops flashing and is fully illuminated. You have now established the address at which breakpoint is to occur (ABCD).




Notice that the DATA [DATA] LED is now flashing. This informs you that the breakpoint examine sequence is still in effect and that you can alter or establish a data value. Also notice that the left BINARY VALUE LED is illuminated and the eight binary bits coincide with the first two digits shown on the VALUE readout. Use the hexadecimal keypad to enter a data byte, for example AA. If you would rather use the binary switches to establish a data byte, press the switches to light all eight BINARY VALUE LEDs resulting a data byte of FF.




Press the CONT [CONT] key. The DATA [DATA] indicator stops flashing and is fully illuminated. Notice that the STATUS [STATUS] indicator is now flashing. This lets you know that the sequence is still in effect and that you can alter or establish the status byte of the breakpoint condition. Also notice that the right BINARY VALUE LED is illuminated. The eight binary keys and LEDs coincide with the last two digits of the VALUE readout and represent the status byte. Enter a status byte by using the keypad or the binary keys, FF for example.




After you have entered the status byte, examine the breakpoint mask by pressing the EXAM MASK [EXAM MASK] key. What happened? The ADDRESS readout displays 'Err'! You have made an operating error. You attempted to start a new sequence without ending the old sequence. So, start all over again by pressing the EXAM COND [EXAM COND] key and establish an address, data byte and status byte. Next time end the sequence with the END/EXEC [END/EXEC] key.

**NOTE**

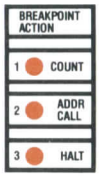
The most common error you will make will be to try to enter a new sequence before completing the old sequence or not paying attention to μSCOPE 820 prompting for proper sequences. When you get an error message, you must restart the sequence from step 1.

At this point you have established a breakpoint condition. The next step is to set up the breakpoint mask. You examine and establish the breakpoint mask using the same procedure used in establishing the

breakpoint condition. The only difference is that you press the EXAM MASK [EXAM MASK] key to start the sequence.

Ready? Press the EXAM MASK [EXAM MASK] key. Enter a mask value for the address you selected for the breakpoint condition (FFFF). Press the CONT [CONT] key. Now, enter a mask value for the data byte you selected for the breakpoint condition (FF). Press CONT [CONT] and, if you like, enter a mask value for the status byte of the breakpoint condition (FF). End the sequence by pressing the END/EXEC [END/EXEC] key.

You have now set the 32-bit breakpoint register of the μSCOPE 820 to a particular value and you have also set the 32-bit breakpoint mask register to all '1s'. This means that the only way a breakpoint match can occur is when the System Under Test program performs an access involving the specific 32-bit address, data and status word determined by the condition registers. In actual practice you will probably mask off the status bits, the data bits, or a portion thereof.




The next step is to program the μSCOPE 820 to take an action when a breakpoint occurs. You have three choices: halt the CPU after a specified number of breakpoint matches, call a subroutine at the first breakpoint match, or halt the CPU at the first breakpoint match. Press the EXAM ACTION [EXAM ACTION] key. The ADDRESS readout displays SEL, the VALUE readout displays HALT and the BREAKPOINT ACTION HALT LED illuminates.

Press the 1 [1] key of the keypad. The COUNT [1 COUNT] LED illuminates and the last two digits of the VALUE display will read 00. Press the CONT [CONT] key. Note that the COUNT [1 COUNT] LED flashes. You have now selected the breakpoint action to be pass count and the flashing LED indicates that you can enter a value into the μSCOPE 820 pass-count register. Using the hexadecimal keypad, enter FF. This is the number of breakpoint passes the μSCOPE 820 counts before the halt action occurs. End the sequence by pressing the END/EXEC [END/EXEC] key.

Change the breakpoint action again. Press the EXAM ACTION key and select ADDR CALL [2 ADDR CALL] by pressing 2 [2] on the keypad. Press the CONT [CONT] key and enter the address of a subroutine contained in the front panel PROM or System Under Test. (For the exercise pick an arbitrary address.) Press the END/EXEC [END/EXEC] key. The μSCOPE 820 is now programmed to have the CPU call this subroutine when a breakpoint match occurs.

Doing the exercise in these last paragraphs should help you gain confidence in the μSCOPE 820. If you feel it is necessary, repeat this exercise as much as you like.

## Operating Instructions

### NOTE

Always press the CONSOLE RESET key to initialize the console after power is first applied to the μSCOPE 820. When power is applied to the console, the 8085 CPU may miss the initial power-on-reset pulse. When this occurs, the 8085 CPU may be performing random routines. The fact that the Address and Value readouts display "Init 8080" does not mean that the console is truly initialized. Therefore, to avoid uncontrolled routines, apply power to the console and immediately press CONSOLE RESET.

This portion of the handbook provides you with the do's and don'ts of the μSCOPE 820. It also describes the different operating modes. The instructions provided in this handbook are not intended to lead you by the hand in troubleshooting or debugging a System Under Test. Your knowledge of the System Under Test and its operating program will dictate how you will use the μSCOPE 820.

Operating the μSCOPE 820 is described in terms of the System Under Test operating condition. You may want to preset or preprogram the console prior to powering up the System Under Test. You may want

to halt the System Under Test CPU and perform certain routines or you may want to perform routines with the system running in real time or running with display. A system error may occur as a result of the System Under Test CPU failing to respond to μSCOPE 820 control.

The following paragraphs describe the functions and sequences allowed when the System Under Test is operating in these four conditions: powered down, CPU halted, running with display, and running real time. Included in this Handbook are flow charts of operating procedures and functions. These flow charts are introduced as you read the paragraphs describing the operating conditions. The flow charts are also useful as quick reference material.

## System Under Test Powered Down

With power applied to the μSCOPE 820 and before power is applied to the System Under Test, all sequences, functions, or modes that do not require System Under Test timing or CPU control are allowed. Figure 3 is an operational flow chart showing allowable μSCOPE 820 functions and sequences. When the μSCOPE 820 is powered up, the instrument is in an initialized state informing you of the panel configuration.

**Self Test.** The self test mode must be performed before power is ever applied to the System Under Test. Pressing the SELF TEST key starts a series of 13 preprogrammed test routines that end with a lamp test. As each test routine is completed and verified, the program steps to the next test until all 13 tests are verified. The lamp test starts by illuminating all segments of the hexadecimal ADDRESS and VALUE readouts. Pressing the CONT key of the keypad toggles the lamp test between the hexadecimal readouts and the discrete LEDs of the μSCOPE 820 panel.

If the instrument fails a test (except lamp test), the ADDRESS readout displays FAIL and the VALUE readout displays a hexadecimal failure code. This code refers to the test that has failed. The failure codes and related tests are described in Appendix A of this handbook. Pressing the END/EXEC key of the key pad ends the self test sequence and reverts the μSCOPE 820 to the initialized state. Figure 7 is an operational flow chart of the self test sequence.

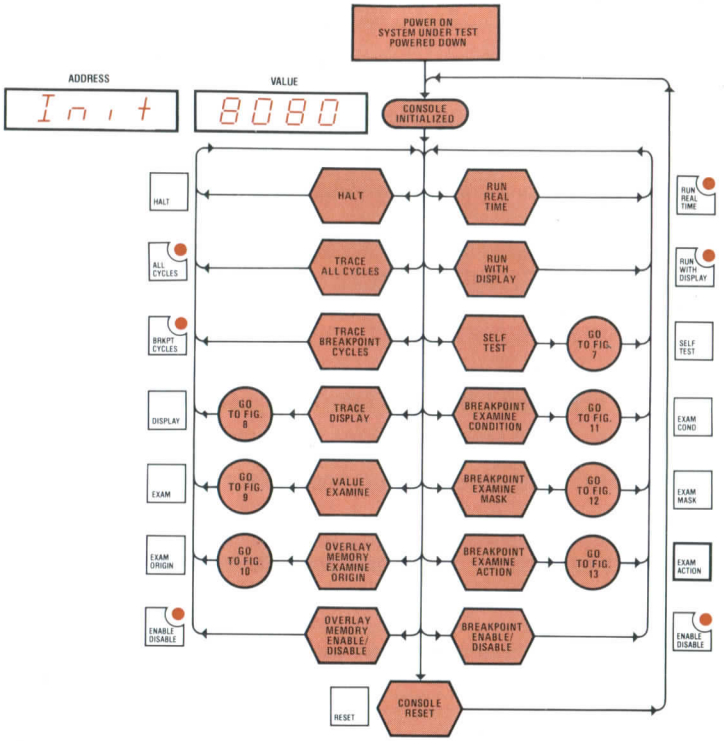



**Fig. 3 Allowable Functions and Sequences With System Under Test Powered Down**

**Halt.** Initiating halt with the System Under Test powered down is a single-key function. You may want to gain control of the System Under Test CPU as soon as possible after the system is powered up. Selecting halt in this condition results in the CPU executing only one instruction cycle when the System Under Test is powered up. The μSCOPE 820 holds the CPU in suspension until you release it by selecting run real time, run with display, single-stepping the System Under Test program, or selecting a subroutine.

**Run Real Time.** Selecting the run real time mode with the System Under Test powered down is a single-key instruction. In this mode, the μSCOPE 820 is configured so that when the system is powered up, μSCOPE 820 commands do not alter system operation. The μSCOPE 820 has no control over the CPU in this mode other than breakpoint.

**Run With Display.** Selecting the run with display mode with the System Under Test powered down is also a single-key instruction. In this mode the μSCOPE 820 can steal cycles from the CPU to accomplish display update or to enable the setup of various μSCOPE 820 functions.

**Breakpoint Examine Condition.** The breakpoint examine condition is the first of three sequences required to establish a breakpoint. The breakpoint examine condition lets you examine and assign a 32-bit value to the μSCOPE 820 breakpoint register. This 32-bit value (16-bit address, 8-bit data byte, and 8-bit status byte) is the System Under Test machine cycle at which the breakpoint is to occur.

Depending upon your requirements and how you want to use breakpoint, you may use any or all portions of the breakpoint condition. You may use the address, the data byte, the status byte, or any combination of the three. Figure 11 is an operational flow chart that shows how to establish a breakpoint condition. You must follow the sequences shown. Should the μSCOPE 820 respond with an error indication at any point during the sequence, you must restart the sequence.

**Breakpoint Examine Mask.** The examine mask is the second sequence required for setting a breakpoint. This sequence allows you to examine and assign a value to the μSCOPE 820 32-bit mask register. For a breakpoint to occur, the machine cycle must satisfy the mask register and the breakpoint condition register. When establishing a breakpoint mask, keep in mind that the mask performs the AND function bit by bit with the breakpoint condition. A 1-bit enables the corresponding bit of the condition and a 0-bit makes the corresponding bit of the condition a "don't care." You may find it necessary to convert hexadecimal values to binary values to use the mask at its maximum potential. Figure 12 is an operational flow chart that shows how to establish a breakpoint mask. Notice that the sequence is identical to the examine condition sequence.

**Breakpoint Examine Action.** The breakpoint examine action is the third required sequence of breakpoint setup. When a breakpoint match occurs, the μSCOPE 820 executes one of three actions. The examine action sequence allows you to examine and select one of the three actions. The μSCOPE 820 initially selects the halt action. This means that the μSCOPE 820 halts (suspends) the System Under Test CPU at the first breakpoint match. Figure 13 is an operational flow chart that shows how to select a breakpoint action. The flow chart also shows how to enter a value into the μSCOPE 820 pass-count register or select an address of a subroutine.

**NOTE**

Execution of the instructions MOV C,M and MOV M,C are decoded by the μSCOPE 820 so access is made to the System Under Test memory regardless of the overlay enable/disable. μSCOPE 820 uses special instruction MOV M,C during execution of subroutine call on a breakpoint. Therefore, if a breakpoint action is call subroutine, and the overlay memory is enabled, the stack pointer should not be assigned a value within the overlay RAM. This will cause the subroutine to return to an unexpected location because the return address will have been written in the System Under Test memory rather than the overlay memory of the μSCOPE 820.

**Breakpoint Enable/Disable.** Breakpoint enable/disable is a single-key function that controls the breakpoint hardware of the μSCOPE 820. By using breakpoint and trace memory together, you can set up the μSCOPE 820 to run a System Under Test operating sequence repetitively until a fault is detected. The sequence can then be narrowed until the problem is isolated.

**Trace All Cycles.** Activating the trace all cycles mode prior to powering up the System Under Test allows recording the first 256 machine cycles. However, you must halt (suspend) the CPU by using breakpoint to maintain a record of the first 256 cycles. The trace memory of μSCOPE 820 is constantly rewritten and only maintains the last 256 cycles recorded.

**Trace Breakpoint Cycles.** The trace breakpoint cycles mode operates in conjunction with the μSCOPE 820 breakpoint mode. Presetting the μSCOPE 820 breakpoint hardware and selecting the trace breakpoint cycles mode results in recording only those cycles that satisfy the μSCOPE 820 breakpoint hardware. Breakpoint can be established and tracing of breakpoint occurs regardless of breakpoint status (enabled or disabled).

**Trace Display.** Trace display is a multiple-key sequence and is allowable with the System Under Test powered down. Selecting trace display prior to powering the system up serves no purpose. However, you can trace all cycles or only breakpoint cycles while the system is running a program, power the system down, and then display trace memory. Figure 8 is an operational flow chart of the trace display sequence. Keep in mind that the ADDRESS and VALUE readouts display the last machine cycle recorded when you select trace display. You can then decrement one machine cycle at a time by pressing the EXAM LAST key. Pressing the EXAM NEXT key increments one machine cycle at a time. Pressing the TRACE DISPLAY key also ends the selected trace mode.

**Overlay Memory Examine Origin.** The overlay memory examine origin sequence allows you to examine and assign the starting address of the 1K or 2K μSCOPE 820 overlay memory. A switch on the personality probe selects the size of the overlay memory block (1K or 2K). When you assign the starting address, be sure you select the first byte on a 1K or 2K boundary. If you assign an address that is not on the proper boundary, the μSCOPE 820 automatically goes to the next lower 1K or 2K boundary, depending upon the selected size of memory overlay block (1K or 2K). Figure 10 is an operational flow chart of the overlay memory examine origin sequence.

**Overlay Memory Enable/Disable.** The overlay memory enable/disable is a single-key function. After you have selected the size of the memory block and assigned a starting address, you enable or disable the μSCOPE 820 overlay memory with the ENABLE/DISABLE key.

**Value Examine.** The value examine mode is also a multiple-key sequence that has limited use when the System Under Test is powered down. Only the CPU pins and the contents of the μSCOPE 820 pass-count register can be examined. The values displayed cannot be altered. Figure 9 is an operational flow chart of the value examine mode sequence with the System Under Test powered down.

**Console Reset.** If you make an error in presetting the μSCOPE 820 hardware or decide to clear all previous programming of μSCOPE 820 hardware, pressing the CONSOLE RESET key will clear all previous entries and establish the initialized state.

## System Under Test CPU Halted (Suspended)

With the System Under Test CPU held in suspension by the μSCOPE 820, all functions and sequences with the exception of self-test are allowed. The halted condition may be the result of pressing the HALT key or the result of a breakpoint match. Figure 4 is an operational flow chart showing the functions and sequences allowed with the System Under Test CPU halted.

A very important feature of the μSCOPE 820 is the ability to halt the CPU, examine a value, single-step through the program and continue to examine the selected value. As an example: When you are examining the program counter, the VALUE readout will display the contents of the program counter. By pressing the END/EXEC key and then single-stepping through the program, the contents of the program counter will remain displayed and be updated as you step through the program.

### NOTE

If a breakpoint action is address call and a halt has been implemented prior to the breakpoint, you cannot single-step through the address call. As you single-step, the μSCOPE 820 will not recognize the breakpoint and you will continue to single-step the program past the breakpoint.

**Run Real Time.** Selecting run real time is a single-key instruction that releases the System Under Test CPU; μSCOPE 820 commands do not alter system operation. The μSCOPE 820 has no control over the CPU in this mode other than breakpoint.

**Run With Display.** Selecting run with display is also a single-key instruction that releases the System Under Test CPU. In this mode the μSCOPE 820 can steal cycles from the CPU to accomplish display update or to enable the setup of various μSCOPE 820 functions.

**Single-Step.** Pressing the SINGLE STEP key releases the CPU for one instruction cycle and the μSCOPE 820 again holds the CPU in suspension. By using the examine functions you can single-step the program and examine the results of each instruction cycle.

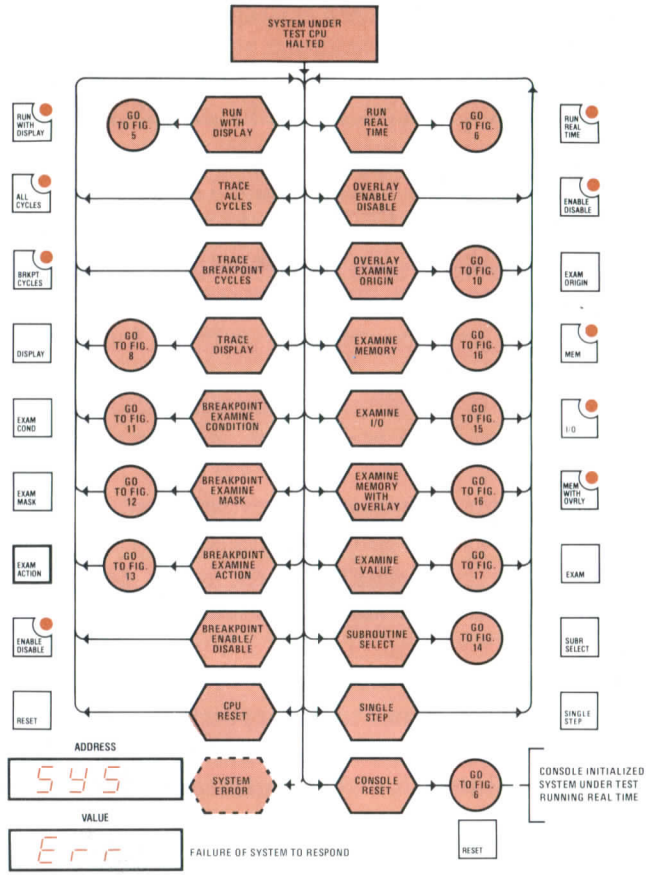



Fig. 4 Allowable Functions and Sequences With System Under Test CPU Halted

**Subroutine Select.** With the System Under Test CPU halted, you can select one of 16 routines defined by the μSCOPE 820 front panel PROM/ROM. Figure 14 is an operational flow chart of the sub-

**Breakpoint Examine Condition.** The breakpoint examine condition is the first of three sequences required to establish a breakpoint. The examine condition sequence allows you to examine and assign a

32- bit value to the μSCOPE 820 breakpoint register. This 32-bit value (16- bit address, 8-bit data byte, and 8-bit status byte) is the System Under Test machine cycle at which breakpoint is to occur.

Depending upon your requirements and how you want to use breakpoint, you may use any or all portions of the breakpoint condition. For example, you may select the address, the data byte, the status byte, or any portion of the three. Figure 11 is an operational flow chart that shows how to establish a breakpoint condition. You must follow the sequences shown. Should the μSCOPE 820 respond with an error indication at any point during the sequence, you must restart the sequence.

**Breakpoint Examine Mask.** The breakpoint examine mask is the second sequence required for a breakpoint. This sequence lets you examine and assign a value to the μSCOPE 820 32-bit mask register. For a breakpoint to occur, the machine cycle must satisfy the mask register and the breakpoint condition register. When establishing a breakpoint mask, keep in mind that the mask performs the AND function bit by bit with the breakpoint condition. A 1-bit enables the corresponding bit of the condition and a 0-bit makes the corresponding bit of the condition a "don't care." You may find it necessary to convert hexadecimal values to binary values in order to use the mask at its maximum potential. Figure 12 shows how to establish a breakpoint mask. Notice that the sequence is identical to the examine condition sequence.

**Breakpoint Examine Action.** The breakpoint examine action is the third required sequence of breakpoint. When a breakpoint match occurs, the μSCOPE 820 executes one of three actions. The examine action sequence is used to examine and select one of the three actions. The μSCOPE 820 initially selects the halt action. This means that the μSCOPE 820 halts (suspends) the System Under Test CPU at the first breakpoint match. Figure 13 is an operational flow chart that shows how to select a breakpoint action. The flow chart also shows how to enter a value into the μSCOPE 820 pass-count register or select an address of a subroutine.

**NOTE**

Execution of the instructions MOV C,M and MOV M,C are decoded by the μSCOPE 820 so that access is made to the System Under Test memory regardless of overlay enable/disable. The μSCOPE 820 uses special instruction MOV M,C during execution of subroutine call on a breakpoint. Therefore, if a breakpoint action is a call subroutine and the overlay memory is enabled, the stack pointer should not be assigned a value that is within the overlay RAM. This will cause the subroutine to return to an unexpected location because the return address will have been written in the System Under Test memory rather than the μSCOPE 820 overlay memory.

**Breakpoint Enable/Disable.** Breakpoint enable/disable is a single-key function that controls the breakpoint hardware of the μSCOPE 820. By using breakpoint and trace memory together, you can set up the μSCOPE 820 to run a System Under Test operating sequence repetitively until a fault is detected. The sequence can then be narrowed until the problem is isolated.

**Trace All Cycles.** You can activate the trace all cycles mode with the System Under Test halted. When the CPU is released by selecting run real time, run with display, running a subroutine, or single-stepped, trace memory will record CPU machine cycles. Keep in mind that the trace memory is constantly refreshed and only maintains the last 256 cycles recorded.

**Trace Breakpoint Cycles.** You can preset the μSCOPE 820 breakpoint hardware and select the trace breakpoint cycles mode with the System Under Test CPU halted. When the CPU is released cycles that satisfy the breakpoint hardware are recorded. Tracing of breakpoint occurs regardless of breakpoint status (enabled or disabled).

**Trace Display.** Trace display is a multiple-key sequence and is allowable with the System Under Test CPU halted. Figure 8 is an operational flow chart of the trace display sequence. Keep in mind that the ADDRESS and VALUE readouts display the last machine cycle recorded when you selected trace display. You can then decrement one machine cycle at a time by pressing the EXAM LAST key. Pressing the EXAM NEXT key increments one machine cycle at a time. Pressing the TRACE DISPLAY key also ends the selected trace mode.

**Overlay Memory Examine Origin.** The overlay memory examine origin sequence allows you to examine and assign the starting address of the 1K or 2K μSCOPE 820 overlay memory. A switch on the

personality probe selects the size of the overlay memory block (1K or 2K). When you assign the starting address, be sure you select the first byte on a 1K or 2K boundary. If you assign an address that is not on the proper boundary, the μSCOPE 820 automatically goes to the next lower 1K or 2K boundary, depending upon the selected size of memory overlay block (1K or 2K). Figure 10 is an operational flow chart of the overlay memory examine origin sequence.

**Overlay Memory Enable/Disable.** The overlay memory enable/disable is a single-key function. After you have selected the size of the memory block and assigned a starting address, you enable or disable the μSCOPE 820 overlay memory with the ENABLE/DISABLE key.

**Examine Memory.** The examine memory sequence allows you to examine System Under Test memory and, if necessary, alter data in the memory locations. Figure 16 is an operational flow chart of the examine memory sequence. Under normal operation you should halt the CPU, initiate the examine memory sequence and enter the address of the memory location to be examined. You can then examine the data and by using the EXAM NEXT or EXAM LAST keys to step through the stored data.

**Examine I/O.** The examine I/O sequence is the same as the examine memory sequence except that you are working with an 8-bit I/O number instead of a 16-bit address. Figure 15 is an operational flow chart of the examine I/O sequences.

**Examine Memory with Overlay.** The examine memory with overlay sequence is used when you assign μSCOPE 820 overlay memory over the System Under Test memory. The examine memory with overlay sequence is the same as the examine memory sequence except that you are examining and altering memory located in the μSCOPE 820 as well as System Under Test memory. An operational flow chart of the examine memory with overlay sequence is shown in Figure 16.

**Value Examine.** The value examine sequence is one of the more complex functions of the μSCOPE 820, since it can be used to examine and—if necessary—alter the working single-byte registers, double-byte registers, stack pointer, program counter, flag register, and accumulator of the CPU. Pressing the EXAM key initiates the sequence. The ADDRESS readout prompts you to select the value to be examined. Figure 17 is an operational flow chart of the examine sequence.

**CPU Reset.** The CPU reset function is allowable with the System Under Test CPU halted. Pressing the CPU RESET key will reset the System Under Test CPU.

**Console Reset.** If you make an error in presetting the μSCOPE 820 or decide to clear all previous setting of μSCOPE 820 hardware, pressing the CONSOLE RESET key clears all previous entries and establishes the initialized state. The System Under Test CPU is released and runs in real time.

## System Under Test Running With Display

The run with display operating condition is the most complex operating condition of the μSCOPE 820. You will find yourself using this condition a majority of the time. Figure 5 is an operational flow chart showing the actions and sequences allowed in the run with display condition. The following paragraphs describe each of these functions and sequences.

### NOTE

When displaying register contents, memory locations, I/O port values, or modifying breakpoint or overlay memory origin, the possibility of missing a breakpoint match exists. Therefore, if a breakpoint action is critical, you should not use the examine modes or modify breakpoint or overlay memory origin.

**Halt.** Halt is a single key instruction that transfers the run with display condition to the halt condition.

**Run Real Time.** Run real time is also a single key instruction that transfers the run with display condition to run real time. μSCOPE 820 commands do not alter system operation in this condition and the μSCOPE 820 has no control over the CPU with the exception of breakpoint.

**Breakpoint Examine Condition.** The breakpoint examine condition is the first of three sequences required to establish a breakpoint. The examine condition sequence allows you to examine and assign a 32-bit value to the μSCOPE 820 breakpoint register. This 32-bit value (16-bit address, 8-bit data byte, and 8-bit status byte) is the System Under Test machine cycle at which breakpoint is to occur.

Depending upon your requirements and how you want to use breakpoint, you may use any or all portions of the breakpoint condition. You may use the address, the data byte, the status byte, or any combination of the three. Figure 11 is an operational flow chart that shows how to establish a breakpoint condition. You must follow the sequences shown. Should the μSCOPE 820 respond with an error indication at any point during the sequence, you must restart the sequence.

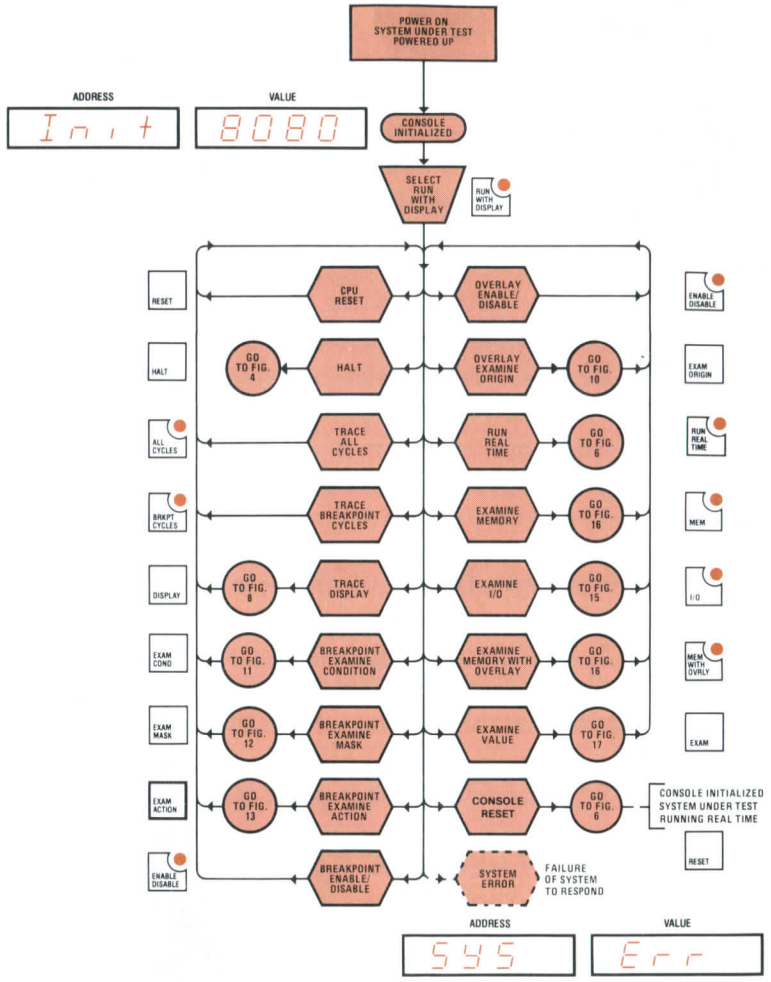



Fig. 5 Allowable Functions and Sequences With System Under Test Running With Display

**Breakpoint Examine Mask.** The breakpoint examine mask is the second sequence required for a breakpoint. This sequence lets you examine and assign a value of the μSCOPE 820 32-bit mask register. For a breakpoint to occur, the machine cycle must satisfy the mask register and the breakpoint condition register. When establishing a breakpoint mask, keep in mind that the mask performs the AND function bit by bit with the breakpoint condition. A 1-bit enables the corresponding bit of the condition and a 0-bit makes the corresponding bit of the condition a "don't care." You may find it necessary

to convert hexadecimal values to binary values in order to use the mask at its maximum potential. Figure 12 shows how to establish a breakpoint mask. Notice that the sequence is identical to the examine condition sequence.

**Breakpoint Examine Action.** The breakpoint examine action is the third required sequence of breakpoint. When a breakpoint match occurs, the μSCOPE 820 executes one of three actions. The examine action sequence is used to examine and select one of the three actions. The μSCOPE 820 initially selects the halt action. This means that the μSCOPE 820 halts (suspends) the System Under Test CPU at the first breakpoint match. Figure 13 is an operational flow chart that shows how to select a breakpoint action. The flow chart also shows how to enter a value into the μSCOPE 820 pass-count register or select an address of a subroutine.

**NOTE**

Execution of the instructions MOV C,M and MOV M,C are decoded by the μSCOPE 820 so that access is made to the System Under Test memory regardless of the overlay enable/disable. The μSCOPE 820 uses special instruction MOV M,C during execution of subroutine call on a breakpoint. Therefore, if a breakpoint action is call subroutine, and the overlay memory is enabled, the stack pointer should not be assigned a value that is within the overlay RAM. This will cause the subroutine to return to an unexpected location because the return address will have been written in the System Under Test memory rather than the μSCOPE 820 overlay memory.

**Breakpoint Enable/Disable.** Breakpoint enable/disable is a single-key function that controls the breakpoint hardware of the μSCOPE 820. By using breakpoint and trace memory together, you can set up the μSCOPE 820 to run a System Under Test operating sequence repetitively until a fault is detected. The sequence can then be narrowed until the problem is isolated.

**Trace All Cycles.** The trace all cycles mode with the System Under Test running with display is a single-key instruction. When using trace memory in the run with display mode keep in mind that the memory holds 256 CPU machine cycles and is constantly refreshed. You should therefore use μSCOPE 820 breakpoint to insure that you are able to record and analyze desired CPU cycles.

**Trace Breakpoint Cycles.** The trace breakpoint cycles mode with the system running with display is used in conjunction with μSCOPE 820 breakpoint. Only those CPU machine cycles that satisfy breakpoint are recorded. The breakpoint does not need to be enabled for the trace breakpoint cycles mode.

**Trace Display.** Trace display is a multiple-key sequence. Pressing the TRACE DISPLAY key ends the trace mode that is in effect and the last CPU machine cycle is displayed by the ADDRESS and VALUE readouts. Figure 8 is an operational flow chart showing how to increment or decrement trace memory. Keep in mind that when the trace display mode is initiated, the machine cycle displayed in the ADDRESS and VALUE readouts is the last cycle recorded.

**Overlay Memory Examine Origin.** The overlay examine origin sequence is used to examine and assign the starting address of the 1K or 2K μSCOPE 820 overlay memory. A switch on the personality probe selects the size of the overlay memory block (1K or 2K). You should power down the System Under Test before changing the size of the block. When you assign the starting address, be sure you select the first byte on a 1K or 2K boundary. If you assign an address that is not on the proper boundary, the μSCOPE 820 automatically sets the origin to the next lowest 1K or 2K boundary, depending upon the selected size of memory overlay block (1K or 2K). Figure 10 is an operational flow chart of the overlay memory examine origin sequence .

**Overlay Memory Enable/Disable.** The overlay memory enable/disable is a single-key function. After you have selected the size of the memory block and assigned a starting address, you enable or disable the μSCOPE 820 overlay memory with the ENABLE/DISABLE key.

**Examine Memory.** The examine memory sequence allows you to examine System Under Test memory and, if necessary, alter data in the memory locations. Figure 16 is an operational flow chart of the examine memory sequence. Under normal operation you should halt the CPU, initiate the examine memory sequence and enter the address of the memory location to be examined. You can then examine the

data and by using the EXAM NEXT or EXAM LAST keys to step through the stored data.

**Examine I/O.** The examine I/O sequence is the same as the examine memory sequence except that you are working with an 8-bit I/O number instead of a 16-bit address. Figure 15 is an operational flow chart of the examine I/O sequence.

**Examine Memory with Overlay.** The examine memory with overlay sequence is used when you assign μSCOPE 820 overlay memory over the System Under Test memory. The examine memory with overlay sequence is the same as the examine memory sequence except that you are examining and altering memory located in the μSCOPE 820 as well as the System Under Test. An operational flow chart of the examine memory with overlay sequence is shown in Figure 16.

**Value Examine.** The value examine sequence is one of the more complex functions of the μSCOPE 820, since it can be used to examine and—if necessary—alter the working single-byte registers, double-byte registers, stack pointer, program counter, flag register, and accumulator of the CPU. Pressing the EXAM key initiates the sequence. The ADDRESS readout prompts you to select the value to be examined. Figure 17 is an operational flow chart of the examine sequence.

**CPU Reset.** The CPU reset function is allowable while the system is running with display. Pressing the CPU RESET key resets the System Under Test CPU.

**Console Reset.** The console reset function is a hardware reset of the μSCOPE 820. Pressing the console reset key clears all μSCOPE 820 registers and places the μSCOPE 820 in the initialized state. The console reset also resets the System Under Test CPU.

## System Under Test Running Real Time

When the μSCOPE 820 is configured in the run real time condition, only front panel controls that do not alter System Under Test timing are allowed. These commands and related functions are shown in Figure 6 and described as follows:

**Halt.** Halt is a single-key instruction that transfers the run real time condition to the halt condition.

**Run With Display.** Run with display is also a single-key instruction that transfers the run real time condition to run with display. In this condition the μSCOPE 820 can steal cycles from the CPU to accomplish display update or to enable the setup of various μSCOPE 820 functions.

**Breakpoint Examine Conditions.** With the System Under Test running real time, you can examine the address, data byte, and status byte in the μSCOPE 820 breakpoint condition register. You cannot alter or change the contents of this register. The only valid key in this sequence is the END/EXEC key.

**Breakpoint Examine Mask.** The contents of the μSCOPE 820 breakpoint mask register can be examined; however, when the System Under Test is running real time you cannot alter or change the contents. The only valid key in this sequence is END/EXEC.

**Breakpoint Examine Action.** You can examine the μSCOPE 820 breakpoint action while the system is running real time; however, you cannot change the action, the contents of the μSCOPE 820 pass-count register, or the subroutine selected. The only valid key in this sequence is END/EXEC.

**Trace All Cycles.** The trace all cycles mode with the System Under Test running real time is a single-key instruction. When using trace memory in the run real time mode, keep in mind that the memory

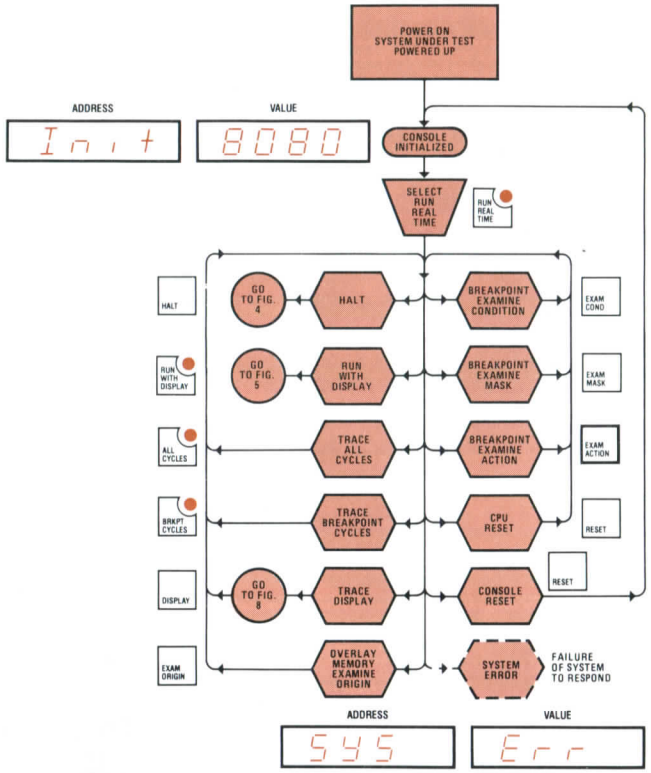



Fig. 6 Allowable Functions and Sequences With System Under Test Running Real Time

holds 256 CPU machine cycles and is constantly refreshed. You should therefore use μSCOPE 820 breakpoint to insure that you are able to record and analyze desired CPU cycles.

**Trace Breakpoint Cycles.** The trace breakpoint cycles mode with the system running real time is used in conjunction with μSCOPE 820 breakpoint. Only those CPU machine cycles that satisfy breakpoint are recorded. Tracing of breakpoint cycles will occur regardless of breakpoint status (enabled or disabled).

**Trace Display.** Trace display is a multiple-key sequence. Pressing the TRACE DISPLAY key ends the trace mode in effect and the last CPU machine cycle is displayed by the ADDRESS and VALUE readouts. Figure 8 is an operational flow chart showing how to increment or decrement trace memory to display recorded trace memory.

**Overlay Memory Examine Origin.** With the System Under Test running real time, you can examine the location of overlay memory, however, you cannot alter or change the starting address of overlay memory. After examining the starting address, the only valid key in this sequence is the END/EXEC key of the keypad.

**CPU Reset.** The CPU reset function is allowable with the System Under Test running real time. Pressing the CPU RESET key resets the System Under Test CPU.

**Console Reset.** The console reset function is a hardware reset of the μSCOPE 820. Pressing the CONSOLE RESET key clears all μSCOPE 820 registers and places the μSCOPE 820 in an initialized state. With the System Under Test running real time, the console reset also resets the System Under Test CPU.

## System Error

Should the μSCOPE 820 lose control of the System Under Test CPU, the ADDRESS and VALUE readouts will display "Sys Err". When a system error occurs, you can only disable breakpoint, examine CPU pins, or examine the contents of the μSCOPE 820 pass count register. However, you can use the trace functions by selecting trace all cycles or trace breakpoint cycles. You can also display the trace memory.

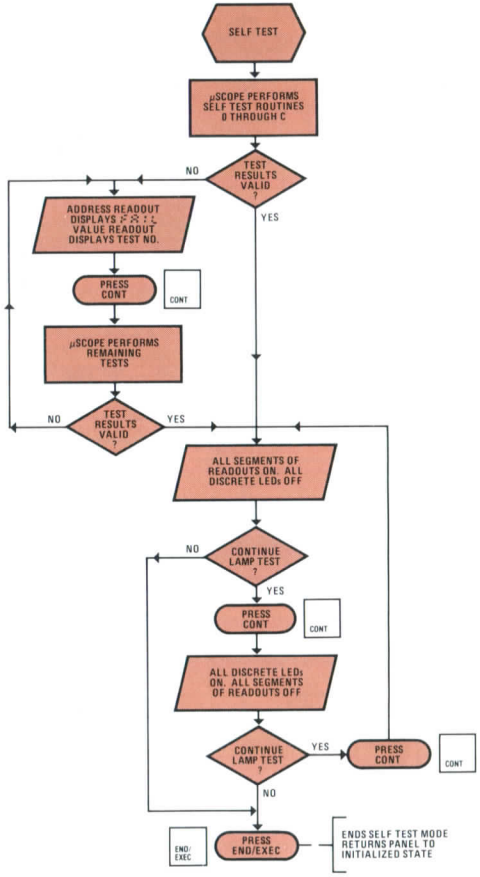



Fig. 7 Self Test Operational Flow Chart

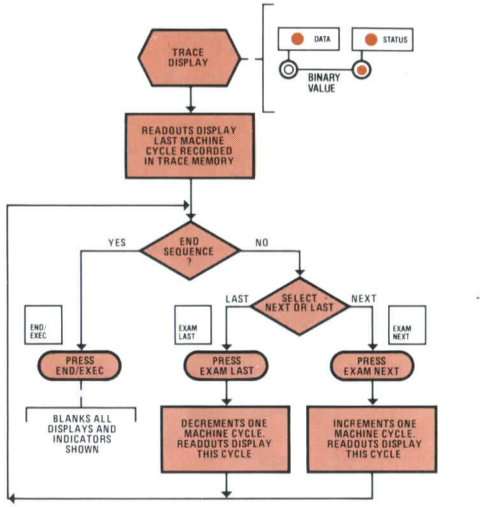



Fig. 8 Trace Display Operational Flow Chart

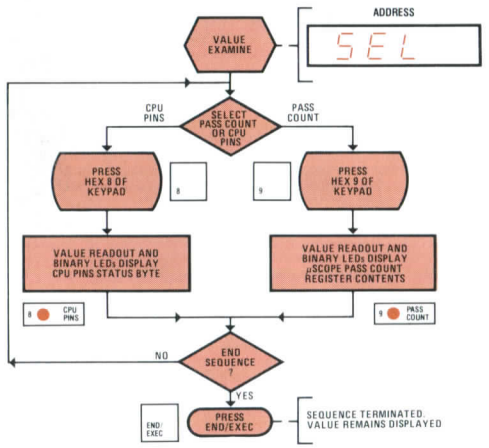



Fig. 9 Value Examine Operational Flow Chart (System Under Test Power Down)

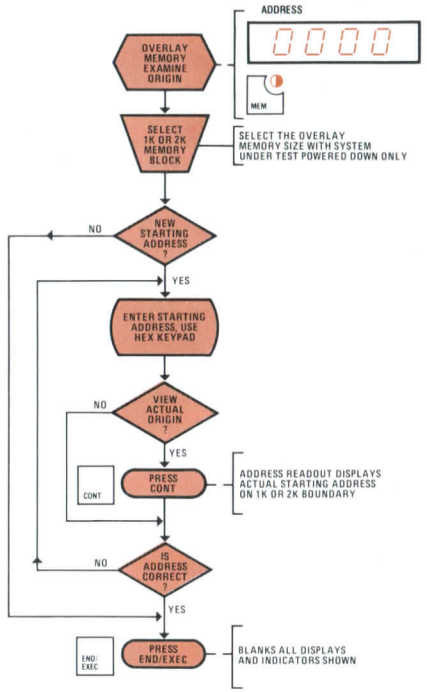



Fig. 10 Overlay Memory Examine Origin Operational Flow Chart

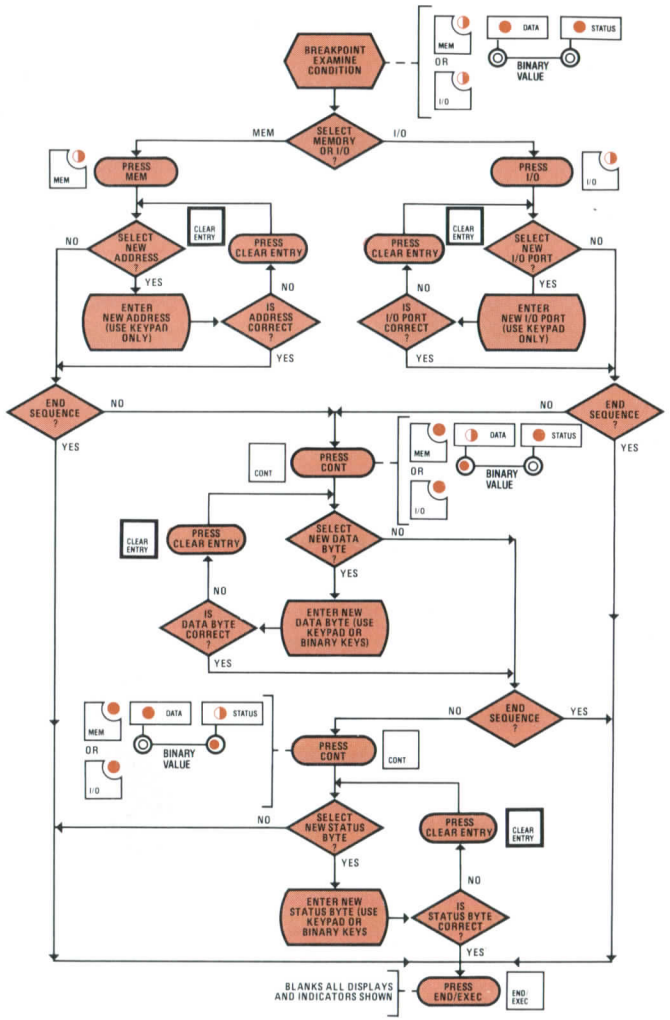



Fig. 11 Breakpoint Examine Condition Operational Flow Chart

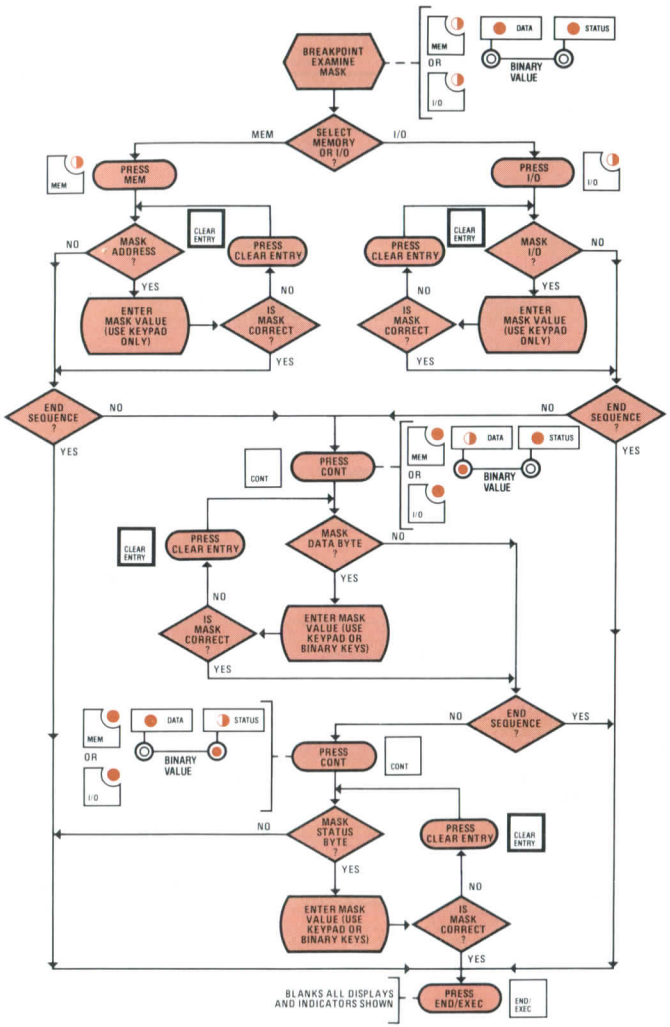



Fig. 12 Breakpoint Examine Mask Operational Flow Chart

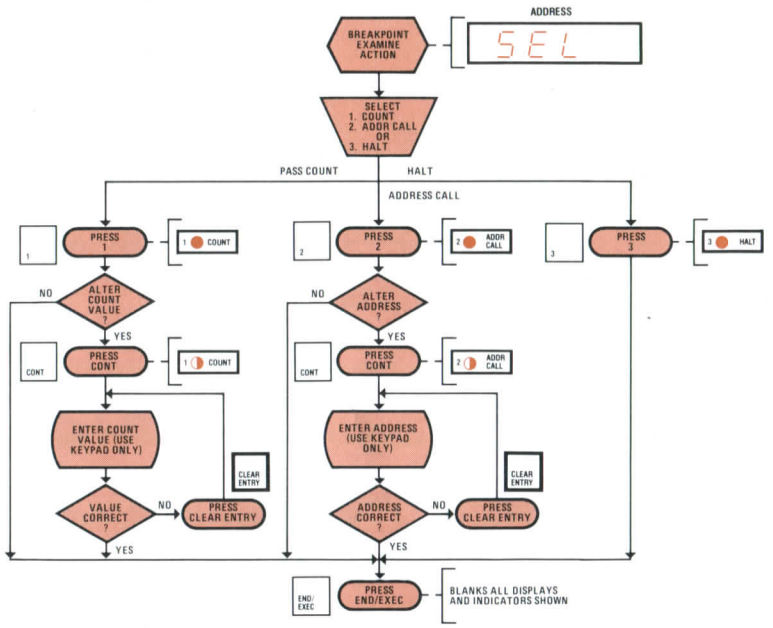



Fig. 13 Breakpoint Examine Action Operational Flow Chart

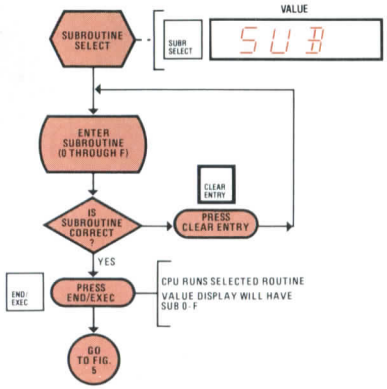



Fig. 14 Subroutine Select Operational Flow Chart

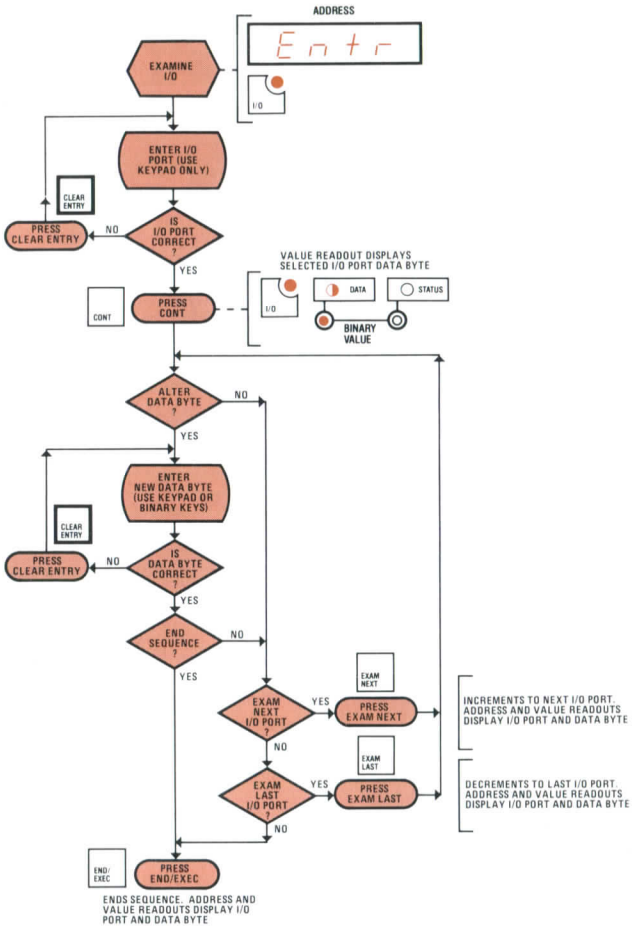



Fig. 15 Examine I/O Operational Flow Chart

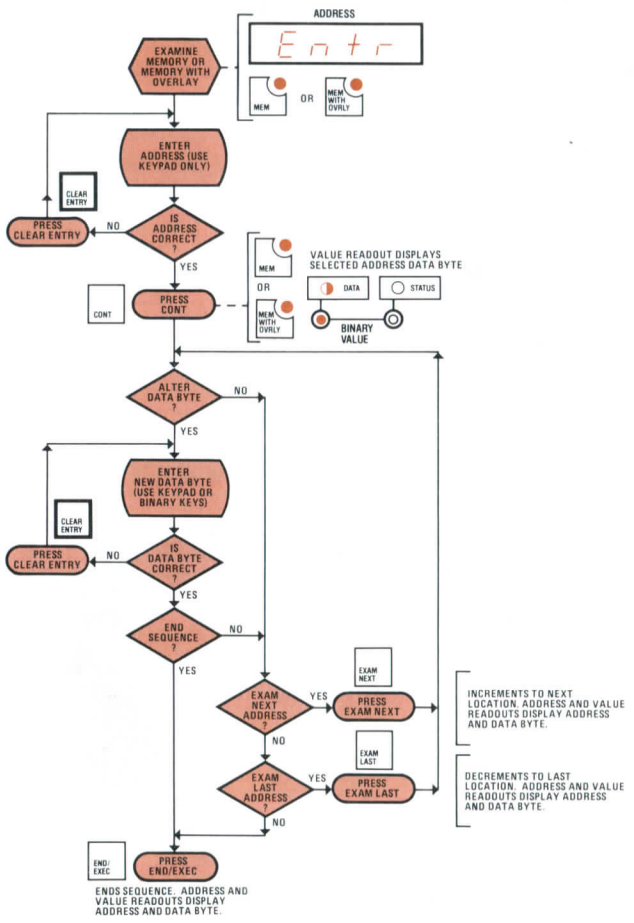



Fig. 16 Examine Memory or Memory With Overlay Operational Flow Chart

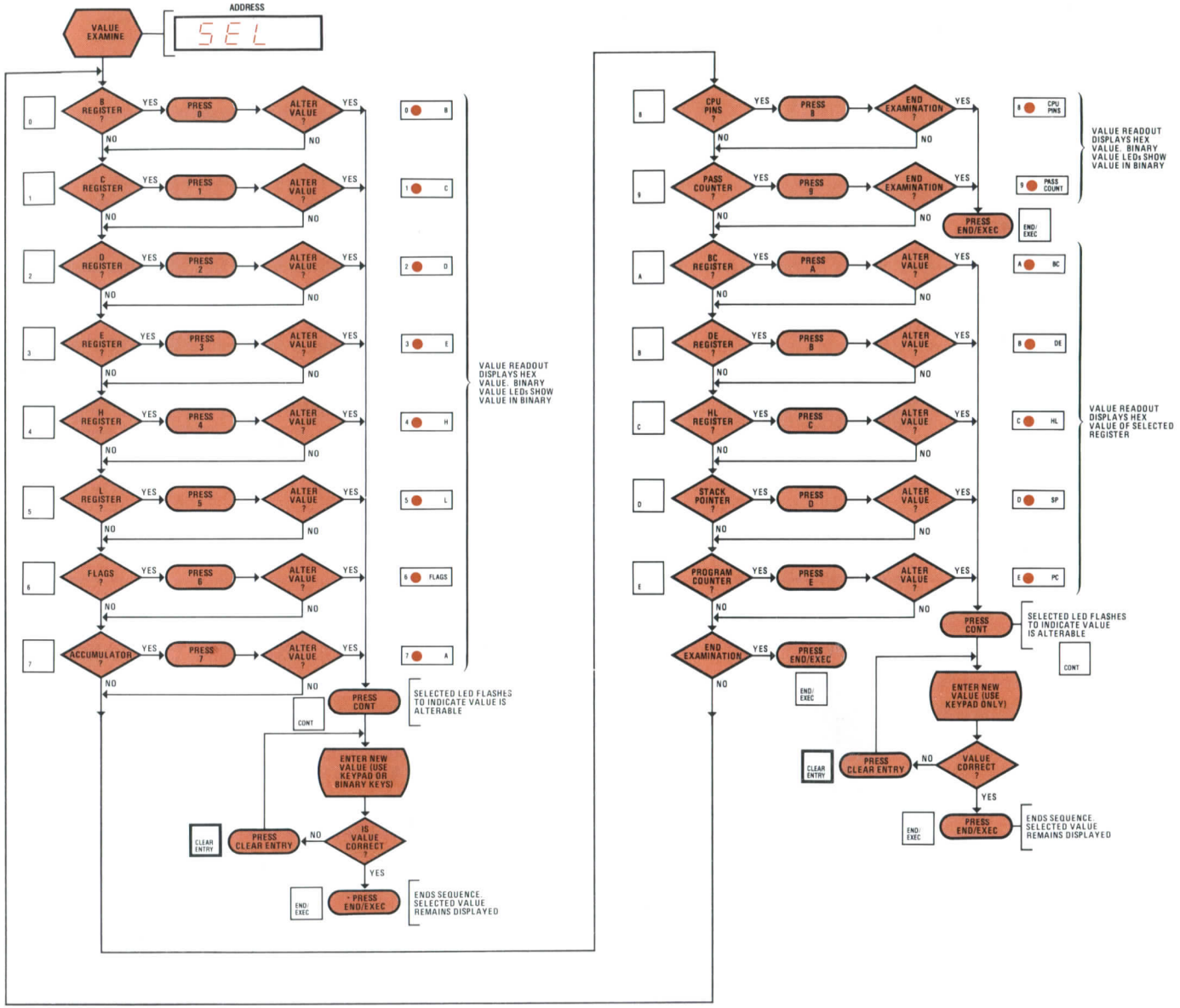



Fig. 17 Value Examine (CPU Working Registers) Operational Flow Chart

# Chapter 3
# Front Panel PROM Format

The ability to execute diagnostic subroutines independently of normal System Under Test program flow is one of the powerful μSCOPE 820 features. Diagnostic subroutines are generated and stored in a 2716 (2K × 8) EPROM or 2316E mask-programmed ROM. Subroutines are accessed by the System Under Test CPU through the PROM socket located on the μSCOPE 820 front panel.

This chapter explains the organization of the μSCOPE EPROM/ROM. To program the EPROM, you must be familiar with programming, understand the System Under Test and be familiar with the instruction set of the System Under Test CPU.

## EPROM/ROM Organization

The EPROM/ROM can have as many as 16 separate subroutines. Eight of these can be programmed with automatic μSCOPE 820 front-panel configurations; the remaining eight subroutines cannot have preset front panel capabilities. To provide these preprogrammed front-panel configurations, the μSCOPE 820 uses 128 bytes of the EPROM/ROM shadowed by the μSCOPE 820 overlay RAM. This area of ROM stores the starting-address vector table as well as the subroutine control blocks. Figure 18 shows the total EPROM/ROM organization.

The starting-address vector table occupies the first 32 bytes (0000 through 001F) with the starting addresses of the 16 subroutines. The remaining 96 bytes contain the eight automatic panel-configuration control blocks. Each of these control blocks is 12 bytes. The control bytes are accessed by the μSCOPE 820 only when routines 0 through 7 are selected. Therefore, routines that do not require front panel configurations should be assigned to the 8 through F subroutine selection range.

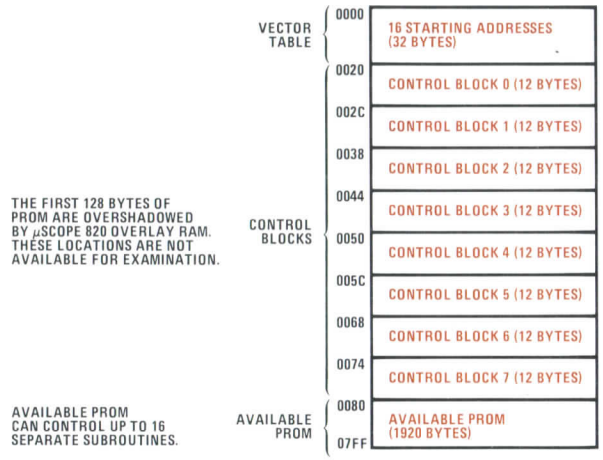



Fig. 18 Overlay PROM Organization

### Starting-Address Vector Table

The EPROM/ROM relative addresses 00 through 1F contain the starting address of each subroutine. Before a subroutine can be selected the System Under Test CPU must be halted, and the appropriate routine must be selected by pressing the SUBR SELECT key and a hex key (0-F) of the keypad. Pressing the END/EXEC key loads the corresponding beginning addresses into the System Under Test CPU and starts program execution in the run with display mode. The starting address selected can be that of a subroutine located in the EPROM/ROM or the beginning address of a subroutine resident in the System Under Test memory. Figure 19 illustrates the organization of the starting address vector table.

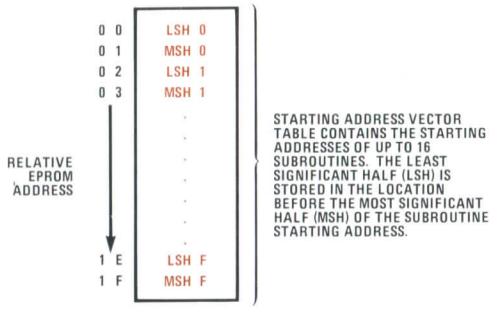



Fig. 19 Starting Address Vector Table

## Panel Configuration Control Block

The μSCOPE 820 accesses the control blocks only when subroutines 0 through 7 are selected. Selecting subroutine 0 causes the μSCOPE 820 to access control block 0, and so on. These blocks are always accessed when the first eight subroutines are selected. The panel configurations provided by the control blocks are as follows:

- CPU Control
- Trace memory
- Breakpoint
- Overlay Memory

Selecting CPU control allows resetting of the System Under Test CPU before selection of the run with display mode. Selecting trace memory configures the panel so the μSCOPE 820 traces all cycles, breakpoint cycles only, or does not trace at all. With the control block configuring the μSCOPE 820 for breakpoint, the breakpoint condition, mask and action can be preset. Overlay memory origin and enable/disable can also be set in the control block. Figure 20 shows the configuration of a typical control block. The starting address for each control block are located at 0020, 002C, 0038, 0044, 0050, 005C, 0068 and 0074.

## User-Available EPROM/ROM

The remaining 1920 bytes of EPROM/ROM beginning at relative address 0080 and ending at 07FF are available for storing diagnostic subroutines. This overlay memory can be displayed by the μSCOPE 820. Keep in mind that interrupts to the System Under Test are disabled when a diagnostic subroutine is selected with the SUBR SELECT key. The interrupts can be re-enabled by placing EI (enable interrupts) in the overlay code.

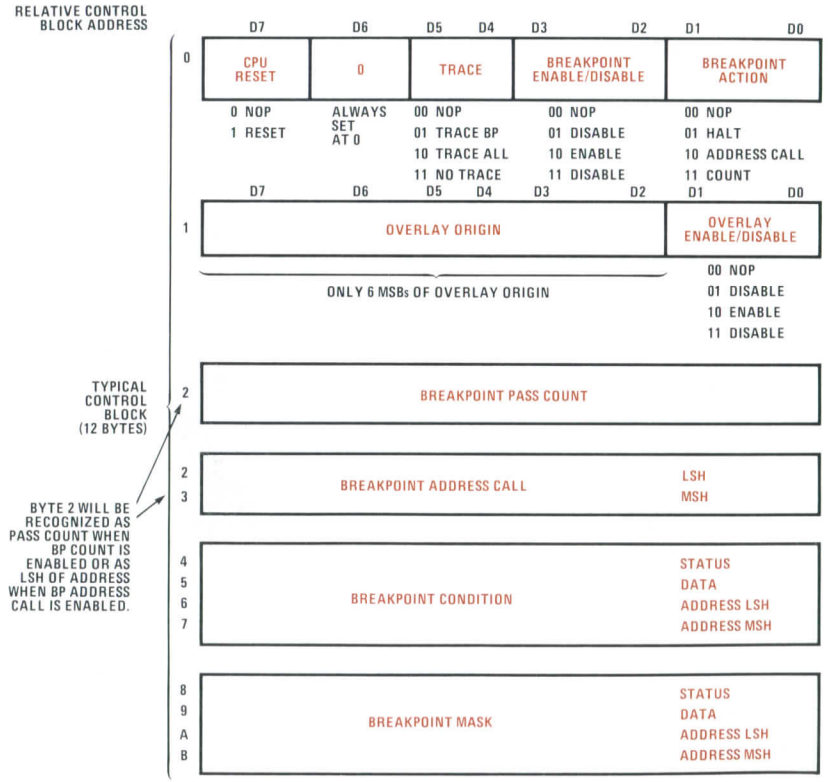



Fig. 20 Control Block Configuration

# Appendix A
# Self-Test Failure Codes

| Failure Code | Test Performed/Malfunctioning Area |
|---|---|
| 0 | Checksum failure of the 8755 EPROM (or 8355 ROM) |
| 1 | Checksum failure of the non-personality 2716 EPROM (or 2316 ROM) |
| 2 | Checksum failure of the personality 2716 EPROM (or 2316 ROM) |
| 3 | Memory check failure of the 8279 display RAM |
| 4 | Read-back check failure of the 8755 (or 8355) I/O ports |
| 5 | Failure of the alarm flip-flop to clear |
| 6 | Read-back check failure of the 8155 ports |
| 7 | Memory check failure of the 8155 RAM |
| 8 | Memory check failure of the 2-port RAM |
| 9 | Failure of a read-after-write check of a breakpoint condition or mask register |
| A | Failure of a read-after-write check of the override CAMs or override cycle latch. |
| B | Failure of trace memory to clear |
| C | Failure of a read-after-write check of the pass count register |

## REQUEST FOR READER'S COMMENTS

The Microcomputer Division Technical Publications Department attempts to provide documents that meet the needs of all Intel product users. This form lets you participate directly in the documentation process.

Please restrict your comments to the usability, accuracy, readability, organization, and completeness of this document.

1. Please specify by page any errors you found in this manual.

2. Does the document cover the information you expected or required? Please make suggestions for improvement.

3. Is this the right type of document for your needs? Is it at the right level? What other types of documents are needed?

4. Did you have any difficulty understanding descriptions or wording? Where?

5. Please rate this document on a scale of 1 to 10 with 10 being the best rating. _____

NAME _____ DATE _____
TITLE _____
COMPANY NAME/DEPARTMENT _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____

Please check here if you require a written reply. ☐

## WE'D LIKE YOUR COMMENTS . . .

This document is one of a series describing Intel products. Your comments on the back of this form will help us produce better manuals. Each reply will be carefully reviewed by the responsible person. All comments and suggestions become the property of Intel Corporation.

First Class
Permit No. 1040
Santa Clara, CA

**BUSINESS REPLY MAIL**

No Postage Stamp Necessary if Mailed in U.S.A.

*Postage will be paid by:*

**Intel Corporation**
3065 Bowers Avenue
Santa Clara, CA 95051

**Attention:** MCD Technical Publications

INTEL CORPORATION, 3065 Bowers Avenue, Santa Clara, California 95051 (408) 987-8080

Printed in U.S.A./B68/0778/CP 5K