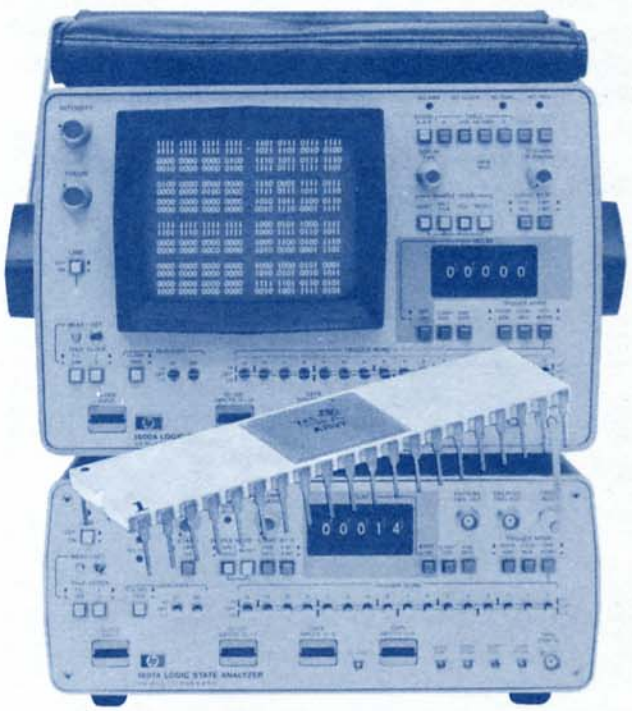
APPLICATION NOTE 167-14
DATA DOMAIN MEASUREMENT SERIES

Functional analysis of 8080 microprocessor systems.



HEWLETT hp PACKARD

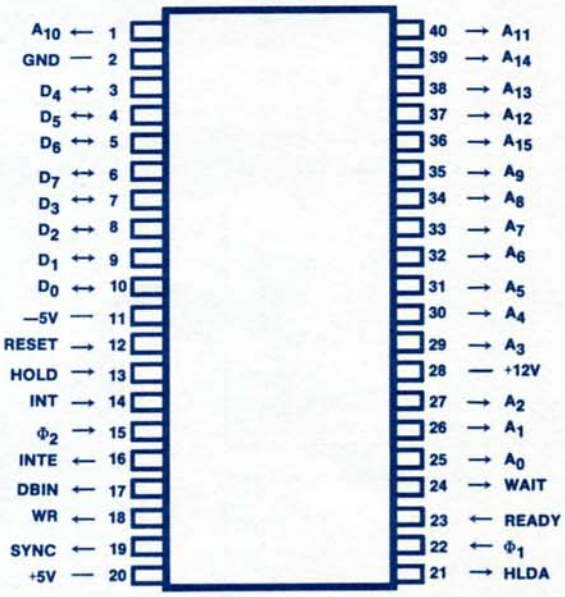
1. INTRODUCTION

This application note is intended to assist the 8080 microprocessor user in the real-time analysis of his system in both the design and troubleshooting environments. This note demonstrates real-time analysis of program flow, triggering on specific events, and paging techniques.

The 8080 microprocessor, core of the 8080 microprocessor family, is constructed with NMOS technology and operates from +12 V, +5 V, and —5 V power sources. Features of the 8080 include an 8-bit, bidirectional, 3-state data bus and a separate 16-bit, 3-state address bus. The 16-bit address bus permits direct addressing of 65k words of memory.

Six timing and control outputs are available from the 8080, while four control inputs and two clock signals are required by the 8080. All buses are TTL compatible. The 8080 operates with a 2 MHz clock.

2. PIN ASSIGNMENTS

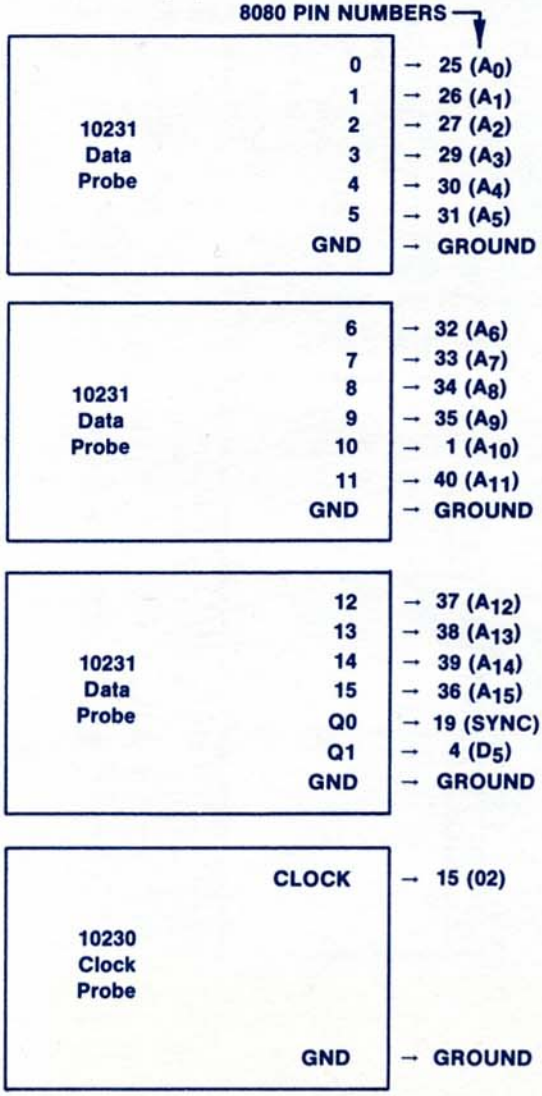


PIN NAME	FUNCTION
A_{15}-A_0	Address to memory or I/O device number for up to 256 input and 256 output devices. A_0 is LSB.
D_7-D_0	Bidirectional communication between memory, CPU, and I/O devices.
SYNC	Signal to indicate beginning of each machine cycle.
DBIN	Date Bus In signal indicates to external circuits that data bus is in input mode. (Enables gating data from I/O or memory onto data bus.)
READY	Valid memory or input data available on data bus. Used to synchronize 8080 with slower memory or I/O devices.
WAIT	Acknowledges that 8080 is in a wait state.
$\overline{WR}$	The WRITE signal used for memory write or I/O output control.
HOLD	Requests 8080 to enter HOLD state.
HLDA	The HOLD ACKNOWLEDGE signal responds to HOLD and indicates that address and data buses will go to high impedance state.
INTE	Interrupt Enable signal indicates content of internal interrupt enable flip-flop. Inhibits interrupt when flip-flop is reset.
INT	Interrupt request is recognized at end of current instruction or while halted.
RESET	While activated, the program counter is reset, program will start at 0 in memory. INTE and HLDA are reset.
01, 02	External clocks; non-TTL compatible.

For More Information, Call Your Local HP Sales Office or, in U.S., East (201) 265-5000. Midwest (312) 677-0400. South (404) 436-6181. West (213) 877-1282. Or, Write: Hewlett-Packard, 1501 Page Mill Road, Palo Alto, California 94304. In Europe, Post Office Box 85, CH-1217 Meyrin 2, Geneva, Switzerland. In Japan, YHP, 1-59-1, Yoyogi, Shibuya-Ku, Tokyo, 151.

3. PROBE CONNECTIONS

A system that will not "come up" can frequently be debugged by monitoring address flow alone. This is a simple process with the 8080 system. Connect the Logic State Analyzer probes to the 8080 as shown to observe a display of the activity on the address lines.



4. SETTING THE CONTROLS

Turn power on and set Logic State Analyzer controls as follows:

Display Mode	Table A
Sample Mode	REPET[1]
Trigger Mode	
NORM/ARM	NORM
LOCAL/BUS	LOCAL
OFF/WORD	WORD
START DISPLAY	ON
CLOCK	⌐¬
THLD	TTL
4-BIT/3-BIT	4-BIT
All other pushbuttons	Out Position
DISPLAY TIME	ccw
COLUMN BLANKING	ccw
QUALIFIER Q1, OFF-Q0, HI	
TRIGGER WORD Switches	Set to match Address that you wish to trigger on

[1]If program is not looping or cycling through the selected address, select SGL sample mode, press RESET, and start the system. The first time the system passes through the trigger point, the display will be generated and stored.

5. DISPLAY INTERPRETATION

In this section, system response to a CALL instruction is illustrated. The CALL instruction initiates a subroutine to check the keyboard for the presence of a stop command and to check system status. Proper operation is confirmed by a comparison of real-time state analysis, figure 1a, and the 8080 cross assembler listing output, figure 1b.

The 8080 responds to a CALL instruction in the following manner:

1. Store content of the program counter in the push-down address stack.
2. Jump unconditionally to the instruction in memory location addressed by byte two and byte 3 of the CALL instruction.
3. Begin execution of the subroutine.

Examine the program listing in figure 1b. Line 1

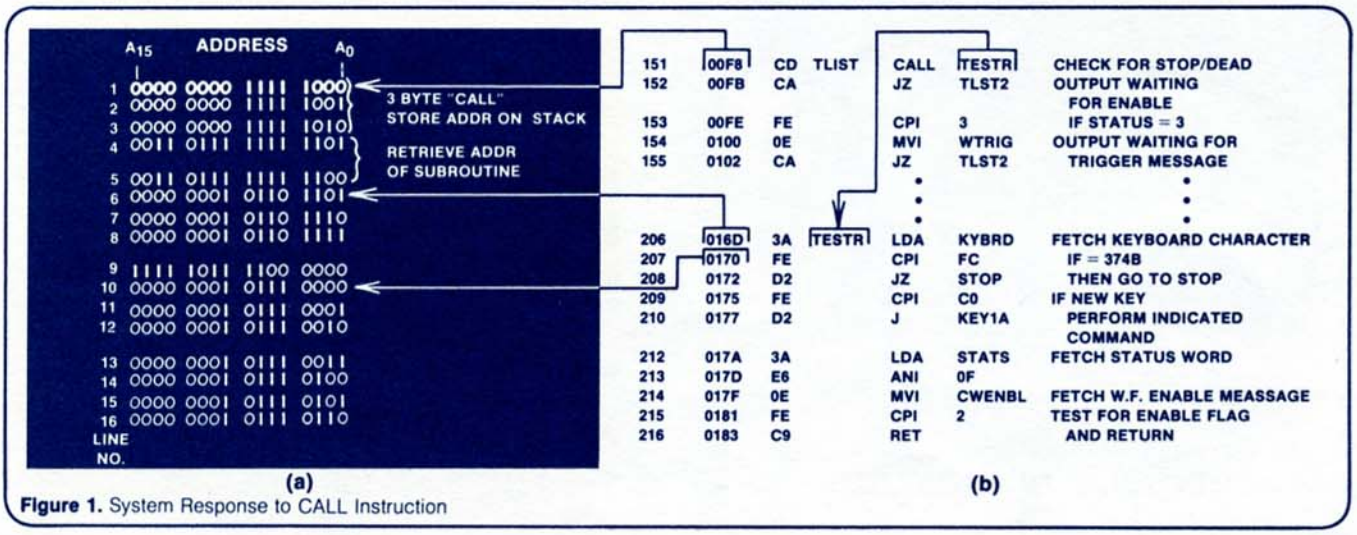


Figure 1. System Response to CALL Instruction

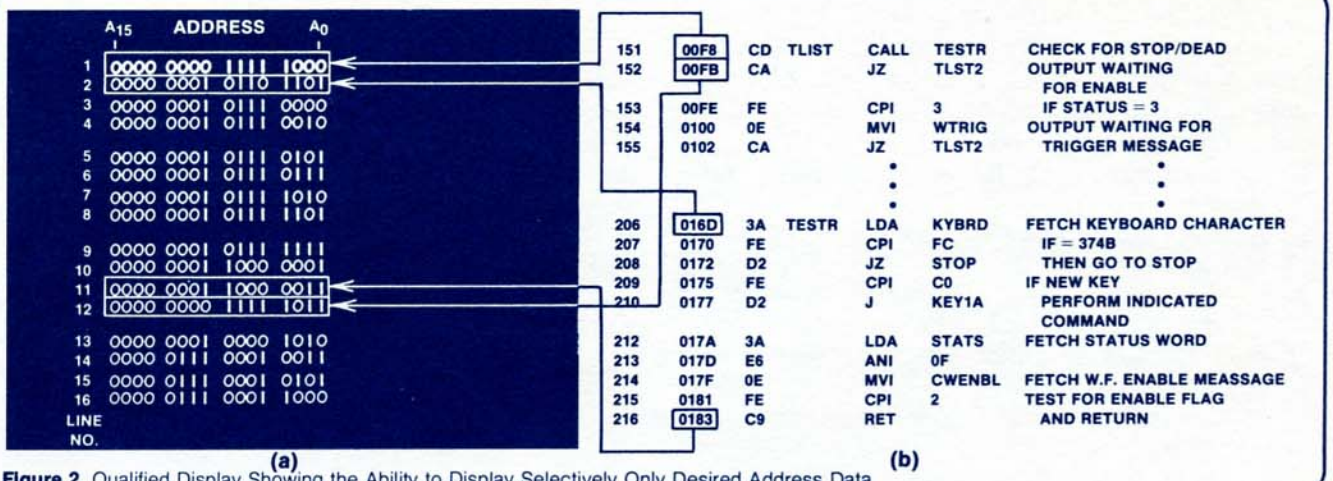


Figure 2. Qualified Display Showing the Ability to Display Selectively Only Desired Address Data

displays the address of the CALL instruction, 00F8. Therefore, during the next four machine cycles: the high-order 8 bits of the next instruction address are stored in the push-down stack; the low-order 8 bits of the next instruction address are stored in the push-down stack; the stack pointer is decremented by 2; and control is transferred to the first address of the subroutine.

Proper operation of the CALL instruction is confirmed by observing that the address on line 6 of the display is the address of the first subroutine instruction. Again, there are multiple cycle operations between first and second subroutine instructions. In similar fashion, address lines may be compared with the 8080 instruction set operation to verify proper subroutine operation.

To view addresses following the last displayed address (view the next "page"), simply set the Trigger Word switches to match the address displayed in line 16. This address then becomes the trigger word in line 1 of the display with the next 15 addresses displayed on lines 2 through 16. Or, if you wish to retain the original address word on the Trigger Word switches, an alternate method is to set the digital delay to 00015 to achieve the desired display.

6. SELECTIVE STORE

It may be desirable not to look at every address, but only those corresponding to Instruction Fetch cycles. We can do this using the Logic State Analyzer "qualifier" feature. Looking at the sample program in figure 1b, it is apparent that the subroutine is 10 instructions long with most instructions requiring at least two memory locations. Because of this, we cannot view the entire subroutine on the 16-word display in figure 1.

By qualifying the display on Instruction Fetch cycles, it is possible to look at addresses corresponding only to instruction fetch operations. This is accomplished in the following manner:

1. Connect Q1 line from the 10231 Data Probe to pin 4 (D_5) of the 8080.
2. Set Q1 switch to HI.

D_5 line of the 8080 goes HI at the beginning of each instruction fetch cycle and we now obtain the display shown in figure 2a. Comparing the table display with the program listing reveals that line 1 is the address for the CALL instruction, lines 2 through 11 are the subroutine and line 12 is the return to the main program. This gives us an overview of the entire subroutine.

7. THE MAP

If a tabular display is not presented in Sections 5 and 6, it means the system did not access the selected address and the No Trigger light will be on. To find where the system is residing in the program, switch to MAP display (figure 3). Using the Trigger Word switches move the cursor (circle in photo) to encircle one of the dots on the screen. Switch to EXPAND and make final positioning of the cursor. The No Trigger light will go out and pressing Table A will display the 16 addresses around that point.

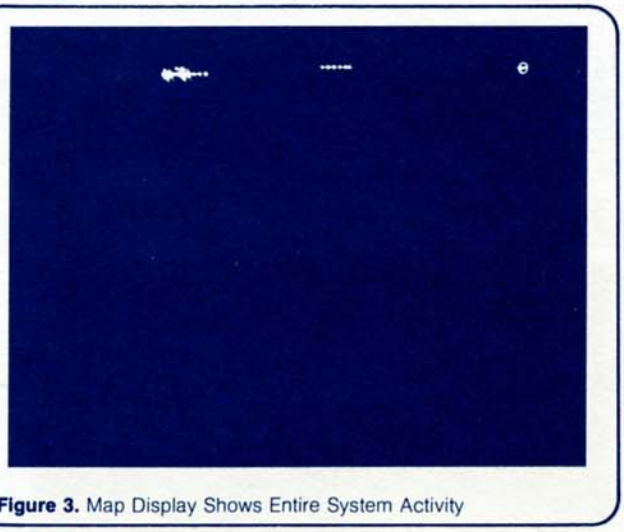


Figure 3. Map Display Shows Entire System Activity

8. VIEWING ADDRESS, DATA, AND CONTROLS

When program deviations are found, the reason may be as simple as a program error or as complicated as a hardware failure on the Data/Control Bus, or other

command lines. Additional input channels now become very desirable.

By combining the 1600A and 1607A, the display and trigger capability can be expanded to 32-bits wide, allowing the 16-bit address, 8 bits of data and up to 8 other active command signals to be viewed simultaneously. The hook-up is easy.

1. Connect data cable between rear panel connectors.
2. Connect trigger bus cable between front panel bus connectors.
3. Connect 1600A data and clock probes as in Section 3 with the following exceptions: connect Q0 line to DBIN (pin 17); connect Q1 line to $\overline{WR}$ (pin 18).
4. Connect 1607A data and clock probes as follows:
   a. Connect 1607A data inputs 0 through 7 to D_0 through D_7 in order.
   b. Connect Q0 to DBIN (pin 17) and Q1 to WR (pin 18).
   c. Connect clock input to 02 (pin 15).
   d. Connect ground(s) to appropriate point(s).
5. Set 1600A controls as detailed in Section 4 with the following exceptions: set Table Display to A+B, set LOCAL/BUS to BUS, set Q1 to HI.
6. Set 1607A controls as follows:

Sample Mode	SGL
Trigger Mode	
NORM/ARM	NORM
LOCAL/BUS	BUS
OFF/WORD	WORD
START DSPL	ON
CLOCK	⌐
THLD	TTL
DISPLAY TIME	CCW
QUALIFIER Q1/Q0	HI
TRIGGER WORD	OFF (Don't Care)
4-BIT/3-BIT	4-BIT
All Other Pushbuttons	Out Position

7. After a display is on the 1600A screen, set 1607A Column Blanking to display 9 columns.

9. DISPLAY INTERPRETATION OF ADDRESS AND DATA BUS/CONTROL LINES

Let's look again at the program for the CALL instruction, figure 4b, and the display in figure 4a. By displaying both address and data, it is possible to confirm system operation with respect to the CALL instruction. The single bit on the right table display indicates that all Read instructions are being displayed. Looking at line 1 of the left table display, observe that the address, 00F8 corresponds to the program address for the CALL instruction. Also on line 1, the data display corresponds to the operating code, CD, for the subroutine being called. Address on line 2 shows the data bus reading the least significant 8-bits of the subroutine address, while line 3 shows the data bus reading the most significant bits of the subroutine address. Since the control line is qualified to show only read operations, we do not see the program counter being written into the push-down stack memory. Line 4 confirms that the subroutine address is called and the operating code for a keyboard fetch is displayed on the data lines.

In a similar manner, each line of the display can be examined to reveal exact system operation.

Application Notes in the 167 series with the primary instrument(s) used in parenthesis.

167-1 The Logic Analyzer (5000A)
167-2 Digital Triggering for Analog Measurements (1601L)
167-3 Functional Digital Analysis (1601L)
167-4 Engineering in The Data Domain Calls for a New Kind of Digital Instrument (Describes measurement problems and various solutions with applicable instruments.)
167-5 Troubleshooting in the Data Domain is Simplified by Logic Analyzers (1600A and 1607A)
167-6 Mapping, a Dynamic Display of Digital System Operation (1600A)
167-7 Supplementary Data from Map Displays without Changing Probes (1600A)
167-8 Stable Displays of Disc System Waveforms Synchronized to Record Address (1620A)
167-9 Functional Analysis of Motorola M6800 Microprocessor Systems (1600A and 1607A)
167-10 Using the 1620A for Serial Pattern Recognition (1620A)
167-11 Functional Analysis of Intel 8008 Microprocessor Systems (1600A and 1607A)
167-12 Functional Analysis of Fairchild F8 Microprocessor Systems (1600A and 1607A)
167-13 The Role of Logic State Analyzers in Microprocessor Based Designs (1600A and 1607A)
167-14 Functional Analysis of 8080 Microprocessor Systems (1600A and 1607A)
167-15 Functional Analysis of Intel 4004 Microprocessor Systems (1600A and 1607A)
167-16 Functional Analysis of Intel 4040 Microprocessor Systems (1600A and 1607A)
167-17 Functional Analysis of National IMP Microprocessor Systems (1600A and 1607A)

VIDEO TAPE SERIES: This four hour series titled "The Data Domain Its Analysis and Measurements" introduces logic state analysis and measurement techniques unique to the data domain. Contact your HP Field Engineer for price and availability of this color tape series.

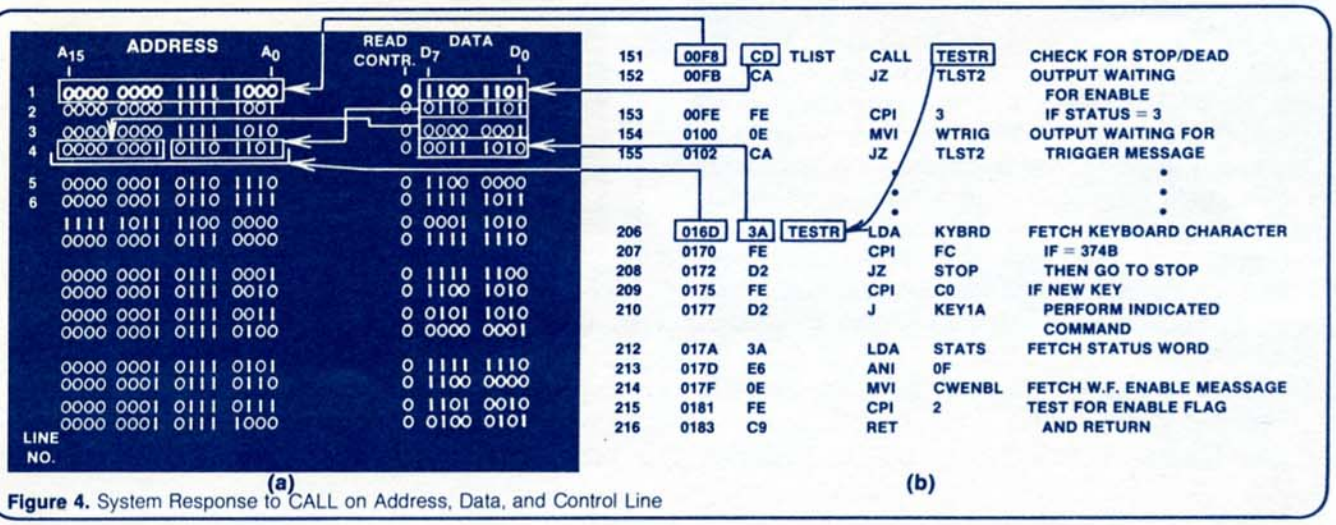


Figure 4. System Response to CALL on Address, Data, and Control Line

PRINTED IN U.S.A.