Tektronix
COMMITTED TO EXCELLENCE

1220/1225/1230
LOGIC ANALYZER

PM406

6809
Microprocessor
Probe

Operator's Manual

The PM406 has a software version number of 2.51. For use with the PM406, the 1220 and 1225 Logic Analyzers require software versions of 2.5 or above; the 1230 Logic Analyzer requires a software version of 3.03 or above.

Please check for change information at the back of this manual

070-6596-01
PRODUCT GROUP 43

FIRST PRINTING JUNE 1988

Tektronix, Inc.
Walker Road Industrial Park
P.O. Box 4600
Beaverton, Or. 97076

TABLE OF CONTENTS

List of Figures

List of Tables

OVERVIEW

The PM406 6809 Microprocessor Probe Personality Module consists of a disassembly probe (with ribbon cables) and this user's manual. This manual shows you how to connect and use the PM406 disassembly probe with the 1220/1225/1230 Logic Analyzers. This manual does not teach you how to use analyzer keypads or menus. For information on using the logic analyzer, refer to the operator's manual for your particular logic analyzer. For more information about the 6809 microprocessor, refer to your microprocessor data book.

The PM406 Version 2.51 firmware works having 1220/1225 Logic Analyzers having software version number 2.5 or higher, and with 1230 Logic Analyzers having software version numbers 3.03 or higher. If you're using a 1220/1225 version 2.5 or higher, or a 1230 version 3.03 or higher, you must use the version 2.51 PM406. You can see what version of analyzer system software you are using by reading the opening menu when you turn on the unit. You can see what version of the PM406 on the first page of Notes in the Disassembly menu. The PM406 must be connected properly to the analyzer.

The PM406 gives you an interface between the 1220/1225/1230 Logic Analyzer and 6509-based systems under test (SUT). Along with regular analyzer features, the PM406 interface lets you sample data synchronously using the SUT clock, and lets you display disassembly data in hardware and software formats.

Conventions. This manual uses these conventions:

- The term analyzer refers to the 1220, 1225, and 1230 Logic Analyzers unless otherwise specified.
- The term SUT refers to the 6809 system under test.
- Active low signals are identified by a bar over the signal name, for example, $\overline{NMI}$.

ANALYZER CONFIGURATION

You must have at least 32 channels in the analyzer to use the PM406. This is because the probe uses 32 channels to acquire synchronous data from the 6809-based SUT. You must also use a version 2.51 PM406 if you're using either a 1220/1225 version 2.5 (or higher) or a 1230 version 3.03 (or higher). Figure 1 shows the analyzer and expansion card configuration.

CONNECTING AND POWERING UP

The PM406 has two probe cables that connect to the analyzer. Figures 1 and 2 show how the analyzer connects to your SUT.

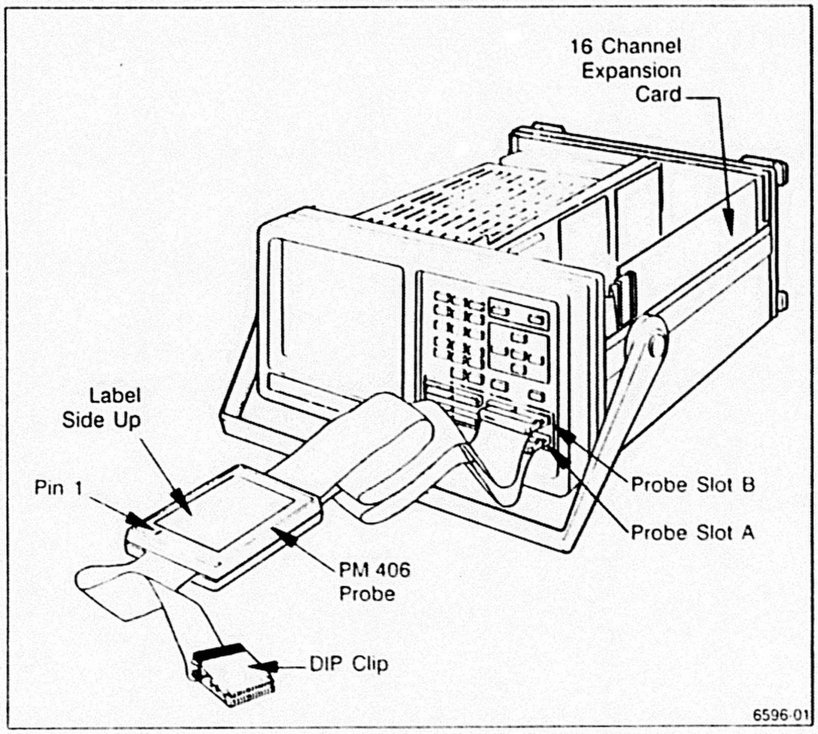


Figure 1. Analyzer configuration with PM406 probe. Note that the bottom cable plugs into probe slot A and the top cable plugs into probe slot B.

To connect the PM406 to your SUT, follow these steps:

1. Make sure that the power to the analyzer and SUT is off.

CAUTION

Do not connect the PM406 to the analyzer unless power to analyzer is off. Do not connect the PM406 disassembly probe to the SUT unless power to the SUT is off. If you connect the disassembly probe to the SUT when power to the SUT is on and power to the analyzer is off, too much power can flow through the probe's inputs and damage the probe.

2. With the PM406 label side up, connect the bottom cable from the probe to input A on the front of the analyzer.
3. Connect the top cable from the probe to input B on the front of the analyzer.

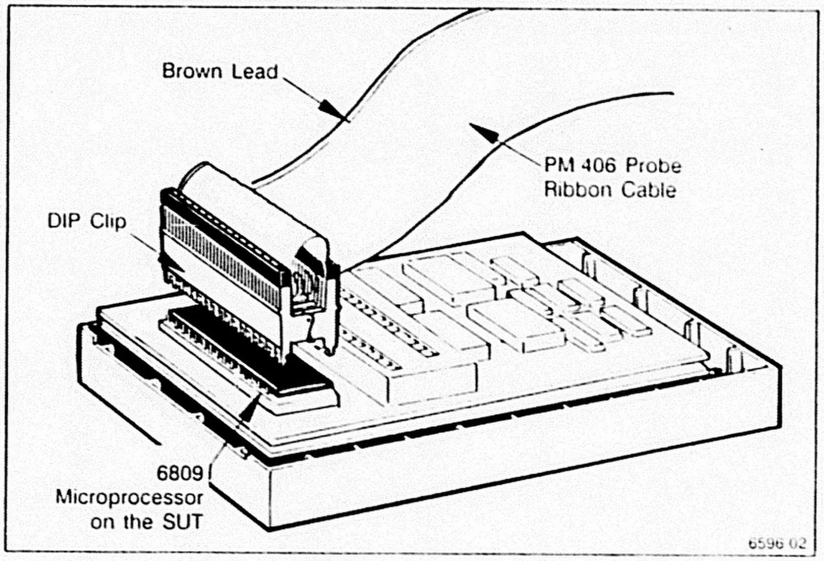


Figure 2. Connecting the DIP clip and SUT.

4. Connect the PM406 probe clip to the SUT as shown in Figure 2 (power to the SUT should be off). The brown lead labeled PIN 1 on the PM406 goes to pin 1 on the 6809 microprocessor. Figure 3 shows the 6809 pinout, and Table 1 lists analyzer-to-6809 signal line connections. Figure 3 and Table 1 are shown after this procedure.

5. Turn on the analyzer; this also supplies power to the probe. The analyzer screen now displays the Initialization menu (Figure 4, shown after this procedure).

6. Press ENTER to upload the PM406 disassembly setup into the analyzer. Pressing ENTER overwrites the existing setup and changes probe links, channel groups, and defined conditions for 6809 disassembly. If you press MENU, the PM406 setup is not uploaded and the analyzer's current setup is unchanged.

7. Turn on power to the SUT.

At this point the analyzer displays the Main menu (Figure 5), which lists setup, data, and utility menus. Since the default disassembly setup defines the setup parameters for you, (probe links, sampling rate and format, conditions, and so on), you can press START at any time to acquire data from your SUT. Example 1, later in this manual, shows a data acquisition with the default setup.

Signal Name	6809 Pin Numbers		Signal Name
Vss	1	40	$\overline{HALT}$
$\overline{NMI}$	2	39	XTAL
$\overline{IRQ}$	3	38	EXTAL
$\overline{FIRQ}$	4	37	$\overline{RESET}$
BS	5	36	MRDY
BA	6	35	Q
Vcc	7	34	E
A0	8	33	$\overline{DMA/BREQ}$
A10	9	32	R/$\overline{W}$
A2	10	31	D0
A3	11	30	D1
A4	12	29	D2
A5	13	28	D3
A6	14	27	D4
A7	15	26	D5
A8	16	25	D6
A9	17	24	D7
A10	18	23	A15
A11	19	22	A14
A12	20	21	A13

6596 03

Figure 3. 6809 pinout.

Table 1
6809 Signals and Analyzer Channels

6809 Signals	122x/1230 Channels	Channel Groups	Description
A15-A00	B15-B00	ADD	Address bus
D07-D00	A15-A08	DAT	Data bus
BA BS	A02 A01	BUS	Processor state
R/$\overline{W}$	A00	BUS	Read/Write
$\overline{NMI}$ $\overline{FIRQ}$ $\overline{IRQ}$	A05 A04 A03	INT	Interrupts
$\overline{HALT}$ $\overline{DMA/BREQ}$	A07 A06	CTL	Program control

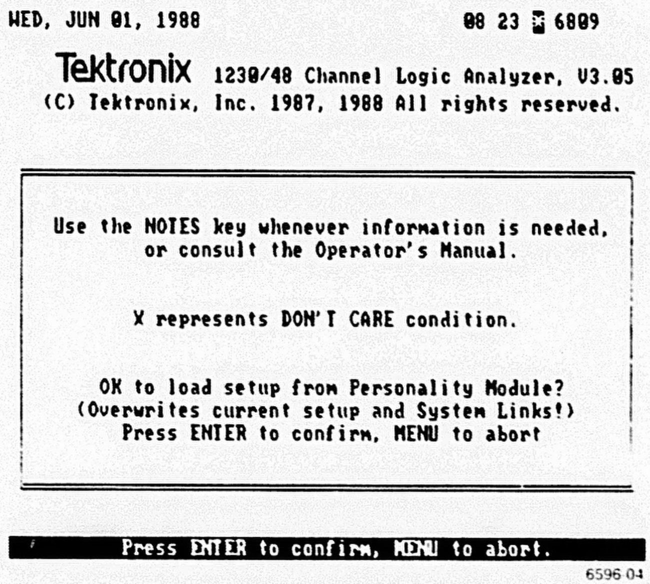
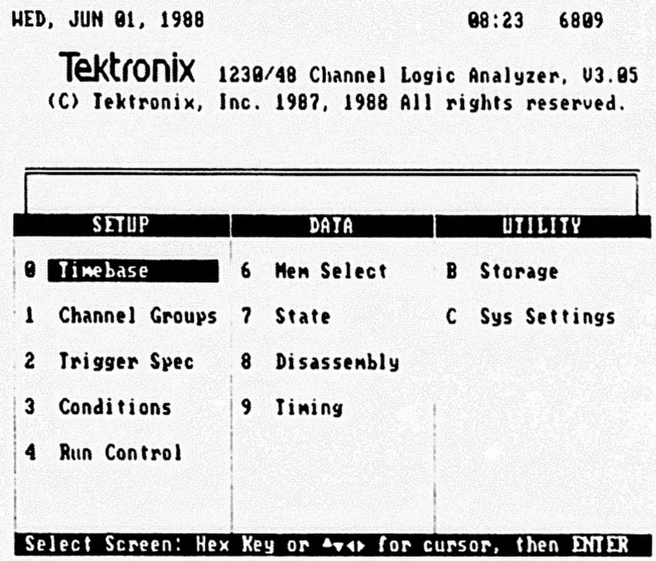


Figure 4. Initialization menu. When you turn the analyzer on with the PM406 plugged in, the Initialization menu includes a message telling you that you can now upload the disassembly setup by pressing ENTER.



6596-05

Figure 5. Main menu. The Main menu always shows disassembly as a menu selection. However, you can display acquired data in disassembly format only when the PM406 is plugged in. As long as the acquisition memory is valid, you can display valid disassembly data.

Loading Disassembly Setups. You don't have to upload the disassembly setup when you see the Initialization menu. However, if you don't, you must enter the disassembly setup manually or reset the analyzer so that the PM406 can upload the disassembly setup for you. You can reset the analyzer by firmly pressing NOTES and ENTER at the same time.

Using Probes

The PM406 must always be plugged into slots A and B on the analyzer front panel. If you have a 1220/1225 Logic Analyzer, you can use slot C for an acquisition probe. If you have a 1230 Logic Analyzer, you can use slots C and D for acquisition probes.

You can use both an acquisition probe and the disassembly probe. The acquisition and disassembly probes can be used together or separately without unplugging either of these.

The probe in slot A must always be connected to the clock in your SUT. If the probe in slot A is not connected to your SUT clock, the analyzer won't trigger when you press START. If you're using more than one probe and the probes are linked synchronously, each probe must be connected to the same clock point in your SUT. Therefore, your connection to the SUT clock is assured.

Using the Menus and Cursor

The PM406 is controlled by selections you make in the analyzer's menus. You can always call up the Main menu by pressing MENU.

The analyzer looks at the probe inputs to find out if the PM406 is connected. For more information about using the menus and cursor, refer to the operator's manual for your particular logic analyzer.

Online Help

At the bottom of the disassembly screen, a one-line help message tells you which keys to press for disassembly functions. If you need more help, press NOTES while the disassembly screen is displayed. The analyzer then displays six pages of in-depth information about 6809 disassembly, including the disassembler's software version number which appears on the first page of disassembly notes. You can press MENU at any time to exit the notes and return to the previous display.

SETTING UP TO ACQUIRE DATA

This discussion shows you how the PM406 sets up the analyzer for 6809 disassembly. The setups shown here are for an analyzer with 32 channels. Example 1, later in this manual, shows a data acquisition using this 32-channel default setup.

A setup is a set of parameters that describes the current analyzer configuration for data acquisition and storage. For example, the setup includes information about probe links, acquisition rates, threshold voltage, and 6809 trigger conditions.

You can use the setup as it is uploaded from the PM406, or you can change any part of the configuration. While the discussion in this section is about the default PM406 setup, you are free to change any part of the analyzer configuration manually.

Timebase

The acquisition timebase, probe links, and threshold voltage for 6809 disassembly are shown in Figure 6. If you're using a 1230, the PM406 uses the synchronous clock rate of your SUT. If you're using a 1220 or 1225, the PM406 is set up for synchronous acquisition at 100 ns or slower.

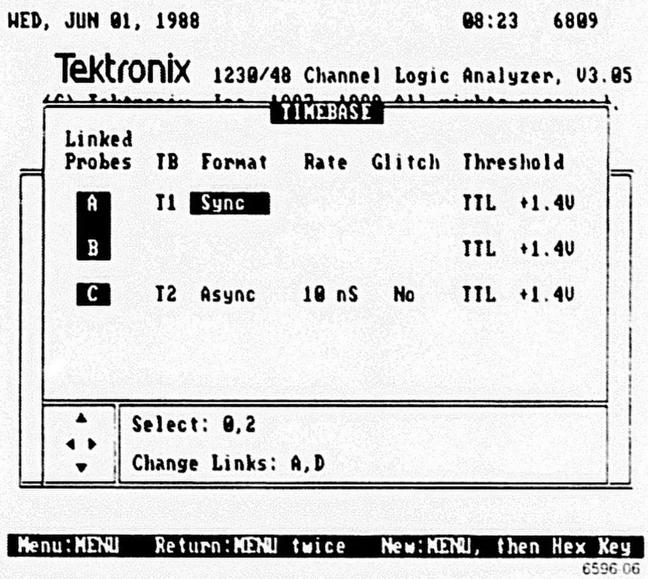


Figure 6. Timebase menu. Probes A and B must be linked synchronously for correct disassembly.

Probe Links. The PM406 is a 32-channel disassembly probe which uses probe slots A and B. For 6809 disassembly, probes A and B are linked together synchronously in timebase T1 so that all disassembly is done with the same acquisition format and rate. If you're also using one or more acquisition probes in addition to the PM406, the acquisition probes are linked asynchronously in T2.

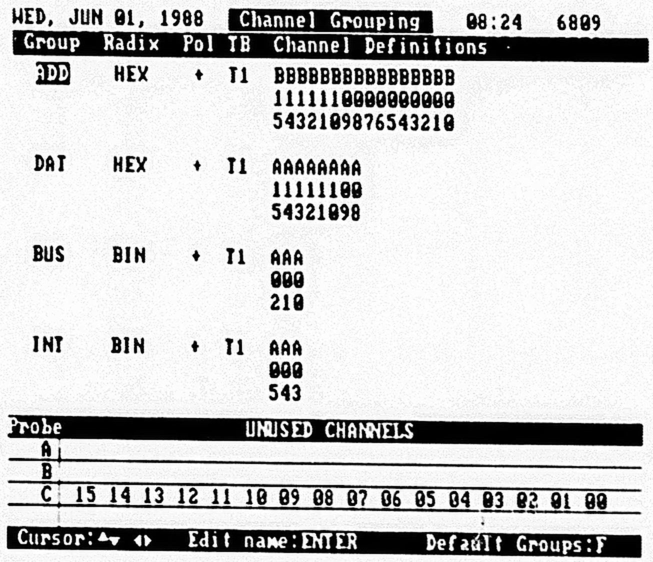
Clocking. The default disassembly clock format is synchronous so that you use the clock rate in your SUT as the data sampling rate. The PM406 automatically qualifies your SUT clock with software internal to the probe. There are no external clock qualifiers for the PM406.

For the 1230, the clock rate is set by your SUT. For the 1220/1225, the clock rate is set to ≥100 ns by default. For 6809 disassembly, you must use a clock rate of ≥100 ns if you're using a 1220/1225.

Glitch Capture. The PM406 does not acquire glitches.

Channel Grouping

The PM406 sets up five channel groups: ADD, DAT, BUS, INT, and CTL, as shown in Figure 7. The screen is large enough to see four groups. To see more groups, scroll up or down the screen. The Channel Grouping menu shows how the channel groups are named; for example, ADD for the address bus. The control lines are separated into three channel groups: BUS, INT (interrupt), and CTL (control). Channels of extra probe(s) are assigned to the asynchronous timebase (T2) are in the unused list. If you want to use those channels, you must manually add them to a group.



6596 07

Figure 7. Channel Grouping menu. The screen shows four channel groups. To see more channel groups, scroll up or down the screen.

Trigger Conditions

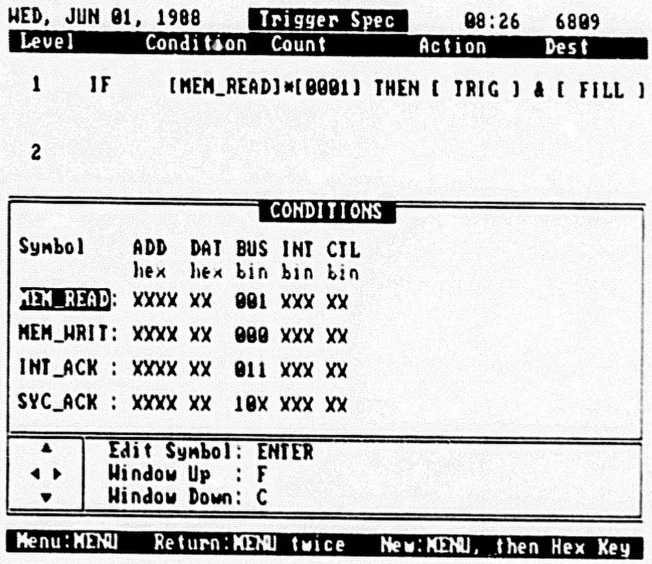
The Conditions menu lets you define data conditions which the analyzer can recognize and trigger on. When you upload the 6809 setup, the 6809 input signals are already grouped to correspond to the analyzer channels as listed earlier in Table 1. The conditions listed in Table 2 show the logic states corresponding to 6809 operations.

Table 2
6809 Cycle Types and Analyzer Conditions

Signal Line	ADD hex	DAT hex	BUS bin	INT bin	CTL bin
MEM READ	X X X X	X X	0 0 1	X X X	X X
MEM WRIT	X X X X	X X	0 0 0	X X X	X X
INT ACK	X X X X	X X	0 1 1	X X X	X X
SYC ACK	X X X X	X X	1 0 X	X X X	X X
HALT ACK	X X X X	X X	1 1 X	X X X	0 X
HALT REQ	X X X X	X X	X X X	X X X	0 X
BG ACK	X X X X	X X	1 1 X	X X X	X 0
BG REQ	X X X X	X X	X X X	X X X	X 0
VMA CYC	F F F F	X X	X 0 1	X X X	X X
RESET	F F F E	X X	X X X	X X X	X X
NMI	X X X X	X X	X X X	0 X X	X X
FIRQ	X X X X	X X	X X X	X 0 X	X X
IRQ	X X X X	X X	X X X	X X 0	X X
SWI1	F F F A	X X	X X X	X X X	X X
SWI2	F F F 4	X X	X X X	X X X	X X
SWI3	F F F 2	X X	X X X	X X X	X X

All signals are sampled synchronously with a 6809 machine cycle, except for $\overline{NMI}$ (nonmaskable interrupt). $\overline{NMI}$ is displayed only at the negative edge for one cycle.

Figure 8 shows the Conditions and Trigger Spec menus. The trigger statement shown in the figure is for a 1230. If you're using a 1220/1225, the default trigger action is START instead of TRIG.



6596-08

Figure 8. Conditions and Trigger Spec menus. The default condition window is large enough to show four defined conditions. Table 2 lists all signals/conditions defined for the PM406. The default trigger statement is an if-then statement with the first condition, MEM READ, as the trigger condition. For the 1230, the trigger action is TRIG. For the 1220/1225, the trigger action is START.

Trigger Specification

The default trigger statement is an if-then statement. At initialization, the analyzer is set to trigger and fill memory when the condition MEM READ occurs. Figure 8 shows the Trigger Spec menu along with the Conditions menu.

Run Control

When you load the 6809 setup, the Run Control menu is set up as shown in Figure 9. The default display for acquired data is a disassembly display. The trigger position is set at memory location 1024, and the analyzer looks for the trigger after the pretrigger memory is full.

The Run Control menu also sets the memory-compare mode to Manual and tells you that the default channel mask for comparing memories is MEM READ, which is also the default trigger condition. A window (or viewport) at the bottom of the screen lists the value for MEM READ. Remember that channels set to X (don't care) are masked, or not compared, during a memory comparison.

```
WED, JUN 01, 1988        Run Control        08 26    6809

 Update Memory    : [1]        Display: [Disassembly]
 Trigger Position: [1024]      0 ____________ 2K
 Look for Trigger: [After Pre-Trigger Memory Full]
=============================================================
 Compare          : [Manual]
 Compare Memory 1 to Memory: [2]


 Compare Mem Locations: [0000] to [1747]
 Use Channel Mask     : [MEM_READ]
 Display Data at least: [5] seconds

             ADD  DAT BUS INT CTL
 Symbol      hex  hex bin bin bin
 MEM_READ:   XXXX XX  001 XXX XX

 Cursor: ▲▼◄►        Select: 0,2
```

6596-09

Figure 9. Run Control menu. The defaults in this menu include the display format set to disassembly and the trigger position set to 1024.

SETTING UP TO DISASSEMBLE CODE

Once you've set up the analyzer for disassembly, you can start to acquire and display data from your SUT. The operator's manual for your particular logic analyzer tells how to display data in state and timing formats. This discussion shows you how to display disassembled 6809 data, which you can do only when the PM406 is connected to the analyzer.

As long as the acquisition memory is valid the disassembly display is also valid. Channel grouping is only used for the timing and state displays.

Disassembly Mnemonics. The PM406 lets you display acquired data in disassembly mnemonics. Disassembly mnemonics are assembly-language instructions that have been disassembled from a machine-language program. For example, 6809 disassembly mnemonics include JMP, CMP, NOP, and DEC instructions. An actual disassembly line might read ADDD, which means "add the memory byte to the D accumulator." Figures 12 and 13 show examples of disassembly mnemonics.

Displaying in Hardware or Software Mode. With the PM406 attached, you can display disassembled data in hardware or software mode. In hardware display mode, the analyzer shows all bus operations and displays every acquired cycle. In software display mode, the analyzer shows only instructions; reads and writes are suppressed so that the display looks like an assembly listing. You can toggle between display modes by pressing DON'T CARE.

After an acquisition, the PM406 deduces when opcode fetches occurred by analyzing acquired data and cycle types. The PM406 then displays disassembled instructions from the *location of the cursor* to the end of memory. If the cursor is near the end of memory, no instructions (or very few) may be displayed. Figures 10 and 11 show hardware and software mode examples where few disassembled instructions are shown because the cursor was positioned at the end of memory.

To re-disassemble the entire memory, use the Jump function to move the cursor to memory location 0000 (press ENTER then type 0000 on the hex keypad). Then, press 6 to mark the first valid opcode fetch. Figures 12 and 13 show re-disassembled data.

Mark Opcode. The Mark Opcode function lets you indicate the beginning of an opcode sequence. To mark an opcode, move the cursor to a Memory Read cycle then press the 6. The acquisition memory is re-disassembled from that location. If the selected cycle is not a Memory Read, then marking starts at the next Memory Read available. Marking can only be done in hardware mode.

```
WED, JUN 01, 1988   Disasm: Memory 1   08:27    6809
Loc   Addr Data 6809  Disassembly   Operation  Status
2035  FFFF 01                       NOT VMA    DMA/BREQ
2036  F573 26                       MEM READ   DMA/BREQ
2037  F574 F4                       MEM READ   DMA/BREQ
2038  FFFF 01                       NOT VMA    DMA/BREQ
2039  F569 BD                       MEM READ   DMA/BREQ
2040  F56A F6                       MEM READ   DMA/BREQ
2041  F56B DB                       MEM READ   DMA/BREQ
2042  FFFF 01                       NOT VMA    DMA/BREQ
2043  F6DB A6                       MEM READ   DMA/BREQ
2044  FFFF 01                       NOT VMA    DMA/BREQ
2045  CF63 6C                       MEM WRITE  DMA/BREQ
2046  CF62 F5                       MEM WRITE  DMA/BREQ
2047  F6DB A6                       MEM READ   DMA/BREQ

0000  0000 00    NEG    <20         OPC FETCH  HALT
0001  F6F1 20                       MEM READ   DMA/BREQ
0002  F6F2 EE                       MEM READ   DMA/BREQ
0003  FFFF 01                       NOT VMA    DMA/BREQ
0004  F6E1 E7                       MEM READ   DMA/BREQ
0005  F6E2 84                       MEM READ   DMA/BREQ
Func:F    Scroll: ▾▴     Cursor: ◂▸     Jump: ENTER
```

6596-10

Figure 10. Disassembly mnemonics in hardware mode. This data was displayed immediately after acquisition was completed. Because the cursor was near the end of memory, very little data was disassembled.

```
WED, JUN 01, 1988   Disasm: Memory 1   08 27    6809
Loc   Addr Data       6809 Disassembly     Operation
0000  0000 0020       NEG     <20
0001  F6F1 20EE       BRA     F6E1         CF62=F5A6

0000  0000 0020       NEG     <20
0001  F6F1 20EE       BRA     F6E1         CF62=F5A6

0000  0000 0020       NEG     <20
0001  F6F1 20EE       BRA     F6E1         CF62=F5A6

0000  0000 0020       NEG     <20
0001  F6F1 20EE       BRA     F6E1         CF62=F5A6

0000  0000 0020       NEG     <20
0001  F6F1 20EE       BRA     F6E1         CF62=F5A6

0000  0000 0020       NEG     <20
0001  F6F1 20EE       BRA     F6E1         CF62=F5A6

0000  0000 0020       NEG     <20
0001  F6F1 20EE       BRA     F6E1         CF62=F5A6
Func:F    Scroll Rate: 7,8 [02]    Mode: X [Software]
```

6596-11

Figure 11. Disassembly mnemonics in software mode. This is the corresponding software display for Figure 10. Note that the information displayed is not valid.

```
WED, JUN 01, 1988   Disasm: Memory 1   08 41     6809
 Loc   Addr Data 6809   Disassembly    Operation  Status

-0000--0000-00---NEG----<20------------OPC FETCH--HALT----
 0001  F6F1 20   BRA     F6E1          OPC FETCH  DMA/BREQ
 0002  F6F2 EE                         MEM READ   DMA/BREQ
 0003  FFFF 01                         NOT VMA    DMA/BREQ
 0004  F6E1 E7   STB     ,X            OPC FETCH  DMA/BREQ
 0005  F6E2 84                         MEM READ   DMA/BREQ
 0006  F6E3 A6                         MEM READ   DMA/BREQ
 0007  C874 AA                         MEM WRITE  DMA/BREQ
 0008  F6E3 A6   LDA     ,X            OPC FETCH  DMA/BREQ
 0009  F6E4 84                         MEM READ   DMA/BREQ
 0010  F6E5 34                         MEM READ   DMA/BREQ
 0011  C874 AA                         MEM READ   DMA/BREQ
 0012  F6E5 34   PSHS    B             OPC FETCH  DMA/BREQ
 0013  F6E6 04                         MEM READ   DMA/BREQ
 0014  FFFF 01                         NOT VMA    DMA/BREQ
 0015  FFFF 01                         NOT VMA    DMA/BREQ
 0016  CF61 DA                         MEM READ   DMA/BREQ
 0017  CF60 AA                         MEM WRITE  DMA/BREQ
 0018  F6E7 A0   SUBA    ,S+           OPC FETCH  DMA/BREQ
 0019  F6E8 E0                         MEM READ   DMA/BREQ

 Func:F     Scroll Rate: 7,8 [02]     Mode: X [Hardware]
```

6596-12

Figure 12. Re-disassembled disassembly mnemonics in hardware mode. Now that the cursor has been jumped to location 0000 and an opcode fetch has been marked, the analyzer now displays valid data.

```
WED, JUN 01, 1988   Disasm: Memory 1   08:41    6809
 Loc   Addr Data          6809 Disassembly    Operation

-0000--0000-0020-----------NEG------<20--------------------
 0001  F6F1 20EE           BRA      F6E1
 0004  F6E1 E784           STB      ,X         C874=AA
 0008  F6E3 A684           LDA      ,X         C874=AA
 0012  F6E5 3404           PSHS     B          CF61=DAAA
 0018  F6E7 A0E0           SUBA     ,S+        CF60=AA
 0024  F6E9 260E           BNE      F6F9
 0027  F6EB C155           CMPB     #55
 0029  F6ED 2604           BNE      F6F3
 0032  F6F3 3502           PULS     A          CF61=DAF5
 0038  F6F5 A784           STA      ,X         C874=DA
 0042  F6F7 4F             CLRA
 0044  F6F8 39             RTS                 CF62=F56C
 0049  F56C 2607           BNE      F575
 0052  F56E 3001           LEAX     01,X
 0057  F570 BCCF33         CMPX     CF33       CF33=D000
 0064  F573 26F4           BNE      F569
 0067  F569 BDF6DB         JSR      F6DB       CF63=6CF5
 0075  F6DB A684           LDA      ,X         C875=DA
 0079  F6DD 3402           PSHS     A          CF62=F5DA

 Func:F     Scroll Rate: 7,8 [02]     Mode: X [Software]
```

6596-13

Figure 13. Re-disassembled disassembly mnemonics in software mode. Here is the software display which corresponds to Figure 12. Now the analyzer displays valid data.

Invalid Codes. The analyzer displays question marks (???) when there isn't enough information to determine the beginning of a valid opcode fetch. Figure 14 shows an example. Use the Mark Opcode function to indicate the beginning of an opcode sequence.

WED, JUN 01, 1988		Disasm: Memory 1		09 22 6809
Loc	Addr Data	6809 Disassembly		Operation
0000	0000 0002	NEG	<02	
0001	F3F9 02	???		F3F0=3402
0000	0000 0002	NEG	<02	
0001	F3F9 02	???		F3F0=3402
0000	0000 0002	NEG	<02	
0001	F3F9 02	???		F3F0=3402
0000	0000 0002	NEG	<02	
0001	F3F9 02	???		F3F0=3402
0000	0000 0002	NEG	<02	
0001	F3F9 02	???		F3F0=3402
0000	0000 0002	NEG	<02	
0001	F3F9 02	???		F3F0=3402
0000	0000 0002	NEG	<02	
0001	F3F9 02	???		F3F0=3402
Func:F	Scroll Rate: 7,8 [20]		Mode: X [Software]	

6596-14

Figure 14. Invalid or unknown opcode displayed in hardware mode. An opcode which the PM406 cannot interpret is shown as ???.

Searching for Events. Searching for events in the Disassembly menu works the same as searching for events in the State Menu. Press 0 or 2 to cycle through the available conditions (including ·' .e trigger event). Press 1 to perform the search.

When the analyzer finds the search event, it redisplays the disassembly screen so that the cursor is positioned in the middle of the screen indicating the search event. If you searched for an event that did not occur, the analyzer displays the message Not Found. One of the menu bars at the bottom of the screen lists the current search event. For more information about searching, refer to the operator's manual for your particular logic analyzer.

The analyzer can display and search for opcode fetches in software mode. However, since reads and writes (which are not opcode fetches) are suppressed in software mode, the analyzer cannot display those instructions if you try to search for them. In software mode, if you search for an event that is not an opcode fetch and it is found, the analyzer sets the cursor to the previous opcode fetch and displays the the search event in the middle of the screen.

Using the Hardware Display Mode

For disassembly displays in hardware mode, the analyzer displays each sample location with address and data from the 6809 bus cycle. Disassembled instructions are displayed at the beginning of each valid machine cycle. Figure 12, earlier in this discussion, shows a hardware disassembly display.

In the displays, the Loc column shows memory locations. The Addr column shows the address, and the Data column displays the data bus. When the PM406 recognizes the beginning of an instruction, the analyzer disassembles that instruction and displays it in the middle column.

BA, BS and R/W are decoded into cycle-type information and displayed in the Operation column. Table 3 shows each Operation type display and its definition.

Table 3
Operation Column

Cycles	Description
OPC FETCH	Opcode fetch
MEM READ	Memory read
MEM WRITE	Memory write
INT ACK	Interrupt acknowledge
SYNC ACK	Sync acknowledge
HALT ACK	Halt acknowledge
NOT VMA	Not valid memory address

The last column displays the status of interrupt lines. In this column, the interrupt with the highest priority is listed. For example, if an $\overline{\text{NMI}}$ and $\overline{\text{IRQ}}$ occur at the same time, the $\overline{\text{NMI}}$ signal is listed in the display. Table 4 lists interrupt priorities.

Table 4
Status Column Interrupt Priorities

Active Line	Description
$\overline{HALT}$	Halt
$\overline{DMA/BREQ}$	Direct memory access/bus request
$\overline{NMI}$	Nonmaskable interrupt
$\overline{FIRQ}$	Fast interrupt request
$\overline{IRQ}$	Interrupt request

Pressing DON'T CARE while in the hardware display mode toggles the disassembly screen to the software display mode, and vice versa.

When you press DON'T CARE to switch display modes, the analyzer goes through memory to find the opcode fetch closest to the cursor position. When it finds the opcode fetch, the analyzer displays the disassembly in software mode with the cursor in the middle of the screen. If it can't find an opcode fetch, the analyzer returns to hardware mode.

Using the Software Display Mode

The software display mode is useful because it displays only instructions; memory reads and writes are suppressed. The display resembles an assembly or program listing because it shows only one opcode fetch per line and each line must be the start of an instruction sequence. Because of this, the locations displayed are not contiguous. Figure 13, earlier in this discussion, shows a software disassembly display.

The Operation column lists the bus operations for the instruction sequence. For each instruction cycle, the analyzer uses the Operation column to tell you the memory address and data activity for that cycle. In this column, the address is displayed on the left of the equals sign; data is displayed on the right. Figure 13 shows address and data information.

Searching for Events. You can search for events in the software disassembly display the same as you search for events in the State menu. However, because memory reads and writes are suppressed, if you search for an event that occurs on a memory read or write cycle, the analyzer displays the instruction that caused the memory read or memory write. To search for a memory read or write cycle, press DON'T CARE to toggle to hardware mode, then select the search event, and then press 1 to search.

EXAMPLES

These four examples show you how to acquire data for disassembly, how to display the data in hardware and software modes, how to trigger on a specific event, how to use multiple levels of triggering, and how to cross-trigger the disassembly probe from a different timebase (using an acquisition probe).

The first example uses the default setup for a simple acquisition. In the second example, you define a specific event on which you want to trigger. The third example uses two levels of triggering. In the fourth example, you cross-trigger the PM406 from an acquisition probe using a different timebase than the disassembly probe.

Example 1. A Simple Acquisition

This example uses the default 6809 setup uploaded when you connected the analyzer to a SUT and initialized the analyzer. This example shows you how to:

- acquire and disassemble data
- jump to a specific location
- re-disassemble data using the Mark Opcode function
- search for a particular event
- toggle between display modes

Follow these steps to make a simple acquisition and begin manipulating data:

1. Make sure the PM406 is connected to your SUT and the analyzer is initialized with the default disassembly setup.

2. Press START to acquire data. In the default setup, the analyzer will trigger on the first memory read that occurs after the pre-trigger memory is full. The Acquisition Process screen is displayed, telling you the status of the acquisition. When the acquisition is complete, the analyzer stops and displays the data in disassembly since that is the default data format.

3. Data is disassembled from the cursor location to the end of memory. Data above the cursor is not disassembled. Figure 15 shows an example of this.

   In order to disassemble the entire memory, you must jump to the beginning of memory and use the Mark Opcode function.

4. Press ENTER to tell the analyzer you want to enter a new location to be displayed, then enter 0000 to jump to the beginning of memory. As you finish entering the digits, the analyzer jumps to the selected memory address and displays the new information, as shown in Figure 16.

5. Press 6 to mark the first opcode fetch. This forces the PM406 to re-disassemble the data in its memory. The analyzer now displays opcode fetches from location 0000 throughout memory. See Figure 17.

6. Press 0 or 2 to cycle through available search functions and choose the trigger as the search event.

7. Press 1 to search for the trigger. Figure 18 shows the trigger event in hardware mode.

8. Press DON'T CARE to toggle to software display mode. In software display mode, only instructions are displayed. Figure 19 shows the software display which corresponds to the hardware display in Figure 18.

   When you switch disassembly modes, the analyzer goes through memory to find the opcode fetch closest to the cursor. If it can't find an opcode fetch, it returns to hardware mode.

The scroll rate, jump, and search features for disassembly displays work the same as they do in the State menu. For more information about these features, refer to the operator's manual for your particular logic analyzer.

THU, JUN 02, 1988 **Disasm: Memory 1** 08:03 6809

Loc	Addr	Data 6809	Disassembly		Operation	Status
0038	F405	27			MEM READ	DMA/BREQ
0039	F406	F9			MEM READ	DMA/BREQ
0040	FFFF	01			NOT VMA	DMA/BREQ
0041	F400	BD			MEM READ	DMA/BREQ
0042	F401	F3			MEM READ	DMA/BREQ
0043	F402	ED			MEM READ	DMA/BREQ
0044	FFFF	01			NOT VMA	DMA/BREQ
0045	F3ED	B6			MEM READ	DMA/BREQ
0046	FFFF	01			NOT VMA	DMA/BREQ
0047	CF65	03			MEM WRITE	DMA/BREQ
0048	CF64	F4			MEM WRITE	DMA/BREQ
0049	F3ED	B6	LDA	BF82	OPC FETCH	DMA/BREQ
0050	F3EE	BF			MEM READ	DMA/BREQ
0051	F3EF	82			MEM READ	DMA/BREQ
0052	FFFF	01			NOT VMA	DMA/BREQ
0053	BF82	00			MEM READ	DMA/BREQ
0054	F3F0	34	PSHS	A	OPC FETCH	DMA/BREQ
0055	F3F1	02			MEM READ	DMA/BREQ
0056	FFFF	01			NOT VMA	DMA/BREQ
0057	FFFF	01			NOT VMA	DMA/BREQ

Func:F Scroll: ▼▲ Cursor: ◀▶ Jump: ENTER

6596-15

Figure 15. Hardware disassembly display immediately after acquisition is complete. Data is disassembled starting at the cursor location.

THU, JUN 02, 1988 **Disasm: Memory 1** 08:04 6809

Loc	Addr	Data 6809	Disassembly		Operation	Status
0000	0000	00	NEG	<ED	OPC FETCH	HALT
0001	F402	ED			MEM READ	DMA/BREQ
0002	FFFF	01			NOT VMA	DMA/BREQ
0003	F3ED	B6			MEM READ	DMA/BREQ
0004	FFFF	01			NOT VMA	DMA/BREQ
0005	CF65	03			MEM WRITE	DMA/BREQ
0006	CF64	F4			MEM WRITE	DMA/BREQ
0007	F3ED	B6			MEM READ	DMA/BREQ
0008	F3EE	BF			MEM READ	DMA/BREQ
0009	F3EF	82			MEM READ	DMA/BREQ
0010	FFFF	01			NOT VMA	DMA/BREQ
0011	BF82	00			MEM READ	DMA/BREQ
0012	F3F0	34			MEM READ	DMA/BREQ
0013	F3F1	02			MEM READ	DMA/BREQ
0014	FFFF	01			NOT VMA	DMA/BREQ
0015	FFFF	01			NOT VMA	DMA/BREQ
0016	CF64	F4			MEM READ	DMA/BREQ
0017	CF63	00			MEM WRITE	DMA/BREQ
0018	F3F2	84			MEM READ	DMA/BREQ
0019	F3F3	7F			MEM READ	DMA/BREQ

Func:F Scroll: ▼▲ Cursor: ◀▶ Jump: ENTER

6596-16

Figure 16. Hardware disassembly display after jumping to location 0000. As you finish entering the digits for the location, the analyzer displays the new information starting at the top of the screen.

THU, JUN 02, 1988 **Disasm: Memory 1** 08:04 6809

Loc	Addr	Data 6809	Disassembly		Operation	Status
0000	0000	00	NEG	<ED	OPC FETCH	HALT
0001	F402	ED	STD	01,X	OPC FETCH	DMA/BREQ
0002	FFFF	01			NOT VMA	DMA/BREQ
0003	F3ED	B6			MEM READ	DMA/BREQ
0004	FFFF	01			NOT VMA	DMA/BREQ
0005	CF65	03			MEM WRITE	DMA/BREQ
0006	CF64	F4			MEM WRITE	DMA/BREQ
0007	F3ED	B6	LDA	BF82	OPC FETCH	DMA/BREQ
0008	F3EE	BF			MEM READ	DMA/BREQ
0009	F3EF	82			MEM READ	DMA/BREQ
0010	FFFF	01			NOT VMA	DMA/BREQ
0011	BF82	00			MEM READ	DMA/BREQ
0012	F3F0	34	PSHS	A	OPC FETCH	DMA/BREQ
0013	F3F1	02			MEM READ	DMA/BREQ
0014	FFFF	01			NOT VMA	DMA/BREQ
0015	FFFF	01			NOT VMA	DMA/BREQ
0016	CF64	F4			MEM READ	DMA/BREQ
0017	CF63	00			MEM WRITE	DMA/BREQ
0018	F3F2	84	ANDA	#7F	OPC FETCH	DMA/BREQ
0019	F3F3	7F			MEM READ	DMA/BREQ

Func:F Mark Opcode: 6

6596-17

Figure 17. Hardware disassembly display after marking the first opcode fetch. The Mark Opcode function re-disassembles memory from the cursor location, in this case location 0000.

THU, JUN 02, 1988 **Disasm: Memory 1** 08:05 6809

Loc	Addr	Data 6809	Disassembly		Operation	Status
1014	CF64	F4			MEM WRITE	DMA/BREQ
1015	F3ED	B6	LDA	BF82	OPC FETCH	DMA/BREQ
1016	F3EE	BF			MEM READ	DMA/BREQ
1017	F3EF	82			MEM READ	DMA/BREQ
1018	FFFF	01			NOT VMA	DMA/BREQ
1019	BF82	00			MEM READ	DMA/BREQ
1020	F3F0	34	PSHS	A	OPC FETCH	DMA/BREQ
1021	F3F1	02			MEM READ	DMA/BREQ
1022	FFFF	01			NOT VMA	DMA/BREQ
1023	FFFF	01			NOT VMA	DMA/BREQ
TRIG	CF64	F4			MEM READ	DMA/BREQ
1025	CF63	00			MEM WRITE	DMA/BREQ
1026	F3F2	84	ANDA	#7F	OPC FETCH	DMA/BREQ
1027	F3F3	7F			MEM READ	DMA/BREQ
1028	F3F4	81	CMPA	#17	OPC FETCH	DMA/BREQ
1029	F3F5	17			MEM READ	DMA/BREQ
1030	F3F6	24	BCC	F3FB	OPC FETCH	DMA/BREQ
1031	F3F7	03			MEM READ	DMA/BREQ
1032	FFFF	01			NOT VMA	DMA/BREQ
1033	F3F8	35	PULS	A	OPC FETCH	DMA/BREQ

Func:F Search For: 0,2 [Trigger] Do Search: 1

6596-18

Figure 18. Hardware disassembly display. The search event in this example is the trigger event, which occurred at memory location 1024 as specified in the Run Control menu.

THU, JUN 02, 1988 **Disasm: Memory 1** 08 05 6809

Loc	Addr	Data	6809 Disassembly		Operation
0978	F3F0	3402	PSHS	A	CF64=F400
0984	F3F2	847F	ANDA	#7F	
0986	F3F4	8117	CMPA	#17	
0988	F3F6	2403	BCC	F3FB	
0991	F3F8	3502	PULS	A	CF63=00F4
0997	F3FA	39	RTS		CF64=F403
1002	F403	8580	BITA	#80	
1004	F405	27F9	BEQ	F400	
1007	F400	BDF3ED	JSR	F3ED	CF65=03F4
1015	F3ED	B6BF82	LDA	BF82	BF82=00
1020	F3F0	3402	PSHS	A	CF64=F400
1026	F3F2	847F	ANDA	#7F	
1028	F3F4	8117	CMPA	#17	
1030	F3F6	2403	BCC	F3FB	
1033	F3F8	3502	PULS	A	CF63=00F4
1039	F3FA	39	RTS		CF64=F403
1044	F403	8580	BITA	#80	
1046	F405	27F9	BEQ	F400	
1049	F400	BDF3ED	JSR	F3ED	CF65=03F4
1057	F3ED	B6BF82	LDA	BF82	BF82=00

Func:F Scroll Rate: 7,8 [20] Mode: X [Software]

6596-19

Figure 19. Corresponding software disassembly. Because the trigger was not an opcode fetch, this software disassembly display (corresponding to the hardware display shown in Figure 18) does not show the trigger event at the cursor. The Operation column shows the address (left side of equals sign) and data information for the instructions that occurred.

Example 2: Trigger on a Specific Event

This example shows you how to trigger on an event which you specify. In this example, a program is yielding unexpected results. The program contains a BEQ, branch on equal, instruction and you suspect the program is branching incorrectly. The BEQ has a data value of 27_{hex}. You want to see where this command makes the program go. You need to define the BEQ as the trigger and then move the trigger towards the beginning of the memory, so that there is enough memory space for the results you want to see.

This example uses the default setup except for defined conditions and trigger specification and the location of the trigger in memory. You don't need to change the timebase or the channel grouping from the default settings. Follow these steps to trigger on the specific event.

1. In the Conditions menu, rename the Q condition word to BRANCH and define it as DAT 27. Figure 20 shows the new BRANCH condition word definition: XXXX 27 XXX XXX XX.
2. In the Trigger Spec menu, define the condition in the level 1 trigger statement to BRANCH. Figure 20 shows this. When the analyzer finds 27_{hex} on the data lines, it triggers and fills the memory with data.
3. In the Run Control menu, move the trigger position to 128 as shown in Figure 21. This position provides a large portion of memory to store the data occurring after the branch, thereby allowing you to see what happens to your program after executing the BEQ command.
4. Start the acquisition by pressing START. The analyzer will display data in hardware mode.
5. In order to have the maximum amount of disassembled data, jump to location 0000 and mark the first opcode fetch.
6. Use the search function to find the trigger event. Figure 22 shows the cursor on the trigger event, BEQ, in hardware display mode. This is the beginning of the branch you were looking for.
7. Press X to toggle to software display mode. Figure 23 shows the trigger event again. You can use the displayed information to see where the branch has led the program.

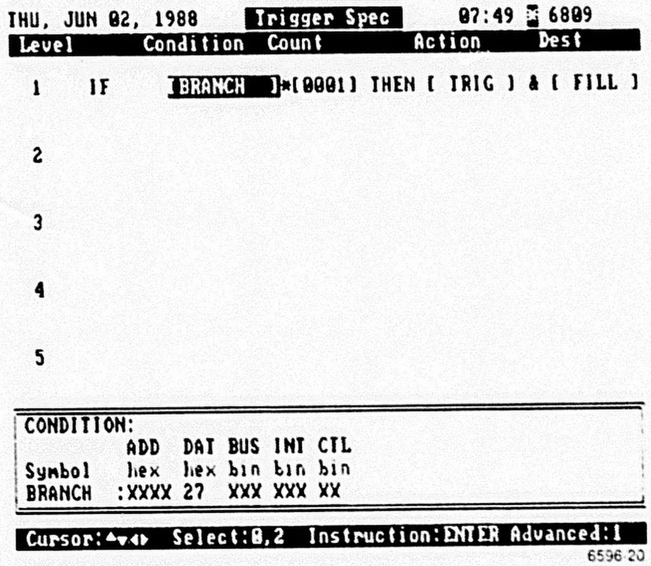


Figure 20. Conditions and Trigger Spec menus. BRANCH is defined as DAT 27hex and as the trigger condition.

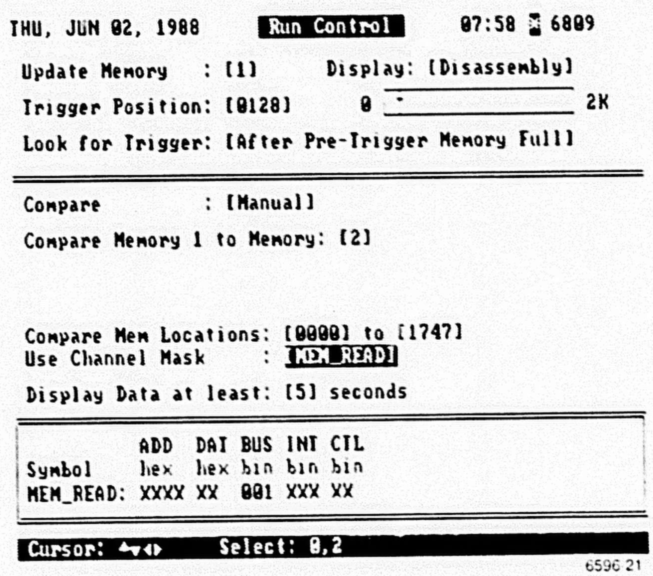


Figure 21. Run Control menu for example 2. The trigger position has been changed to 128, leaving most of memory to store the data that occurs after the trigger event.

```
THU, JUN 02, 1988   Disasm:  Memory 1    08:00 ▒ 6809
 Loc    Addr Data 6809   Disassembly  Operation  Status

 0118   FFFF 01                       NOT UMA     DMA/BREQ
 0119   CF63 00                       MEM READ    DMA/BREQ
 0120   CF64 F4                       MEM READ    DMA/BREQ
 0121   F3FA 39     RTS               OPC FETCH   DMA/BREQ
 0122   F3FB 86                       MEM READ    DMA/BREQ
 0123   CF64 F4                       MEM READ    DMA/BREQ
 0124   CF65 03                       MEM READ    DMA/BREQ
 0125   FFFF 01                       NOT UMA     DMA/BREQ
 0126   F403 85     BITA     #80      OPC FETCH   DMA/BREQ
 0127   F404 80                       MEM READ    DMA/BREQ
-TRIG---F405-27-----BEQ------F400-----OPC FETCH---DMA/BREQ
 0129   F406 F9                       MEM READ    DMA/BREQ
 0130   FFFF 01                       NOT UMA     DMA/BREQ
 0131   F400 BD     JSR      F3ED     OPC FETCH   DMA/BREQ
 0132   F401 F3                       MEM READ    DMA/BREQ
 0133   F402 ED                       MEM READ    DMA/BREQ
 0134   FFFF 01                       NOT UMA     DMA/BREQ
 0135   F3ED B6                       MEM READ    DMA/BREQ
 0136   FFFF 01                       NOT UMA     DMA/BREQ
 0137   CF65 03                       MEM WRITE   DMA/BREQ

 Func:F     Search For: 0,2 [Trigger ]     Do Search: 1
```

6596-22

Figure 22. Trigger event in hardware display mode. The cursor marks the trigger event, BEQ.

```
THU, JUN 02, 1988   Disasm:  Memory 1    08:01 ▒ 6809
 Loc    Addr Data        6809 Disassembly     Operation

 0086   F405 27F9        BEQ     F400
 0089   F400 BDF3ED      JSR     F3ED         CF65=03F4
 0097   F3ED B6BF82      LDA     BF82         BF82=00
 0102   F3F0 3402        PSHS    A            CF64=F400
 0108   F3F2 847F        ANDA    #7F
 0110   F3F4 8117        CMPA    #17
 0112   F3F6 2403        BCC     F3FB
 0115   F3F8 3502        PULS    A            CF63=00F4
 0121   F3FA 39          RTS                  CF64=F403
 0126   F403 8580        BITA    #80
-TRIG---F405-27F9--------BEQ-----F400---------------------
 0131   F400 BDF3ED      JSR     F3ED         CF65=03F4
 0139   F3ED B6BF82      LDA     BF82         BF82=00
 0144   F3F0 3402        PSHS    A            CF64=F400
 0150   F3F2 847F        ANDA    #7F
 0152   F3F4 8117        CMPA    #17
 0154   F3F6 2403        BCC     F3FB
 0157   F3F8 3502        PULS    A            CF63=00F4
 0163   F3FA 39          RTS                  CF64=F403
 0168   F403 8580        BITA    #80

 Func:F     Scroll Rate: 7,8 [20]     Mode: X [Software]
```

6596-23

Figure 23. Trigger event in software display mode. The cursor marks the trigger event, BEQ. This figure corresponds to the hardware display mode of Figure 22.

Example 3: Triggering with Two Levels

You can work through this example only if you have a 1230 logic analyzer. If you have a 1220 or 1225 analyzer, you should go on to example 4.

This example shows you how to acquire a block of specific data. In this example, a particular subroutine occurs a number of times in the program. You want to acquire only the subroutine. You need to define two conditions: the beginning of the subroutine and the end of the subroutine. The beginning of the subroutine is JSR (jump to new location saving return address) and has a data value of BD_{hex}. The end of the subroutine is RTS (return from subroutine) and has a data value of 39_{hex}.

In order to show an example of a multi-level acquisition, you'll have the analyzer acquire the subroutine, then loop back and acquire it again until we stop the acquisition manually.

This example uses the default setup except for defined conditions and trigger statements. You don't need to change the timebase or channel grouping control information from the default 32-channel setup for this example. Follow these steps to trace a subroutine and trigger on the subroutine:

1. In the Conditions menu, rename the Q condition word to JUMP and define it as DAT BD_{hex}. Figure 24 shows the new JUMP condition word definition: XXXX BD XX XXX.
2. Rename the R condition word to RETURN and define it as DAT 39_{hex}. Figure 24 shows the new RETURN condition word definition: XXXX 39 XX XXX.
3. In the Trigger Spec menu, define four levels of if-then trigger statements as shown in Figure 25. The analyzer starts storing information immediately, so you want to turn off storage at level 1. Level 2 indicates that when the analyzer finds BD_{hex} on the data lines (the beginning of the subroutine) it starts to store data, then moves on to level 3.

   Because the RTS (return from subroutine) is actually five hex data numbers long, you can't just turn storage off when the analyzer encounters a 39_{hex}. If you did this, the

PM406 would not have enough information to disassemble the command. So level 3 indicates that when the analyzer finds 39hex on the data lines, the analyzer stores that information and goes on to level 4. Level 4 indicates that after four more hex numbers occur (the end of the RTS command), the analyzer stops storing data, then loops back up to level 2 again to look for BDhex.

4. In the Run Control menu, the data display should default to Disassembly. Because you will halt the acquisition manually, the stop point will be the trigger position. Change the Trigger Position to 1920 so that the stop point is near the end of memory allowing for more storage of data. Figure 26 shows the Run Control menu for this example.

5. Press START. At the first occurance of the subroutine, the analyzer starts storing the information. When the return is encountered, the analyzer waits until the end of the RTS command, then stops storage and loops back up to level 2 to search for the next occurance of JUMP.

6. After a few seconds, press STOP. The analyzer stops acquiring data and displays the acquisition in disassembly format.

7. Jump to location 0000 and mark the first opcode fetch. Figure 27 shows the hardware mode display for this acquisition.

8. Press X to toggle to software display mode. You can see in Figure 28 that the software display stops listing instructions after location 1919.

9. Toggle back to hardware display mode and search for the trigger. Figure 29 shows that there were no valid opcode fetches after the trigger event. This happens when you stop the acquisition by pressing STOP.

```
FRI, JUN 03, 1988   Trigger Spec      09:35   6809
Level     Condition  Count       Action      Dest

1    IF     [S          ]*[0001] THEN [STROFF] & [GOTO 2]

2    IF     [JUMP       ]*[0001] THEN [STR ON] & [GOTO 3]

3    IF     [RETURN     ]*[0001] THEN [ NOP  ] & [GOTO 4]
                     CONDITIONS
Symbol      ADD  DAT BUS INT CTL
            hex  hex bin bin bin
JUMP      : XXXX BD  XXX XXX XX
RETURN    : XXXX 39  XXX XXX XX
S         : XXXX XX  XXX XXX XX

  ▲     Edit Symbol: ENTER
 ◀ ▶    Window Up  : F
  ▼     Window Down: C

Menu:MENU   Return:MENU twice    New:MENU, then Hex Key
```

6596-24

Figure 24. Conditions setup. The two new conditions define the beginning and end of the subroutine you're tracing.

```
FRI, JUN 03, 1988   Trigger Spec      09 34   6809
Level     Condition  Count       Action      Dest

1    IF     [S          ]*[0001] THEN [STROFF] & [GOTO 2]

2    IF     [JUMP       ]*[0001] THEN [STR ON] & [GOTO 3]

3    IF     [RETURN     ]*[0001] THEN [ NOP  ] & [GOTO 4]

4    IF     [S          ]*[0004] THEN [STROFF] & [GOTO 2]

5

CONDITION:
          ADD  DAT BUS INT CTL
Symbol    hex  hex bin bin bin
S        :XXXX XX  XXX XXX XX

Cursor:▲▼◀▶  Select:0,2  Instruction:ENTER Advanced:1
```

6596-25

Figure 25. Trigger Spec setup. The four levels of trigger statements tell the analyzer to store everything between the beginning and end of the subroutine, and then loop back up to level 2, search for the beginning of the next occurance of the same subroutine, and store it again.

```
THU, JUN 02, 1988        Run Control       08:14   6809

 Update Memory    : [1]       Display: [Disassembly]
 Trigger Position: [1920]     0 ________________ 2K
 Look for Trigger: [After Pre-Trigger Memory Full]

 Compare          : [Manual]
 Compare Memory 1 to Memory: [2]


 Compare Mem Locations: [0000] to [1747]
 Use Channel Mask     : [MEM_READ]
 Display Data at least: [5] seconds

            ADD  DAT BUS INT CTL
 Symbol     hex  hex bin bin bin
 MEM_READ: XXXX XX  001 XXX XX

 Cursor: ▲▼◄►       Select: 0,2
```

6596-26

Figure 26. Run Control menu. The default display is set to Disassembly and the trigger position is changed to 1920.

```
FRI, JUN 03, 1988  Disasm: Memory 1   09 51    6809
 Loc   Addr Data 6809   Disassembly  Operation  Status
 2040  0000 00                       MEM WRITE  HALT
 2041  0000 00                       MEM WRITE  HALT
 2042  0000 00                       MEM WRITE  HALT
 2043  0000 00                       MEM WRITE  HALT
 2044  0000 00                       MEM WRITE  HALT
 2045  0000 00                       MEM WRITE  HALT
 2046  0000 00                       MEM WRITE  HALT
 2047  0000 00                       MEM WRITE  HALT

 0000  EF64 01   ???                 OPC FETCH  DMA/BREQ
-0001--F3ED-BD---JSR-----F3ED--------OPC FETCH--DMA/BREQ
 0002  F3EE F3   ADDD    ED01        OPC FETCH  DMA/BREQ
 0003  F3EF ED   STD     01,X        OPC FETCH  DMA/BREQ
 0004  FFFF 01                       NOT VMA    DMA/BREQ
 0005  BF82 B6                       MEM READ   DMA/BREQ
 0006  F3F0 01                       MEM READ   DMA/BREQ
 0007  F3F1 03                       MEM WRITE  DMA/BREQ
 0008  FFFF F4                       MEM WRITE  DMA/BREQ
 0009  FFFF B6                       NOT VMA    DMA/BREQ
 0010  EF64 BF   STX     8201        OPC FETCH  DMA/BREQ
 Func:F    Scroll Rate: 7,8 [20]    Mode: X [Hardware]
```

6596-27

Figure 27. Hardware display. Location 0000 shows the end of an instruction. The analyzer displays question marks (???) because it doesn't have enough information to disassemble this instruction.

```
FRI, JUN 03, 1988  Disasm:  Memory 1   09 51   6809
Loc   Addr Data        6809 Disassembly    Operation

1896  F3ED B6BF82      LDA     BF82        BF82=00
1901  F3F0 3402        PSHS    A           EF64=F400
1907  F3F2 847F        ANDA    #7F
1909  F3F4 8117        CMPA    #17
1911  F3F6 2403        BCC     F3FB
1914  F3F8 3502        PULS    A
1918  EF63 00F4        NEG     <F4
1919  EF64 F40000      ANDB    0000        0000=0000

0000  EF64 01          ???
-0001-F3ED-BDF3ED------JSR-----F3ED-----------------
0003  F3EF ED01        STD     01,X        F3F0=0103
0010  EF64 BF8201      STX     8201        F3F8=00
0020  F3F9 847F        ANDA    #7F
0023  EF63 172403      LBSR    1369
0027  EF64 3502        PULS    A           F400=0101
0031  F402 00F4        NEG     <F4         EF64=0301
0038  F3EE BDF3ED      JSR     F3ED        F3F0=B601
0046  EF64 B6BF82      LDA     BF82        F3F4=00
0051  F3F5 3402        PSHS    A
Func:F     Scroll Rate: 7,8 [20]    Mode: X [Software]
```

6596-28

Figure 28. Software display. You can see the subroutine sequence in a more compact form in software mode since only instructions are displayed.

```
FRI, JUN 03, 1988  Disasm:  Memory 1   09 52   6809
Loc   Addr Data 6809   Disassembly  Operation  Status

1910  F3F5 17                       MEM READ   DMA/BREQ
1911  F3F6 24   BCC    F3FB         OPC FETCH  DMA/BREQ
1912  F3F7 03                       MEM READ   DMA/BREQ
1913  FFFF 01                       NOT VMA    DMA/BREQ
1914  F3F8 35   PULS   A            OPC FETCH  DMA/BREQ
1915  F3F9 02                       MEM READ   DMA/BREQ
1916  FFFF 01                       NOT VMA    DMA/BREQ
1917  FFFF 01                       NOT VMA    DMA/BREQ
1918  EF63 00   NEG    <F4          OPC FETCH  DMA/BREQ
1919  EF64 F4   ANDB   0000         OPC FETCH  DMA/BREQ
-TRIG-0000-00-----------------------MEM WRITE--HALT-----
1921  0000 00                       MEM WRITE  HALT
1922  0000 00                       MEM WRITE  HALT
1923  0000 00                       MEM WRITE  HALT
1924  0000 00                       MEM WRITE  HALT
1925  0000 00                       MEM WRITE  HALT
1926  0000 00                       MEM WRITE  HALT
1927  0000 00                       MEM WRITE  HALT
1928  0000 00                       MEM WRITE  HALT
1929  0000 00                       MEM WRITE  HALT
Func:F    Search For: 0,2 [Trigger ]     Do Search: 1
```

6596-29

Figure 29. Hardware display at trigger event. Notice that there are no more opcode fetch instructions after the trigger. This is a result of using STOP to end the acquisition.

Example 4: Cross-Triggering

If you're using a 1225 or 1230 Logic Analzyer, you can acquire data on a 16-channel acquisition probe at the same time you use the PM406. You can also set the PM406 to trigger off the timebase of the acquisition probe, or vice versa. This example shows you how to set up the PM406 to trigger off the acquisition probe.

Configuration. This example uses a 1225/1230 with 48 channels. The PM406 is still plugged into probe slots A and B. The 16-channel acquisition probe (P6443 or P6444) is plugged into probe slot C.

What This Example Shows. This example shows how to set up an acquisition probe to trigger on a condition, then set up the disassembly probe to automatically cross-trigger and show the acquired information in disassembly display. In this example, you want to know how the code is executed when you trigger the acquisition probe on a particular event. The analyzer then automatically cross-triggers the disassembly probe so that you can display the disassembly data for that acquisition.

Figures 29 through 34 show the setup menus for this example. The menus show how to set up the 1225/1230 with these parameters:

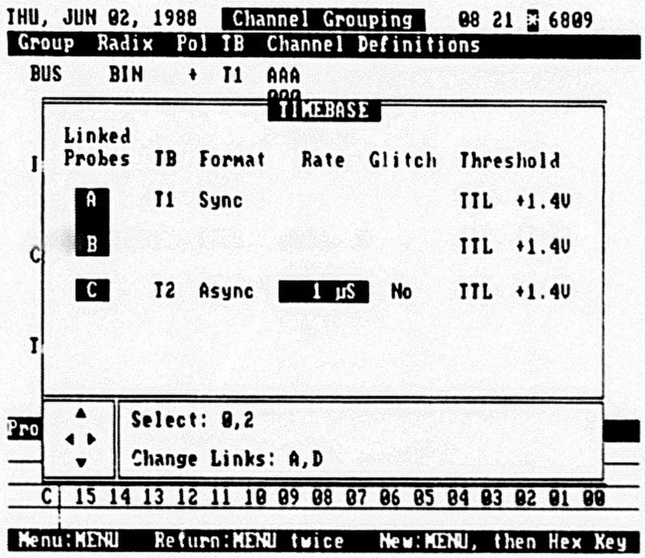
- Probes A and B are in T1; probe C is in T2.
- Channel group GPF is renamed to TST and contains 16 channels from probe C.
- The trigger condition GET is defined for the specific event upon which you wish to trigger.
- The trigger timebase is T2 (the acquisition probe) so that the 1225/1230 recognizes the trigger condition GET and automatically cross-triggers the disassembly probe when GET occurs.

The Steps for Cross-Triggering. Follow these steps to cross-trigger the PM406 off the acquisition probe and search for the trigger event in the resulting disassembly display:

1. In the Timebase menu, probes A and B are linked by default in timebase T1 (separately from probe C, which should be in T2). Change the rate of timebase T2 to 1 μs. Refer to Figure 30.

2. In the Channel Grouping menu, scroll to channel group GPE and change the channel group name to TST. Add channels C15-C00 to this new group. Refer to Figure 31.
3. In the Conditions menu, define a condition GET to the value $D4F1_{hex}$ in group TST. Figure 32 shows the Trigger Spec menu and the value of the trigger condition GET.
4. In the Trigger Spec menu, set the trigger condition to GET. Figure 32 shows the Trigger Spec menu.
5. Look at the menu bar at the bottom of the Trigger Spec menu, and press D to toggle the trigger timebase to T2. Refer to Figure 32.
6. In the Run Control menu, make sure the 1225/1230 looks for the trigger GET after the pretrigger memory is full. The default data display format should still be set to Disassembly.
7. Press START. When the analyzer recognizes the trigger condition, the analyzer triggers all modules, fills memory, and stops. The disassembly screen is displayed.
8. Press 0 or 2 to cycle through available search events until you select Trigger, then press 1 to locate the trigger. Figure 33 shows the trigger event in a hardware disassembly display. The corresponding software mode display is in Figure 34.
9. In order to view the data from the acquisition probe, you must display data from a different timebase (T2). Go to the State display and you'll see the disassembly information in state format. Press F until you see the Timebase: field at the bottom of the screen. Press 9 to change to Timebase T2. The State display will now show the data that was acquired on the acquisition probe. Figure 35 shows the state display for channel TST on probe C.

Once you've made the acquisition, you can call up state, disassembly, and timing displays for the acquired data. Since you used two timebases to make the acquisition, you must change pages to display what happened in T2 on the acquisition probe, and then what happened in T1 on the disassembly probe. To change the timebase in the State menu, use the 9. To change the timebase in the Timing menu, use the 0 and 1.



6596-30

Figure 30. Timebase for cross-trigger. Probes A and B (the PM406) are linked in T1, and probe C (the acquisition probe) is in T2. This lets you acquire data with different timebases.

FRI, JUN 03, 1988 **Channel Grouping** 09 35 6809

Group	Radix	Pol	TB	Channel Definitions
ADD	HEX	+	T1	BBBBBBBBBBBBBBBB 1111110000000000 5432109876543210
DAT	HEX	+	T1	AAAAAAAA 11111100 54321098
BUS	BIN	+	T1	AAA 000 210
INT	BIN	+	T1	AAA 000 543
CTL	BIN	+	T1	AA 00 76
TST	HEX	+	T2	CCCCCCCCCCCCCCCC 1111110000000000 5432109876543210

6596-31

Figure 31. Channel Grouping for cross-trigger. The analyzer screen shows only four channel groups at a time. This figure is a modification of two combined screens so you can see all five channel group definitions. The channel group shows that the fifth channel group is renamed to TST and contains 16 channels for acquisition probe C (timebase T2).

```
THU, JUN 02, 1988     Trigger Spec      08 29   6809
Level        Condition  Count          Action      Dest

  1    IF     [GET      ]*[0001] THEN [ TRIG ] & [ FILL ]

  2

  3

  4

  5

CONDITION:
          ADD   DAT BUS INT CTL TST
Symbol    hex   hex bin bin bin hex
GET      :XXXX  XX  XXX XXX XX  D4F1

Cur:▲▼◄► Sel:0,2 Instr:ENTER TrigTB:D[T2] Advanced:1
```

6596-32

Figure 32. Conditions and Trigger Spec for cross-trigger. The trigger condition GET is defined as D4F1hex. The menu bar at the bottom of the Trigger Spec screen shows that the trigger timebase is T2.

```
THU, JUN 02, 1988  Disasm:  Memory 1   08:31   6809
Loc   Addr Data 6809  Disassembly   Operation  Status
1014  F3F4 81    CMPA   #17          OPC FETCH  DMA/BREQ
1015  F3F5 17                        MEM READ   DMA/BREQ
1016  F3F6 24    BCC    F3FB         OPC FETCH  DMA/BREQ
1017  F3F7 03                        MEM READ   DMA/BREQ
1018  FFFF 01                        NOT VMA    DMA/BREQ
1019  F3F8 35    PULS   A            OPC FETCH  DMA/BREQ
1020  F3F9 02                        MEM READ   DMA/BREQ
1021  FFFF 01                        NOT VMA    DMA/BREQ
1022  FFFF 01                        NOT VMA    DMA/BREQ
1023  CF63 00                        MEM READ   DMA/BREQ
-TRIG-CF64-F4------------------------MEM READ---DMA/BREQ
1025  F3FA 39    RTS                 OPC FETCH  DMA/BREQ
1026  F3FB 86                        MEM READ   DMA/BREQ
1027  CF64 F4                        MEM READ   DMA/BREQ
1028  CF65 03                        MEM READ   DMA/BREQ
1029  FFFF 01                        NOT VMA    DMA/BREQ
1030  F403 85    BITA   #80          OPC FETCH  DMA/BREQ
1031  F404 80                        MEM READ   DMA/BREQ
1032  F405 27    BEQ    F400         OPC FETCH  DMA/BREQ
1033  F406 F9                        MEM READ   DMA/BREQ
Func:F     Search For: 0,2 [Trigger ]     Do Search: 1
```

6596-33

Figure 33. Hardware disassembly display. After searching for the trigger event, the cursor is on TRIG. TRIG marks the event that occurred in timebase T1 when trigger event GET occurred in timebase T2.

```
THU, JUN 02, 1988   Disasm:  Memory 1   08:31 ▒ 6809
Loc    Addr Data             6809 Disassembly      Operation
 0977  F3F8 3502             PULS     A            CF63=00F4
 0983  F3FA 39               RTS                   CF64=F403
 0988  F403 8580             BITA     #80
 0990  F405 27F9             BEQ      F400
 0993  F400 BDF3ED           JSR      F3ED         CF65=03F4
 1001  F3ED B6BF82           LDA      BF82         BF82=00
 1006  F3F0 3402             PSHS     A            CF64=F400
 1012  F3F2 847F             ANDA     #7F
 1014  F3F4 8117             CMPA     #17
 1016  F3F6 2403             BCC      F3FB
-1019--F3F8-3502-------------PULS-----A------------CF63=00F4--
 1025  F3FA 39               RTS                   CF64=F403
 1030  F403 8580             BITA     #80
 1032  F405 27F9             BEQ      F400
 1035  F400 BDF3ED           JSR      F3ED         CF65=03F4
 1043  F3ED B6BF82           LDA      BF82         BF82=00
 1048  F3F0 3402             PSHS     A            CF64=F400
 1054  F3F2 847F             ANDA     #7F
 1056  F3F4 8117             CMPA     #17
 1058  F3F6 2403             BCC      F3FB
 Func:F     Scroll Rate: 7,8 [20]     Mode: X [Software]
```

6596-34

Figure 34. Software disassembly display. This software display corresponds to the hardware display shown in Figure 34. Note that the trigger event was in the middle of an instruction, so the cursor in the display is near the trigger event, not on it.

```
THU, JUN 02, 1988   State: Memory 1   08 32 ▒ 6809
 Loc   TST
       hex
 1014  3B7F
 1015  397F
 1016  397F
 1017  397F
 1018  397F
 1019  D4F0
 1020  D4F0
 1021  D4F0
 1022  D4F0
 1023  D4F0
-TRIG--D4F1-------------------------------------------
 1025  D4F1
 1026  D4F1
 1027  D4F1
 1028  D4F2
 1029  D4F2
 1030  D4F2
 1031  D4F2
 1032  D4F2
 1033  7CF3
 Func:F Scroll Rate:7,8[20] Timebase:9[T2 Async   1 µS]
```

6596-35

Figure 35. State display for timebase T2. Condition GET (D4F1hex) is the trigger event.