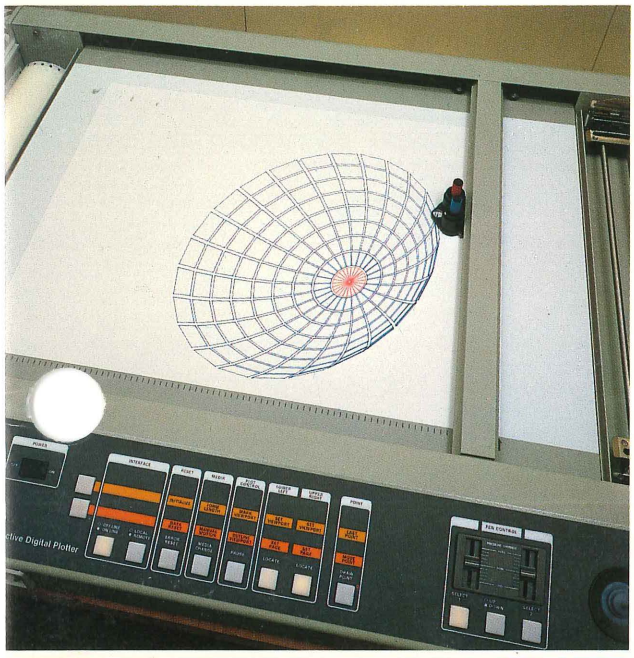
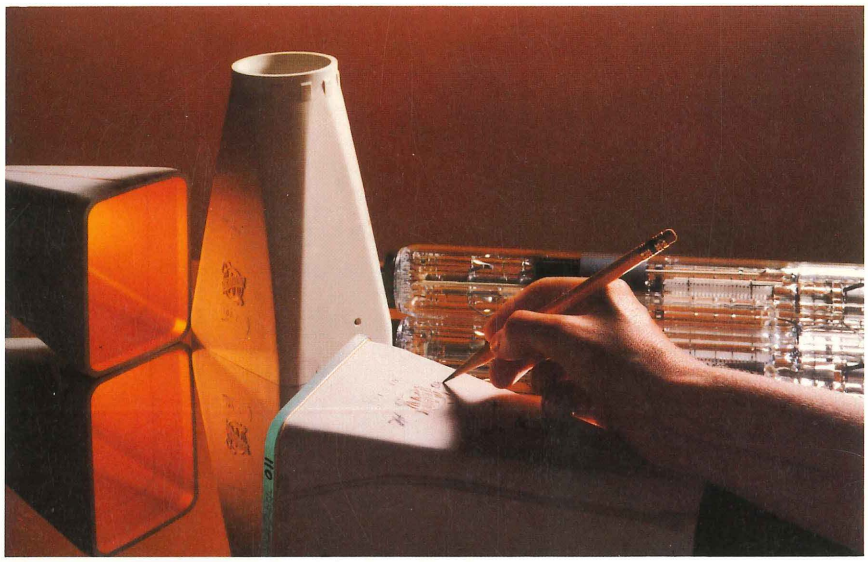
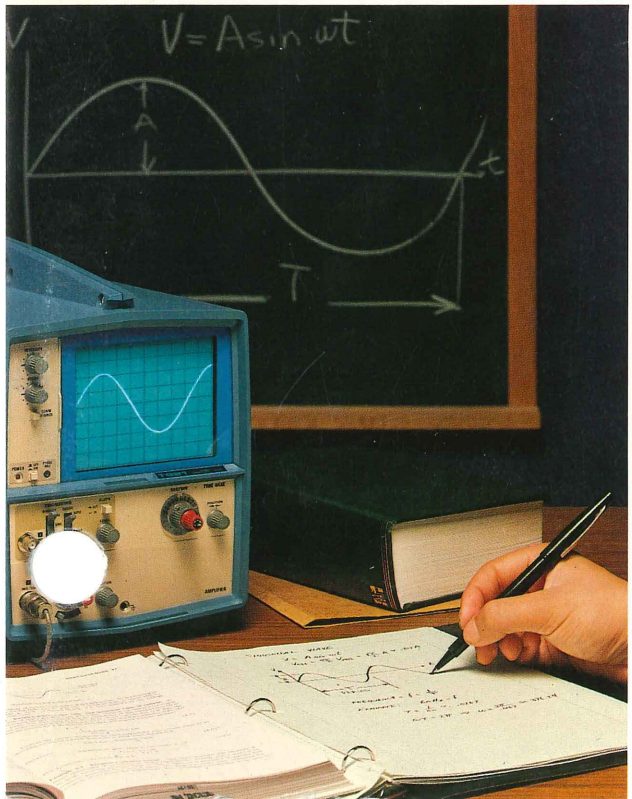
Logic Analyzer Concepts







Tektronix®
COMMITTED TO EXCELLENCE

Training Information

Tektronix training programs are developed to assure customers maximum return on their investments. Several types of training programs meet that objective: formal classroom training; audiotapes on operation, circuit description, and calibration; videotapes on basic concepts, operation, and applications; and stand-alone multi-media training packages which combine printed materials with audiotapes and/or videotapes in a format designed for independent learning.

Ordering Information

You may order additional copies of this booklet through your nearest Tektronix Field Office, listed inside the back cover. For more information regarding training classes, audiotapes, videotapes, and training packages, request a copy of the current Customer Training Catalog from your local Tektronix Field Representative.

Copyright © 1980. Tektronix, Inc. All rights reserved. Printed in U.S.A. Tektronix products are covered by U.S. and foreign patents, issued and pending. Information in this publication supersedes that in all previously published material. Specification and price change privileges reserved. TEKTRONIX, TEK, SCOPE-MOBILE, and ® are registered trademarks of Tektronix, Inc. TEL-EQUIPMENT is a registered trademark of Tektronix U.K. Limited. For further information, contact Tektronix, Inc., P.O. Box 500, Beaverton, OR 97077. Phone 503-644-0161. TWX 910-467-8708. Cable: Tektronix. Subsidiaries and distributors worldwide.

062-5429-00

Marketing Training
12/80

Logic Analyzer Concepts

Tektronix®
COMMITTED TO EXCELLENCE

Preface

The intent of this booklet and the two associated videotapes is to cover, in a clear and concise manner, the terms and concepts relevant to general purpose logic analyzers, serial logic analyzers, data analyzers, microprocessor analyzers, and signature analyzers. Each topic is followed by a quiz, with an answer key, to give immediate feedback. Concepts covered include: comparisons of logic analyzers vs. oscilloscopes, logic analyzers vs. microprocessor development labs/aids, different types of logic analyzers, data acquisition, word recognition, synchronous, asynchronous, types of formatted displays, digital delay, glitch latch and glitch memory. The videotapes enhance the communication of this information; however, the training manual may be used alone.

For maximum, effectiveness, work through this training package in the following order:

1. View the videotape General Purpose Logic Analyzer Concepts (068-0118-00).
2. Read chapters 1-4 in this booklet, taking the quizzes as encountered.
3. View the videotape Advanced Logic Analyzer Concepts (068-0121-00).
4. Read chapters 5-7 in this booklet, taking the quizzes as encountered.

Illustrations

Page

Figure 2-1. Trigger after data, or pre-trigger. 7
Figure 2-2. Centered 7
Figure 2-3. Trigger before data, or post-trigger. 7
Figure 2-4. Binary display 8
Figure 2-5. Octal display. 8
Figure 2-6. Hex display. 9
Figure 2-7. Binary. 9
Figure 2-8. 16-bit word divided into groups of 3. 9
Figure 2-9. 16-bit word divided into two 8-bit bytes, then divided into groups of 3. 9
Figure 2-10. Microprocessor mnemonic display combines with 8 general purpose lines. 9
Figure 2-11. Microprocessor mnemonic display. 9
Figure 2-12. ASCII display. 9
Figure 2-13. GPIB display. 10
Figure 2-14. Map display. 10
Figure 2-15. Signature display. 11
Figure 3-1. General purpose logic analyzer block diagram. 13
Figure 3-2. 8080A timing waveforms. 14
Figure 3-3. Clock qualifier. 14
Figure 3-4. First order latch on narrow pulse. 15
Figure 3-5. First order latch misses glitch next to actual data. 15
Figure 4-1. Asynchronous word recognition. 17
Figure 4-2. Synchronous word recognition. 18
Figure 4-3. Word recognition with additional word qualifier input. 18
Figure 4-4. Asynchronous filter action for word recognition. 19
Figure 5-1. Serial asynchronous data. 23
Figure 5-2. Serial synchronous data. 24

Contents

Objectives vii
General Purpose Logic Analyzer Concepts Videotape Quiz 1
Chapter 1: Overview of Logic Analyzers **3**
Logic Analyzers vs. Oscilloscopes 3
Logic Analyzers vs. MDL/MDA 3
Types of Logic Analyzers 4
Chapter 1 Quiz 5
Chapter 2: Types of Displays **7**
Timing Diagram 7
State Table 8
Mnemonic 8
GPIB 10
Map 10
Signature 11
Chapter 2 Quiz 12
Chapter 3: Data Acquisition **13**
Comparator 13
Probe Pod 13
Synchronous Data 13
Asynchronous Data Acquisition 13
Clock Qualifier 13
Digital Latch 15
Chapter 3 Quiz 16
Chapter 4: Triggering 17
Triggering 17
Word Recognition 17
Synchronous Word Recognition 17
Asynchronous Word Recognition 17
Word Qualifier 18
Word Recognition Filter 18
Range Recognition 19
Digital Delay 19
Chapter 4 Quiz 20
Advanced Logic Analyzer Concepts Videotape Quiz 21
Chapter 5: Serial Logic Analyzers **23**
Serial Data Acquisition and Word Recognition 23
Protocols 24
Chapter 5 Quiz 25

Chapter 6: Microprocessor Analyzers **27**
Application 27
General Purpose Channels 27
Mnemonics 27
General Purpose Module 27
Triggering 27
Execution Time 27
Chapter 6 Quiz 28
Chapter 7: Signature Analyzers **29**
Implementation 29
Documentation 29
Transition Counting vs. Signature Analysis 29
Chapter 7 Quiz 30
Appendix A: Glossary 31
Appendix B: Basic Analyzer Applications 37
Appendix C: Comparison of Features 38
Quiz Answer Key 39

Objectives

After completing this training package, the student will be able to:

1. State two advantages of logic analyzers over oscilloscopes.
2. State two advantages of oscilloscopes over logic analyzers.
3. List five types of digital analyzers associated with logic analyzers.
4. Differentiate between logic analyzer instrumentation and microprocessor development instrumentation.
5. Define pre-trigger, center trigger and post-trigger.
6. Describe a timing diagram display and give a typical use.
7. Describe a state table display and give a typical use.
8. Differentiate between octal and split octal.
9. Describe an ASCII display and give a typical use.
10. Describe a GPIB display and give a typical use.
11. Describe a map display and give a typical use.
12. Describe a mnemonic display and give a typical use.
13. Define data acquisition.
14. Define external clock and describe its use in data acquisition.
15. Define synchronous data acquisition.
16. Define asynchronous data acquisition.
17. Define digital latch.
18. Define clock qualification.
19. State the main benefit of a clock qualifier.
20. Define word recognition.
21. Define synchronous word recognition.
22. Define word qualification.
23. Define asynchronous word recognition.
24. State the function of a word recognizer filter.
25. Contrast serial logic analyzers with parallel logic analyzers.
26. Define serial word recognition.
27. Differentiate between data acquisition modes for asynchronous, isochronous, and synchronous protocol.
28. Define signature analysis.
29. State the use of signature analysis.
30. State the benefit of signature analysis over transition counting.
31. State the two limitations of signature analysis.
32. Differentiate between microprocessor analyzers and general purpose analyzers.
33. Describe the special triggering capabilities found in microprocessor analyzers.

General Purpose Logic Analyzer Concepts Videotape Quiz

First view the videotape entitled *General Purpose Logic Analyzers*. Then answer the questions below and check your responses against the answer key at the back of this book.

1. Logic analyzers have the following advantages over non-digital oscilloscopes:
   a. More channels, amplitude vs. time, storage.
   b. More channels, various formatted displays, amplitude vs. time.
   c. More channels, various formatted displays, data before trigger.
   d. More channels, data before trigger, data after trigger.

2. Oscilloscopes have the following advantages over logic analyzers:
   a. Show actual waveform, 8 channels, store fast signals.
   b. Show actual waveform, store fast signals, finer time resolution.
   c. Show actual waveform, 8 channels, simultanous acquisition.
   d. Show actual waveform, store fast signals, simultanous acquisition.

3. A microprocessor analyzer:
   a. Usually will display mnemonics.
   b. Usually will display GPIB.
   c. Is a logic analyzer with an internal microprocessor.
   d. All of the above.

4. A logic state analyzer:
   a. Is a serial analyzer.
   b. Is a microprocessor analyzer.
   c. Is a data analyzer.
   d. Does not use an internal clock to clock in data.

5. Data pretrigger:
   a. Allows the user to see the data before the trigger.
   b. Is used with a timing diagram display only.
   c. Is used with a state table only.
   d. Is used with a GPIB display only.

6. With synchronous data acquisition:
   a. The logic analyzer internal clock is synchronized with the data.
   b. The logic analyzer uses the external clock of the system under test.
   c. Data is often sampled five times faster than the system clock.
   d. None of the above.

7. With asynchronous word recognition:
   a. The logic analyzer internal clock is used.
   b. The logic analyzer uses the external clock of the system under test.
   c. No clock is used.
   d. A clock qualifier is used.

8. A clock qualifier:
   a. Is used to select which line the incoming clock will be on.
   b. Allows selective acquisition of data.
   c. May be used with synchronous or asynchronous data acquisition.
   d. May be used with asynchronous data acquisition only.

9. A word recognizer is like:
   a. A big OR gate.
   b. A big AND gate.
   c. A digital delay.
   d. Pre-trigger

Overview of Logic Analyzers

CHAPTER 1

The various types of logic analyzer functions offer unique capabilities not found in other instruments. This chapter compares logic analyzers to oscilloscopes and microprocessor development equipment. Different generic types of logic/digital/data analyzers are introduced. More than one type or function may be found in a given instrument. A glossary of appropriate terms can be found in Appendix A. Basic logic analyzer applications vs. number of channels and bits per channel can be found in Appendix B. A comparison of logic analyzers and analog oscilloscopes is summarized in Appendix C.

Logic Analyzers vs. Oscilloscopes

Today, the largest number of channels that can be displayed on an oscilloscope screen is 8. For some applications this is not enough. A microprocessor-based product may require between 24 and 48 channels to be observed simultaneously. Here's where logic analyzers come in. Not only can certain logic analyzers display many channels, but they can acquire data on these channels simultaneously. An oscilloscope must time share between channels, in an alternate or chop mode, to acquire any data on its 8 signals.

An advantage oscilloscopes have over logic analyzers is the ability to display the actual signal amplitude versus time. All logic analyzers, which are capable of displaying a timing diagram, must display a pseudo waveform. This is because logic analyzers never acquire the actual voltage level for each point in time. Instead, most analyzers simply determine whether the signal is above or below a given threshold voltage. Some oscilloscopes can also be made that will acquire, store, and display pulses that are less than a nanosecond in width. Present-day logic analyzers cannot acquire subnanosecond signals, nor resolve nanosecond or subnanosecond time intervals.

Note: Logic Analyzer data acquisition will be discussed in more detail in Chapter 3.

A special capability found in all logic analyzers and some digitizing oscilloscopes, is the ability to position data so events before the trigger may be observed. This is a powerful feature. Now occurrences that led up to a fault can be seen. The fault itself can easily be triggered on , thanks to the word recognition feature found in almost all logic analyzers. This is the ability to trigger on a pattern of high-level and low-level signals. Word recognizer accessories are also available for triggering logic analyzers and oscilloscopes.

Note: Word recognition is covered in more detail in Chapter 4.

Another advantage logic analyzers have over oscilloscopes is formatting the display. In addition to displaying data as a timing diagram, many logic analyzers can display these same hi and lo logic levels as: binary, octal, hexidecimal, decimal, mnemonics, ASCII, GPIB, map, or signatures (a signature analyzer display) . Thus, the operator has the data presented in a way that is most meaningful.

Note: Examples of different types of displays are shown and discussed in Chapter 2.

Logic Analyzers vs. MDL/MDA

An MDA (microprocessor development aid) is any piece of hardware/software used to develop a microprocessor-based product. The MDA could be a microcomputer on a board. An MDL (microprocessor development lab) is a more powerful and elaborate MDA. Both can interact with the microprocessor program. A logic analyzer will either not interact at all with the device under test or interact in a very limited fashion. One line from the logic analyzer might be connected to cause the device to reset or halt. In contrast, MDL can run the user's program inside of itself, or partially in the MDL and partially in the user's product.

MDL and specialized logic analyzers may have some things in common, such as the ability to display data in a similar format, or that connections are only made at the microprocessor socket. If the user desires to look at data anywhere other than at the microprocessor socket, either a specialized logic analyzer or a general-purpose logic analyzer is necessary. Both types can also be hooked up to other points on the board. To get the most versatility the user will need both an MDL and a logic analyzer with general-purpose capabilities.

Types of Logic Analyzers

The capabilities of a digital analyzer determine its type. Types include: general purpose, serial, signature, and microprocessor. More than one type may be found in the same instrument. All general purpose logic analyzers can be connected to various points on a system. Some can even handle more than one logic voltage level, such as TTL at +1.4 volts and HTL at +7.5 volts. Most general purpose logic analyzers can acquire data with or without an external clock. The types of general purpose displays will vary, but will always include a timing diagram and/or state table. Data can be acquired in parallel channels with timing between channels compared, or one of the channels may be viewed to look at serial timing information.

However, a serial analyzer is necessary to do a serial-to-parallel conversion on serial data for a meaningful display and flexible triggering. Displays will vary, but hexidecimal and ASCII are common.

Note: Serial analyzers are covered in Chapter 5.

A type of digital analyzer that is available in a package by itself, or in combination with an other analyzer, is a signature analyzer. This instrument displays a 4-character signature that is a compression of data acquired at a single point on a board. The signature is not interpreted, but rather compared with a known good signature to determine whether the system is operating properly to that point on the board.

Note: Signature analyzers are discussed in Chapter 7.

Another type of analyzer is a microprocessor analyzer. This instrument is ideal for use on a microprocessor-based system. It will have 24 or more input channels to acquire data on the data bus, address bus and control lines. The microprocessor socket may be the only allowed source of connection or there may be additional generalpurpose channels available. The rate of data acquisition is usually limited to 20 megahertz. Displays will vary, but often include mnemonics which is disassembled machine code. To display mnemonics a personality card or module is required for each microprocessor to which the analyzer is connected.

Note: Microprocessor analyzers are covered in more detail in Chapter 6.

A term used by some manufacturers is "data analyzer." A data analyzer is simply a logic analyzer. In other words, a serial data analyzer is a serial logic analyzer. A data analyzer may also be a logic analyzer with an additional capability.

Another term used is "logic state analyzer." This means that the logic analyzer has no internal clock for acquiring data. An external clock must be used. As long as the user is only interested in data that occurs at the system clock edge, this is not a problem.

Chapter 1 Quiz

Answer the questions below. Then check your responses against the answer key at the back of this book.

1. An advantage many logic analyzers have over nondigitizing oscilloscopes is:
   a. That no oscilloscope has 8 channels.
   b. Display formatting.
   c. Timing diagram.
   d. More narrow time resolution.

2. Another advantage many logic analyzers have over nondigitizing oscilloscopes is:
   a. Data prior to the trigger is displayed.
   b. Overshoot is displayed.
   c. Undershoot is displayed.
   d. The display can be stored.

3. An advantage some oscilloscopes have over logic analyzers is:
   a. Digital memory.
   b. More narrow time resolution.
   c. More channels.
   d. None of the above.

4. An advantage all oscilloscopes have over logic analyzers is:
   a. Smaller time-per-division increments.
   b. Larger time-per-division increments.
   c. The actual amplitude of the signal is displayed.
   d. Formatting.

5. The following are five common types of digital analyzers that are associated with logic analyzers:
   a. General purpose, serial, signature, latch, microprocessor.
   b. General purpose, signature, latch, microprocessor, digital.
   c. General purpose, serial, signature, microprocessor, logic state.
   d. General purpose, signature, latch, microprocessor, delay.

6. An advantage of a microprocessor development aid/system/lab over a logic analyzer is:
   a. Faster clock rate.
   b. A larger screen.
   c. Considerable ability to interact with the data.
   d. Data displayed in mnemonics.

7. An advantage of a logic analyzer over a microprocessor development aid/system/lab is:
   a. It can be connected to points other than the microprocessor socket.
   b. A larger screen.
   c. Mnemonic display.
   d. Hexadecimal display.

Types of Displays

A major benefit to the logic analyzer user is the formatting of acquired data into a display which is relevent to his or her needs. The best display for one application might be a timing diagram; for another case, a GPIB-oriented display might be most suitable. The availability of a variety of displays assures ease of use, less time for data interpretation, and fewer errors. This chapter describes the types of logic analyzer displays available today.

Timing Diagram

To look for timing sequences and propagation delay times, a timing diagram display is appropriate. Figure 2-1 shows a typical example. Notice that the waveforms are very rectangular and clean looking. Information, such as ringing and risetimes, never gets into the logic analyzer memory. For this reason, the waveform in memory for each point in time that was sampled is simply put on the display as a high horizontal line or a low horizontal line. The formatter draws a vertical line to connect any transition from a hi to lo, or lo to hi.

Because the logic analyzer can acquire and display data that occurred before a trigger, the location of the trigger point might be on the right side of the display (as in Figure 2-1) rather than the left, as would be the case with an oscilloscope. When the trigger is on the right and the data is to the left of the trigger, this is referred to as either "trigger after" or "data pre-trigger." If the user chose to see half the data that occurred before the trigger and half the data that occurred after the trigger, data would be acquired and displayed in the centered or center mode, as in Figure 2-2.

To see all, or almost all, of the data that occurred after the trigger, the user would select, acquire, and display data post-trigger ("trigger before") (as in Figure 2-3).

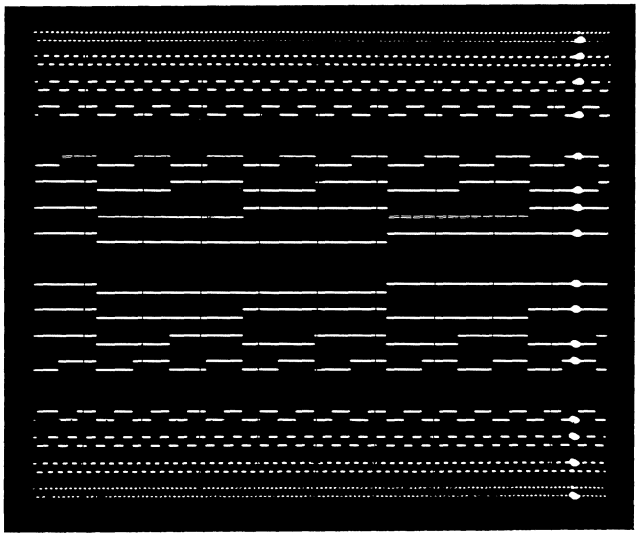


Figure 2-1. Trigger after data, or pre-trigger.

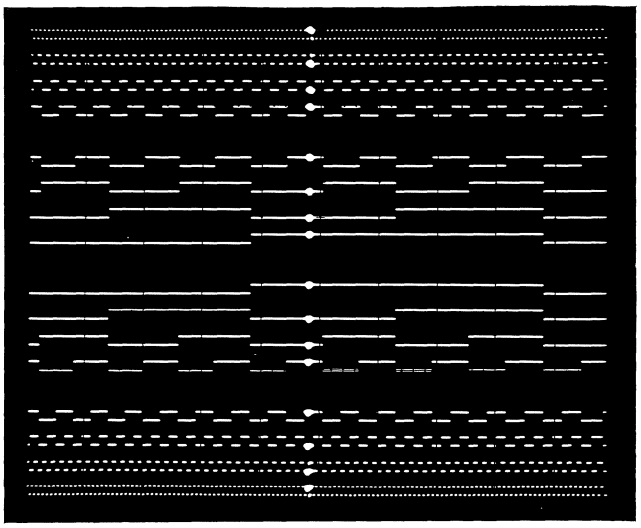


Figure 2-2. Centered.

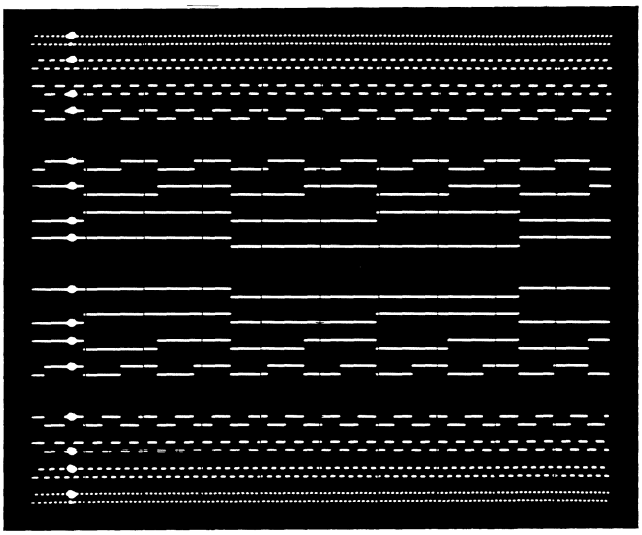


Figure 2-3. Trigger before data, or post-trigger.

State Table

Figures 2-4, 2-5, and 2-6 show state table displays. A binary state table shows the binary value of each individual channel, for a word acquired in parallel. Channel 0 is the right column. The octal and hexadecimal state tables simply condense this same information.

Most logic analyzers are capable of storing more words than a state table can display at one time. To see these other state tables in memory, the user scrolls the display, thus moving the next table onto the display.

There are two ways to convert binary to octal. One way is to treat all of the channels as if they were forming a single word and do a binary-to-octal conversion on the entire word. The other way is to break up the channels into 8-bit bytes and do a binary-to-octal conversion on each byte separately. This way would be used for an octal address that consisted of an upper and lower byte. Notice in Figures 2-7, 2-8. and 2-9 that two different results are produced, depending upon which way the conversion is done. A few logic analyzers are capable of both methods, selectable by the operator. Breaking up a 16-bit word into two 8-bit bytes for octal conversion is referred to as "split octal."

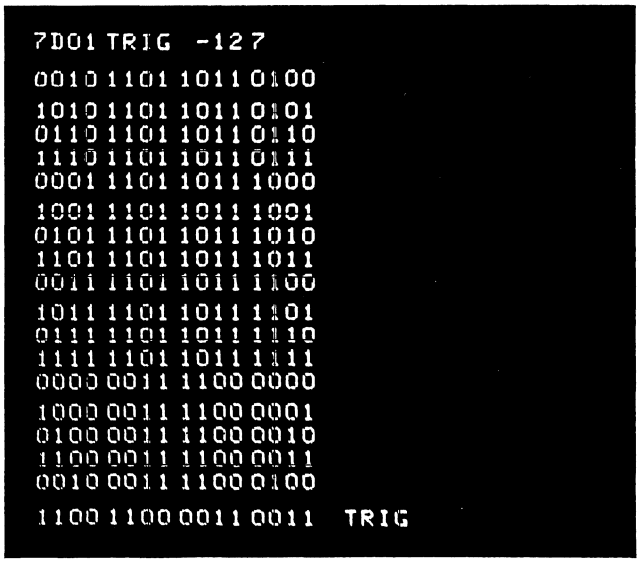


Figure 2-4. Binary display

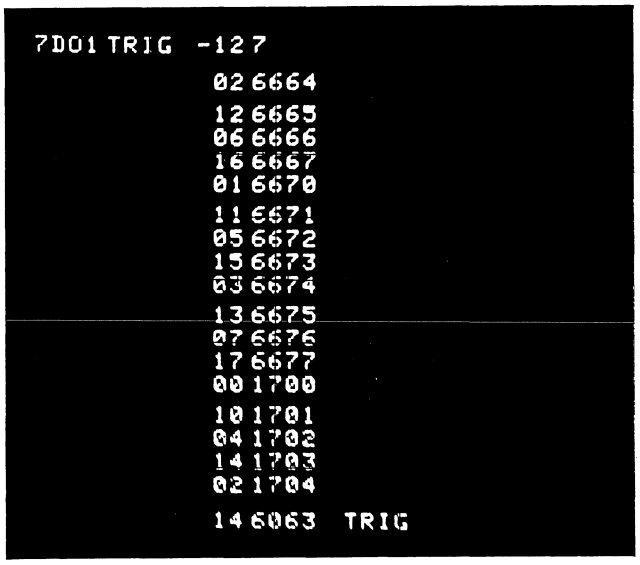


Figure 2-5. Octal display.

Mnemonic

A typical mnemonic display is shown in Figure 2-10. A mnemonic display is useful for those who work with assembly-level languages. "JNZ" (Jump if Not Zero) might be more meaningful than just "C2", the hex equivalent for an 8080/8085 microprocessor.

Shown in Figure 2-11 is another typical mnemonic display. Data displayed are: address bus signals, in hex, decimal; data bus signals, in hex, decimal; data bus signals, in mnemonic code if the microprocessor control lines indicate that the data bus is doing a memory read fetch function; and what the data bus is doing. Figure 2-12 illustrates another useful display: ASCII. The logic analyzer formatter does the conversion to this common code, saving the user time and possible errors.

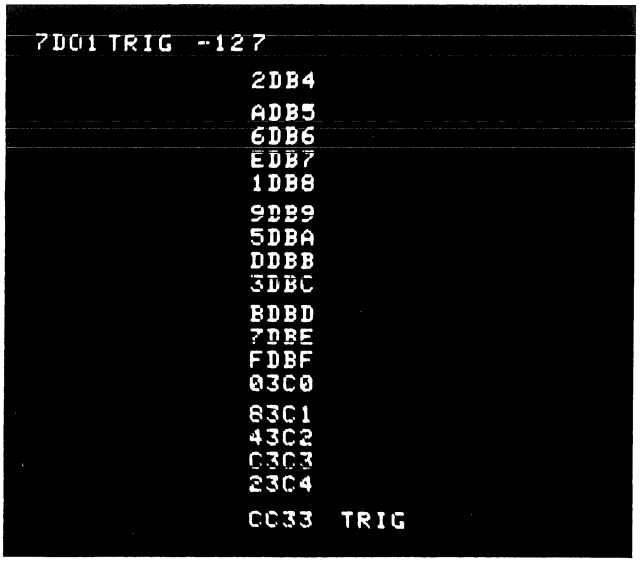


Figure 2-6. Hex display.

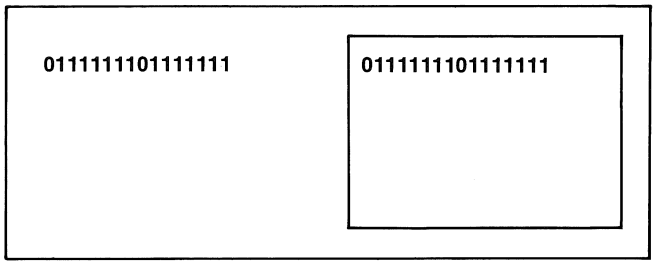


Figure 2-7. Binary.

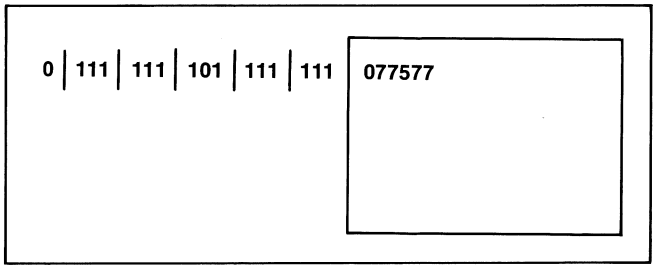


Figure 2-8. 16-bit word divided into groups of 3.

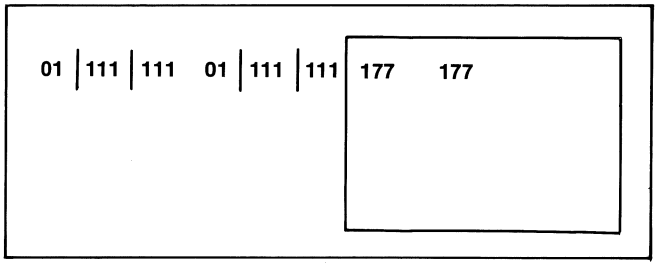


Figure 2-9. 16-bit word divided into two 8-bit bytes, then divided into groups of 3.

ADDR	DATA	MNEMONIC		EXTERNAL	BUS
0388	03			00000000	M R
037B	C2	JNZ	M	00000000	M R F
037C	00			00000000	M R
0326	00			00000000	M W--
037D	7E	MOV	A,M	00000000	M R F
0326	00			00000000	M R
037E	B9	CMP	C	00000000	M R F
037F	C2	JNZ		00000000	M R F
0380	85			00000000	M R
0381	03			00000000	M R
0385	2B	DCX	H	00000000	M R F
0386	C3	JMP		00000000	M R F
0387	7B			00000000	M R
0388	03			00000000	M R
037B	36	MVI	M	00000000	M R F
037C	00			00000000	M R

Figure 2-10. Microprocessor mnemonic display combines with 8 general purpose lines.

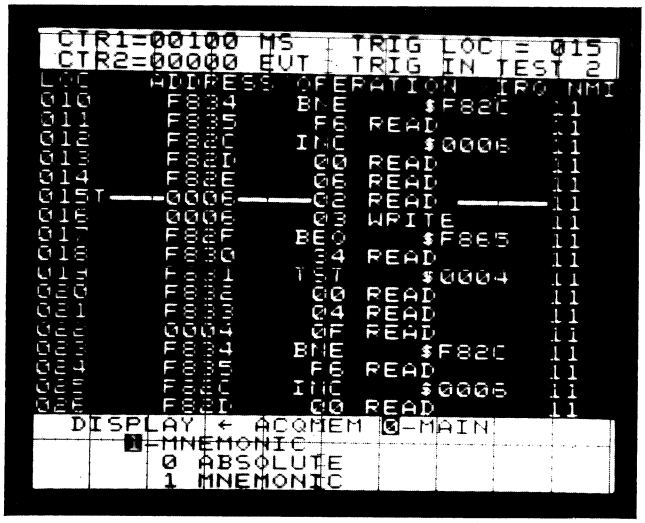


Figure 2-11. Microprocessor mnemonic display.

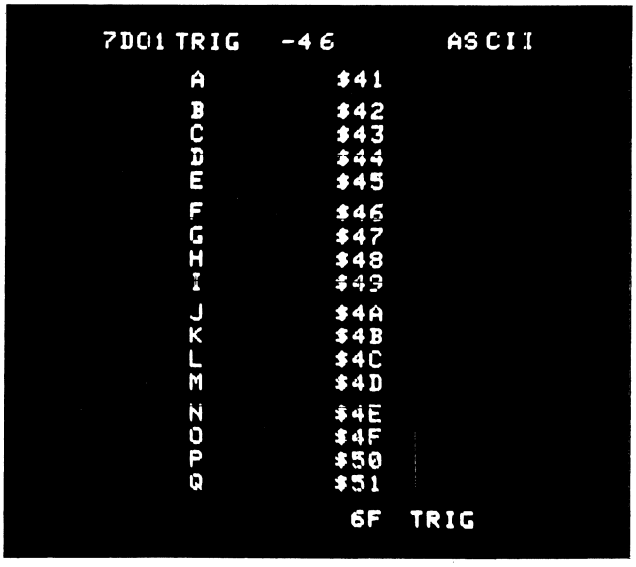


Figure 2-12. ASCII display.

GPIB

Figure 2-13 shows a display of information taken from the GPIB (General Purpose Interface Bus). When GPIB lines such as Attention, End or Identify, Service Request, and Remote Enable are asserted, they are indicated by ATN, EOI, SRQ, and REN. These acronyms are more meaningful than zeroes and ones. Activity on the GPIB data bus is also shown in a way most useful to the user. Since ASCII is a common, but not required, code for the GPIB, the data is also shown both in decoded ASCII and hexidecimal. The ATN mnemonic is also shown, when the Attention line is asserted. Listen, Talk, and secondary addresses are displayed in decimal.

In Figure 2-13, there are four general purpose channels; three of which might be used to monitor the GPIB handshake bus. Since timing of the handshake lines is important to the proper operation of the GPIB, the logic analyzer operator might choose additionally to acquire and display the handshake lines as a timing diagram.

Map

The map display, shown in Figure 2-14, is produced by taking each acquired word in memory and dividing it into two parts. An 8-bit logic analyzer, for example, would divide each 8-bit word in memory into 4-bit halves. Each half then goes to a digital-to-analog converter and then to the vertical and horizontal amplifiers, respectively, of the X-Y display device. A raster scan display uses a more complicated means to achieve the same results.

A 4-bit half word, sent to the vertical amplifier, will deflect the display crt beam to one of 16 (or 2 to the 4th power) positions. The other 4-bit half of the 8-bit word, sent to the horizontal amplifier, will deflect the display crt beam to one of 16 positions. Unblanking the Z axis produces a dot whose position corresponds to the binary value of the word stored in the logic analyzer memory. Sixteen possible vertical positions by 16 possible horizontal positions yields 256 (16x16) possible positions each time a dot is produced. Another way to derive this is to raise 2 to the 8th power which equals 256. (Remember, we started out with an 8-bit word.) Similarly, a 4-bit word produces 16 possible positions, and a 16-bit word produces 65,536 possible positions.

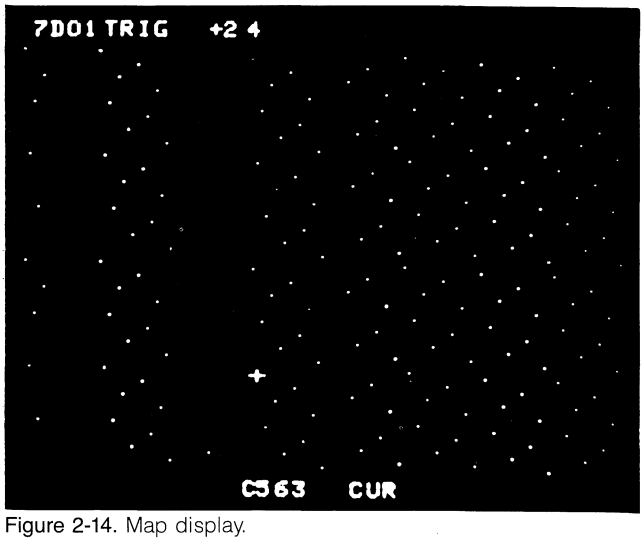


Figure 2-14. Map display.

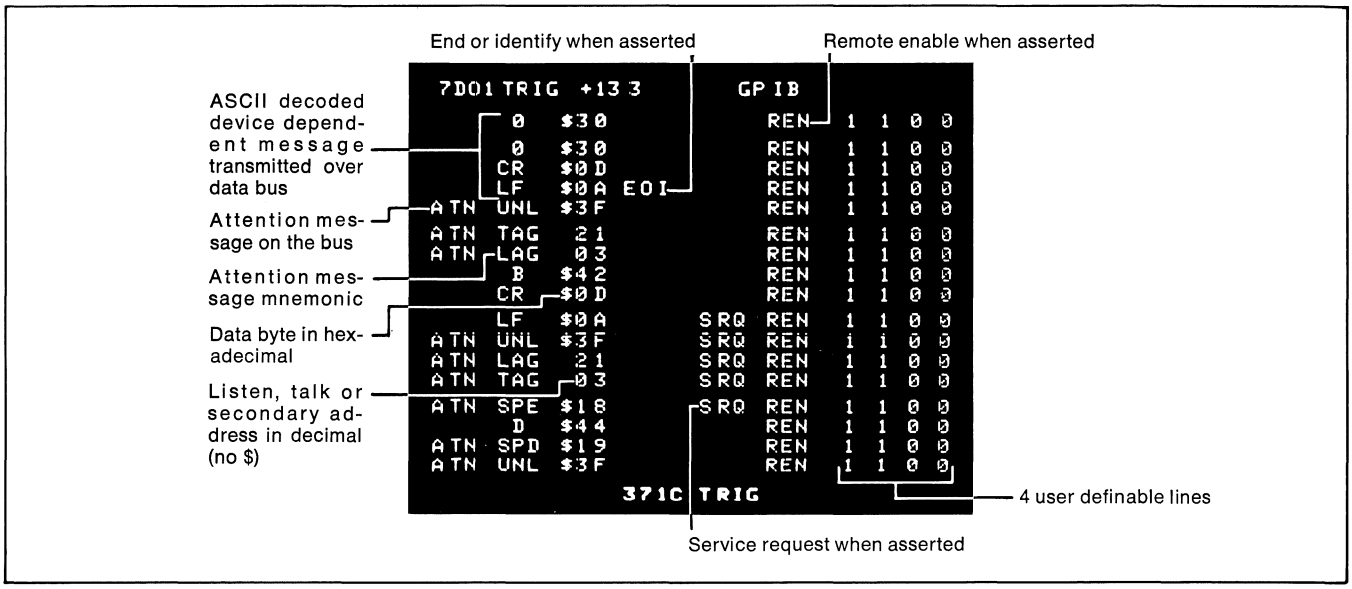


Figure 2-13. GPIB display.

One use of the map display is in watching program flow. In this case the logic analyzer is connected to the address bus of the computer being tested. These addresses are stored in the logic analyzer's memory, then displayed as a dot whose position corresponds to each address acquired by the analyzer. Subroutine addresses will likely be acquired more than once, thus producing brighter dots on the display.

In the example shown in Figure 2-14, a cursor is displayed on screen. The movement of this cursor corresponds to the sequence of acquired addresses from the address bus of the computer under test. It is this movement, and the brightness of the dots, that is of interest to the user.

Another use for a map display is to look for a change, any change. A known good pattern is displayed and documented. This is a sort of signature, which is compared to each newly acquired map. A difference of just one bit, in one out of all the words acquired, can produce a dramatic change in the displayed map.

Signature

Another display that will produce an easy-to-recognize pattern is a signature display. An example is shown in Figure 2-15. Each of the four digits can display any one of 16 different characters, thus giving 65,536 (or 16 to the 4th power) different possible signatures. The 16 characters correspond to the 16 hexidecimal values, 0 through F. In order to present characters on a 7-segment display that are easy to recognize, the actual characters displayed are: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, C, F, H, P, and U. Since the user is matching signatures with documentation rather than interpreting them, this is not a limitation.

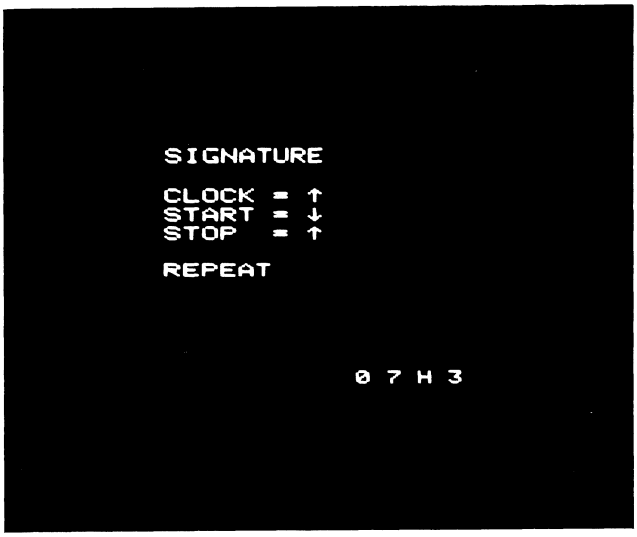


Figure 2-15. Signature display.

Chapter 2 Quiz

Answer the questions below. Then check your responses against the answer key at the back of this book.

1. **The trigger is before the data and the data is after the trigger in the:**
   a. Pre-trigger mode.
   b. Center-trigger mode.
   c. Post-trigger mode.
   d. None of the above.

2. **Propagation delay measurements can be made best with:**
   a. A timing diagram display.
   b. A map display.
   c. An X-Y display.
   d. A raster scan display.

3. **In a binary state table, channel 0 is located (with respect to the operator):**
   a. At the top.
   b. At the bottom.
   c. At the left column.
   d. At the right column.

4. **In a map display, the position of each dot represents:**
   a. The location in logic analyzer memory.
   b. The binary value of the word.
   c. The ASCII value of the word.
   d. The location in the logic analyzer reference memory.

CHAPTER 3

Data Acquisition

This chapter discusses the means by which a logic analyzer brings data into its memory in preparation for formatting the data for the display chosen by the user.

Comparator

Figure 3-1 shows a block diagram of a logic analyzer. The individual probes to each parallel channel are connected to comparators. A single-threshold comparator will output a hi if the input is above the threshold selected by the user, and a lo if the input is below that threshold. As long as the input is 100 millivolts or more above or below the threshold, the comparator will output a hi or lo, respectively. This is what is stored in memory, not the actual voltage of the input signal.

Probe Pod

The input channel comparators might be located inside the logic analyzer or in probe pods that are external to the logic analyzer. Regardless of where the comparators are located, probe pods are more convenient because there are fewer connections to the logic analyzer than with individual probes.

Synchronous Data

Exactly when each hi or lo is obtained and put into memory is determined by a clock. The logic analyzer user might choose to use a clock that is external to the logic analyzer and synchronous to the system being monitored. This is "synchronous data acquisition." The clock being used is a part of the system under test and tells that system when data is valid. In some cases this clock only tells part of the system when data is valid. There may be other parts of the system under test that are asynchronous to this clock.

Asynchronous Data Acquisition

In fact the entire system under test may be asynchronous, with no clock at all. Many general purpose logic analyzers can use a clock that is internal to the logic analyzer to acquire this data. This is "asynchronous data acquisition." Whether all or part of the system under test is asynchronous, the clock internal to the logic analyzer serves to tell the logic analyzer when to acquire data. The asynchronous clock interval is selectable by the user.

Clock Qualifier

Often a digital bus in the user's system will not only have data of interest, but irrelevant data as well. The logic analyzer user might only be interested in data on a data bus during memory read but not memory write; or perhaps what it wanted is the reverse: memory write, but not read. There are always one or more control lines in the system under test which will tell the sytem when there is a memory read, when there is a memory write, and so on.

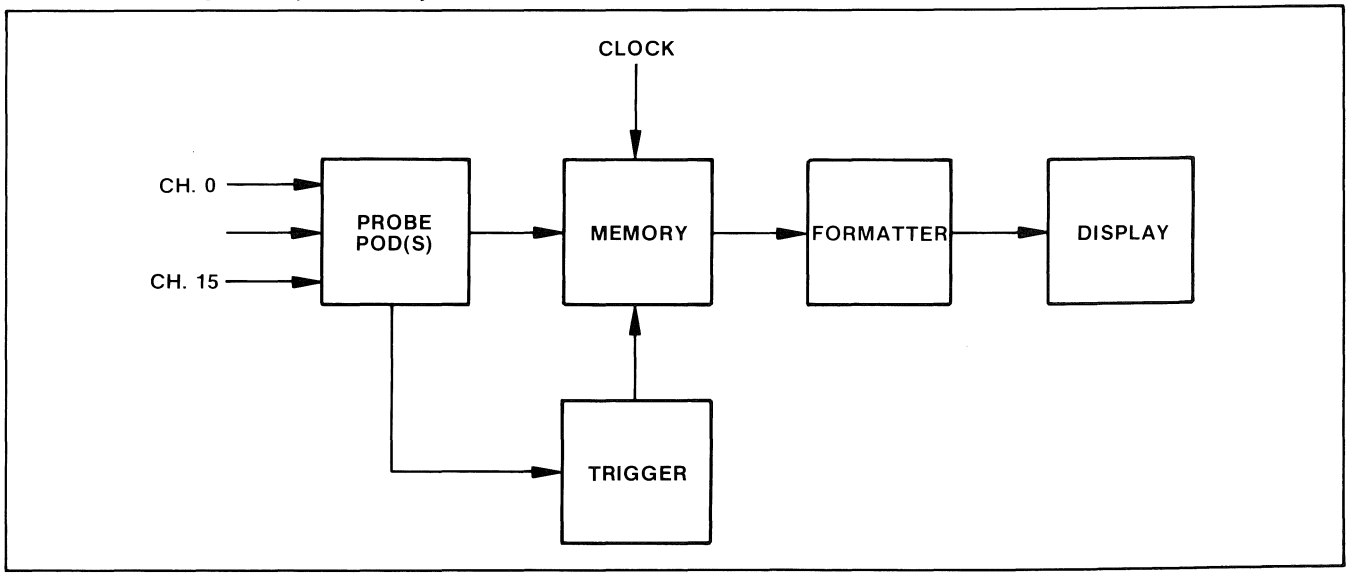


Figure 3-1. General purpose logic analyzer block diagram.

A logic analyzer with a clock qualifier can gate the external synchronous clock through the logic analyzer with the selected control line. Figure 3-2 shows an example of this for an 8080A. Notice that the microprocessor line labeled DBIN is hi when the data bus (D7-D0) has data input. If the logic analyzer clock qualifier line is connected to the DBIN pin on the microprocessor, the external synchronous clock can only get into the logic analyzer to clock in data, when DBIN is hi. As a result, data out never gets into the logic analyzer, which is just what the logic analyzer user intended.

This keeps irrelevent data off the display and saves memory space in the logic analyzer.

Note: How a clock qualifier works is shown in Figure 3-3.

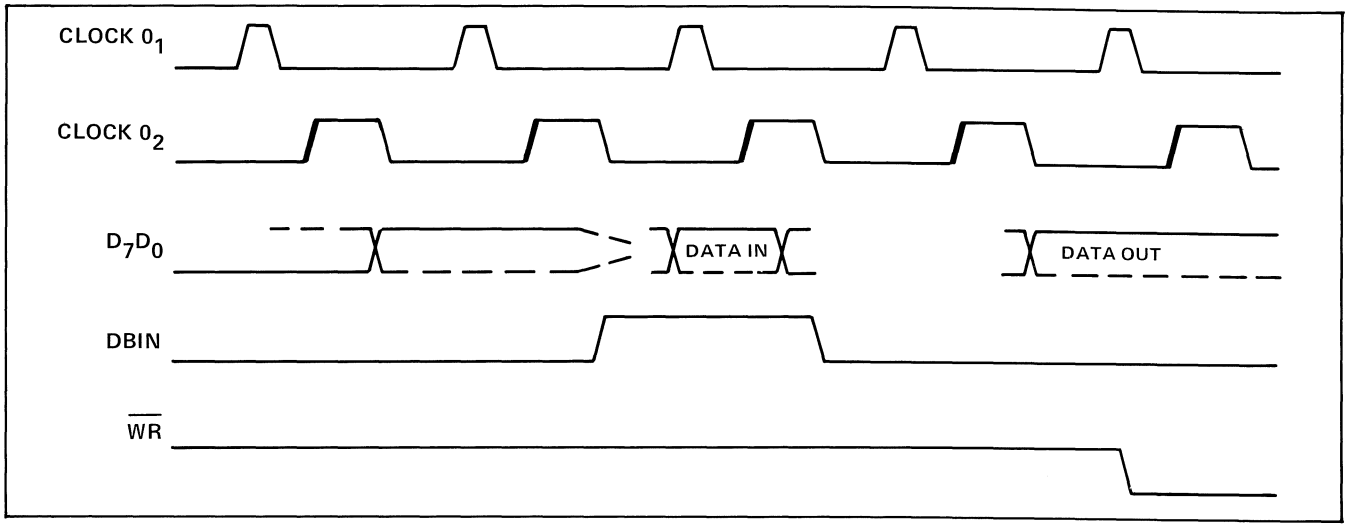


Figure 3-2. 8080A timing waveforms.

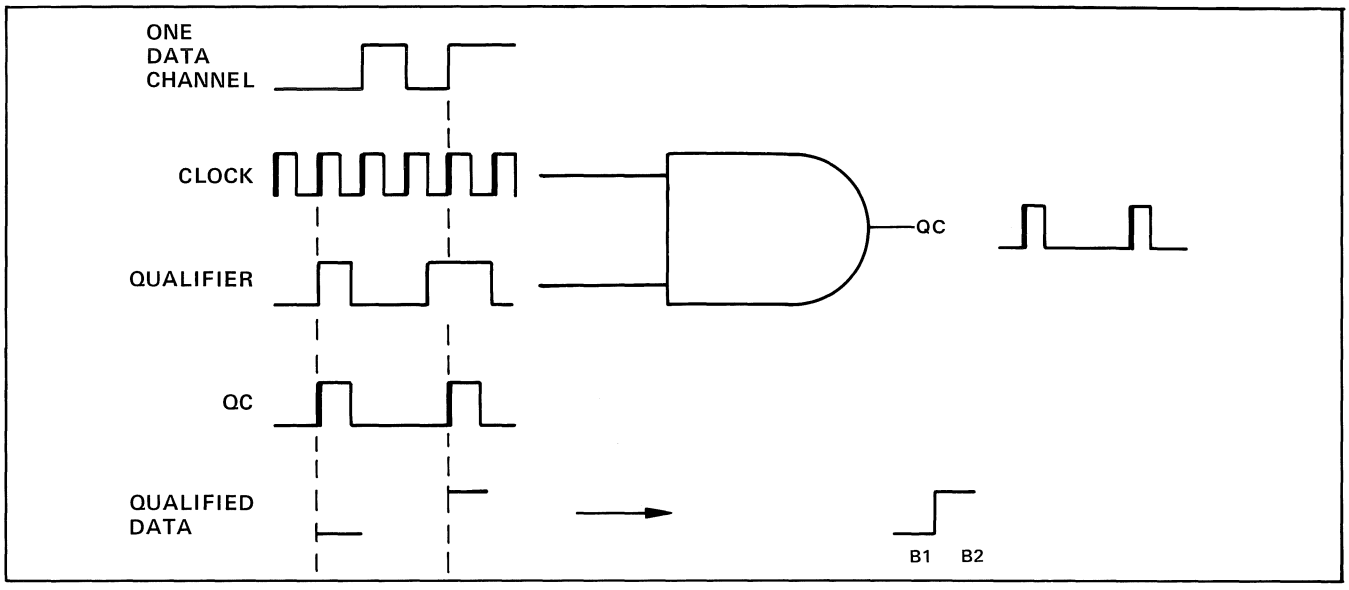


Figure 3-3. Clock qualifier.

Digital Latch

A glitch catcher, or digital latch, simply latches very narrow pulses a logic analyzer would otherwise miss. An example of this operation is seen in Figure 3-4. The positive and negative pulses shown in the actual signal are not coincident with the clock edge. A digital latch feature (or accessory) will effectively stretch these pulses for one clock interval. The stretched pulse can now be stored in the logic analyzer memory. A digital latch feature is not the same as having a short sample interval clock (shorter than the narrow pulse), inside of the logic analyzer, however. As can be seen in Figure 3-5, a narrow pulse (or glitch) next to other data can go undetected.

Glitch memory with this limitation is referred to as a "first-order" glitch capture. A "second-order" glitch capture feature gets around this problem by looking for more than one transition during a clock interval. Refer to Figure 3-5. A glitch next to actual data, all within one clock interval, causes three transitions to be detected by second-order glitch circuitry. With no glitch there is only one transition, which is the actual data going positive. The status of glitch(es) or no glitch is stored in a glitch memory, and can be indicated on the display.

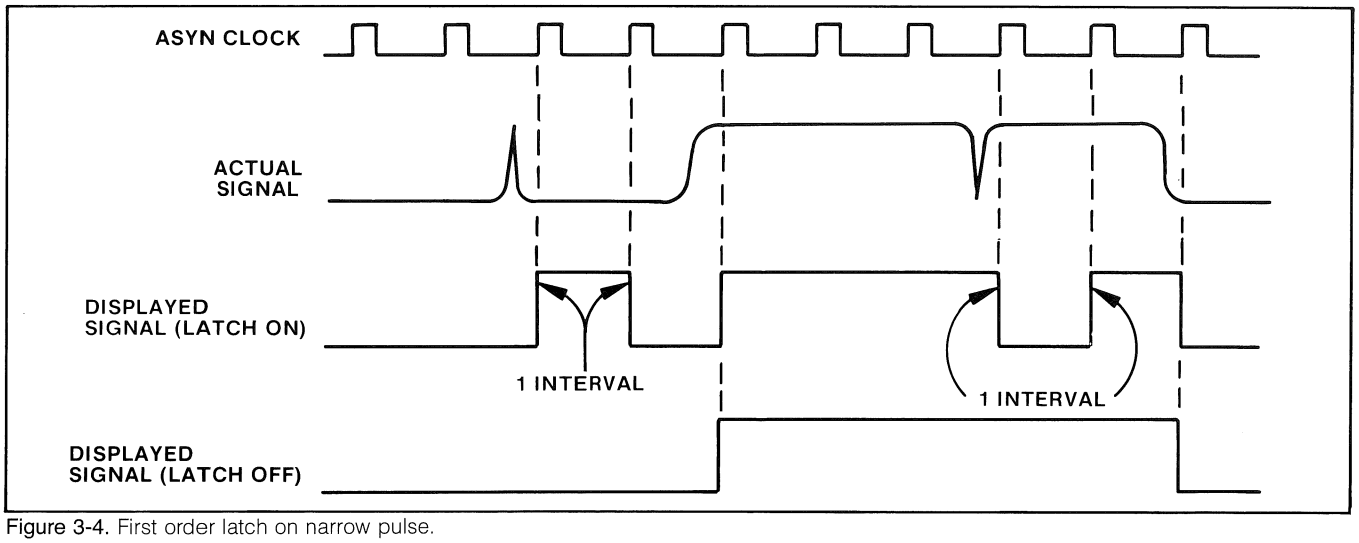


Figure 3-4. First order latch on narrow pulse.

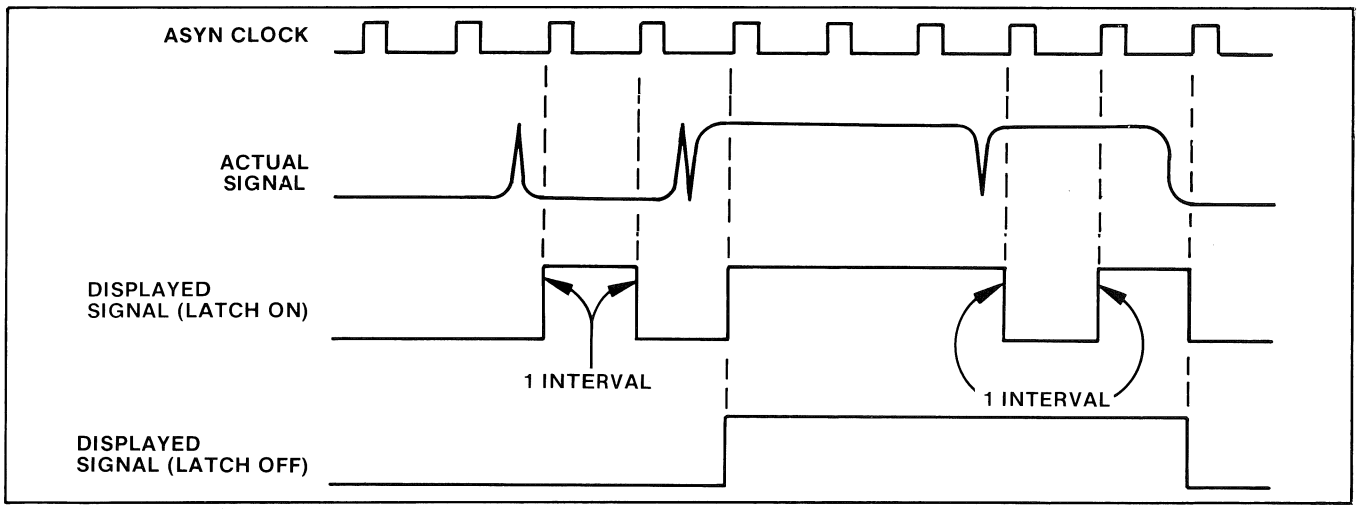


Figure 3-5. First order latch misses glitch next to actual data.

Word Qualifier

It is common to word recognize and acquire data from the same lines on the system under test. However, there are applications where one or two additional lines are needed for word recognition, that need not be acquired and displayed. These lines are referred to as word-qualifier inputs. As can be seen in Figure 4-3, the extra line (Q) effectively extends the width of the word recognizer. This line is not acquired for display.

Word Recognizer Filter

In an asynchronous system under test, narrow glitches might be present, but ignored by that system. The glitches are often found next to a transition from a lo to a hi or hi to lo. Glitches between clock edges are ignored in synchronous word recognition, but could by coincidence produce the right pattern of hi's and lo's, which would be recognized in asynchronous word recognition. By extending the rise and fall times of each line prior to the AND gate, these glitches can effectively be filtered out of the asynchronous word recognition process. Figure 4-4 shows the effect of extending the transition times to 50 nanoseconds.

Because the word recognizer filter is only needed and used in asynchronous word recognition it is often referred to as an "asynchronous filter."

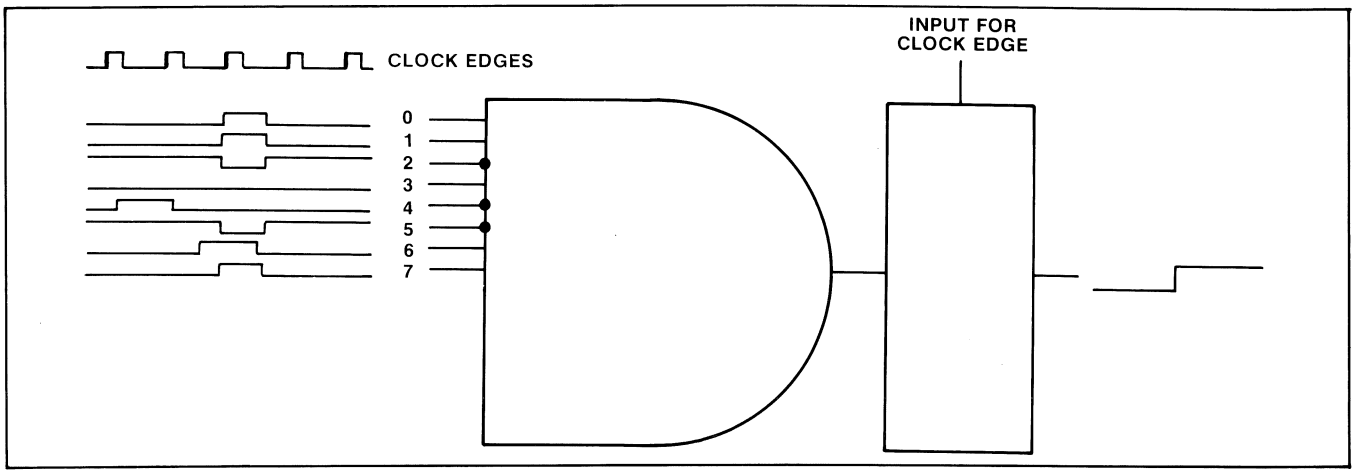


Figure 4-2. Synchronous word recognition.

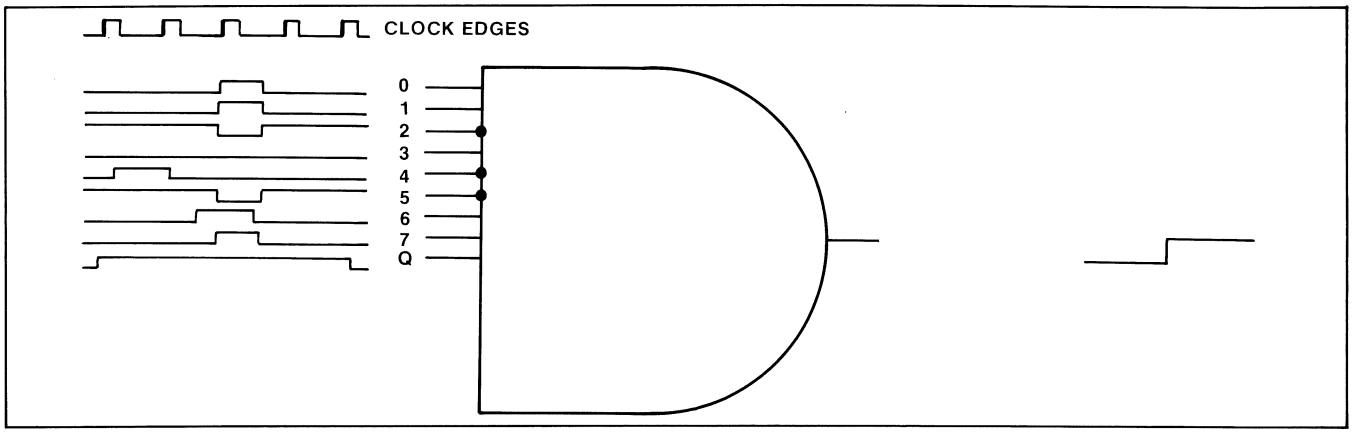


Figure 4-3. Word recognition with additional word qualifier input.

Range Recognition

Range recognition can be useful in finding problems in a program. As an example, a program stored in the address range between 0200 and 0300 may jump out of this range for no apparent reason. Also where the program jumps to may not be known or may vary. A range recognition feature may generate a trigger if any address (trigger word) outside of 0200 to 0300 occurs.

An example of the opposite situation is a subroutine stored in the address range between 0200 and 0300 that is entered at different points. In other words, one time the main program may jump to address 0200, but at other times the main program is expected to jump to address 0250 or 0270. In this case a trigger needs to be generated whenever an address within the range of 0200 to 0300 occurs. A range recognition feature causes a trigger within that range.

Digital Delay

At times it is desirable to capture a block of data that occurs many clock cycles downstream from the trigger. One example would be to acquire the block of data that occurs just after the block of data presently stored in the logic analyzer memory. If the logic analyzer memory holds 250 words, then a digital delay could be set to count 250 clock cycles after the trigger word. The output of the digital delay then stops the data acquisition process. The result is that the next block of data can be captured without changing the trigger word.

Another use for digital delay is to generate a trigger, not on the first occurrence of the trigger word, but the second, third, or 100th. The digital delay would be set to count the appropriate number of trigger words, then output a trigger to stop the data acquisition process.

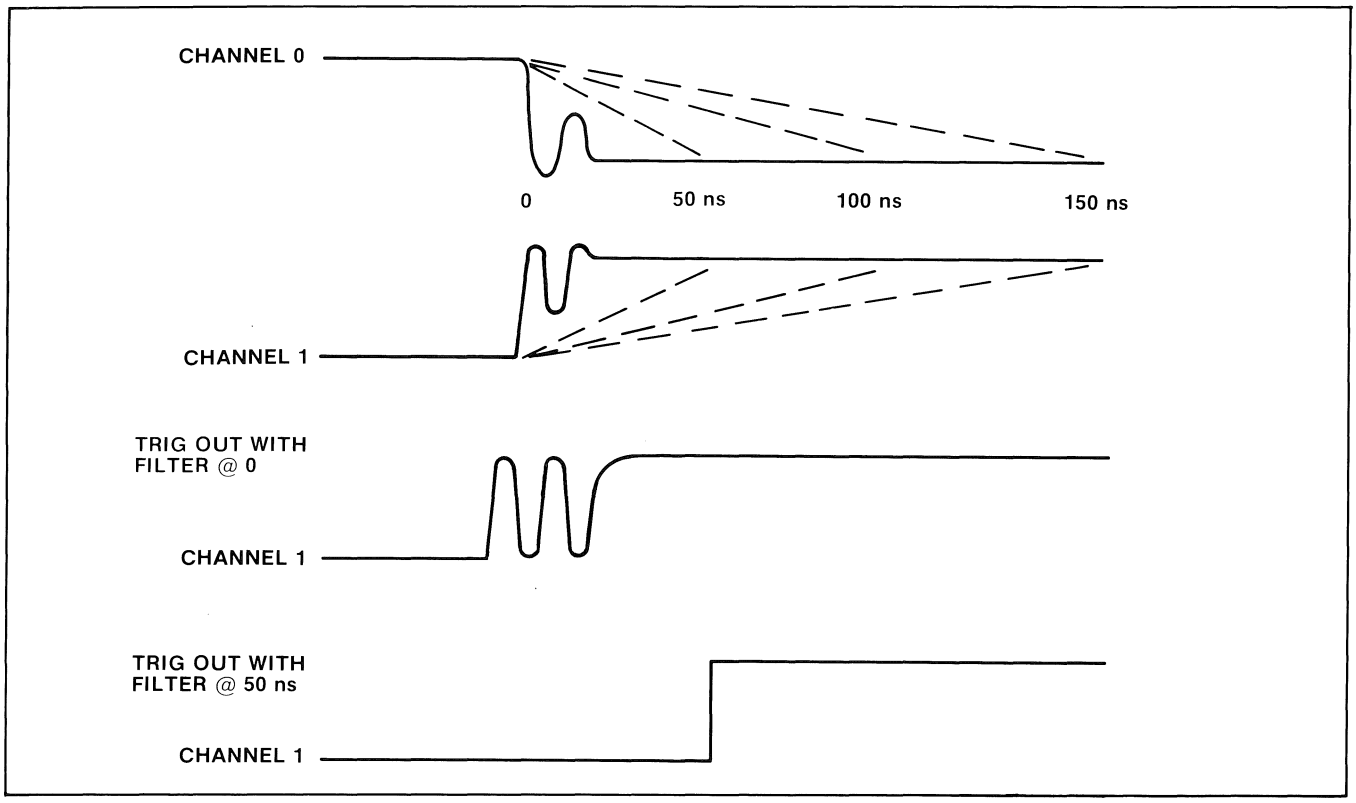


Figure 4-4. Asynchronous filter action for word recognition.

Chapter 4 Quiz

Answer the questions below. Then check your responses against the answer key at the back of this book.

1. Word recognition is:
   a. Triggering on a pattern of hi's and lo's.
   b. Starting data acquisition on a pattern of hi's and lo's.
   c. Both a and b for post-trigger.
   d. Searching through reference memory for a match with the chosen word.

2. Triggering:
   a. Starts data acquisition.
   b. Stops data acquisition.
   c. Resets the logic analyzer.
   d. None of the above.

3. Asynchronous word recognition:
   a. Uses the internal logic analyzer clock.
   b. Uses the external system clock.
   c. Uses no clock.
   d. Is the same as asynchronous data acquisition.

4. The word qualifier:
   a. Selectively acquires data.
   b. Is normally not stored in the logic analyzer memory.
   c. Both a and b.
   d. Is used when there is no normal system clock.

5. Range recognition:
   a. Will generate a trigger if the trigger word is inside a given range.
   b. Will generate a trigger if the trigger word is outside a given range.
   c. Either a or b.
   d. None of the above.

6. A digital delay:
   a. Can count clock cycles.
   b. Can count trigger words.
   c. Either a or b.
   d. Allows viewing of a clock.

7. A digital delay:
   a. Allows capturing data many clock cycles before a trigger.
   b. Allows capturing data many clock cycles after a trigger.
   c. Either a or b.
   d. None of the above.

8. To avoid word recognizing on glitches:
   a. A glitch latch would be used.
   b. A digital latch would be used.
   c. Both a and b.
   d. A word recognizer filter would be used.

Advanced Logic Analyzer Concepts Videotape Quiz

First view the videotape entitled *Advanced Logic Analyzer Concepts*. Then answer the questions below and check your responses against the answer key at the back of this book.

1. A general purpose parallel logic analyzer can be used on serial data, to show:
   a. ASCII characters.
   b. EBCIDIC characters.
   c. ASCII or EBCIDIC characters.
   d. Timing information.

2. A serial analyzer can be used on serial data:
   a. To word recognize.
   b. To handle data protocol.
   c. All of the above.
   d. None of the above.

3. A microprocessor analyzer:
   a. Can acquire data, address, and control information.
   b. Is any general purpose logic analyzer with a microprocessor in it.
   c. All of the above.
   d. None of the above.

4. A microprocessor analyzer:
   a. Will only work with 8-bit microprocessors.
   b. Will usually display mnemonics.
   c. Will usually display GPIB.
   d. Will not display hexadecimal.

5. A personality module:
   a. Is part of the system under test.
   b. Is used to provide an ASCII display.
   c. Is used to provide a GPIB display.
   d. Will determine when an instruction is fetched.

6. Signature analysis:
   a. Uses three data channels.
   b. Uses one control channel.
   c. Uses two control channels, including the clock.
   d. Uses three control channels, including the clock.

7. Signature analysis provides a display of:
   a. one character.
   b. two characters.
   c. four characters.
   d. seven characters.

8. Documentation for signature analysis includes:
   a. Good signature, bad signature, clock.
   b. Clock, strobe, number of characters in the signature.
   c. Clock, start, stop, duration.
   d. Clock, start, stop, jumpers to be lifted.

CHAPTER 5

Serial Logic Analyzers

This chapter describes serial logic analyzers as a generic type, and contrasts them to parallel logic analyzers.

Serial Data Acquisition and Word Recognition

Serial data is encoded and organized in logical groupings with bits in a group transmitted one after the other. An example of an encoding and synchronization scheme can be seen in Figure 5-1. A serial logic analyzer will acquire this data, one bit at a time on its single channel, by using a matching decoding and synchronization scheme to change the serial data into parallel for storage and word recognition.

Once the data is converted to parallel, word recognition for triggering can take place in a manner similar to that in a parallel logic analyzer. Since two consecutive characters can have a special meaning in some protocols, the triggering capabilities often include word recognition of two digital words (characters).

Another feature of serial analysis is its ability to deal with various aspects of data protocol, such as number of bits per character, parity selection, sync word, and hunt word.

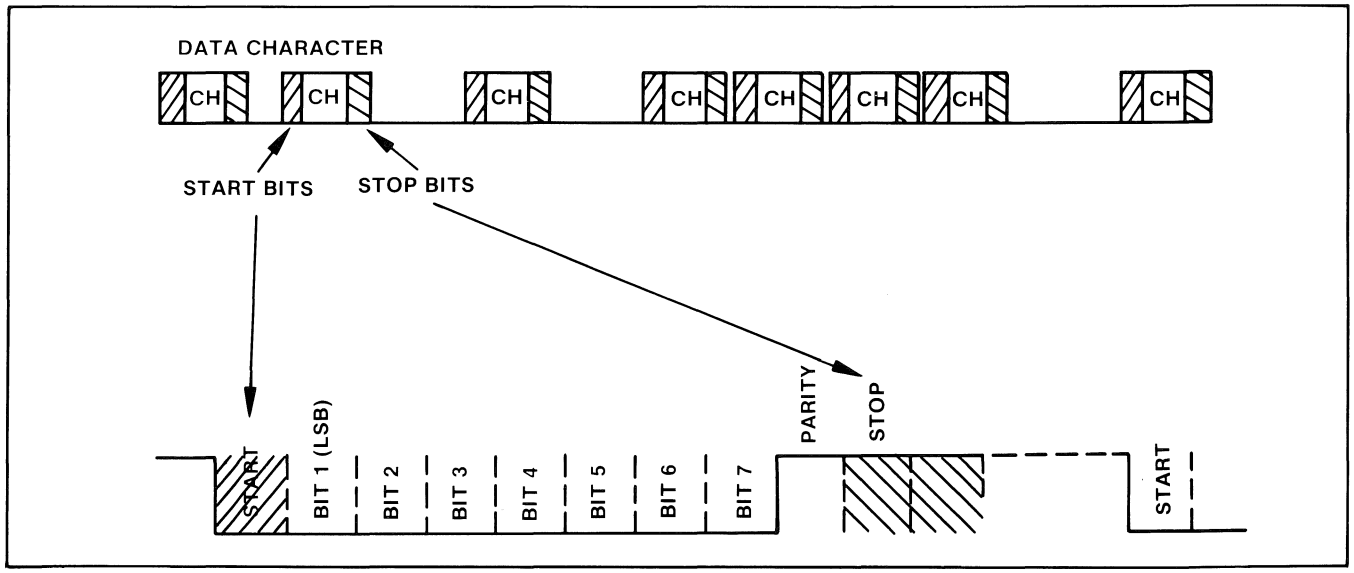


Figure 5-1. Serial asynchronous data.

Protocols

Synchronization for both data acquisition and word recognition can be either asynchronous or synchronous. Figure 5-1 shows an example of asynchronous protocol data. The logic analyzer internal clock is normally used, as with a parallel logic analyzer. The start and stop bits do not need to be stored, but do tell when each character starts and stops. That is, the start and stop bits frame the bits that encode a character. The term "asynchronous protocol" refers to the variance in timing between bytes or characters. The bits within a byte are synchronized to the start bit. The logic analyzer internal clock is also synchronized with each start bit.

Some data communications systems are similar to asynchronous protocol, but with an additional line for a clock. Isochronous protocol is an example of this. Any time a clock is available from the system under test, the logic analyzer can be put in the external clock mode.

Another method of sending data is with synchronous protocol. An example of this is shown in Figure 5-2. The bits in each byte and the group of bytes in a block of data are all in synchronization with the clock available from the system under test. The logic analyzer is used in the external clock mode and directed to look for the two (usually identical) "sync" characters to start acquiring data. When the logic analyzer finds the hunt word (usually an inverted hex FF), that indicates the end of a block of data. The analyzer stops acquiring data, momentarily, and looks for the "sync" characters again.

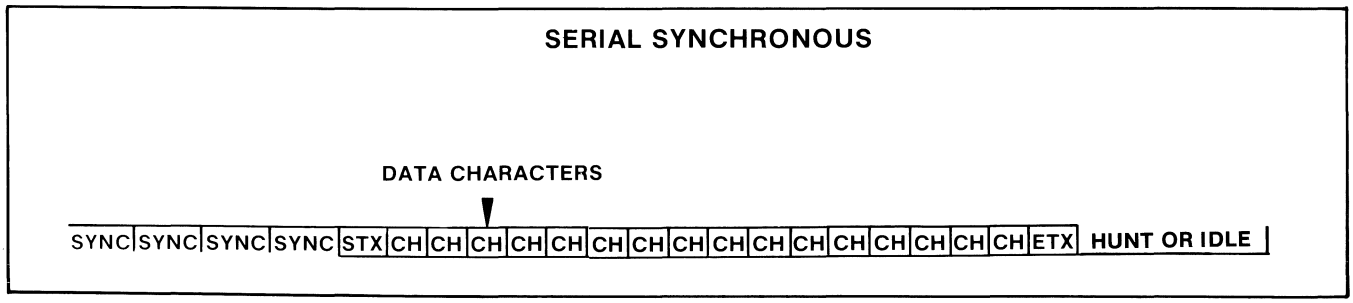


Figure 5-2. Serial synchronous data.

Chapter 5 Quiz

Answer the questions below. Then check your responses against the answer key at the back of this book.

1. A serial logic analyzer:
   a. Is the same as a parallel logic analyzer, with only one channel used.
   b. Converts parallel data to serial for processing and handles serial protocol.
   c. Converts serial data to parallel for processing and handles serial protocol.
   d. None of the above.

2. For isochronous protocols:
   a. The internal logic analyzer clock is used.
   b. The external system clock is used.
   c. No clock is used.
   d. None of the above.

3. For asynchronous protocols:
   a. The internal logic analyzer clock is used.
   b. The external system clock is used.
   c. No clock is used.
   d. None of the above.

4. For synchronous protocols:
   a. The internal logic analyzer clock is used.
   b. The external system clock is used.
   c. No clock is used.
   d. None of the above.

Microprocessor Analyzers

This chapter contrasts microprocessor analyzers with general purpose logic analyzers.

Application

Microprocessor analyzers are ideal for use on systems under test that contain one or more microprocessors. This type of analyzer will have more than 24 input channels to allow acquisition of 8 parallel bits of data, 16 parallel bits of address and certain control lines, at the same time, from an 8-bit microprocessor. To also acquire additional control lines, or a wider data bus, or a wider address bus, requires even more channels. Probe connection is always at the microprocessor socket.

General Purpose Channels

To input any other data, such as peripheral activity, requires general purpose channels. The data from these lines will generally be displayed as binary, octal, or hexadecimal.

Mnemonics

Many microprocessor analyzers have the additional ability to take the hi's and lo's acquired from the microprocessor socket and display this information as binary, octal, hexadecimal, decimal, or mnemonics. Mnemonics are abbreviations for the instructions the microprocessor is carrying out. Which mnemonic matches with a given pattern of hi's and lo's varies with different microprocessors. A pattern equal to 20 in hexadecimal matches with the mnemonic "RIM" (Read Interrupt Mask) for an 8085 microprocessor; but for a 6502 microprocessor, the matching mnemonic is "JSR" (Jump to new location Saving Return address). For this reason an item known variously as a personality module, personality probe, or personality card must be used to tell the microprocessor analyzer which mnemonic matches with the data word, on the microprocessor data bus, during a fetch cycle.

General Purpose Module

A general purpose module (general purpose probe) can be used to support microprocessors for which personality probes are unavailable. This probe can be connected to the microprocessor pins, but will not produce a mnemonic display. A general purpose module can also be useful for data other than at the microprocessor socket.

Triggering

A microprocessor-based system under test will often have a program containing many loops. This complex looping can be monitored using event counters, timers, and word recognizers to provide a trigger.

For this reason, the operating capabilities of a microprocessor analyzer may include multiple test triggering. An example is when the analyzer is programmed by the user for Test One to look for one word, look for another word, look for yet another word, go to Test Two to perform other tasks, then trigger. These capabilities include recognition of words in parallel and in sequence. Conditional branching is also needed to follow the complex jumping of the microprocessor-based system.

Execution Time

Minimizing the amount of time needed to run (execute) a program or subroutine is often desirable. To reduce running time, a counter may be incorporated in the microprocessor analyzer that will start on one trigger word (the start of the program or subroutine) and stop on another trigger word (the ending address of the program or subroutine). The count from the counter will then be displayed to show either the amount of time elapsed or the number of system clock cycles.

Chapter 6 Quiz

Answer the questions below. Then check your responses against the answer key at the back of this book.

1. A microprocessor analyzer:
   a. Has no formatter.
   b. Has no general purpose channels.
   c. Will always have an ASCII display.
   d. Will usually have a mnemonic display.

2. A microprocessor analyzer:
   a. Must have 10 ns sample interval capability.
   b. Has at least 24 channels.
   c. Has at least 16 channels.
   d. Must have 20 ns sample interval capability.

3. A general purpose module:
   a. Is a general purpose probe.
   b. Can be connected to the microprocessor to acquire data and address.
   c. Can be used for general purpose logic analyzer capability.
   d. All of the above.

4. A personality module:
   a. Allows mnemonic display capability.
   b. Allows ASCII display capability.
   c. Allows EBCIDIC display capability.
   d. All of the above.

Signature Analyzers

This chapter discusses the implementation and documentation of signature analysis.

Implementation

A signature analyzer is used by moving the single channel data acquisition probe from one electrical point (node) in the hardware to another, obtaining signatures. Each signature is made up of 4 of the following characters: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, C, F, H, P, U. There are similar instruments with the same characters:0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, b C, d, E, F. They are not compatible! The signature is used by comparing it to a previously-documented good signature. If the signature matches, the user moves the probe to the next point or node in the hardware, then acquires and displays another signature. When a mismatch is found, the user has isolated the problem area.

Signature analysis came from the use of check characters, used in data communication to verify that the message received had no errors. With the cyclic redundancy check, as an example, continuous calculation is performed as the message is sent, resulting in 4 hex (equal to two ASCII) check characters. A 4-character signature is produced in a similar manner.

Documentation

When documenting good signatures, the following must be considered:

1. Enough points in the hardware must be chosen to allow isolation of the problem to the level desired. More documented points are needed to isolate a problem to the component level than to the board level.
2. For each point where a signature is acquired, a clock source and edge must be specified.
3. A source and edge must be given for a start signal and a stop signal. The data between these two signals will form the signature.
4. The operating mode of the system under test must be specified in order to have consistent signatures. For example, the system diagnostic mode might be to sequence through blocks of addresses.
5. Finally, a troubleshooting tree that lists the sequence of these points is needed to go directly to any problem.

Transition Counting vs. Signature Analysis

The use of transition counting is similar to that of signature analysis. The number of transitions between start and stop events is documented. If the transition counter counts and displays a different number than documented, something is wrong at that test point. If the displayed number matches the documented number, then it is likely that the system is good to that test point. Problems arise because the transitions are counted independently of the sequence in which they occur. Two faults could cancel each other out. A signature analyzer, however, will produce a different signature if the sequence of data changes because a clock is used to acquire the signature (making it time-dependent data). A bit that is in the right sequence, but faulty because of a timing problem, will produce a different signature. The probability of bad data producing a good signature is 1 out of 65,536 (or 1 out of 2 to the 16 power). This is only because there are 65,536 possible bad signatures, only 1 of which is the same as the good signature.

Bad signatures will not tell the user what is wrong with the system under test. For this reason a signature analyzer is often in the same instrument as a logic analyzer or other digital analyzer. Knowing what is wrong may not be necessary if the documentation allows isolation of the problem to a small enough area. Also the user now needs to know very little about the system being tested.

Chapter 7 Quiz

Answer the questions below. Then check your responses against the answer key at the back of this book.

1. Signature analysis is:
   a. Analysis by the user of the signature.
   b. A compaction of the number of transitions in the data.
   c. Obtaining data at one electrical point to produce a 4-character signature.
   d. Obtaining data at one electrical point to produce a 16-character signature.

2. Required additional inputs to a signature analyzer include:
   a. Clock, start, stop, trigger.
   b. Start, stop, trigger.
   c. Start, stop, ETX.
   d. Clock, start, stop.

3. An advantage of signature analysis over transition counting is that:
   a. No clock is required.
   b. A timing problem with the data can produce a fault-indicating signature.
   c. Both a. and b.
   d. None of the above.

4. A limitation of signature analysis is:
   a. The use of other than the standard hex characters.
   b. That there is a high probability of getting the right signature with bad data.
   c. That a high skill level is required to interpret the signatures.
   d. That it will not tell the user what is wrong with the bad area after it is located.

@ 12/80

Appendix A Glossary

address

The information or signal pattern representing the physical location of the source or destination of data within a digital system. A computer generates signals that act at an address for input devices, output devices, or individual words within its memory.

algorithm

A term used by mathematicians and programmers to describe a set of procedures by which a given result is obtained.

aliasing

Sampling data at an insufficient rate to produce an output that is not related to data.

ALU

Abbreviation for arithmetic logic unit. A computational subsystem which performs the mathematical operations of a digital system.

ASCII

Abbreviation for American Standard Code for Information Interchange. A 7-bit code representing 128 characters and control functions. A parity bit is frequently appended making an 8-bit byte. In some systems a 64-character subset is used.

asynchronous data

Digital information transferred independently from any timing signal at the receiving end.

asynchronous device

A device in which the speed of operation is not related to any frequency in the system to which it is connected.

asynchronous word recognition

Word recognition occurs any time the selected word occurs.

binary coded decimal

bcd, abbreviation for binary coded decimal. A binary numbering system for coding decimal numbers in groups of 4 bits. The binary value of these 4-bit groups ranges from 0000 to 1001 and codes the decimal digits 0 through 9. To count 9 takes 4 bits, to count to 99 takes two groups of 4 bits, to count to 999 takes 3 groups of 4 bits, etc.

binary number system

A numbering system using 2 as the base. $11001_2 = 2^4 + 2^3 + 2^0 = 16 + 8 + 1 = 25_{10}$.

bit

Elemental unit or quantum of binary information. Definite position within a binary word (e.g., bit 0, bit 1, bit 2....bit n) where the number represents a power of 2. Bits sometimes are designated in ways that don't relate to powers of 2.

byte

An ordered set of bits, usually 8, that make up a character or a segment of a word. Computer peripherals usually communicate in 8-bit bytes.

CRC

Abbreviation for cyclic redundancy check. Used to check for errors in data transmission. A given polynomial is divided into the data (message) creating a reminder which is the CRC. This number is put on the end of the message for transmission. The receiving end performs the same division on the message, which now contains the CRC characters. If there are no errors in transmission, the remainder is 0.

CPU

Abbreviation for central processing unit. Part of a computer system which contains the main storage, arithmetic unit, and special register groups. It performs arithmetic operations, controls instruction processing, and provides timing signals and other housekeeping operations.

clock

A digital system timing signal that paces data transfer, usually on one edge.

clock (strobe) qualifier

Data acquisition occurs only if certain external qualifying channel(s) are true at the clock (strobe) edge.

combinatorial trigger

A trigger generated when a number of preselected requirements (voltage) meet simultaneously, a trigger is generated.

controller

Digital subsystem responsible for implementing "how" a system is to function. Not to be confused with "timing" which tells the system "when" to perform its function.

critical race

Timing situation related to synchronous or asynchronous operation. A "race" can occur when two variables are asked to change states simultaneously. "Critical" refers to the outcome that will determine the state of the machine.

crosstalk

Interference which appears in a given channel but has its origin in another channel.

cursor

Intensified portion of a display that denotes position or reference for crt readout.

data

An ordered set of words representing information. Computers take in data, process it, then generate more data.

data acquisition

The process by which input logic signals are compared at a discrete instant to a threshold level and recorded.

data skew

Misrepresentation of data caused by parallel channels with different propagation delays.

decoder

A conversion circuit that accepts digital input information that appears as a small number of lines and selects and activates one line of a large number of output lines.

diagnostic

Pertaining to the detection and isolation of a malfunction or mistake.

display clock

In a logic analyzer, the signal that determines the stored data output rate. Same as read clock.

don't care

A condition that does not matter, as in a channel that is ignored in word recognition.

dual-mode threshold

Each channel has two preselected thresholds against which input signals may be compared.

EBCDIC

Abbreviation for Extended Binary Coded Decimal Information Code.

ECL

Abbreviation for Emitter Coupled Logic.

flag

1. Any of various indicators used for identification, e.g., a wordmark.
2. A character that signals the occurrence of some condition, such as the end of a word.
3. Synonymous with mark, sentinel, tag.

glitch

1. Pulse(s) more narrow than the minimum pulse duration criteria.
2. More than one transition between clock edges.

glitch filter

A word must be present for a minimum selectable time before word recognition may occur.

glitch latch

1. First Order: detects pulses too narrow to meet minimum pulse-width criteria.
2. Second Order: detects more than one transition between clock edges.

glitch memory

A separate memory for recording detected glitches.

glitch trigger

A trigger event is generated if a glitch occurs on one or more operator-specified channels. A glitch trigger may be used alone or in combination with word recognition.

handshake

An interlocked sequence of sending and receiving control signals to either establish synchronism between data sender and receiver or otherwise guarantee an orderly flow of data.

hazard

Transient output of a circuit that allows an undesired output value to appear during transition from one state to another.

hexadecimal number system

A number system using 16 as the base. The digits are normally represented by the decimal digits plus the letters A through F. $55F_{16} = 5 \times 16^2 + 5 \times 16^1 + 15 \times 16^0 = 1280 + 80 + 15 = 1375_{10}$.

hold time

The time after the clock that data must be stable at an input in order to be reliably transferred.

hunt word

Allows data communications equipment to break synchronization at the end of text and begin to look for sync words.

impedance

The input impedance is the load presented by the logic analyzer to the signal source. Note: The impedance is represented by the values of a resistor and a capacitor, connected in parallel, which produce an equivalent impedance.

mainframe

Same as central processing unit.

map

Logic states are presented in a matrix so that the user system activity can be identified by unique patterns. One map form portrays each stored logic word as a separate dot, with the vertical position of the dot proportional to the most significant half of the word, and the horizontal position proportional to the least significant half of the word.

memory depth

The maximum number of storable words.

memory width

The number of input channels recorded in memory.

minimum pulse duration (MDP)

The narrowest pulse that can be detected with 100% certainty. MPD = (1/strobe rate) + k, where k is some specified constant.

microprogramming

Control technique used to implement the stored program control function. The typical technique is to use a preprogrammed read-only memory chip to contain several control sequences which normally occur together.

mixed-mode threshold

Different threshold settings on separate channels or groups of channels.

mnemonic symbol

A symbol chosen to assist the human memory (e.g., an abbreviation such as "MPY" for "multiply").

not (false trigger)
All words are recognized except the one specified.

NRZ
Mnemonic for nonreturn to 0. A method of data transmission where data bits are contiguous without interleaved 0's.

octal number system
A number system using 8 as the base. $214_8 = 2 \times 8^2 + 1 \times 8^1 = 4 \times 8^0 = 128 + 8 + 4 = 140_{10}$.

operating system
Software which controls the execution of computer programs and which may provide scheduling, debugging, input/output control, accounting, compilation, storage assignment, data management, and related services.

operation
1. A defined action. The act of obtaining a result from one or more operands in accordance with a rule that completely specifies the result for any permissible combination of operands.
2. The set of such acts specified by such a rule, or the rule itself.
3. The act specified by a single computer instruction.
4. A program step undertaken or executed by a computer (e.g., addition, multiplication, extraction, comparison, shift, transfer). The operation is usually specified by the operator part of an instrument.
5. The even or specific action performed by a logic element.

parallel data
Digital information transferred several bits at a time on multiple lines or channels.

parity bit
A bit added to a transmitted data word to enable error detection. The bit is a 1 or 0 as required to make the total number of 1's (including the parity bit) even in even parity systems, or odd in odd parity systems.

probe
An input device, constructed as a separate unit that transmits the input signal(s) from the circuit under test to the logic analyzer. Probes may be passive or active, single or grouped.

program
Data that defines the operation of a digital system. A computer program (instructions) directs the computer to produce output data that relates to the input data in some very definite way.

PROM
Abbreviation for Programmable Read Only Memory. A ROM that can be programmed after manufacture.

qualifier
The clock qualifier allows data to be sampled selectively for storage into memory. The word qualifier and other trigger criteria defines the trigger.

RAM
Abbreviation for Random Access Memory. A data storage system or device that can be addressed in any sequence, usually a read-write memory. A computer usually uses RAM for data.

resolution
The minimum detectable time between transitions. For a single channel the resolution is +1 sample interval. For two or more channels, this uncertainty is increased by skew.

ROM
Abbreviation for Read Only Memory. A data storage system that provides no means of altering the stored data programmed in the manufacturing process.

RZ

Mnemonic for Return to Zero. A method of serial data transmission where every data bit is followed by a logic 0. The result is a distinct pulse for each logic 1.

sensitivity

Minimum required voltage swing for a signal to be detectable.

serial data

Digital information transferred one bit at a time on a single line or channel.

set-up time

The time before the clock that data must be stable at an input in order to be reliably transferred.

shift register

A data storage system or device that must be addressed sequentially. A shift register moves all bits one position when it receives a clock signal.

skew

The time difference between input channels or between user system strobe (or clock) and input channels.

state table

The logic states of the input data channels at each successive sample time are represented in tabular form. For example, the logic states may be represented in binary, octal, hexadecimal, decimal, ASCII, or mnemonic notation.

store clock

The signal that determines the input data sampling and storage rate. Same as write clock.

strobe

A signal, such as the clock of the system under test, may be used to sample input conditions at some instant in time.

sync word

Allows data communication equipment to become synchronized at the beginning of a string of characters (text).

synchronous data

Digital information transferred in step with a timing signal at the receiving end of the transfer.

synchronous word recognition

For word recognition to occur, the selected word must be true at the edge of a strobe that is phase-related to that word (follows constraints of set-up and hold times).

threshold

The input signal(s) are compared to a preselected threshold voltage level. Any input voltage level that is more positive (or less negative) than the threshold voltage is recorded as a logic high value in memory. Any input voltage level that is less positive (or more negative) than the threshold is recorded as a logic low.

time out

If a trigger event does not occur within a preset time, a trigger event will be forced.

time in

If a trigger event occurs before a preset time, it will not be recognized.

timing diagram

Recorded data states and timing relationships are represented as a pseudo waveform.

trigger output signal

A signal, time-related to the trigger event, that is made available to the user.

trigger event

An event or sequence of events that controls or references the acquisition of logic signal(s). The trigger event may originate from internal circuitry, be derived internally from the signal pattern, or be obtained from an external trigger source.

TTL

Abbreviation for Transistor-Transistor Logic.

word

An ordered set of bits, usually some multiple of 4 — 12, 16, 24, 64, . . . , — that constitutes a functional entity within a digital system. A word may represent a number or a combination of states or some other information. Computers usually process one or two words at a time; a 16-bit computer processes 16-bit words.

word qualifier

Input channel(s) that may optionally be recorded and/or displayed as an additional condition that must be met simultaneously with an incoming word before word recognition can occur.

word recognition

The matching of a preset word due to the presence or absence of that word in the logic signal(s) acquired by the logic analyzer.

Appendix B
Basic Analyzer Applications

CHANNELS	MEMORY BITS PER CHANNEL	APPLICATIONS
1—2		Analyzing serial data transmissions
4		Comparing control signals
8		Comparing complex control signals and analyzing data lines in peripherals
16		Analyzing transactions on microprocessor and other buses and tracing programs
28 or more		Examining total processor operations including addresses, data, and control signals
	1	Storing status information
	16	Recording sequences of logic states
	256	Analyzing timing relationships and long sequences of logic states
	>1024	Making high resolution timing analyses and analyzing serial data transmissions

Appendix C
Comparison of Features

Oscilloscope Features (not available on logic analyzers)	**Benefits**
Actual waveform is displayed	Marginal voltage levels and noise problems can be seen
500 ps/div sweep speed available	Acquire, store, and display data which is too fast for existing logic analyzers to store
2.5 cm/ns storage	Resolve timing differences of 50-100 ps

Logic Analyzer Features (not available on non-digitizing oscilloscopes)	**Benefits**
More than eight channels	See 8-bit (and larger) words
Multi-channel simultaneous data acquisition	No loss of data due to sweep switching
Data pretrigger	See probelms that occur before a trigger event
Display formatting	Hi and lo voltages are displayed in a meaningful way such as hexadecimal or ASCII

Quiz Answer Key

General Purpose Logic Analyzer Concepts Video Tape Quiz

1. c
2. b
3. a
4. d
5. a
6. b
7. c
8. b
9. b

Chapter 1

1. b
2. a
3. b
4. c
5. c
6. c
7. a

Chapter 2

1. c
2. a
3. d
4. b

Chapter 3

1. d
2. c
3. a
4. b

Chapter 4

1. a
2. b
3. c
4. c
5. c
6. c
7. b
8. d

Advanced Logic Analyzer Concepts Video Tape Quiz

1. d
2. c
3. a
4. b
5. d
6. d
7. c
8. d

Chapter 5

1. c
2. b
3. a
4. b

Chapter 6

1. d
2. b
3. d
4. a

Chapter 7

1. c
2. d
3. b
4. c

Tektronix®
COMMITTED TO EXCELLENCE