



**PLEASE CHECK FOR CHANGE INFORMATION
AT THE REAR OF THIS MANUAL.**

**7D02
LOGIC ANALYZER
OPERATORS**

INSTRUCTION MANUAL

**Tektronix, Inc.
P.O. Box 500
Beaverton, Oregon 97077**


070-2918-00
Product Group 57

Serial Number _____

First Printing AUG 1980
Revised MAY 1982

Copyright © 1980 Tektronix, Inc. All rights reserved.
Contents of this publication may not be reproduced in
any form without the written permission of Tektronix, Inc.

Products of Tektronix, Inc. and its subsidiaries are
covered by U.S. and foreign patents and/or pending
patents.

TEKTRONIX, TEK, SCOPE-MOBILE, and  are
registered trademarks of Tektronix, Inc. TELEQUIP-
MENT is a registered trademark of Tektronix U.K.
Limited.

Printed in U.S.A. Specification and price change
privileges are reserved.

FOREWORD

The 7D02 Operators Reference Manual is divided into seven sections and several appendices.

Section 1—The Learning Guide. This section teaches the new 7D02 user enough of the instrument's operation to allow him to perform useful logic analysis after a minimal amount of learning time.

Section 2—General Description. This section is an overview of the 7D02. It contains a hardware functional description, a brief description of the keyboard, and describes the screen display formats.

Section 3—Modes Of Operation. This section discusses each of the five modes of operation of the 7D02, the keys that are permitted in each mode, how each mode is entered and exited, and the various types of data displays. It also provides further information on screen display formats.

Section 4—Programming The 7D02. This section explains each component that can be included in a 7D02 program, data qualification, and the difference between conflicting and contradictory commands. It discusses and illustrates the problem solving abilities of the 7D02, and provides some information on problem solving techniques.

Section 5—Key Encyclopedia. This section is a detailed discussion of the functions and limitations of each front panel key.

Section 6—The Timing Option. This section presents the information a user needs to have if his 7D02 has the Timing Option installed. This section discusses the timing option displays and illustrates some of the more frequently used tests which utilize the 7D02's unique timing data acquisition capabilities.

Section 7—Rules Of Operation And Special Notes. This section describes the run-time operation of the 7D02 (when it is "running", what stops data acquisition, what messages appear), use of the Timing Option "link" bit, the rules of operation for the External Trigger Out, and clock qualification.

The appendices contain the less frequently used operational reference information such as the glossary, shipping information, mainframe controls, system and diagnostic messages, and a foldout containing a diagram of the front panel.

TABLE OF CONTENTS

	Page
FOREWARD	i
LIST OF ILLUSTRATIONS	vi
LIST OF TABLES	vii
OPERATOR'S SAFETY SUMMARY	ix
Section 1 LEARNING GUIDE	1-1
INSTALLING AND REMOVING THE 7D02	1-1
HOW TO CONNECT A 7D02 TO A SYSTEM-UNDER-TEST	1-3
POWER-UP DIAGNOSTICS	1-4
PROGRAM STRUCTURE	1-5
DISPLAY SCREEN CHARACTERISTICS	1-6
7D02 PROGRAMMING EXAMPLES	1-8
CORRECTING PROGRAMMING MISTAKES	1-45
STORING AND DISPLAYING MEMORY WITH IMMEDIATE MODE COMMANDS	1-47
Section 2 GENERAL DESCRIPTION	2-1
7D02 COMPONENTS	2-1
FUNCTIONAL OVERVIEW	2-2
MODES OF OPERATION	2-4
FRONT PANEL FAMILIARIZATION	2-5
DISPLAY SCREEN CHARACTERISTICS	2-7
Section 3 MODES OF OPERATION	3-1
PROGRAMMING MODE	3-2
RUN MODE	3-3
DISPLAY MODE	3-4
IMMEDIATE MODE	3-10
FORMAT MODE	3-12
Section 4 PROGRAMMING CONCEPTS	4-1
EVENTS	4-1
COMMANDS	4-2
COMPOUND EVENTS	4-2
COMPLEMENTED EVENTS	4-3
COMPOUND COMMANDS	4-3
IF-THEN-ELSE	4-4
TESTS	4-5

TABLE OF CONTENTS (cont)

Section 4	PROGRAMMING CONCEPTS	Page
	CONFLICTING COMMANDS	4-6
	CONTRADICTIONARY COMMANDS	4-7
	DATA QUALIFICATION	4-8
	PROBLEM SOLVING	4-9
Section 5	7D02 KEY ENCYCLOPEDIA	5-1
	ACQ MEM	5-2
	COUNTER	5-5
	CURSOR	5-10
	DATA SCROLLING	5-11
	DELETE	5-13
	DISPLAY	5-14
	DON'T CARE	5-15
	ELSE	5-16
	END	5-17
	FORMAT	5-18
	GOTO	5-19
	IMMEDIATE	5-21
	NOT	5-23
	OR	5-24
	PROGRAM	5-25
	QUALIFY	5-26
	START/STOP	5-29
	STORE MEM	5-30
	TRIGGER	5-32
	WD RECOGNIZER	5-36
	X	5-39
	0—9, A—F	5-40
	[]	5-41
Section 6	TIMING OPTION	6-1
	DEFINITIONS (In The Context Of The Timing Option)	6-4
	SYNCHRONOUS OPERATION	6-5
	SYNCHRONOUS OPERATION EXAMPLES	6-5
	SUMMARY OF SYNCHRONOUS MODE	6-15
	ASYNCHRONOUS OPERATION	6-16

TABLE OF CONTENTS (cont)

Section 6	TIMING OPTION	Page
	ASYNCHRONOUS OPERATION EXAMPLES	6-22
	SUMMARY OF ASYNCHRONOUS MODE	6-34
	TIMING OPTION CLOCKING MODE SELECTION	6-35
	TIMING OPTION DISPLAYS	6-35
	TIMING OPTION PARAMETERS	6-41
	HARDWARE FUNCTIONAL DIAGRAMS	6-45
Section 7	7D02 RULES OF OPERATION	7-1
	SPECIAL NOTES	7-14
Appendix A	STATUS AREA MESSAGES	A-1
Appendix B	SHORTCUTS	B-1
Appendix C	7000-SERIES MAINFRAME SET-UP CONDITIONS	C-1
Appendix D	SELF-TEST PROCEDURES AND ERROR MESSAGES	D-1
Appendix E	OPTIONS AND SPECIFICATIONS	E-1
Appendix F	7D02 PERFORMANCE CHECKOUT PROCEDURE	F-1
Appendix G	GLOSSARY OF TERMS	G-1
Appendix H	7D02 ASCII DISPLAY CHARACTERS	H-1
Appendix I	7D02 FRONT PANEL	I-1

LIST OF ILLUSTRATIONS

Figure No.		Page
1-1	Installing the 7D02 in a 7000-Series oscilloscope	1-2
1-2	Location of the 7D02 release latch	1-2
1-3	PM-102 Personality Module plug and prototype socket	1-3
1-4	Successful power-up verification display	1-5
1-5	Typical power-up verification display with a failed component	1-6
1-6	7D02 screen display components	1-7
1-7	Program display concepts	1-11
1-8	Trigger locations with respect to acquired data	1-12
1-9	How the DELETE command works	1-46
2-1	7D02 front panel	2-5
2-2	7D02 screen display areas	2-7
2-3	7D02 parameters	2-8
3-1	7D02 screen display areas	3-1
4-1	State transition diagram	4-10
4-2	State transition diagram translated into 7D02 syntax	4-11
6-1	7D02 timing diagram	6-6
6-2	7D02 timing state table	6-7
6-3	7D02 Main acquisition memory display	6-8
6-4	Main section mnemonic display	6-10
6-5	Main section absolute display	6-11
6-6	Main section mnemonic display	6-13
6-7	Timing state table display	6-14
6-8	Asynchronous Timing Option event recognizer relationships	6-17
6-9	Glitch triggering	6-18
6-10	Asynchronous timing diagram display	6-23
6-11	Asynchronous timing state table display	6-24
6-12	Timing diagram	6-31
6-13	Timing state table	6-32
6-14	Timing diagram display	6-36
6-15	Glitch definition and display	6-37
6-16	Display of trigger relative to the cursor	6-38
6-17	Timing Option state table display	6-39
6-18	Apparent data movement on the display screen	6-41
6-19	Expanded Timing Option-Related menus	6-42
6-20	Composite Timing Option functional block diagram	6-45
6-21	Functional block diagram for synchronous operation	6-46
6-22	Functional block diagram for asynchronous operation	6-47

LIST OF ILLUSTRATIONS (cont)

Figure No.		Page
D-1	Successful power-up display	D-2
D-2	Unsuccessful power-up display	D-3
D-3	Diagnostic monitor module menu	D-4
D-4	Individual module test instruction display	D-5
D-5	Individual module test result display	D-6
D-6	Module test "LOOPING ENABLED" display	D-7
D-7	Module test result display when "LOOPING ENABLED" on sub-test 4	D-8
D-8	Signature exerciser menu	D-9
D-9	Typical signature exerciser display	D-10
E-1	Counter and delay characteristics	E-7
F-1	Screen display for the key-independence test	F-3

LIST OF TABLES

Table No.		Page
6-1	Summary of synchronous mode	6-15
6-2	Summary of asynchronous mode	6-34
E-1	Electrical characteristics	E-1
E-2	Environmental characteristics	E-9
E-3	Physical characteristics	E-9
H-1	7D02 ASCII display characteristics	H-1

OPERATORS SAFETY SUMMARY

The general safety information in this part of the summary is for both operating and servicing personnel. Specific warnings and cautions will be found throughout the manual where they apply, but may not appear in this summary.

TERMS

In This Manual

CAUTION statements identify conditions or practices that could result in damage to the equipment or other property.

WARNING statements identify conditions or practices that could result in personal injury or loss of life.

As Marked on Equipment

CAUTION indicates a personal injury hazard not immediately accessible as one reads the marking, or a hazard to property including the equipment itself.

DANGER indicates a personal injury hazard immediately accessible as one reads the marking.

SYMBOLS

In This Manual



This symbol indicates where applicable cautionary or other information is to be found.

As Marked on Equipment



DANGER — High voltage.



Protective ground (earth) terminal.



ATTENTION — refer to manual.

Power Source

This product is intended to operate from a power source that will not apply more than 250 volts rms between the supply conductors or between either supply conductor and ground. A protective ground connection by way of the grounding conductor in the power cord is essential for safe operation.

Grounding the Product

This product is grounded through the grounding conductor of the power cord. To avoid electrical shock, plug the power cord into a properly wired receptacle before connecting to the product input or output terminals. A protective ground connection by way of the grounding conductor in the power cord is essential for safe operation.

Danger Arising From Loss of Ground

Upon loss of the protective-ground connection, all accessible conductive parts (including knobs and controls that may appear to be insulating) can render an electric shock.

Use the Proper Power Cord

Use only the power cord and connector specified for your product.

Use only a power cord that is in good condition.

For detailed information on power cords and connectors, see maintenance section.

Refer cord and connector changes to qualified service personnel.

Use the Proper Fuse

To avoid fire hazard, use only the fuse of correct type, voltage rating and current rating as specified in the parts list for your product.

Refer fuse replacement to qualified service personnel.

Do Not Operate in Explosive Atmospheres

To avoid explosion, do not operate this product in an explosive atmosphere unless it has been specifically certified for such operation.

Do Not Remove Covers or Panels

To avoid personal injury, do not remove the product covers or panels. Do not operate the product without the covers and panels properly installed.



2918-1

The 7D02 Logic Analyzer.

Section 1

7D02 LEARNING GUIDE

TABLE OF CONTENTS

INTRODUCTION

INSTALLING AND REMOVING THE 7D02

- Installing The 7D02 In The Mainframe
- Selecting The 7000-Series Mainframe Controls
- Removing The 7D02 From The Mainframe

HOW TO CONNECT A 7D02 TO A SYSTEM-UNDER-TEST

- How To Connect A Personality Module To A Logic Analyzer
- How To Connect A Personality Module To A System-Under-Test

POWER-UP DIAGNOSTICS

PROGRAM STRUCTURE

DISPLAY SCREEN CHARACTERISTICS

7D02 PROGRAMMING EXAMPLES

- Introduction
- Format Of The Examples
- Assumptions Made In The Examples
- Documentation Conventions Used In The Manual
- Word Recognizer Examples
 - Example 1—Simple Word Recognition—
 - User-defined Data And Address Words
 - Getting Back To Program Mode
 - Example 2—Simple Word Recognition—
 - A Shorter Way
 - Example 3—Sequential Word Recognition
 - (Arming Function)
 - Example 4—Sequential Word Recognition
 - (Count Data Word Occurrences)
 - Example 5—Sequential Word Recognition
 - With Else Clause
 - Example 6—Simultaneous Event Recognition
- Counter Examples
 - Example 7—Count Time Interval
 - Example 8—Trigger If Event Occurs Within Time "Window"
 - Example 9—Counter Tips
- Test For The Absence Of An Event
 - Example 10—NOT Word Recognition
- Qualified Data Storage
 - Example 11—Block Qualification
 - Example 12—Qualify Command
 - Example 13—Block Qualification And Qualify Command Combined
 - Example 14—Start And Stop Qualification

CORRECTING PROGRAMMING MISTAKES

STORING AND DISPLAYING MEMORY WITH IMMEDIATE MODE COMMANDS

7D02 LEARNING GUIDE

INTRODUCTION

This section tells how to install the 7D02 and describes its use in the most frequently encountered logic/data analyses. These analyses are performed by using the 10 front panel programming keys to create a program. Most of these keys are explained through the use of examples found in the Learning Guide. Each example illustrates a type of logic analysis that you may want to perform.

INSTALLING AND REMOVING THE 7D02

The 7D02 Logic Analyzer is designed to occupy a three-compartment segment of any TEKTRONIX non-storage 7000-Series Oscilloscope mainframe.

Due to the unique requirements of the 7D02, best results will be obtained with 7603 or 7704A mainframes. Use of storage mainframes is not recommended as these CRTs are optimized for storage applications only.

Installing the 7D02 in the Mainframe

To install the 7D02 in the mainframe, first make sure that there is no power to the mainframe. Then align the 7D02 tracks with the rails of a vertical and left horizontal plug-in compartment. Gently slide the 7D02 into the mainframe; then push firmly to lock the 7D02 front panel flush with the mainframe. See Figure 1-1.

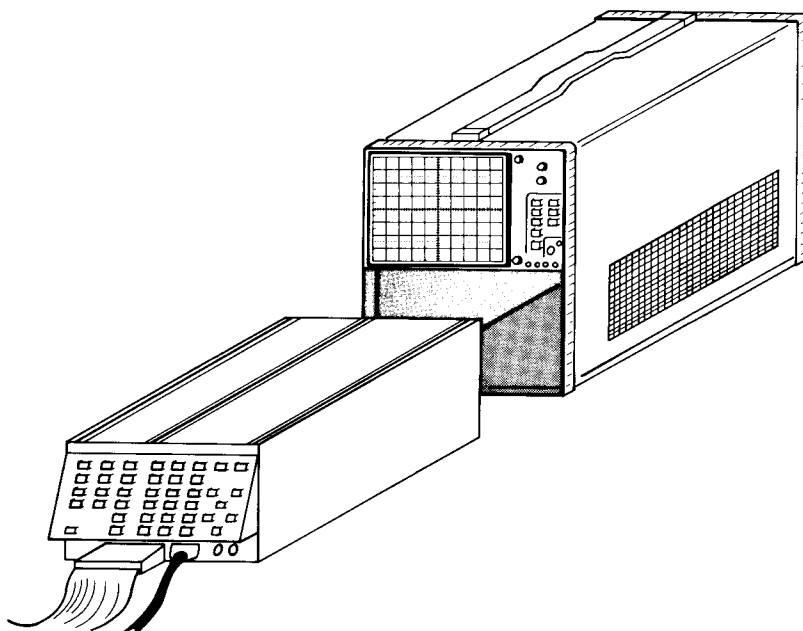
NOTE

Some rack-mounted mainframes have vertically-mounted rods in the front of the plug-in compartments; these rods interfere with the installation of the 7D02. Detach these rods by removing the mounting screws at each end before installing the 7D02.

Selecting the 7000-Series Mainframe Controls

Each 7000-Series mainframe has a number of front panel controls which may need to be adjusted if the 7D02 is to operate properly. See Appendix C for a list of adjustments required for your mainframe. In general, for 7000-Series mainframes:

1. Select horizontal mode as the rightmost mainframe compartment occupied by the 7D02;
2. Select vertical mode as the leftmost mainframe compartment occupied by the 7D02;
3. Adjust intensity and focus (some mainframes have separate intensity controls for each horizontal compartment); and
4. If the 7D02 is installed in a storage mainframe, the mainframe should be set to the non-store mode.

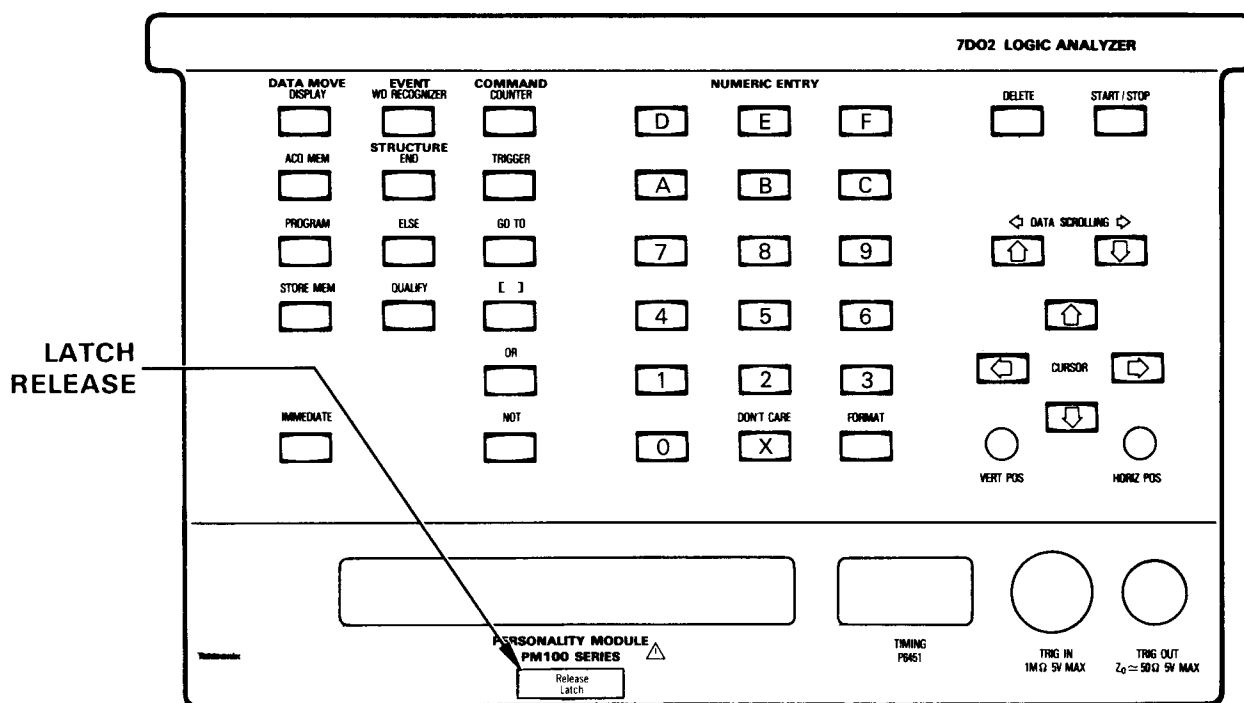


2918-2

Figure 1-1. Installing the 7D02 in a 7000-series oscilloscope.

Removing the 7D02 From the Mainframe

To remove the 7D02, first make sure that there is no power to the mainframe. Disconnect all probes and connectors from the front panel. Then grasp the release latch at the bottom center of the 7D02 and pull the 7D02 out of the mainframe. See Figure 1-2.



2918-3

Figure 1-2. Location of the 7D02 release latch.

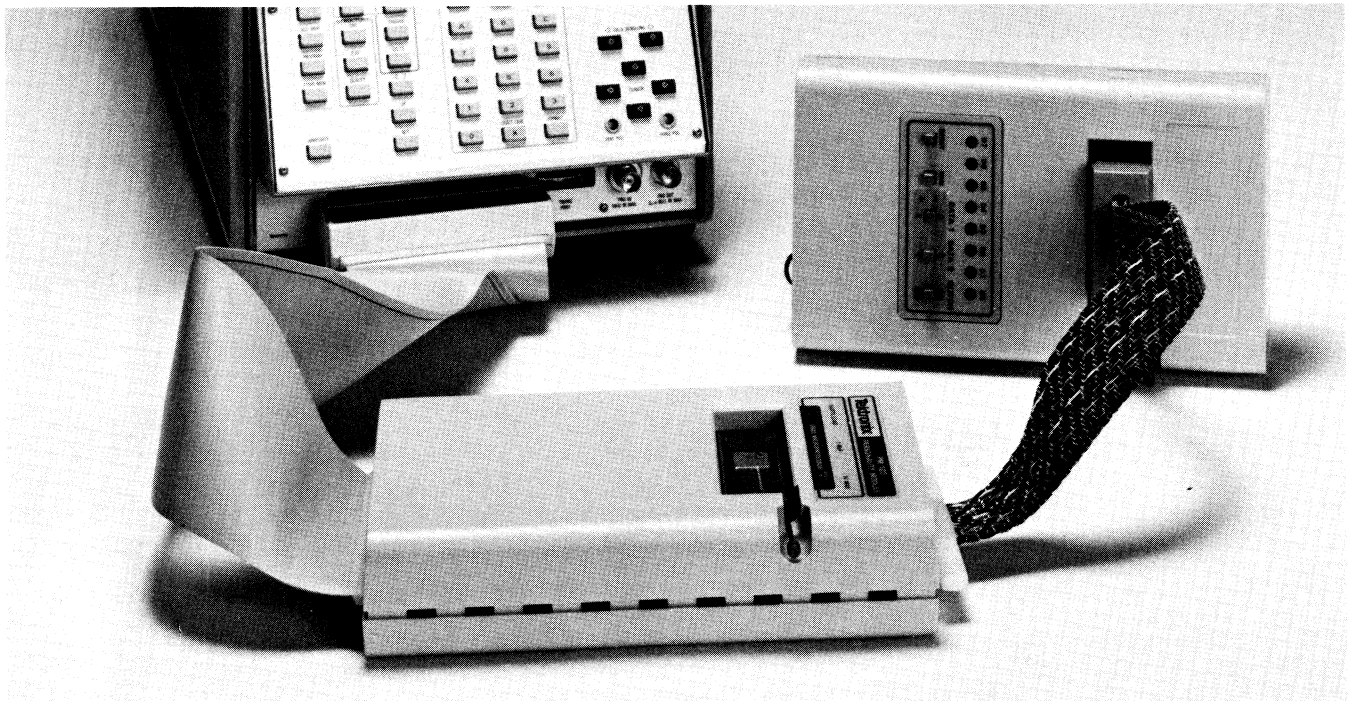
HOW TO CONNECT A 7D02 TO A SYSTEM-UNDER-TEST

How To Connect A Personality Module To A Logic Analyzer

CAUTION

Always turn the mainframe power switch OFF before connecting or disconnecting the Personality Module.

Turn the mainframe power switch to the OFF position. Insert the plug of the Personality Module into the designated socket on the front of the 7D02 Logic Analyzer. Make sure that the label is facing up. The shape of the plug makes it difficult to insert it incorrectly. See Figure 1-3.



2918-4

Figure 1-3. PM-102 Personality Module plug and prototype socket.

How To Connect A Personality Module To A System-Under-Test

CAUTION

1. *Always turn off power to the system-under-test (s.u.t.) before connecting or disconnecting the Personality Module Plug.*
2. *Before handling the microprocessor in your test circuit, ground yourself to the 7D02 mainframe and to the system-under-test to discharge any static electricity you may have built up.*

Whether you are using the General Purpose Personality Module or a microprocessor-specific Personality Module, initial connections of the 7D02 to the system-under-test (s.u.t.) should be made according to the directions in the manual for that Personality Module.

Do not remove the processor when using the General Purpose Personality Module or when using a low-profile DIP-clip.

Turn off the power to the system-under-test. Ground yourself to discharge static electricity. Remove the microprocessor from your test circuit and insert it into the Zero Insertion Force (ZIF) socket on the Personality Module. Be sure to insert it correctly, with pin 1 of your microprocessor aligned with the pin 1 indicator on the ZIF socket.

Now insert the Personality Module plug into the socket on the system-under-test. Again, make sure to insert the plug correctly.

CAUTION

Read the appropriate Personality Module Instruction Manual for specific instructions on installation and use of the PM-100 Series probe that you are using.

After you have finished connecting the Personality Module to the 7D02 and to the system-under-test, turn on the 7D02 mainframe power switch and power up the system-under-test.

NOTE

Most microprocessors will be halted until power is applied to the 7D02.

POWER-UP DIAGNOSTICS

The power-up verification routines run automatically each time the 7D02 is powered up. (See the 7D02 Service manual for more information on power-up diagnostics.)

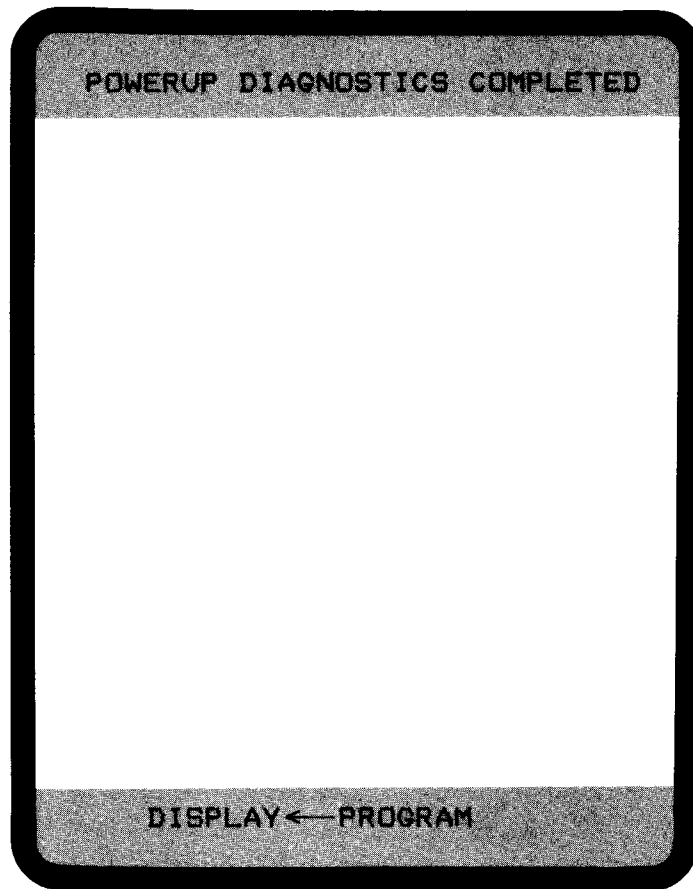
If the diagnostic checks are successful, the display shown in Figure 1-4 should appear.

NOTES

1. *The Expansion Option and the Timing Option checks will not be listed on the screen unless the appropriate hardware is installed.*
2. *It is normal for the display to flash check patterns during power-up diagnostics.*

If one or more of the power-up verification checks fails, the diagnostics program assumes that you will want to run additional diagnostic checks. To run them, you must press the X key. (See Figure 1-5.) The requirement that you must press the X key allows you to note which checks failed, and then to decide which check you want to run first. These additional diagnostic checks are described in Appendix D.

If neither of the above displays appears, or if the 7D02 screen appears unreadable in any way, check your 7000-Series mainframe control settings against those given for it in Appendix C.



2918-5

Figure 1-4. Successful power-up verification display.

PROGRAM STRUCTURE

A 7D02 program consists of one or more TESTS (up to a total of four) plus, optionally, one block Qualification clause which may be placed before or after any TEST. Basically, a TEST consists of:

```

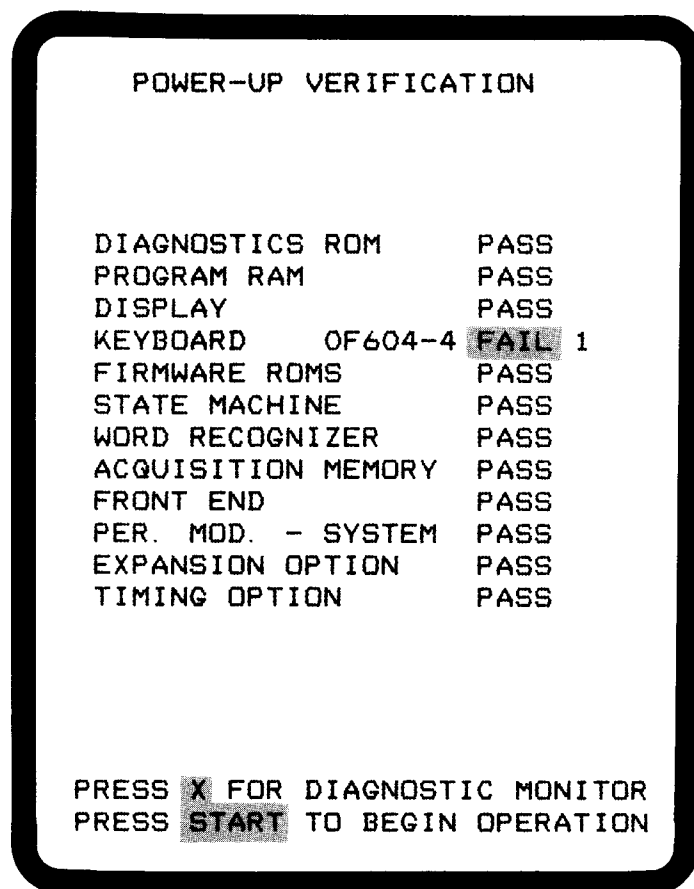
TEST #
IF event clause
THEN DO command clause
OR IF event clause
THEN DO command clause } Optional, may occur any number of times.
ELSE command clause (optional)
END TEST #

```

An event clause describes a condition which at any instant is either TRUE or FALSE. It may be either word recognition on the s.u.t. bus, or a counter reaching a previously specified value, or a negation of either of those events, or a combination of any or all of the preceding.

A command clause specifies what action is to be taken by the 7D02 when an event occurs.

An ELSE clause specifies what action is to be taken if none of the previously defined events in the TEST are found.



2918-6

Figure 1-5. Typical power-up verification display with a failed component.

Only one TEST in the program is active at a time. Pressing the START key is an implied GOTO TEST #1. Other TESTS are executed as the result of a GOTO from some other TEST, or as the result of an IMMEDIATE GOTO <test #> command.

All conditions, i.e., events, in a single test are looked for simultaneously.

DISPLAY SCREEN CHARACTERISTICS

The 7D02's screen is divided into three parts. (See Figure 1-6.) At the top of the screen is a two-line band of inverse video, called the Status Area, in which statistics, error, prompting, run-time, and diagnostic messages appear.

There is a similar band of inverse video at the bottom of the screen, called the Immediate mode area, which is used to display Immediate Mode commands. (Immediate Mode commands will be explained later.)

The central portion of the screen is used to display the test program, the FORMAT mode display, or the acquired data or STORE MEM contents. If the program display exceeds about 20 lines (depending on the size of the Immediate Mode area), not all of the program is visible at once. The program is scrolled automatically as the cursor is moved about in it.

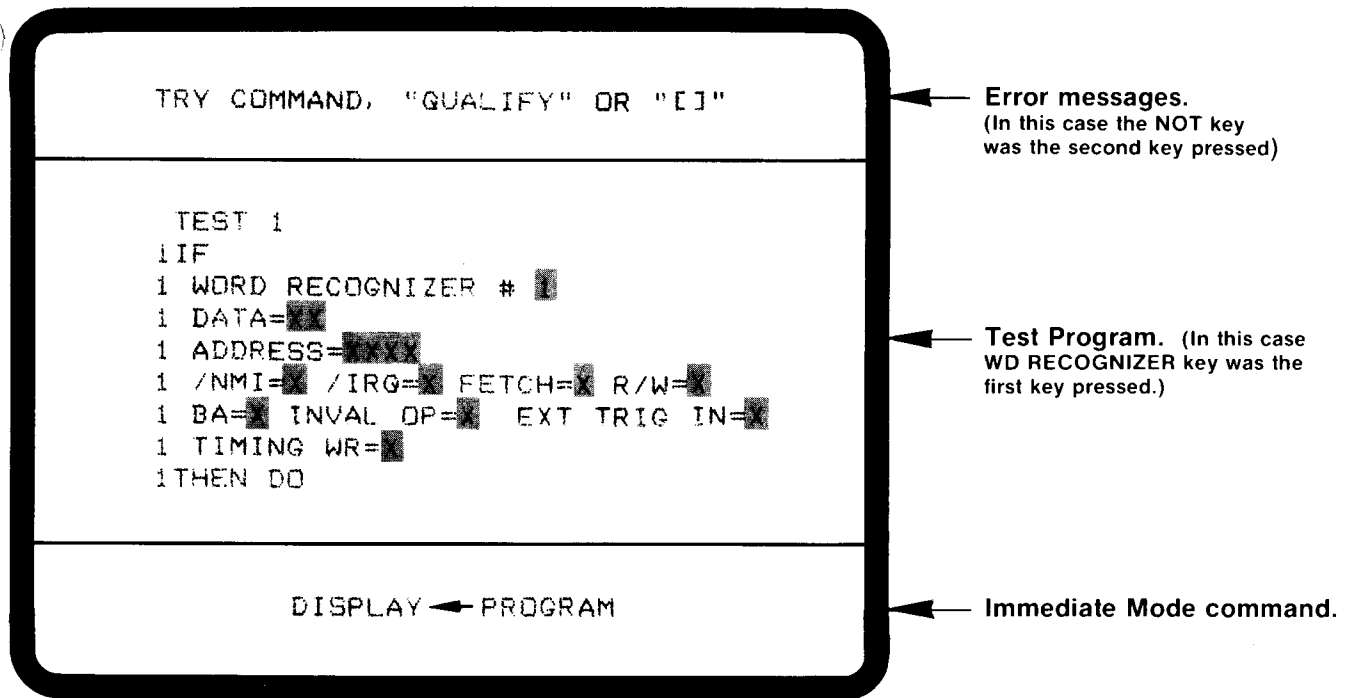


Figure 1-6. 7D02 screen display components.

2918-7

A blinking cursor also appears on the screen. This cursor can be positioned on a character which can be changed, at the end of the program, or in the Immediate Mode area during data displays and IMMEDIATE command sequences. In order to change a program entry, first move the cursor to the selected field using the cursor movement keys. If the field is a "Menu", then the possible choices for it will be listed on the lines following the cursor position. The desired choice can be selected with the numeric keypad. If it is a numeric field (such as a Word Recognizer value), then the desired value can be entered.

NOTES

1. In this manual, the word "field" is to denote any numeric value or menu selection displayed in inverse video.
2. Throughout this manual all examples, unless otherwise specified, are 6800/6802 displays. Refer to the Instruction manual for your Personality Module for a detailed description of the mnemonics displayed.
3. To decrease the visibility of a CRT's graticule lines, use of a green screen filter is recommended. In addition, a very low level of graticule illumination can significantly decrease their visibility in the central area of the display.

7D02 PROGRAMMING EXAMPLES

Introduction

Because of the versatility of the 7D02's keystroke programming, it is impossible to describe every type of logic analysis measurement procedure that you can make. However, some of the most commonly performed tasks are discussed here. Many of these tasks can be repeated or combined to create complex measurement tests.

For reference purposes, a fold-out illustration of the 7D02 front panel is included as Appendix I.

Format of the Examples

Each example has the following format:

Example—Descriptive Task Name

Program Definition

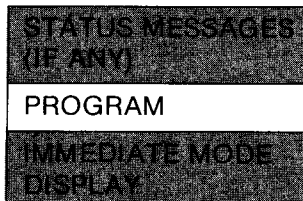
What this program does.

Program Explanation

A list of keys pressed, with a description of what occurs on the screen, and other helpful information explaining what this program does.

Used in the manual to indicate that information on the screen has changed. The test is a repeat of the previous screen display with the changes included.

Program Displays



Screen
Display

Run-time Displays (Optional)

Explanation of Run-time Displays

An explanation of visible changes in the error/status display area.

Data Displays (Optional)

Data Display Explanations (Optional)



An explanation of what the data acquisition display could look like.

Assumptions Made in the Examples

- In the following examples, several assumptions are made:
1. A 6802 Personality Module is used,
 2. The Timing Option is installed, and
 3. All user-selectable radices are hexadecimal.

Documentation Conventions Used in the Manual

The following documentation conventions are used:

1.  — Represents an inverse video field;
2.  — Represents a key to be pressed.

Word Recognizer Examples

Example 1—Simple Word Recognition—User-defined Data and Address Words

Program Definition

This program is a simple word recognition. It looks at the bus transactions surrounding data F6.

Program Displays

```
TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=F6
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1
```

2918-8

Program Explanation

WD RECOGNIZER key is pressed. The 7D02 provides the first nine lines of the program.

Figure 1-7 explains some of the basic program display concepts.

The blinking cursor is positioned on the first digit of the DATA field.

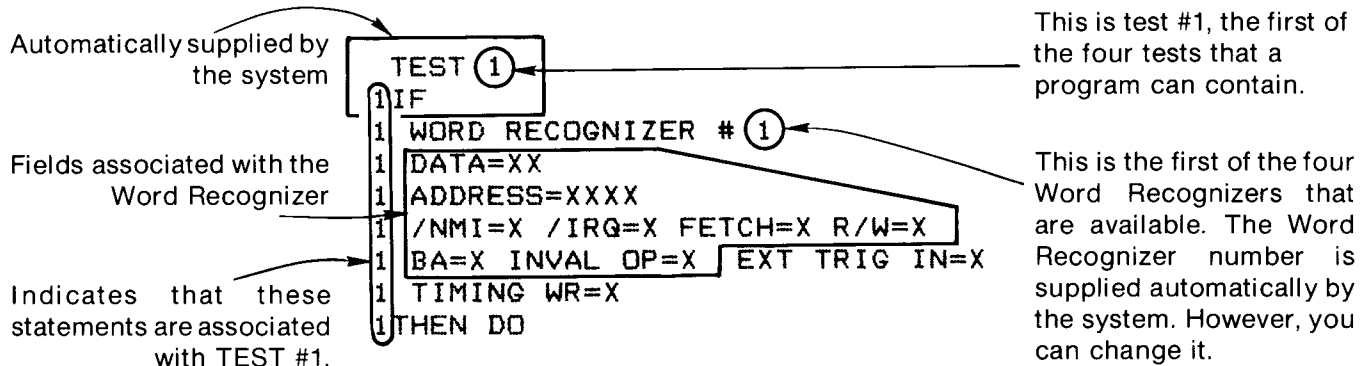
You can now enter any hex digit to replace the X; i.e., DON'T CARE, characters. The DATA= field will accept two hexadecimal digits. In this example those digits (F6) define the value on the system data bus the 7D02 is to trigger on.

The next line, ADDRESS=, is used to specify the address that is to be recognized as it moves on the system-under-test address data bus. In this example it's all DON'T CARE digits, meaning that the address is to be ignored by this Word Recognizer. F6 is to be found regardless of its address.

The next two lines on the display contain six control line parameters (/NMI, /IRQ, FETCH, R/W, BA, INVAL OP) and the external trigger input (EXT TRIG IN). The control line parameters and screen display (/NMI, /IRQ, etc.) are Personality Module-dependent. Changing the DON'T CARE to a 0 or a 1 indicates that that control line is to be used as part of the event. In this example none of the control lines are used.

The presence or absence of the external signal (specified by EXT TRIG IN) is used in the same manner as the control lines.

The last parameter, TIMING WR, is only displayed if the Timing Option is installed in the 7D02. This parameter is discussed in Section 6.



2918-9

Figure 1-7. Program display concepts (a WORD RECOGNIZER key was pressed).

NOTE

All fields of the Word Recognizer definition are ANDed together. A field with a DON'T CARE in it is considered as always TRUE.

TRIGGER key is pressed. The 7D02 provides the next 7 lines of the program. The cursor is positioned at the 0-MAIN field in the new block of program lines; the choices you have available for that parameter are listed below 0-MAIN. You can now specify whether you want to trigger the main memory or the Timing Option memory. Main memory is the default value and is used here. Note that this choice will not appear if there is no Timing Option installed.

NOTE

A menu is expanded only when the blinking cursor is placed on the inverse video portion of a field; in this case, 0-MAIN.

```

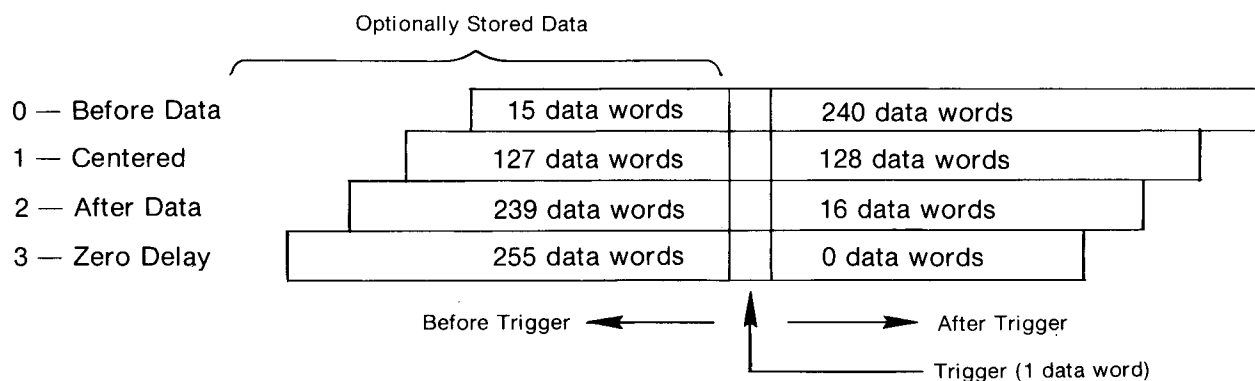
TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=F6
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 TRIGGER 0-MAIN
1      0 MAIN
1      1 TIMING
1      0-BEFORE DATA
1      0-SYSTEM UNDER TEST CONT.
1      0-STANDARD CLOCK QUAL.
1 OR IF
1

```

2918-10

When you position the cursor in the next field (0-BEFORE DATA), the choices for 0-MAIN disappear and the choices for 0-BEFORE DATA are presented.

You can now specify the position of the trigger in relation to the acquired data. See Figure 1-8 for a graphic explanation of the four trigger locations.



2918-11

Figure 1-8. Trigger locations with respect to acquired data.

Moving the cursor again places it at 0-SYSTEM UNDER TEST CONT. Here you specify the action you want the 7D02 to take once it has found the data field, triggered, and acquired data.

0-SYSTEM UNDER TEST CONT indicates that the microprocessor of the system-under-test is to continue running after data is acquired.

1-SYSTEM UNDER TEST HALT would indicate that the microprocessor is to halt after data is acquired.

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=F6
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 TRIGGER 0-MAIN
1     0-BEFORE DATA
1     0-SYSTEM UNDER TEST CONT.
1     0-STANDARD CLOCK QUAL.
END TEST 1

```

2918-12

Moving the cursor to the next menu item allows you to specify the clock to be used for determining clock cycles. 0-STANDARD CLOCK QUAL says that the clock is to be defined by the Personality Module; i.e., the microprocessor. 1-USER CLOCK QUAL says that you want to define the clock using the microprocessor clock as the basic signal source. (Proper disassembly is NOT guaranteed in this case.)

This example uses the default values 0-SYSTEM UNDER TEST CONT and 0-STANDARD CLOCK QUAL. (The parameters for 1-USER CLOCK QUAL will be discussed in another section of the manual.

NOTE

The cursor can be positioned at any field at any time. That is, fields can be specified or changed in any sequence.

END key is pressed. The 7D02 replaces:

1 OR IF

1

with the END statement for the TEST.

START key is pressed. The 7D02 begins monitoring for the data value.

Run-time Displays

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=F6
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

DISPLAY←PROGRAM

2918-13

Explanation of Run-time Displays

When the **START** key is pressed, the screen display changes from:

to:

RUNNING

PRESENT TEST=1

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=F6
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

DISPLAY←PROGRAM

2918-14

Then, when the trigger statement is executed, the screen display changes to:

RUNNING

MAIN TRIGGER TEST = 1

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=F6
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

DISPLAY ← PROGRAM

2918-15

Data Displays

Explanation of Data Displays

After the trigger event occurs, i.e., F6 is found, and the data acquisition is complete, the screen display changes from the program to the acquired data (in mnemonic format).

```

CTR1=00000 EVT   TRIG LOC = 015
CTR2=00000 EVT   TRIG IN TEST 1
LINE ADDRESS OPERATION /IRQ/NMI
015T—F835—F6 READ —11
016 F82C INC $0006 11
017 F82D 00 READ 11
018 F82E 06 READ 11
019 0006 01 READ 11
020 0006 02 WRITE 11
021 F82F BEQ $F865 11
022 F830 34 READ 11
023 F831 TST $0004 11
024 F832 00 READ 11
025 F833 04 READ 11
026 0004 80 READ 11
027 F834 BNE $F82C 11
028 F835 F6 READ 11
029 F82C INC $0006 11
030 F82D 00 READ 11
031 F82E 06 READ 11

```

```

DISPLAY ← ACQMEM 0-MAIN
              0 MAIN
              1 TIMING

```

1-MNEMONIC

2918-16

CTR1=00000 EVT TRIG LOC = 015			
CTR2=00000 EVT TRIG IN TEST 1			
LINE	ADDRESS	DATA	CNTL
015T	F835	F6	R11
016	F82C	7C	F11
017	F82D	00	R11
018	F82E	06	R11
019	0006	01	R11
020	0006	02	W11
021	F82F	27	F11
022	F830	34	R11
023	F831	7D	F11
024	F832	00	R11
025	F833	04	R11
026	0004	80	R11
027	F834	26	F11
028	F835	F6	R11
029	F82C	7C	F11
030	F82D	00	R11
031	F82E	06	R11
032	0006	02	R11
033	0006	03	W11
DISPLAY ← ACQMEM 0-MAIN			
0-ABSOLUTE			

2918-17

NOTE

Although the 7D02 monitors and tests for up to six control lines, only two (/IRQ and /NMI) are explicitly displayed by the 6802 Personality Module in mnemonic mode.

Up to 256 data words may be stored in acquisition memory. In this example, the trigger word and at least 240 other data words will be stored. The total number of words stored will be trigger word plus 240 data words plus the number of data words (up to 16) stored before the trigger word.

Press the SCROLLING keys to see data that is not on the screen. The trigger word is at location 015 and is indicated by a T next to the location number and a horizontal bar, as shown in the example.

Moving the cursor to the 1-MNEMONIC field changes the Immediate mode display to:

```

DISPLAY ← ACQMEM 0-MAIN
1-MNEMONIC
0 ABSOLUTE
1 MNEMONIC

```

The same acquired data can be presented as an absolute display by pressing the 0 key.

Getting Back to Program Mode

Once you have finished with the acquired data and you want to run another program, you need to change the operating mode of the 7D02 from Display mode to Program mode. You do this by pressing the following key sequence:



At this point the screen display will change from the acquired data to the program which was run to acquire the data.

Now you can delete the current program, then enter a new program. Just move the cursor to the end of the program, then press the **DELETE** key. Each keystroke will delete one or more lines of the current program, starting with the last line of the program. The **DELETE** key will repeat if you keep the key pressed down. More information on the use of the DELETE command is presented later in this section.

NOTE

*If you want to run the same program again, just press the **START** key. You may be in either Display mode or Program mode when you press the **START** key.*

Example 2—Simple Word Recognition—A Shorter Way

Program Description

This is a "short cut" way of creating the same basic program as shown in Example 1.

Program Displays

```

TEST 1
11F
1 WORD RECOGNIZER # 1
1 DATA=
1 ADDRESS=
1 /NMI= /IRQ= FETCH= R/W=
1 BA= INVAL OP= EXT TRIG IN=
1 TIMING WR=
1 THEN DO
1 TRIGGER 0-MAIN
1      0 MAIN
1      1 TIMING
1      0-BEFORE DATA
1      0-SYSTEM UNDER TEST CONT.
1      0-STANDARD CLOCK QUAL.
END TEST 1

```

Program Explanation

TRIGGER key is pressed. The 7D02 provides an entire test program.

All of the previous values for the Word Recognizer and the trigger are assumed.

Now press the **START** key.

NOTE

When an item with numeric fields is deleted (for instance, a Word Recognizer), its value is not set back to default. When it is used again, the previously entered value will still be there.

2918-18

Example 3—Sequential Word Recognition (Arming Function)

Program Definition

This program monitors the system-under-test for Word Recognizer #1. When Word Recognizer #1 comes TRUE, the bus is monitored for Word Recognizer #2. When it comes TRUE, the trigger occurs and data is acquired.

Program Displays

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=
1 ADDRESS=XXXX
1 /NMI= /IRQ= FETCH= R/W=
1 BA= INVAL OP= EXT TRIG IN=
1 TIMING WR=
1THEN DO
1 GOTO
END TEST 1
TEST 2
2IF
2 WORD RECOGNIZER # 2
2 DATA=
2 ADDRESS=
2 /NMI= /IRQ= FETCH= R/W=
2 BA= INVAL OP= EXT TRIG IN=
2 TIMING WR=
2THEN DO
2 TRIGGER -MAIN
2 -BEFORE DATA
2 -SYSTEM UNDER TEST CONT.
2 -STANDARD CLOCK QUAL.
END TEST 2

```

Program Explanation

WD RECOGNIZER key is pressed. Define the first Word Recognizer value (17) to be found on the data bus. Once this value has been found, then the 7D02 will look for the next specified value. Naturally, you can specify any value, but 17 was chosen here.

WD RECOGNIZER key is pressed again. Now you can define the second Word Recognizer value to be found in the sequential testing. In this example, data of 7D and address F831.

NOTE

You do not have to press either the **GOTO** or **END** keys. The system will supply them automatically when you press the next **WD RECOGNIZER** key.

TRIGGER key is pressed and any changes to the previous trigger parameters are entered. In this case there are no changes to be made.

START key is pressed, the 7D02 starts monitoring the s.u.t. data bus for Word Recognizer #1. The 7D02 automatically replaces the OR IF prompt with END TEST 2.

Data Displays

```

CTR1=00000 EVT   TRIG LOC = 015
CTR2=00000 EVT   TRIG IN TEST 2
LINE ADDRESS OPERATION /IRQ/NMI
015T—F831—TST      $0004—11
016  F832    00 READ      11
017  F833    04 READ      11
018  0004    5F READ      11
019  F834    BNE      $F82C 11
020  F835    F6 READ      11
021  F82C    INC      $0006 11
022  F82D    00 READ      11
023  F82E    06 READ      11
024  0006    17 READ      11
025  0006    18 WRITE     11
026  F82F    BEQ      $F865 11
027  F830    34 READ      11
028  F831    TST      $0004 11
029  F832    00 READ      11
030  F833    04 READ      11
031  0004    5F READ      11
032  F834    BNE      $F82C 11
033  F835    F6 READ      11
DISPLAY←ACGMEM 0—MAIN
1—MNEMONIC

```

Explanation of Data Displays

This is a standard mnemonic display of the acquired data.

The run-time information now shows that the Main trigger occurred in TEST 2. If you have multiple tests that can cause the Main trigger to occur, this message tells you which one actually occurred.

2918-20

Example 4—Sequential Word Recognition (Count Data Word Occurrences)**Program Definition**

In TEST 1, the 7D02 monitors the s.u.t. bus for a value (DATA=34 in this example) on the data bus, counts 23 occurrences of that event, then transfers to TEST 2. TEST 2 monitors the bus for the data value F6. After the twelfth occurrence of F6, the program transfers to TEST 3. In TEST 3 the 7D02 looks for the first occurrence of the data value FF, then triggers.

Program Displays

2918-21

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=34
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1OR IF
1 COUNTER # 1 = 00023 0-EVENTS
1THEN DO
1

```

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=34
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1OR IF
1 COUNTER # 1 = 00023 0-EVENTS
1THEN DO
1 GOTO 2

```

2918-22

Program Explanation

WD RECOGNIZER key is pressed and the parameters defined. The data word is specified to be 34. This word must be found on the s.u.t. data bus and counted the specified number of times before the 7D02 will look for the next data word.

COUNTER key is pressed. The default values are correct for this example: EVENTS are to be counted and the COUNTER is to be INCREMENTED. (COUNTER is used here as a command.)

COUNTER key is pressed again and the number of occurrences (23) of the first event specified via the numeric keypad. Note that the counter field defaults to 00002. (The COUNTER key is used here to define an event.)

GOTO key is pressed. Once the TEST 1 counter has reached the specified value, go to TEST 2. This provides for the sequential testing of events. Without the GO TO, both events (Word Recognizer 1 and Word Recognizer 2) would be tested for simultaneously.

NOTE

The ONLY way to transfer program execution from one test to another is via the GO TO command.

END key is pressed to specify the end of TEST 1. The OR IF prompt is replaced by END TEST 1.

```

END TEST 1
TEST 2
2IF
2 WORD RECOGNIZER # 2
2 DATA=FF
2 ADDRESS=XXXX
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2THEN DO
2 COUNTER # 2 0-EVENTS
2 0-INCREMENT
2OR IF
2 COUNTER # 2 = 00012 0-EVENTS
2THEN DO
2 GOTO 3
END TEST 2
TEST 3
3IF
3 WORD RECOGNIZER # 3
3 DATA=FF
3 ADDRESS=XXXX
3 /NMI=X /IRQ=X FETCH=X R/W=X
3 BA=X INVAL OP=X EXT TRIG IN=X
3 TIMING WR=X
3THEN DO
3 TRIGGER 0-MAIN
3 0-BEFORE DATA
3 0-SYSTEM UNDER TEST CONT.
3 0-STANDARD CLOCK QUAL.
END TEST 3

```

2918-23

WD RECOGNIZER **COUNTER** **COUNTER** **GOTO**

keys are pressed to define TEST 2 (it has the same logic as TEST 1). The Word Recognizer and counter values are entered. **END** key is pressed to specify the end of TEST 2. Again, the OR IF prompt is replaced, this time by END TEST 2.

NOTE

The COUNTER number must be changed to a 2 before you enter the counter value; i.e., the default counter number is always 1. Otherwise, when you enter the value 12 in TEST 2, the value of the counter in TEST 1, i.e., 23, will change to 12. The reason for this is that each of the two available counters can only count one value in any one program.

WD RECOGNIZER key is pressed. The 7D02 looks for the third data word (FF) on the s.u.t. data bus only after the specified number of occurrences of data word #2 (4C) have been counted.

TRIGGER key is pressed. The default parameters are used in this example. Once the third data word is found, the trigger will occur.

START key is pressed and the 7D02 begins to look for the first Word Recognizer.

NOTE

When this program reaches TEST 3, it will remain in TEST 3. There is no GO TO to another test.

Data Displays

CTR1=00023 EVT TRIG LOC = 015			
CTR2=00012 EVT TRIG IN TEST 3			
LINE	ADDRESS	DATA	CNTL
015T	0011	FF	R11
016	0012	F7	R11
017	0013	F6	R11
018	0014	FF	R11
019	0015	B7	R11
020	0016	A3	R11
021	0017	BA	R11
022	0018	56	F11
023	0019	AC	R11
024	001A	6E	F11
025	001B	73	R11
026	001C	75	R11
027	001D	FC	R11
028	001E	F7	F11
029	001F	C9	R11
030	0020	D9	R11
031	0021	18	R11
DISPLAY ← ACGMEN 0 MAIN			
0 ABSOLUTE			
0 ABSOLUTE			
1 MNEMONIC			

Explanation of Data Displays

This is an absolute display of acquired data.

The run-time information indicates that the event in TEST 1 occurred 23 times (CTR1=00023). In TEST 2, event 2 occurred 12 times (CTR2=00012). Then control was transferred to TEST 3 where the trigger occurred (TRIG IN TEST 3).

2918-24

Example 5—Sequential Word Recognition With Else Clause

Program Definition

This program checks to see if a specific value on the address bus (783D) is immediately followed by another specific value (9107) on the address bus. If it is, then the trigger occurs. If it is not, then counter #1 is incremented and then program execution is transferred back to TEST 1.

Program Displays

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=783D
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 GOTO 2
END TEST 1
TEST 2
2 IF
2 WORD RECOGNIZER # 2
2 DATA=XX
2 ADDRESS=9107
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2 THEN DO
2 TRIGGER 0-MAIN
2 0-BEFORE DATA
2 0-SYSTEM UNDER TEST CONT.
2 0-STANDARD CLOCK QUAL.
2 ELSE
2
2 GOTO 1
2 COUNTER # 1 0-EVENTS
2 0-INCREMENT
2
END TEST 2

```

Program Explanation

WD RECOGNIZER key is pressed and the address specified as 783D.

GOTO key is pressed.

END key is pressed to end TEST 1.

WD RECOGNIZER key is pressed and the address is specified as 9107.

TRIGGER key is pressed.

ELSE key is pressed.

I key is pressed.

GOTO key is pressed.

COUNTER key is pressed and the default value 0-INCREMENT is used.

I key is pressed.

END key is pressed.

2918-25

Example 6—Simultaneous Event Recognition

Program Displays

```

TEST 1
1 IF
1
1 WORD RECOGNIZER # 1
1 DATA=41
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 OR
1 WORD RECOGNIZER # 2
1 DATA=XX
1 ADDRESS=F836
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1
1 THEN DO
1 TRIGGER O-MAIN
1 O-BEFORE DATA
1 O-SYSTEM UNDER TEST CONT.
1 O-STANDARD CLOCK QUAL.
END TEST 1

```

2918-26

Program Definition

This program simultaneously monitors the data bus for two different word recognitions. If the data value for Word Recognizer #1 or the address specified for Word Recognizer #2 is found, the trigger occurs.

Program Explanation

[] (square brackets) key is pressed. The first three lines of the program appear on the screen.

This key must precede the first WD RECOGNIZER or COUNTER key that defines an item that is to be included in the simultaneous testing.

This construction is used to create compound events (in this case) or compound commands.

WD RECOGNIZER key is pressed.

The next six lines of the program are displayed. This is the first of the two Word Recognizers to be checked for simultaneously. The data is defined as a 41.

OR key is pressed. Notice that no THEN DO statement is provided inside the square brackets.

WD RECOGNIZER key is pressed again. This is the second of the two Word Recognizers to be included in the simultaneous testing. The address is defined as F836.

[] key is pressed again. This indicates the end of the events to be included in the simultaneous testing. The end of the square bracket is displayed, followed by a THEN DO.

TRIGGER key is pressed. Its parameters can then be defined. This example uses the default values.

END key is pressed.

START is pressed. As in the previous examples, the display at the bottom of the screen changes.

NOTE

The key sequence **WD RECOGNIZER** **TRIGGER** **WD RECOGNIZER** **TRIGGER** will produce the following program. It gives the same results (once the same data values are entered in Word Recognizers #1 and #2) as the first program shown in this example.

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=41
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 TRIGGER Q-MAIN
1 Q-BEFORE DATA
1 Q-SYSTEM UNDER TEST CONT.
1 Q-STANDARD CLOCK QUAL.
1 OR IF
1 WORD RECOGNIZER # 2
1 DATA=XX
1 ADDRESS=FB36
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 TRIGGER Q-MAIN
1 Q-BEFORE DATA
1 Q-SYSTEM UNDER TEST CONT.
1 Q-STANDARD CLOCK QUAL.
END TEST 1

```

2918-27

COUNTER EXAMPLES

Example 7—Count Time Interval

Program Displays

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=1D
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1OR IF
1

```

2918-28

Program Definition

This program will trigger a specified period of time (12 μ S) after the data value (1D) is found.

Program Explanation

WD RECOGNIZER key is pressed and the parameters specified.

COUNTER key is pressed. This is where you define what is to be counted, events or time. In this case it is time.

When you position the cursor in the EVENTS field, the menu expands to:

```

0-EVENTS
0 EVENTS
1  $\mu$ S
2 MS

```

Selecting a 1 or a 2 causes the 0-EVENTS prompt to be replaced with:

1- μ S

if a 1 is selected, or with:

2-MS

if a 2 is selected. In either case the 0-INCREMENT menu changes to:

```

0-RUN
0 RUN
1 STOP
2 RESET AND RUN

```

This menu item is expanded to three choices. 0-RUN says continue counting from whatever value the counter contains. 1-STOP says stop counting intervals of time. 2-RESET AND RUN says reset the counter to 0 and begin counting intervals of time again.

This example requires the counter to reach a value of 12 for the counter event to occur, so 0-RUN is used.

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=1D
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 COUNTER # 1 1-uS
1 0-RUN
1 OR IF
1 COUNTER # 1 = 00012 1-uS
1 THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

COUNTER key is pressed and the number of microseconds to be counted, i.e., 12, is specified.

TRIGGER is pressed and its parameters defined. The default values are used here.

END key is pressed.

START key is pressed and the 7D02 starts monitoring the s.u.t. data bus. When the trigger event occurs the 7D02 will display the acquired data.

Data Displays

2918-29

```

CTR1=00012 uS   TRIG LOC = 015
CTR2=00000 EVT  TRIG IN TEST 1

```

2918-30

Explanation of Data Displays

Counter 1 shows the number of microseconds counted. Note that the time is shown, i.e., EVTS changed to μ S.

COUNTER EXAMPLES

Example 7—Count Time Interval

Program Displays

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=1D
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1OR IF
1

```

2918-28

Program Definition

This program will trigger a specified period of time (12 μ S) after the data value (1D) is found.

Program Explanation

WD RECOGNIZER key is pressed and the parameters specified.

COUNTER key is pressed. This is where you define what is to be counted, events or time. In this case it is time.

When you position the cursor in the EVENTS field, the menu expands to:

```

0-EVENTS
0 EVENTS
1  $\mu$ S
2 MS

```

Selecting a 1 or a 2 causes the 0-EVENTS prompt to be replaced with:

1- μ S

if a 1 is selected, or with:

2-MS

if a 2 is selected. In either case the 0-INCREMENT menu changes to:

```

0-RUN
0 RUN
1 STOP
2 RESET AND RUN

```

This menu item is expanded to three choices. 0-RUN says continue counting from whatever value the counter contains. 1-STOP says stop counting intervals of time. 2-RESET AND RUN says reset the counter to 0 and begin counting intervals of time again.

This example requires the counter to reach a value of 12 for the counter event to occur, so 0-RUN is used.

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=1D
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 COUNTER # 1 1-μS
1   0-RUN
1 OR IF
1 COUNTER # 1 = 00012 1-μS
1 THEN DO
1 TRIGGER 0-MAIN
1   0-BEFORE DATA
1   0-SYSTEM UNDER TEST CONT.
1   0-STANDARD CLOCK QUAL.
END TEST 1

```

COUNTER key is pressed and the number of microseconds to be counted, i.e., 12, is specified.

TRIGGER is pressed and its parameters defined. The default values are used here.

END key is pressed.

START key is pressed and the 7D02 starts monitoring the s.u.t. data bus. When the trigger event occurs the 7D02 will display the acquired data.

Data Displays

2918-29

```

CTR1=00012 μS   TRIG LOC = 015
CTR2=00000 EVT  TRIG IN TEST 1

```

2918-30

Explanation of Data Displays

Counter 1 shows the number of microseconds counted. Note that the time is shown, i.e., EVTS changed to μ S.

Example 8—Trigger If Event Occurs Within Time "Window"

Program Definition

If Word Recognizer 2 occurs within 100 MS after Word Recognizer 1 occurs, then trigger the 7D02 and show the time period between the occurrence of the two Word Recognizers.

Program Displays

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=0006
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1
1 GOTO 2
1 COUNTER # 1 2-MS
1 2-RESET AND RUN
1
END TEST 1
TEST 2
2IF
2 WORD RECOGNIZER # 2
2 DATA=XX
2 ADDRESS=F82E
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2THEN DO
2
2 TRIGGER 0-MAIN
2 0-BEFORE DATA
2 0-SYSTEM UNDER TEST CONT.
2 0-STANDARD CLOCK QUAL.
2 GOTO 3
2 COUNTER # 1 2-MS
2 1-STOP
2

```

Program Explanation

WD RECOGNIZER key is pressed and the address 0006 is specified.

[] key is pressed. The commands within the square brackets will be executed simultaneously.

GOTO key is pressed.

COUNTER key is pressed and EVENTS changed to 2-MS and 0-RUN to 2-RESET AND RUN.

[] key is pressed to end the compound command.

END key is pressed to end TEST 1.

WD RECOGNIZER key is pressed and the address is specified (F82E).

[] key is pressed. Starts compound command.

TRIGGER key is pressed. The default values are used in this example.

GOTO key is pressed.

COUNTER key is pressed. STOP is specified.

[] key is pressed to end compound command.

WD RECOGNIZER key is pressed. Address 0006 is specified.

2918-31


```

2ORIF
2 WORD RECOGNIZER # 1
2 DATA=XX
2 ADDRESS=0006
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2THEN DO
2 COUNTER # 1 2-MS
2 2-RESET AND RUN
2OR IF
2 COUNTER # 1 = 00100 2-MS
2THEN DO
2 GOTO 1
  END TEST 2
  TEST 3
  END TEST 3

```

COUNTER key is pressed and is specified as RESET AND RUN.

COUNTER key is pressed. The default value of 00002 has to be changed to 00100 to define the time "window" in which Word Recognizer 2 must occur if triggering is to occur.

GOTO key is pressed. The 3 must be changed to a 1. This completes the timing loop.

END key is pressed twice: once to end TEST 2 and once to create the dummy program TEST 3.

START key must be pressed to start program execution.

2918-32

Program Discussion

In this program the 7D02 uses one Word Recognizer to enable a time window during which it searches for a second Word Recognizer, the trigger event.

Each of the TESTS in this program has a specific purpose. In TEST 1 the 7D02 is waiting for its enabling event (ADDRESS=0006). When it is found, the timer is started and the 7D02 goes to TEST 2.

TEST 2 is more complicated. Here there is a race between the occurrence of the trigger event (ADDRESS=F82E) and the expiration of the time window. Whenever the enabling event (Word Recognizer 1) is found, the time window is restarted. If the time window expires, i.e., 100 MS after the enabling event, the 7D02 goes back to TEST 1 to look for the enabling event again. If, on the other hand, ADDRESS F82E (Word Recognizer 2) is found, then the 7D02 triggers to capture the data, stops the counter so as to tell how long into the window it was, and goes to TEST 3.

TEST 3 might seem unnecessary, but it isn't. TEST 3 tells the 7D02 to "ignore all inputs and do nothing". The 7D02 will simply wait until the delay counter is satisfied. TEST 3 is used to protect the value in the counter from further occurrences of Word Recognizer 1 after the trigger. If the 7D02 were to stay in TEST 2, a further occurrence of Word Recognizer 1 would cause the counter to be reset, and the value would be lost.

It is important to note that if the trigger event (Word Recognizer 2) happens while the 7D02 is in TEST 1 (looking for the enabling event), then no trigger will occur because Word Recognizer 2 is not used in TEST 1.

Data Displays

```

CTR1=00061 MS   TRIG LOC = 015
CTR2=00000 EVT  TRIG IN TEST 2
LOC  ADDRESS OPERATION /IRQ/NMI
015T F82E 06 READ 11
016 0006 15 READ 11
017 0006 16 WRITE 11
018 F82F BEQ $F865 11
019 F830 34 READ 11
020 F831 TST $0004 11
021 F832 00 READ 11
022 F833 04 READ 11
023 0004 09 READ 11
024 F834 BNE $F82C 11
025 F835 F6 READ 11
026 F82C INC $0006 11
027 F82D 00 READ 11
028 F82E 06 READ 11
029 0006 16 READ 11
030 0006 17 WRITE 11
031 F82F BEQ $F865 11
DISPLAY←ACGMEM 0-MAIN
1-MNEMONIC
0 ABSOLUTE
1 MNEMONIC

```

Explanation of Data Displays

This is the mnemonic display of data acquired with this program. The trigger is at location 015 and it occurred at address F82E. Counter #1 has a value of 61 MS, i.e., the distance into the window at which the trigger occurred.

2918-33

Example 9—Counter Tips

Program Displays

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1 COUNTER # 2 0-EVENTS
1 0-INCREMENT
1
1 OR IF
1 COUNTER # 1 = 00500 0-EVENTS
1 THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

2918-34

Data Displays

```

CTR1=00500 EVT TRIG LOC = 015
CTR2=00741 EVT TRIG IN TEST 1

```

2918-35

Program Definition

This program illustrates the fact that counters continue to count until they reach their specified value or until the last data acquisition takes place.

Program Explanation

WD RECOGNIZER key is pressed.

1 key is pressed.

COUNTER key is pressed. This gives counter #1.

COUNTER key is pressed and the 1 is changed to a 2, giving counter #2.

In both counter commands, the default parameter, 0-INCREMENT, is selected.

1 key is pressed to end the compound command.

COUNTER key is pressed and a value of 00500 is specified for the event for counter #1.

TRIGGER key is pressed.

END key is pressed.

Data Explanations

This example points out one of the basic operating principles of 7D02 counters . . . counters continue to count until they either reach their specified value (500 for count #1) or until the delay counter reaches its programmed value (741 for counter #2).

If you change the trigger parameter from
0-BEFORE DATA
to:

2-AFTER DATA

the display changes to:



```
CTR1=00500 EVT TRIG LOC = 239
CTR2=00517 EVT TRIG IN TEST 1
```

2918-36

The reason for the change in the display for counter #2 is that the counter has less time to run, i.e., continue counting, until the delay counter reaches its programmed value.

Note that for 3-ZERO DELAY, counter #2 stops counting on the next clock cycle. As a result, its value would be 501 in this example.

Test for the Absence of an Event

Example 10—NOT Word Recognition

Program Displays

```

TEST 1
1 IF
1
1 NOT
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=1 /IRQ=1 FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1
1 THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

2918-37

Program Definition

This program triggers whenever any interrupt, either IRQ or NMI, occurs.

Program Explanation

NOT key is pressed. The first four lines of the program appear on the screen. The system automatically provides a square brackets ([]) key. The NOT applies only to the event which immediately follows it in the program. In this case, Word Recognizer #1.

WD RECOGNIZER key is pressed and the next six lines of the program are displayed. A 1 is entered for both /NMI and /IRQ. Since these are negative logic fields (indicated by the slash), a 1 means "no interrupt". Word Recognizer 1 will be TRUE when there is no interrupt. The total event clause, therefore, will be TRUE when either interrupt occurs.

[] key is pressed. This specifies the end of a compound event. The end of the square bracket is followed by the THEN DO prompt.

NOTE

The **NOT** key must always appear within square brackets. The **NOT** key only influences the first event immediately after the NOT. A **NOT** **[]** key sequence is illegal.

TRIGGER key is pressed.

END key is pressed.

START key is pressed and the 7D02 starts acquiring data. When an interrupt occurs, the 7D02 will trigger..

Qualified Data Storage

Use of the Qualify key allows you to screen data as it passes on the s.u.t. system bus and to select the data that you want to be stored in acquisition memory. This process is called data qualification.

The **QUALIFY** key can be used to implement two types of data qualification: block and command.

Example 11—Block Qualification

Block qualification is defined once for the entire program. It may be defined before or after any TEST. When QUALIFY is used as a block qualifier, data will only be stored on clock cycles in which data on the system-under-test bus satisfies the Qualify event.

The program in Example 11 will store only on occurrence of instruction fetches.

Program Displays

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=F82F
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1
QUALIFY
Q STORE ON
Q WORD RECOGNIZER # 2
Q DATA=XX
Q ADDRESS=XXXX
Q /NMI=X /IRQ=X FETCH=1 R/W=X
Q BA=X INVAL OP=X EXT TRIG IN=X
Q TIMING WR=X
END QUALIFY

```

2918-38

Program Explanations

WD RECOGNIZER key is pressed and the trigger event entered (ADDRESS=F82F).

TRIGGER key is pressed and the default values used.

END key is pressed.

QUALIFY key is pressed to start the block qualification.

WD RECOGNIZER key is pressed and FETCH is entered as a 1.

END key is pressed to end the block qualification.

Data Displays

CTR1=00000 MS TRIG LOC = 000			
CTR2=00000 EVT TRIG IN TEST 1			
LOC	ADDRESS	DATA	CNTL
000T	F82F	27	F11
001	F831	7D	F11
002	F834	26	F11
003	F82C	7C	F11
004	F82F	27	F11
005	F831	7D	F11
006	F834	26	F11
007	F82C	7C	F11
008	F82F	27	F11
009	F831	7D	F11
010	F834	26	F11
011	F82C	7C	F11
012	F82F	27	F11
013	F831	7D	F11
014	F834	26	F11
015	F82C	7C	F11
016	F82F	27	F11
017	F831	7D	F11
018	F834	26	F11
DISPLAY←ACGMEM 0-MAIN			
0-ABSOLUTE			

Explanation of Data Displays

This display was changed from the mnemonic display to emphasise the fact that every data acquisition that was stored was, in fact, a FETCH. Note that each entry has an F in the CNTL column, indicating that this was a fetch.

2918-29

Example 12—Qualify Command

Command qualification allows data to be stored only on those clock cycles in which a Qualify command is executed. Command qualification is defined in the individual tests and can be defined as often as you need it.

The program in example 12 stores 100 WRITES, then stores the trigger word, then stores the next 127 data acquisitions following the trigger word.

In TEST 1, each time a WRITE is found (Word Recognizer 1), the cycle is stored (by the QUALIFY command) and counted. When 100 WRITES have been found, the trigger occurs, the trigger word is stored, and the 7D02 transfers to TEST 2.

TEST 2 is written to say "store every cycle". When the delay counter reaches its programmed value, the 7D02 goes to Display mode.

It is very important to note that if there were no QUALIFY command in TEST 2, that the 7D02 would never automatically stop. If TEST 2 had been a null TEST (like TEST 3 in example 8), then the memory of the 7D02 would never fill because no further QUALIFY commands would be executed once the TRIGGER occurred. If data qualification is used anywhere in a 7D02 program, only cycles which are QUALIFIED will ever be stored. Since the 7D02 only stops when its delay counter reaches its programmed value, the 7D02 would run until the STOP key was pressed.

Program Displays

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=0
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1
1 QUALIFY
1 COUNTER # 1 = 00100 0-EVENTS
1 0-INCREMENT
1
1 OR IF
1 COUNTER # 1 = 00100 0-EVENTS
1 THEN DO
1
1 QUALIFY
1 TRIGGER 0-MAIN
1 1-CENTERED
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
1 GOTO 2
1

```

Program Explanations

WD RECOGNIZER key is pressed and the R/W parameter is specified as a 0; i.e., a WRITE.

IF key is pressed to begin a compound command.

QUALIFY key is pressed.

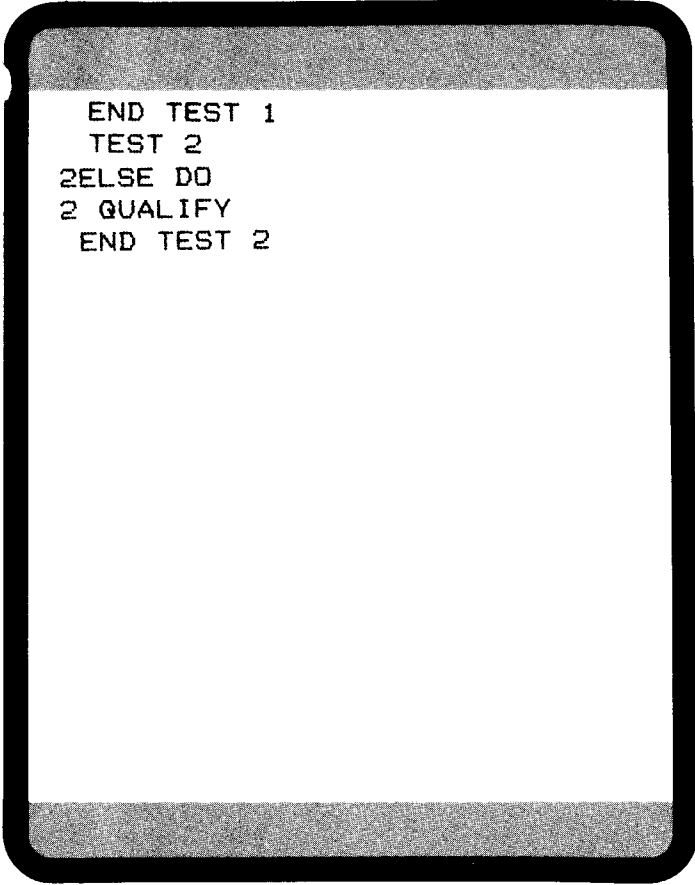
COUNTER key is pressed and 0-EVENTS and 0-INCREMENTS used.

IF key is pressed to end the compound command.

COUNTER key is pressed and a value of 00100 is specified.

IF key is pressed to begin another compound command.

QUALIFY key is pressed.



```
END TEST 1
TEST 2
2ELSE DO
2 QUALIFY
END TEST 2
```

TRIGGER key is pressed and 1-CENTERED, 0-SYSTEM UNDER TEST CONT., and 0-STANDARD CLOCK QUAL. are used.

GOTO key is pressed.

□ key is pressed to end the second compound command.

END key is pressed to end TEST.

ELSE key is pressed.

QUALIFY key is pressed.

END key is pressed to end TEST 2.

2918-40

Data Displays

Data Display Explanation

NOTE

There are 100 WRITES, then the trigger; followed by the 127 acquisitions specified by the 1-CENTERED.

```

CTR1=00100 EVT  TRIG LOC = 015
CTR2=00000 EVT  TRIG IN TEST 1
LOC  ADDRESS OPERATION /IRQ/NMI
092  082D    03 WRITE      11
093  082E    04 WRITE      11
094  0006    05 WRITE      11
095  082F    06 WRITE      11
096  0830    06 WRITE      11
097  0831    07 WRITE      11
098  0832    08 WRITE      11
099  0833    0A WRITE      11
100T—F84F—BNE    $F82C—11
101  F850    F6 READ       11
102  F851    INC    $0006   11
103  F852    00 READ       11
104  F853    06 READ       11
105  F854    0A READ       11
106  F854    0B WRITE      11
107  F82F    BEQ    $F865   11
108  F830    34 READ       11
109  F831    TST    $0004   11
110  F832    00 READ       11

```

```

DISPLAY←ACQMEM 0-MAIN
1-MNEMONIC

```

2918-41

Example 13—Block Qualification and Qualify Command Combined

If both a QUALIFY block and a QUALIFY command are used, then storage occurs if either of the following conditions is true:

1. The current event satisfies the QUALIFY block event, or
2. A QUALIFY command is executed.

The program in example 13 demonstrates this concept. Data will be stored if Word Recognizer 2 (address F832) is found in TEST 1, if Word Recognizer 3 (address F82E) is found in TEST 2, or in any case when Word Recognizer 4 (data 7C) is found.

Program Displays

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=0004
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 GOTO 2
1OR IF
1 WORD RECOGNIZER # 2
1 DATA=XX
1 ADDRESS=F832
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 QUALIFY
  END TEST 1
  TEST 2
2IF
2 WORD RECOGNIZER # 3
2 DATA=XX
2 ADDRESS=F82E
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2THEN DO
2
2  TRIGGER 0-MAIN
2    0-BEFORE DATA
2    0-SYSTEM UNDER TEST CONT.
2    0-STANDARD CLOCK QUAL.
2 QUALIFY
2 GOTO 1
2

```

Program Explanations

WD RECOGNIZER key is pressed and address 0004 is specified.

GOTO key is pressed.

WD RECOGNIZER key is pressed and address F832 is specified.

QUALIFY key is pressed for the line qualify command.

END key is pressed to end TEST 1.

WD RECOGNIZER key is pressed and address F82E is specified.

I key is pressed to begin the compound command.

TRIGGER key is pressed.

QUALIFY key is pressed.

GOTO key is pressed and 3 changed to 1.

I key is pressed to end the compound command.

END key is pressed to end TEST 2.

```
END TEST 2
QUALIFY
Q STORE ON
Q WORD RECOGNIZER # 4
Q DATA=7C
Q ADDRESS=XXXX
Q /NMI=X /IRQ=X FETCH=X R/W=X
Q BA=X INVAL OP=X EXT TRIG IN=X
Q TIMING WR=X
END QUALIFY
```

QUALIFY key is pressed to begin the block qualification.

WD RECOGNIZER key is pressed and data 7C is specified.

END key is pressed to end the block qualification.

2918-43

Example 14—Start and Stop Qualification

This program starts data qualification on the occurrence of Word Recognizer 1, stops data qualification on the following occurrence of Word Recognizer 3, and triggers only if Word Recognizer 2 occurs during data qualification.

Program Displays

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=0004
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING=X
1 THEN
1
1 QUALIFY
1 GOTO 2
1
END TEST 1
TEST 2
2 IF
2 WORD RECOGNIZER # 2
2 DATA=XX
2 ADDRESS=F82E
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING=X
2 THEN DO

```

2918-44

Program Explanation

WD RECOGNIZER key is pressed and the address (0004) is specified.

I key is pressed to start definition of a compound command.

QUALIFY key is pressed. Information on the data bus is stored each time 0004 is found.

GOTO key is pressed.

I key is pressed to end the compound command definition.

END key is pressed to end TEST 1.

WD RECOGNIZER key is pressed and the address is specified as F82E.

I key is pressed to start definition of the compound command.

TRIGGER key is pressed and the trigger location changed to AFTER DATA.

QUALIFY key is pressed. This causes the trigger word to be stored.

I key is pressed to end the compound command definition.

WD RECOGNIZER key is pressed and an address of 0006 is specified.

```

2
2 TRIGGER 0-MAIN
2 2-AFTER DATA
2 0-PROCESSOR CONTINUE
2 0-STANDARD CLOCK QUAL.
2 QUALIFY
2
2OR IF
2 WORD RECOGNIZER # 3
2 DATA=XX
2 ADDRESS=0006
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING=X
2THEN DO
2
2 QUALIFY
2 GOTO 1
2
2ELSE
2 QUALIFY
END TEST 2

```

2918-45

[] key is pressed to start a compound command qualification.

QUALIFY key is pressed. Information is to be stored each time 0006 is found.

GOTO key is pressed and the TEST number is changed from 3 to 1. This allows the program to transfer to TEST 1.

[] key is pressed to end the compound command definition.

ELSE key is pressed.

QUALIFY key is pressed. (If none of the events in TEST 2 occur, then store this data in acquisition memory.)

END key is pressed to end TEST 2.

START key can now be pressed to begin program execution.

NOTES

1. No data is stored in TEST 1 if Word Recognizer #1 is not TRUE.
2. Every cycle in TEST 2 will be stored.

CORRECTING PROGRAMMING MISTAKES

As a rule, people make three kinds of programming mistakes when they use the 7D02:

- pressing the wrong event or command key; i.e., structural errors;
- entering the wrong menu or parameter value; and
- designing a logically incorrect program.

This section will not help you to correct problems in programming logic.

To correct a wrong menu entry, move the cursor to the incorrect entry. Then enter the correct value via the numeric keypad. The new value will replace the old one.

To correct an incorrect event or command keystroke, move the cursor to the end of the program and then use the **DELETE** key.

The **DELETE** key is used to correct structural mistakes in a program and to delete an entire program from memory. It only works from the end of the program back toward the beginning of the program.

In Figure 1-9 the program is boxed into groups of lines. Each group is deleted by one **DELETE** keystroke. The boxes are only for instructional purposes, and each box is numbered in the sequence in which the **DELETE** keystroke removes it from the screen.

In general:

1. Menus and numeric fields, i.e., parameters, are treated as a single unit.
2. Groups of lines are deleted regardless of where the cursor is positioned within that group of lines.
3. System-supplied lines, such as, IF, OR IF, THEN DO, STORE ON, are automatically deleted when appropriate.
4. **DELETE** can only be used on the last entry in the program; i.e., you can only delete lines from the last keystroke to the first.

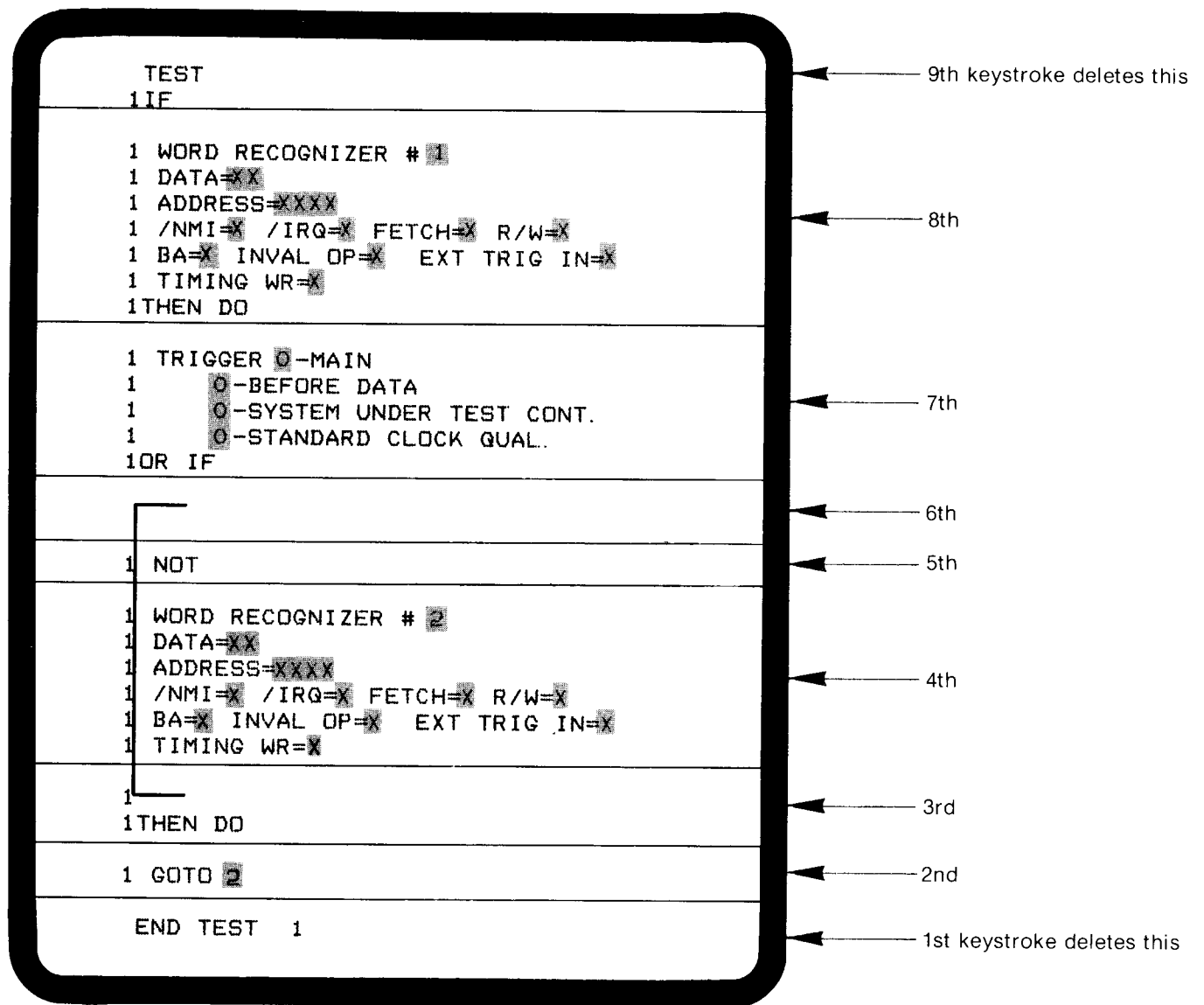


Figure 1-9. How the DELETE command works.

2918-46

STORING AND DISPLAYING MEMORY WITH IMMEDIATE MODE COMMANDS

Storing data and programs in memory and retrieving information from memory is a three-keystroke sequence. The first key is always the **IMMEDIATE** key. It is followed by one of the two-keystroke sequences listed below. The second key specifies the destination of the information. The third key specifies the source of the information. The resulting command is shown in the Immediate mode area at the bottom of the screen.

DISPLAY
ACQ MEM

Displays the contents of the acquisition memory.

DISPLAY
PROGRAM

Displays the contents of program memory.

STORE MEM
PROGRAM

Stores the program in storage memory. Swaps programs if STOREMEM contains a program.¹

STORE MEM
ACQ MEM

Stores the current data (residing in one of the acquisition memories) in storage memory.¹

DISPLAY
STORE MEM

Displays the contents of storage memory.

¹ These commands require moving the cursor to the EXECUTE field in order for the command to be executed.

Section 2

GENERAL DESCRIPTION

TABLE OF CONTENTS

INTRODUCTION

7D02 COMPONENTS

Expansion Option

Timing Option

FUNCTIONAL OVERVIEW

MODES OF OPERATION

FRONT PANEL FAMILIARIZATION

Keys

Display Positioning Potentiometers

BNC's

P6451 Probe Slot

Personality Module Slot

DISPLAY SCREEN CHARACTERISTICS

Screen Layout

Parameters

GENERAL DESCRIPTION

INTRODUCTION

The 7D02 Logic Analyzer is a design, debugging, and troubleshooting aid for use in the development of digital systems, especially microprocessor-based systems. It is capable of supporting both 8-bit and 16-bit microprocessors. The 7D02 is a plug-in for any 3-wide (or larger) Tektronix 7000-Series oscilloscope mainframe. Due to the unique requirements of the 7D02, best results will be obtained with 7603 or 7704A mainframes. Use of storage mainframes is not recommended as these crts are optimized for storage applications only.

7D02 COMPONENTS

The basic 7D02 offers a wide (28-channel) Acquisition Memory, four-word recognizers, two general purpose counters, a user-configurable clock, data qualification circuitry, a state machine architecture, three memories, and a simple facility for customizing the 7D02 for a wide variety of microprocessors.

The basic 7D02 will acquire data from the system under test (s.u.t.) on 28 channels, usually organized as:

- 16 address bus lines
- 8 data bus lines
- 4 control lines

The word recognizers perform recognition on these 28 channels plus an additional three channels (an external trigger line and two additional control lines from the s.u.t.). Any of the four word recognizers can be used to recognize user-defined patterns on all 31 channels or on any subset of the channels. The word recognizers can also be complemented, i.e., they can recognize the absence of a pattern.

The counters can count up to 65,534 events, microseconds, or milliseconds. Execution of commands (such as qualifying data or triggering) can be conditioned on a counter reaching a user-specified value.

The state machine architecture provides tremendous flexibility in triggering. Occurrences of events (counters reaching user-specified values, word recognitions) are input to the state machine. The state machine can examine all of its inputs simultaneously and generate a set of simultaneous commands to the counters, trigger(s), and data qualification circuitry. The commands can include a transition into a new state. Different states can associate different commands with the same inputs. The state machine executes in real time with the s.u.t. and can enter any of its states in any order, any number of times. This makes the 7D02 ideal for following the complicated, convoluted sequences of bus transactions often associated with software/firmware-based systems. A simple yet powerful user language makes detailed understanding of the state machine unnecessary.

The basic 7D02 contains three memories:

1. Program Memory contains the current user-entered program.
2. Acquisition Memory (or Main Acquisition Memory) contains the data acquired the last time a 7D02 program was run.
3. Storage Memory can be used to save either a program or the contents of the Acquisition Memory.

General Description—7D02 Operators

The PM-100 series of Personality Modules makes it possible to reconfigure the 7D02 for a wide range of microprocessors by simply plugging the appropriate module into the front panel of the 7D02. A General Purpose Personality Module (PM-101) is available for non-microprocessor systems, or for systems based on microprocessors for which a specific Personality Module is not available.

Expansion Option

Option 3 increases the width of Acquisition Memory and word recognition by 16 channels. The 44 channels of memory are generally assigned as:

- 24 address bus lines
- 16 data bus lines
- 4 control lines

As in the basic 7D02, two additional control lines and the external trigger line are also available for word recognition.

Timing Option

Option 1 turns the 7D02 into two logic analyzers in one box. The Timing Option has its own trigger, acquisition memory (Timing Option Acquisition Memory), and word recognizer.

Used synchronously, the Timing Option acts as an 8-channel extension of the Main Acquisition hardware.

Used asynchronously (with its own internally-generated time base), it is a timing logic analyzer that recognizes (and stores) both data and glitch information on 8 channels.

The Timing Option can be used to trigger the Main section, and vice versa.

FUNCTIONAL OVERVIEW

This overview will introduce some important terms and concepts. Understanding them will make programming of the 7D02 easier.

Using the 7D02 keyboard, the user enters a program describing an event or sequence of events in the system-under-test. The program usually directs the 7D02 to trigger (acquire data and stop running) when the event or sequence is detected. It may also describe the type of data the user wishes to acquire.

When the program is run, information is extracted from the program to set up the various acquisition hardware components (counters, word recognizers, state machine, clock and data qualifiers, trigger delay counter, etc.).

The acquisition hardware is then activated and data sampling begins. Data is always sampled on an edge of the system-under-test (s.u.t.) master clock, but not necessarily on every s.u.t. clock edge. The clock qualification circuitry (which was set up with information from the user's program and/or Personality Module-defined defaults) selects the edges of the s.u.t. master clock on which data is considered to be valid. This qualified s.u.t. clock becomes the 7D02 State Clock, which defines a single cycle of the 7D02. Data is sampled only on the State Clock.

On each cycle of the 7D02 (at the State Clock), s.u.t. data is sampled, program-defined conditions are tested, and commands associated with conditions that have occurred are executed. For example, the occurrence of a particular word recognition can cause a counter to be incremented. The state machine checks all of its inputs simultaneously and causes the relevant programmed commands to be executed simultaneously in real time with the system-under-test.

Although data is sampled on every State Clock, it is not necessarily stored in the Acquisition Memory. The data qualification circuitry uses program-defined (or default) criteria to determine whether or not each particular word of sampled data is to be stored. A word of data consists of the values on all channels being sampled on the system-under-test at a State Clock.

Words of data that meet the data qualification criteria are stored in consecutive locations in the Acquisition Memory. Data is stored continuously from the moment the 7D02 begins running the program. Data will continue to be stored even after the Acquisition Memory is filled, with the oldest word of data being lost every time a new one is stored.

Execution of a trigger command has the effect of activating a delay counter associated with the Acquisition Memory. Once activated, the delay counter will increment every time a word of data is stored into the Acquisition Memory. When the value in the delay counter reaches the user-specified value, acquisition is considered to be complete and the 7D02 stops executing the program.

NOTE

The Timing Option has its own trigger and delay counter. If the Timing Option is installed and data is being acquired with both the Main and Timing Option sections of the 7D02, then both sections must trigger and both delay counters must reach their programmed values before the program stops executing.

Once the program has stopped executing, acquired data is formatted and displayed. The user is now free to move acquired data to Storage Memory; to move programs to or from Storage Memory; to examine acquired data in different formats; to enter a new program; to rerun the program, etc.

MODES OF OPERATION

Powering-up the 7D02 causes diagnostic routines to be run automatically. Assuming there is no failure, the routines will require 5—10 seconds to complete. If the 7000-Series mainframe is "cold", the diagnostics will be completed before the crt even warms up. Otherwise, diagnostic displays and test patterns may be seen by the user on the crt.

After the completion of diagnostic testing, the 7D02 will always be in one of five modes of operation. They are discussed in depth in Section 3. Briefly, in the order they are likely to be encountered, they are:

1. Programming Mode

This is the mode into which the 7D02 powers-up. The user enters the program describing the data to be sought and stored.

2. Run Mode

Data is acquired under the direction of the program in Program Memory.

3. Display Mode

The acquired data is formatted and displayed on the crt display.

4. Immediate Mode

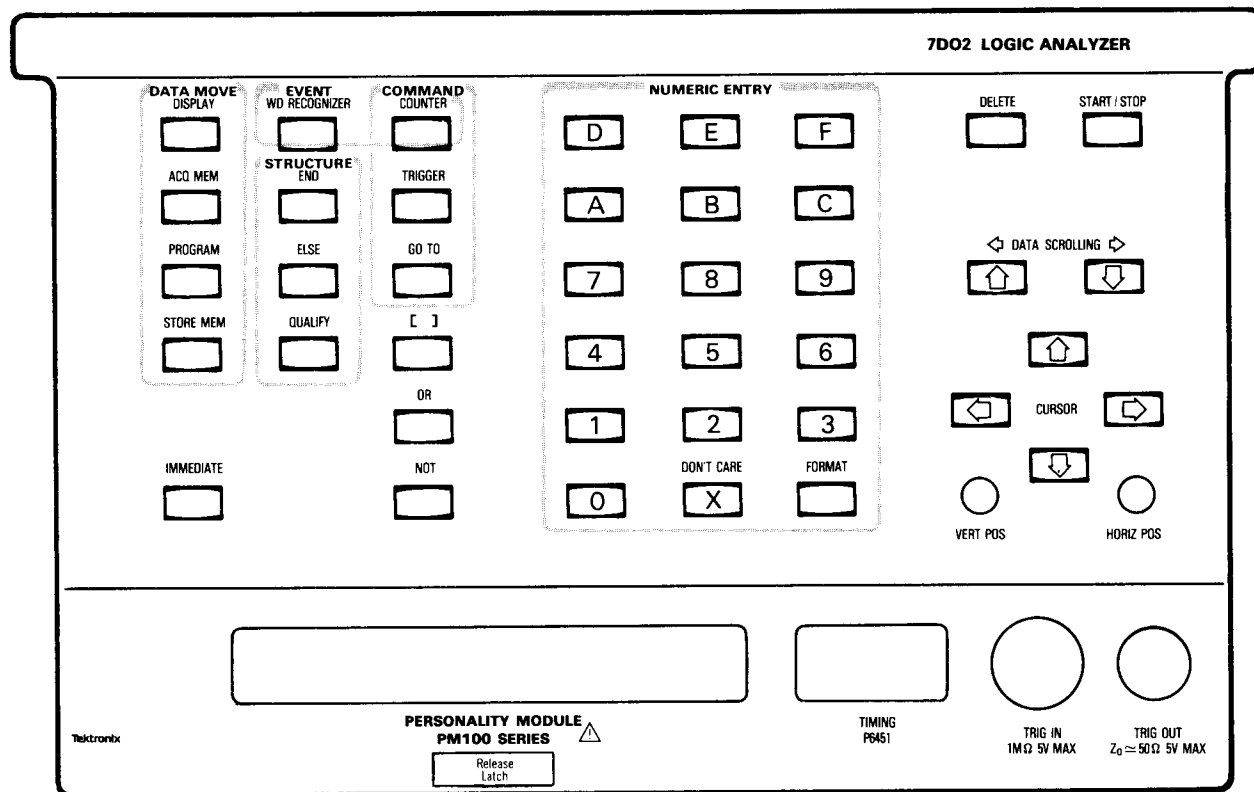
The user enters commands that are executed immediately, rather than when the program is run. These include moving information to or from Storage Memory, changing displays, and changing operation modes.

5. Format Mode

Characteristics of program and data displays (such as the radices in which word recognizer fields and acquired data are displayed) can be changed by the user.

FRONT PANEL FAMILIARIZATION

The 7D02 front panel appears in Figure 2-1.



2918-47

Figure 2-1. 7D02 front panel.

Keys

Each key on the front panel is described in detail in Section 5. Most keys are grouped by function within colored borders. Briefly, the functions of the key groups are as follows:

1. The Data Movement keys are used to move information among the 7D02 memories and to/from the display.
2. The Event, Command, and Structure keys, together with the **IF**, **OR**, and **NOT** keys, are used to enter the user's program.
3. The **IMMEDIATE** key is used to enter Immediate mode commands.
4. The Numeric Entry keys are used to enter command and display parameters.
5. The **DELETE** key removes unwanted items from a program.
6. The **START/STOP** key runs or aborts execution of the current program.
7. The **DATA SCROLLING** keys move a display "window" through acquired data or a stored program.
8. The **CURSOR** keys move the blinking cursor around the screen, allowing programs to be entered/deleted and parameters to be changed.

Display Positioning Potentiometers

The VERT POS and HORIZ POS potentiometers allow the user to position the display vertically and horizontally. Due to differences in 7000-Series mainframes, it may be necessary to adjust the display position for a particular mainframe. Once adjusted, there should be little need to readjust.

BNCs

The "TRIG IN" BNC allows other devices such as a coded scope probe or the A6701 18-bit word recognizer to trigger the 7D02. This input is TTL-compatible. This line is labeled "EXT TRIG IN" in 7D02 program displays.

The "TRIG OUT" BNC allows the 7D02 to trigger other instruments. The TRIG OUT signal is generated when a trigger command is executed. See Section 7 for details.

P6451 Probe Slot

The P6451 probe plugs into the P6451 Probe slot. The P6451 is used to acquire data with the Timing Option.

Personality Module Slot

The PM-100 Series Personality Module plugs into the Personality Module slot. The Personality Module configures the 7D02 for a particular microprocessor. It acquires data from the s.u.t. and determines data display formats, disassembly of data, word recognizer formats, etc.

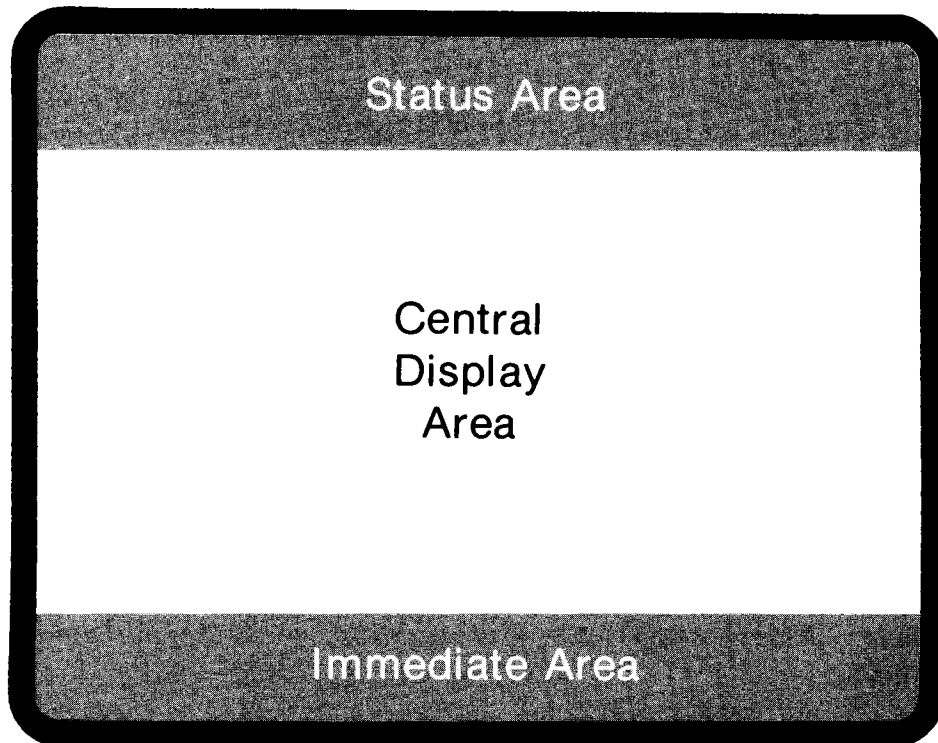


The 7D02 should never be operated without a Personality Module attached. The Module should never be plugged in with the 7D02 powered-on. To do so could cause permanent damage to the 7D02 or the Personality Module.

DISPLAY SCREEN CHARACTERISTICS

Screen Layout

Once diagnostics have been completed, the 7D02 display is generally divided into three regions (see Figure 2-2).



2918-48

Figure 2-2. 7D02 screen display areas.

They are:

1. Status area

Two lines of inverse video at the top of the screen are reserved for error and status messages. Monitoring information appears here during Run mode. Vital statistics (such as final counter values) appear during Display mode.

2. Central Display Area

The bulk of the display (a varying number of lines, depending on the size of the Immediate area) is used to display the contents of one of the memories (Program, Acquisition, or Storage) or the Format mode display.

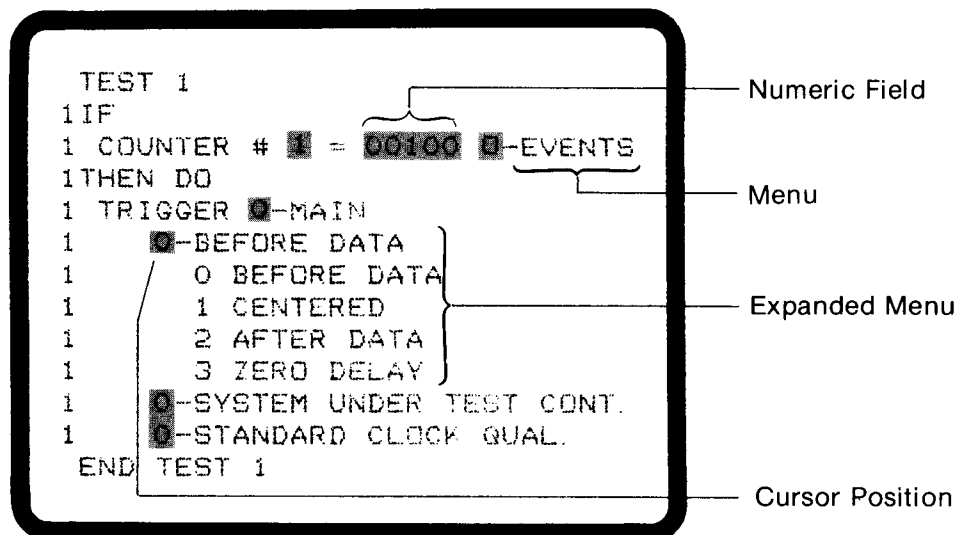
3. Immediate Area

An inverse-video area at the bottom of the screen is used for the entry of Immediate mode commands. When the user is not in Immediate mode, the area contains the display command that was last entered in Immediate mode (i.e., the one that was responsible for, and identifies, the current contents of the Central Display area). During Display mode, parameters associated with the display command can be manipulated. During Format mode, this area goes away completely.

Parameters

Program and other parameters are entered with the Numeric Entry keys. User-changeable parameters always appear on the display as digits in the opposite video mode as their background (i.e., inverse video in the Central Display area, normal video in the Immediate area). There are two types of parameters that appear in 7D02 displays (see Figure 2-3):

1. A menu gives the user a number of choices, each labeled with an arbitrary digit. The user selects a choice by moving the blinking cursor to the menu and pressing the digit key associated with the desired choice. Only the current selection for the menu is displayed, unless the user positions the cursor onto the digit associated with the selection; at that time, the menu will be "expanded" to display all possible choices. Since there is only one cursor, at most one menu can be expanded at any time.
2. A numeric field is a parameter in which the digits entered have meaning of themselves; i.e., they are not used for selection of a choice but actually are the parameter values.



2918-49

Figure 2-3. 7D02 parameters.

Section 3

MODES OF OPERATION

TABLE OF CONTENTS

INTRODUCTION

PROGRAMMING MODE

- Display Contents
- Valid Keys

DISPLAY MODE

- Display Contents
 - Main Data Displays
 - Timing Option Data Displays
 - Stored Program Displays
- Valid Keys
- Scrolling
- Highlighting Memory Differences

IMMEDIATE MODE

- Display Contents
- Valid Keys
- Valid Immediate Mode Commands
 - GOTO Commands
 - Data Movement Commands
 - Display Commands

FORMAT MODE

- Display Contents
- Valid Keys
- Features Controlled In Format Mode
 - Radices
 - Glitch Display
 - Memory Difference Highlighting
 - Channel Inversion

MODES OF OPERATION

INTRODUCTION

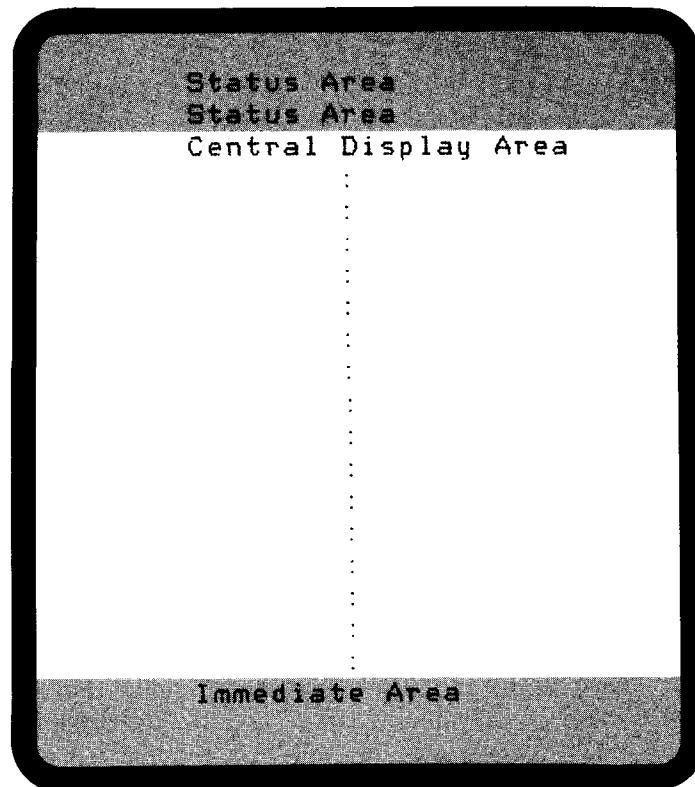
Once power-up diagnostics have been completed, the 7D02 will always be operating in one of five modes. In the order in which they are most likely to be encountered, they are: Programming mode, Run mode, Display mode, Immediate mode, and Format mode.

The 7D02 powers-up in Programming mode. Once the user enters a program and presses the **START** key, the 7D02 goes into Run mode to acquire data under direction of the program. When acquisition is complete, the 7D02 automatically enters Display mode. In this mode, the acquired data is formatted and displayed.

Immediate mode makes it possible to enter commands that are executed immediately (rather than when the program is running), including commands that transfer between Programming and Display modes.

Format mode is somewhat outside the normal flow of 7D02 usage. It can be entered whenever the 7D02 is not in Run mode, suspending whatever activity the user was involved in. When the user leaves Format mode, the 7D02 returns to the suspended activity. In Format mode, the user has the ability to change features (such as radices) of the program and data displays.

Each mode can be distinguished by the contents of the three screen areas (see Figure 3-1), the location of the blinking cursor on the screen, and the keys that are valid in that mode. The modes will be discussed within this framework.



2918-50

Figure 3-1. 7D02 screen display areas.

PROGRAMMING MODE

When diagnostic testing has been completed, the 7D02 will be in Programming mode. Programming mode can also be entered from Display mode by pressing the keystroke sequence **IMMEDIATE** **DISPLAY** **PROGRAM**.

In Programming mode, the user enters a program directly into Program Memory or modifies a program that is already there. Modification of a program can include deleting part or all of it, changing parameter values (menu selections and the contents of numeric fields), and/or adding new sections to the end of the program.

Display Contents

The Central Display area of the screen will contain a display of the program currently in Program Memory. The blinking cursor will be in this area of the screen, indicating the current position within the program. Parameters can be changed by moving the cursor onto them and pressing the appropriate digit key(s). Items can be added to or deleted from the program when the cursor is at the end of the program.

The Status area will be blank except for error and informational messages that will appear when the user attempts to make an invalid keystroke.

The Immediate area will be only one line long and will contain the text "DISPLAY ← PROGRAM", indicating that the contents of the Program Memory are being displayed in the Central Display area.

Valid Keys

Programs are entered using the Programming keys: i.e., the Event, Command, and Structure groups, plus the **OR**, **NOT** and **II** keys. Section 4 provides an overview of how to create a program.

The **DELETE** key is used to remove keystroke elements from a program. Keystroke elements may be deleted from the end of a program only.

The **CURSOR** keys, in conjunction with the digit keys (0—F) and the **DON'T CARE** key, can be used to change menu selections and the contents of numeric fields. If the program is too large to fit on the screen at one time, it will automatically be scrolled as the cursor is moved with the **CURSOR** keys.

The **START**, **IMMEDIATE**, and **FORMAT** keys cause mode changes. They enter the Run, Immediate, and Format modes, respectively. Typically, once the program has been entered or modified, the user will press the **START** key.

The 7D02 will then acquire data (using the program in Program Memory) and transfer automatically into Display mode.

All other keys are invalid in Programming mode and will cause an error message to appear in the Status area.

See Section 5 for details on the function and use of all keys.

RUN MODE

Run mode is entered by pressing the **START** key (in any mode except Format) or by entering a GOTO command in Immediate mode. The 7D02 will not enter Run mode if the program in Program Memory is incomplete or contains contradictory bracketed commands; instead, an error message appears in the Status area.

Otherwise, the program in Program Memory is run. The previous contents of Acquisition Memory are lost, and new data is acquired. When the program stops executing, the 7D02 will automatically transfer into Display mode so the acquired data can be viewed.

Display Contents

The contents of the Immediate and Central areas of the screen do not change during Run mode. They will be the same as they were before Run mode was entered.

In Run mode, the 7D02 monitors the running program and displays periodically updated information in the Status area. The blinking cursor disappears and a blinking "RUNNING" message usually appears in the Status area. See Section 7 for details on the contents of the Status area during Run mode.

Valid Keys

Only the **STOP** key is valid in Run mode. If it is pressed, execution of the program is aborted immediately and Display mode is entered. The Acquisition Memory will contain whatever data, if any, was acquired before the **STOP** key was pressed.

All other keys are ignored in Run mode.

DISPLAY MODE

When the 7D02 leaves Run mode, it automatically transfers directly into Display mode. If data was acquired by the Timing Option only (the program contained a Timing Option trigger command and no Main trigger command), the Timing Option Acquisition Memory is initially displayed; in all other cases, the contents of the Main Acquisition Memory are initially displayed.

As long as the 7D02 is not in Run mode or Format mode, Display mode can also be entered with either the **IMMEDIATE** **DISPLAY** **ACQ MEM** keystroke sequence or the **IMMEDIATE** **DISPLAY** **STORE MEM** keystroke sequence. (Exception: an error message is displayed if the latter sequence is entered when the Storage Memory is empty.)

NOTE

*In the automatic transfer from Run to Display mode, the **IMMEDIATE** **DISPLAY** **ACQ MEM** sequence is provided by the 7D02.*

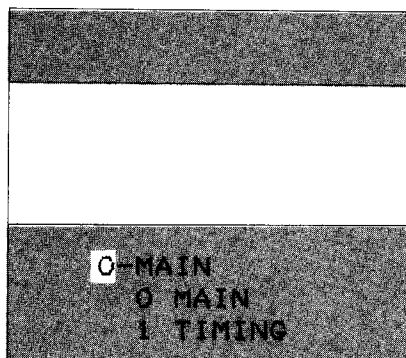
In Display mode the user can examine data in either Storage Memory or one of the Acquisition Memories (Main or Timing Option). Programs stored in Storage Memory can also be viewed.

Display Contents

During Display mode, the Immediate area contains the display command ("DISPLAY - ACQMEM" or "DISPLAY - STOREMEM") that caused Display mode to be entered. The blinking cursor remains in the Immediate area throughout Display mode; any parameters associated with the display command can be modified at any time during Display mode. Modification of these parameters immediately results in different displays.

The exact contents of the screen areas will depend on the display command that was used to enter Display mode, on the current choices for the parameters associated with the display command, and on the presence or absence of the Timing Option.

In the basic 7D02, the **DISPLAY** **ACQ MEM** key sequence causes the contents of the (Main) Acquisition Memory to be displayed. If the Timing Option is installed, the "DISPLAY - ACQMEM" command in the Immediate area will be followed by this menu:



2918-51

Selecting "0" causes the Main Acquisition Memory contents to be displayed; selecting "1" causes the Timing Option Acquisition Memory to be displayed. By changing this menu selection, the user can switch back and forth between displays of these two memories without leaving Display mode.

If the Immediate command was **DISPLAY** **STORE MEM**, the display will depend on the contents of the Storage Memory (Main data, Timing Option data, or a program). Main or Timing data in the Storage Memory will be displayed exactly as they would be from the Acquisition Memory, except that the text "DISPLAY ← STOREMEM" appears in the Immediate area instead of "DISPLAY ← ACQMEM" (and the MAIN/TIMING menu will never appear).

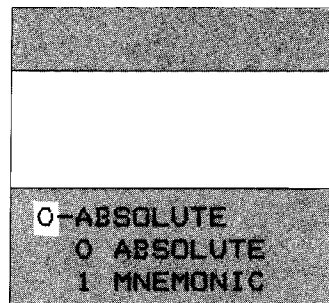
In order to switch between Storage Memory and Acquisition Memory displays, the user must enter a new display command in Immediate mode.

The exact contents of the screen areas, based on what is being displayed, are:

1. MAIN DATA DISPLAYS

Immediate and Central Areas

When Main data is displayed, either from Acquisition Memory or from Storage Memory, the following additional menu will appear in the Immediate area:



2918-52

The choice for this menu determines the format in which the data appears in the Central area of the screen.

Selecting "0" causes the data to be displayed in a tabular numeric form. The radices in which the data fields are displayed can usually be chosen in Format mode.

Selecting "1" causes the data to be displayed in a form similar to a listing from an assembler. The radix choices in Format mode do not affect this display. Numeric fields will be displayed in a radix compatible with the target processor (usually hexadecimal).

NOTE

The exact contents of the Central area of the Main data display is Personality Module-dependent. Consult the relevant Personality Module manual for details.

Status Area

The status area contains the counter and trigger information that was gathered when the displayed data was acquired. The display takes the following form for all Main data displays:

CTR1=00017 EVT	TRIG LOC = 127
CTR2=01745 uS	TRIG IN TEST 2

2918-53

The fields on the left side of the Status area give the final values of the counters and the units in which they were counting. The five digits will be replaced with the text "OVRFL" when a counter overflows (exceeds a value of 65,534).

Data in memory is stored one word per memory location, beginning with the oldest word in location 0. The TRIG LOC readout tells the location in Main Acquisition Memory of the data word that is labeled as the trigger word. This is the word that was acquired on the cycle on which the Main Acquisition Memory was triggered. If no data was acquired on the trigger cycle (because of data qualification), the last word stored before the trigger occurred will be labeled as the trigger. If no data was stored before the trigger, the word in location 0 will be labeled as the trigger.

The TRIG IN TEST readout tells which TEST was active on the cycle on which the Main Acquisition Memory was triggered.

If the 7D02 is stopped with the **STOP** key, the TRIG LOC readout will show three asterisks instead of a location. In this case there is no TRIG IN TEST readout and no trigger is displayed in the data itself. Instead, the LAST TEST readout tells which TEST was active on the cycle on which the STOP key was pressed. This display (below) will occur, even if the 7D02 had triggered, if the delay counter had not reached its programmed value at the time the **STOP** key was pressed.

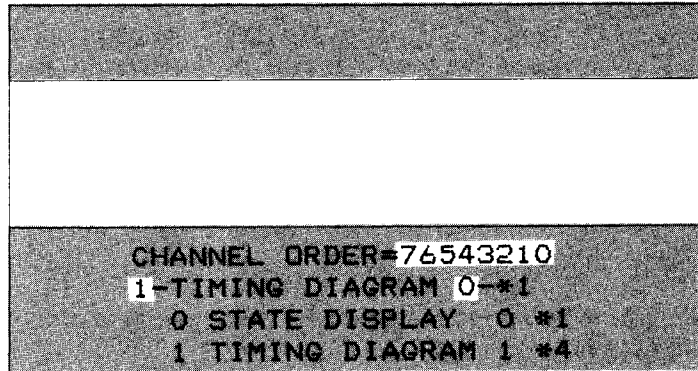
CTR1=10729 MS	TRIG LOC = ***
CTR2=00207 EVT	LAST TEST = 2

2918-54

2. TIMING OPTION DATA DISPLAYS

Immediate and Central Areas

When Timing Option data is displayed, either from the Timing Option Acquisition Memory or from the Storage Memory, the following additional parameters appear in the Immediate area:



2918-55

They are shown here with both menus expanded. This will never happen in the actual display.

The CHANNEL ORDER field allows suppression (if an "X" is entered) or re-ordering of the Timing Option channels in the display. The same channel order applies to both the Timing Diagram and the State Display.

The "STATE DISPLAY/TIMING DIAGRAM" menu determines the form of the Timing Option data display. If "0" is selected, the data is displayed as a State Table. If "1" is selected, the data is displayed as a Timing Diagram (pseudo-waveform).

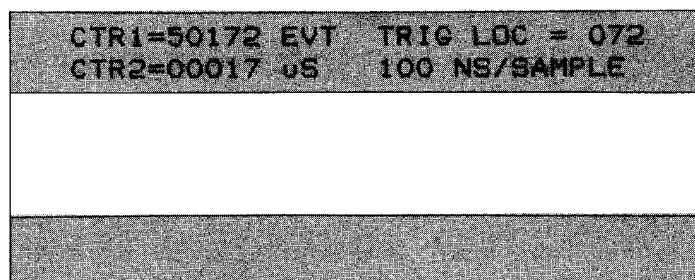
The "1*/4" menu appears only if the Timing Diagram is being displayed. When Timing Option data is displayed as a timing diagram, the data can be displayed in a compact form with 124 samples on the screen at a time ("*1") or in a more expanded form with only 31 samples on the screen ("*4").

NOTE

Detailed information about the Timing Option displays can be found in Section 6.

Status Area

The Status area contains the counter and trigger information gathered when the displayed data was acquired, and identifies the mode in which the Timing Option was used to acquire the data. The display in the Status area takes the following form for all Timing Option data displays:



2918-56

The fields on the left side of the Status area give the final values of the counters and the units in which they were counting. The five digits will be replaced with the text "OVRFL" when a counter overflows (exceeds a value of 65,534).

Data in memory is stored one word per memory location, beginning with the oldest word of data in location 0. The TRIG LOC readout tells the location in Memory of the data word that is labeled as the trigger word. This is the word that was acquired on the cycle on which the Timing Option Acquisition Memory was triggered. If the 7D02 was stopped with the **STOP** key, the TRIG LOC readout will show three asterisks instead of a location.

The word "SYNCHRONOUS" will appear on the right side of the second line if the data was acquired synchronously; if the data was acquired asynchronously, the sample period that was used will be displayed there.

3. STORED PROGRAM DISPLAYS

Immediate Area

Only the "DISPLAY ← STOREMEM" command appears in the Immediate area when a stored program is displayed. There are no parameters.

Central Area

Programs in Storage Memory are displayed in the Central area of the screen in the same format as programs stored in Program Memory. However, since the blinking cursor remains in the Immediate area during Display mode, none of the menus in the program will be expanded. The **DATA SCROLLING** keys must be used instead of the **CURSOR** keys to scroll through the stored program.

A program in the Storage memory cannot be modified or run. To modify or run the program, it must be transferred back to the Program Memory (see the section on data movement in the following Immediate mode discussion).

Status Area

The Status area is blank except when an incorrect keystroke causes an error message to appear.

Valid Keys

The **CURSOR**, **DON'T CARE**, and digit keys are used to modify parameters of the display command in the Immediate area.

The **DATA SCROLLING** keys move the viewing window through the memory being displayed. See the following discussion of scrolling.

The **START**, **IMMEDIATE**, and **FORMAT** keys put the 7D02 in Run, Immediate, and Format modes, respectively.

All other keys are invalid in Display mode and will cause an error message to appear in the Status area.

See Section 5 for details on the function and use of all keys.

Scrolling

It is rarely possible to show the entire contents of an Acquisition Memory or of the Storage Memory on the screen at once. A "window" the size of the Central area of the screen can be moved through the memory with the **DATA SCROLLING** keys.

The window is based on a pointer called the "data cursor". For the Timing Diagram, the data cursor is represented by the vertical line displayed through the center of the screen. For the Timing State display it is represented by a horizontal bar across the data. For all Main Acquisition data displays and for stored Program displays, it is implied to be on the top line of data or program on the screen.

There is a different data cursor maintained for each of the memories (Main Acquisition Memory, Timing Acquisition Memory and Storage Memory). The **DATA SCROLLING** keys control the data cursor for the memory being displayed. The **DATA SCROLLING** keys move the data cursor (and the window) in the direction of the arrow on the key; e.g., the **DATA SCROLLING** key moves the data cursor toward the beginning of the memory.

Two additional features of scrolling are available only when displaying data: acceleration and wraparound. Scrolling normally takes place one data word at a time. However, a **DATA SCROLLING** key will repeat if held down; if there are more than thirty words of data in the memory, after the first ten scrolls, scrolling will take place ten words at a time. When either end of the memory is reached, the data cursor wraps around to the other end of the memory.

When the 7D02 is run and new data is acquired, an attempt is made to maintain the location of the data cursor relative to the trigger word. This allows the user to see data at the same offset from the trigger on successive acquisitions without having to scroll. For instance, assume the trigger word was at location 128 in Acquisition Memory and the data cursor was at 138. If the 7D02 is run and the new trigger is at location 16, the data cursor is placed at location 26. When it is not possible to maintain the relative position of the data cursor (because one of the acquisitions did not contain a trigger, or because the trigger word is too close to one end of memory), the data cursor is put as close to the previous offset as possible, or on the trigger word.

HIGHLIGHTING OF MEMORY DIFFERENCES

If Display mode is enabled via a menu selection in Format mode (described below), differences between data stored in Storage Memory and data in Acquisition Memory can be highlighted in inverse video.

Difference highlighting occurs **only** if the Storage Memory contains data (not if it contains a program). Main data will only be compared to Main data, and Timing data will only be compared to Timing data. If the Storage Memory is displayed, it is compared to either the Timing Option Acquisition Memory or the Main Acquisition Memory, depending on the Storage Memory contents. If an Acquisition Memory is displayed, difference highlighting occurs only if the Storage Memory contains the same type of data.

Data in a memory is stored one word per memory location, beginning with the oldest word of data in location 0. Highlighting is on a location-by-location basis—location 0 in the Storage Memory is always compared to location 0 in the Acquisition Memory, etc. It is possible for the two memories to contain different amounts of data; in this case, those locations for which only one memory contains data will always be highlighted.

Highlighting of Main data is done in the same manner for both the Absolute and Mnemonic displays. If there are any differences at all between two words of data, **all** of the information displayed for the word of data (even those fields which are not different) will be highlighted.

Highlighting of Timing data depends on the type of display. In a State display, comparison will be on a digit-by-digit basis with only the different digits being highlighted. In a Timing diagram, only differences in the cursor readout will be highlighted (i.e., the word displayed under the data cursor is compared to the word in the same location in the reference memory).

IMMEDIATE MODE

Immediate mode is entered from either Programming mode or Display mode by pressing the **IMMEDIATE** key.

In Immediate mode, the user can issue commands that are executed immediately, as opposed to those that are executed only when a program is run. The commands fall into three categories: the GOTO command, data movement commands, and display commands.

Execution of a command causes a transfer into a new mode of operation. Execution of a GOTO causes Run mode to be entered; execution of a display command causes either Programming or Display mode to be entered; execution of a data movement command causes the 7D02 to return to the mode from which Immediate mode was entered.

DISPLAY CONTENTS

The contents of the Central Display area of the screen are not changed during Immediate mode. They will be the same as they were before the **IMMEDIATE** key was pressed.

The same is true of the Status area, except when an invalid key is pressed or an invalid sequence is entered. At those times, the contents of the Status area are temporarily replaced by an error message.

When Immediate mode is entered, the Immediate area is cleared and the blinking cursor appears there. Text associated with the data movement keys and the **GOTO** key will appear at the cursor location as the keys are pressed. If the command includes any parameters, they can be modified in the usual manner with the **CURSOR** and digit keys. Some commands cause the word "EXECUTE" to appear in the Immediate area; these commands are not actually executed until the cursor is moved onto the EXECUTE field with either the **CURSOR** key or the **CURSOR** key.

VALID KEYS

The user can transfer data into and out of several data areas within the 7D02. These are represented by the Data Movement keys as follows:

- DISPLAY** — The CRT display screen
- ACQ MEM** — Acquisition Memory
- PROGRAM** — Program Memory
- STORE MEM** — Storage Memory

If the Timing Option is installed and the **ACQ MEM** key is pressed, a menu appears allowing a choice of Main Acquisition Memory or Timing Option Acquisition Memory. Data movement and display commands are entered by pressing two Data Movement keys in succession. The first represents the destination of the move; the second represents the source. Text associated with the keys appears in the Immediate area with an arrow pointing from the source to the destination.

The **CURSOR** and digit keys are used to modify parameters associated with an Immediate mode command.

The **GOTO** key can be used to enter an immediate GOTO command. See the discussion of GOTO commands below.

Every time the **IMMEDIATE** key is pressed, the contents of the Immediate area are deleted and a new command can be entered. The **IMMEDIATE** key can therefore be used to delete an erroneous keystroke in Immediate mode. (The **DELETE** key only operates on Program Memory, during Programming mode.)

The **START** key causes the 7D02 to enter Run mode. Any incomplete Immediate mode command is ignored. A data movement command or GOTO command is ignored if the cursor has not yet been moved onto the EXECUTE field.

The **FORMAT** key causes the 7D02 to enter Format mode.

All other keys are invalid in Immediate mode and will cause an error message to appear in the Status area.

See Section 5 for details on the function and usage of all keys.

VALID IMMEDIATE MODE COMMANDS

GOTO Commands

The START key always begins execution of the current program with Test 1. The GOTO command allows the user to enter Run mode beginning with any test in the program.



Enter a GOTO command by pressing the **IMMEDIATE** key after the **GOTO** key. The command appears in the Immediate area, followed by a single parameter and an EXECUTE field. The parameter is the number of the test with which to begin execution. It defaults to the test which was last active when the 7D02 was last run. The command is not executed until the cursor is moved onto the EXECUTE field.





Note that if the key sequence **IMMEDIATE** **GOTO** **3** is entered, and the **START** key is pressed before the cursor is moved onto the EXECUTE field, the 7D02 will start with Test 1, not Test 3.

Data Movement Commands

A data movement command causes information to be transferred from a source memory to a destination memory. The old contents of the destination memory are usually lost. A data movement command is entered with two keystrokes: the destination memory followed by the source memory. There are three valid data movement commands, entered with the following keystroke sequences:



When the source of a data movement command is entered, the word "EXECUTE" appears in the Immediate area. The command is only executed when the blinking cursor is moved onto the EXECUTE field with either the **CURSOR**  key or the **CURSOR**  key. Once the command is executed, the 7D02 returns to the mode it was in before the **IMMEDIATE** key was pressed.

The "PROGRAM  STORE MEM" and "STORE MEM  PROGRAM" commands are displayed with a two-headed arrow between the destination and the source. This arrow implies that the contents of the Storage Memory will be swapped with those of the Program Memory if the Storage Memory already contains a program. If the Storage Memory does not already contain a program, the "STORE MEM  PROGRAM" command will cause the contents of the Storage Memory to be replaced with those of the Program Memory, and the "PROGRAM  STORE MEM" command will cause only an error message to appear in the Status area.

Display Commands

Display commands are essentially data movement commands in which the 7D02 display screen is specified as the destination. Display commands do not alter the contents of any of the memories. There are three valid display commands, entered with the following keystroke sequences:



Display commands are executed as soon as the second keystroke in the sequence (the source) is entered. The contents of the source memory are displayed on the Central portion of the display, and a mode change takes place. If the source is Program memory, Programming mode is entered; otherwise, Display mode is entered.

FORMAT MODE

Format mode can be entered by pressing the **FORMAT** key whenever the 7D02 is not in Run mode. The current mode is temporarily suspended and the entire display changes. Pressing the **FORMAT** key a second time causes the 7D02 to return to the suspended mode.

Format mode gives the user the ability to change the following characteristics of the program and data displays: (1) radices of numeric fields, (2) Timing Option glitch display, (3) memory difference highlighting, and (4) channel inversion.

DISPLAY CONTENTS

The contents of the Status area are replaced by a message indicating that Format mode is active.

The Immediate area disappears completely.

The Format display appears in the Central area of the screen. This area will contain menus and numeric fields that control various aspects of the program and data displays. These parameters can be examined and modified with the blinking cursor which appears in this area.

The Format display generally consists of 2-line entries. The first line in each pair contains a text string describing the function controlled by the menu or numeric field which appears on the second line of the pair. The exact contents of the Format display depend on the attached Personality Module and on the presence or absence of the Timing Option.

An example of the display for a 7D02 with a 6802 Personality Module (PM-103) and the Timing Option installed as shown in Figure 3-2. The first three parameters control the radices of numeric fields that appear in programs. The next three control radices of numeric fields in acquired data displays. The last three control the Timing Option glitch display, memory difference highlighting, and Timing Option channel inversion, respectively. The items beginning with the words "TIMING OPTION" will not appear in the display if the Timing Option is not installed.


```

TIMING OPTION WORD RECOGNIZER
 0-BINARY
WORD RECOGNIZER ADDRESS FIELD
 2-HEX
WORD RECOGNIZER DATA FIELD
 2-HEX
TIMING OPTION DATA DISPLAY
 0-BINARY
ADDRESS FIELD DISPLAY
 2-HEX
DATA FIELD DISPLAY
 2-HEX
HIGHLIGHT MEMORY DIFFERENCES?
 1-NO
DISPLAY GLITCHES?
 0-YES
TIMING OPTION DATA INVERSION
DATA=00000000

```

2918-57

VALID KEYS

The **CURSOR** and digit keys can be used to modify the parameters in the Format display. If the Format display does not all fit onto the screen at one time, it will automatically scroll when the cursor is moved.

The **FORMAT** key allows the user to leave Format mode.

All other keys are invalid in Format mode and will cause an error message to appear in the Status area.

See Section 5 for details on the function and use of all keys.

FEATURES CONTROLLED IN FORMAT MODE

The display in Figure 3-2 contains examples of all four classes of features that are controlled in Format mode. The features are discussed in the order in which they will always appear in the Format display.

Radices

"Input fields" are user-changeable numeric fields (parameters) that appear in programs and in the Format display (e.g., word recognizer fields). "Output fields" are numeric fields that appear in acquired data displays. The radices of some input and output fields can be changed via menu selections made in Format mode. If the Timing Option is installed, the radices of the data portions of the Timing Option word recognizer and the Timing Option data displays are under user control. The other fields whose radices can be changed are determined by the attached Personality Module.

Modes of Operation—7D02 Operators

The input fields always appear first in the Format display, followed by the output fields. Valid radices for input fields are binary, octal, and hexadecimal. Valid radices for output fields are binary, octal, hexadecimal, and 7-bit ASCII. (For a list of the ASCII conversions, see Appendix H.)

Radices for input and output fields are entered independently. For example, it is possible for the word recognizer ADDRESS field and the address portion of the acquired data to be displayed in different radices. To change the radices of both ADDRESS input and output, two separate menus must be used.

If a word recognizer field is displayed in a radix other than binary, each digit represents three (in octal) or four (in hexadecimal) channels. In different radices, different channels may be grouped together into each digit. It is possible, after a radix change, to find that channels for which X's were entered are grouped with channels for which values (0 or 1) were specified. This is legitimate, but it makes it impossible to display a digit in this position. When "care" and "don't-care" channels become mixed in one digit position, a "\$" will be displayed in that position. Since each binary digit represents a single channel, "\$" will never appear in a binary field.

Depending on the radix selected, the high-order digit in a word recognizer field may have restricted valid values. This will be true if the number of channels represented by an octal field is not a multiple of three, or if the number of channels represented by a hexadecimal field is not a multiple of four. For example, an 8-bit field in hexadecimal can take on all values between 00 and FF inclusive (00000000—11111111), but if the radix is changed to octal it can only take on the values 000 through 377. In the latter case, an attempt to enter a high-order digit in the range 4—7 will be detected as an error.

If acquired Timing Option data is displayed in a radix other than binary, the value displayed depends on the current channel order selected in Display mode. A binary value is first found by ignoring the "don't-care" channels and using the values (0 or 1) of the other channels in the order specified. The number is padded to 8 bits with high-order 0's (one for each "don't-care" channel). This 8-bit number is converted to the target radix. If that radix is ASCII, the high-order ("parity") bit is first stripped off.

Memory Difference Highlighting

A "YES/NO" menu controls the enabling/disabling of memory difference highlighting. The power-up default is "1-NO" (don't highlight differences). The effects of selecting "0-YES" are described in the Display mode discussion above.

Glitch Display

When the Timing Option is used asynchronously, it stores and displays glitch information along with data on all eight channels. Display of glitch information in Timing Option data displays can be controlled with a "YES/NO" menu in Format mode. The default is "0-YES" (display glitches). If the Timing Option is not installed, this menu will not appear in the Format display.

Channel Inversion

Each channel of the Timing Option can be independently inverted, as can each channel of the General Purpose Personality Module (PM-101).

Inversion of a channel means that:

1. Values entered for that channel in a word recognizer are complemented before being used (e.g., if the user tells the word recognizer to look for a 1, it really looks for a 0); and
2. Data gathered for that channel is complemented before it is displayed (e.g., if the data was a 1, a 0 is displayed).

This enables the user to think in terms of a "positive-true" logic system when the channel is really "negative-true". This can be especially useful when the independent leads of the P6451 probe or the PM-101 Personality Module are attached to a mix of positive- and negative-logic channels; the user can invert one group and think in one system.

If the Timing Option is installed, a numeric field will appear in which the user indicates channels to be inverted with a 1, and channels to be treated normally with a 0. The radix in which the field is displayed is the same as that currently chosen for the Timing Option word recognizer.

If the PM-101 is attached, a multi-field display appears for inversion. The radices of the field will be the same as those currently chosen for the corresponding word recognizer fields. Again, channels for which a 1 is specified will be inverted.

If neither the Timing Option nor the PM-101 is installed, no channel inversion displays will appear in Format mode. The processor-specific Personality Modules will supply any necessary inversion to make the system positive-true.

Section 4

7D02 PROGRAMMING CONCEPTS

TABLE OF CONTENTS

INTRODUCTION

EVENTS

COMMANDS

COMPOUND EVENTS

COMPLEMENTED EVENTS

COMPOUND COMMANDS

IF-THEN-ELSE

TESTS

CONFLICTING COMMANDS

CONTRADICTIONARY COMMANDS

DATA QUALIFICATION

PROBLEM SOLVING

7D02 PROGRAMMING CONCEPTS

INTRODUCTION

This section presents a general philosophy for programming the 7D02. The basic programming concepts will be explained and then developed into a program example. It is recommended that the user read Chapter 1, The Learning Guide, in conjunction with this chapter.

The sequence of events leading up to a problem in a digital system can be very complex. The 7D02 user language provides a means for describing this sequence and for acquiring the data necessary to find the cause of the problem when the sequence occurs.

The problem can usually be detected by creating a program consisting of a series of up to four TESTs. Each TEST is constructed with an IF-THEN-ELSE syntax. This allows the user to define events related to the occurrence of the problem and to execute commands which acquire data. The TESTs can be enabled and disabled as necessary to detect the sequences leading to the problem.

The keys used to enter a program can be broken into four groups: event keys, command keys, structure keys and the **IF**, **OR** and **NOT** keys. Some keys may overlap into more than one group (for example the **COUNTER** key is both an event key and a command key). The keys in each group are usually located together on the keyboard. The **QUALIFY** key is not grouped with the command keys, but may be used as a command key. Pressing one of these keys in a valid context will generate one or more lines in the program.

EVENTS

An event is a condition, detectable by the 7D02, that is either TRUE or FALSE. A simple event in the 7D02 consists of either a word recognition (using one of the four word recognizers) or a counter reaching a user-defined value. A word recognizer allows the user to detect the occurrence of a specified value on the system-under-test bus (see **WD RECOGNIZER**, Chapter 5). A counter allows the user to detect a user-defined length of time or to detect a user-defined number of counts (see **COUNTER**, Chapter 5).

Examples of Events:

```
WORD RECOGNIZER # 1
DATA=XX
ADDRESS=440F
/NMI=X /IRQ=X FETCH=X R/W=1
BA=X INVAL OP=X EXT TRIG IN=X
TIMING WR=X
```

This event becomes TRUE when address 440F is detected on a read cycle.

```
COUNTER # 1 = 00100 0-MS
```

This event becomes TRUE when the counter reaches 100 ms.

COMMANDS

A command is an action to be taken by the 7D02 when a particular event becomes TRUE. These actions perform the tasks of: acquisition completion (the TRIGGER command), data storage (the QUALIFY command), program sequencing (the GOTO command) and counter control (the COUNTER command).

Examples of Commands:

GOTO 2

This command allows a new set of event-command clauses to be activated.

QUALIFY

This command allows selected data to be stored.

COUNTER # 1 0-EVENTS

0-INCREMENT

This command increments a counter.

TRIGGER 0-MAIN

0-BEFORE DATA

This command causes the Acquisition Memory to stop storing data after the specified delay.

0-SYSTEM UNDER TEST CONT.

0-STANDARD CLOCK PROC.

COMPOUND EVENTS

A compound event consists of one or more simple events enclosed within brackets. If more than one simple event is included within the brackets, each simple event must be separated by an **OR** keystroke. The compound event is enclosed within brackets by pressing the **[]** key following an "IF" prompt, an "OR IF" prompt, or a "STORE ON" prompt and then pressing the **[]** key again when all of the desired simple events have been entered.

A compound event is either TRUE or FALSE. It is TRUE whenever one or more of the simple events within it is TRUE. It is FALSE only when all of the simple events within it are FALSE. Compound events may not contain other compound events (i.e., they may not be nested).

Example of a compound event:

WORD RECOGNIZER # 1

DATA=XX

ADDRESS=1234

/NMI=X /IRQ=X FETCH=X R/W=X

TIMING WR=X

OR

COUNTER # 1 = 00100 2-MS

This compound event becomes TRUE when either address 1234 is detected or the counter reaches 100 ms.

COMPLEMENTED EVENTS

A simple event within a compound event may be complemented by preceding the event with the **NOT** key. The complemented event will be TRUE whenever the simple event is FALSE. The **NOT** key may be used with either a word recognizer or a counter. A compound event cannot be complemented with the **NOT** key; i.e., the NOT may not appear outside of a bracket.

Example of a complemented event:

```
NOT
WORD RECOGNIZER # 1
DATA=7F
ADDRESS=XXXX
/NMI=X /IRQ=X FETCH=X R/W=X
BA=X INVAL OP=X EXT TRIG IN=X
TIMING WR=X
```

This event becomes TRUE whenever the value 7F is not detected on the system-under-test data bus.

If the **NOT** key precedes a word recognizer, the complemented event will become TRUE when any one or more of the individual fields within the word recognizer becomes FALSE. If the **NOT** key precedes a counter, the complemented event is TRUE until the counter reaches its user-defined value.

COMPOUND COMMANDS

A compound command consists of one or more commands enclosed within brackets. All commands within a compound command are executed together (simultaneously) whenever the event associated with them becomes TRUE. Compound commands cannot contain other compound commands (i.e., they cannot be nested).

Example of a Compound Command:

```
TRIGGER 0-MAIN
0-BEFORE DATA
0-SYSTEM UNDER TESTS CONT.
0-STANDARD CLOCK QUAL.
GOTO 2
COUNTER # 1 0-EVENTS
0-INCREMENT
```

When this compound command is executed, the 7D02 will trigger the Main Acquisition Memory, go to TEST 2, and increment counter 1 simultaneously.

IF-THEN-ELSE

The 7D02 user language uses an IF-THEN-ELSE syntax to condition the execution of a command on the occurrence of an event. Events (simple or compound) are entered under an "IF" or an "OR IF" prompt. Commands (simple or compound) are entered under a "THEN DO" prompt or under an "ELSE DO" prompt.

IF-THEN-ELSE Syntax:

```

IF
  Event
THEN DO
  Command
OR IF
  Event
THEN DO
  Command
OR IF
  Event
THEN DO
  Command
:
:
ELSE DO
  Command
    
```

The "IF", "THEN DO", and "OR IF" prompts are supplied automatically by the 7D02 whenever events or commands are entered (following the syntax shown in the example above). If an else clause is desired, it can be entered by pressing the **ELSE** key whenever the cursor is resting immediately below an "IF" or an "OR IF" prompt. This causes the "IF" or "OR IF" prompt to be replaced by the "ELSE DO" prompt. The command associated with the ELSE clause can then be entered.

Whenever the event associated with an "IF" prompt or an "OR IF" prompt becomes TRUE (on a given State clock), the command under the following "THEN DO" prompt is executed. If more than one event is TRUE on any particular State clock, all of the commands associated with those events are executed simultaneously. If none of the events are TRUE on a state cycle, then the command in the ELSE clause is executed (if one exists). The user can have as many "OR IF" — "THEN DO" clause pairs as necessary (subject to memory limitations). Only one ELSE clause can exist per TEST.

TESTS

A TEST is a structure that encloses the IF-THEN-ELSE syntax described above. The TEST allows a program to contain up to four separate IF-THEN-ELSE setups. When the program is run, each TEST will be either enabled or disabled (all events and commands within a disabled TEST are ignored). Only one TEST can be active at any one time.

A TEST is automatically created when a valid programming keystroke is entered into an empty program, after an "END TEST" keystroke or after an "END QUALIFY" keystroke (see also the "shortcuts" in Appendix B). A TEST is completed when an **END** keystroke is entered. A TEST takes the following form:

```

TEST 1
1 IF
1 Event
1 THEN DO
1 Command
1 OR IF
1 Event
1 THEN DO
1 Command
1 :
1 :
1 ELSE
1 Command
END TEST 1

```

The number following the "TEST" prompt and the number in the left column of the screen indicate the TEST number. There can be up to four different TESTs in a 7D02 program. These TESTs are automatically numbered from 1 to 4 as they are used. Each TEST can contain completely different combinations of events and commands (subject to resource limitations). Only one TEST can be active at a time. This means that if TEST 1 is active and an event in TEST 2 becomes TRUE, then the command associated with the event in TEST 2 will not be executed. A TEST becomes active by executing a GOTO command to that TEST (this also deactivates the TEST executing the GOTO command).

The following example shows a program containing two tests.

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=4325
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 GOTO 2
END TEST 1
TEST 2
2 IF
2 WORD RECOGNIZER # 2
2 DATA=XX
2 ADDRESS=694F

```

```

2 /NMI=1 /IRQ=1 FETCH=1 R/W=1
2 BA=1 INVAL OP=1 EXT TRIG IN=1
2 TIMING WR=1
2 THEN DO
2 TRIGGER 1-MAIN
2      1-BEFORE DATA
2      1-SYSTEM UNDER TEST CONT.
2      1-STANDARD CLOCK QUAL.
END TEST 2

```

In the above example, TEST 1 is initially active. (TEST 1 is active by default whenever the START key is pressed.) If address 694F is detected it is ignored. When address 4325 is detected the event in TEST 1 becomes TRUE and the GOTO to TEST 2 is executed. This deactivates TEST 1 and activates TEST 2. If address 694F is now detected, the Main Acquisition Memory will be triggered. However, if address 4325 is detected now, it will be ignored.

It is important to remember that all events within a TEST are evaluated simultaneously. Therefore, if a user wishes to detect a sequence of events (for example, the execution of subroutine A followed by the execution of subroutine B), two separate TESTs must be used. The first TEST will detect the occurrence of subroutine A. The second TEST will detect the occurrence of subroutine B and perform the desired function (e.g., count or trigger). The above example will detect the occurrence of address 4325 followed (at any time later) by address 694F.

If a TEST containing no GOTOs is activated, that TEST will remain active until the 7D02 stops acquiring data.

A TEST may be empty (i.e., it may have no events or commands defined within it). If an empty test is ever activated, the 7D02 will do nothing except continue to acquire data.

A TEST may also contain only an ELSE clause. In this case the command in the ELSE clause will be executed on every State clock in which that TEST is active.

CONFLICTING COMMANDS

If two events within an active TEST become TRUE at the same time, the commands associated with each event will be executed simultaneously. For example, a GOTO command, a TRIGGER command and a COUNTER command can all be executed at the same time (if their events all become TRUE at the same time). It is possible to specify two commands that conflict with each other, however. This condition is not an error and may actually be desirable in some programs.

The rule for the execution of conflicting commands is that the command defined first within the TEST will be executed and the one defined later in the TEST will not be executed. If the conflicting commands are defined within a compound command, only the conflicting commands will not be executed (the remainder of the commands within the compound command will be executed).

There are only two types of conflicting commands: GOTOs to different TESTs and two different commands for the same counter.

Example of conflicting commands:

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=31
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=1
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 GOTO 2
1 OR IF
1 WORD RECOGNIZER # 2
1 DATA=XX
1 ADDRESS=10XX
1 /NMI=X /IRQ=X FETCH=1 R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 GOTO 3
END TEST 1

```

This event is TRUE whenever the value 31 is detected on the data bus.

This command will always be executed when the above event is TRUE.

This event is TRUE for all addresses between 1000 and 10FF.

This command will not be executed when both events are TRUE.

In the above example, the command for word recognizer 1 is a GOTO to TEST 2. The command for word recognizer 2 is a GOTO to TEST 3. If both word recognizers become TRUE at the same time, (this will happen if the data value 31 ever occurs with an address in the 1000 to 10FF range) both GOTOs cannot be executed. These commands conflict because only one TEST may be active at a time. When both events become TRUE the GOTO to TEST 2 will be executed and the GOTO to TEST 3 will not be executed.

CONTRADICTIONARY COMMANDS

If two conflicting commands are included within a compound command they are considered contradictory and are flagged as an error. As described above, conflicting commands may or may not arise simultaneously during program execution. However, the act of including contradictory commands within the same bracket pair requires that they be executed simultaneously; since this can never be meaningful, it is not permitted.

Example of Contradictory Commands

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1 COUNTER # 1 0-EVENTS
E 1-RESET
1
END TEST 1

```

The error is indicated as a blinking "E" in the left column of the program line on which the first contradiction occurs, and a "CONTRADICTION BRACKETED COMMANDS" error message is displayed at the top of the screen. The 7D02 will not enter Run mode until this error is corrected (either by deleting the contradictory command or by changing its menu to a value that is non-contradictory).

DATA QUALIFICATION

A program may contain a data qualification block. This can be entered by pressing the QUALIFY key in an empty program or after an "END TEST" prompt. Only one QUALIFY block is allowed in a program. It appears as follows:

```

QUALIFY
QSTORE ON
Q Event
END QUALIFY

```

The event within the QUALIFY block may be a compound event and may contain one or more complemented events.

The QUALIFY block controls the storage of data into the Main Acquisition Memory (and into the Timing Option Memory if the Timing Option is used synchronously). Data is only stored in these memories on State clocks in which the event in the QUALIFY block is TRUE. The QUALIFY block is independent of the TESTs in a program (i.e., the QUALIFY block acts the same regardless of which TEST is currently active).

Data qualification can also be accomplished by using the QUALIFY key as a command within a TEST. In this case the data qualification is performed only when the TEST is active and the QUALIFY command is executed (see the discussion on the QUALIFY key in Section 5 for more details). If no QUALIFY block is specified and no QUALIFY commands are used, the 7D02 stores the data present on all State clocks. If either the QUALIFY block or the QUALIFY command is used, the 7D02 stores only when either the QUALIFY command is executed or the event within the QUALIFY block becomes TRUE.

Example of Data Qualification:

```

QUALIFY
QSTORE ON
Q WORD RECOGNIZER # 1
Q DATA=04
Q ADDRESS=XXXX
Q /NMI=X /IRQ=X FETCH=X R/W=X
Q BA=X INVAL OP=X EXT TRIG IN=X
Q TIMING WR=X
END QUALIFY

```

The data might be displayed as:

LINE	ADDRESS	OPERATION	/IRQ/NMI
000	4044	04 READ	10
001	6664	04 READ	10
002	552F	04 WRITE	10
003	6664	04 READ	10
004	4044	04 READ	10
005	552F	04 WRITE	10
006	4044	04 READ	10
007	6664	04 READ	10
008	4044	04 READ	10
009	552F	04 WRITE	10
010	4044	04 READ	10
011	6664	04 READ	10
012	4044	04 READ	10
013	6664	04 READ	10
014	4044	04 READ	10
015	6664	04 READ	10
016	4044	04 READ	10
017	6664	04 READ	10
018	4044	04 READ	10

In the above example, data is only stored in the Acquisition Memory on State cycles in which the value on the data bus is 04.

PROBLEM SOLVING

The ability to recognize events simultaneously and to execute various commands when these events become TRUE is a powerful debugging tool. Simultaneous recognition and execution does not lend itself to flow charts, however. The sequence of conditions leading up to the detection of a problem can usually be best expressed as a state transition diagram. The state transition diagram can then be translated directly into a 7D02 program.

A state transition diagram is used to describe a problem as a series of decision-making nodes or states. In each independent state, the set of possible inputs is the same. The actions based on those inputs may vary from one state to another. One of the actions taken may be a transition to another state or to the same state.

Each state is represented in the diagram as a circle. Transitions between states are shown as arrows from the old state to the new; each arrow is labeled with the input that causes the transition. Actions associated with the input are indicated by having the arrow pass through a labeled box on its way to its new state. All inputs that are not explicitly indicated are assumed to result in no action and no state change.

For example, it might be desired to trigger the Main Acquisition memory whenever address F82E is detected within 100 ms after address F4F0 is detected. The first step is to define a state in which the occurrence of F4F0 on the address bus is looked for. When this address is found, the counter starts to count the 100 ms time window.

The next step would be to look for address F82E. This must be done as part of a separate state since it is only valid as a triggering event when F82E is found **after** F4F0 is found. (If they were both looked for in the same state, the trigger would occur any time F82E was found.) If 100 ms pass before F82E is found, it is necessary to go back to state 1 and start the sequence over again. If F4F0 is detected a second time while counting the 100 ms time window, the window count must be restarted.

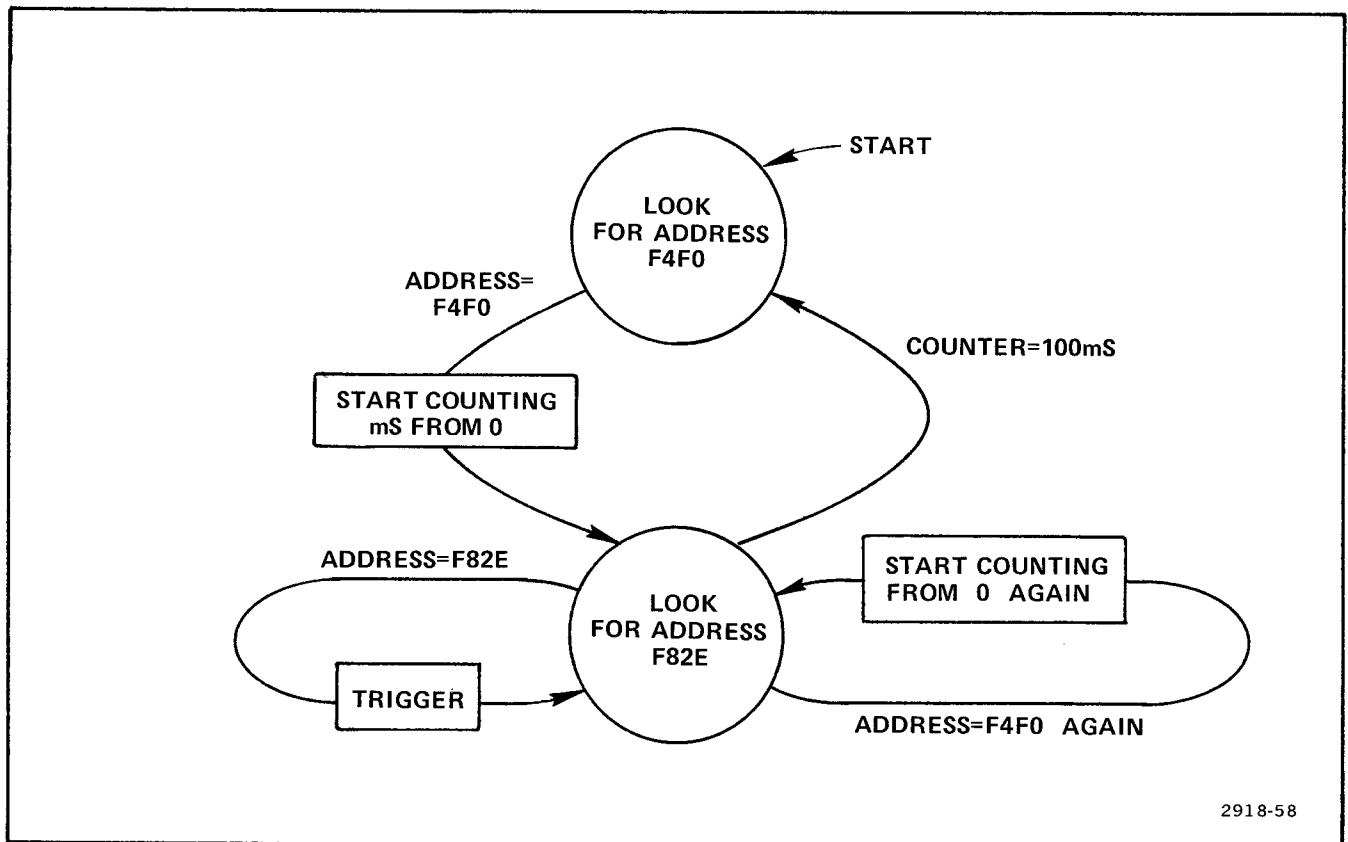


Figure 4-1. State transition diagram.

The labels on the arrows are the events to be looked for; the boxes are the actions to be taken. When an arrow leaves one state and enters a new one, a GOTO is necessary.

Translating the labels into the 7D02 syntax gives:

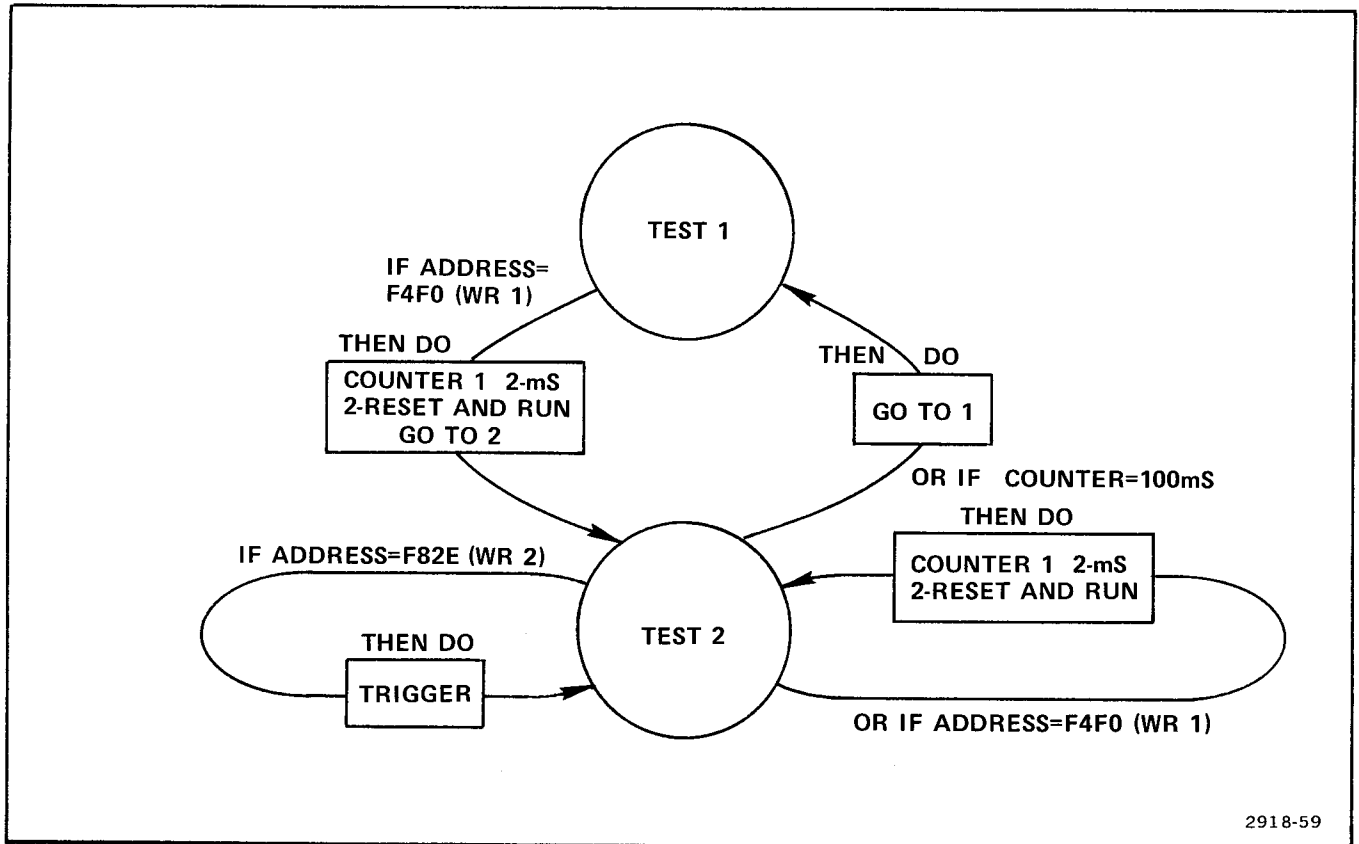


Figure 4-2. State transition diagram translated into 7D02 syntax.

Now the program can be virtually read off the diagram to appear as follows:

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=F4F0
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1
1 1-MS
1 2-RESET AND RUN
1 GOTO 2
1
END TEST 1
TEST 2
2IF
2 WORD RECOGNIZER # 2
2 DATA=XX
2 ADDRESS=F82E
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2THEN DO
2 TRIGGER 0-MAIN
2 0-BEFORE DATA
2 0-SYSTEM UNDER TEST CONT.
2 0-STANDARD CLOCK QUAL.
2OR IF
2 WORD RECOGNIZER # 1
2 DATA=XX
2 ADDRESS=F4F0
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2THEN DO
2 COUNTER # 1 2-MS
2 2-RESET AND RUN
2OR IF
2 COUNTER # 1 = 00100 2-MS
2THEN DO
2 GOTO 2
END TEST 2

```

Beginning address of sequence.

When the starting address is detected, start counting the 100 ms time period and go to state 2 to look for the second address in the sequence.

This is the second address in the sequence. If this is found before the counter reaches 100 ms, trigger the Main Acquisition Memory.

If the beginning address of the sequence is detected, reset the counter to begin the 100 ms count again.

If the counter reaches the 100 ms time limit, go back to TEST 1 and start the sequence over.

Section 5

7D02 KEY ENCYCLOPEDIA

TABLE OF CONTENTS

INTRODUCTION

ACQ MEM
COUNTER
CURSOR
DATA SCROLLING
DELETE
DISPLAY
DON'T CARE
ELSE
END
FORMAT
GOTO
IMMEDIATE
NOT
OR
PROGRAM
QUALIFY
START/STOP
STORE MEM
TRIGGER
WD RECOGNIZER
X
0—9, A—F
[]

7D02 KEY ENCYCLOPEDIA

INTRODUCTION

This section discusses the function and use of all of the keys on the 7D02 keyboard. The keys are discussed in alphabetic order.

Where examples of programs appear, a 6802 Personality Module (PM-103) is assumed and variable-radix fields are assumed to be set to hexadecimal. These assumptions will be reflected in the Word Recognizer displays only.

It is also assumed that the Timing Option is installed. It is always made clear where the installation of the Timing Option would change the function or performance of a key, but details about Timing Option parameters are not presented here. See Section 6 for details about the Timing Option.

Each key description is organized along the following lines:

1. Key Group

The name of the group of keys in which the key is found is given. The Data Movement, Event, Command, Structure, and Numeric Entry groups are delineated on the keyboard with colored borders. The remaining keys (**IMMEDIATE**, **[]**, **OR**, **NOT**, **DELETE**, **START/STOP**, **DATA SCROLLING**, **CURSOR**) are classified as Miscellaneous.

2. Mode of Operation

The mode(s) of operation (Programming, Run, Display, Immediate, Format) during which the key can legally be pressed are listed. See Section 3 for a description of the 7D02 modes of operation.

3. General Description

The purpose and use of the key is described.

4. Parameters

If pressing the key can cause additional menu or numeric fields to appear on the display, the purpose and valid values of each such parameters are described.

5. Notes

Any additional information or clarifications are given here.

6. Example(s)

Some key descriptions include one or more examples of 7D02 programs in which the key is used.

ACQ MEM

KEY GROUP

Data Movement.

MODE OF OPERATION

Valid in Immediate mode only.

GENERAL DESCRIPTION

The **ACQ MEM** key specifies the Acquisition Memory as the source of a data movement command in Immediate mode. If the Timing Option is installed, a menu will allow the user to distinguish the Main Acquisition Memory from the Timing Option Acquisition Memory. The **ACQ MEM** key is only valid in the following keystroke sequences:



The **IMMEDIATE** key puts the 7D02 into Immediate mode. The second keystroke in each sequence indicates the destination to which the contents of the Acquisition Memory are to be moved.

The first sequence causes the contents of the Acquisition Memory to be displayed on the CRT. It is executed as soon as the **ACQ MEM** key is pressed. It causes the 7D02 to enter Display mode, but leaves the "DISPLAY — ACQMEM" command in the Immediate area where the parameters of the display can be manipulated with the **CURSOR** and digit keys.

The second sequence causes the contents of the Acquisition Memory to be stored in Storage Memory. It causes an "EXECUTE" field to appear in the Immediate area. The command is not actually executed until the blinking cursor is moved onto the "EXECUTE" field. Until then, the parameters of the command can be manipulated with the **CURSOR** and digit keys. After execution, the 7D02 returns to the operating mode (Programming or Display) it was in before the **IMMEDIATE** key was pressed.

PARAMETERS

1. Displaying Acquisition Memory:

```

DISPLAY←ACQMEM 0-MAIN
                0 MAIN
                1 TIMING

0-ABSOLUTE
0 ABSOLUTE
1 MNEMONIC

```

```

0-MAIN
0 MAIN
1 TIMING

```

This menu appears **only** if the Timing Option is installed.

If "0" is entered or if the Timing Option is not installed, then Main Acquisition Memory will be displayed and the ABSOLUTE/MNEMONIC menu will appear (see below).

If "1" is entered, the Timing Option Acquisition Memory will be displayed and the ABSOLUTE/MNEMONIC menu (see below) will be replaced by a new set of parameters. See Section 6 and the Display mode discussion in Section 3 for details.

```

0-ABSOLUTE
0 ABSOLUTE
1 MNEMONIC

```

If "0" is entered here, the Main Acquisition Memory is displayed in Absolute format. Data will generally appear as numeric fields whose radices are changeable in Format mode.

If "1" is entered here, the Main Acquisition Memory is displayed in Mnemonic format. The value on the s.u.t. data bus on Fetch cycles is generally displayed as the vendor op-code for the target processor. Other mnemonics are supplied where possible (such as some control line values); other fields are displayed in a radix compatible with the target processor.

The exact formats and attributes of these two displays are Personality Module-dependent. See the relevant Personality Module manual and the Display mode discussion in Section 3 of this manual for details.

2. Storing Acquisition Memory:

```
STOREMEM ← ACQMEM
0-MAIN EXECUTE
0 MAIN
1 TIMING
```

```
0-MAIN
0 MAIN
1 TIMING
```

This menu appears **only** if the Timing Option is installed.

If "0" is entered or if the Timing Option is not installed, the contents of the Main Acquisition Memory will be copied into the Storage Memory.

If "1" is entered, the contents of Timing Option Acquisition Memory will be copied into the Storage Memory.

Notes

When the 7D02 comes out of Run mode (via execution of a Trigger command or by pressing the **START/STOP** key) a **DISPLAY ACQ MEM** command is automatically placed in the Immediate area, putting the 7D02 into Display mode. The user can manipulate the parameters of the command with the **CURSOR** and Digit keys.

COUNTER

KEY GROUP

Event or Command.

The **COUNTER** key falls into two key groups. The context in which it appears determines its use.

1. It is a Command key if it is entered into a program immediately following a "THEN DO" or "ELSE DO" prompt, or as part of a compound command.
2. It is an Event key if it is entered into a program immediately following an "IF", "OR IF", or "STORE ON" prompt, or as part of a compound event.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The 7D02 has two general-purpose counters. A 7D02 counter can count discrete events, or it can measure and/or accumulate time intervals. The numeric keypad is used to select the counter mode (i.e., events or time).

A counter command is actually executed only during normal program flow. Through it, the user can direct an event counter to increment or reset to 0, or can direct a time counter to begin accumulating time, to stop accumulating time, or to reset to 0 and begin accumulating time.

A counter event is defined in terms of a user-specified terminating value. Once the counter reaches the terminating value, the event becomes TRUE. The numeric keypad is used to program this value into the counter.

Both counters are initialized to a value of 0 each time a program is run.

PARAMETERS

1. Counter command.

```

COUNTER # 1 0-EVENTS
              0 EVENTS
              1 uS
              2 MS
0-INCREMENT  or  0-RUN
  0 INCREMENT      0 RUN
  1 RESET          1 STOP
                  2 RESET AND RUN
  
```

COUNTER #1

This numeric field is used to select one of the two counters. The default value is counter #1. Valid values are "1" and "2".

0-EVENTS

0 EVENTS

1 μ S

2 MS

The counter mode is programmed here.

Entering "0" selects event mode.

Entering "1" selects time mode, with a time unit of microseconds.

Entering "2" selects time mode, with a time unit of milliseconds.

0-INCREMENT

0 INCREMENT

1 RESET

This menu appears only if the event mode has been selected.

Entering "0" directs the counter to increment by 1 each time the command is executed.

Entering "1" directs the counter to reset to 0 each time the command is executed.

0-RUN

0 RUN

1 STOP

2 RESET AND RUN

This menu appears only when Timing mode has been selected.

Entering "0" directs the counter to start running (accumulating time) whenever the command is executed. The counter will then increment automatically, from whatever its previous value was, every time the specified time interval (microsecond or millisecond) elapses. Once the command is executed, the counter will continue to run until a COUNTER Stop command is executed. If the counter is already running, executing this command will have no effect on it (the counter just continues running).

Entering "1" directs the counter to stop running, and to retain its current value, whenever the command is executed. If the counter is already stopped when this command is executed, the command has no effect.

Entering "2" directs the counter to reset to a value of 0 and to start running each time the command is executed. If the counter is already running, it **will** reset to 0 and continue running.

2. Counter Event

```
COUNTER # 1 = 00002 0-EVENTS
                0 EVENTS
                1  $\mu$ S
                2 MS
```

COUNTER #1

This numeric field is used to select one of the two counters. The default value is counter #1. Valid values are "1" and "2".

= 00002

This is the terminating value of the counter. The counter event becomes TRUE when the counter reaches its terminating value. The value entered must be a decimal number between 00002 and 65534, inclusive. Attempts to enter a value outside this range will produce an error message.

```
0-EVENTS
  0 EVENTS
  1  $\mu$ S
  2 MS
```

The counter mode is entered here.

Entering "0" selects event mode.

Entering "1" selects time mode, with a time unit of microseconds.

Entering "2" selects time mode, with a time unit of milliseconds.

NOTES

General

1. Final counter values are displayed in the status area of the screen when acquired data is displayed.
2. The value of any counter that overflows will be displayed as "OVRFL". Note that time counters will overflow after 65,534 μ s or 65,534 ms, depending on the units selected with the Counter mode menu.
3. Each counter can be used in only one mode (event/microsecond/millisecond) in a program. Whether it is used in commands, events, or both, changing the mode in one instance of the counter will automatically cause it to be changed in all instances. When a given counter is changed from Event mode to Time mode, all commands for that counter are changed to "0-RUN". When a given counter is changed from Time mode to Event mode, all commands for that counter are changed to "0-INCREMENT".
4. A counter's mode and terminating value are reprogrammed **only** when the user enters new values into them with the numeric keypad. In particular, deleting all instances of a counter from a program will mean the counter is not used in the program, but will not reprogram the counter. If that same counter is subsequently entered into a 7D02 program, it will contain the same mode and/or terminating value it had before it was deleted. The power-up defaults are: mode = Events, terminating value = 00002.

5. Care must be taken in defining the State clock via clock qualification (see Section 7). Erratic State clocks will generate incorrect time measurements.

Counter Commands

1. Once the program is running, only one command per counter can be executed on each 7D02 cycle. If more than one command is indicated for a counter (either because two commands for a counter are included in a compound command or because two different events with associated counter commands come TRUE simultaneously) only the one which appears nearest the top of the TEST will be executed. In particular, note that this implies that the value of an event counter cannot be increased by more than 1 on a single cycle.

2. Counter commands will continue to be executed when encountered during normal program flow until all data acquisition stops. The mere act of triggering **does not disable them!** In particular, increment and reset commands will be executed if encountered, and time counters that are running will continue to run until explicitly stopped or until the 7D02 leaves Run mode.

Counter Events

1. The act of defining a counter event does not automatically cause the counter to increment or measure time when the program is run. The **COUNTER** key must also be used somewhere in the program as a command or it will be impossible for the event to ever come TRUE. (Of course, even if a command is included, the event coming TRUE will depend on the program it is used in and/or the data generated by the system-under-test.)

2. Each counter can be used to define only **one event per program**; i.e., the counter can be compared to only one terminating value in the entire program. This event may be used (tested for) any number of times in the program, but always with the same terminating value. Changing the value in one instance of the counter event will automatically cause it to be changed in all instances in the program.

3. If a counter is used to define an event, its value can never get larger than the specified terminating value! Further commands to increment an event counter or to run a time counter will have no effect on the counter's value unless the counter is reset to 0. Note that this implies the event "COUNTER # 1 = 00010" is really the same as "COUNTER # 1 \geq 00010"; the event becomes TRUE as soon as the value of Counter #1 reaches 10, and it remains TRUE until the Counter is reset.

4. A counter cannot be incremented to its terminating value and generate a TRUE event on the same cycle; i.e., the event will be recognized on the cycle **after** the counter reaches its terminating value.

5. If (1) a counter is used in time mode, and (2) its terminating value is shorter than the 7D02's State clock period, and (3) a "2-RESET AND RUN" command is used to start the counter, then it will take **two** State clocks (two 7D02 cycles) to detect the counter event. If "2-RESET AND RUN" commands for the counter are issued continuously, the event will **never** be recognized. (See Section 2 for a description of the State clock.)

EXAMPLE

```

TEST 1
1 IF
1 COUNTER # 1 = 00100 2-MS
1 THEN DO
1 GOTO 2
1 ELSE
1 COUNTER # 1 2-MS
1 0-RUN
END TEST 1

```

The first use of Counter #1 in the Test above is as an event; the second is as a command. The 7D02 will transfer to TEST 2 100 ms after the program is started.

Note that the COUNTER command in the ELSE clause will be executed on every cycle the program is executing in TEST 1 except the one on which the GOTO is executed. Only the first execution will have any effect (the counter will already be running on subsequent ones), but that first execution is necessary to get the Counter running.

CURSOR






KEY GROUP

Miscellaneous.


MODE OF OPERATION


Valid in Programming, Display, Immediate and Format modes.


GENERAL DESCRIPTION



The four  keys (, , , ) move the blinking cursor in the direction indicated by the arrow on the key.

The only valid positions for the blinking cursor are: (1) digits of numeric fields, (2) menus, (3) the EXECUTE field that appears after certain Immediate mode commands, and (4) the end of a program or Immediate mode command.

The  key moves the cursor up to the next line containing a valid cursor position and places the cursor on the leftmost position on that line. If there are no legal positions above the cursor, the cursor moves to (or remains on) the leftmost position on the current line.

The  key moves the cursor down to the next line containing a valid cursor position and places the cursor on the leftmost position on that line. If there are no legal positions below the cursor, the cursor moves to (or remains on) the rightmost position on the current line.




The  key moves the cursor to the next legal position to the left. If there are no more legal positions on the same line, it moves the cursor to the rightmost position of the line.

The  key moves the cursor to the next legal position to the right. If there are no more legal positions on the same line, it works like the  key.

PARAMETERS

None.

NOTES

1. All  keys repeat when held down.
2.  keys never move the cursor into the Immediate area from the Central Display area, or into the Central Display area from the Immediate area.
3. Often, a program or Format mode display will not fit on the screen. Moving the cursor with the  keys automatically causes the display to scroll until the top or bottom of the display is reached.
4. If the cursor is moved onto the EXECUTE field that follows some Immediate mode commands, the Immediate mode command is executed.

DATA SCROLLING

KEY GROUP

Miscellaneous.

MODE OF OPERATION

Valid in Display mode only.

GENERAL DESCRIPTION

During Display mode, the two **DATA SCROLLING** keys are used to scroll through the displayed memory (Storage Memory or one of the Acquisition Memories). The **CURSOR** keys cannot be used for this purpose because they are associated with the blinking cursor which remains in the Immediate area during Display mode (the blinking cursor can be used to modify parameters of the display command there).

Each **DATA SCROLLING** key is associated with two arrows. The ← and → arrows on the front panel are for the Timing Diagram, which scrolls from right to left, or left to right. The ↑ and ↓ arrows on the keys are for the other displays (Timing State, all Main Acquisition Memory displays, and stored Programs) which scroll from top to bottom, or bottom to top.

The ←/↑ key will scroll toward the beginning of data; the →/↓ key scrolls toward the end of data.

NOTE

*The arrows indicate the direction in which the data window will move. The data will actually appear to move in the opposite direction on the screen. This however, is consistent with the manner in which the **CURSOR** ↑ and **CURSOR** ↓ keys work.*

The data window is defined in terms of a data cursor, which indicates the viewer's current position in the viewed memory. The Timing displays contain explicit data cursors; the data cursor for Main Acquisition Memory displays and stored-program displays is implied to be at the top line of the display.

NOTE

The data cursor is not the same as the blinking cursor used for parameter entry and menu selection.

PARAMETERS

None.

NOTES

General

1. The **DATA SCROLLING** keys repeat if they are held down.

Scrolling Through Acquired Data

1. Initially, data will scroll one word at a time. If there is a large amount of data and a **DATA SCROLLING** key is held down, after the first 10 words have been scrolled the scrolling rate changes to 10 words at a time.
2. Data displays will wrap around. Scrolling "up" from the first item in memory will cause the last item to appear. Scrolling "down" from the last item will cause the first to appear.

Scrolling Through a Program in Storage Memory

1. Scrolling is always one program line at a time.
2. A program can never be modified when it is in Storage Memory. Merely displaying a program from Storage Memory does not make it modifiable; the blinking cursor remains in the Immediate area and the **DATA SCROLLING** keys must be used to view the program. In order to modify the program, it must be moved to Program memory.

DELETE




KEY GROUP

Miscellaneous.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION









The DELETE key removes unwanted items from the end of the program currently in Program Memory. Generally, a program "item" consists of all of the lines in a program that were associated with a single programming keystroke. (The programming keys are the Event keys, Command keys, Structure keys, , , and .)

The DELETE key will only work when the blinking cursor is at the end of the program or on one of the lines associated with the last item in the program.

PARAMETERS

None.

NOTES

1. The  key will repeat if held down. To delete the entire program in Program Memory, move the blinking cursor to the end of the program and hold the  key down.
2. The  key will **not** reprogram the Counters, Word Recognizers, and Triggers when it removes them from the program. Generally, it will not return parameters to their power-up defaults; the next time they are used they will contain the same values as they had before being deleted. See the encyclopedia entries for the , , and  keys for details.
3. Items can never be deleted from a program in Storage Memory. In particular, merely displaying a program from Storage Memory does not make the  key work; the program must be transferred to Program Memory.
4. To delete a command in the Immediate area, use the  key.

DISPLAY

KEY GROUP

Data Movement.

MODE OF OPERATION

Valid in Immediate mode only.

GENERAL DESCRIPTION

The **DISPLAY** key specifies the CRT as the destination of a data movement command in Immediate mode. The contents of the source memory will be displayed in the central display area of the screen. The **DISPLAY** key is only valid in the following keystroke sequences:

IMMEDIATE	DISPLAY	ACQ MEM
IMMEDIATE	DISPLAY	PROGRAM
IMMEDIATE	DISPLAY	STORE MEM

The last keystroke in each sequence is the source for the move. As soon as the source key is pressed, the source memory will be displayed on the screen and a mode change will take place. The 7D02 enters Programming mode if the source is **PROGRAM**. The 7D02 enters Display mode if the source is **ACQ MEM** or **STORE MEM**.

See Section 3 for more information.

PARAMETERS

None.

NOTES

None.

DON'T CARE

KEY GROUP

Numeric Entry.

MODE OF OPERATION

Valid in Programming and Display modes of operation.

GENERAL DESCRIPTION

The **DON'T CARE** key can be entered into certain numeric fields related to the Personality Module and Timing Option channels to indicate that the user is not concerned with the values on the corresponding channels.

The **DON'T CARE** key displays as an "X".

PARAMETERS

None.

NOTES

1. The **DON'T CARE** key will repeat if it is held down.
2. In Programming mode, the **DON'T CARE** can be entered in place of any digit(s) in a Word Recognizer pattern. This indicates that the user is not concerned with the data seen on the corresponding channels; the Word Recognizer will be considered to always match incoming data on the channel(s) regardless of the actual value(s) of the data.
3. Some Personality Modules permit programming of clock qualification (see Section 7). Menus and numeric fields are provided as part of the display generated with the **TRIGGER** key. The **DON'T CARE** can be used in the numeric fields to indicate control lines **not** to be used in deriving the 7D02 State clock from the system-under-test's raw clock.
4. In Timing Option displays, a "Channel Order" field appears in the Immediate area. If a **DON'T CARE** is entered, the corresponding trace in the timing diagram display will be left blank; the data/glitch words in the timing state display will be one channel shorter for each **DON'T CARE** entered.
5. The number of channels (bits) affected by a **DON'T CARE** in a Word Recognizer depends on the radix of the field in which it is entered; e.g., one in a binary field, four in a hexadecimal field. Changing the radix (e.g., from binary to hex) could cause a digit in the new radix to encompass some channels that are "DON'T CARE" and some that are not. Digits representing a mixture of "DON'T CARE" and "CARE" channels are displayed as "\$".

ELSE

KEY GROUP

Structure.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The ELSE key replaces an event in a program and indicates that the following command or compound command is to be executed whenever **none** of the events in the Test are TRUE. It may be entered as the first keystroke in a Test, or whenever an "IF" or "OR IF" prompt appears in a program.

The ELSE clause can be used only once in each Test. It must be the **last** clause in the Test; i.e., no events may follow the ELSE clause in that Test.

PARAMETERS

None.

NOTES

If a Test contains nothing but an ELSE clause, the command(s) associated with the ELSE clause will be executed on every cycle on which the 7D02 stays in the Test. (This means every cycle until the Program stops running, unless the clause includes a GOTO to another Test.)

END

KEY GROUP

Structure.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The **END** key is used to terminate a Test or Qualify block in a program.

Subsequent legal keystrokes will automatically be entered into the next unused Test, or into the Qualify block if the **QUALIFY** key is pressed after the **END** key.

PARAMETERS

None.

NOTES

An **END** keystroke need not be entered at the end of the last block (Test or Qualify) in the program. If omitted, it will be supplied automatically when the program is run.

FORMAT

KEY GROUP

Numeric Entry.

MODE OF OPERATION

Valid in Programming, Display, Immediate and Format modes.

GENERAL DESCRIPTION

The **FORMAT** key can be pressed whenever the 7D02 is not in Run mode. If the 7D02 is in Programming, Display, or Immediate mode, it will go into Format mode. If the 7D02 is already in Format mode when the **FORMAT** key is pressed, it returns to the mode it was in before Format mode was entered.

During Format mode, a special display replaces whatever else was on the screen. This display contains menus and numeric fields that allow the user to change the form in which data can be entered or displayed on the 7D02. The exact display varies; it is dependent on the Personality Module attached to the 7D02 and on the presence or absence of the Timing Option.

In Format mode, the user can:

1. Change the radices of Word Recognizer fields, acquired data displays, etc.
2. Enable or disable the highlighting of differences between Storage Memory and Acquisition Memory in Display mode.
3. Enable and disable the display of acquired glitch information in the Timing Option data displays (if the Timing Option is installed).
4. Invert the Timing Option data channels (if the Timing Option is installed) and/or the system-under-test buses (if the General Purpose Personality Module, PM-101, is attached). Inversion of a channel means that word recognizer values entered for the channel are complemented before being used, and data gathered on the channel is complemented before being displayed.

For more information about Format mode, see Section 3.

PARAMETERS

None.

NOTES

None.

GOTO

KEY GROUP

Command.

MODE OF OPERATION

Valid in Programming mode and Immediate mode.

GENERAL DESCRIPTION

When executed, the GOTO command causes the 7D02 to transfer control to one of the tests in the current program. The specified test becomes active and all other tests in the program are deactivated.

When used in a program, the GOTO is executed when it is reached in normal program flow.

To use GOTO as an Immediate mode command, the user must enter the keystroke sequence **IMMEDIATE** **GOTO** (plus an optional test number). The command will not be executed until the blinking cursor is moved (with the → or ↓ key) onto the EXECUTE field that appears in the Immediate area; this gives the user the opportunity to change the test number that appears with the GOTO. The effect of executing the command is to put the 7D02 into Run mode, starting execution of the program currently in Program Memory at the specified test.

PARAMETERS

GOTO 1

The single digit parameter that appears when the **GOTO** key is pressed is the number of the test in the current program to which control is to be transferred. The digit entered must be in the range 1—4. If the GOTO is entered in Programming mode, the default is the current test number plus one. (Exception: "4" is the default in Test #4.) In Immediate mode, the default is the number of the Test in which the 7D02 stopped the last time it was run.

NOTES

GOTO in a Program

1. If a "GOTO n" command appears in a program and "n" is the number of a test which has not been defined in the program, a dummy test will be provided. It will be equivalent to:

```
TEST n  
END TEST n
```

(This test will not actually appear in the program.) The 7D02 will do nothing in this test except acquire data (subject to the data qualification block in the program, if any) and continue incrementing the delay counter(s) for any memories that were triggered prior to entering this test. It will never leave this test, except by stopping the program, once the GOTO is executed.

2. A "GOTO n" command in Test "n" is a "no-operation". It can be used when it is desired to stay in a test (and carry out no other activity) whenever a specified event occurs.

GOTO as an Immediate Mode Command

1. If the user attempts to execute a "GOTO n" where "n" is the number of a test that has not been defined in the program in Program Memory, an error message will be displayed in the status area. The 7D02 will return to the operational mode (Program or Display) in which it was before the user pressed the **IMMEDIATE** key.

IMMEDIATE

KEY GROUP

Miscellaneous.

MODE OF OPERATION

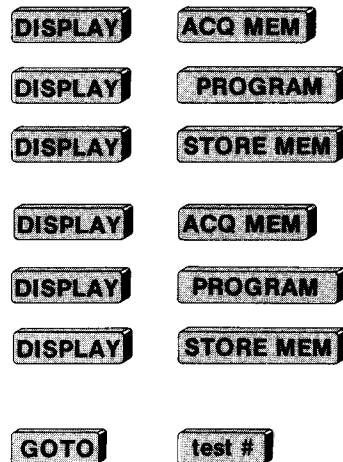
Valid in Programming and Display modes of operation.

GENERAL DESCRIPTION

Pressing the **IMMEDIATE** key puts the 7D02 into Immediate mode; i.e., makes it ready to execute a command immediately (as opposed to entering one into a program).

Every time the **IMMEDIATE** key is pressed, the command currently in the Immediate area (at the bottom of the screen) is deleted. The blinking cursor appears at the beginning of the area, indicating readiness to accept a new command.

In Immediate mode, Data Movement and GOTO commands can be entered. Data movement is further subdivided into non-destructive (Display) and destructive (data movement). Legal keystroke sequences in the Immediate area are:



Except for the GOTO command, each keystroke pair indicates a destination followed by a source for a data movement command. The first three sequences above are executed as soon as the last keystroke is entered and cause transition into a new mode of operation (Programming or Display). The rest are executed only when the cursor is moved onto the "EXECUTE" field that appears at the end of the command; until then, CURSOR and digit keys can be used to change parameters associated with the Data Movement and GOTO keystrokes. See Section 3 and the individual key discussions in this section.

PARAMETERS

None.

NOTES

The **DELETE** key works only on the current program in Programming mode. To correct an incorrect keystroke in an Immediate mode command, press the **IMMEDIATE** key; this deletes the entire command and prepares the 7D02 to accept a new one.

NOT

KEY GROUP

Miscellaneous.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The **NOT** key complements an event; i.e., it defines an event as a counter **not** having reached its terminating value or as a Word Recognizer **not** matching a word of data.

A **NOT** key can only appear within the brackets of a compound event, either immediately following the opening bracket, or after an **OR**. It must be immediately followed in the program by an event (**WD RECOGNIZER** or **COUNTER**). It affects only the single event immediately following it. It can never be used to negate a compound event as a whole.

PARAMETERS

None.

NOTES

If the **NOT** key is pressed after an "IF", "OR IF" or "STORE ON" prompt, an opening bracket is automatically supplied before the NOT. The user must supply the closing bracket at the end of the compound event by pressing the **NOT** key.

OR




KEY GROUP

Miscellaneous.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The  key is used to form compound events within a 7D02 program. It must be used within brackets. Each event within a compound event must be followed by either an  and another event, or by a  keystroke to terminate the compound event.

The implication of the OR is that the compound event will be TRUE when one or more of the component events is TRUE. The component events within a compound event are evaluated simultaneously.

PARAMETERS

None.

NOTES

None.

PROGRAM

KEY GROUP

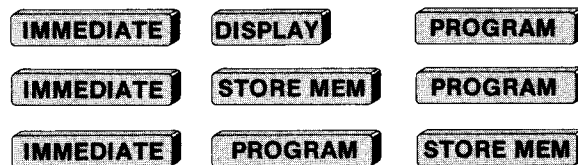
Data Movement.

MODE OF OPERATION

Valid in Immediate mode only.

GENERAL DESCRIPTION

The **PROGRAM** key is used to specify Program Memory as either the source or destination of a data movement command in Immediate mode. It is valid only in the following keystroke sequence.



The **IMMEDIATE** key puts the 7D02 into Immediate mode. The second key in each sequence is the destination of the data movement. The third key in each sequence is the source.

The first sequence displays the program in Program Memory on the screen and places the 7D02 in Programming mode.

The second sequence stores the contents of Program Memory in Storage Memory. It is not actually executed until the blinking cursor is moved onto the "EXECUTE" field appearing in the Immediate area. If Storage Memory already contains a program, that program is swapped into Program Memory; otherwise, Program Memory is unchanged. The 7D02 returns to the operating mode it was in before the **IMMEDIATE** key was pressed.

The third sequence transfers the program in Storage Memory into Program Memory. It is not actually executed until the blinking cursor is moved onto the "EXECUTE" field in the Immediate area. If Storage Memory does not contain a program, an error message appears in the Status area at the top of the screen. Either way, the 7D02 returns to the operating mode it was in before the **IMMEDIATE** key was pressed.

See Section 3 and the entries in this section for the specific keys involved.

PARAMETERS

None.

NOTES

None.

QUALIFY

KEY GROUP

Structure or Command.

The **QUALIFY** key falls into two key groups (although only Structure is indicated on the keyboard). The context in which it appears determines its use.

1. It is a Structure key if it is entered into an empty program or immediately following an "END TEST" prompt.
2. It is a Command key if it is entered into a program immediately following a "THEN DO" or "ELSE DO" prompt, or as part of a compound command.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The 7D02 samples data from the system-under-test on every cycle (as determined by the State clock—see Section 2). The default operation is to store every word of sampled data into the Acquisition Memory. The **QUALIFY** key gives the user the ability to set up criteria that must be met for the sampled data to be stored into Acquisition Memory on any given cycle. Sampled data will not be stored on cycles on which the criteria are not met.

When used as a Structure key, **QUALIFY** creates a Qualify block. The Qualify block is always active when the program is running, regardless of which of the Tests in the program is currently active. The block is defined by pressing the **QUALIFY** key, followed by an event key (or compound event), followed by the END key. A compound event may include complemented events (i.e., the NOT key may be used). Data is stored into Acquisition Memory only on those cycles on which the event in the Qualify block is TRUE.

When used as a Command key, **QUALIFY** means "store". On every cycle on which the command is executed, the sample data will be stored into Acquisition Memory (regardless of the status of the event in the Qualify block, if any).

PARAMETERS

None.

NOTES

General.

1. If the **QUALIFY** key is not used at all in a program, the data sampled on every cycle will be stored into Acquisition Memory. If the **QUALIFY** key is used **anywhere** in the program, data will be stored **only** when the event in the Qualify block is TRUE (if there is a Qualify block) **or** if a Qualify command is executed **or** if both occur.
2. Data qualification is entirely different from clock qualification. Clock qualification determines **when** (relative to the system-under-test raw clock) data will be sampled. Data qualification sets criteria to determine whether or not sampled data is subsequently stored into Acquisition Memory. See Section 7 for details on clock qualification.

3. It is possible, because of data qualification, for the data sampled on the cycle on which the trigger occurs not to be stored. The trigger indicator on the resulting data display will be placed on the last word that was stored before the trigger occurred. If no data is stored before the trigger occurs, the indicator will appear on the first word in Acquisition Memory.

Qualify Block

1. Only one data qualification block may be defined in a program.
2. The qualification block may appear before or after any Test in the program.

Qualify Command

1. Any number of **QUALIFY** commands may appear in a program.

EXAMPLE

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=3021
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 GOTO 2
END TEST 1
TEST 2
2IF
2 WORD RECOGNIZER # 2
2 DATA=XX
2 ADDRESS=8022
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2THEN DO
2
2 QUALIFY
2 GOTO 1
2
2ELSE DO
2 QUALIFY
END TEST 2

```

The program above will store data only when Test 2 is active; i.e., after seeing address 3021 up to and including the next occurrence of address 8022. Because the QUALIFY command is used in Test 2, data will never be stored unless a QUALIFY command is executed, but there is no QUALIFY command in Test 1; therefore, data will never be stored when Test 1 is active. Note that two QUALIFY commands are needed in Test 2: one to store when the Word Recognizer is TRUE and one to store on the other cycles. Also note that this program has no TRIGGER command and can only be stopped by pressing the **START/STOP** key.

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=1 R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1 OR IF
1 WORD RECOGNIZER # 2
1 DATA=22
1 ADDRESS=4410
1 /NMI=X /IRQ=X FETCH=X R/W=0
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
1 QUALIFY
1
END TEST 1
QUALIFY
QSTORE ON
Q COUNTER # 1 =00010 0-EVENTS
END QUALIFY

```

This program will only store data after 10 fetches have occurred. A QUALIFY command has been compounded with the TRIGGER command to make sure the trigger word is stored even if 10 fetches have not occurred before the trigger occurs.

START/STOP

KEY GROUP

Miscellaneous.

MODE OF OPERATION

Valid in Programming, Run, Display, and Immediate modes of operation.

GENERAL DESCRIPTION

Pressing the **START/STOP** key in Programming, Display or Immediate mode causes the 7D02 to attempt to execute the program currently in Program Memory. If the program contains no errors, the 7D02 will enter Run mode and begin Test 1 of the program.

Pressing the **START/STOP** key while the 7D02 is in Run mode causes it to stop acquiring data and terminate execution of the program. The 7D02 will enter Display mode.

PARAMETERS

None.

NOTES

1. If the final "END" is missing from the program, it will be supplied automatically when the **START/STOP** key is pressed (if there is nothing else wrong with the program).
2. If the current program contains contradictory commands (conflicting commands within a single compound command) or an incomplete program (such as a "THEN DO" prompt at the end of the program), then an error message will be displayed in the Status area, and the 7D02 will not enter Run mode.
3. A program can never be run out of Storage Memory. In particular, merely **displaying** a program from Storage Memory does not make it executable; it must first be transferred to Program Memory.

STORE MEM

KEY GROUP

Data Movement.

MODE OF OPERATION

Valid in Immediate mode of operation only.

GENERAL DESCRIPTION

The **STORE MEM** key is used to specify the Storage Memory as the source or destination of a data movement command in Immediate mode. The **STORE MEM** key is only valid in the following keystroke sequences:

IMMEDIATE	DISPLAY	STORE MEM
IMMEDIATE	ACQ MEM	ACQ MEM
IMMEDIATE	PROGRAM	PROGRAM
IMMEDIATE	STORE MEM	STORE MEM

The **IMMEDIATE** key puts the 7D02 into Immediate mode. The second keystroke in each sequence indicates the destination of the data movement. The third keystroke indicates the source.

The first sequence is executed as soon as the **STORE MEM** key is pressed. It causes the 7D02 to enter Display mode, but leaves the "DISPLAY STOREMEM" command and the blinking cursor in the Immediate area where the parameters of the display can be manipulated with the **CURSOR** and Digit keys.

The other sequences cause an "EXECUTE" field to appear in the Immediate area. The commands are not actually executed until the blinking cursor is moved onto the "EXECUTE" field. Until then, the parameters of the command (if any) can be manipulated with the **CURSOR** and Digit keys. After execution, the 7D02 returns to the operating mode (Programming or Display) it was in before the **IMMEDIATE** key was pressed.

PARAMETERS

The **STORE MEM** key has no parameters unless it is used in the **DISPLAY** **STORE MEM** sequence, in which case the parameters depend on the type of information stored in the Storage Memory.

1. Storage Memory Contains Main Acquisition Data

```

DISPLAY← STOREMEM
0-ABSOLUTE
0 ABSOLUTE
1 MNEMONIC
  
```

0-ABSOLUTE
0 ABSOLUTE
1 MNEMONIC

If "0" is entered here, the Main Acquisition data is displayed in Absolute format. Data will generally appear as numeric fields whose radices are changeable in Format mode.

If "1" is entered here, the Main Acquisition data is displayed in Mnemonic format. The value on the s.u.t. data bus on Fetch cycles is generally displayed as the vendor op-code for the target processor. Other mnemonics are supplied where possible (such as some control line values); other fields are displayed in a radix compatible with the target processor.

The exact formats and attributes of these two displays are Personality Module-dependent. See the relevant Personality Module manual and the Display mode discussion in Section 3 for details.

2. Storage Memory Contains Timing Option Acquisition Data

See Section 6 and the Display mode discussion in Section 3.

3. Storage Memory Contains Program

No parameters are displayed.

NOTES

None.

TRIGGER

KEY GROUP

Command.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The Trigger command allows the user to indicate the point of interest in the data being sampled by the 7D02. It defines the window within which data gathered from the system-under-test is to be stored in the Acquisition Memory.

By specifying the location of the trigger point within the acquired data, the user indicates a value for the delay counter associated with the Acquisition Memory. Execution of the Trigger command will activate the delay counter. The delay counter will then increment once with each word of data that is stored into the Acquisition Memory. Acquisition is considered to be complete when the counter reaches its programmed value. The program is stopped, and the 7D02 transitions into Display mode.

The trigger, in conjunction with the delay counter value, defines the data window. If the delay is D , the number of words of data acquired after the trigger is D , and the number of words acquired up to and including the trigger is $[256 - D]$ (or less, if fewer than $[256 - D]$ qualified storage cycles preceded the trigger).

Data acquisition and program execution stop when each memory (Main, Timing, or both) for which a Trigger command appears in the program: (1) has triggered, and (2) has had its delay counter reach its programmed value. (See "When is the 7D02 Running?" in Section 7 for details.) Therefore, triggering indirectly leads to the transition from Run mode to Display mode.

If the Timing Option is not installed, every execution of a Trigger command also causes the External Trigger Out line to go TRUE (i.e., HI). If the Timing Option is installed, raising this line is a little more complicated; for full details on programming the External Trigger Out, see Section 7.

For information about the Timing Option Trigger, see Section 6. The rest of this discussion describes the Main Acquisition Memory TRIGGER command.

PARAMETERS

```

) TRIGGER 0-MAIN
      0 MAIN
      1 TIMING
0-BEFORE DATA
  0 BEFORE DATA
  1 CENTERED
  2 AFTER DATA
  3 ZERO DELAY
0-SYSTEM UNDER TEST CONT.
  0 SYSTEM UNDER TEST CONT.
  1 SYSTEM UNDER TEST HALT
0-STANDARD CLOCK QUAL.
  0 STANDARD CLOCK QUAL.
  1 USER CLOCK QUAL.

```

```

0-MAIN
  0 MAIN
  1 TIMING

```

This menu will only appear if the Timing Option is installed! It indicates which memory (Main or Timing) is to be triggered with this command. If the Timing Option is **not** installed, or if a "0" is entered for this menu, the following menus (those for the Main Trigger) will appear. If the Timing Option **is** installed and a "1" is entered here, a different set of menus and numeric fields will follow—see Section 6 for a description of the parameters in the Timing Option Trigger.

```

0-BEFORE DATA
  0 BEFORE DATA
  1 CENTERED
  2 AFTER DATA
  3 ZERO DELAY

```

This menu is used to program the delay counter value by specifying the position of the trigger within the acquired data.

If "0" is entered, the trigger point will be BEFORE the bulk of the data—240 acquisitions will take place after the Trigger occurs.

If "1" is entered, the trigger point will be CENTERED in a full (256-word) Acquisition Memory—128 Acquisitions will take place after the Trigger occurs.

If "2" is entered, the trigger point will be AFTER the bulk of the data—only 16 Acquisitions will take place after the Trigger occurs.

If "3" is entered, the trigger point will appear at the end of the acquired data. No acquisition will take place after the cycle on which the trigger occurs.

0-SYSTEM UNDER TEST CONT.
0 SYSTEM UNDER TEST CONT.
1 SYSTEM UNDER TEST HALT.

This menu is used to specify the action to be taken after the Main Acquisition Memory has been triggered and acquisition is complete; i.e., the delay counter has reached its programmed value.

If "0" is entered, the system-under-test will be allowed to continue running.

If "1" is entered, the Halt line will be asserted. The Personality Module will determine the definition and effect of the Halt line, but in general it will halt the microprocessor in the system-under-test. See the relevant Personality Module manual for more information.

0-STANDARD CLOCK QUAL.
0 STANDARD CLOCK QUAL.
1 USER CLOCK QUAL.

Whether or not this menu appears is Personality Module-dependent. This menu enables the user to program the method with which the raw clock from the system-under-test is qualified to provide the State clock that defines the 7D02 cycle and determines when data is sampled. The degree to which this is permitted varies with the Personality Module.

Most users will want to leave this menu set to its default value of "0"; a set of values that is standard for the target microprocessor or system will then be used.

Entering a "1" causes a new set of menus and numeric fields to appear; they are described in "Clock Qualification" in Section 7.

NOTES

1. A Trigger command is reprogrammed **only** when the user enters new values into it from the numeric keypad. In particular, deleting all instances of a Trigger command from a program will mean the Trigger is not used in the program, but will **not reprogram the Trigger**. If the Trigger is subsequently entered into a 7D02 program, it will contain the same values it had before it was deleted.
2. It is especially important to note that clock qualification information entered into a Main Trigger command stays in effect until it is changed, even if the Trigger command is deleted from the program; i.e., not only does it reappear when the **TRIGGER** key is next pressed, but it is used to qualify the system-under-test clock even if the program contains no Main Trigger! The user **must** explicitly re-enter a "0" into the Clock Qualification menu to restore the default values.
3. If there is no Trigger in a program, the program will only stop running when the user presses the **START/STOP** key.
4. If there is no Main Trigger in a program, the system-under-test will **not** be halted when the 7D02 program stops execution. This is true even if the user had selected the "1-SYSTEM UNDER TEST HALT" menu entry and then deleted the Trigger Command.
5. Once a Trigger command has been executed, subsequent executions of the Trigger command (while the delay counter is incrementing) will not cause triggering to re-occur. They **will**, however, have their normal effect (if any) on the External Trigger Out line.

6. If there is no Trigger command in a program, only Main Memory will acquire data, even if the Timing Option is installed. If the program contains only a Timing Option Trigger, then only Timing Option data will be acquired. If a program contains both a Main and a Timing Trigger, both types of data will be acquired.

7. The number of words of data stored **before** a Trigger command is executed can affect the absolute placement of the Trigger in Acquisition Memory. For example, if "0-BEFORE DATA" is selected and at least 15 words are stored before the Trigger, the trigger point will be at location 15 in Acquisition Memory. However, if only 10 acquisitions occur before the Trigger, the trigger point will be at location 10 and acquisition will complete with only 251 (instead of 256) words of data: 10 prior to the trigger point, 1 at the trigger point, and 240 after the trigger point. In other words, the number of words stored after the Trigger is fixed; the number of words stored before it may vary.

To guarantee a full (256-word) Acquisition Memory, the user could include a test to count off a delay that will guarantee the appropriate number of acquisitions before the Trigger. (See the example below.)

```

TEST 1
1IF
1 COUNTER # 1 = 00015 0-EVENTS
1THEN DO
1 GOTO 2
1ELSE DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
END TEST 1
TEST 2
2IF
2 Trigger event
2THEN DO
2 TRIGGER 0-MAIN
2 0-BEFORE DATA
2 0-SYSTEM UNDER TEST CONT.
2 0-STANDARD CLOCK QUAL.
END TEST 2

```

Test 1 delays searching for the triggering event until 15 cycles have occurred after the program has been started. This assures at least 15 data acquisitions will occur before the Trigger. The selection of "0-BEFORE DATA" assures 240 acquisitions after the Trigger. The Main Acquisition Memory should be full when the 7D02 leaves Run mode.

WD RECOGNIZER

KEY GROUP

Event.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The **WD RECOGNIZER** key permits the user to define a word recognition event in a 7D02 program. The event is TRUE when the incoming data matches a pattern entered by the user.

The numeric keypad is used to enter the pattern into a multi-line "form" that appears on the screen when the **WD RECOGNIZER** key is pressed. Any Personality Module channel for which "X" ("DON'T CARE") is entered will be considered to match the incoming data on that channel, regardless of the actual value of the data. The event is TRUE when the incoming data on **all** the channels match the corresponding entries programmed into the Word Recognizer; i.e., the value of the Word Recognizer is the AND of the matches on the individual channels.

Each word normally consists of 30 bits, usually grouped as 8 data bus bits, 16 address bus bits, and 6 control bits. The Expansion Option increases Word Recognizer size to 46 bits (usually 16 data, 24 address and 6 control). Installation of the Timing Option adds a single Timing Option link bit to the 7D02 Word Recognizers. In all configurations, the Word Recognizer also includes a single bit for the External Trigger In line.

The 7D02 provides four distinct Word Recognizers. Each may be programmed with a different pattern. Each may appear any number of times in a test or program. Only the Word Recognizer(s) in the current Test and the Qualify block (if any) are evaluated when the program is running.

PARAMETERS

```
WORD RECOGNIZER # 1
DATA=XX
ADDRESS=XXXX
/NMI=X /IRQ=X FETCH=X R/W=X
BA=X INVAL OP=X EXT TRIG IN=X
TIMING WR=X
```

WORD RECOGNIZER #1

This numeric field is used to select one of the four Word Recognizers. The default value is the lowest numbered unused Word Recognizer. This value may be changed to a digit in the range 1—4.

DATA=XX
 ADDRESS=XXXX
 /NMI=X /IRQ=X FETCH=X R/W=X
 BA=X INVAL OP=X

The pattern to be matched on the system-under-test bus is entered here. The format is **Personality Module-dependent**. The format shown above is generated by the M6802 Personality Module (PM-103). Each Personality Module groups channels into "fields" in a manner consistent with the processor or system it replaces (e.g., data bus, address bus, control lines). The Personality Module also determines the labeling of fields and the initial radices in which they are displayed. See the manual for the Personality Module you are using.

EXT TRIG IN=X

A "0" here indicates the External Trigger In must be FALSE for a match to occur.

A "1" here indicates the External Trigger In must be TRUE for a match to occur.

TIMING WR=X

This parameter only appears if the Timing Option is installed. See Section 6 for its use.

NOTES

1. Via Format mode it is possible to change the radices in which some of the Word Recognizer fields are displayed. (This is Personality Module-dependent.)
2. The 7D02 provides four Word Recognizers. Each may be programmed with a different pattern. Each one may be used any number of times within a program, but each time it is used it will contain the same pattern. Changing the pattern in one instance of that Word Recognizer will automatically cause it to be changed in all instances.
3. A Word Recognizer is reprogrammed **only** when the user enters a new pattern into it with the numeric keypad. In particular, deleting all instances of a Word Recognizer from a program will mean the Word Recognizer is not used in the program, but will **not reprogram the Word Recognizer**. If that same Word Recognizer is subsequently entered into a 7D02 program, it will contain the same pattern it had before it was deleted.
4. The power-up default for all Word Recognizers is a pattern containing all "X"s; i.e., "DON'T CAREs". This pattern will match **any** incoming data; i.e., the Word Recognizer will always be TRUE.
5. The four Word Recognizers are level-sensitive. This means they are evaluated by comparing the programmed pattern to the actual values on the channels at the instant of data sampling.

EXAMPLE

```

TEST 3
3IF
3 WORD RECOGNIZER # 2
3 DATA=XX
3 ADDRESS=F836
3 /NMI=X /IRQ=X FETCH=1 R/W=X
3 BA=X INVAL OP=X EXT TRIG IN=X
3 TIMING WR=X
3THEN DO
3 COUNTER # 1 0-EVENTS
3 0-INCREMENT
END TEST 3

```

In the above program fragment, Word Recognizer #2 will come TRUE whenever an instruction is fetched from address F836. The Test will count the number of times such an instruction fetch takes place.

X

See DON'T CARE key description.

0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
A, B, C, D, E, F

KEY GROUP

Numeric Entry.

MODE OF OPERATION

Valid in Programming, Display, Immediate and Format modes.

GENERAL DESCRIPTION

The digit keys (**0** - **9**) and (**A** - **F**) provide a hexadecimal keypad that is used to make menu selections and enter numeric parameters.

Menu and numeric values may be changed by moving the blinking cursor to the menu or numeric field to be modified and pressing the appropriate digit key(s). As each digit is entered, it replaces the previous value, and (except on Format mode menus) the blinking cursor moves to the next digit position.

PARAMETERS

None.

NOTES

1. All digit keys repeat if they are held down.
2. The digit keys that are valid in a menu will depend on the number of choices in the menu.
3. The digit keys that are valid in a numeric field will depend on the current radix of the field. Some fields also impose maximum and/or minimum legal values.

[]

KEY GROUP

Miscellaneous.

MODE OF OPERATION

Valid in Programming mode only.

GENERAL DESCRIPTION

The **[]** key is used to combine events to form compound events or to combine commands to form compound commands.

When pressed after an "IF", "OR IF" or "STORE ON" prompt, the **[]** key generates an "open bracket" indicating the start of a compound event. The next use of the key will close the bracket, terminating the compound event. Between the brackets, event keys and the **[OR]** key can be pressed in an "event **[OR]** event **[OR]** . . . **[OR]** event" sequence. The event keys can also be preceded by the **[NOT]** key to negate component events. The compound event is TRUE when any one (or more) of the component events comes TRUE.

When pressed after a "THEN DO" or "ELSE DO" prompt, the **[NOT]** key generates an "open bracket" on the screen, indicating the start of a compound command. The next use of the key closes the bracket, terminating the compound command. Any number of command keys may be entered between the brackets. The commands are connected with an implied logical AND operator; i.e., execution of the compound command during normal program flow implies simultaneous execution of **all** the component commands within the brackets.

PARAMETERS

None.

NOTES**Compound events**

A compound event cannot be negated with the **[NOT]** key. Individual component events within the compound event **can** be negated.

Compound commands

If contradictory commands are placed within a pair of brackets, an error message will be displayed in the Status area of the screen, and a blinking "E" will appear in column 1 of the program line containing the first contradiction. Contradictory commands are (1) GOTOs to two different tests, or (2) two different commands for the same Counter.

Section 6

TIMING OPTION

TABLE OF CONTENTS

INTRODUCTION

- Acquisition
- Event Recognition
- Triggering

DEFINITIONS (In The Context Of The Timing Option)

SYNCHRONOUS OPERATION

SYNCHRONOUS OPERATION EXAMPLES

- Trigger Timing From Main
- Trigger Main From Timing
- Trigger Main And Timing From Timing

SUMMARY OF SYNCHRONOUS MODE

ASYNCHRONOUS OPERATION

- Event Recognition
- Word Recognizer
- Glitch Recognizer
- Triggering On A Glitch
- Timebase Programming
- Timing Option Asynchronous Triggering
- Main Acquisition And Asynchronous Timing Option Relationships

ASYNCHRONOUS OPERATION EXAMPLES

- Arm Timing Option From Main, Then Trigger Timing
- Trigger Main From Timing Option Event Recognizer
- Arm Timing Option Immediately, Trigger The Timing Option From Its Own Word Recognizer
- Triggering On A Glitch
- How The Filter Delays The Timing Option Trigger
- Arm Timing Option From Timing Option Event Recognizer, Then Trigger Timing Option
- Trigger Main And Timing Option From Timing Option Asynchronous Word Recognizer

SUMMARY OF ASYNCHRONOUS MODE

TIMING OPTION CLOCKING MODE SELECTION

TIMING OPTION DISPLAYS

- Timing Diagram
 - Data and Glitches
 - Channel Ordering
 - Tick Marks
 - Cursor Information
 - Magnification
- State Table Display
 - Channel Ordering
 - Highlighting Storage Memory Differences
- Scrolling The Timing Option Displays

TABLE OF CONTENTS (cont)

TIMING OPTION PARAMETERS

Parameter Definitions

HARDWARE FUNCTIONAL DIAGRAMS

Synchronous Operation

Asynchronous Operation

TIMING OPTION

INTRODUCTION

The Timing Option is a purchasable hardware option for the 7D02 whose primary purpose is to acquire asynchronous information. However, it can also be used to acquire synchronous information. It provides the means for monitoring an additional 8 lines on the system-under-test.

The 7D02's Timing Option operates in two modes—synchronous or asynchronous. Three distinct functions can be performed in either mode—acquisition, recognition, and triggering.

ACQUISITION

In both synchronous and asynchronous operation, the data window is positioned about the trigger point with a programmable delay counter. The four choices for the trigger location, and their effect are:

- 0-BEFORE DATA — 240 memory locations will be stored after the trigger
- 1-CENTERED — 128 memory locations will be stored after the trigger
- 2-AFTER DATA — 16 memory locations will be stored after the trigger
- 3-DELAYED — Gives the ability to "window" data far downstream from the Trigger event.
The range is 00000 to 65534. If the delay is 00000, only the data prior to the trigger and the trigger are stored. If the delay is greater than 255, only the last 255 qualified data acquisitions are stored, and not the trigger location.

In synchronous operation:

1. The Acquisition Memory is 255 locations x 8 channels of acquisition data.
2. The State clock derived from the s.u.t. clock is used as the storage clock. This clock is identical to the Main Storage clock and its rising or falling edge and qualification aspects are programmed under the TRIGGER 1-MAIN command.
3. The block QUALIFY and the QUALIFY command, if included in the program, operate in the same way in storing Timing Option data as they do in storing Main section data.

In asynchronous operation:

1. The Acquisition Memory consists of 255 locations x 8 channels of data.
2. The internal timebase gives a sample period selection from 20 NS to 5 MS in a 1-2-5 sequence.
3. There are 255 locations x 8 channels of glitch memory. (A glitch on a channel is defined as TRUE for a channel when more than one transition occurs on that channel between two consecutive clock samples.)
4. No data qualification takes place (all samples are stored).

EVENT RECOGNITION

In both synchronous and asynchronous modes, the Timing Option Event recognizer signal is available to the Main word recognizers in the form of the programmable TIMING WR= bit (also referred to as the "link" bit). This bit reduces the Timing Option event recognizer to a 1-bit TRUE/FALSE/DON'T CARE choice. A 1 means recognize the TRUE condition of the Timing Option event recognizer; a 0 means recognize the FALSE condition; an X means that the event recognizer is not considered in determining the TRUE/FALSE condition of the Main word recognizer.

In synchronous operation:

1. The Timing Option event recognizer is TRUE whenever the P6451 channel input values match the pattern specified in the WORD RECOGNIZER= field and the signal meets the set-up and hold specifications for the State clock.
2. Selection of the Timing Option synchronous Event recognizer is only available under the TIMING WR= (0 or 1) of the Main word recognizer.

In asynchronous operation:

1. The Timing Option Event recognizer is the ANDed function of the Timing Option word recognizer, a glitch word recognizer, and the External Trigger In signal from a front panel BNC.
2. Timing Option word recognizer (WORD RECOGNIZER=) is 8 channels wide. It is TRUE when the data on the P6451 matches the user-specified pattern. Its output is not connected to the internal sample interval. Each channel can be selected to look for a 0, a 1, or to "not look"; i.e., the X = DON'T CARE.
3. The External Trigger In can be considered as a ninth word recognizer channel that comes from the front panel BNC.
4. The programmable filter, in series with the Timing Option word recognizer and External Trigger In, causes the output from these nine channels to match their programmed levels continuously for the amount of time specified by the filter value in order to generate a TRUE output from the filter. The filter values range from 000 to 300 ns in 20 ns increments.
5. The glitch word recognizer is the ORed function of the occurrence of a glitch on any of the 8 channels with a 1 specified in that channel's position in the GLITCH RECOGNIZER= field.
6. The TIMING WR= signal to the Main word recognizer is the Timing Option Event recognizer output. If the event recognizer is TRUE or goes TRUE anytime during the State clock period, it remains TRUE for that period. (It is pulse-stretched TRUE.) The converse is not true. The Event recognizer must stay FALSE the entire s.u.t. clock period to be recognized FALSE.

TRIGGERING

Triggering means to start the Timing Option delay counter, then stop data acquisition when the delay counter reaches its programmed value.

In synchronous operation:

1. The Timing Option is triggered when the TRIGGER 1-TIMING command is executed in the program.

In asynchronous operation:

1. The Timing Option is ARMED (not triggered) to look for its own event (from its event recognizer) when TRIGGER 1-TIMING is executed in the program.
2. The Timing Option actually triggers when it has been armed (item 1) and its event recognizer subsequently goes from FALSE to TRUE when 1-ARM ASYNC, TRIG ON WR $\overline{\text{L}}$ is selected, or its event recognizer subsequently goes from TRUE to FALSE when 2-ARM ASYNC, TRIG ON WR $\overline{\text{L}}$ is selected.

DEFINITIONS (In The Context Of The Timing Option)

Synchronous Mode—	A mode of operation in which the Timing Option is used as an 8-bit extension to the main data acquisition and word recognizers and uses the State clock as the sampling clock. Synchronous word recognition is "level sensitive". (Level sensitive is defined below.)
Asynchronous Mode—	A mode of operation in which the Timing Option is used as a separate 8-bit logic analyzer and uses the internal programmable timebase as the sample clock. Asynchronous word recognition is "edge sensitive". (Edge sensitive is defined below.)
Acquisition—	The data sampling process whose frequency is controlled by the user-specified internal timebase (asynchronous) or the 7D02 State clock (synchronous).
Word Recognition—	The detection of a user-specified bit pattern on the 8 input channels of the P6451 Timing Option Probe.
Timing Option Event Recognition—	The detection of a user-specified bit pattern on the 8 input channels of the Timing Option probe. In asynchronous mode, it is ANDed with the External Trigger In signal and the glitch recognizer.
Triggering—	Causes the 7D02 to start its delay counter and complete storage of data in its Acquisition Memory when the delay counter reaches its programmed value.
Level sensitive—	A word recognizer mode of operation in which a TRUE condition exists when the input signal levels match the programmed values at the State clock edge. The Timing Option word recognizer is level sensitive in synchronous mode.
Edge sensitive—	A word recognizer mode of operation in which a TRUE condition is recognized when the input signals change from not matching to matching the user-specified value (FALSE to TRUE). A FALSE condition is recognized when the input signals change from matching to not matching the specified values. The Timing Option word recognizer is edge sensitive in asynchronous mode.
Main (section)—	Those parts of the program data and displays that relate to synchronous acquisition through the PM-100 Series Personality Modules.
Timing Option—	The hardware installed in the 7D02 which provides for an additional 8 channels of data acquisition and 8 channels of glitch information. Additional menus and data displays appear on the 7D02's screen when the Timing Option is installed.
Arm—	Arm means to set up or prepare to accept a trigger. Arm applies to the Timing Option in asynchronous mode. If the Timing Option is not armed first, it cannot be triggered. It is armed when the TRIGGER 1-TIMING command is executed.
Glitch—	A glitch on a channel is defined as more than one transition occurring between successive samples on the channel.
State Clock—	The clock derived from the s.u.t. clock. This clock is conditioned by the 7D02 front-end circuitry as specified by the user clock processing menus under the Main trigger.

SYNCHRONOUS OPERATION

Because the Timing Option's synchronous operation can be used as an extension of the Main data acquisition, it is discussed first. Asynchronous operation is discussed later in this section.

The Timing Option synchronous Event recognizer is simply an 8-bit extension of the Main Word Recognizers and is linked or unlinked by a single bit, TIMING WR=. It is not defined any place else. Note that there are four Main word recognizers, but only one Timing Option word recognizer. That means that the same Timing Option word recognizer appears in each and every one of the Main word recognizers if it is used. If you change a Timing Option word recognizer value in one Main word recognizer, it is automatically changed in all of the other occurrences in your program.

The Timing Option trigger signal is separated from (i.e., is independent of) the Timing Option Word Recognizer. Therefore, the Timing Option is only triggered by a TRIGGER 1-TIMING command in the program.

When data qualification is used in the program, either as a QUALIFY command or as a Qualify block, the number of data words stored in memory may be less than the total number of clock cycles which have occurred. (Asynchronous data acquisition is NOT subject to qualification.)

The following program examples illustrate some of the triggering combinations of the Main and Timing Option sections with the Timing Option in synchronous mode.

SYNCHRONOUS OPERATION EXAMPLES

TRIGGER TIMING FROM MAIN

"Trigger Timing From Main" means that when the trigger conditions specified in the Main section are met (i.e., the event occurs), start the Timing Option delay counter (i.e., execute the trigger command) and complete the storage of synchronous, qualified data into the Timing Option Acquisition Memory.

One program that will do this is:

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=F830
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 TRIGGER 1-TIMING
1 0-BEFORE DATA
1 0-THRESHOLD V. = 0-PLUS 1 40
1 0-SYNC, TRIGGER IMMEDIATE
END TEST 1

```

When the system-under-test bus values match those specified in the Word Recognizer (ADDRESS=F830), then trigger the Timing Option and finish acquiring data on the Timing Option probe (P6451).

Synchronous mode is entered by selecting the 0-SYNC, TRIGGER IMMEDIATE menu selection. This is the only place in this program where you are able to select the Timing Option acquisition and triggering mode.

Timing Option—7D02 Operators

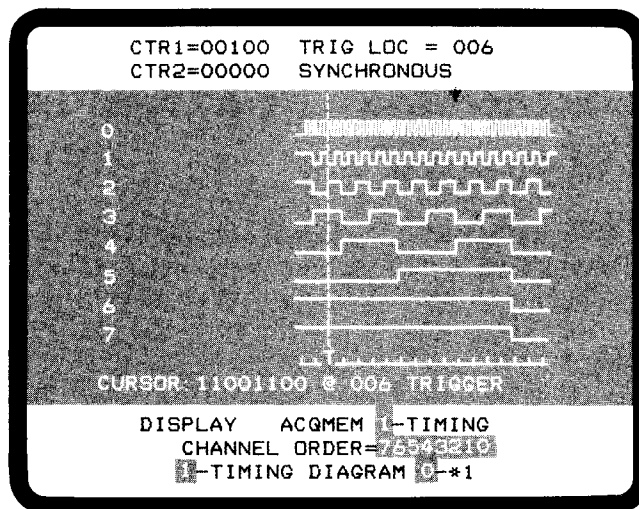
Note that:

The TRIGGER 1-TIMING command was executed when the Main event occurred (Word Recognizer #1 became TRUE).

The acquired timing data can be displayed as:

1. A timing diagram (Figure 6-1), or
2. A timing state table (Figure 6-2).

Note that no Main acquisition data was acquired because there was no TRIGGER 0-MAIN command in the program (Figure 6-3).



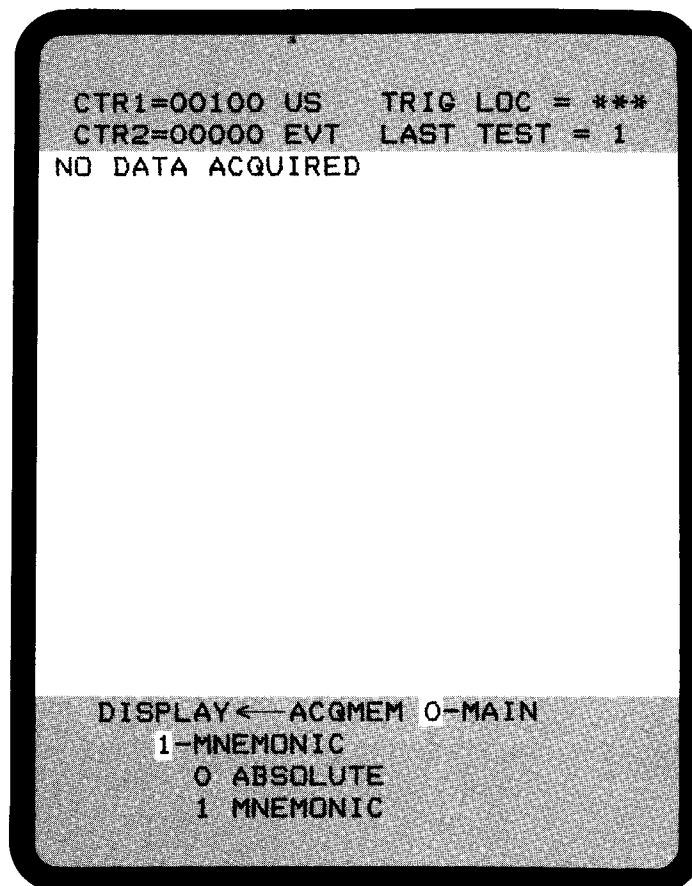
2918-60

Figure 6-1. 7D02 timing diagram.

CTR1=00100 US		TRIG LOC = 006	
CTR2=00000 EVT		SYNCHRONOUS	
LOC DATA			
000	11000110		
001	11000111		
002	11001000		
003	11001001		
004	11001010		
005	11001011		
006	11001100	—————	TRIGGER—————
007	11001101		
008	11001110		
009	11001111		
010	11010000		
011	11010001		
012	11010010		
013	11010011		
014	11010100		
015	11010101		
016	11010110		
017	11010111		
DISPLAY←ACGMEM 1-TIMING			
CHANNEL ORDER=76543210			
0-STATE DISPLAY			

2918-61

Figure 6-2. 7D02 timing state table.



2918-62

Figure 6-3. 7D02 Main acquisition memory display.

TRIGGER MAIN FROM TIMING

The Timing Option synchronous word recognizer is connected to the Main word recognizer through the TIMING WR= bit. "Trigger Main From Timing" means that when the Timing Option word recognizer goes TRUE, trigger the Main acquisition process. That means start the delay counter and complete storage of qualified data.

One program that will do this is:

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=1
1 THRESHOLD V. = 0-PLUS 1.40
1 0-SYNC
1 WORD RECOGNIZER=00XX00XX
1 THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

When the Timing Option recognizes a 00XX00XX bit pattern from the P6451 probe, trigger the Main data acquisition process.

Note that the selection of 0-SYNC in the Timing Option Event Recognizer specification is the only place in this program in which you can choose between synchronous and asynchronous operation of the Timing Option Event Recognizer.

The acquired data can be displayed as:

1. A mnemonic display (Figure 6-4), or
2. An absolute display (Figure 6-5).

There is no Timing Option acquisition data to be displayed because there is no TRIGGER 1-TIMING command in the program.

Note that:

1. The Trigger Main command was executed when the Timing Option event (WR=00XX00XX) occurred. This trigger word is not observable because no Timing Option data is stored by this program.
2. A Main word recognizer was used, but only to link in, i.e., provide access to, the Timing Option word recognizer.
3. The Timing Option word recognizer was selected by entering a 1 in the TIMING WR= field. This says that the output of the Timing Option word recognizer must be TRUE for the event to be TRUE. (A 0 says that the output must be FALSE.)
4. The Timing Option word recognizer is the only "event" in the entire program.

CTR1=00000 US TRIG LOC = 015				
CTR2=00000 EVT TRIG IN TEST 1				
LOC	ADDRESS	OPERATION	/IRQ/NMI	
015T	0006	OD WRITE	11	
016	F82F	BEQ	\$34	11
017	F830	34 READ	11	
018	F831	TST	\$0004	11
019	F832	00 READ	11	
020	F833	04 READ	11	
021	0004	75 READ	11	
022	F834	BNE	\$F6	11
023	F835	F6 READ	11	
024	F82C	INC	\$0006	11
025	F82D	00 READ	11	
026	F82E	06 READ	11	
027	0006	OD READ	11	
028	0006	0E WRITE	11	
029	F82F	BEQ	\$34	11
030	F830	34 READ	11	
031	F831	TST	\$0004	11
DISPLAY←ACGMEM 0-MAIN				
1-MNEMONIC				
0 ABSOLUTE				
1 MNEMONIC				

2918-63

Figure 6-4. Main section mnemonic display.

CTR1=00000 US TRIG LOC = 015			
CTR2=00000 EVT TRIG IN TEST 1			
LOC	ADDRESS	DATA	CNTL
015T	0006	0D	W11
016	F82F	27	F11
017	F830	34	R11
018	F831	7D	F11
019	F832	00	R11
020	F833	04	R11
021	0004	75	R11
022	F834	26	F11
023	F835	F6	R11
024	F82C	7C	F11
025	F82D	00	R11
026	F82E	06	R11
027	0006	0D	R11
028	0006	0E	W11
029	F82F	27	F11
030	F830	34	R11
031	F831	7D	F11
032	F832	00	R11
033	F833	04	R11
DISPLAY← ACQMEN 0-MAIN			
0-ABSOLUTE			

2918-64

Figure 6-5. Main section absolute display.

TRIGGER MAIN AND TIMING FROM TIMING

The following program is an example of acquiring synchronous data with both the Timing Option and the Main section being triggered from the Timing Option P6451 probe data.

```


TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=1
1 THRESHOLD V. = 0-PLUS 1.40
1 0-SYNC
1 WORD RECOGNIZER=00XX00XX
1 THEN DO
1
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
1 TRIGGER 1-TIMING
1 0-BEFORE DATA
1 THRESHOLD V. = 0-PLUS 1.40
1 0-SYNC, TRIGGER IMMEDIATE
1
END TEST 1

```

When the Timing Option recognizes the first 00XX-00XX bit pattern, trigger both the Main and Timing Option data acquisition processes.

Note that SYNC specification occurs in two places in this program (once for recognition mode, and once for triggering/acquisition mode). If one is changed to ASYNC, the other is automatically changed to ASYNC by the 7D02.

Note that:

- Both trigger commands are executed when the Main word recognizer #1 goes TRUE; i.e., when The Timing Option word recognizer detects the 00XX00XX bit pattern (in this case it was a 00000001).
- The  key was used to cause both triggers to be executed simultaneously; i.e., square brackets are used to form compound commands.
- Both trigger commands were included within the square brackets.
- Other than the changes needed to cause both triggers to be executed simultaneously, this is the same program as was used in the previous example.

The Timing Option synchronously acquired data can be displayed as:

- A timing diagram, or
- A timing state table.

The Main acquisition data can be displayed as:

- A Main mnemonic display, or
- A Main absolute display.

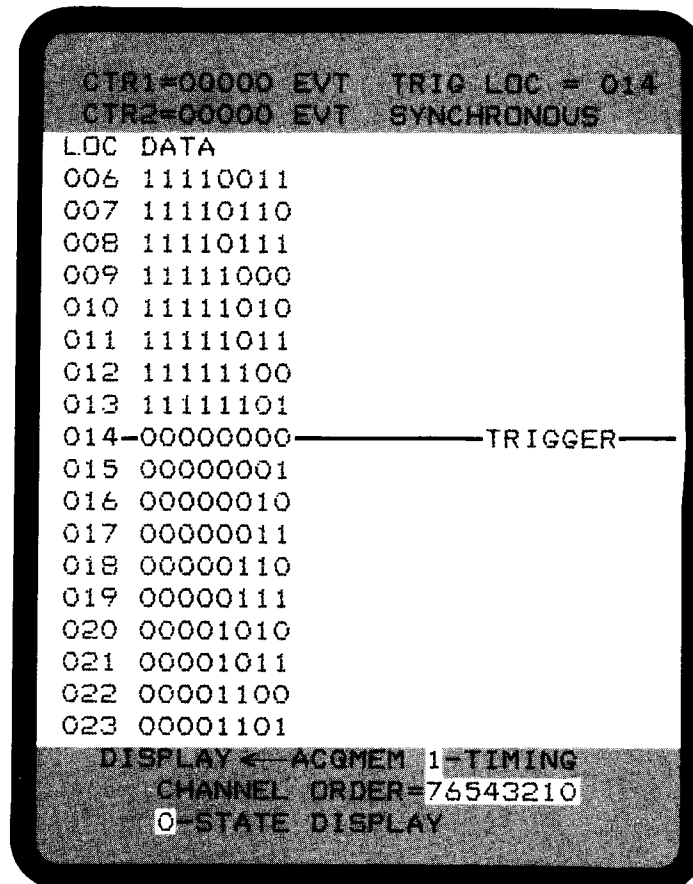
Figures 6-6 and 6-7 are the Main mnemonic and timing state table displays from the program. Notice that the Timing Option did, indeed, trigger on a 00XX00XX bit pattern.

CTR1=00000 US					TRIG LOC = 015				
CTR2=00000 EVT					TRIG IN TEST 1				
LOC	ADDRESS		OPERATION		/IRQ/NMI				
015T	0006		0D WRITE		11				
016	F82F		BEQ		\$34		11		
017	F830		34 READ		11				
018	F831		TST		\$0004		11		
019	F832		00 READ		11				
020	F833		04 READ		11				
021	0004		75 READ		11				
022	F834		BNE		\$F6		11		
023	F835		F6 READ		11				
024	F82C		INC		\$0006		11		
025	F82D		00 READ		11				
026	F82E		06 READ		11				
027	0006		0D READ		11				
028	0006		0E WRITE		11				
029	F82F		BEQ		\$34		11		
030	F830		34 READ		11				
031	F831		TST		\$0004		11		
032	F832		00 READ		11				
033	F833		04 READ		11				

DISPLAY←ACQM 0-MAIN	
1-MNEMONIC	

2918-65

Figure 6-6. Main section mnemonic display.



2918-66

Figure 6-7. Timing state table display.

The same techniques used in the last example can be used with the first program presented in this section to cause both Trigger commands to be executed from the Main section.

SUMMARY OF SYNCHRONOUS MODE

Table 6-1 is a summary of the key points to remember about the use of the Timing Option in synchronous mode.

Table 6-1
SUMMARY OF SYNCHRONOUS MODE

Synchronous Mode	
Acquisition	<ol style="list-style-type: none"> 1. Can be treated as an extension of Main memory, but displayed separately. 2. Uses the State clock. 3. Data qualification, if specified in the program, is identical to that for the Main section.
Recognition	<ol style="list-style-type: none"> 1. Can be used as an 8-bit extension of the Main word recognizers. 2. Level sensitive. Can specify TRUE or FALSE via TIMING WR= field.
Triggering	<ol style="list-style-type: none"> 1. Functionally identical to the Main trigger. 2. Uses the TRIGGER 1-TIMING command. 3. Has a delay (after triggering) counter separate from the Main trigger delay counter.

ASYNCHRONOUS OPERATION

In asynchronous operation, the Timing Option behaves as a separate logic analyzer with its own internal timebase. Asynchronous acquisition is entirely separate from the Main data acquisition. It uses its own internal sample clock, not the State clock. The Timing Option is armed by the TRIGGER 1-TIMING command and triggers on its own Event Recognizer.

The Timing Option has the ability to recognize glitches and to compare the duration of a TRUE Word Recognizer output signal to a user-specified filter value to determine if the signal is to be recognized as being TRUE.

EVENT RECOGNITION

The Timing Option Event Recognizer consists of the following:

1. An 8-bit asynchronous Word Recognizer ANDed with
2. An External Trigger In signal from the front panel BNC.
3. The above two items are in series with a programmable filter (0—300 ns in 20 ns increments) which requires the word to be TRUE for the prescribed number of nanoseconds before its output goes TRUE.
4. 8 channels of glitch recognition ORed together.
5. Items 3 and 4 are ANDed together to generate the Event recognizer output which goes to the TIMING WR= bit in the Main word recognizer and to a selectable edge detector to trigger the Timing Option.

See Figure 6-8 for an illustration of these relationships.

This asynchronous Event recognizer specification can either appear in conjunction with the Main word recognizers (under the TIMING WR=) or with the Timing Option trigger specification. Regardless of where it is defined, the Timing Option word recognizer value will be identical in all locations. (The 7D02 automatically enters the same values in all appropriate locations.)

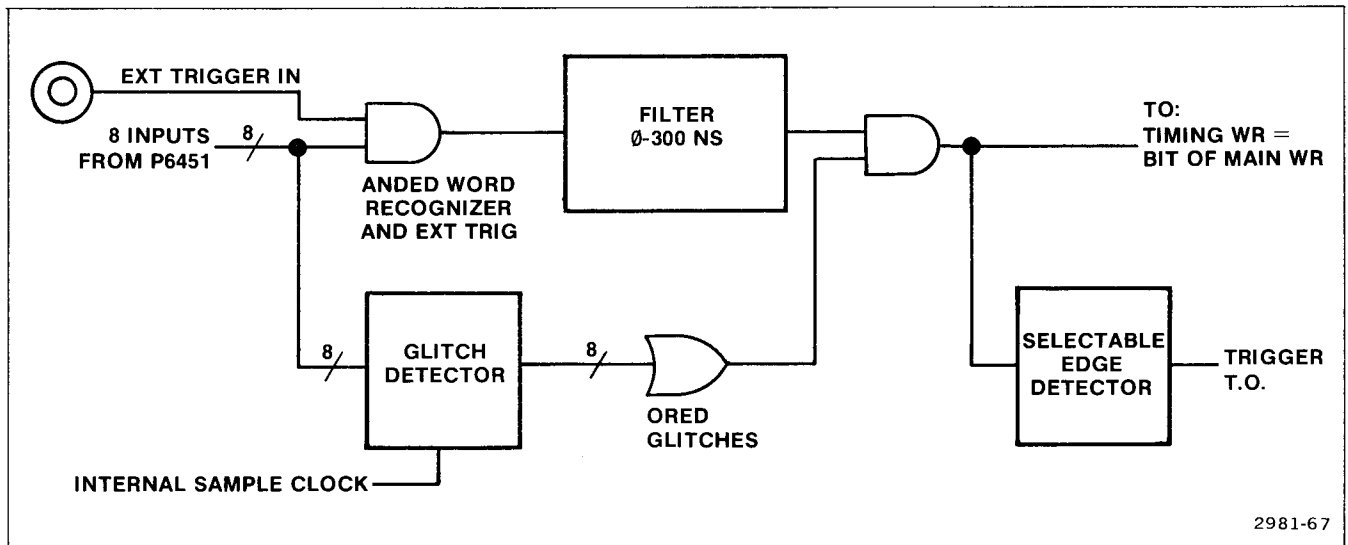


Figure 6-8. Asynchronous Timing Option event recognizer relationships.

WORD RECOGNIZER

The asynchronous Timing Option word recognizer and the External Trigger In detect and trigger on momentary states on the order of 10 ns (or the current filter value, if it is non-zero), regardless of the sampling rate (which is in the range 20 ns to 5 ms). As a result, the asynchronous Timing Option can trigger on a value that is not actually stored, but this momentary state would show up in the display at the trigger point as a glitch.

GLITCH RECOGNIZER

A glitch is defined as more than one transition between successive sample periods. The glitch recognizer is the ORing of all 8 channels of glitch detection.

The only valid values that can be entered in the glitch recognizer are 1 and X, because the glitch recognizer field is always displayed in binary, and the glitch recognizer does not recognize the absence of a glitch on a channel (only the presence of a glitch). Glitch polarity is not selectable. The 1 simply means that a glitch is being looked for on that channel.

The glitch recognizer is considered to be edge sensitive because the Timing Option Event recognizer is edge sensitive. If the Event recognizer is set to trigger on a glitch on channel 0, the data on channel 0 has to change from no glitch generated to a condition of a glitch in order to trigger the Timing Option.

A large number of glitches indicates that the sampling rate selected is too slow to see all of the transitions in the data.

TRIGGERING ON A GLITCH

In asynchronous mode, when triggering on a glitch, the Timing Option generally displays the trigger location as one location after the glitch on the screen. For example, in state table mode:

DATA	GLITCH
00110011	-----*-----
00110101	----- TRIGGER

The reason is inherent in the glitch detection process. Triggers which occur before a sample clock mark the data which is recorded at the sample clock as the trigger point. Glitches which occur before a sample clock are also recorded at the sample clock. However, the glitch information is not available to the glitch recognizer until after the sample clock. Therefore the glitch trigger is one clock late.

Figure 6-9 illustrates this. The glitch trigger output is TRUE between sample clocks 2 and 3 for the glitch that occurred between sample clocks 1 and 2. If the Timing Option glitch recognizer was set to trigger on a glitch on that channel, the trigger (T) would show up on sample 3 because the glitch trigger output is only valid after sample 2 and so, is picked up on sample 3.

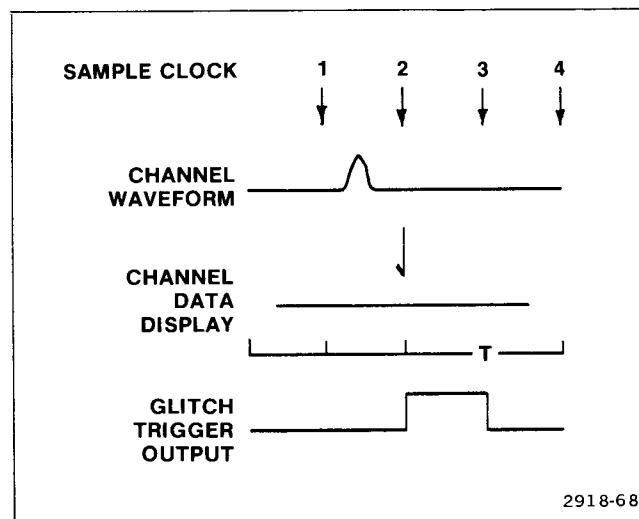


Figure 6-9. Glitch triggering.

It is remotely possible for a glitch to be detected the instant the Timing Option is started. The state machine can recognize this, but the Timing Option does not record data the first clock after it starts. Hence, you might trigger on, but not record, the first glitch with the following program:

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=1
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ASYNC
1 SAMPLE PERIOD 1 * 1-100NS
1 WORD RECOGNIZER=XXXXXXXX
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXX1XXXX
1 FILTER=000 NS
1 THEN DO
1 TRIGGER 1-TIMING
1 0-BEFORE DATA
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ARM ASYNC TRIG ON WR
1 SAMPLE PERIOD 1 * 1-100 NS
1 WORD RECOGNIZER=XXXXXXXX
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXX1XXXX
1 FILTER=000 NS
END TEST 1

```

2918-69

The first glitch satisfies the Timing Option Event word recognizer, which satisfies the Main word recognizer 1, which arms the Timing Option. The Timing Option will then trigger on the second occurrence of the glitch. Normally both glitches will be recorded, but it is possible the first one will not be stored as a result of the system architecture design. See Section 7 for more details.

TIMEBASE PROGRAMMING

The Timing Option asynchronous timebase is defined by entering values in two fields: a multiplier and an exponent. The multiplier can only have a value of 1, 2, or 5. The clock multiplier and exponent fields cannot both have their minimum values (1 and 0, respectively) simultaneously. This corresponds to a clock of 1×10 ns, and the shortest valid clock period is 20 ns. Any other combination of selections is valid. If 1 and 10 ns selections are attempted, 2 and 10 ns will be inserted and an error message presented at the top of the display.



TIMING OPTION ASYNCHRONOUS TRIGGERING

Asynchronous triggering of the Timing Option is more complex than its synchronous counterpart. This is because its Event recognizer is edge sensitive, and triggering is a two-part sequence: ARM the Timing Option, then TRIGGER the Timing Option.


To arm the Timing Option means to set it up to accept a trigger. If the Timing Option is not armed first, it cannot be triggered. The Timing Option is armed when the TRIGGER 1-TIMING command is executed in the program.

The Timing Option is then triggered when its Event recognizer generates the user-specified edge; i.e., the Event recognizer output goes from FALSE to TRUE or from TRUE to FALSE.

The trigger location displayed on the screen may be plus or minus one memory location from the actual trigger word. If a filter value has been specified, the displacement from the trigger word is dependent on the sample speed selected, and can be up to the value of the filter in sample time plus or minus one location.

Assuming the Timing Option is already armed, the Timing Option actually triggers only on a transition edge; i.e., when the Timing Option word recognizer goes from NOT TRUE to TRUE when 1-ARM ASYNC, TRIG ON WR , or from TRUE to NOT TRUE when 2-ARM ASYNC, TRIG ON WR . If there is no activity on the Timing Option probe, the Timing Option will never trigger itself. The only exception to this is a trigger command containing an Event Recognizer that is all DON'T CAREs (including the 8-channel Word Recognizer, External Trigger In, and the 8-channel glitch recognizer). Such a command will trigger immediately. Note that recognition of transitions from NOT TRUE to TRUE are delayed by the amount specified in the asynchronous filter menu item. But the reverse is not true.

If the program takes the following form:


```
TEST 1
1ELSE DO
1 TRIGGER 1-TIMING
1 0-BEFORE DATA
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ARM ASYNC, TRIG ON WR 
1 SAMPLE PERIOD 1 * 1-100NS
1 WORD RECOGNIZER=XXXXXXXX
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXXXXXXX
1 FILTER=000 NS
END TEST 1
```

the Timing Option may be used as a stand-alone logic analyzer; i.e., it requires no State clocks to trigger. If the program contains any other events or commands, two State clocks from the Personality Module must occur and an additional 60 us must pass before the Timing Option can be triggered.

MAIN ACQUISITION AND ASYNCHRONOUS TIMING OPTION RELATIONSHIPS

An event, such as a Main word recognizer is recognized two clock cycles after the actual data occurs from the State clock.

For example, in the short form program:

```
IF
WR # 1
THEN DO
TRIGGER 1-TIMING
1-ARM ASYNC, TRIG ON WR 
```

the actual occurrence of the arming signal to the Timing Option is 2 clock cycles plus approximately 60 ns after the data clocked into the 7D02 matches the Word Recognizer #1 pattern. The trigger location of asynchronous acquisition then appears on the data occurring on the P6451 channels two State clocks after the Word Recognizer is matched.

When using fast timing resolution and/or the system-under-test has a slow clock rate, it may be necessary to arm the Timing Option two clock cycles PRIOR to the asynchronous event whose timing is to be examined via the P6451 channels.

Halting the system-under-test is under the control of the Main acquisition section only. The Timing Option acquisition and triggering have no effect on it. If 1-SYSTEM UNDER TEST HALT is selected (beneath the TRIGGER 0-MAIN menus), then the halt line is asserted (this is Personality Module-dependent) when the Main section has triggered and its delay counter has been satisfied.

In some cases, with some Personality Modules, the 7D02 Main system-under-test clock stops when the Personality Module-dependent halt line is asserted. If the Timing Option is in synchronous mode, it may fail to finish acquiring data if it wasn't finished at the time the Main acquisition did finish. Even if the Timing Option were in asynchronous mode, if it wasn't armed by a TRIGGER 1-TIMING command before the system-under-test clock stops, it may never be armed. The STOP key would have to be pressed to terminate the program.

ASYNCHRONOUS OPERATION EXAMPLES

The following examples illustrate some of the ways in which the Timing Option can be used asynchronously.

ARM TIMING OPTION FROM MAIN, THEN TRIGGER TIMING

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=01
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1 OR IF
1 COUNTER # 1 = 00100 0-EVENTS
1 THEN DO
1 TRIGGER 1-TIMING
1 0-BEFORE DATA
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ARM ASYNC, TRIG ON WR
1 SAMPLE PERIOD 1 * 1-100NS
1 WORD RECOGNIZER=0000XXXX
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXXXXXXX
1 FILTER=000 NS
END TEST 1

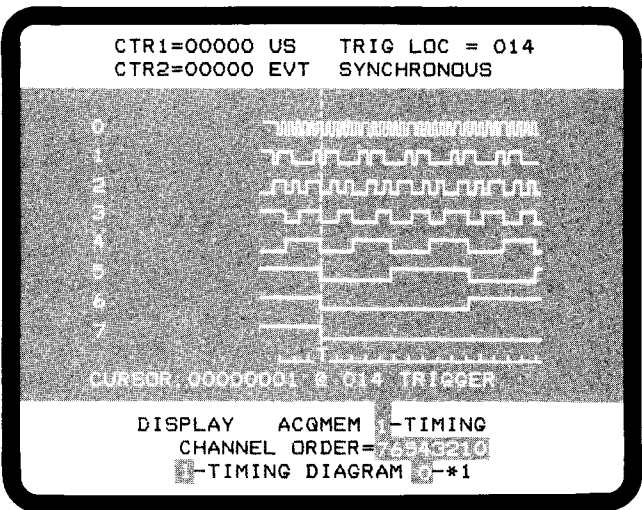
```

When the value 01 has appeared on the s.u.t. data bus 100 times, arm the Timing Option. Trigger the Timing Option when 0000XXXX is seen by the Timing Option word recognizer.

The acquired data can be displayed as:

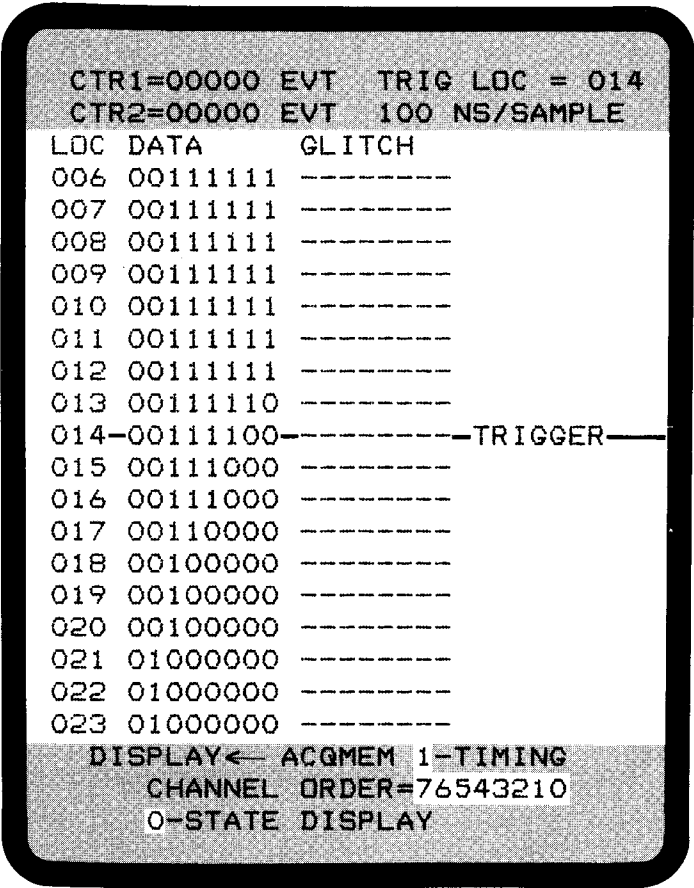
1. A timing diagram (Figure 6-10), or
2. A timing state table (Figure 6-11).

There is no Main acquisition data because there is no TRIGGER 0-MAIN command in the program.



2918-70

Figure 6-10. Asynchronous timing diagram display.



2918-71

Figure 6-11. Asynchronous timing state table display.

TRIGGER MAIN FROM TIMING OPTION EVENT RECOGNIZER

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=0
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ASYNC
1 SAMPLE PERIOD 1 * 1-100NS
1 WORD RECOGNIZER=0000XXXX
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXXXXXXX
1 FILTER=000 NS
1THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

When the Timing Option Event recognizer detects the absence of the bit sequence 0000XXXX for the duration of the s.u.t. clock period, then trigger the Main data acquisition process.

NOTE

The sample period selection in the Timing Option event recognizer specification defines the period over which a glitch can occur. (Greater than one transition between samples.)

This program has TIMING WR=0. If the Timing Option word recognizer output signal is FALSE for the entire state clock cycle, then the signal is considered FALSE and TIMING WR=0 becomes TRUE. If the output of the Timing Option word recognizer goes TRUE for any detectable period of time during a clock cycle, then that TRUE signal is pulse stretched to the end of that clock cycle, and TIMING WR=0 is FALSE. In other words, a FALSE signal must be FALSE the entire clock cycle, but a TRUE signal only needs to be recognized as TRUE sometime during the clock cycle.

In the preceding example, the Main section will be triggered when the data at the P6451 probe remains unequal to 0000XXXX for an entire state clock period.


The acquired data can be displayed as:

1. A mnemonic display, or
2. An absolute display.

There is no timing data acquired because there is no TRIGGER 1-TIMING command in the program.

ARM TIMING OPTION IMMEDIATELY, TRIGGER THE TIMING OPTION FROM ITS OWN WORD RECOGNIZER

```

TEST 1
1ELSE DO
1  TRIGGER 1-TIMING
1    0-BEFORE DATA
1    THRESHOLD V. = 0-PLUS 1.40
1    1-ARM ASYNC, TRIG ON WR 
1    SAMPLE PERIOD 1 * 1-100NS
1    WORD RECOGNIZER=1111XXXX
1    EXT TRIG IN=X
1    GLITCH RECOGNIZER=XXXXXXXX
1    FILTER=000 NS
END TEST 1

```

In this program the TRIGGER 1-TIMING command in the ELSE clause arms the asynchronous trigger "immediately," (subject to the conditions described above).

However, this program is a special case, in that the Timing Option is armed even when there are no State clocks. The External Trigger Out signal goes TRUE as soon as TRIGGER 1-TIMING is executed, and the Timing Option is armed to look for its own trigger. This allows the 7D02 to be used as an 8-channel timing logic analyzer even where there is no clock supplied by the s.u.t.

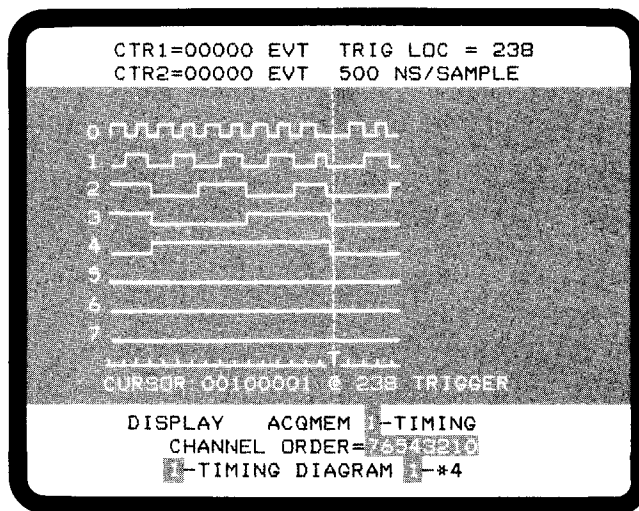
TRIGGERING ON A GLITCH

```

1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=F83C
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 TRIGGER 1-TIMING
1 2-AFTER DATA
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ARM ASYNC, TRIG ON WR
1 SAMPLE PERIOD 5 * 1-100NS
1 WORD RECOGNIZER=XXXXXXXX
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXXXXXXX1
1 FILTER=000 NS
END TEST 1

```

This program will trigger on the first occurrence of a glitch (on channel 0 of the Timing Option P6451 probe) after F83C appears on the s.u.t. address bus (as recognized by Word Recognizer #1).



Once a glitch has occurred on channel 0, this display appears on the screen. Note that the presence of a glitch is indicated by a glitch mark prior to the actual trigger location.

2918-72

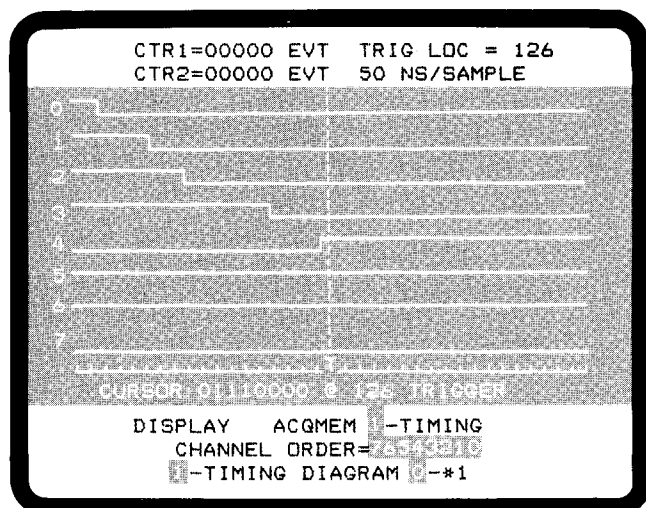
HOW THE FILTER DELAYS THE TIMING OPTION TRIGGER

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=F83C
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 TRIGGER 1-TIMING
1 1-CENTERED
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ARM ASYNC, TRIG ON WR
1 SAMPLE PERIOD 5 * 0-10 NS
1 WORD RECOGNIZER=XXXX1XXXX
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXXXXXXXX
1 FILTER=000 NS
END TEST 1

```

After the Main Word Recognizer #1 recognizes the address F83C, the Timing Option is armed. When channel 4 of the P6451 goes HI, the Timing Option is triggered.

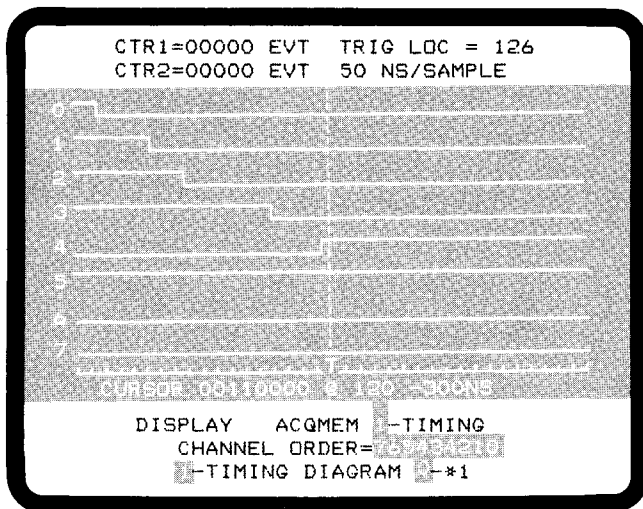


2918-73

For a filter value of 000, the trigger indicator, T, on the tic mark line of the screen, immediately follows the transition from LO to HI on channel 4.

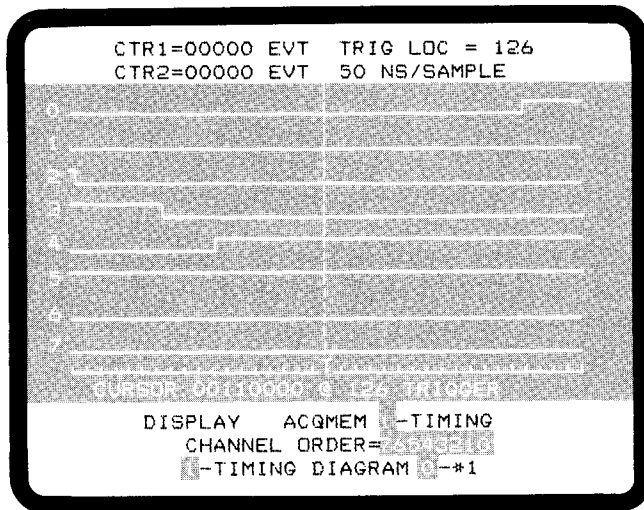
If this transition does not appear in the data, it will be displayed as a glitch.

However, when a filter is specified, the display will be slightly different. This display is shown below.



2918-74

With a filter value of 300 ns, channel 4 is required to stay HI for at least 300 ns before the trigger can occur. This display shows that the channel 3 edge is -300 ns from the T; i.e., the amount of the filter specified.




2918-75


This display shows the T at the cursor position and the LO to HI transition 300 ns before the trigger.


ARM TIMING OPTION FROM TIMING OPTION EVENT RECOGNIZER, THEN TRIGGER TIMING OPTION

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=1
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ASYNC
1 SAMPLE PERIOD 1 * 1-100NS
1 WORD RECOGNIZER=00111100
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXXXXXXXXX
1 FILTER=000 NS
1THEN
1 TRIGGER 1-TIMING
1 THRESHOLD V. = 0-PLUS 1.40
1 1-ARM ASYNC, TRIG ON WR 
1 SAMPLE PERIOD 1 * 1-100NS
1 WORD RECOGNIZER=00111100
1 EXT TRIG IN=X
1 GLITCH RECOGNIZER=XXXXXXXXXX
1 FILTER=000 NS
END TEST 1

```

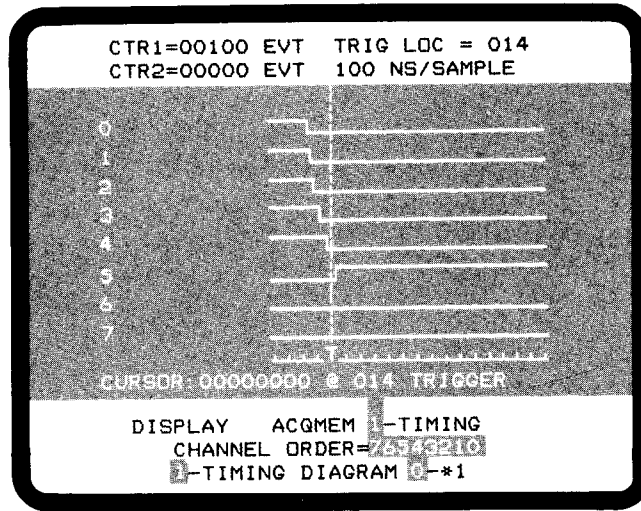
In this program the first occurrence of 00111100 on the P6451 probe arms the Timing Option because the TRIGGER 1-TIMING command is executed. The recognition of 00111100 must go from TRUE to FALSE to TRUE again to cause a trigger. This is because the user selected 1-ARM ASYNC, TRIG ON ; i.e., rising EDGE of the Timing Option Event recognizer to trigger on.

If the program is rewritten to have 2-ARM ASYNC, TRIG ON WR , then as soon as the 00111100 recognition becomes FALSE, the trigger will occur.

The acquired data can be displayed as:

1. A timing diagram (Figure 6-12), or
2. A timing state table (Figure 6-13).

There is no Main acquisition data because there is no TRIGGER 0-MAIN command in the program.



2918-76

Figure 6-12. Timing diagram.

CTR1=00100 EVT TRIG LOC = 014		
CTR2=00000 EVT 100 NS/SAMPLE		
LOC	DATA	GLITCH
006	00011111	-----
007	00011111	-----
008	00011110	-----
009	00011100	-----
010	00011000	-----
011	00011000	-----
012	00010000	-----
013	00010000	-----
014	00000000	-----TRIGGER-----
015	00000000	-----
016	00100000	-----
017	00100000	-----
018	00100000	-----
019	00100000	-----
020	00100000	-----
021	00100000	-----
022	00100000	-----
023	00100000	-----
DISPLAY←ACGMEM 1-TIMING		
CHANNEL ORDER=76543210		
0-STATE DISPLAY		

2918-77

Figure 6-13. Timing state table.

TRIGGER MAIN AND TIMING OPTION FROM TIMING OPTION ASYNCHRONOUS WORD RECOGNIZER

```

TEST 1
1ELSE
1
1  TRIGGER 1-TIMING
1    0-BEFORE DATA
1    THRESHOLD V. = 0-PLUS 1.40
1    1-ARM ASYNC, TRIG ON WR
1    SAMPLE PERIOD 1 * 1-100NS
1    WORD RECOGNIZER=0101XXXX
1    EXT TRIG IN=X
1    GLITCH RECOGNIZER=XXXXXXXXXX
1    FILTER=000 NS
1  GOTO 2
1
END TEST 1
TEST 2
2IF
2 WORD RECOGNIZER # 1
2 DATA=XX
2 ADDRESS=XXXX
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=0
2    THRESHOLD V. = 0-PLUS 1.40
2    1-ASYNC
2    SAMPLE PERIOD 1 * 1-100NS
2    WORD RECOGNIZER=0101XXXX
2    EXT TRIG IN=X
2    GLITCH RECOGNIZER=XXXXXXXXXX
2    FILTER=000 NS
2THEN DO
2 TRIGGER 0-MAIN
2    0-BEFORE DATA
2    0-SYSTEM UNDER TEST CONT.
2    0-STANDARD CLOCK QUAL.
END TEST 2

```

Arm the Timing Option immediately to look for its own trigger.

Note that in this example the selection of the TIMING WR is 0; i.e., FALSE.

In this program the ELSE clause in TEST 1 arms the asynchronous Timing Option and transfers to TEST 2 in the next State clock after the START key is pressed. The Timing Option triggers as soon as its word recognizer goes from FALSE to TRUE. The Main section triggers as soon as the Timing Option Event recognizer stays FALSE for a full State clock cycle.

SUMMARY OF ASYNCHRONOUS MODE



Table 6-2 is a summary of the key points to remember about the use of the Timing Option in asynchronous mode.

Table 6-2
SUMMARY OF ASYNCHRONOUS MODE

Asynchronous Mode	
Acquisition	<ol style="list-style-type: none"> 1. Separate logic analyzer. 2. Uses internal timebase for data and glitch acquisition. 3. Stores glitch information and acquisition data.
Recognition	<ol style="list-style-type: none"> 1. Two sub-modes: <ol style="list-style-type: none"> a. Used with Main word recognizers (pulse stretched). b. Used to trigger Timing Option (edge sensitive). 2. Defined in Main word recognizer and/or in Timing Option trigger command. 3. Performs glitch recognition. 4. Uses a filter value in the range 000—300 ns to allow removal of spurious events.
Triggering	<ol style="list-style-type: none"> 1. Executing a TRIGGER 1-TIMING command merely arms the Timing Option. Subsequently, the Timing Option event recognizer going FALSE to TRUE or TRUE to FALSE (programmable edge) causes the trigger to occur.

TIMING OPTION CLOCKING MODE SELECTION

The Timing Option can only be used in one mode (synchronous or asynchronous) at any given time. The Main word recognizers with the TIMING WR= set to 0 or 1, and the Timing Option trigger specification contain mode menus which are linked together. If you change the value of one to SYNC, the other one also automatically changes to SYNC. If you change the value of one menu to ASYNC, the other menu will change to ASYNC, too. It is not possible to set one of these menus to SYNC and the other to ASYNC. Specifically, the fields are:

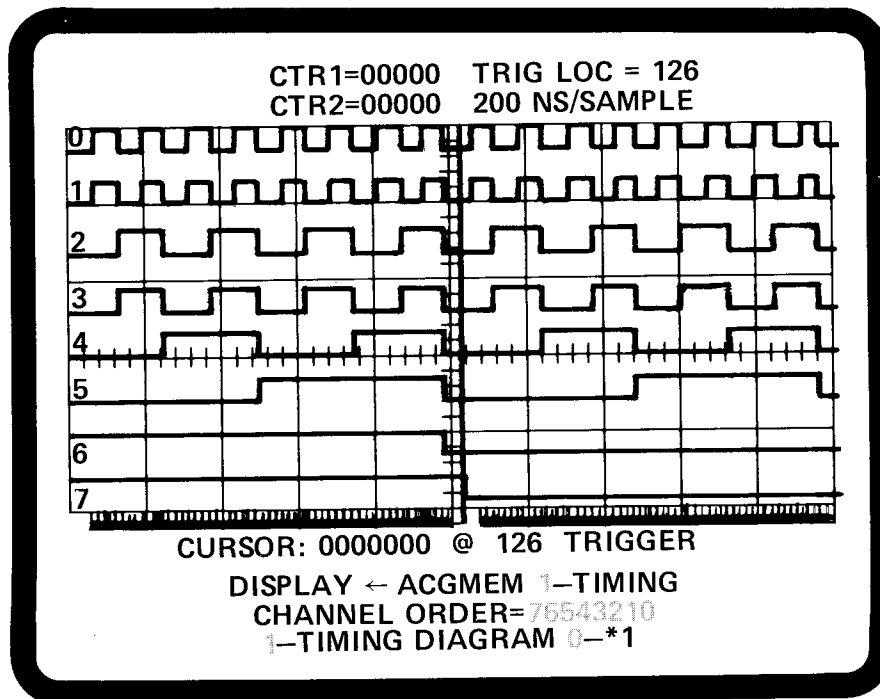
1. TIMING WR= bit in the Main word recognizer—when it is set to 0 or 1, Timing Option word recognizer information is displayed below that field. One of those items is the Timing Option mode specification menu whose choices are SYNC and ASYNC.
2. TRIGGER command—when 1-TIMING is selected, a series of menu items is displayed. One of the items is the Timing Option acquisition mode menu, whose choices are SYNC, ASYNC WR , and ASYNC WR .

TIMING OPTION DISPLAYS

The eight lines of the Timing Option are displayed separately from the Main acquisition data. Timing Option data can be displayed as a timing diagram or as a state table. The selection of the mode in which to display data is made as a menu selection in the Immediate area at the bottom of the screen. Also selected there are the channel ordering, and for the timing diagram, whether the data display magnification is times 1 (*1) or times 4 (*4). These features are discussed below. (Also see the Display Mode description in Section 3).

TIMING DIAGRAM

The timing diagram is a graphic display of the rise and fall of each data line and the timing relationship between changing data on each channel and on other channels. Glitches are also shown above each data channel. Figure 6-14 is an example of a timing diagram display.



2918-78

Figure 6-14. Timing diagram display.

DATA AND GLITCHES

The data display consists of 8 channels. Each channel display consists of a pair of screen lines, the upper for glitch information and the lower for timing information. (There is no glitch display in synchronous acquisition mode.)

Glitches are displayed for asynchronously acquired data if they occur. Their display can be disabled via a menu in the FORMAT mode. A glitch occurs when more than one transition occurs between successive sample periods. Glitches are displayed on the timing diagram as arrows placed above the channel, i.e., trace, on which they occur.

First order glitches are defined as an even number (greater than 1) transitions between samples. Second order glitches are defined as an odd number (greater than 1) transitions between samples.

A first order glitch appears as an arrow, i.e., a glitch mark, over the high or low level on that channel. A second order glitch is indicated by a glitch mark over a low-to-high or high-to-low transition. See Figure 6-15. In state table displays, a glitch is indicated by an * in the appropriate column. See Figure 6-16.

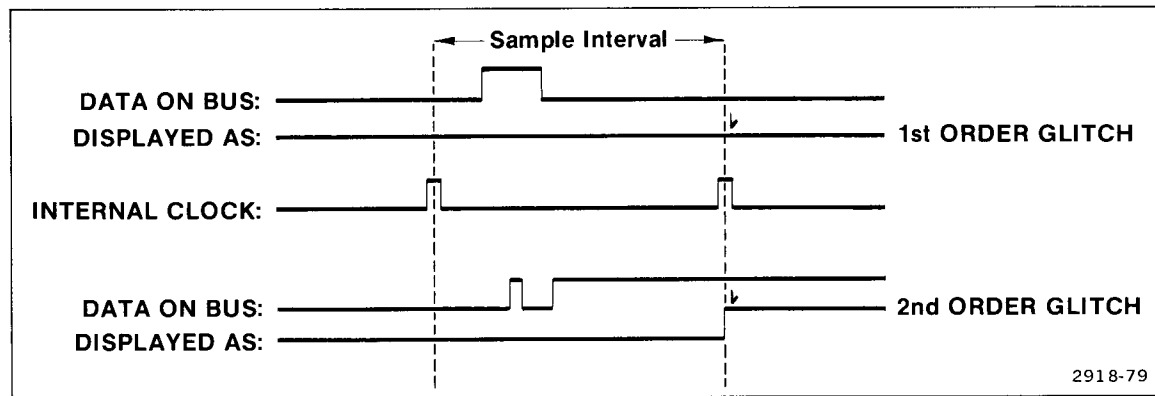


Figure 6-15. Glitch definition and display.

CHANNEL ORDERING

The channel portrayed on any trace is determined by the channel order defined in the Immediate area CHANNEL ORDER= field. The top trace on the display corresponds to the low-order (rightmost) channel in the channel order (and the cursor word). For example, CHANNEL ORDER=53216704 will place channel 4 data at the top of the screen followed below it by channels 0, 7, 6, 1, 2, 3, and finally channel 5 at the bottom, just above the sampling tick marks. Any trace for which an X is selected in the channel order definition area is blank on the screen.

The channel order of the Timing Option Data Inversion field (found in the FORMAT display) and the Timing Option Event recognizer (found in the user-created program) are never affected by the channel order specified for the Timing Option data display; i.e., the leftmost bit always corresponds to channel 7, the rightmost bit to channel 0, and the intermediate bits to channels 6 through 1 respectively.

TICK MARKS

Tick marks appear on the line immediately below the channel waveforms in the display. There is one tick mark for each word of valid data stored in memory. One tick (in *4 magnification) or four tick marks (in *1 magnification) are replaced by a "T" beneath the trigger word. See Figure 6-16.

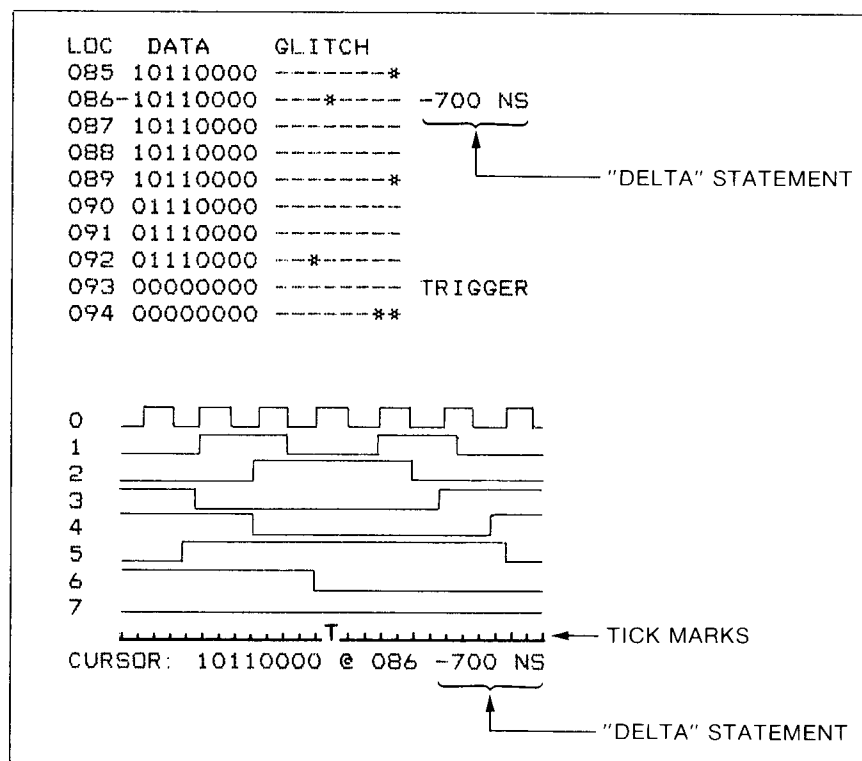


Figure 6-16. Display of trigger placement relative to the cursor.

2918-80

CURSOR INFORMATION

The cursor information line contains:

Data = The value of the data word at the current scroll position (i.e., under the data cursor), reordered according to the current channel order and displayed in the current radix as defined in Format Mode.

If highlighting has been enabled and if Storage Memory contains Timing Option data, then differences between the data stored in the Storage Memory and the data in Timing Option Acquisition Memory are highlighted in inverse video.

loc = The location of the cursor word relative to the start of the acquired data.

delta = Distance between the cursor location and the trigger location (or TRIGGER, if it is the trigger word). See Figure 6-16.

For synchronously acquired data, delta equals the number of stored data words between the trigger and the present cursor location; for asynchronously acquired data, delta equals the time difference between the data cursor and the trigger (i.e., the number of clock cycles multiplied by the sample period that was used to acquire the data).

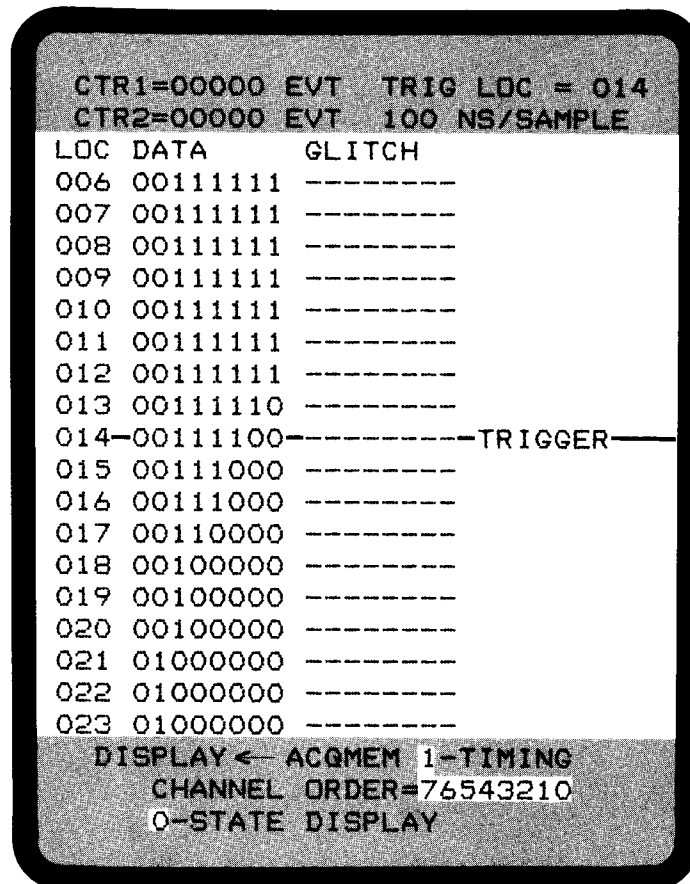
MAGNIFICATION

The timing diagram can be displayed in two magnifications: times 1 (0-*1) and times 4 (1-*4). The magnification is selected when a timing diagram is displayed on the screen. Selection is made from the last menu on the bottom line of the Immediate mode area. The selected magnification remains in effect until changed by the user. The default is times 1 magnification.

STATE TABLE DISPLAY

The state table display contains three fields (LOCation, DATA, GLITCH). Both the DATA and GLITCH fields display data lines in the order specified in the CHANNEL ORDER= field).

A typical state table display might look like Figure 6-17.



2918-81

Figure 6-17. Timing Option state table display.

LOC = location of the displayed cursor word in Acquisition Memory.

TRIGGER = location of the trigger word.

DATA = data word, reordered according to the current channel order (shown at the bottom of the display).

Timing Option—7D02 Operators

GLITCH = glitch information reordered according to the current channel order (where "*" indicates a glitch on the channel and "—" indicates no glitch). This display is replaced with blanks if the data was acquired synchronously or if the glitch display was disabled by the user (by a change to the FORMAT mode menu).

——— = data cursor location.

If there was no acquired data to display, a NO DATA ACQUIRED message is displayed.

CHANNEL ORDERING



Channel ordering for state table displays is specified in the Immediate area, the same as in the Timing diagram display. The low-order (rightmost) channel of the data display corresponds to the rightmost channel of the channel order field, and on up (to the left) to channel 7. Any channel for which an X is selected in the channel order definition area is deleted from the display, and the remaining channels are brought together. If the Timing Option state table display is in a radix other than binary, the deleted channels are taken out of the data word and the value is taken from the remaining channels. Each time a channel is deleted, a 0 is added to the high-order end of the value before its radix is converted.





For example, if the channel order is 76543210, with data (in hex) = F7 (11110111), then if the leftmost channel gets deleted with an X, the channel order is now X6543210 and the data = 77 (01110111). If the rightmost channel is deleted instead, the channel order becomes 7654321X and the data becomes 7B (01111011).

HIGHLIGHTING STORAGE MEMORY DIFFERENCES

Differences between the Timing Option Acquisition Memory and Storage Memory can be highlighted on the display in reverse video, on a digit by digit basis via a FORMAT mode menu. (See the description of Format Mode in Section 3.) This means that the digits of line 000 of the Timing Option Acquisition Memory are compared to the digits of line 000 of the Storage Memory, etc. If there is no data in the Timing Option Acquisition Memory for a given valid address in the Storage Memory, all characters in the data and glitch fields are highlighted on the display.

SCROLLING THE TIMING OPTION DISPLAYS

Once either of the Timing Option displays is on the screen, use the SCROLLING keys ( , ) to move the data past the display cursor. The display will change as the data moves. Note that these keys will move the data up and down the screen for the state table, and left and right across the screen for the Timing diagram. See Figure 6-18.

Use the CURSOR keys ( ,  ,  , ) to move the blinking cursor from field to field in the Immediate Mode command display area.

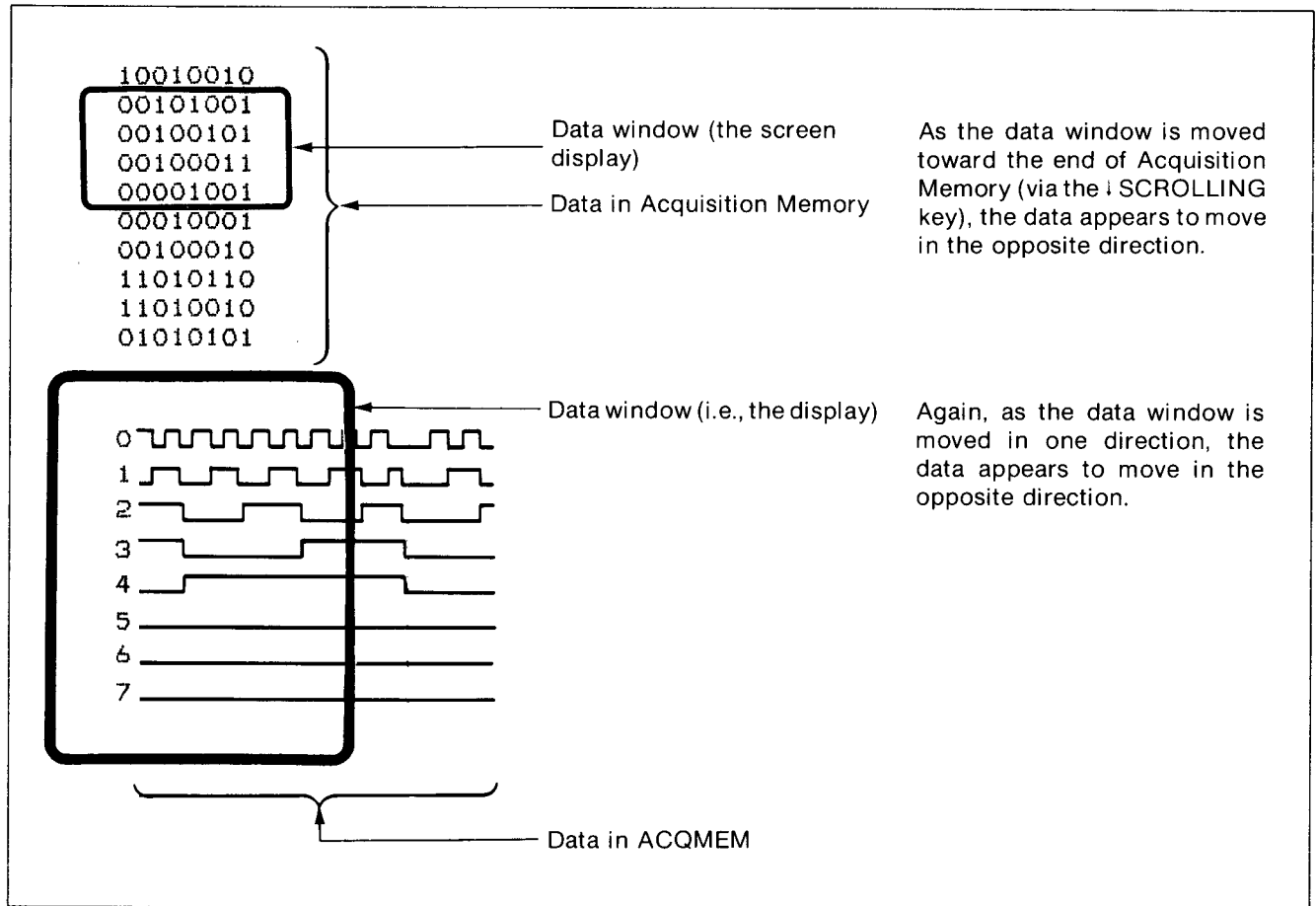


Figure 6-18. Apparent data movement on the display screen.

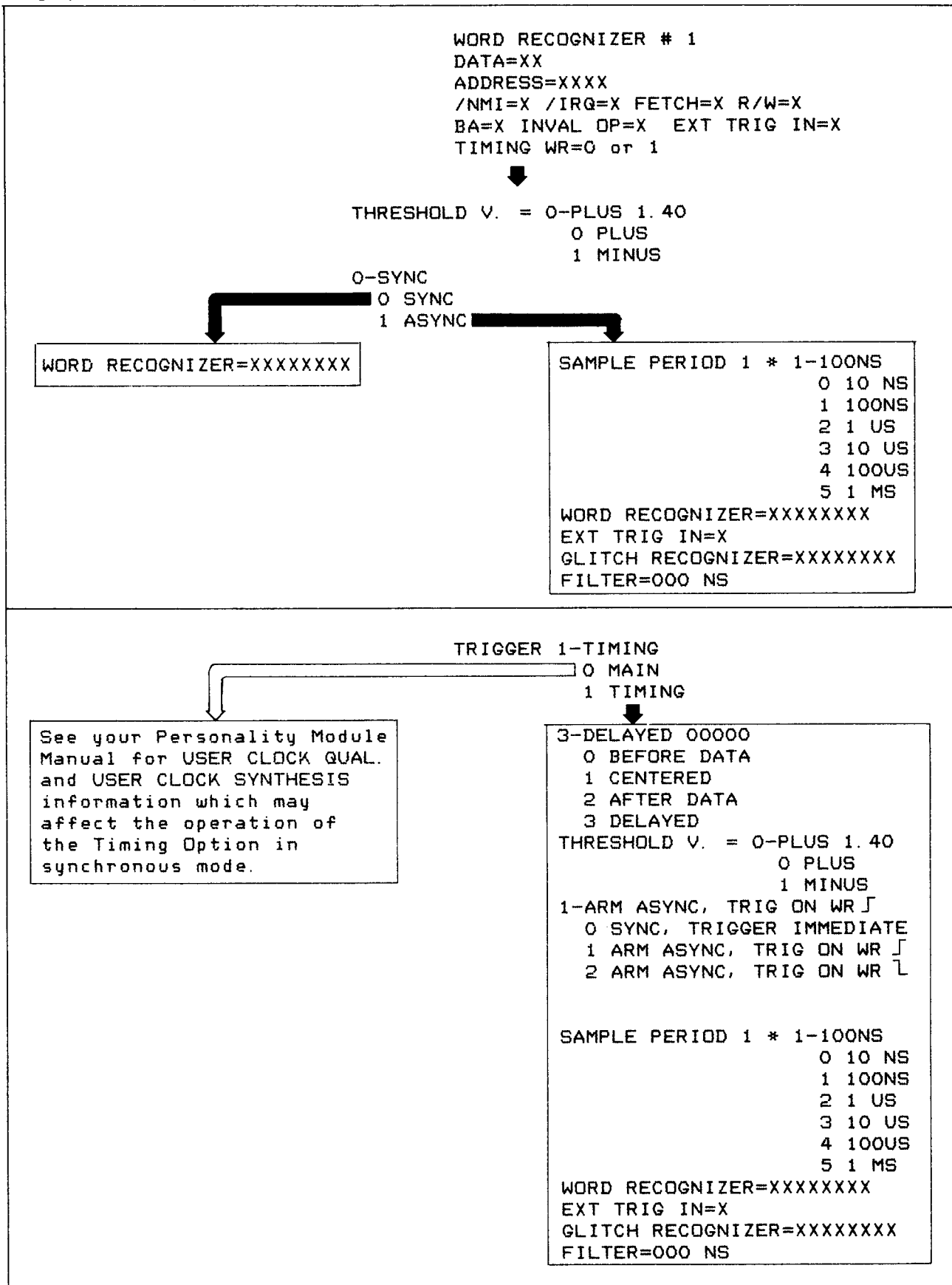
2918-82

TIMING OPTION PARAMETERS

The following paragraphs briefly describe the parameters (menus or numeric fields) associated with the Timing Option.

Figure 6-19 shows the parameters you can expect to see in each of the operating modes—synchronous and asynchronous. It shows all of the fields with their expanded menus (for PM-103 6802 Personality Module). (Incidentally, you will never see this on the 7D02's screen. Only one menu is actually expanded at a time.)

Notice that some fields appear more than once and that the SYNC/ASYNC selection appears twice, but in slightly different words.



2918-105

Figure 6-19. Expanded Timing Option-related menus.

PARAMETER DEFINITIONS

TIMING WR=	<p>This field specifies the desired output signal from the Timing Option Event recognizer. In asynchronous mode, the output signal is the ANDed result of the Timing Option word recognizer, External Trigger In, and the glitch recognizer. In synchronous mode, the output is only the 8-bit word recognizer.</p> <p>X = DON'T CARE 0 = FALSE 1 = TRUE</p>
THRESHOLD V. =	<p>A selectable voltage level for the P6451 probe which defines the input signal level below which the value is a logic LO, and above which the value is a logic HI. The range of selectable values is -6.35 to +6.35 V in 50 mV increments.</p>
Event Recognizer SYNC-ASYNC menus	<p>This menu specifies the mode in which the Timing Option Event recognizer is to be connected to the Main word recognizer.</p> <p>0 = SYNChronous mode; i.e., as an extension of the Main Word Recognizer. 1 = ASYNChronous mode; i.e., the Event Recognizer runs asynchronously.</p>
WORD RECOGNIZER	<p>This field is the 8-channel data specification for the Timing Option Event recognizer.</p>
SAMPLE PERIOD	<p>Under TIMING WR=, this parameter defines the sample period of a glitch. Under TRIGGER 1-TIMING, this parameter specifies the time between successive samples of the inputs of the P6451 probe channels. The valid multipliers are: 1, 2, and 5. The valid clock rates range from the slowest at 5 ms per sample to the fastest at 20 ns per sample.</p>
EXT TRIG IN	<p>This field, appearing only in asynchronous mode, is a ninth channel to the Timing Option word recognizer. The signal comes from the front panel BNC and can be set to:</p> <p>X = DON'T CARE 0 = LO. 1 = HI.</p>
GLITCH RECOGNIZER	<p>This field, appearing only in asynchronous mode, specifies which channels are to be monitored for glitches. The output of the glitch recognizer is the ORed result of the 8 channels.</p> <p>X = DON'T CARE. 1 = Look for a glitch on this channel.</p>
FILTER	<p>This field specifies the amount of time the P6451 and External Trigger inputs must match the Timing Option word recognizer and external trigger programmed values before the input word will be treated as TRUE. The range is 000 to 300 ns in 20 ns increments.</p>
TRIGGER 1-TIMING	<p>The trigger command specifies which acquisition section is to be used to capture data:</p> <p>0 = Main section. 1 = Timing Option.</p>

Timing Option—7D02 Operators



Relative Trigger Position

This menu guarantees the number of memory locations after the trigger.

- 0 = BEFORE DATA, i.e., 240 locations.
- 1 = CENTERED, i.e., 128 locations.
- 2 = AFTER DATA, i.e., 16 locations.
- 3 = DELAYED, i.e., gives the ability to "window" data far downstream from the trigger event. The range is 00000 to 65534. If the delay is 00000, only the data prior to the trigger and the trigger are stored. If the delay is greater than 255, only the last 255 qualified data acquisitions are stored, and not the trigger location.

Triggering/ Acquisition Mode

This menu specifies the mode in which the Timing Option is to operate; and if ASYNC, which edge of the Event recognizer is to cause the actual trigger (after being armed by the TRIGGER 1-TIMING command).

- 0 = SYNC, TRIGGER IMMEDIATE.
- 1 = ARM ASYNC, TRIG ON WR .
- 2 = ARM ASYNC, TRIG ON WR .

TIMING OPTION HARDWARE FUNCTIONAL DIAGRAMS

The following diagrams are provided as an added way of explaining how the 7D02 Timing Option operates. They are not exhaustive! For a more detailed hardware explanation of how the 7D02 works, see the 7D02 Service manual.

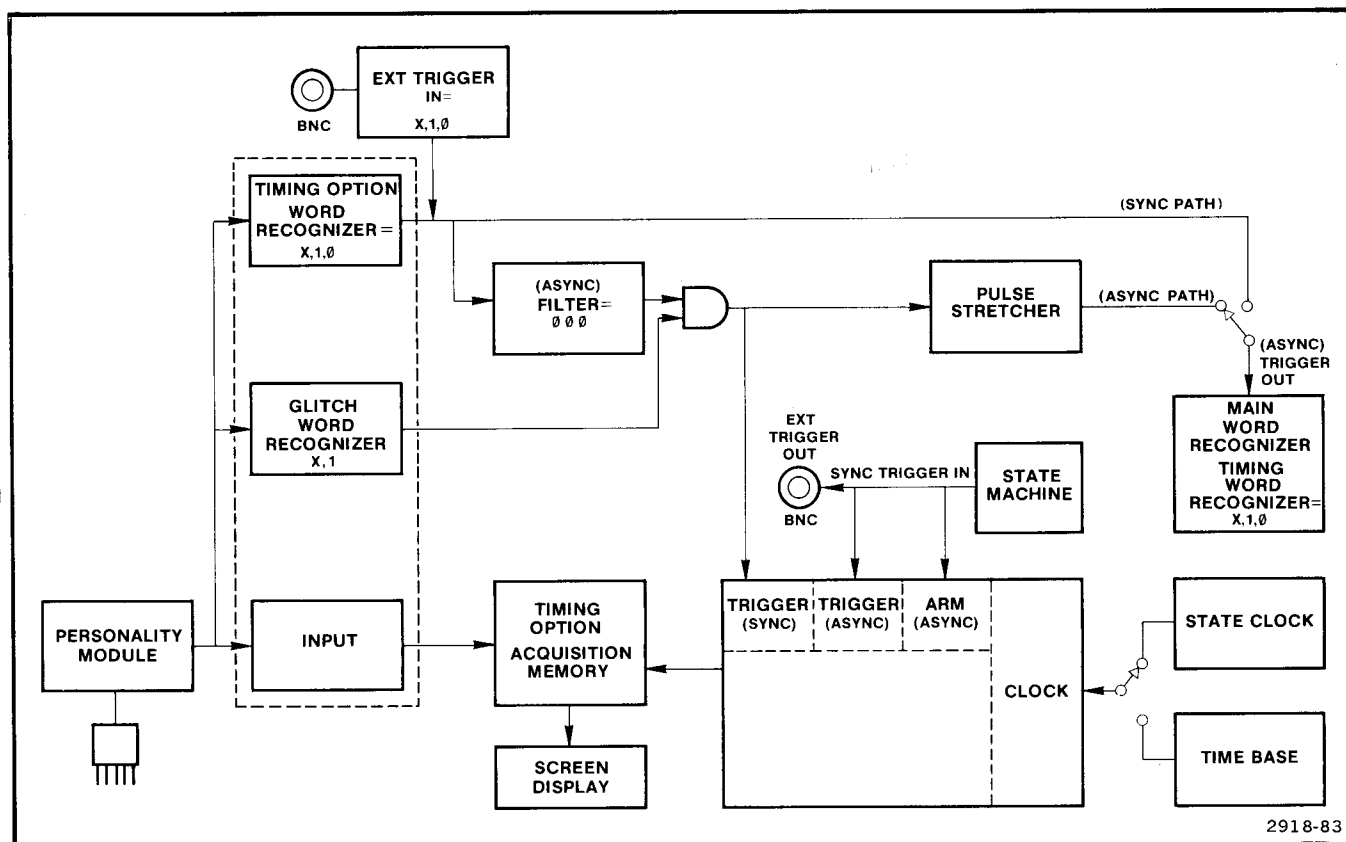


Figure 6-20. Composite functional block diagram for the Timing Option.

SYNCHRONOUS OPERATION

The functional block diagram for synchronous operation is shown in Figure 6-21.

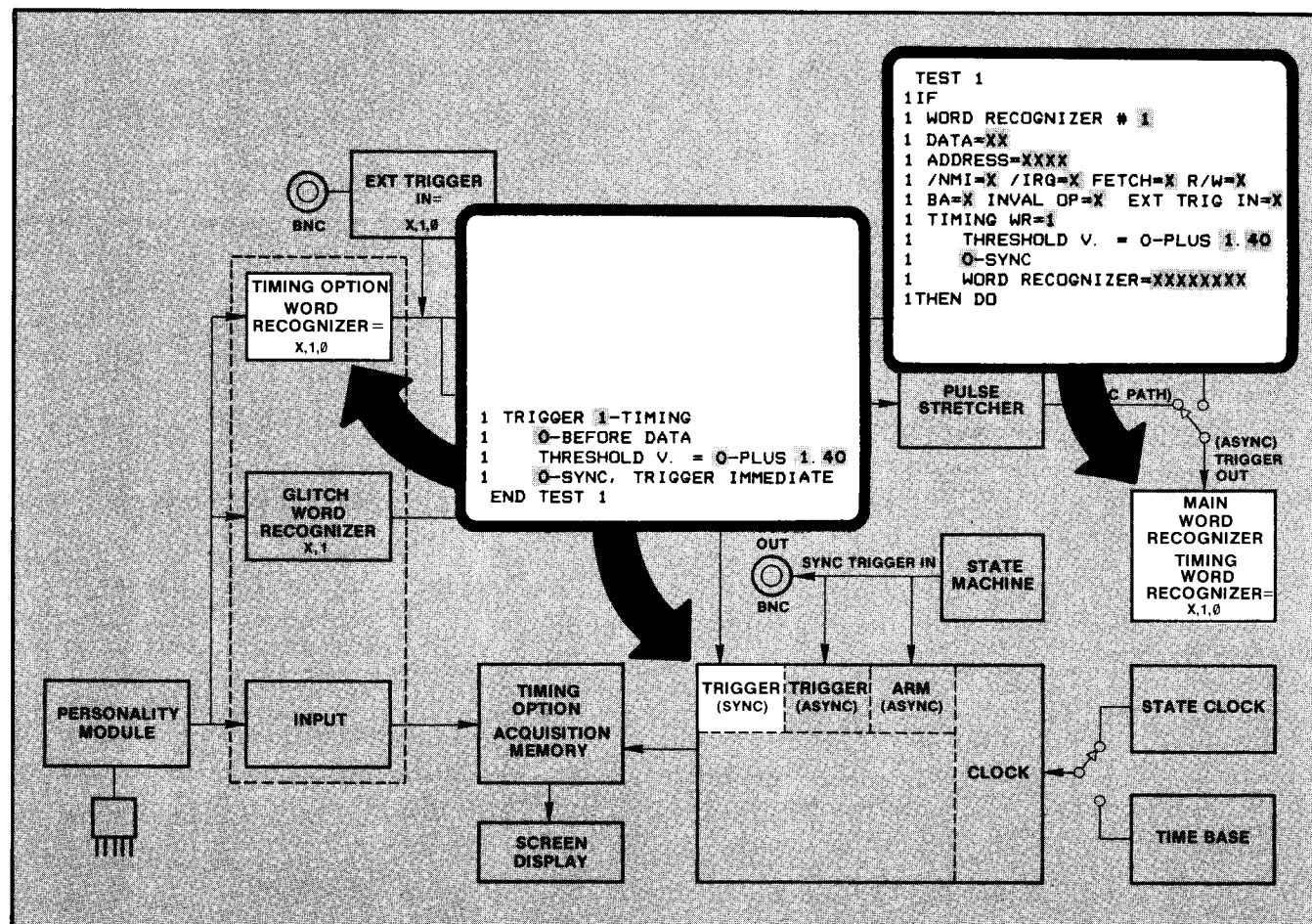


Figure 6-21. Functional block diagram of synchronous Timing operation.

2918-84

A synchronous trigger occurs only as the result of the TRIGGER 1-TIMING command being executed.

ASYNCHRONOUS OPERATION

The functional block diagram for asynchronous operation is shown in Figure 6-22.

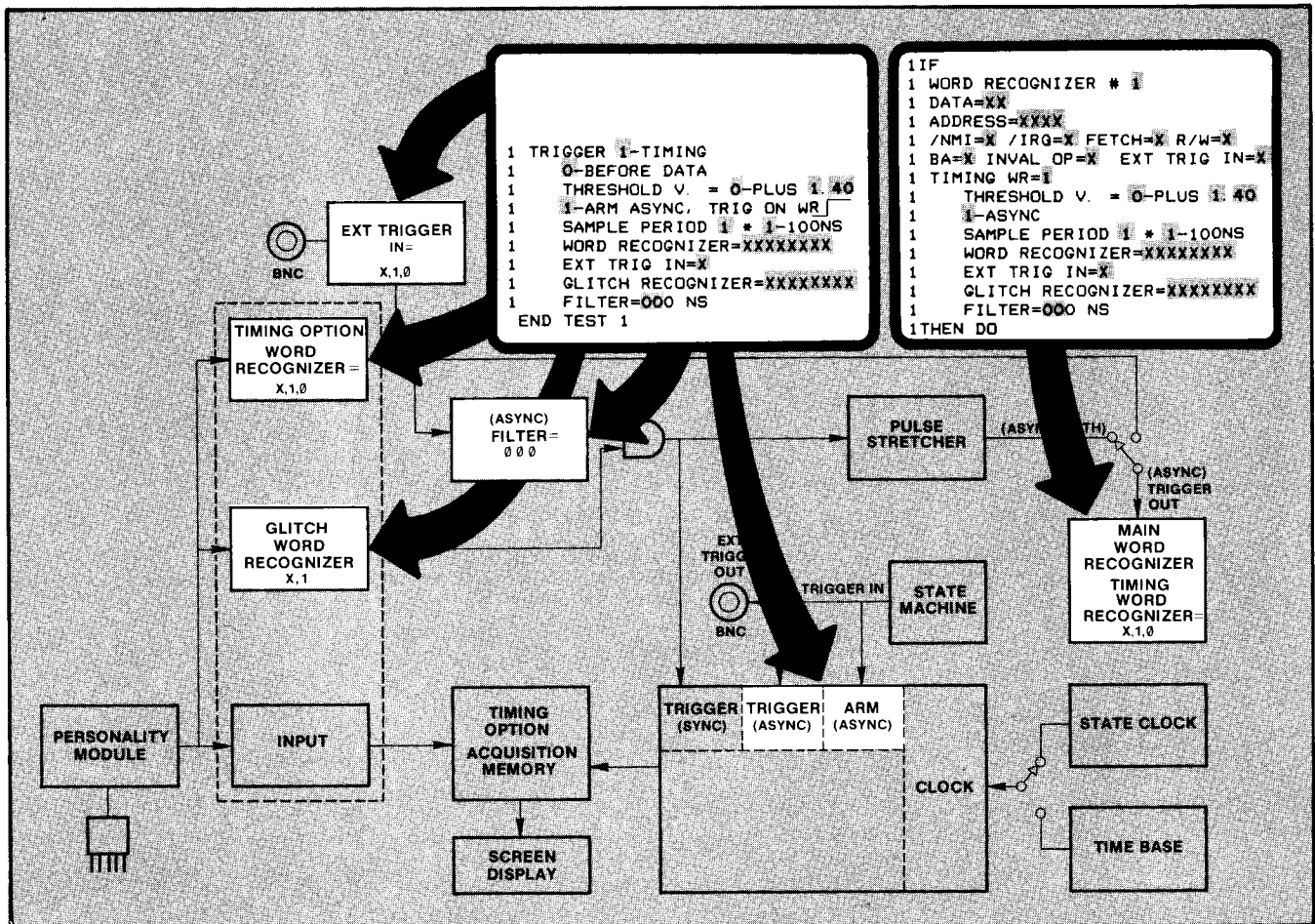


Figure 6-22. Functional block diagram of asynchronous Timing Option operation.

2918-85

An asynchronous "trigger after arming" occurs as the result of two conditions being met. First, some pre-condition must cause the TRIGGER 1-TIMING command to be executed, thus arming the Timing Option word recognizer. Second, the Timing Option word recognizer must generate the specified edge.

Section 7

7D02 RULES OF OPERATION AND SPECIAL NOTES

TABLE OF CONTENTS

7D02 RULES OF OPERATION

- Introduction To This Section
- Requirements For Stopping Data Acquisition
- When Is The 7D02 Running?
- Run-time Operation
- Messages That Appear While The 7D02 Is Running
- Rules Of Concurrency
- Conditions For No Data Stored In The Main Section
- Conditions For No Trigger Location For The Main Section
- Conditions For No Data Stored In The Timing Option
- Conditions For No Timing Option Trigger Location
- Timing Option "Link" Bit To Main Word Recognizer
- Rules Of Operation For External Trigger Out
- Clock Qualification

SPECIAL NOTES

7D02 RULES OF OPERATION AND SPECIAL NOTES

7D02 RULES OF OPERATION

INTRODUCTION TO THIS SECTION

This section consists of several independent discussions which are only loosely related. They talk about how the 7D02's architecture affects the way in which programs are executed and the rules that govern program execution.

REQUIREMENTS FOR STOPPING DATA ACQUISITION

No Trigger 1-TIMING In Program		TRIGGER 1-TIMING In Program
No TRIGGER 0-MAIN In Program	User must press key to end acquisition. Only Main data acquired.	Stops when Timing Option triggers and its delay counter is satisfied. Only Timing Option data is acquired.
TRIGGER 0-MAIN In Program	Stops when Main is triggered and its delay counter is satisfied. Only Main data is acquired.	Stops when both Main and Timing Option are triggered and separate delay counters for each have been satisfied. Both Main and Timing Option data are acquired.

WHEN IS THE 7D02 RUNNING?

The 7D02 acquisition is started by pressing the **START** key or by executing an **IMMEDIATE GOTO** **test #** command. While the 7D02 is running, clock qualification, word recognition, the State machine, and counters (if included in the program) are all running. Acquisition continues as long as those sections (Main and Timing Option) that have a trigger command in the program have not been triggered or have been triggered but their respective delay counters have not been satisfied, unless the user presses the **STOP** key.

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1
TEST 2
2IF
2 WORD RECOGNIZER # 2
2 DATA=XX
2 ADDRESS=XXXX
2 /NMI=X /IRQ=X FETCH=X R/W=X
2 BA=X INVAL OP=X EXT TRIG IN=X
2 TIMING WR=X
2THEN DO
2 TRIGGER 1-TIMING
2 0-BEFORE DATA
2 THRESHOLD V. = 0-PLUS 1.40
2 1-ARM ASYNC, TRIG ON WR
2 SAMPLE PERIOD 1 * 1-100NS
2 WORD RECOGNIZER=XXXXXXXXX
2 EXT TRIG IN=X
2 GLITCH RECOGNIZER=XXXXXXXXX
2 FILTER=000 NS
END TEST 2

```

Only the first of the two triggers will ever occur because there is no path (GOTO) in the program to get to TEST 2 for the TRIGGER 1-TIMING command to be executed. The 7D02 will continue to run until manually stopped.

The important point here is that the delay counters must finish counting down before the State machine stops. This means that if one of the programmable counters (generated by the COUNTER command) is being incremented by a word recognizer or a counter is measuring time, these counting functions can continue after a trigger command is executed. For example:

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=F83C
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1 OR IF
1 WORD RECOGNIZER # 2
1 DATA=XX
1 ADDRESS=F865
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
END TEST 1

```

When Word Recognizer #2 goes TRUE, the Main section is triggered. If the Main section is in BEFORE DATA delay, 240 more acquisitions are needed before the 7D02 stops. If Word Recognizer #1 goes TRUE during any of those 240 acquisitions, count #1 will be incremented for each one.

If the user wants a function to stop when a trigger occurs, the function must be told to stop either with a specific command or with an implied command in another test. For example, to stop the counter from incrementing after the Main section is triggered, the user would enter:

```

TEST 1
1 IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=F83C
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
1 OR IF
1 WORD RECOGNIZER # 2
1 DATA=XX
1 ADDRESS=F865
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1 THEN DO
1
1 TRIGGER 0-MAIN
1 0-BEFORE DATA
1 0-SYSTEM UNDER TEST CONT.
1 0-STANDARD CLOCK QUAL.
1 GOTO 2
1
END TEST 1
TEST 2
END TEST 2

```

This is the same program as the previous example, slightly modified.

In a TEST which uses no counter functions (TEST 2 in this case), the operation of the counters is unchanged. In "Events" mode, the counter value is unchanged. In "time" mode, the counters stay in their previous mode (running or stopped).

In this example, counter #1 will not be incremented after the Main section has triggered because the program goes to TEST #2 and stays there doing nothing.

RUN-TIME OPERATION

When the **START** key is pressed, the 7D02 begins executing the program. It does the following things:

1. Sets up all of the hardware to the conditions and parameters specified in the program and/or to the default conditions and parameters.
2. Starts the 7D02 acquiring data.
3. Monitors the running instrument and displays status messages on the top two lines of the screen (when appropriate). This is repeated until the acquisition is finished or until the user presses the STOP key.
4. Stops any hardware that is still running, reads the pertinent data, and makes calculations.
5. Changes from Run mode to Display mode (see Section 3) and displays the acquired data.

MESSAGES THAT APPEAR WHILE THE 7D02 IS RUNNING

RUNNING

When the **START** key is pressed, the message RUNNING is displayed in the top left corner of the screen and blinked. This message remains until the acquisition is complete or the user presses the **STOP** key. Note that if the 7D02 remains running, e.g., there is no TRIGGER command in the program, the **STOP** key is the only way to remove the 7D02 from the running state.

Line-level Message

One signal from the Personality Module is monitored while the 7D02 is running. The level of this line is displayed in the top right corner of the screen. The text of the HI/LO messages are Personality Module-dependent.

For a microprocessor-based Personality Module, the halt line of the microprocessor is usually monitored. When it is not halted, no message is put in this area; i.e., it is filled with blanks. If the microprocessor is halted, a message is put into the area to indicate the halt condition.

For the General Purpose Personality Module, the status of control line 7 is monitored and a C7 HIGH or C7 LOW message is right-justified in the upper right corner of the screen. Thus, while the 7D02 is running (acquiring data), the level of control line 7 is being reported in real time with an update time of less than one half of a millisecond.

With this real-time update feature, control line 7 can function like a logic probe. While the 7D02 executes a do-nothing program (TEST 1 END TEST 1), the user can probe around in a circuit and read the level of the line from the display at the top right corner of the screen.

MAIN DONE

TIMING DONE

The center of the top line of the screen holds either the message MAIN DONE or TIMING DONE if the following conditions occur:

1. Both the Main section and the Timing Option are being used. "Being used" means that a TRIGGER 0-MAIN and a TRIGGER 1-TIMING command both appear in the program.
2. For the MAIN DONE message to be displayed, the Main section must have triggered and its acquisition finished while the Timing Option is still acquiring data.
3. For the TIMING DONE message to be displayed, the Timing Option must have triggered and its acquisition finished while the Main section is still acquiring data.

NO CLOCK

SLOW CLOCK

The left side of the second line on the screen is normally blank, but can contain clock exception messages.

1. If no State clocks have occurred after the **START** key is pressed, the NO CLOCK message is displayed.
2. If the State clock rate is sporadic or has a rate of less than 1 kHz, then a SLOW CLOCK message is displayed.

One exception exists. If the entire program consists of ELSE TRIGGER 1-TIMING, and the system-under-test input clock is not connected or is not running, the message will correctly read NO CLOCK when the START key is pressed. However, if the input clock is connected and receives transitions after the **START** key has been pressed, the message will not change while the 7D02 is running.

PRESENT TEST =
MAIN TRIGGER TEST =

The 7D02 program can have up to four tests with GOTO commands directing run-time program flow between the tests. The right half of the second line of the screen contains the update of what test the 7D02 is currently executing. This is indicated by the PRESENT TEST = n message (n = the test number).

NOTE

This display never reports on a Timing Option trigger. When the Main section is triggered, the test in which the trigger occurred is noted and the message is changed to MAIN TRIGGER TEST = n (n = the test number). The 7D02 can still be running if the Timing Option is being used in the program and it is still acquiring data, or if the Main section has not yet finished acquiring data (for instance, if there is heavy data qualification specified in the program).

RULES OF CONCURRENCY

Everything occurs concurrently in the same test, unless a conflict exists. In that case, the first event and command of the conflict takes precedence.

1. All bracketed ORed events are looked for simultaneously. The following skeletal programs illustrate this.

IF

```
WORD RECOGNIZER # 1
OR
WORD RECOGNIZER # 2
```

THEN DO

IF

```
WORD RECOGNIZER # 1
OR
COUNTER # 1 = 00100
OR
COUNTER # 2 = 11111
```

THEN DO

On each clock cycle the acquired data is compared to both Word Recognizer #1 and Word Recognizer #2. If either compare is TRUE, then the command following the THEN DO is executed.

On each clock cycle the acquired data is compared to Word Recognizer #1, the value of counter #1 is compared to 00100, and the value of counter #2 is compared to 11111. If any one of the compares is TRUE, the command following the THEN DO is executed.

2. All ORed event/command combinations are monitored simultaneously. The following skeletal program illustrates this.

IF

```
WORD RECOGNIZER # 1
THEN DO
  TRIGGER
OR IF
  COUNTER # 1 = 05500
THEN DO
  GOTO n
```

On each clock cycle the acquired data is compared to Word Recognizer #1 and the value of the counter is compared to 05500. If the word recognizer compare is TRUE, the trigger occurs. If the counter compare is TRUE, program execution transfers to TEST n. If both compares are TRUE, in the same clock cycle, both events occur; i.e., the trigger occurs and program execution transfers to TEST n.

3. All bracketed commands are executed simultaneously when the associated event goes TRUE. The following skeletal program illustrates this.

IF

```
WORD RECOGNIZER # 1
THEN DO
```

```
COUNTER # 1
  INCREMENT
COUNTER # 2
  INCREMENT
```

On each clock cycle the acquired data is compared to Word Recognizer #1. If the compare is TRUE, both counter #1 and counter #2 are incremented.

```

TEST 1
1IF
1 WORD RECOGNIZER # 1
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 COUNTER # 1 0-EVENTS
1 1-RESET
1OR IF
1 WORD RECOGNIZER # 2
1 DATA=XX
1 ADDRESS=XXXX
1 /NMI=X /IRQ=X FETCH=X R/W=X
1 BA=X INVAL OP=X EXT TRIG IN=X
1 TIMING WR=X
1THEN DO
1 COUNTER # 1 0-EVENTS
1 0-INCREMENT
END TEST 1

```

If both word recognizers are TRUE on a single clock cycle, counter #1 will be reset because it is specified first in the program. (The counter cannot be RESET and INCREMENTED at the same time.)

In general, if there are two or more event/command pairs in a test whose commands are contradictory, the conflict is resolved in the following manner:

1. If only one event is TRUE, then the command clause associated with that event is executed.
2. If both events are TRUE, then:
 - a. All non-contradictory commands are executed, and
 - b. Where contradictory commands exist, the command defined earliest in the text is executed.

The following skeletal program and truth table illustrate this.

```

IF
  WORD RECOGNIZER # 1
THEN DO
  COUNTER # 1
    INCREMENT
  TRIGGER 0-MAIN

OR IF
  WORD RECOGNIZER # 2
THEN DO
  COUNTER # 1
    RESET
  GOTO 3

END TEST 1
TEST 2
:
:
END TEST 2
TEST 3
:
:
END TEST 3

```

WR #1 FALSE		WR #1 TRUE
WR #2 FALSE	No Operation	COUNTER #1 INCREMENT TRIGGER 0-MAIN
WR #2 TRUE	COUNTER #1 RESET GOTO 3	COUNTER #1 INCREMENT TRIGGER 0-MAIN GOTO 3

CONDITIONS FOR NO DATA STORED IN THE MAIN SECTION

The conditions under which the 7D02 does not store data in Main Acquisition Memory (ACQMEM) are:

1. The Main section is not being used; i.e., there is no TRIGGER 0-MAIN command in the program, and the Timing Option is used (TRIGGER 1-TIMING).

OR

2. Less than three input clocks occurred before the user pressed the **STOP** key.

OR

3. The data qualification conditions were not met.

CONDITIONS FOR NO TRIGGER LOCATION FOR THE MAIN SECTION

A TRIG LOC = *** message appears on the top right of the acquired data display when:


1. No data was acquired (see the previous discussion);

OR

2. The user pressed the **STOP** key before acquisition was finished. This can occur even if the Main section was triggered, but the delay counter had not yet reached its programmed value before the **STOP** key was pressed.

CONDITIONS FOR NO DATA STORED IN THE TIMING OPTION

The following conditions will result in no data being stored in the Timing Option Acquisition Memory.

1. There is no TRIGGER 1-TIMING in the program.
 - OR
 2. The user pressed the **STOP** key, and
 - a. Less than three s.u.t. input clocks have occurred when the Timing Option is in synchronous mode;
 - OR
 - b. There is heavy qualification of data with the qualification criteria not met (when using the Timing Option in synchronous mode);
 - OR
 - c. No State clocks have occurred. The following program is an exception. It will acquire data with no State clocks.
- ELSE
 TRIGGER 1-TIMING
 1-ARM ASYNC, TRIG ON 

CONDITIONS FOR NO TIMING OPTION TRIGGER LOCATION

The following conditions can result in no Timing Option trigger location being displayed.

1. The Timing Option acquired no data (see the discussion above).
- OR
2. The Timing Option delay is programmed with a value larger than 254.
- OR
3. The user pressed the **STOP** key before the acquisition was finished. This can occur even if the Timing Option was triggered, but its delay counter had not reached its programmed value before the **STOP** key was pressed.

TIMING OPTION "LINK" BIT TO MAIN WORD RECOGNIZER

The Timing Option "link" bit connects the Timing Option Event recognizer to a Main word recognizer. "Connects" means the polarity specified is ANDed to the rest of the parallel bits making up the Main word recognizer. If only the Timing "link" bit is set, the Main word recognizer is TRUE as long as the Timing Option Event recognizer output matches the "link" bit polarity.

In asynchronous mode, the Timing Option Event recognizer is the ANDing together of the 8 channels of P6451 data and the External Trigger In bit (subject to the specified filter value), and 8 channels of ORed glitch recognition. Whenever they all become TRUE, the link bit goes TRUE.

NOTE

The External Trigger In bit can be thought of as a ninth channel of Timing Option 8-bit word recognition because it is subject to the specified filter value.

Since the glitch recognizer channels are based on the internal sample interval (a glitch is defined as more than one transition on a channel between clock samples), so the link bit value is based on the sample interval whenever one or more "1's" are specified in the glitch recognizer.

The Main word recognizers are level sensing devices. That is, the output is TRUE if the input signals match the specified setup values when the signal is sampled on a State clock edge and remains TRUE for as many consecutive clock cycles as the input signal continuously matches the specified values. As soon as there is no match, the output goes FALSE.

This leads us to the discussion of the Timing Option "link" bit in relation to the separate clock speeds of the Main section and the Timing Option.

When the Timing Option is in asynchronous mode and the sample rate is faster than the State clock, the Timing Option Event recognizer's ANDed output can go FALSE before being sampled by the Main word recognizer. Therefore, this signal is stretched, i.e., latched, TRUE until the next State clock sample occurs. This guarantees that a fast asynchronous event will be seen by the synchronous Main section.

If the Main section has a fast State clock (say a 100 ns period) and the Timing Option is running at some slower speed (say a 1 μ s internal sample period), the link bit can be TRUE for multiple State clock samples when a glitch is found. (A glitch, by definition, is determined over the entire 1 μ s.) This is true because the specified resolution of the Timing Option is now much less than that of the Main section. Therefore, use care when glitches are used to control counters or to transfer program execution from one TEST to another.

RULES OF OPERATION FOR EXTERNAL TRIGGER OUT

It should be noted that the actual External Trigger Out signal occurs two State clocks plus a small delay after the event recognized occurred at the Personality Module tip. See Appendix E.

The following rules apply to the operation of the external trigger:

1. When the Timing Option is installed in the 7D02:
 - a. If the TRIGGER 1-TIMING command is not in the program, the External Trigger Out is always TRUE, i.e., HI.
 - b. If the TRIGGER 1-TIMING command is in the program, the External Trigger Out goes TRUE, i.e., HI, when the TRIGGER 1-TIMING command is executed. At all other times it is FALSE, i.e., LO.
2. When the Timing Option is NOT installed in the 7D02, the External Trigger Out signal:
 - a. Goes TRUE, i.e., HI, every time the TRIGGER 0-MAIN command is executed. At all other times it is FALSE, i.e., LO.
 - b. Is FALSE (LO) when the 7D02 is not running.

NOTE

This may happen multiple times in one program run, even though only the first will trigger the data acquisition.

CLOCK QUALIFICATION

Clock qualification allows the user to select which edge of which system-under-test clock phase will be used by the 7D02 as its State (storage) clock. (This is not to be confused with data qualification described in Sections 4 and 5.)

The 7D02 always does synchronous data acquisition on a system-under-test Master clock edge, but not necessarily on EVERY system-under-test Master clock edge.

The 7D02's method of clock qualification is one of taking away those clock edges from the system-under-test Master clock for which the data is not valid or not desired. It can also be thought of as a selection method in which the circuitry is programmed to select only the edges of the system-under-test Master clock on which the data is valid and desired to be clocked into the 7D02. C4 through C9 are the control lines which are the inputs to the logic functions that enable this selection process.

The menu items that are displayed are Personality Module-dependent. The most detailed menus are displayed by the General Purpose Personality Module. They are shown below in their expanded form. All other Personality Modules display some subset of these. Clock qualification only appears in the TRIGGER command for the Main section of the 7D02.

```

TRIGGER @-MAIN
  0-BEFORE DATA
  0-SYSTEM UNDER TEST CONT.
  1-USER CLOCK QUAL.
    0 STANDARD CLOCK QUAL.
    1 USER CLOCK QUAL.
  0-RISING EDGE OF CLOCK
    0 RISING EDGE OF CLOCK
    1 FALLING EDGE OF CLOCK
  C9-C4 (ANDED CLOCKS)=XXXXXX
  1-USER CLOCK SYNTHESIS
    0 STANDARD CLOCK SYNTHESIS
    1 USER CLOCK SYNTHESIS
  1-DIVIDE CLOCK BY 1
    0 DELAY CLOCK BY
    1 DIVIDE CLOCK BY
  ESYNC: C6=X OR C8=X
  WAIT: C7=X OR C9=X

```

A brief explanation of each of these fields is given below.

1-USER CLOCK QUAL. This menu item states that the user is going to specify some or all of the characteristics of the Master clock; otherwise, the default values are used. Selecting 1-USER CLOCK QUAL causes the next three menus to appear.

0-RISING EDGE OF CLOCK
1-FALLING EDGE OF
CLOCK

These menu selections specify which edge of the Master clock is the active edge; i.e., which edge is to be used to sample data. In other words, it changes the clock polarity.

C9-C4 (ANDED CLOCKS)= This field specifies which of the six control lines, on a specified condition, are to be ANDED together to be used as qualifiers to select portions of the Master clock to be the State clock for data storage.

0-STANDARD CLOCK SYNTHESIS

This menu item says that the user does not intend to manipulate the clock signal from the s.u.t.

1-USER CLOCK SYNTHESIS This menu item specifies the user's intent to manipulate the clock signal from the s.u.t.; selecting 1-USER CLOCK SYNTHESIS causes the next three menus to appear.

0-DELAY CLOCK BY

This menu item selects a user-specified number of Master clock edges after Esync goes TRUE to generate a data strobe for the 7D02. (A data strobe is the actual clock that causes data storage into the 7D02.) The number of data strobes, then, is equal to the number of times Esync goes TRUE. The range of delay values is 0 to 4.

1-DIVIDE CLOCK BY

This menu item divides down the Master clock to generate a data strobe for the 7D02. The user can select a division from 1 to 4. The Esync line going TRUE synchronizes the division. For example, DIVIDE CLOCK BY 2 selects every other Master clock edge for clocking data in to the 7D02, starting when Esync goes TRUE. This differs from the delay mode in that there can be more input clocks selected from the Master clock than Esync pulses.

ESYNC: C6=X OR C8=X

These lines are ORed together to produce the synchronization for the State (sampling) clock from the s.u.t.; a 0 indicates a negative TRUE signal, while a 1 indicates a positive TRUE. A DON'T CARE causes the line to be ignored.

NOTE

If DELAY is specified and Esync is set to DON'T CAREs, no clock will result; i.e., no storage occurs. If DIVIDE is specified and Esync is set to DON'T CARE, an invalid clock will result.

WAIT: C7=X OR C9=X

These lines are ORed together to produce a wait signal which will add wait cycles to the Master clock. For each Master clock cycle that this wait line is asserted, one extra cycle will be added to the divided or delayed clock before the 7D02 samples the data for storage or word recognition. A 0 indicates a negative TRUE, while a 1 indicates a positive TRUE signal. DON'T CAREs cause these lines to be ignored. The ESYNC function is suspended whenever the WAIT signal is asserted.

NOTE

The user must be careful when using clock qualification and/or clock synthesis with the Timing Option in synchronous mode. The reason is that all of the clock syntheses modifications are done in the Main trigger. If these changes are made and then the Main trigger is deleted from the program, there is no indication in the Timing Option TRIGGER command that these changes were made. However, all of the changes are still in effect and will alter the operation of the Timing Option.

SPECIAL NOTES

1. If the Timing Option is installed, the word recognizer and the State machines continue to operate as long as either the Timing Option or the Main section are running. Neither section by itself can stop the 7D02. If one stops and the other is running, the user has the option of stopping the 7D02 by pressing the **STOP** key.
2. Counters that are being used in time mode, if not specified to stop in a program, will be turned off when both the Main section and the Timing Option are stopped. In other words, it is the last of the two sections (Main or Timing) that determines when the counters stop.
3. Deleting the TRIGGER 0-MAIN with USER CLOCK from the program does not return the clock qualification to its default values.
4. When using the counters in the time mode, errors can occur due to erratic State clock signals. Due to the system architecture of the 7D02, erratic clock signals can delay commands to run, stop, and reset-and-run the counters. These momentary delays will cause errors in the values seen in the counter displays on the screen.

In addition, if the counters are being used as a windowing event, erratic clocks can cause the counters to time out incorrectly.

Correct operation of the counters is assured when the input clocks and the qualified clocks have a regular and consistent time interval.

STATUS AREA MESSAGES

This section lists error and informational messages that can appear in the 2-line status area at the top of the 7D02 screen.

The explanations for many of these messages contain some of the more common actions which cause the messages to appear. Users of the 7D02 should consider these as guidelines, not as inflexible rules of 7D02 operation.

MESSAGE	EXPLANATION
ALL 4 TESTS ALREADY USED	A 7D02 program is limited to a total of four tests.
BINARY FIELD	An attempt was made to enter a value other than 0, 1, or X into a binary field.
BREAK: SELECT REGISTER (0—F)? PC=xxxx	The 7D02 experienced a hardware or firmware failure in a Personality Module. This may happen if you ignored a PERSONALITY MODULE REQUIRED! error message. Power-down the 7D02 and attach the Personality Module correctly. If problems persist, call your Tektronix service representative.
CONTRADICTIONARY BRACKETED COMMANDS	Two commands that cannot be executed simultaneously were placed within a pair of brackets. A blinking E is placed in column 1 to indicate the location of the error. If more than one such error exists, only the first one is marked with a blinking E. When that error is corrected, then the 7D02 will place a blinking E at the location of the next such error.
DECIMAL FIELD	An attempt was made to enter a hexadecimal value (A—F) into a decimal field.
DISASSEMBLER ERROR #X AT XXXX	The 7D02 experienced a hardware or firmware failure. This may happen if you ignored a PERSONALITY MODULE REQUIRED! error message. Power-down the 7D02 and attach the Personality Module correctly. If problems persist, call your Tektronix service representative.

Appendix A—7D02 Operators

"DON'T CARE" INVALID IN THIS FIELD

An attempt was made to enter an key into a field that only accepts numeric keys. Try 0—F.

ENTER 1, 2, OR 5

The multiplier for the Timing Option sampling time base is restricted to 1, 2, or 5.

FATAL ERROR AT xxxx

The 7D02 experienced a hardware or firmware failure. This may happen if you ignored a

PERSONALITY MODULE REQUIRED!

error message. Power-down the 7D02 and attach the Personality Module correctly. If problems persist, call your Tektronix service representative.

FASTEST CLOCK IS 20 NS

User tried to enter 1 for multiplier and 0 for exponent of the asynchronous Timing Option sample period. This exceeds the fastest clock available.

ILLEGAL SCROLLING—USE CURSOR MOVEMENT KEYS

A data scrolling key was pressed when ACQMEM or STORE MEM data was not being displayed on the screen.

KEY INVALID IN FORMAT MODE.
PRESS "FORMAT" TO EXIT

Only cursor movement keys, numeric keys, and **FORMAT** are valid in the Format mode.

MOVE CURSOR TO END OF AREA

A programming key or **DELETE** was pressed while the blinking cursor was not within 1 item of the end of the program.

MUST BE A MULTIPLE OF 20 NS

An attempt was made to enter an odd digit in the second position of the Timing Option filter field.

MUST BE A MULTIPLE OF 50 MV

The Timing Option threshold must have either a 5 or a 0 in the rightmost digit of the voltage value.

NO PROGRAM STORED IN STOREMEM

The Immediate mode command sequence **IMMEDIATE** **PROGRAM** **STORE MEM** was issued when no program was stored in storage memory.

NOT A NUMERIC FIELD

A **0** thru **F** or an **X** key was pressed when the cursor was not on a numeric field.

NOT A VALID SELECTION

A numeric entry other than an allowable value was attempted.

OCTAL FIELD

An attempt was made to enter a value greater than 7 into an octal field.

ONLY "X" AND "1" ARE VALID HERE

The glitch recognizer field only accepts these two values.

PERSONALITY MODULE REQUIRED!
POWER-DOWN BEFORE ATTACHING!

The Personality Module has become disconnected. Continued keyboard operation may result in fatal errors and/or non-standard screen displays. RE-ATTACHING THE PERSONALITY MODULE WITH POWER ON CAN CAUSE PERMANENT DAMAGE TO THE 7D02!

NOTE

This error could occur due to static problems. In that case, striking a new key should make the message disappear.

POWERUP DISGNOSTICS COMPLETE

The 7D02 can now be used to perform logic analysis testing.

PRESS "DISPLAY" "PROGRAM" TO
RETURN TO PROGRAM

An **IMMEDIATE** keystroke was followed by a programming keystroke.

PRESS "IMMEDIATE" TO ENTER A NEW
COMMAND

1. The **DELETE** key was pressed in Immediate mode; or
2. A data movement key was pressed in Program or Display mode, or after a complete Immediate mode command has been entered, but before it was executed.

The **IMMEDIATE** key must be the first key pressed in any data movement operation.

PROGRAM TOO LARGE

The test program exceeds the 7D02's program memory size.

QUALIFY BLOCK ALREADY DEFINED

Only one Qualify block is allowed in a program. The Qualify block is placed before or after any of the four tests. (Qualify commands, however, may be included within any test, in any number.)

STOREMEM IS EMPTY

The key sequence **IMMEDIATE** **DISPLAY** **STORE MEM** was issued, but nothing was previously stored in Storage memory.

NOTE

TRY as the first word of an error message is a flag to the user.

Those error messages in which TRY appears are generated by an incorrect keystroke when the user is entering a program or an Immediate mode command. The list of legal keystrokes following the TRY is not exhaustive! The list only includes the valid programming and/or data movement keystrokes. It may also be legal to press any of the following keys:

FORMAT , **START/STOP** , **IMMEDIATE** , **DELETE** , and **CURSOR** .

TRY "ACQMEM" OR "PROGRAM"

There are only two valid sources following an **IMMEDIATE** **STORE MEM** sequence.

TRY COMMAND, "QUALIFY" OR "[]"

A THEN DO was followed by an **END** **ELSE** , **OR** , **NOT** , or **START/STOP** .

TRY "COUNTER" OR "WD RECOGNIZER"

A **TRIGGER** , **GOTO** , or a Structure key was pressed following a **NOT** key.

TRY EVENT, STRUCTURE, "START", "NOT", OR "[]"

User attempted to begin a test with a GOTO or an OR.

TRY "GOTO", "DISPLAY", "PROGRAM", OR "STOREMEM"

The key sequence **IMMEDIATE** **ACQ MEM** is illegal because **ACQ MEM** cannot be a data movement destination. Use any of the data movement destinations or a **GOTO** command.

TRY "OR" OR "[]"

A bracketed (compound) event must be continued with an **OR** or terminated with a **[]** key.

TRY "STOREMEM"

When **PROGRAM** is the destination of a data transfer, only **STORE MEM** can be the source.

TRY "STOREMEM", "PROGRAM" OR "ACQMEM"	A source must be specified for transfers to the display.
TRY "TRIGGER", "COUNTER", "GOTO" OR "QUALIFY"	The opening bracket of a compound command must be immediately followed by a command.
UNDEFINED TEST	An IMMEDIATE GOTO test # command sequence was issued for a test which does not exist in the program.
VALUE TOO LARGE. LARGEST VALID VALUE USED	An attempt was made to: <ol style="list-style-type: none"> 1. Enter a threshold value greater than 6.35; or 2. Enter a counter value greater than 65,534; or 3. Enter a delay value greater than 65,534; or 4. Enter a filter value greater than 300.
VALUE TOO LARGE FOR FIELD	A digit was entered in the most significant end of a field where there are not enough bits to hold the digit specified. For example, an 8-bit octal field, if a 7 is specified for the most significant digit, this message appears.
VALUE TOO SMALL. SMALLEST VALID VALUE USED	An attempt was made to set a counter to a value of 0 or 1.
"X" INVALID IN THIS FIELD	The data field requires a numeric value.

SHORTCUTS

SHORTCUTS

When entering programs on the 7D02 it is not always necessary to enter every programming keystroke. ("Programming keys" are **WD RECOGNIZER**, **COUNTER**, **TRIGGER**, **GOTO**, **[]**, **OR**, **NOT**, **END**, **ELSE**, and **QUALIFY**). Under some circumstances the 7D02 will supply them for you. These are known as "Shortcuts".

AUTOMATICALLY SUPPLIED ENDS

After the event of a Qualify block, or after the command of an ELSE clause, only the **END** key is legal. If any other programming key is pressed, an END will be supplied automatically. The programming key will be treated in the same manner as it would had an **END** been pressed by the user.

After an IF or OR IF prompt, pressing **START** will supply an END. The 7D02 will run normally.

NOT KEY

After IF or OR IF prompts, or after any END, or in an empty program, pressing the NOT key will supply an opening bracket (thus acting as if **[] NOT** had been entered). Note that the user must manually supply the close bracket at the end of the event (by pressing the **[]** key).

WD RECOGNIZER IN A COMMAND CLAUSE

After a THEN DO prompt, or after the **ELSE** key has been pressed, pressing the **WD RECOGNIZER** key acts as if **GOTO** **END** **WD RECOGNIZER** were entered.

Entering a sequential trigger with this shortcut is simple. Press the **WD RECOGNIZER** key once for each step desired; then press **TRIGGER**.

EMPTY-PROGRAM SHORTCUTS

If the **TRIGGER** key is the first key pressed in a program, it is as if **WD RECOGNIZER** **TRIGGER** **END** had been entered except that the cursor is left within the TRIGGER block.

Pressing the **START** key with an empty program will provide the same program (i.e., **WD RECOGNIZER**, **TRIGGER**, **END**) and run it.

Word Recognizer #1 and the Trigger will contain the same values as the last time they were used. If the 7D02 has just been powered up then Word Recognizer #1 will be all DON'T CARES, so the 7d02 will trigger immediately.

7000-SERIES MAINFRAME SET-UP CONDITIONS

The following table lists the appropriate settings for the various switches and controls on those 7000-Series mainframes in which the 7D02 can be installed.

NOTE

For all 4-wide mainframes, this chart assumes that the 7D02 occupies the three leftmost compartments of the mainframe.

SCOPE NO.	VERT MODE	STORE UPPER & LOWER	ENHANCE UPPER & LOWER	AUTO ERASE	NON-STORE	SAVE	READ OUT INTENSITY	HORIZ MODE
7104 7313 7313R	Left Left Left	OFF OFF	OFF OFF	OFF OFF			OFF OFF	A
7403 7403N 7503	Left Left Left						OFF	
7504 7504A 7603	Left Left Left						OFF	A
7603R 7603N11S 7613	Left Left Left					ON	OFF	
7623 7623A 7633	Left Left Left				ON ON ON		OFF OFF OFF	
7704 7704A 7834	Left Left Left	See NOTE. See NOTE.			ON		OFF OFF	A A
7844 ^a R7903 7804	Vert Beam 1 = Left Beam 2 = OFF Left Left						OFF OFF	Beam 1 = A A

^a Use beam 1 only. Turn OFF Beam 2.

NOTE

Intensity should be turned up in ALL places. The 7704 mainframe has three intensity controls. The A sweep intensity should be turned full on and the main intensity adjusted for brightness. In all other 4-wide mainframes there are two controls for intensity: one for A sweep and one for B sweep. In these mainframes it is only necessary to adjust the A sweep intensity.

SELF-TEST PROCEDURES AND ERROR MESSAGES

INTRODUCTION

This appendix explains the diagnostics for the 7D02. The diagnostics package can be broken into two main parts:

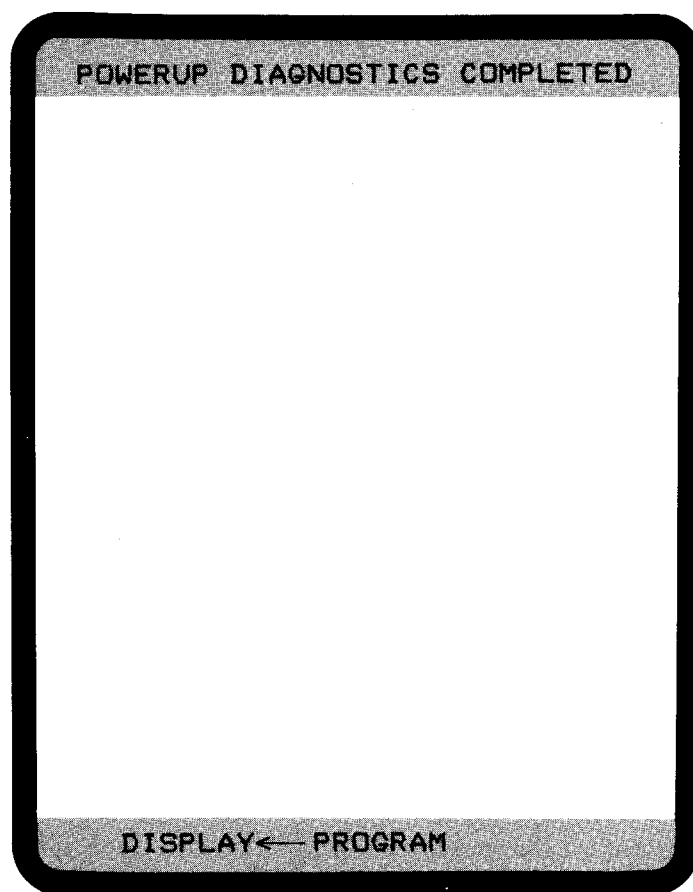
1. Power-up verification which runs automatically at power-up, and
2. Diagnostics mode which can be run by a qualified service person at the conclusion of power-up verification if an error has occurred.

NOTE

Diagnostics mode can be entered by holding a key down during power-up verification, causing a keyboard failure.

POWER-UP VERIFICATION

The power-up verification routines run automatically each time the 7D02 is powered-up. (See the 7D02 Service manual for more information on this subject.) At power-up the screen is cleared of any previous information, the test is run and a message is displayed. If the diagnostic tests are successful, a message is displayed automatically (see Figure D-1), and the system is ready to accept keyboard commands.



2918-86

Figure D-1. Successful power-up display.

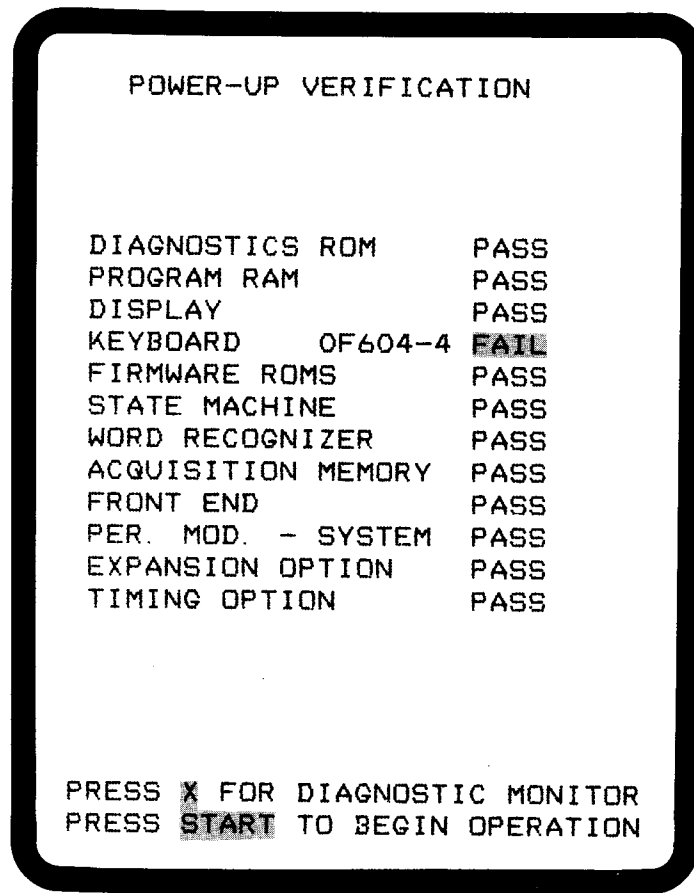
If the power-up verification is not successful, the display indicates which test(s) failed. (See Figure D-2.) At this point you have two choices:

1. Press the **X** key to select the diagnostic menu, or
2. Press the **START** key to place the 7D02 in its normal operating mode. By doing this you say, in effect, that the diagnostic test(s) which failed are in parts of the 7D02 hardware which your programs will not use. For example, if the Timing Option fails and you will not do timing data acquisition, then you can usually operate the instrument.

DIAGNOSTICS

The 7D02 diagnostics are divided into two parts:

1. Module Test mode, and
2. Signature Exerciser mode.



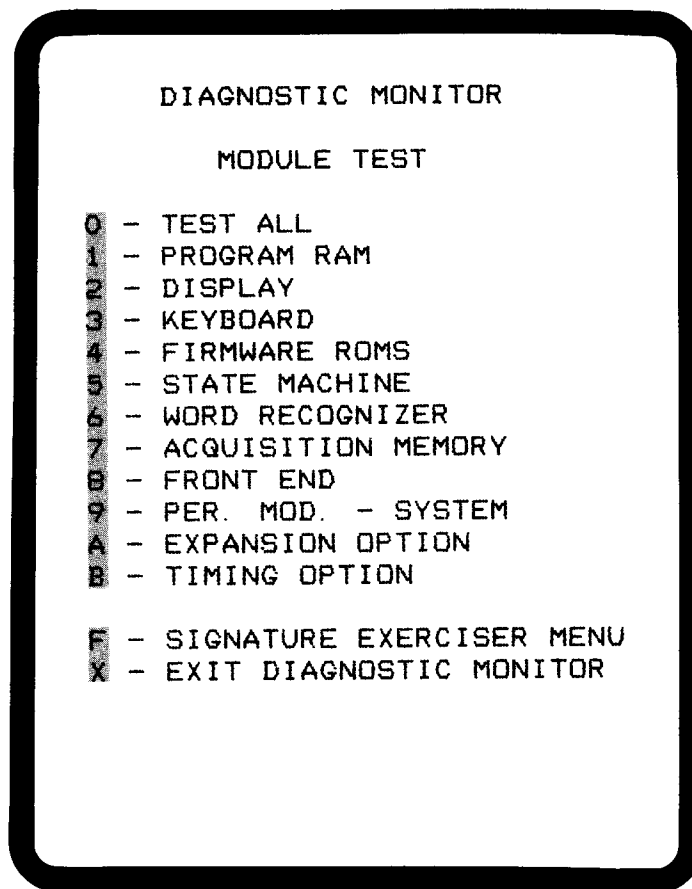
2918-87

Figure D-2. Unsuccessful power-up display.

MODULE TEST MODE

The Module Test mode is entered if there is a test failure during power-up verification. Pressing the **X** key results in the diagnostic menu being displayed on the screen in place of the power-up message. (See Figure D-3.)

The selection of individual diagnostic tests from the diagnostic module test menu can begin now. The individual test numbers appear in reverse video.



2918-88

Figure D-3. Diagnostic monitor module menu.

Menu items 1 through B are individually selectable test modules. Item 0 selects all tests (1 thru B) and executes them in ascending numerical sequence. Menu items A and B will only appear if the appropriate hardware is present in the instrument. Item F is not a test, but a branch to the menu for another set of diagnostic tests using signature analysis.

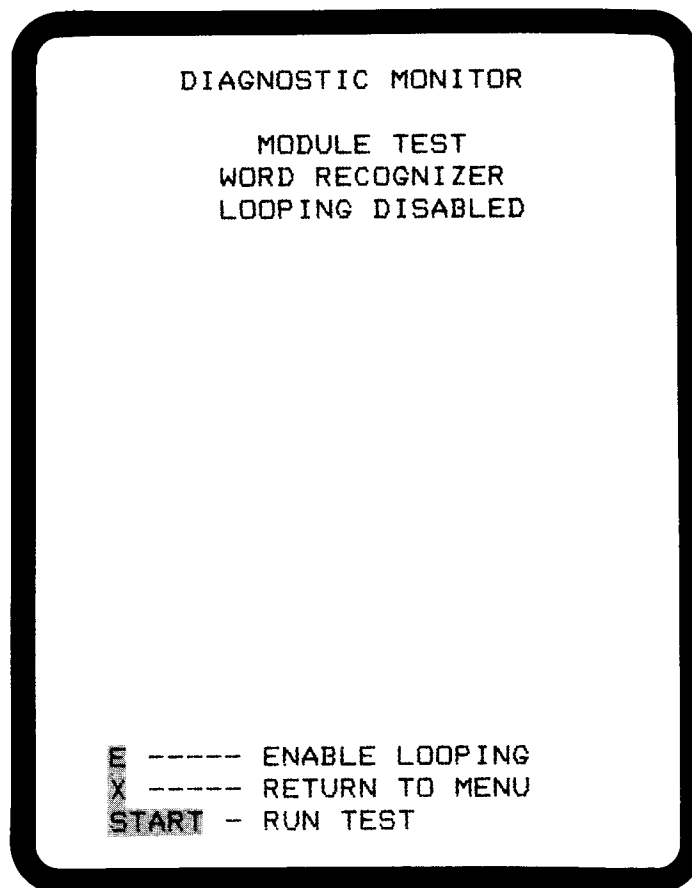
Once the selected test(s) are completed, you can return to the module test menu or rerun the selected test(s). If you return to the menu, any test (0 thru B) can be selected, or the signature exerciser menu can be selected by pressing the **F** key.

If an individual test is selected, a display similar to Figure D-4 is displayed on the screen (in this case the key was pressed). Once the **START** key is pressed, the central portion of the screen is cleared of instructions. This area is used to display the results of individual tests within the selected module test. See Figure D-5.

NOTES

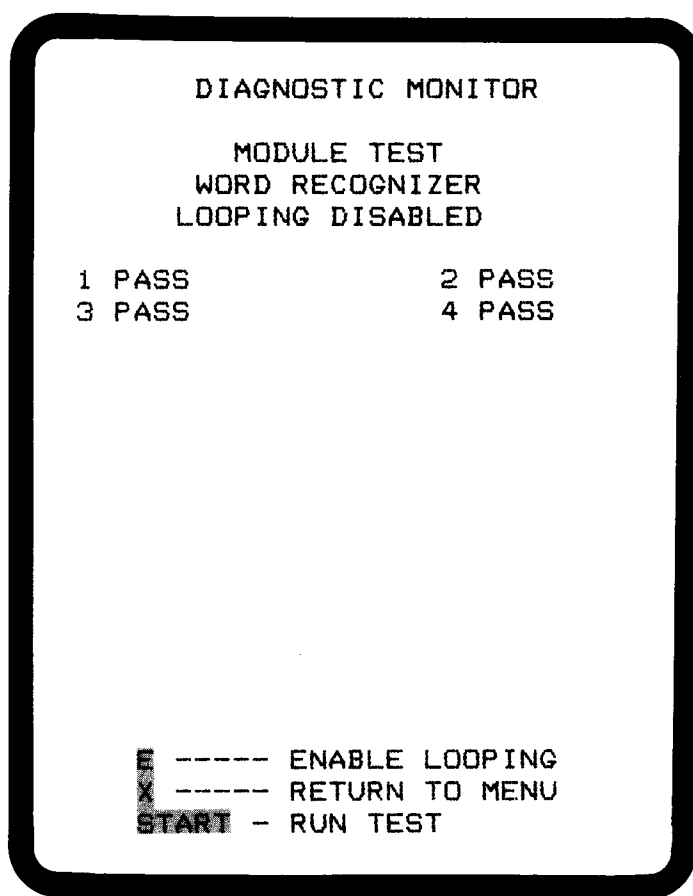
If a test fails, the PASS indication in Figure D-5 is replaced by a FAIL indication. A particular failure code directs a service person to a particular portion of the circuitry.

If a particular test requires that a Personality Module be connected to the Service Test, a message to that effect will be displayed when the test is selected by the user.



2918-89

Figure D-4. Individual module test instruction display.



2918-90

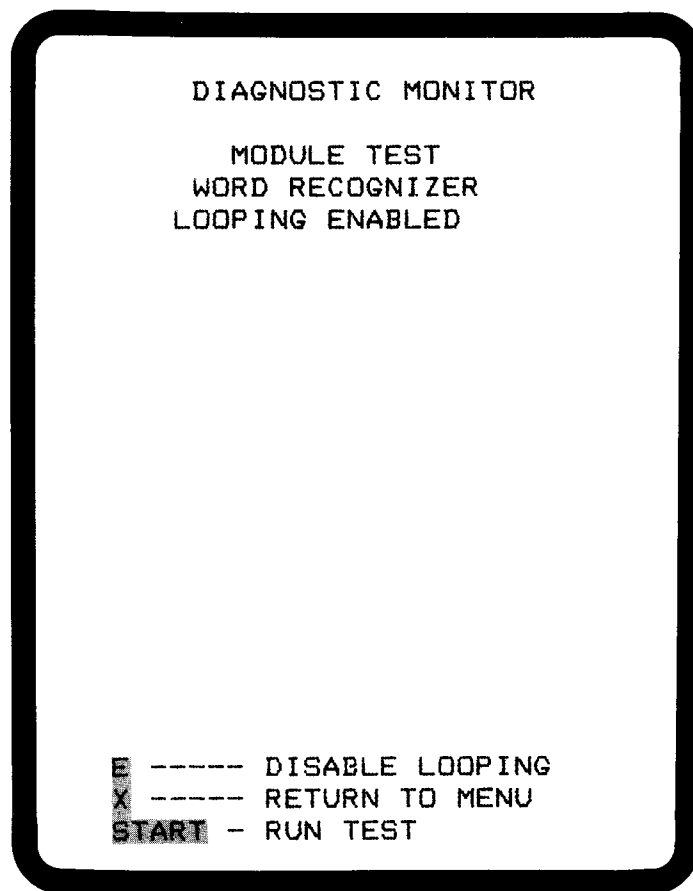
Figure D-5. Individual module test result display.

If the **STOP** key is pressed while a series of tests is being run, the tests currently being executed run to completion and stop. If the test is the Program RAM test or ALL tests, it may be necessary to press the **STOP** key for up to a minute.

If a test module has looping enabled, the program loops on the first test in the module test. At the completion of that "subtest", the word PASS or FAIL will be displayed beside the test number. If the test failed, then the address at which the failure occurred is also displayed. The subtest will be repeated until the **STOP** key is pressed. Then the next subtest will be executed.

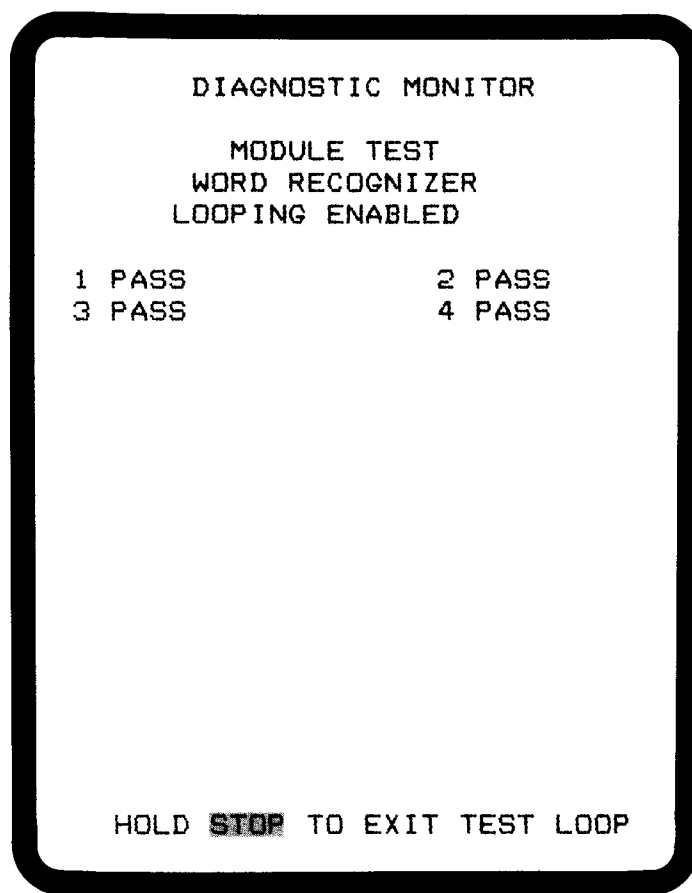
If a subtest fails more than once, the failure code of the latest failure will be displayed; i.e., the failure code is updated each time a failure occurs.

At the completion of the module test the user may press the **START** key to perform the module test again, press the **D** key to disable looping, or press the **X** key to return to the diagnostic menu display. (See Figures D-6 and D-7).



2918-91

Figure D-6. Module test "LOOPING ENABLED" display.



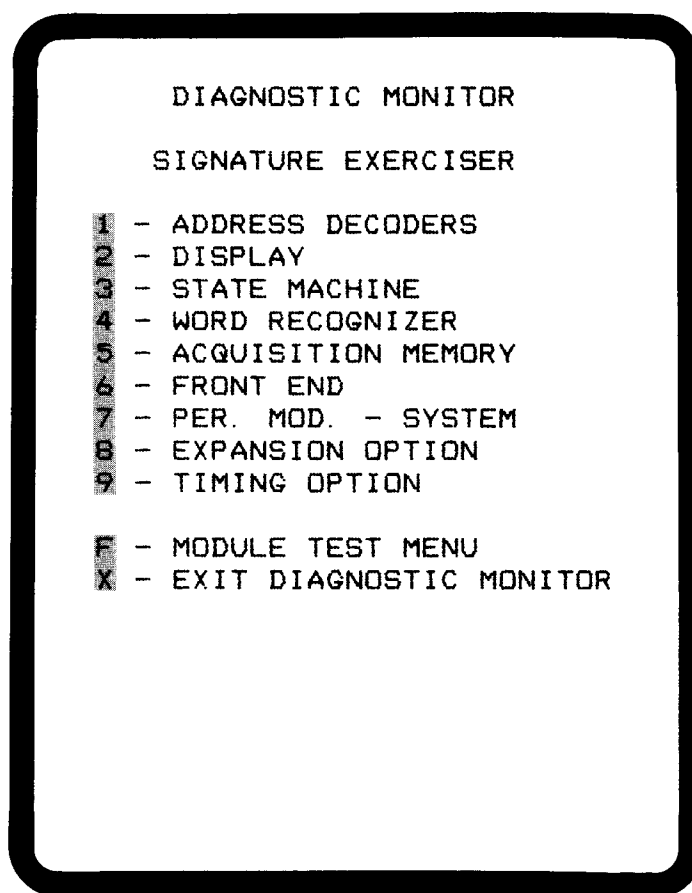
2918-92

Figure D-7. Module test result display when "LOOPING ENABLED" on sub-test 4.

SIGNATURE EXERCISER MODE

In Signature Exerciser mode, the 7D02 diagnostic program executes the user-specified test. The Signature Exerciser menu is shown in Figure D-8.

Each of the Signature Exerciser mode selections generates a series of signals which stimulate the 7D02's circuitry, thus allowing the user to check the signals found in the instrument (by using a signature analyzer test instrument) against a list of predetermined signals.



2918-93

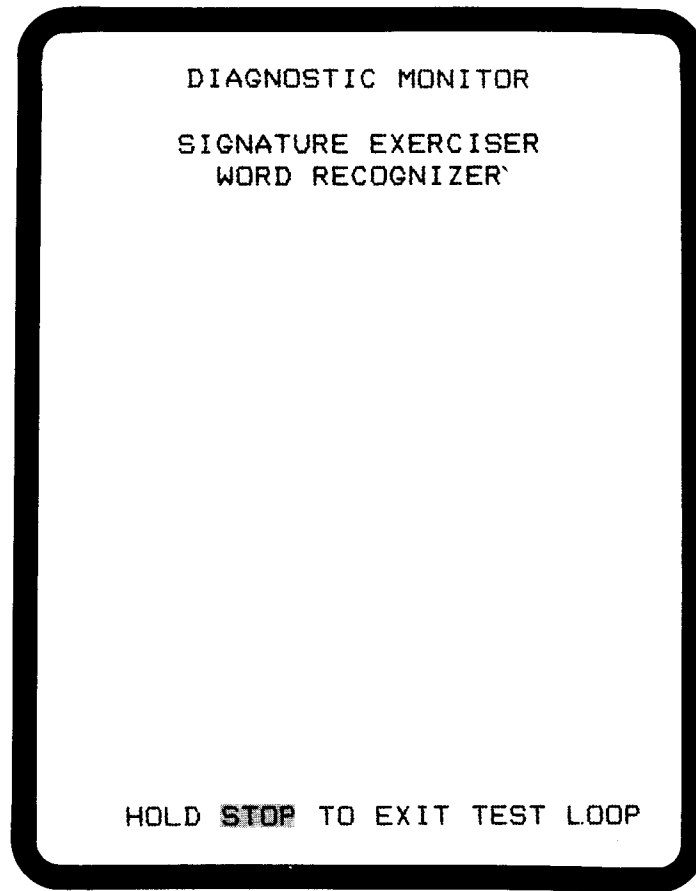
Figure D-8. Signature exerciser menu.

Appendix D—7D02 Operators

Selecting a test (1 thru 9) from the Signature Exerciser menu automatically starts that specific exerciser which runs until the **X** key is pressed. While the selected signature exerciser is running, a display similar to Figure D-9 is displayed on the screen (in this case the 4 key was pressed).

Tests 8 and/or 8 and 9 only appear if one or both of the 7D02 options (Expansion Option and Timing Option) are installed in the instrument. If there is no Expansion Option installed, but there is a Timing Option installed, then test number 8 will be the Timing Option.

Pressing the **F** key replaces the Signature Exerciser menu with the Module Test menu.



2918-94

Figure D-9. Typical signature exerciser display.

See the 7D02 Service manual for a discussion of signature analysis of the 7D02 and for a list of valid signatures.

OPTIONS AND SPECIFICATIONS

ACCESSORIES AND OPTIONS

STANDARD ACCESSORIES

1. Users Manual—070-2918-00
2. Pocket Reference—070-2915-00
3. Service Manual—070-2919-00

OPTIONAL ACCESSORIES

1. PM-101 General Purpose Personality Module
2. PM-102 6800 Personality Module
3. PM-103 6802 Personality Module
4. PM-104 8085 Personality Module
5. PM-105 Z80 Personality Module
6. 067-0939-00 Service Support Kit
7. 016-0669-00 Kit to permanently secure 7D02 in 7603

OPTIONS

1. Option 01—7D02 Timing Option with P6451 Probe
2. Option 03—Adds Expanded Memory
3. 7D02 F01—Field Installed Timing Option
4. 7D02 F03—Field Installed Expanded Memory

Table E1
ELECTRICAL CHARACTERISTICS

Characteristics	Description
State Table Display	
Type:	Raster Scan—24 lines x 32 characters per line
Number of Channels Displayed	(Basic 7D02) 28 Data 8 Address 16 Control 4
Max Number Acquisition Memory Locations Displayed	19
Radices Available	
Data	Mnemonic disassembly for each supported microprocessor. ASCII HEX BINARY OCTAL

Table E1 (cont)

Characteristics	Description
Address	HEX BINARY OCTAL ASCII
Control	MNEMONIC DISASSEMBLY BINARY
Signal Inputs	Signal inputs for the 7D02 are obtained through optional Personality Modules. These Personality Modules, along with Option 03, determine the number of channels that will be input to the 7D02.
Basic 7D02	Data 8 Address 16 Control 10 Ext. Trigger 1
Input Impedance	Determined by the Personality Module (PM) used.
Setup/Hold Time	Depends on PM used.
Synchronous Clock	
Setup/Hold Time	Determined by PM used.
Clock Qualifiers	Maximum number = 6 7D02 is capable of shifting or dividing qualified clocks by up to four positions or times respectively.
Memory Size	
Acquisition Memory	(Basic 7D02) = 28 x 256
Storage Memory	(Basic 7D02) = 28 x 256
Counters	
2 Universal Counters	
Time Mode:	
Resolution	1 ms or 1 μ s
Maximum count	65,534
Event Mode:	
Maximum count	65,534
Control Mode	
Time Mode:	
Resolution	1 ms or 1 μ s
Minimum interval generated	2
Maximum interval generated	65,534
Events Mode	
Minimum interval generated	2
Maximum interval generated	65,534

Table E1 (cont)

Characteristics	Description												
Word Recognizers													
Number of Word Recognizers	4												
Number of Channels	<table> <tr> <td>(Basic 7D02)</td><td>32</td></tr> <tr> <td>Data</td><td>8</td></tr> <tr> <td>Address</td><td>16</td></tr> <tr> <td>Control</td><td>6</td></tr> <tr> <td>Ext. Trigger</td><td>1</td></tr> <tr> <td>Timing Option Link</td><td>1 (if Timing Option installed)</td></tr> </table>	(Basic 7D02)	32	Data	8	Address	16	Control	6	Ext. Trigger	1	Timing Option Link	1 (if Timing Option installed)
(Basic 7D02)	32												
Data	8												
Address	16												
Control	6												
Ext. Trigger	1												
Timing Option Link	1 (if Timing Option installed)												
Triggering	The 7D02 can be triggered from any of the Word Recognizers or from either of the Counters, in the Control mode. This gives the 7D02 the ability to track and trigger on very complex program flows.												
Trigger Position	<p>Number of qualified clocks displayed after the trigger point:</p> <table> <tr> <td>Trigger Before Data</td><td>240</td></tr> <tr> <td>Trigger Centered</td><td>128</td></tr> <tr> <td>Trigger After Data</td><td>16</td></tr> <tr> <td>Zero Delay</td><td>0</td></tr> </table>	Trigger Before Data	240	Trigger Centered	128	Trigger After Data	16	Zero Delay	0				
Trigger Before Data	240												
Trigger Centered	128												
Trigger After Data	16												
Zero Delay	0												
Data Qualification	Complex data qualification is provided through the use of the Word Recognizers and Counters.												
Option 01—Timing Option	The Timing Option provides the 7D02 user with 8 additional channels of asynchronous data acquisition at sample rates of up to 50 MHz.												
Signal Inputs													
Number of Channels	8 (using a P6451 Data Probe)												
Clock													
Asynchronous													
Sample Rates	20 ns to 5 ms in a 1, 2, 5 sequence												
Synchronous Sample													
Maximum Frequency	10 MHz obtained from system under test via the Personality Module												
Memory Size													
Acquisition Memory	8 x 255												
Glitch Memory	8 x 255												
Word Recognizers													
One Data Word Recognizer	8 channels (ANDed together)												
One Glitch Word Recognizer	8 channels (ORed together but ANDed with the Data Word Recognizer)												
External Trigger In	Front panel bnc TTL compatible												
Asynchronous Filter	Programmable 0 to 300 ns in 20 ns increments												

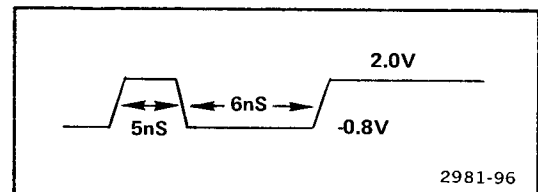
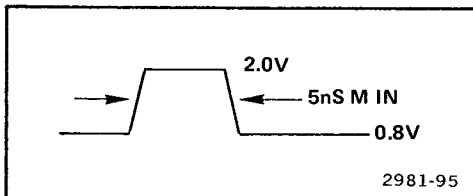
Table E1 (cont)

Characteristics	Description
Triggering The Timing Option can be triggered from any or all of the following sources:	1. Timing Option Data Word Recognizer 2. Timing Option Glitch Word Recognizer 3. Main Section (7D02) Word Recognizer 4. External Trigger In
Digital Delay	
Maximum Delay	65,534 sample clocks
Timing Diagram Display	
Number of Channels	8
Window Size	124 words in X 1 mode or 31 words in X 4 mode. Data channels can be relocated by the user.
Numeric Formats	HEX OCTAL BINARY ASCII
Glitch Display	Glitches are displayed by an ↓ above the line where the glitch occurred.
State Table Display	
Max Number of Words Displayed	19
Numeric Formats	HEX OCTAL BINARY ASCII Glitch displayed as a * in the table beside DATA
Number of words scrolled	255
Miscellaneous	The Timing Option comes standard with a P6451 Data Acquisition Probe
Option 03—Expansion	The Expansion Option provides the 7D02 with the ability to support 16-bit microprocessors.
Signal Inputs	
Adds an additional 16 bits to the 7D02	
Data	8
Address	8
Word Recognition	
Maximum Number of Channels	48 Data 16 Address 24 Control 6 Ext. Trigger 1 Timing Option Link 1 (if Timing Option installed)

Table E1 (cont)

Characteristics	Description
Display	
Maximum Number of Channels	44 Data 16 Address 24 Control 4
Memory Size	
Acquisition Memory	44 x 256
Storage Memory	44 x 256

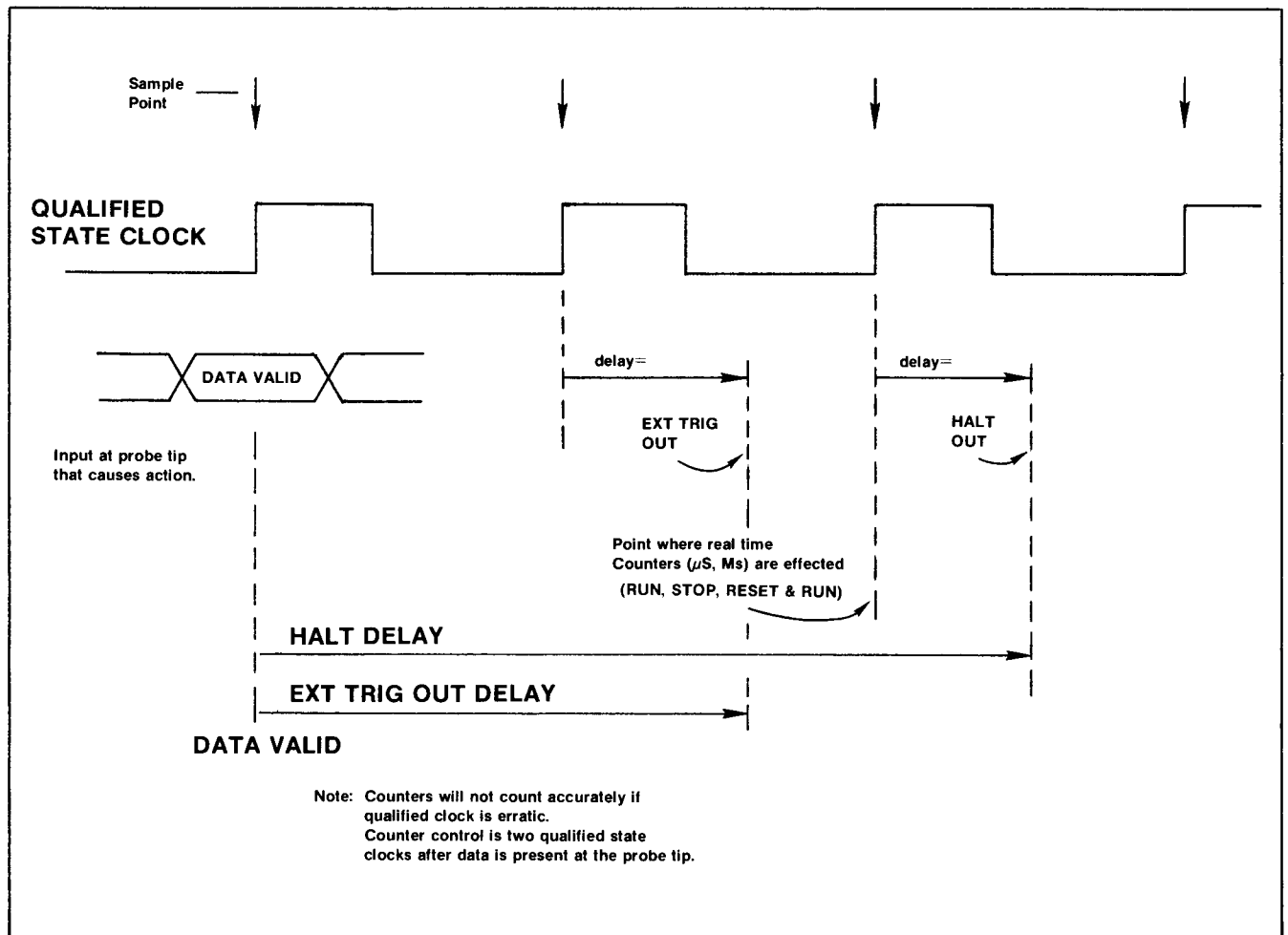
Characteristics	Performance Requirements	Supplemental Information
Timing Option		
Asynchronous Timing Mode	Sample period crystal controlled $\pm 0.01\%$	
Glitch Position	+0 -1 Bit	
Trigger Position Accuracy	± 1 Bit	Asynchronous only
System Specs with P6451		
Max Data Skew		0 ± 5 ns
Glitch Latch	Type 1 glitch	Type 2 glitch
Minimum Pulse Width	5 ns	6 ns



System Specs with P6451 and PM 101		
Data Setup Time	20 ns maximum	
Data Hold Time	2 ns maximum	
Timing Word Recognizer		
Setup	40 ns maximum	
Hold	2 ns maximum	
P6451 Probe		
Input R and C		See P6451 literature for more information. Approximately 1 M Ω Approximately 5 pF without leads Approximately 12 pF with leads

Table E1 (cont)

Characteristics	Performance Requirements	Supplemental Information
Minimum Logic Swing		500 mV +2% threshold voltage range centered on threshold voltage
Maximum Logic Swing		−15 V to threshold voltage plus 10 V
Maximum non-destruct input voltage		−40 to +40 V
Threshold voltage accuracy	±60 mV	System spec
Input Current		Approximately 8 μ A measured at 6.4 V into probe tip
External Trigger In		
System Spec using PM 101: Input Impedance	1.0 M \pm 2%	Compatible with 10X coded probe, nominal capacitance 20 pF. P6105 10X 2-meter probe recommended.
Threshold Voltage		1.4 V nominal
Hysteresis		60 mV nominal at bnc 600 mV nominal with P6105
Input Voltage		
High State		2.0 V minimum
Low State		0.8 V maximum
Main Word Recognizer		
Setup Time	10 ns @ bnc maximum	P6105 adds 9 ns to setup time; subtracts 9 ns from hold time
Hold Time	18 ns @ bnc maximum	
Timing Word Recognizer		
Synchronous Mode		
Setup Time	36 ns at bnc maximum	
Hold Time	9 ns at bnc maximum	
Asynchronous Mode		
Minimum Pulse Width	25 ns	
Trigger Output		TTL level signal that is 1 qualified state clock pulse approximately 86 ns after the timing trigger event is clocked into the 7D02. If timing option is not installed, the main trigger command initiates the trigger out. See Fig. 1; 50 Ω series-terminated driver.



2981-97

Figure E-1. Specification for counters and delays.

Table E1 (cont)

Characteristics	Performance Requirements	Supplemental Information
Low State Voltage		0.5 V into $R_L = 2k$
High State Voltage		2.4 V into $R_L = 2k$ Output can drive one STTL load.
System Specifications with PM 101		Measured per manual performance check procedure
Clock Qualifier Data Channels C4—C9	Setup Time = 55 ns maximum Hold Time = 0.0 ns maximum	Typical setup time = 35 ns
Data/Address/C0—C5	Setup Time = 45 ns maximum Hold Time = 0.0 ns maximum	Typical setup time = 25 ns
When C6—C9 are used to generate /ESYNC and /WAIT	Setup Time = 55 ns maximum Hold Time = 0.0 ns maximum	Typical setup time = 35 ns
Clock Period Minimum	100 ns	
Clock Pulse Width, Minimum	25 ns high, 25 ns low	
Minimum Interval Between Qualified Clocks	100 ns	
Halt Delay at PM 101		Schottky TTL level signal that is approximately 58 ns after the 7D02 stops acquisition. In zero delay trigger mode, the 7D02 stops acquisition 2 qualified state clocks after the main trigger event is clocked into the 7D02.
Counters		
Counting Mode Time ms or μs Clock Accuracy	0.01%	Error bound ($\pm 0.01\%$ ± 1 count) X (number of Start/Stop cycles)
Control Mode Time ms or μs Clock Accuracy	0.01%	Error bound ($-0 + 1$ count $\pm 0.01\%$ of value $-0 + 2 \mu s$) X number of Start/Stop cycles. Note: If 7D02 is stopped prior to terminal count, add one count to error bound.

Table E2
ENVIRONMENTAL CHARACTERISTICS

Characteristics	Performance Requirements	Supplemental Information
Environmental Specification		
Non-Operating Temperature		–55° C to 75° C
Operating Temperature		0° C to 50° C
Relative Humidity		90 to 95%
Non-Operating Altitude		15 km (50,000 feet)
Operating Altitude		4.5 km (15,000 feet)
Vibration		10–55 Hz (SW), 75 min. total Tektronix Std. 062-2858-00, C1-3
Shock		50 g's Half-sine, 11 ms, 18 repetitions Tektronix Std. 062-2858-00, C1-3
Bench Handling		45 degrees or 4 inches All significant faces Tektronix Std. 062-2858-00, C1-3
Package Transportation Vibration (non-operating)		25.4 mm (1 inch) Approximately 270 rpm Tektronix Std. 062-2858-00, C1-3
Package Drop (non-operating)		0.914 m (3 feet), 10 drops Tektronix Std. 062-2858-00

Table E3
PHYSICAL CHARACTERISTICS

Characteristics	Performance Requirements	Supplemental Information
Mechanical Specifications		
Weight		≈8 lbs (3.6 kg)
Outside Dimensions		21 x 13 x 37 cm (8.25" wide x 5" high x 14.6" deep)
Mainframe Compatibility		Compatible with all 3- and 4-wide 7000-Series oscilloscopes Temperature-derated in rack- mounted oscilloscopes to 35° C

7D02 PERFORMANCE CHECKOUT PROCEDURE

STEP ONE

1. Install the 7D02 with any Personality Module attached. (Note: this is not required for the power-up test to be executed.)
2. Turn on the 7000-series mainframe. The following display should appear within 10—15 seconds:

POWER-UP DIAGNOSTICS COMPLETED

This indicates that the power-up verification routines have been run successfully.

3. Turn off the 7000-series mainframe. Disconnect the Personality Module from the 7D02. Turn on the 7000-series mainframe. The following display should appear on the screen within 10—15 seconds:

```

POWERUP VERIFICATION

DIAGNOSTIC ROM          PASS
PROGRAM ROM             PASS
DISPLAY                 PASS
KEYBOARD                PASS
FIRMWARE ROMS           PASS
STATE MACHINE           PASS
WORD RECOGNIZER         PASS
ACQUISITION MEMORY      PASS
FRONT END               FAIL 2
PER. MOD. - SYSTEM      PASS
EXPANSION OPTION        PASS
TIMING OPTION           PASS

PRESS X FOR DIAGNOSTIC MONITOR
PRESS START TO BEGIN OPERATION
  
```

2918-98

Item 3 checks the firmware routines which verify the presence/absence of a Personality Module.

4. Turn off the 7000-series mainframe.

STEP TWO

1. Connect any Personality Module to the 7D02.
2. Hold down any key for at least five seconds while simultaneously turning on the 7000-series mainframe. This will cause the 7D02 power-up verification routine to fail on Test #4, subtest #1 (the keyboard verification routine). The resulting display will be:

```

POWER-UP VERIFICATION

DIAGNOSTIC ROM          PASS
PROGRAM ROM             PASS
DISPLAY                PASS
KEYBOARD                OF620-3 FAIL 1
FIRMWARE ROMS          PASS
STATE MACHINE          PASS
WORD RECOGNIZER        PASS
ACQUISITION MEMORY     PASS
FRONT END              PASS
PROBE                  PASS
EXPANSION OPTION       PASS
TIMING OPTION          PASS

PRESS X FOR DIAGNOSTIC MENU
PRESS START TO BEGIN OPERATION

```

2918-99

3. Press the **X** key to display the diagnostic menu. (If you press the **START** key, the 7D02 leaves Diagnostic mode and enters its normal operating mode . . . if it can.)
4. Press the **3** key to select the keyboard diagnostics program.

The resulting display will be:

```

DIAGNOSTIC MONITOR

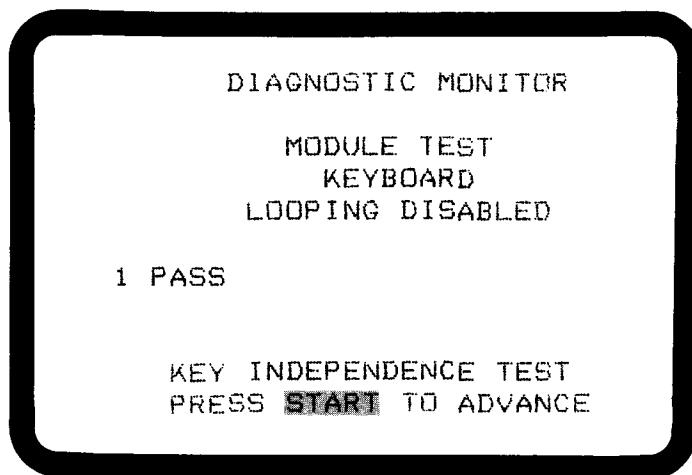
MODULE TEST
KEYBOARD
LOOPING DISABLED

E ----- ENABLE LOOPING
X ----- RETURN TO MENU
START - RUN TEST

```

2918-100

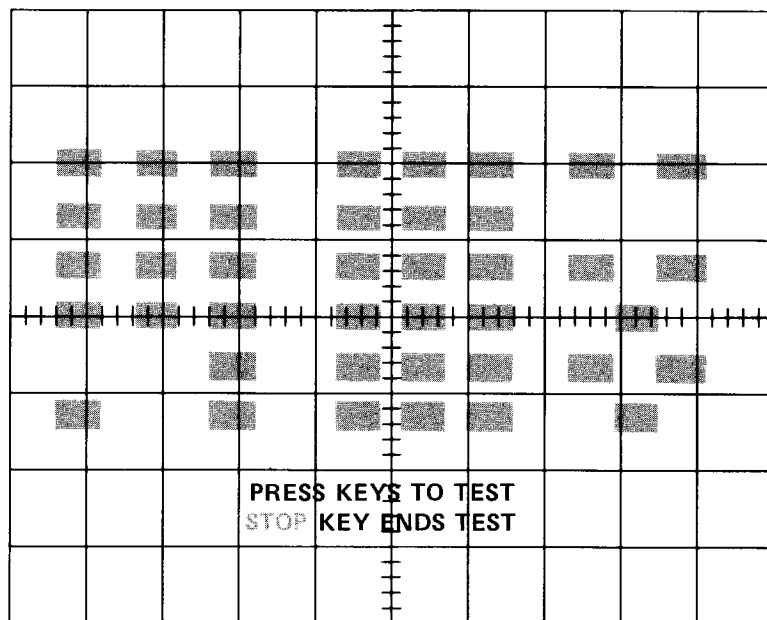
5. Press the **START** key. The resulting display will be:



6. Press the **START** key.

2918-101

The second test in this program checks for the presence of and independence of all keys in the 7D02 front panel. The test displays a pattern of inverse video blocks identical in arrangement to the keys on the front panel. See Figure F-1.



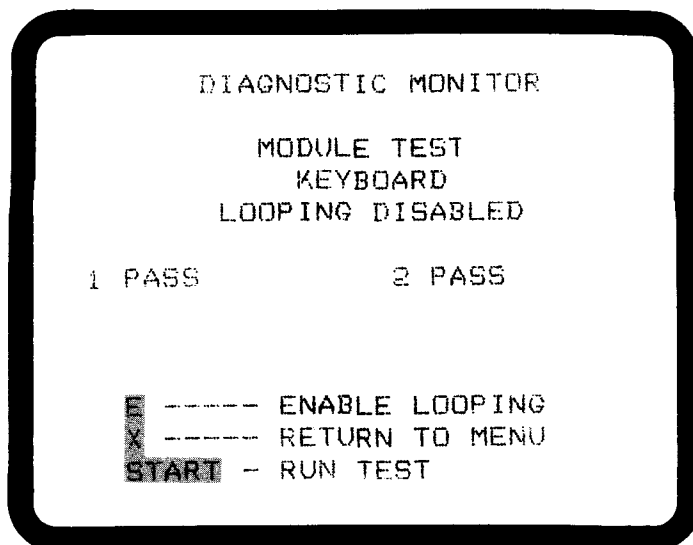
2918-102

Figure F-1. Screen display for the key-independence test.

Appendix F—7D02 Operators

Press each key except the **STOP** key. As you press a key, the inverse video block representing that key, and only that key, will disappear. This tells you that the key is working properly. If more than one key disappears, there is a problem with the keyboard that must be repaired before correct operation of the 7D02 can be expected.

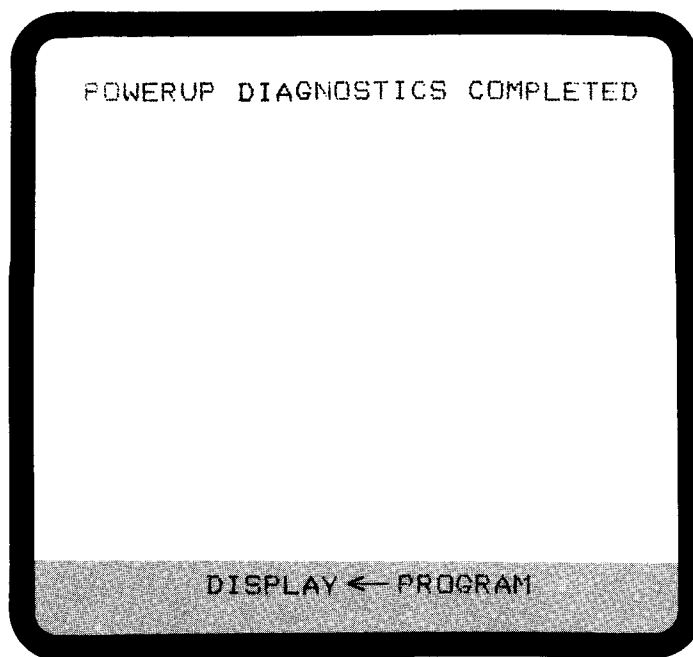
When you have pressed all of the keys except the **STOP** key and each has tested out satisfactorily, then press the **STOP** key. The resulting display is:



2918-103

The 7D02 has finished the keyboard verification program including testing the **STOP** key for correct functioning.

7. Press the **X** key twice. The screen should now have the following display:



2918-104

8. This completes the 7D02 performance checkout procedure.

GLOSSARY OF TERMS

ASYNCHRONOUS

The temporal relation of two or more signals that have unrelated time bases. The 7D02 Timing Option can sample data asynchronously using a sampling time base supplied by the 7D02 that is unrelated to the time base of the system under test.

ASYNCHRONOUS FILTER

A method of rejecting transient events that fail to meet a minimum time criteria. The 7D02 Timing Option uses an Asynchronous Filter to validate word recognition.

CLOCK

A train of periodic pulses that serves as a timing reference.

CLOCK EDGE

The positive-going (rising) portion of a clock pulse or the negative-going (falling) portion of the clock pulse. A clock pulse transition.

CLOCK QUALIFICATION

A means of selecting specific clock pulses from an external time base clock (from the system under test). The 7D02 uses any combination of clock edge, signal level, or preselected delay following a synchronizing signal to select or qualify these external time base clock pulses. Some personality modules provide additional qualification criteria, as well. The selected pulses are used as the 7D02 State clock.

COMMAND

Some user-specifiable action that the 7D02 performs during data acquisition, such as triggering, qualification, counter commands, go to, etc.

CYCLE

The time between two consecutive clock pulses.

DATA QUALIFICATION

The application of user-defined criteria to select sampled data for storage in acquisition memory. Any event recognizable by the 7D02 can be used to qualify data.

DATA WINDOW

A defined subset of the total data available for acquisition or display. Three commonly defined data windows are data acquired according to the conditions specified by the user, data stored in the acquisition memory, and data from the acquisition memory available for viewing on the screen at one time.

DON'T CARE

An indication that a user is not concerned with the condition on a channel. On the 7D02, the X key is used to indicate a Don't Care.

EDGE SENSITIVE

Responsive to the positive-going or negative-going edge of a signal. The word recognizer in the 7D02 Timing Option is edge sensitive in Asynchronous mode. (See also LEVEL SENSITIVE.)

EXPANSION OPTION

An option to the 7D02 that expands its memory width from 28 to 44 bits and its word recognition width from 31 to 47 bits. This expanded capability makes the 7D02 more convenient for use in 16-bit microprocessor systems or for bit-slice systems of long word length.

EVENT

A condition specified by the user that can be detected by the 7D02 during data acquisition and evaluated as true or false; i.e., successful word recognition, a counter reaching a selected value, or a logical combination or complement of these.

GLITCH

In asynchronous operation of the 7D02 Timing Option, a glitch is defined to be more than one transition of an input channel occurring between two consecutive sampling clock pulses.

GLITCH RECOGNITION

In the 7D02 Timing Option, this is the detection of glitches on one or more user-specified channels of input.

HOLD TIME

The length of time for which data must be present and stable after a clock edge to assure that the data will be acquired. (See also SETUP TIME.)

LEVEL SENSITIVE

Responsive to the logical value of a signal at a specified point in time such as a clock edge. (See also EDGE SENSITIVE.)

LOGIC POLARITY

A convention for assigning logic values to voltage levels. In positive logic, an input high level = 1, an input low level = 0. In negative logic, an input high level = 0, and input low level = 1.

MEMORY DEPTH

The maximum number of data words that can be stored in memory (up to 256 in the 7D02).

MEMORY WIDTH

The maximum number of input channels that can be stored in memory (up to 44 in Main memory and 8 in Timing Option memory for the 7D02). This number corresponds to the number of bits in a word of stored data.

MENU

A 7D02 display that provides the operator with an expanded list of valid parameters for the field in which the cursor is positioned. An arbitrary number is assigned to each parameter.

PERSONALITY MODULE

An accessory interface for the 7D02 that optimizes data acquisition and display for a particular microprocessor type. Personality modules are available for many popular microprocessors, each providing convenient processor-specific instruction disassembly and data formatting.

POWER-UP VERIFICATION

That part of the system diagnostic testing that the 7D02 performs automatically at power up.

PROGRAM

In the 7D02 a program defines the process for locating and acquiring the data of interest. It is created by a series of keystrokes that direct the 7D02 to monitor events and execute related commands during data acquisition. A program consists of 1 to 4 tests plus an optional data qualification block.

SET-UP TIME

The length of time for which data must be present and stable prior to a clock pulse to assure reliable acquisition. (See also HOLD TIME.)

SIGNATURE ANALYSIS

A troubleshooting method in which the circuit under test is subjected to a known stimulus, and an instrument (called a Signature Analyzer) is used to acquire the resulting data from various nodes in the circuit. The Signature Analyzer acquires, compresses, and displays the node data as a simple numeric value (the signature), which is compared to an expected value for that node.

SIGNATURE EXERCISE(R)

A program available in the 7D02 Diagnostic Mode that generates test patterns for signature analysis.

STATE CLOCK

A clock derived from the s.u.t. raw clock by clock qualification. The State clock drives the 7D02 and the synchronous mode Timing Option during data acquisition; and it determines the timing for data sampling, evaluation of events, and execution of commands.

STATE TABLE

A representation in tabular, alphanumeric form of the logic states of the input data channels at each successive sampling clock pulse. Data acquired by the 7D02 itself is displayed in "mnemonic" or "absolute" formats. Data acquired with the Timing Option can be displayed in state table format if desired.

SYNCHRONOUS

The temporal relation of two or more signals with a common time base. The standard 7D02 samples data synchronously, as does the Timing Option when in synchronous mode. The time base used is always the State clock, as derived from the s.u.t. clock.

SYSTEM UNDER TEST (S.U.T.)

The digital system that the 7D02 is monitoring.

THRESHOLD VOLTAGE

A voltage level against which data is compared to determine its logic level. In the 7D02 a simple Personality Module-dependent threshold distinguishes logic 1's from logic 0's. An adjustable threshold is available for the Timing Option.

TIMING DIAGRAM

Recorded data states and timing relationships displayed as pseudo-waveforms. Data acquired with the 7D02 Timing Option can be displayed in Timing Diagram format.

TIMING OPTION

An option to the 7D02 that allows it to perform asynchronous data sampling, or expands the synchronous sampling capability of the 7D02 by eight channels. Additional menus, menu selections, and data displays are available when the Timing Option is installed.

TRIGGER

A way to define the point of interest in the sample data. In the 7D02, the trigger determines the data window for storage by setting off the mechanism (a delay counter) for terminating acquisition.

WORD

-) A group of bits that are treated as a unit. The standard 7D02 recognizes a 31-bit word, and stores a 28-bit word; the Expansion Option increases these word sizes to 47 and 44 bits, respectively. The Timing Option adds eight bits of recognition and eight bits of storage.

WORD RECOGNITION

An event that occurs when data on channels being monitored from the s.u.t. match a word defined by the user. The Word Recognizer performs word recognition and outputs the results as a simple true or false event.

7D02 ASCII DISPLAY CHARACTERS

INTRODUCTION

Table H1 presents the text displayed for each 8-bit "digit" when ASCII is selected as the radix for a data field. Note that 7-bit ASCII is used. The high-order bit of each 8 bits of data is treated as a parity bit and is stripped off before the conversion to ASCII. Thus, 41 and C1 will both display as "A".

Table H1
7D02 ASCII DISPLAY CHARACTERS

HEX	ASCII	HEX	ASCII	HEX	ASCII	HEX	ASCII
00	NUL	20		40	@	60	<@>
01	SOH	21	!	41	A	61	<A>
02	STX	22	"	42	B	62	
03	ETX	23	#	43	C	63	<C>
04	EOT	24	\$	44	D	64	<D>
05	ENQ	25	%	45	E	65	<E>
06	ACK	26	&	46	F	66	<F>
07	BEL	27	'	47	G	67	<G>
08	BS	28	(48	H	68	<H>
09	HT	29)	49	I	69	<I>
0A	LF	2A	*	4A	J	6A	<J>
0B	VT	2B	+	4B	K	6B	<K>
0C	FF	2C	,	4C	L	6C	<L>
0D	CR	2D	-	4D	M	6D	<M>
0E	SO	2E	.	4E	N	6E	<N>
0F	SI	2F	/	4F	O	6F	<O>
10	DLE	30	0	50	P	70	<P>
11	DC1	31	1	51	Q	71	<Q>
12	DC2	32	2	52	R	72	<R>
13	DC3	33	3	53	S	73	<S>
14	DC4	34	4	54	T	74	<T>
15	NAK	35	5	55	U	75	<U>
16	SYN	36	6	56	V	76	<V>
17	ETB	37	7	57	W	77	<W>
18	CAN	38	8	58	X	78	<X>
19	EM	39	9	59	Y	79	<Y>
1A	SUB	3A	:	5A	Z	7A	<Z>
1B	ESC	3B	;	5B	[7B	<[>
1C	FS	3C	<	5C	\	6C	<\>
1D	GS	3D	=	5D]	7D	<]>
1E	RS	3E	>	5E	↑	7E	<↑>
1F	US	3F	?	5F	←	7F	DEL

NOTE

Lower case characters are displayed as capital letters surrounded by angle brackets. For example, an "a" is displayed as "[A]".

7D02 FRONT PANEL

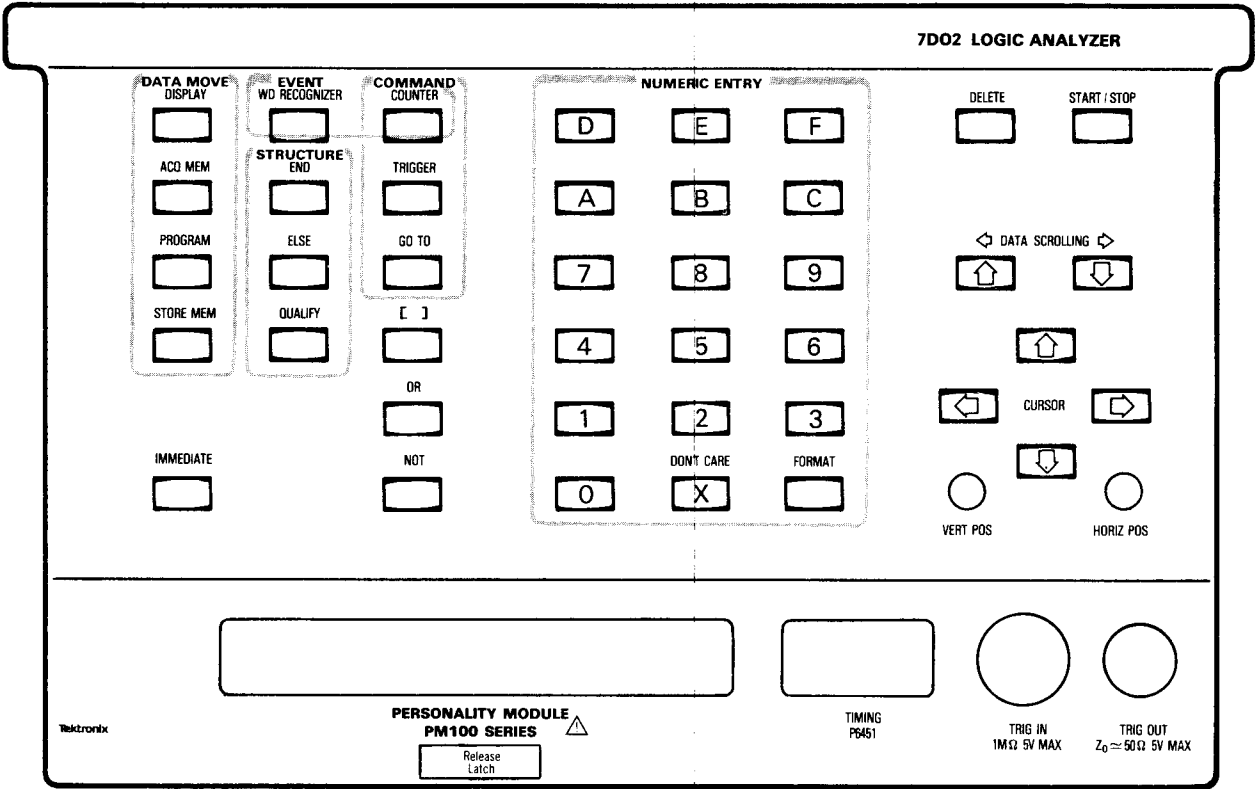


Fig. I-1. 7D02 Front Panel.

Date: 4-22-81 Change Reference: C1/481

Product: 7D02 LOGIC ANALYZER OPERATORS Manual Part No.: 070-2918-00

DESCRIPTION

TEXT CHANGE

SECTION 1 7D02 LEARNING GUIDE

p. 1-1 **INSTALLING AND REMOVING THE 7D02**

CHANGE the first sentence to read as follows:

The 7D02 Logic Analyzer is designed to occupy a three-compartment segment of a TEKTRONIX non-storage 7000-Series Oscilloscope (the 7104 is NOT recommended).

Date: 5-29-81

Change Reference: 02/581

Product: 7D02 LOGIC ANALYZER

Manual Part No.: 070-2918-00

DESCRIPTION

TEXT CHANGES

SECTION 1 LEARNING GUIDE

page 1-22 1st paragraph under NOTE

CHANGE TO:

WD RECOGNIZER key is pressed. The 7D02 looks for the third data word (FF) on the s.u.t. data bus only after the specified number of occurrences of data word #2 (F6) have been counted.

page 1-32 1st paragraph under Data Explanations

CHANGE TO:

This example points out one of the basic operating principles of 7D02 counters ...counters continue to count until they either reach their specified value (500 for count #1) or until data acquisition is completed (741 for counter #2).

page 1-38 4th paragraph down

CHANGE TO:

TEST 2 is written to say "store every cycle after trigger executed".

page 1-40 display upper right corner

CHANGE TO:

TRIG LOC = 100

SECTION 3 MODES OF OPERATION

page 3-9 3rd paragraph down last sentence

CHANGE TO:

The DATA SCROLLING keys move the data cursor (and the window) in the direction of the arrow on the key; e.g., the DATA SCROLLING ↑ key moves the data cursor toward the beginning of the memory.

DESCRIPTION

page 3-10 under DISPLAY CONTENTS last paragraph last sentence

CHANGE TO:

Some commands cause the word "EXECUTE" to appear in the Immediate area; these commands are not actually executed until the cursor is moved onto the EXECUTE field with either the CURSOR ↓ key or the CURSOR → key.

page 3-11 under GOTO COMMANDS 2nd paragraph 1st sentence.

CHANGE TO:

Enter a GOTO command by pressing the GOTO key after the IMMEDIATE key.

SECTION 5 KEY ENCYCLOPEDIA

page 5-10 under GENERAL DESCRIPTION 5th paragraph

CHANGE TO:

The ← key moves the cursor to the next legal position to the left. If there are no more legal positions on the same line, it works like the ↑ key, except that it leaves the cursor at the right most position on a line.

page 5-21 keystroke sequences in the middle of the page

CHANGE TO:

DISPLAY	ACQ MEM
DISPLAY	PROGRAM
DISPLAY	STORE MEM
PROGRAM	STORE MEM
STORE MEM	PROGRAM
STORE MEM	ACQ MEM
GOTO	test#

page 5-23 last paragraph last sentence

CHANGE TO:

The user must supply the closing bracket at the end of the compound event by pressing the [] key.

page 5-30 under GENERAL DESCRIPTION the keystroke sequences

CHANGE TO:

IMMEDIATE	DISPLAY	STORE MEM
IMMEDIATE	STORE MEM	ACQ MEM
IMMEDIATE	PROGRAM	STORE MEM
IMMEDIATE	STORE MEM	PROGRAM

page 5-41 under GENERAL DESCRIPTION 3rd paragraph 1st sentence

CHANGE TO:

When pressed after a "THEN DO" or "ELSE DO" prompt, the [] key generates an "open bracket" on the screen , indicating the start of a compound command.

DESCRIPTION

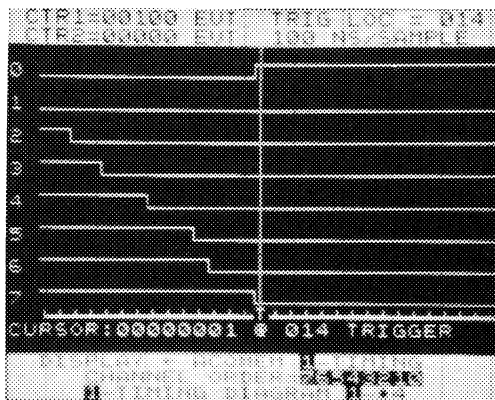
SECTION 6 TIMING OPTION

page 6-12 1st item under Note that:

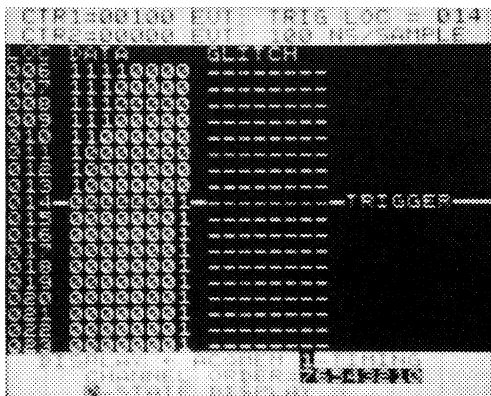
CHANGE TO:

1. Both trigger commands are executed when the Main word recognizer #1 goes TRUE; i.e., when The Timing Option word recognizer detects the 00XX00XX bit pattern (in this case it was a 00000000).

page 6-23 Figure 6-10

CHANGE AS SHOWN:

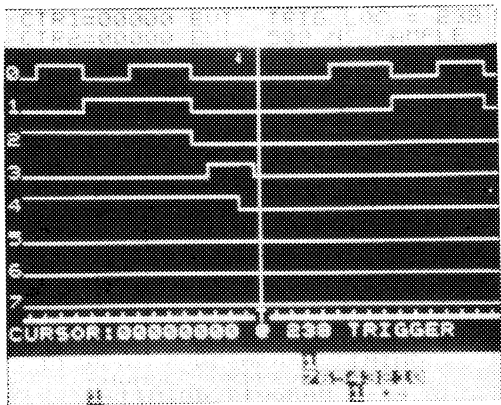
page 6-24 Figure 6-11

CHANGE AS SHOWN:

DESCRIPTION

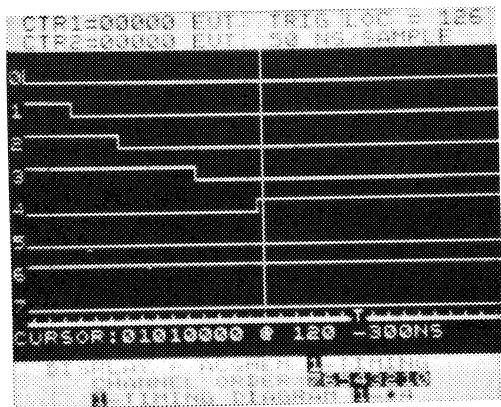
page 6-27 screen display

CHANGE AS SHOWN:



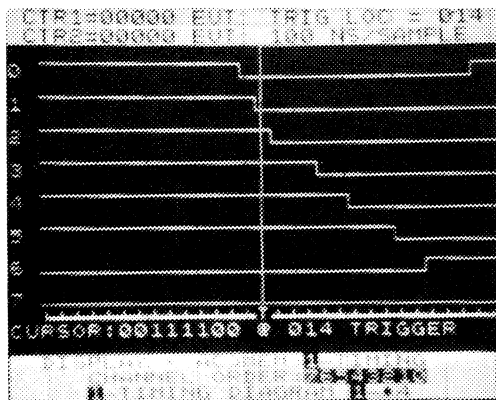
page 6-29 upper screen display

CHANGE AS SHOWN:



page 6-31 Figure 6-12

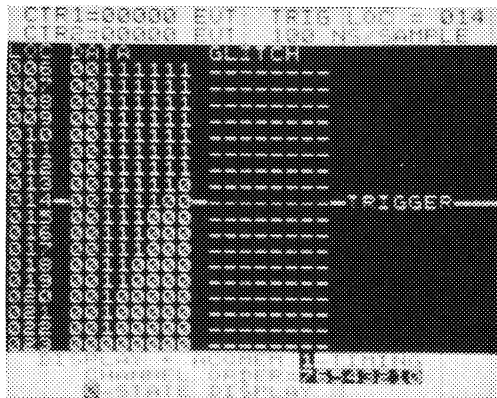
CHANGE AS SHOWN:



DESCRIPTION

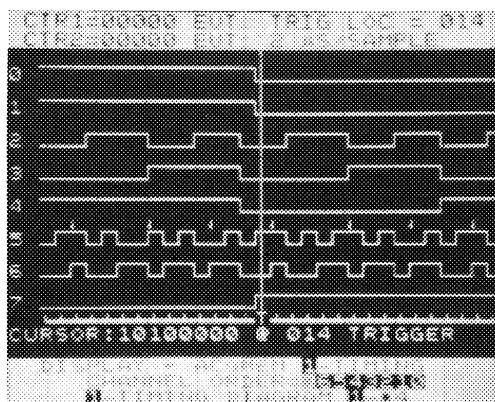
page 6-32 Figure 6-13

CHANGE AS SHOWN:



page 6-36 Figure 6-14

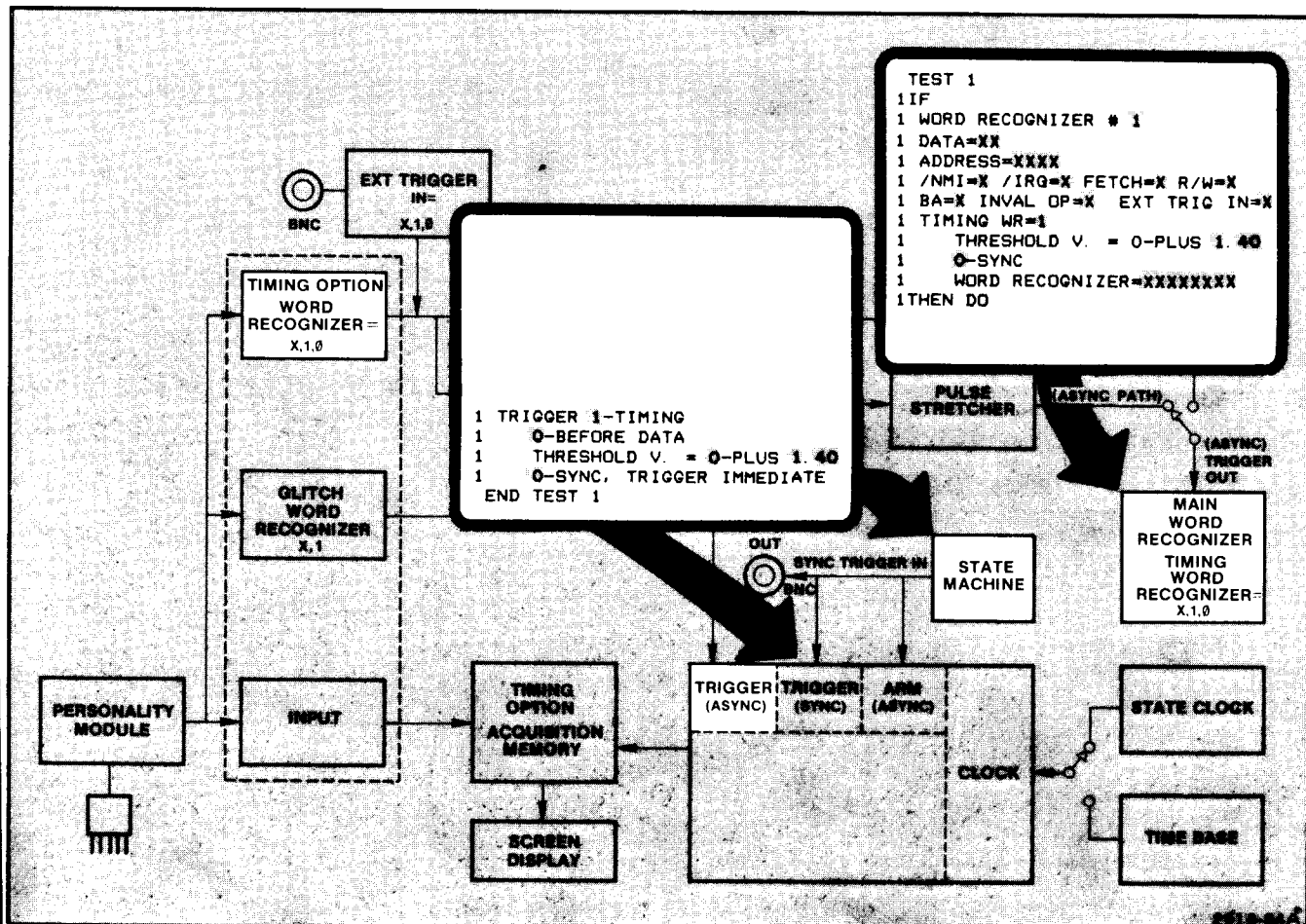
CHANGE AS SHOWN:



DESCRIPTION

page 6-46 Figure 6-21

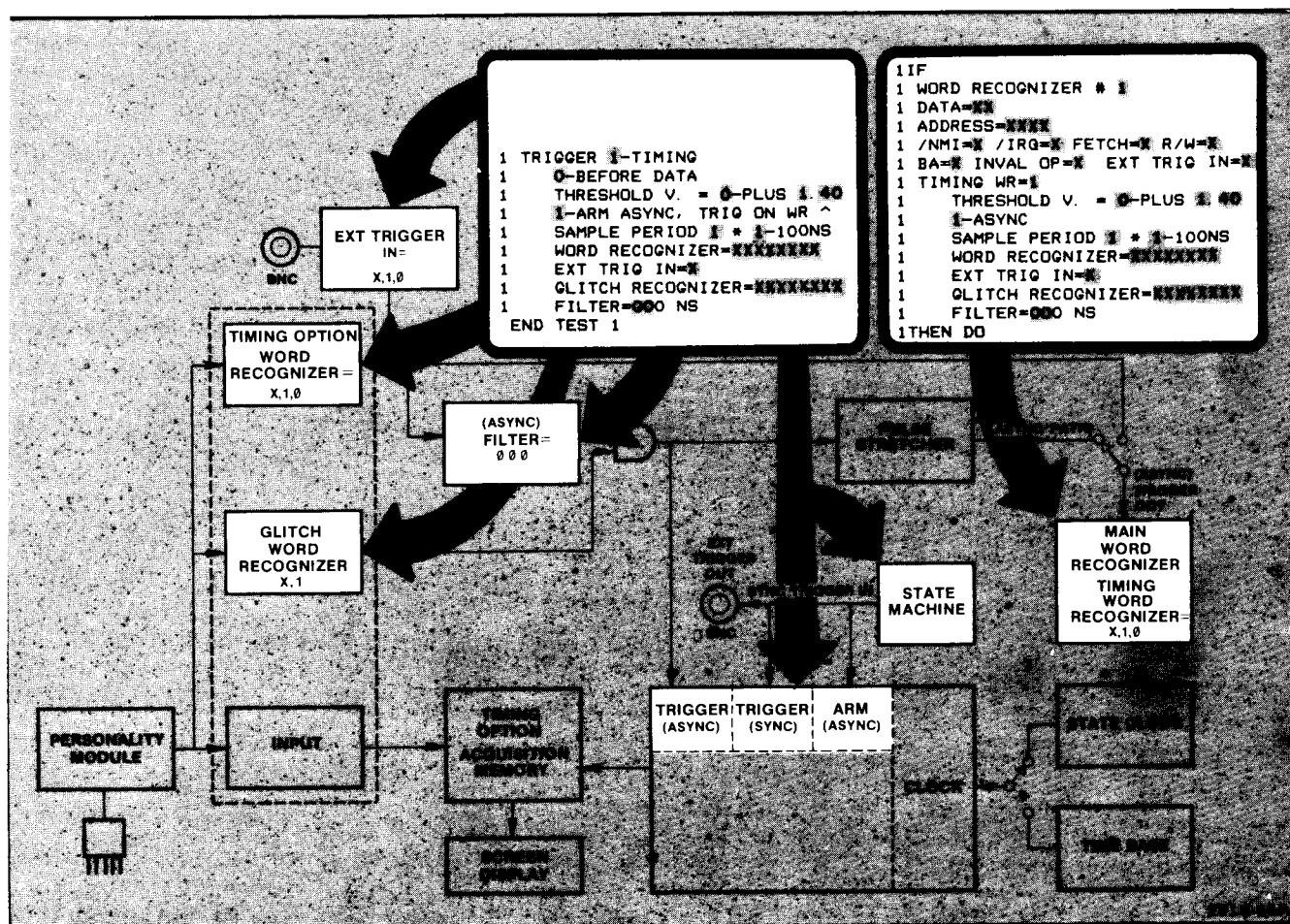
CHANGE AS SHOWN:



DESCRIPTION

page 6-47 Figure 6-22

CHANGE AS SHOWN:



DESCRIPTION

SECTION 7 7D02 RULES OF OPERATION AND SPECIAL NOTES

page 7-1 under REQUIREMENTS FOR STOPPING DATA ACQUISITION 1st row top of 2nd column

CHANGE TO:

User must press **STOP** key to end acquisition.

APPENDIX A STATUS AREA MESSAGES

page A2 top of page right hand side

CHANGE TO:

An attempt was made to enter an **X** key into a field that only accepts numeric keys. Try 0-F.

APPENDIX D SELF-TEST PROCEDURES AND ERROR MESSAGES

page D-5 1st paragraph 1st sentence

CHANGE TO:

If an individual test is selected, a display similar to Figure D-4 is displayed on the screen (in this case the **6** key was pressed).