ADVANTEST®

ADVANTEST CORPORATION

INSTRUCTION MANUAL

TR4751

Logic Analysis System

MANUAL NUMBER OEC00 906

Before reselling to other corporations or re-exporting to other countries, you are required to obtain permission from both the Japanese Government under its Export Control Act and the U.S. Government under its Export Control Law.

Printed in Japan

RECORD OF REVISIONS

Rev. No.	Date	Remarks
OEA	May 15/86	First revision
OEB	Nov 11/86	

Rev. No.	Date	Remarks

LIST OF EFFECTIVE PAGES

#	Record of Revisions	R - 1	Nov 11/86	3 - 19	May 15/86
#	List of			4 - 1	May 15/86
	Effective Pages	P - 1	Nov 11/86	4 - 2	May 15/86
		P - 2	May 15/86	4 - 3	May 15/86
		P - 3	Nov 11/86	4 - 4	May 15/86
	Table of Contents	C - 1	May 15/86	4 - 5	May 15/86
		C - 2	May 15/86	4 - 6	May 15/86
		C - 3	May 15/86	4 - 7	May 15/86
	1 - 1		May 15/86	4 - 8	May 15/86
	1 - 2		May 15/86	# 4 - 9	Nov 11/86
	1 - 3		May 15/86	# 4 - 10	Nov 11/86
	1 - 4		May 15/86	# 4 - 11	Nov 11/86
	1 - 5		May 15/86	# 4 - 12	Nov 11/86
	1 - 6		May 15/86	# 4 - 13	Nov 11/86
	1 - 7		May 15/86	# 4 - 14	Nov 11/86
	1 - 8		May 15/86	# 4 - 15	Nov 11/86
	2 - 1		May 15/86	4 - 16	May 15/86
	2 - 2		May 15/86	4 - 17	May 15/86
	2 - 3		May 15/86	4 - 18	May 15/86
	2 - 4		May 15/86	4 - 19	May 15/86
	2 - 5		May 15/86	4 - 20	May 15/86
	2 - 6		May 15/86	4 - 21	May 15/86
	2 - 7		May 15/86	4 - 22	May 15/86
	2 - 8		May 15/86	4 - 23	May 15/86
	3 - 1		May 15/86	4 - 24	May 15/86
	3 - 2		May 15/86	4 - 25	May 15/86
	3 - 3		May 15/86	4 - 26	May 15/86
	3 - 4		May 15/86	4 - 27	May 15/86
	3 - 5		May 15/86	4 - 28	May 15/86
	3 - 6		May 15/86	4 - 29	May 15/86
	3 - 7		May 15/86	4 - 30	May 15/86
	3 - 8		May 15/86	4 - 31	May 15/86
	3 - 9		May 15/86	4 - 32	May 15/86
	3 - 10		May 15/86	4 - 33	May 15/86
	3 - 11		May 15/86	4 - 34	May 15/86
	3 - 12		May 15/86	5 - 1	May 15/86
	3 - 13		May 15/86	5 - 2	May 15/86
	3 - 14		May 15/86	5 - 3	May 15/86
	3 - 15		May 15/86	5 - 4	May 15/86
	3 - 16		May 15/86	5 - 5	May 15/86
	3 - 17		May 15/86	5 - 6	May 15/86
	3 - 18		May 15/86	5 - 7	May 15/86

Note: Pages with # are revised.
Pages with ## are added.
Pages with ( ) are deleted.

LIST OF EFFECTIVE PAGES (Cont'd)

5 - 8	May 15/86	7 - 34	May 15/86
5 - 9	May 15/86	7 - 35	May 15/86
5 - 10	May 15/86	7 - 36	May 15/86
5 - 11	May 15/86	7 - 37	May 15/86
5 - 12	May 15/86	7 - 38	May 15/86
6 - 1	May 15/86	7 - 39	May 15/86
6 - 2	May 15/86	7 - 40	May 15/86
6 - 3	May 15/86	7 - 41	May 15/86
6 - 4	May 15/86	7 - 42	May 15/86
7 - 1	May 15/86	7 - 43	May 15/86
7 - 2	May 15/86	7 - 44	May 15/86
7 - 3	May 15/86	7 - 45	May 15/86
7 - 4	May 15/86	7 - 46	May 15/86
7 - 5	May 15/86	7 - 47	May 15/86
7 - 6	May 15/86	7 - 48	May 15/86
7 - 7	May 15/86	7 - 49	May 15/86
7 - 8	May 15/86	7 - 50	May 15/86
7 - 9	May 15/86	7 - 51	May 15/86
7 - 10	May 15/86	7 - 52	May 15/86
7 - 11	May 15/86	7 - 53	May 15/86
7 - 12	May 15/86	7 - 54	May 15/86
7 - 13	May 15/86	7 - 55	May 15/86
7 - 14	May 15/86	7 - 56	May 15/86
7 - 15	May 15/86	7 - 57	May 15/86
7 - 16	May 15/86	7 - 58	May 15/86
7 - 17	May 15/86	7 - 59	May 15/86
7 - 18	May 15/86	7 - 60	May 15/86
7 - 19	May 15/86	7 - 61	May 15/86
7 - 20	May 15/86	7 - 62	May 15/86
7 - 21	May 15/86	7 - 63	May 15/86
7 - 22	May 15/86	7 - 64	May 15/86
7 - 23	May 15/86	7 - 65	May 15/86
7 - 24	May 15/86	7 - 66	May 15/86
7 - 25	May 15/86	7 - 67	May 15/86
7 - 26	May 15/86	7 - 68	May 15/86
7 - 27	May 15/86	7 - 69	May 15/86
7 - 28	May 15/86	7 - 70	May 15/86
7 - 29	May 15/86	7 - 71	May 15/86
7 - 30	May 15/86	7 - 72	May 15/86
7 - 31	May 15/86	7 - 73	May 15/86
7 - 32	May 15/86	7 - 74	May 15/86
7 - 33	May 15/86	7 - 75	May 15/86

Note: Pages with # are revised.
Pages with ## are added.
Pages with ( ) are deleted.

LIST OF EFFECTIVE PAGES (Cont'd)

7 - 76	May 15/86		9 - 1		May 15/86
7 - 77	May 15/86		9 - 2		May 15/86
7 - 78	May 15/86		9 - 3		May 15/86
7 - 79	May 15/86		9 - 4		May 15/86
7 - 80	May 15/86		9 - 5		May 15/86
7 - 81	May 15/86		9 - 6		May 15/86
7 - 82	May 15/86		9 - 7		May 15/86
7 - 83	May 15/86		9 - 8		May 15/86
7 - 84	May 15/86		10 - 1		May 15/86
7 - 85	May 15/86		10 - 2		May 15/86
7 - 86	May 15/86		10 - 3		May 15/86
7 - 87	May 15/86		11 - 1		May 15/86
7 - 88	May 15/86		11 - 2		May 15/86
7 - 89	May 15/86		12 - 1		May 15/86
7 - 90	May 15/86		12 - 2		May 15/86
7 - 91	May 15/86		12 - 3		May 15/86
7 - 92	May 15/86		12 - 4		May 15/86
7 - 93	May 15/86		12 - 5		May 15/86
7 - 94	May 15/86		12 - 6		May 15/86
7 - 95	May 15/86		12 - 7		May 15/86
7 - 96	May 15/86		12 - 8		May 15/86
7 - 97	May 15/86	#	List of Figures	F - 1	Nov 11/86
7 - 98	May 15/86			F - 2	May 15/86
7 - 99	May 15/86		List of Tables	T - 1	May 15/86
7 - 100	May 15/86		List of Examples	E - 1	May 15/86
7 - 101	May 15/86				
7 - 102	May 15/86				
7 - 103	May 15/86				
7 - 104	May 15/86				
7 - 105	May 15/86				
7 - 106	May 15/86				
7 - 107	May 15/86				
7 - 108	May 15/86				
7 - 109	May 15/86				
7 - 110	May 15/86				
7 - 111	May 15/86				
7 - 112	May 15/86				
7 - 113	May 15/86				
7 - 114	May 15/86				
7 - 115	May 15/86				
8 - 1	May 15/86				
8 - 2	May 15/86				

Note: Pages with # are revised.
Pages with ## are added.
Pages with ( ) are deleted.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

1. GENERAL INFORMATION

Preface:
This chapter describes how to use this manual, giving the TR4751 general description and procedures for setting up TR4751 and starting a measurement. We assume that you have already read this chapter before performing a measurement.

1.1 HOW TO USE THIS MANUAL

This manual has been edited so that even a beginner with the logic analyzer can operate the basic and applied functions of the TR4751 with little effort.

Underscored alphabetic characters, e.g. "Program ___", represent the soft keys located under the CRT display.

The skilled user of the logic analyzer and pattern generator can start a measurement after reading Chapter 3. See Chapter 4 and 5 for further information on the functions used in the examples of Chapter 3.

1.2 TR4751 GENERAL DESCRIPTION

The TR4751 Logic Analysis System is a new-type measuring unit incorporating the logic analyzer with up to 400 MHz sampling speed and the pattern generator with up to 100 MHz rate output.

The logic analyzer (acquisition section) is provided with 16-channel inputs. Each channel contains 4K bit data memory and glitch memory. In 400 MHz sampling, 4K bit memory is automatically extended to 8K bit. This enables a broad range of measuring data.

The logic analyzer permits up to 64-channel inputs by mounting an optional (low-speed acquisition module). The maximum sampling speed of the low-speed acquisition module is 50 MHz. Each channel contains 4K bit data memory.

The various trigger functions are provided to capture complex data.

The captured data is displayed in terms of timing waveforms, state lists, comparisons with the reference data, graphs, times and frequencies.

The pattern generator has 16-channel output and can output flexible stimulus patterns using two pattern modes. The TR4751 can output NRZ (non-return to zero) and RZ (return to zero) waveforms by using this pattern generator.

ADVANTEST supports a label-symbolical editor function and the sequence LSI to enable various commands. Using these commands allows the effective programming of output patterns.

The pattern generator also permits up to 64-channel outputs using an optional accessory unit (TR47501 Extended Pattern Generator).

1K bit pattern memory is provided for every 16 channels of basic configurations. 4K bits is provided for the TR47501. Beside the above functions, the TR4751 incorporates the following functions:

Not only GPIB but also RS-232C is provided as a standard interface. This allows the TR4751 to be connected to various systems. GPIB is provided with a built-in controller for automatic measurement without any external controller.

3.5-inch micro floppy drive is equipped for easy managing and setting of measurement data, and for setting parameters and programs.

Features of the TR4751 are as follows:

(1) Up to 400 MHz sampling
(2) 8K bit acquisition memory
(3) Various trigger functions
(4) Up to 100 MHz stimulus pattern generation
(5) 1K/4K bit pattern memory
(6) Various pattern commands
(7) Standard installation of GPIB controller
(8) Standard installation of 3.5-inch micro floppy disk drive

1.3 PRECAUTIONS AND PREPARATIONS BEFORE USE

1.3.1 Inspection

Upon receipt of the TR4751, check whether the TR4751 has been damaged during transport.

Then also check the quantities and ratings of the standard accessories using the following table. If the TR4751 is damaged or some accessories are missing, contact your nearest ADVANTEST representative.

Table 1-1 TR4751 Configuration List

No.	Item	Stock No.	Standard	Option		Remarks
				71	73	
1	Main unit	TR4751	1			
2	Full keyboard		1			
3	Data acquisition probe G	TR14703-01	1			
4	Data acquisition probe H	TR14703-02	1			
5	Pattern generator cable A (a pair of cables for 0 to 7 ch and 8 to F ch)	A04703-11	1			
6	Clock/strobe cable	A04703-21	1			
7	Cable conversion adapter (a set of ten cables) UM-QJ connector type UM-QJ pin socket type	A04701-95 A04701-76	2 1			Cable A for CLK/STRB and TEST
8	PG external power cable A	A04702-31	1			
9	PG control probe	TR14702-03	1			
10	Data acquisition probe A	TR14701-01			1	
11	Data acquisition probe B	TR14701-02			1	
12	Data acquisition probe C	TR14701-03		1	1	
13	Clock/qualifier probe	TR14702-04		1	1	
14	Input cable (for TRG, READY, ADVANCE)	MI-02	1			BNC 50Ω
15	Output cable (for VIDEO OUT)	MO-15	1			BNC 75Ω

Table 1-1 TR4751 Configuration List (Cont'd)

No.	Item	Stock No.	Standard	Option		Remarks
				71	73	
16	System software package	P4751-0001FJ	1			System disk
17	Floppy disk		1			User disk

1.3.2 Operating Environment

(1) Avoid using the TR4751 in places exposed to dust, corrosive gases, or direct sunlight. The ambient temperature and humidity must be +5°C to +40°C and 80% or less, respectively.

(2) Natural ventilation from the funnel removes heat from the TR4751. Don't close the funnels on the top and rear panel during operation of the TR4751. During use of the TR4751, place it ten or more centimeters from the back wall or any object.

(3) The TR4751 is designed with the AC power line noise in mind. However, use it in an environment as free as possible from noise. If the AC power line has excessive noise, use the noise filter for noise suppression.

(4) Avoid using the TR4751 where there is considerable vibration.

(5) The storage temperature for the TR4751 is -10°C to +60° (+10°C to +60°C for floppy disks). When the TR4751 is not used for a long time, protect it with a vinyl cover, or put the TR4751 in a cardboard box, and store the box in a location with low humidity and away from direct sunlight.

1.3.3 Setting Up TR4751

To set up the TR4751, connect the data probe to the acquisition section and an output cable to the pattern generator.

(1) Connect the POD G probe to the connector labeled ACQ-G on the rear panel.

(2) Connect the POD H probe to the connector labeled ACQ-H on the rear panel.

(3) Connect the PG-A (0 to 7) cables to the connectors labeled PG-A (0 to 7) on the rear panel.

(4) Connect the PG-A (8 to F) cables to the connectors labeled PG-A (8 to F) on the rear panel.

(5) Connect the STRB/CLK cable to the connector labeled STRB/CLK on the rear panel.

(6) Connect the EXT-PS cable to the connector labeled EXT-PS on the rear panel.

(7) Connect the PG-CONT probe to the connector labeled PG-CONT on the rear panel.

(8) Connect the keyboard to the connector labeled KEYBOARD on the front panel.

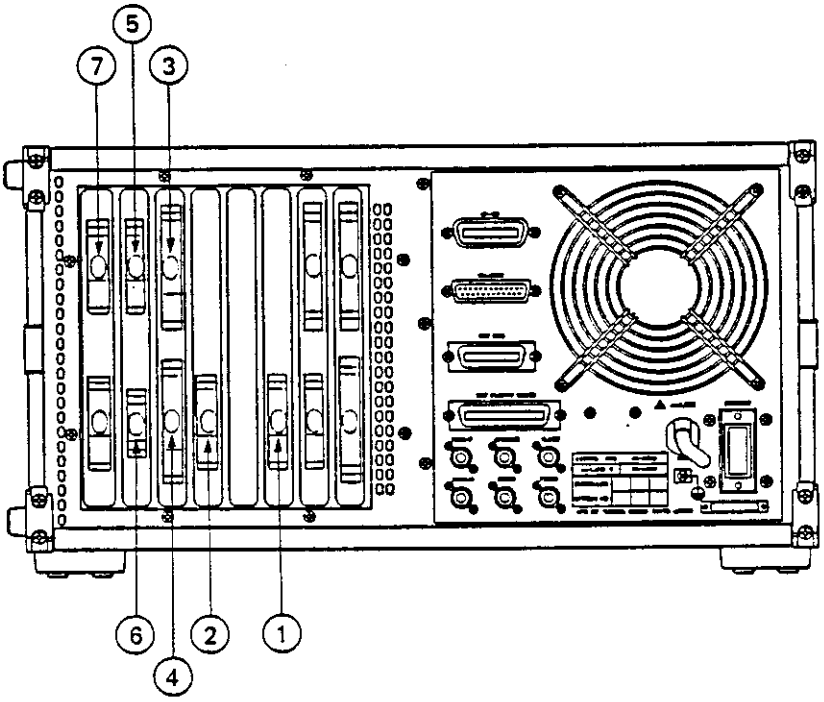


Figure 1-1 Connecting Cables (1/2)

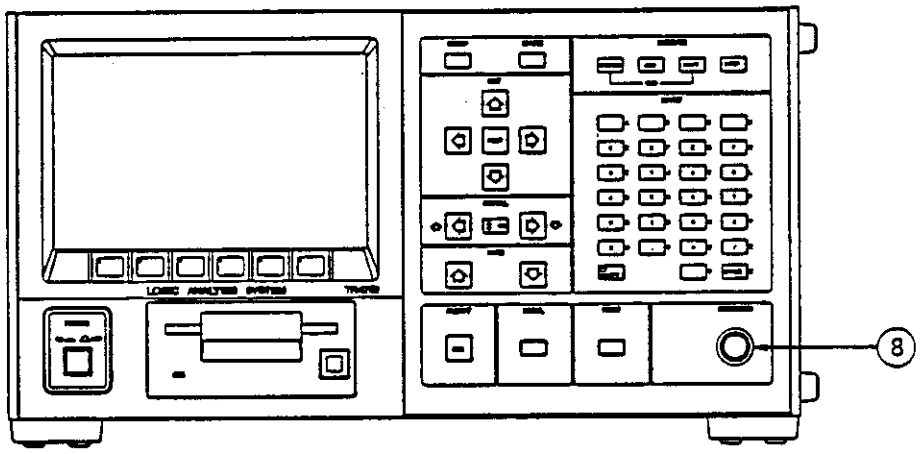


Figure 1-1 Connecting Cables (2/2)

(9) Check whether the POWER switch on the front panel is set off.

(10) Check whether the BREAKER on the rear panel is set on.

(11) Connect the supplied power cable to LINE connector.

Power cables
The power cable plug has three prongs. The central round prong of the plug is for grounding.
When connecting the plug to an outlet using an adapter, connect the ground lead of the adapter or the ground terminal on the rear panel to an external earth.
Two electrodes (A and B) of adapter A09034 are different in width as shown in Figure 1-2 (b). When plugging this adapter to an outlet, check the direction. Purchase adapter KPR-13 when A09034 cannot be connected to the outlet.

Breaker
The TR4751 is provided with a breaker. Don't use the breaker instead of the power switch for safety consideration. Check whether the POWER switch is off and the breaker is on before plugging the power cable into the outlet.

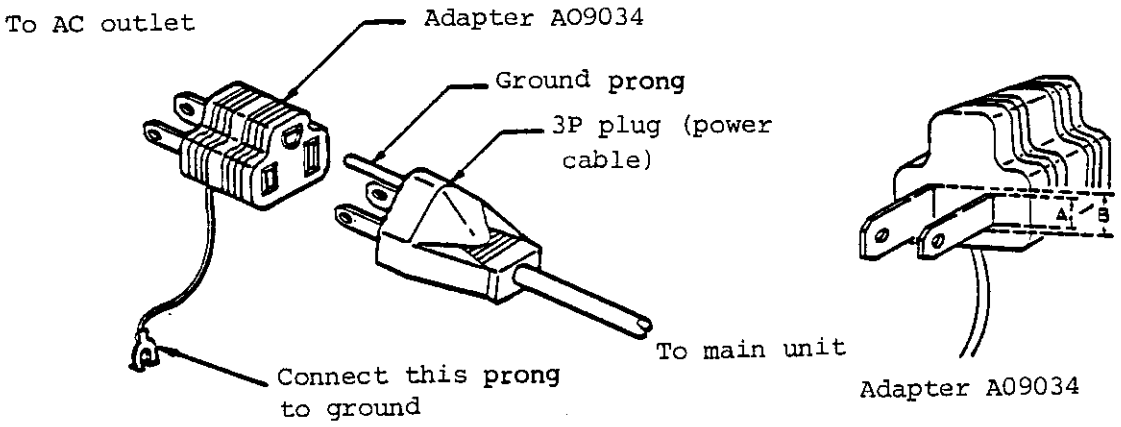


Figure 1-2 Power Cable Plug and Adapter

2. FOR THE TR4751 BEGINNERS

2.1 ACTIVATING THE TR4751

Check that the POWER switch is off and the breaker on the rear panel is on. Then connect the power cable to the AC outlet. After this, insert the system floppy disk into the drive slot and turn the POWER switch on.

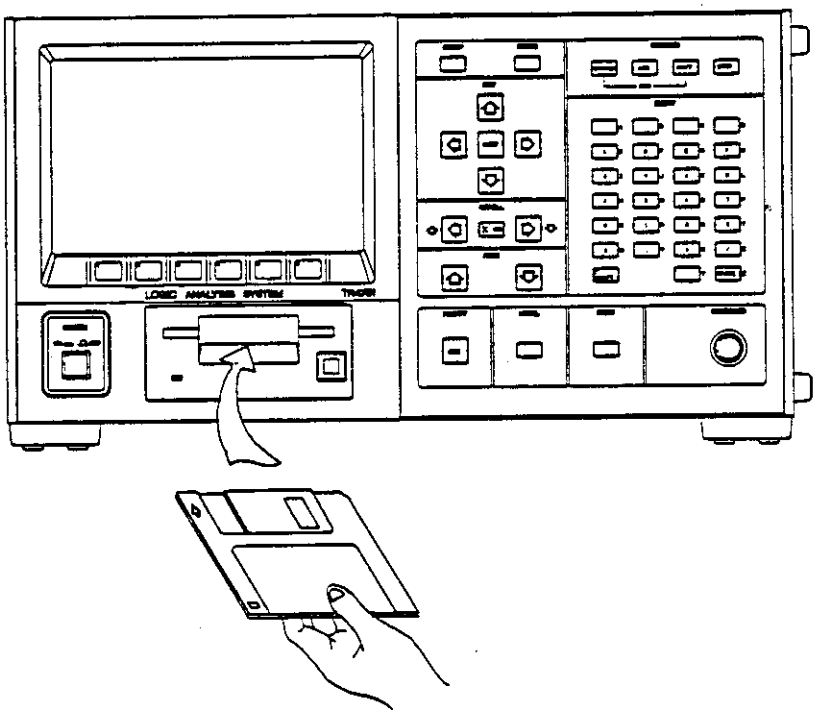


Figure 2-1 Inserting Floppy Disk into Drive Slot

The screen turns as shown in Figure 2-2 approximately ten seconds after turning the POWER switch on. This display means that the system program is loading to the TR4751.

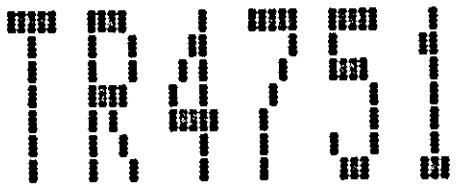


Figure 2-2 Program Loading

If you turn the POWER switch on again several seconds after power-on, the memory test display appears as shown in Figure 2-3.

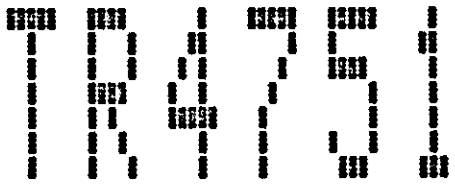
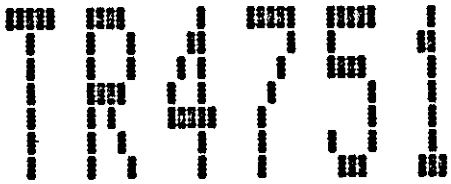


Figure 2-3 Memory Test Screen

The display shown in Figure 2-4 appears if no disk is placed in the drive slot at power-on. This display is kept until a disk is correctly inserted.



REV. 1.8

Please enter system software package

Figure 2-4 Prompt Screen for Disk Entry

The display shown in Figure 2-5 appears when the system program is loaded to the TR4751.

TR4751 *** Configuration & Self test ***

1.	Controller	pass
2.	CRT controller	pass
3.	GP-IB controller	pass
4.	16 ch acquisition (ACQ_H,G)	pass
5.	16 ch acquisition (ACQ_C)	Not install
6.	32 ch acquisition (ACQ_B,A)	Not install
7.	16 ch pattern generator (PG_A)	pass
8.	16 ch pattern generator (PG_B)	Not install
9.	16 ch pattern generator (PG_C)	Not install
10.	16 ch pattern generator (PG_D)	Not install

*** Setup menu select ***

1.	ACQ spec:	Data acquisition
2.	PATT spec:	Pattern generation
3.	Display:	Display type of result
4.	COMM spec:	RS-232C,GP-IB communication
5.	REF. data:	Reference data display
6.	HELP	Operating manual for use

ACQ spec PATT spec Display COMM spec REF. data HELP

Figure 2-5 Main Setup Menu Screen

2.2 BASIC KEY SWITCHES: THESE KEYS MUST BE REMEMBERED.

This section explains the functions of the keys on the front panel and how to use them.

Figure 2-6 shows the TR4751 front panel.

(1) Soft keys
Selects an item of each parameter or a screen. The items are displayed on the bottom line of the screen according to the position of the cursor (inverse-blinking display characters). You can select the desired item by pressing the corresponding software key.

(2) Edit keys
Moves the cursor (inverse-blinking display chararcter) to the desired position.

⇧ : Moves the cursor up.
⇩ : Moves the cursor down.
⇨ : Moves the cursor right.
⇦ : Moves the cursor left.
NEXT: Moves the cursor right; if the cursor exists at the end of the line, it is moved to the beginning of the next line, or moved to the top line if it on the bottom line.

(3) Scroll keys
Scrolls the screen when "Scroll: ↑↓ " or "Scroll: ← → " is displayed on the lower right of the screen. Each time the middle key is pressed, the scroll direction switches (up/down, or right/left).

(4) Page keys
Used to page the screen when the "Scroll: ↑↓" is displayed on the screen in the same way as the scroll keys. The line count to be paged depends on the selected screen.

(5) Entry keys
Sets words, numeric values or labels. Normally, the entry keys must be 0 thru F and the special keys. Also the alphabetic characters of A thru Z can be used by pressing the shift key along with it.

(6) Setup key
Turns the current screen to the main setup menu screen.

(7) Store key
Stores the captured data to the reference memory and its array data.

(8) Floppy executing key
Calls the floppy function and executes the floppy operation.

(9) Local key
Specifies the manual mode when the TR4751 is placed in the remote control mode by GPIB. Note that the local key is invalid in the local lockout mode.

(10) Print key
Prints the data on the screen or the data related to it.

(11) Execution keys
SYSTEM: Activates the acquisition section and the pattern generator.
ACQ : Activates the acquisition section only.
PATT : Activates the pattern generator only.
STOP : Deactivates the acquisition section and the pattern generator.

(12) Eject button
Ejects the floppy disk from the drive slot. Do not press the eject button while the floppy drive monitor lamp is lit.

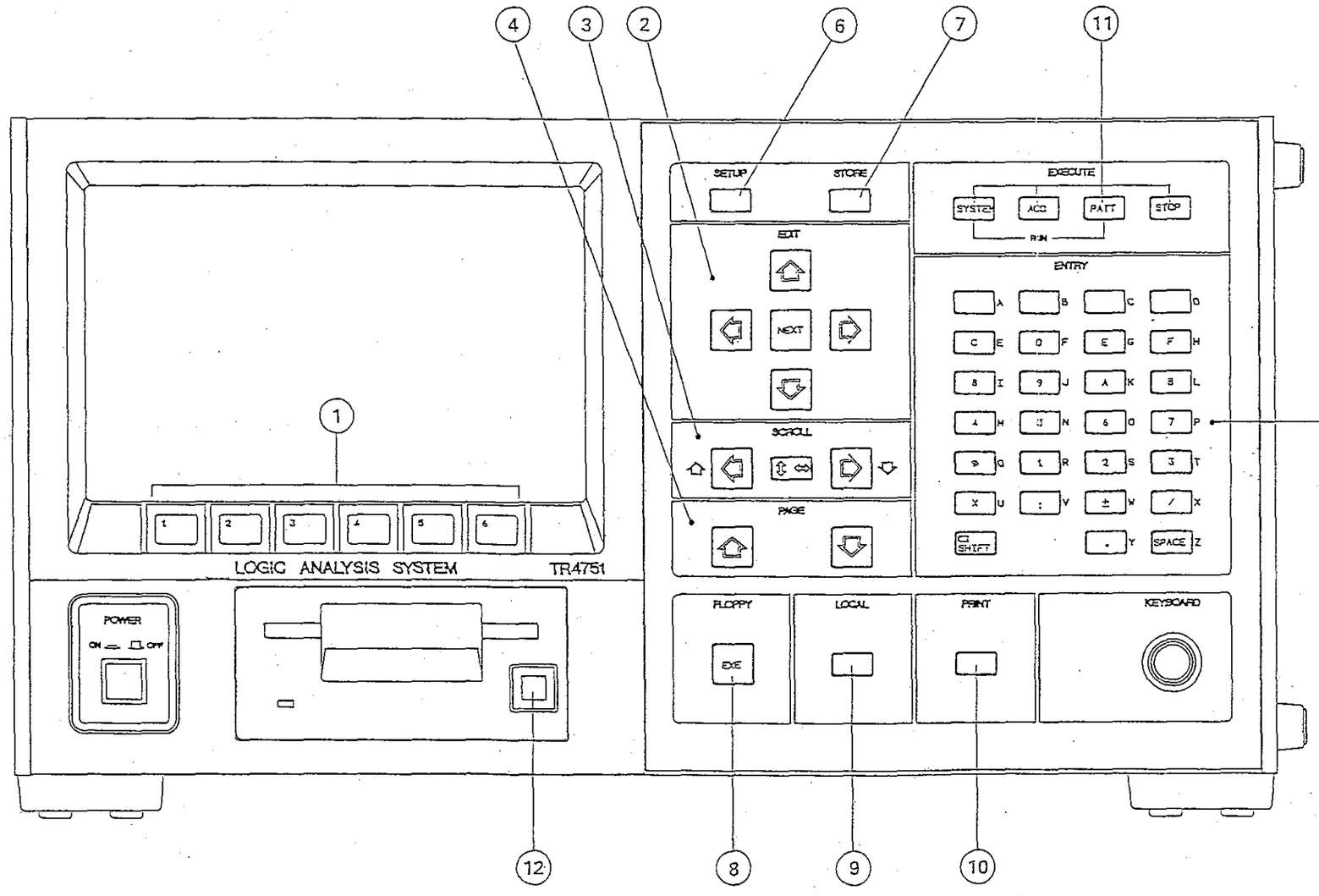
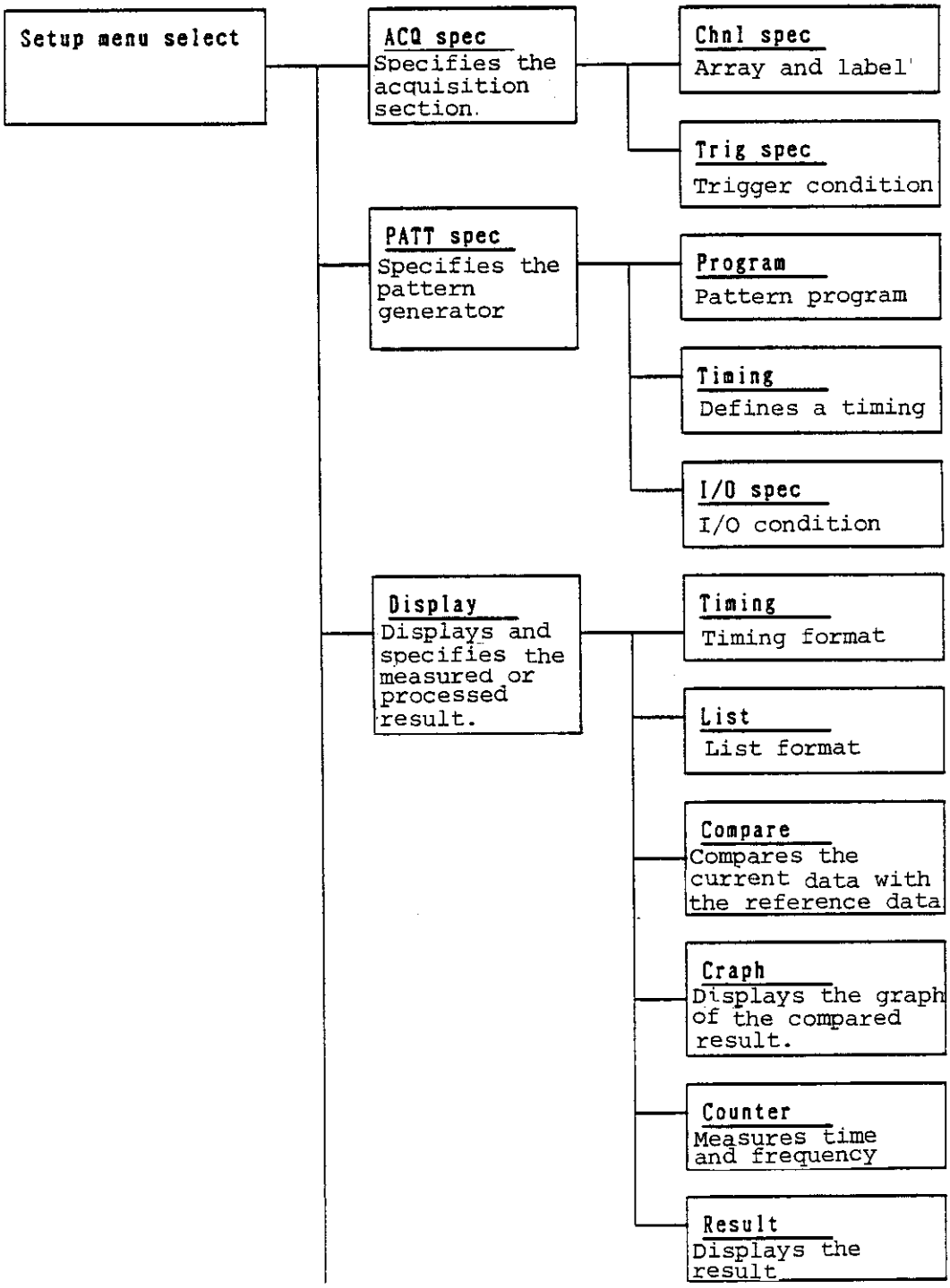


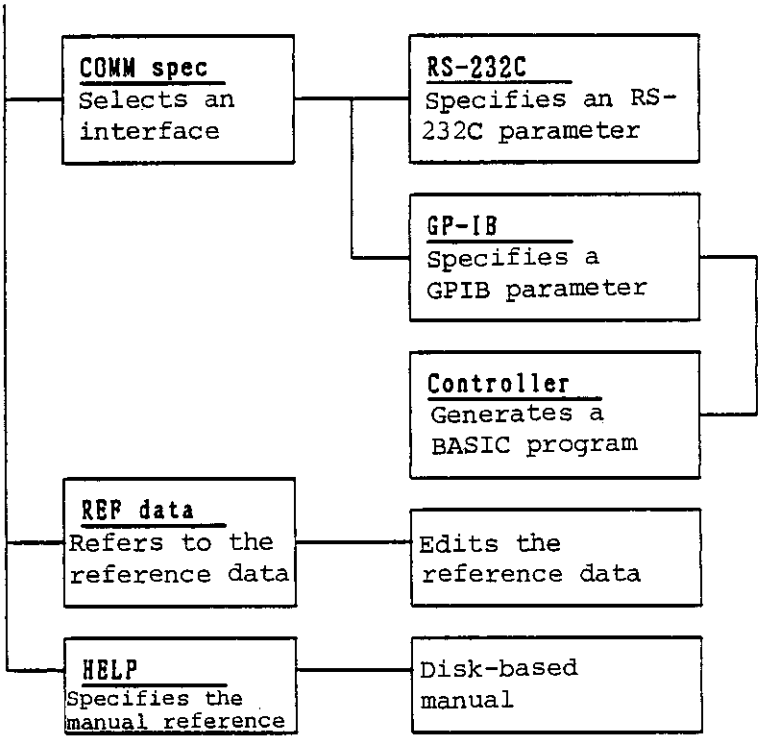
Figure 2-6 TR4751 Front Panel

2.3 SCREEN CONFIGURATION



Continued to the next page

Continued



Use of the soft keys switches the screen to the right screen in the above configuration list; the screen is switched to the left one by using the exit or setup key.

3. BASIC OPERATION OF THE TR4751

This chapter explains practically how to operate the TR4751, giving an example of the ECL memory measurement. See Chapter 4 for detailed information on the commands used or further operation of the TR4751.

3.1 PATTERN GENERATION

The ECL memory is a general static RAM. This memory generates the address, write-enable and block selecting signals, and the write data.

Figure 3-1 shows the relationship among them.

Set the amplitude to the ECL level (V_{OH} = -0.9 V and V_{OL} = -0.3 V), and the access time to 100 ns.

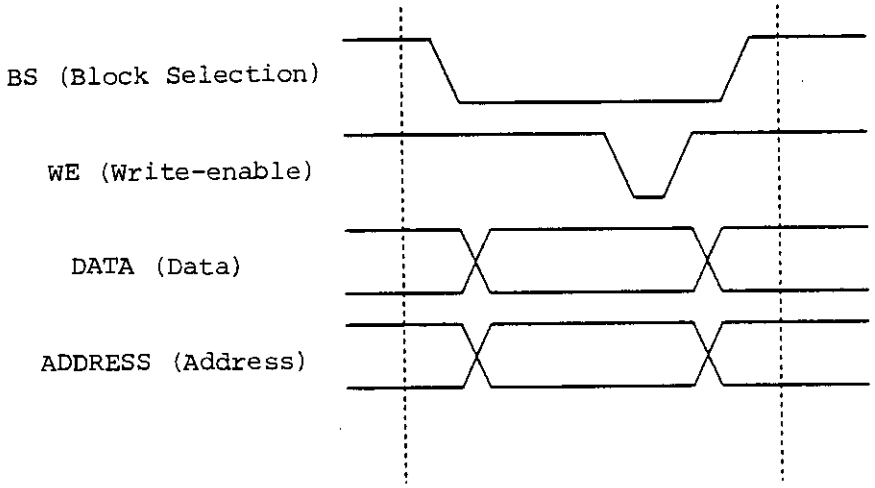


Figure 3-1 RAM Timing Chart

(1) Press the SETUP key to display the screen shown in Figure 3-2 (a) on the lower part of the CRT.

(2) Select PATT spec to display the screen shown in Figure 3-2 (b) on the lower part of the CRT.

(3) Select Program to set the screen as shown in Figure 3-2 (c).

(4) Create the program as shown in Figure 3-2 (d) according to the following procedure:

① Enter the label on the first line and its data by using the entry keys (enter "1" for data.).
② Move the cursor to the second line pressing ⇩ and select soft key INC/DEC . Then enter "A255".
③ Move the cursor down using the NEXT key to input data.
④ Move the cursor one line down pressing ⇩ , then enter "A225".
⑤ Move the cursor in the same way as described in step ③ to input data.
⑥ Move the cursor one line down pressing ⇩ , and select DEC by pressing INC/DEC twice. Then enter "A023".
⑦ Move the cursor to the left end of the next line, and enter "/END".
⑧ Specify the pattern starting position by moving the cursor to "Start position:" and use entry key "x" (don't care) key to blank the setting column.

```
                *** Setup menu select ***

          1. ACQ spec:  Data acquisition
          2. PATT spec: Pattern generation
          3. Display:   Display type of result
          4. COMM spec: RS-232C,GP-IB communication
          5. REF. data: Reference data display
          6. HELP       Operating manual for use

ACQ spec  PATT spec  Display  COMM spec  REF. data  HELP
```

(a)

```
             *** Pattern generator setup menu ***

          1.  Pattern program editor
          2.  Timing and channel assignment
          3.  Output level and input control
          4.
          5.
          6.  Return to the main setup menu

Information:                                  Scroll:

Program  Timing  I/O spec.  _______  _______  exit
```

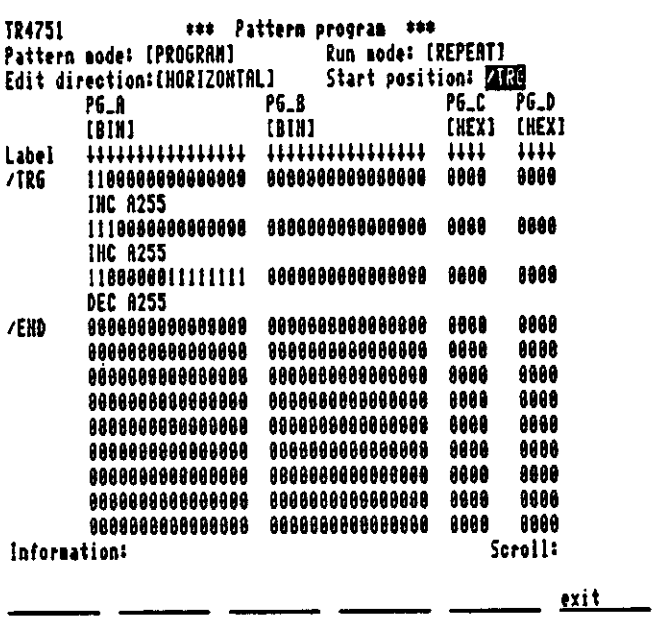
(b)

```
TR4751          *** Pattern program ***
Pattern mode: [PROGRAM]        Run mode: [REPEAT]
Edit direction:[HORIZONTAL]    Start position: ____
       PG_A             PG_B             PG_C   PG_D
       [BIN]            [BIN]            [HEX]  [HEX]
Label  ↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓ ↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓ ↓↓↓↓   ↓↓↓↓
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
       0000000000000000 0000000000000000 0000   0000
Information: 0000                          Scroll: ↑

insert  delete  _______  _______  new  exit
```

(c)

Figure 3-2 Procedure for Setting Pattern Program



(d)

Figure 3-2 Procedure for Setting Pattern Program (Cont'd)

(5) Select exit to set a timing for each signal. This turns the screen as shown in Figure 3-2 (b). Then select Timing to turn the screen as shown in Figure 3-3.

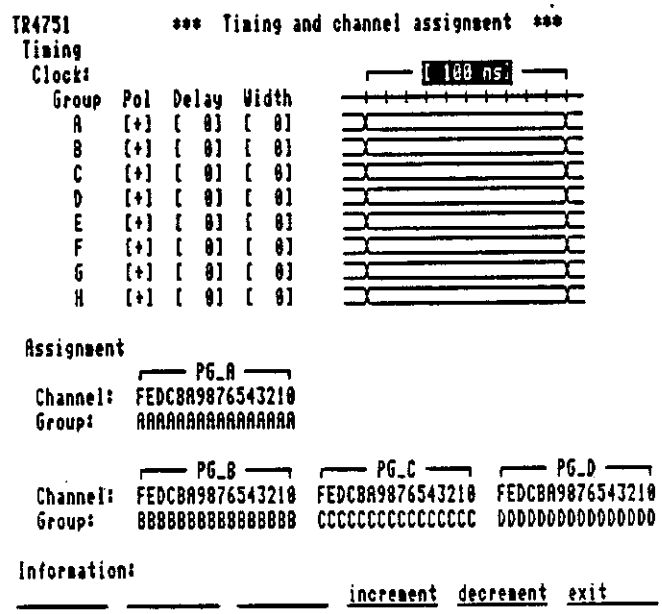


Figure 3-3 "Timing" Initial Screen

Define timing groups as follows:
- Group A: Address signal and write data
- Group B: Write-enable signal
- Group C: Block selecting signal

Select each signal output channel as follows:

Address signal	:	Channels 0 to 7
Write data	:	Channel D
Block selecting signal	:	Channel E
Write-enable signal	:	Channel F

Set parameters according to the following procedure:

① Move the cursor to "Pol" of Group B using the edit keys, and select negative. Then move the cursor to "Delay" and press increment to set "5". In the same way, set "Width" to 3. These values correspond to the timing for 1 access cycle of the write-enable signal.

② Define the timing for the block selecting signal in the same way given in step ①.

③ Arrange channel assignment in the default status. To assign the defined timing group to each channel, move the cursor to "Group" in "Assignment" using the edit keys. Then set the screen as shown in Figure 3-4 using the entry keys.

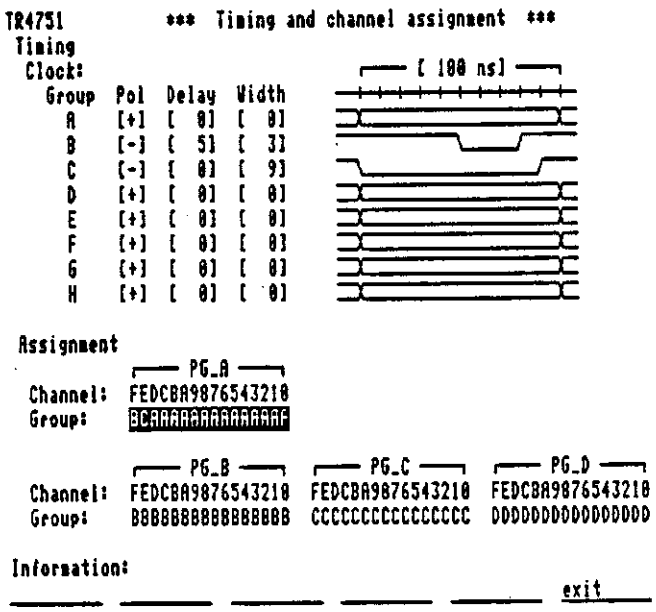


Figure 3-4 Timing Setting

(6) Set the positions of the output amplitude and acquisition. Select exit when the setting given in (5) completes to turn the screen as shown in Figure 3-2 (b). Then select I/O spec to turn the screen as shown in Figure 3-5.

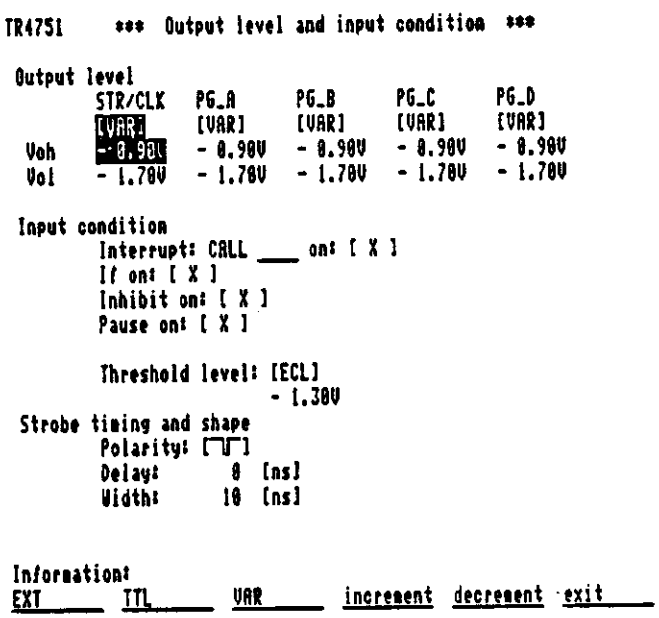


Figure 3-5 "I/O spec" Initial Screen

Set the positions of amplitude and acquisition according to the following procedure:

① Now, the cursor is at "STR/CLK" and the output type of the selected item is set to [VAR]. Item "Voh" is displayed in reverse. This means that "Voh" value can be changed by using the soft keys. The [VAR] default value is set to the ECL level (V_{CC} = 0 V and V_{EE} = -5.2 V). The value in [VAR] can be changed by using the increment and decrement keys.

② Pressing edit key ⇩ displays "Vol" in reverse. "Vol" value can be changed by using the increment or decrement key.

③ Set the acquisition position in "Strobe timing and shape". Only "Delay:" is available for strobe signal. ("polarity:" and "Width:" are available for the strobe signal output from the connector on the rear panel.)
Move the cursor to the numeral item "Delay:". Enter "30" using the entry keys. If you make an error entry, press "x" (don't care) and reenter correctly.

```
TR4751     *** Output level and input condition ***

Output level
        STR/CLK   PG_A      PG_B      PG_C      PG_D
        [VAR]     [VAR]     [VAR]     [VAR]     [VAR]
 Voh    - 0.90V   - 0.90V   - 0.90V   - 0.90V   - 0.90V
 Vol    - 1.70V   - 1.70V   - 1.70V   - 1.70V   - 1.70V

Input condition
        Interrupt: CALL ___ on: [ X ]
        If on: [ X ]
        Inhibit on: [ X ]
        Pause on: [ X ]

        Threshold level: [ECL]
                         - 1.30V
Strobe timing and shape
        Polarity: [ ]
        Delay:    30 [ns]
        Width:    10 [ns]


Information:
_____ _____ _____ _____ _____ exit
```

Figure 3-6 Setting Positions of Amplitude and Acquisition

The pattern generator setting is completed through these operations.

Connect an accessory adapter to the oscilloscope, and then connect the block selecting signal to one channel and strobe, and address or write-enable signal to the other channel.

No waveform appears on the oscilloscope yet. Press the PATT-RUN key to specify a synchronization for the oscilloscope on the block selecting signal, and the waveform will be monitored as shown in Figure 3-7. (Figure 3-7 shows the monitor operation using a four-channel oscilloscope.)

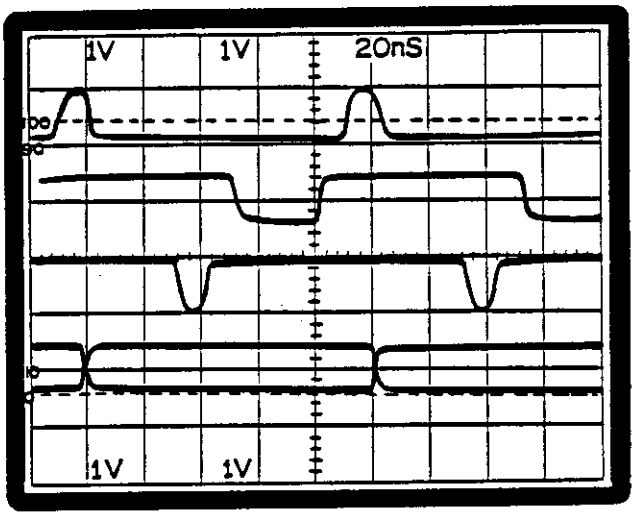


Figure 3-7 Output Pattern Waveforms

3.2 ACQUISITION AND DISPLAYING RESULT

The patterns generated in Section 3.1 are captured and then displayed in the specified data format by the acquisition section.

(1) Press the setup key to turn the screen to the main setup menu screen shown in Figure 3-8 (a).

(2) Select ACQ spec to turn the screen as shown in Figure 3-8 (b).

(3) Select Chnl spec to turn the screen as shown in Figure 3-8 (c).

(4) Set the threshold voltage to capture data. The individual threshold voltage can be set for each probe pod of ACQ-H and ACQ-G. The pattern generator is set to the default mode. If the setting of the pattern generator is changed, set the threshold voltages to the middle voltages of "Voh" and "Vol". To do this, move the cursor to the threshold type selecting item, and select VAR, then use the increment and decrement keys.

(5) Name each signal so that the corresponding result can be easily monitored, and define which channel to be used to capture each result.
Move the cursor using the edit keys, and set the screen as shown in Figure 3-8 (c) using the entry and soft keys.

```
*** Setup menu select ***

1. ACQ spec:  Data acquisition
2. PATT spec: Pattern generation
3. Display:   Display type of result
4. COMM spec: RS-232C,GP-IB communication
5. REF. data: Reference data display
6. HELP       Operating manual for use

Information:
ACQ spec  PATT spec  Display  COMM spec  REF. data  HELP
```

(a)

```
                *** ACQ spec menu select ***

          1.  Channel specification
          2.  Trigger specification
          3.
          4.
          5.
          6.  Return to the main setup menu

Information:
Chnl spec  Trig spec  _______  _______  _______  exit
```

(b)

```
TR4751                *** Channel spec. ***
  Threshold level
     ACQ_H    ACQ_G
     [ECL]    [ECL]
     -1.30V   -1.30V

  Label and channel assignment
     Label Pol
     BS    [+]  POD  6
                CH   7

     WE    [+]  POD  6
                CH   6

     DATA  [+]  POD  6
                CH   5

     ADD   [+]  POD  00000HHH
                CH   43210765

           [+]  POD
                CH
Information:
insert   delete   ACQ_H   ACQ_G   _______   exit
```

(c)

Figure 3-8 "ACQ spec" Screen Setting

(6) Press exit to turn the screen as shown in Figure 3-8 (b). Then press Trig spec to turn the screen as shown in Figure 3-9.
The trigger condition for an acquisition is set by using the "/TRG" command described in the program.
To do this, set "CLOCK(CLK1):" to 5 ns using the decrement key, and move the cursor to "Link module:" to select PAT.

```
TR4751              *** Trigger spec. ***
[ACQ_M_G]
  Clock (CLK1):[INT   5 ns]                    Memory size:[4K]
  Trigger mode:[ MATCH ]          Position:[BEGIN ]
           BS      WE      DATA    ADD
          [BIN]   [HEX]   [HEX]   [HEX]
  Trigger: X       X       X       XX
  Glitch:  X       X       X       XX
    Filter: 1    Event:     1     Link module:[PAT]
  Clock qualifier: Q1(X)*Q2(X)

Information:
EXT     PG strobe  INT     increment  decrement  exit
```

Figure 3-9 Leading "Trig spec" Screen and Setting Trigger Condition

(7) Connect the pattern generator outputs to the data acquisition probes by using the accessory adapter. (See Figure 3-10.)

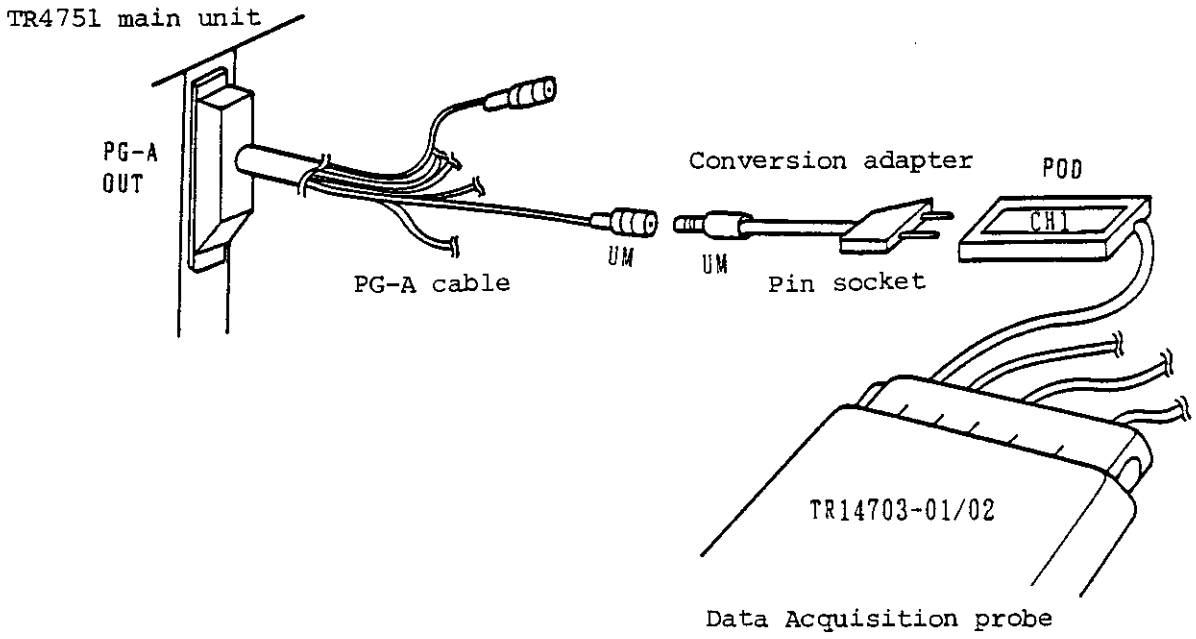


Figure 3-10 Connecting Pattern Generator to Data Probes

(8) Now, the acquisition section setting has been completed. Press the SYSTEM-RUN key to start a measurement. When terminating the measurement, the screen as shown in Figure 3-11 appears.

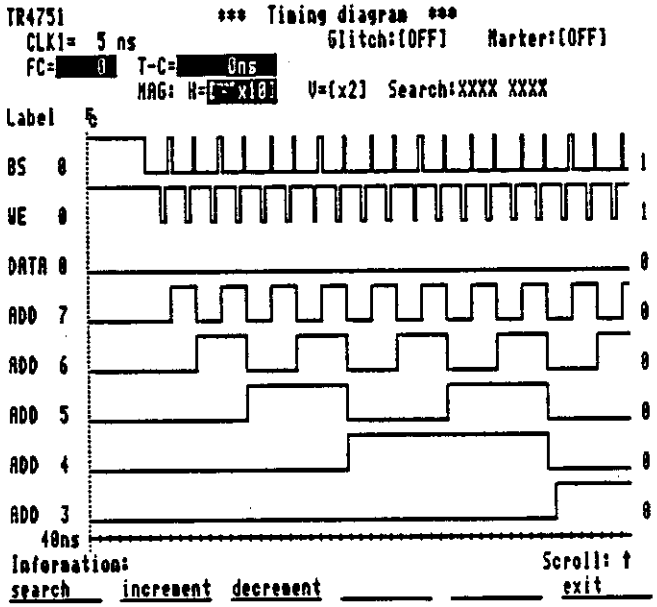


Figure 3-11 "Timing" Screen (from the default mode)

(9) Press the exit key to turn the screen as shown in Figure 3-12, and select List. This turns the screen as shown in Figure 3-13 and displays the captured data in a list form.

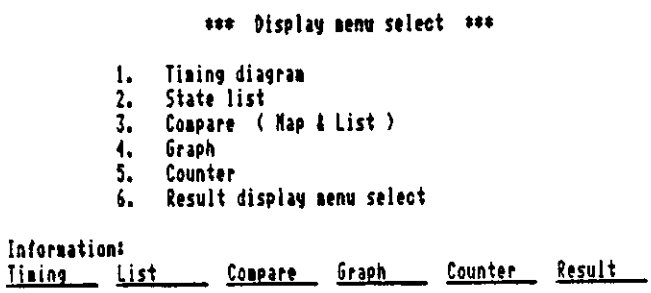


Figure 3-12 "Display menu select" Screen

```
TR4751                  *** State list ***

Search:X        X       X       XX
SEQ.    BS      WE      DATA    ADD
[DEC]   [BIN]   [BIN]   [HEX]   [HEX]
0000 F  1       1       0       00
0001    1       1       0       00
0002    1       1       0       00
0003    1       1       0       00
0004    1       1       0       00
0005    1       1       0       00
0006    1       1       0       00
0007    1       1       0       00
0008    1       1       0       00
0009    1       1       0       00
0010    1       1       0       00
0011    1       1       0       00
0012    1       1       0       00
0013    1       1       0       00
0014    1       1       0       00

Information:                                        Scroll: ↑
Search  _____   _____   _____   _____   _____       exit
```

Figure 3-13 List Screen

(10) Press exit to turn the screen as shown in Figure 3-12. Then select Compare to display the screen as shown in Figure 3-14.

```
TR4751          *** Compare (Map) ***
   Run mode: SINGLE             Memory size: 4K

0000                                  0031
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||    Total error:4853
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||    Error in cursor:
    ||||||||||||||||||||||||||||||||    0000  0032  0064  0096
    ||||||||||||||||||||||||||||||||    0128  0160  0192  0224
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
    ||||||||||||||||||||||||||||||||
4064                                  4095

Information:
Preview  Next    Zoom    _____   List    exit
```

Figure 3-14 Mismatch Screen at "Compare"

While the screen shown in Figure 3-14 is appearing, press the STORE key on the front panel to display the data shown in Figure 3-15.

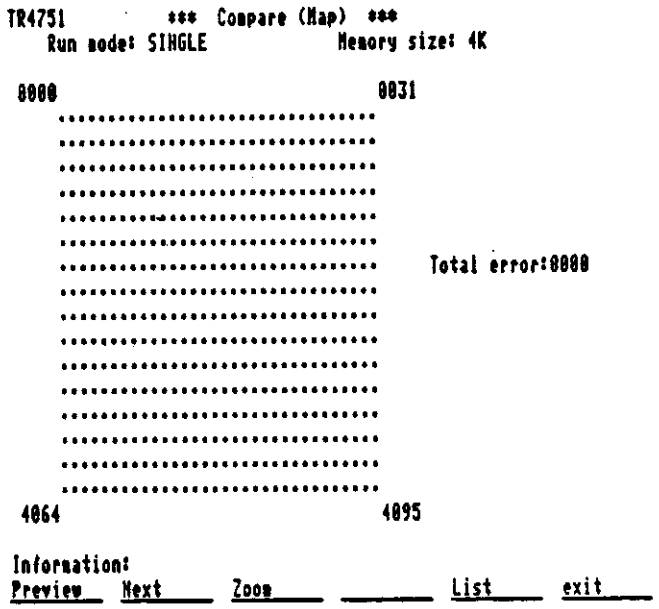


Figure 3-15 Match Screen at "Compare"

(11) Press exit to display the display selection menu shown in Figure 3-12 on the screen. Then select Counter to turn the screen as shown in Figure 3-16.
Here, select polarity to specify measuring ⊓. This measures and displays the negative time interval.

TR4751 *** Timer/Counter ***
Polarity: ⊓

Label	T(u)max	T(u)min	T(u)ave	Period	Frequency
[redacted]	90ns	85ns	85ns	100ns	10MHz
WE 0	25ns	20ns	20ns	----	----
DATA 0		Data continue			
ADD 7	95ns	95ns	95ns	200ns	5MHz
ADD 6	195ns	195ns	195ns	400ns	2MHz
ADD 5	395ns	395ns	395ns	800ns	1MHz
ADD 4	795ns	795ns	795ns	1.60μs	625KHz
ADD 3	1.70μs	1.63μs	1.65μs	3.20μs	312KHz
ADD 2		Data continue			
ADD 1		Data continue			
ADD 0		Data continue			
ADD 4	795ns	795ns	795ns	1.60μs	625KHz
ADD 3	1.70μs	1.63μs	1.65μs	3.20μs	312KHz
ADD 2		Data continue			
ADD 1		Data continue			
ADD 0		Data continue			

Information: Scroll: ↑
polarity next preview exit

Figure 3-16 Counter Screen

(12) Press exit to display the screen shown in Figure 3-12. Then select Result to turn the screen as shown in Figure 3-17. This screen prompts you to specify which type of result to be displayed when a data acquisition finishes. To do this, select Timing and move the cursor to "Run mode:", then select Repeat. This displays the acquisition result and repeats the measurement.

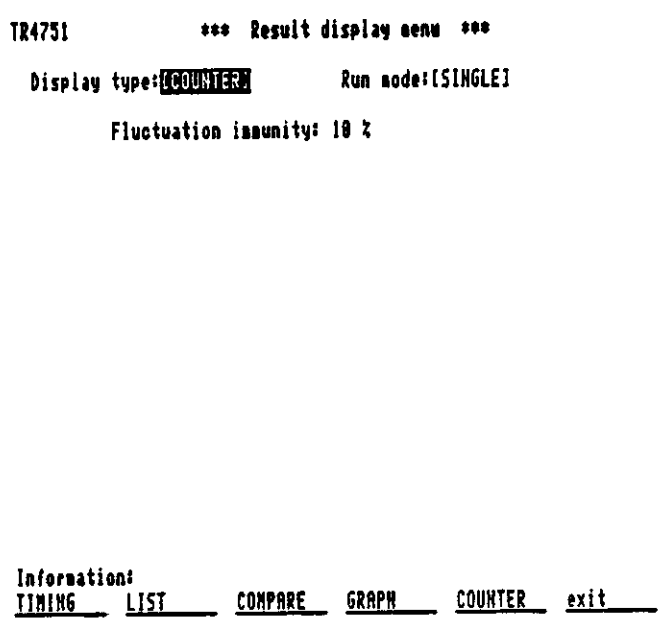


Figure 3-17 Result Screen

3.3 BASIC OPERATING FLOPPY DISK

This section describes how to store the acquisition parameters and pattern program created in Section 3.1 onto a floppy disk.

(1) Press the STOP key on the front panel. This terminates the measurement and displays timing waveforms. This means the system suspension. Insert the user disk into the disk drive. See Chapter 6 "Floppy Disk Operation" if the floppy disk to be used has not been formatted. Then press the FLOPPY-EXE key on the front panel to display the screen as shown in Figure 3-18.

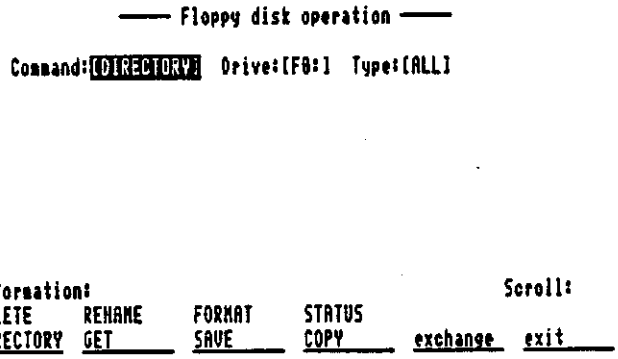


Figure 3-18 Floppy Disk Operation Screen

The upper-half portion on the CRT is the half-luminance display.
To store data onto a floppy disk, use the SAVE command. Move the cursor to "File:" and name the file within six characters. Then move the cursor to "Type:" to define the attribute of the named file.
Select AQP or PGU for the acquisition parameters and pattern program, respectively.
When the setting is completed, press the FLOPPY-EXE key. This lights the floppy drive monitor lamp and stops the cursor blinking. When writing data to the floppy disk ends, the monitor lamp goes off and the cursor starts blinking again. Create the necessary file by repeating these operations.
To return the Floppy disk operation screen to the initial mode, use exit key.
Use the GET key to load the data in the generated file to the system.
For details, see Chapter 6 "Floppy Disk Operation".

MEMO

4. DETAILED OPERATION OF THE TR4751

This chapter describes the meanings and setting procedures of the item displayed on each screen in detail.

Before going on to Chapter 4, the beginners of the logic analyzer and pattern generator should have read Chapter 3 to master the basic operation of the TR4751.

4.1 ACQ spec (THE ACQUISITION SECTION SETTING)

4.1.1 Chnl spec (The Input Channels Definition)

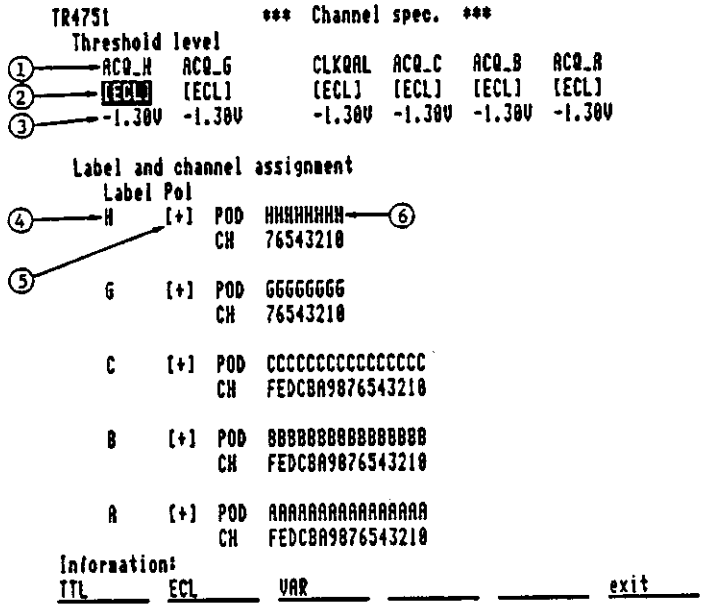


Figure 4-1 "Chnl spec" Screen

(1) Pod names
The TR4751 is provided with 16-channel inputs. These 16 channels are divided into two probe pods every eight channels. External clocks and clock qualifier inputs are contained in the same pod. The two probe pods are named as POD-H and POD-G. These pods correspond respectively to ACQ-H and ACQ-G on the screen, and the rear panel. Figure 4-2 shows the channel assignments.

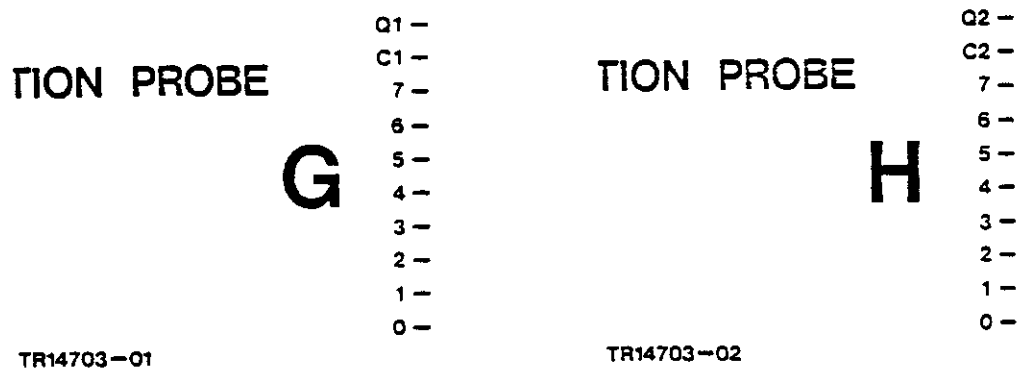


Figure 4-2 POD-H/G Channel Assignments

(2) Selecting threshold voltage type
You can select one of the following items as the threshold voltage:

TTL ECL VAR exit

Selecting TTL sets the threshold voltage to +1.40 V and selecting ECL sets -1.30 V.
The threshold voltage can be set to any value by selecting VAR and using the increment and decrement keys. The setting resolution is 50 mV and the range must be within +6.35 V to -6.35 V. When the threshold voltage type is selected, the values in the lower column are displayed in reverse.

TTL ECL VAR increment decrement exit

(3) Threshold voltage value
Displayed in three digits.

(4) Labeling
A label is specified with up to four alphanumeric characters by using the entry keys.

(5) Changing logical polarity
Selecting positive specifies the positive logic (displaying "+") and selecting negative specifies the negative logic (displaying "-").
The input logic is replaced with the one displayed on the screen.

positive negative exit

(6) Assigning channels
Specify the pod name and channel name in pairs when defining the channel assignments. When the cursor exists on the ACQ side, the following soft key labels appear on the screen:

insert	delete	POD-H	POD-G		exit

POD-H	POD-G				exit

(When without any label definition)

When the cursor is on the CH side, the following soft key labels appear on the screen:

insert	delete				exit

insert allows you to insert a space on the cursor location shifting right channels to the right and define a new channel there.
delete deletes the channel pointed to by the cursor, and shifts the right channels of the cursor to the left.

CAUTION

(1) If soft keys are operated too fast, a soft key label may be overlapped and an error occur.
(2) To release from this screen, use the setup key or exit key on the front panel.

4.1.2 Trig spec (Trigger Conditions Setting)

Defines the trigger conditions, sampling clock and memory size in this screen.

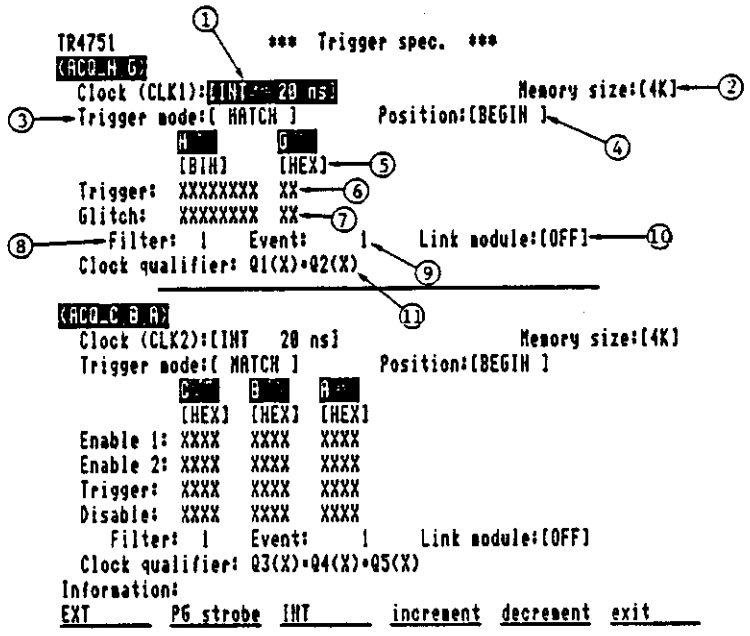


Figure 4-3 "Trig spec" Screen

(1) Setting sampling clock CLK1 (Clock)
You can select one of the following items as the sampling clock:

EXT | PG strobe | INT | | | exit

EXT specifies an external clock and is provided for every probe pod. Selecting EXT displays the screen as shown in Figure 4-4, and enables selection of the logical polarity of an external clock input. Then the soft key labels are displayed as follows: The logical sum is performed on C1 and C2.

positive | negative | off(X) | | | exit

Selecting PG strobe to performs a sampling in the "Strobe (delay)" timing specified on the "I/O spec" screen.
Select INT to perform sampling on data asynchronously. The sampling rate must be within 50 ms to 2.5 ns range in 1-2-5 step. The soft key labels are displayed as follows:

EXT | PG strobe | INT | increment | decrement | exit

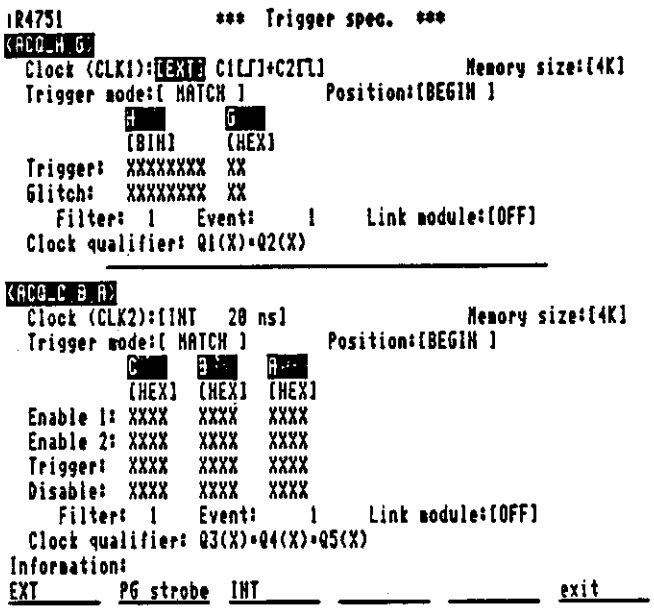


Figure 4-4 "Trig spec" Screen with EXT Selected

(2) Selecting memory size (Memory size)
You can select a long size or a short size as the memory size. Move the cursor to "Memory size:" to display the following soft key labels:

4K 1K exit

When the sampling clock is set to 2.5 ns with INT selected, the following labels appear on the screen:

8K 2K exit

(3) Selecting trigger mode (Trigger mode)

MATCH RELEASE exit

The MATCH trigger generates a trigger when the word defined in "Trigger:" is captured. The RELEASE trigger generates a trigger when the word defined in "Trigger:" and the continuity of this word is disabled.

(4) Setting trigger position (Position)
"Position" sets the position of the trigger word defined in "Trigger mode:" and "Trigger:" in memory.

BEGIN CENTER END DELAY _____ exit

BEGIN positions the trigger word at the beginning of address in memory.

CENTER positions the trigger word at the center of address in memory.

END positions the trigger word at the end of address in memory.

DELAY positions the trigger word to the specified value at the beginning of memory.

A positive value positions the trigger word in the former half of memory and a negative value positions it in the latter half. DELAY must be specified within -4091 to 28659. If an error is made in specification, delete it by pressing "x" (don't care) key and enter the correct value again.

(5) Selecting trigger word display format
The trigger word is displayed in one of the following three formats:

BIN OCT HEX _____ _____ exit

The display format defined here has no effect on the format for the result displays.

(6) Setting trigger word
Set the desired trigger word in the specified display format by using the entry keys.

(7) Setting glitch trigger (Glitch)
The TR4751 is provided both with the pattern and glitch triggers. To set the glitch trigger, select BIN as the display format and set "1" to the bit corresponding to the desired channel. (If a format other than BIN is selected, "$" may appear on the screen.) The glitch trigger has a logical sum relation to a trigger word. Logical sum is taken also between channels.

(8) Setting filter (Filter)
The TR4751 is provided with the capability for filtering a trigger word to identify a longer trigger word. The value in "Filter" is specified with the number of clocks and must be within 1 to 15.

(9) Setting the number of repeating trigger words (Event)
The TR4751 is equipped with an event counter to perform a trigger for the number of times a trigger word is repeated. "Event" must be within 1 to 32769.

(10) Trigger arming from pattern generator (Link module)
The trigger condition defined on the "Trig spec" screen is armed from the pattern generator. When the "/TRG" command described in the pattern program is executed, an arming signal is sent from the pattern generator to the acquisition section, then the trigger circuit of the acquisition section is set to the enable mode to receive a trigger word. After this, the following selection items appear on the screen:

OFF PAT exit

(11) Setting clock qualifier (Clock qualifier)
The TR4751 is provided with clock qualifier inputs for every probe pod. These clock qualifier inputs can be used employing a logical sum. "Clock qualifier" must be with "1", "0" or "x".

CAUTION

(1) The item whose soft key labels are not shown in the above description displayed as follows:

exit

(2) When the sampling clock is set to 2.5 ns, "Filter:" and "Event:" are set to the values equivalent to twice as much as those specified.

4.2 PATT spec (PATTERN GENERATOR SETTING)

4.2.1 Program (Pattern Program Generation)

The TR4751 pattern program is easy to see because it is described with label symbols, and also is designed so that you can edit it with less effort.

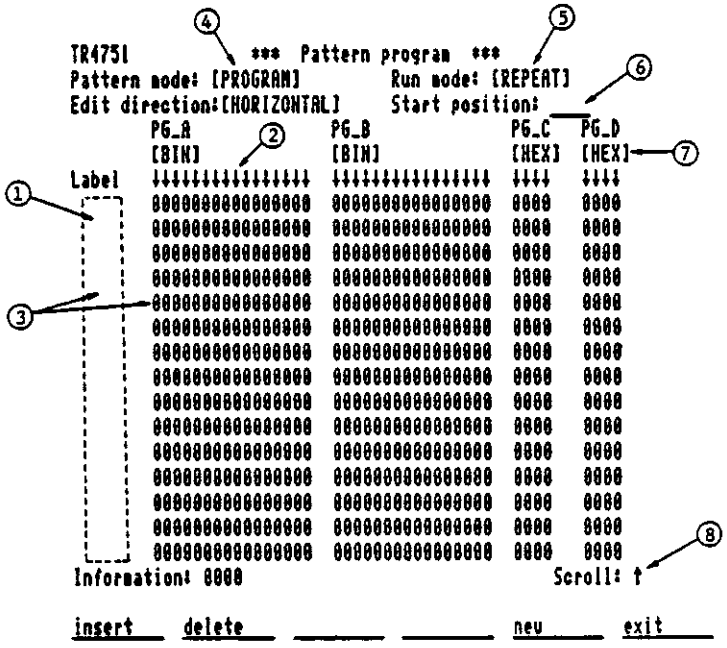


Figure 4-5 "PATT-Program" Screen

(1) Describing label area
The following three words can be described in the label area:

a. Character string having no meaning for pattern control (as a comment)
A character string must consist of up to four alphanumeric characters.
b. Label necessary for pattern control
A label must consist of up to four alphanumeric characters.
c. Command necessary for pattern control
This command, called a label command is divided into the following three commands:

/END : Terminates program. If /END is omitted, the program is executed up to the end. (/END is unnecessary if a loop is specified in program by the control command.)
/TRG : The trigger condition for the data acquisition is enabled when the line containing /TRG is executed. /TRG is disabled if "link module:" is set to OFF on the "Trig spec" screen.

CAUTION

When the pattern mode is set to [MEMORY], the line containing /TRG is ignored and the execution of the first line of a pattern program enables the trigger condition of aquisition.

/RDY : The ready signal is sent from the READY terminal on the rear panel and the pattern output stops when the line containing /RDY is executed. The program is executed from the position where the pattern output stops when the advance signal is input to the ADVANCE terminal.

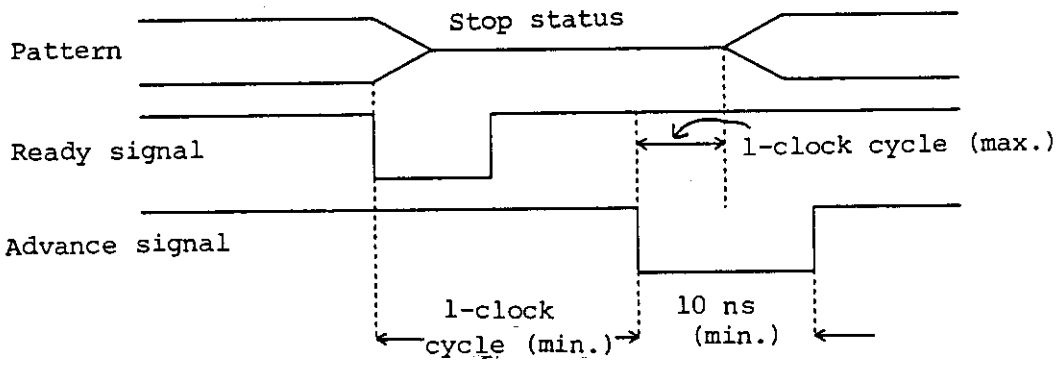


Figure 4-6 Controlling Patterns Using /RDY Command

CAUTION

When the pattern mode is set to [MEMORY], the label command function of /RDY is nullified.
/RDY command is available only when the Run mode is set to [Single].

(2) Describing data area
The pattern data and control commands can be described in the data area.
The pattern data is displayed in the BINARY, OCTAL or HEXADECIMAL format.
The control commands are divided into the following nine commands:

CALL : Shifts a vector to the subroutine specified by the label described on the right of CALL. Up to 16 levels of nesting are allowed.

<Example> CALL ABC

RETURN : Returns control from the specified subroutine to the original routine.

<Example> RETURN

JUMP : Shifts a vector to the pattern specified by the label described on the right of JUMP.

<Example> JUMP XYZ

REPEAT : Repeats outputting patterns for number of times of the value (the number of clocks) described on the right of REPEAT. REPEAT must be within 1 to 1023.

<Example> REPEAT 34

HOLD : Repeats outputting patterns for number of times of the value (the number of clocks) described on the right of HOLD. Patterns are not output for the RZ waveform. HOLD must be within 1 to 1023.

<Example> HOLD 56

INC : Repeats outputting the value of which 1 is added to the pattern for number of times of the value (the number of clocks) described on the right of INC. INC, however, outputs only the described pattern output group. INC operates for the other patterns in the same way as the REPEAT command. INC must be within 1 to 1023.

<Example> INC A120

DEC : Repeats outputting the value of which 1 is subtracted from the pattern for number of times of the value (the number of clocks) described on the right of DEC. DEC, however, outputs only the described pattern output group. INC operates for the other patterns in the same way as the REPEAT command. INC must be within 1 to 1023.

<Example> DEC A45

HALT : Stops the pattern output. The pattern output is restarted by an interrupt signal input.

<Example> HALT

IF : Used along with the other commands. The commands described along with IF are executed when the line containing IF is executed and the IF signal as an input signal for the PG CONT pod is set active. Otherwise, IF is ignored.

<Example> on IF then JUMP XYZ

CAUTION

When the pattern mode is set to [MEMORY], the HALT, IF functions through the control commands are nullified.

(3) Edit function
The system displays the following soft key labels when the cursor exists in the label area:

insert delete exit

insert moves the line pointed by the cursor one line down and inserts a new line there.
delete deletes the inserted line.
The system displays the following soft key labels when the cursor is in the data area:

HALT	INC/DEC	HOLD	IF		
CALL	RETURN	JUMP	REPEAT	exchange	exit

The labels in the lower column are active and effective. exchange changes the active labels by switching the upper and lower columns.

CALL	RETURN	JUMP	REPEAT		
HALT	INC/DEC	HOLD	IF	exchange	exit

A command must be entered directly on the line pointed to by the cursor. A command entry moves the data containing the command one line down. Another command entry moves the cursor to the next entry position.

<Example> JUMP ■
↑ ↑
Set automatically.
The beginning of a command is set to the left end of the data area.

You can select the direction for moving the cursor after data entry in "Edit direction:". The cursor is moved in the following two directions:

HORIZONT	VERTICAL				
(Right)	(Down)				

The cursor moves to the next item displayed without stopping at the area containing data or a numeral when edit keys ⇦ and ⇨ are used.

CAUTION

(1) A line number (memory address) is displayed in "Information:". However, it is not displayed while an error message is appearing on the screen.
(2) Commands cannot be entered on the top line.
(3) Commands cannot be entered over two lines.
(4) Labels cannot be specified on the command lines.
(5) Paging is performed every 14 lines.
(6) Switch INC and DEC by pressing the INC/DEC key. INC is always specified first.

INC A200 ⇨ DEC ⇨ INC

(7) exit specifies the "PATT-spec" menu screen.

(4) Selecting execution format
The TR4751 is supported with two execution formats. These formats must be selected according to measuring condition.

PROGRAM MEMORY _____ _____ _____ exit

PROGRAM converts the commands for the pattern program generated by editor to micro codes and stores these codes into micro code memory. Patterns are controlled when the micro code sequencer compiles the converted codes. This is why control can be branched by using external control signals (IF signal or interrupt signal).

MEMORY stores the commands for the pattern program in memory on the pattern generator by re-expanding them to bit patterns with the micro processor, then executes these commands. This permits high-speed operation. This MEMORY mode is automatically entered when the clock is 10 ns.

(5) Selecting output format (Run mode)
The TR4751 is provided with the following three output formats:

REPEAT SINGLE STEP _____ _____ exit

REPEAT repeats generating the same pattern string by linking the beginning of the pattern program to its end.

SINGLE terminates the pattern program at its last line.

STEP outputs a pattern each time the PATT-RUN key on the front panel is pressed. Pressing the STOP key returns execution to the first line.

(6) Setting pattern output starting position
The pattern output starting position is set by entering the label described in the line to be executed in "Start position:". The label entry must be up to four alphabetic and numeric characters. When no label is entered, patterns are output from the first line (Its memory address is 0000.).

(7) Selecting display format
Data can be described in the following three formats:

BIN OCT HEX exit

(8) Scrolling data
Data is scrolled and paged by using the scroll and page keys on the front panel, respectively. The scrolling and paging directions can be indicated by arrow signs with "Scroll:".

4.2.2 Timing (Timing Definitions and Output Channel Assignment)

The "Timing" screen is used to define timings and assign them to the corresponding channels, then to specify to which channel the data in the program is output.

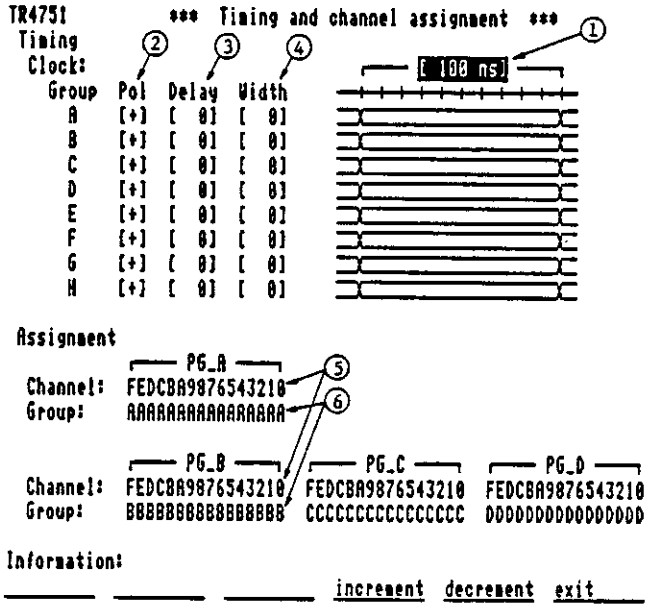


Figure 4-7 "PATT-Timing" Screen

(1) Setting data output speed (rate) (Clock)
The data rate is set by using the following keys:

_____ _____ _____ increment decrement exit

The data rate can be changed within 10 ns to 50 ms and in 1-2-5 step. When the decrement key is pressed with 10-ns data rate, [EXT⌐], [EXT⌐] and [SPECIAL] appear on the screen and patterns are output at the speed of the external clock.
[SPECIAL] enables measuring the delay speed and width. The output rate is set 1/10 time as much as the external clock.
[EXT⌐] and [EXT⌐] disable setting of the delay speed and width.
The delay speed and width must be changed as follows:

Table 4-1 Relationship between Clocks and its Delay and Width

100 ns -	Delay + Width < = 9
50 ns	Delay + Width < = 4
20 ns	Delay + Width < = 1
10 ns (EXT)	Delay + Width = 0

(2) Selecting logical polarity (Pol)
A polarity is selected by using the following keys:

positive negative _____ _____ _____ exit

positive displays a plus sign (+) and outputs the NRZ pattern as described in the program ("1" is output at high level; "0" at low level), or outputs the RZ pattern by using positive pulses.
negative displays a minus sign (-) and outputs the NRZ pattern in the reverse logic described in the program ("1" is output at low level; "0" at high level), or outputs the RZ pattern by using negative pulses.

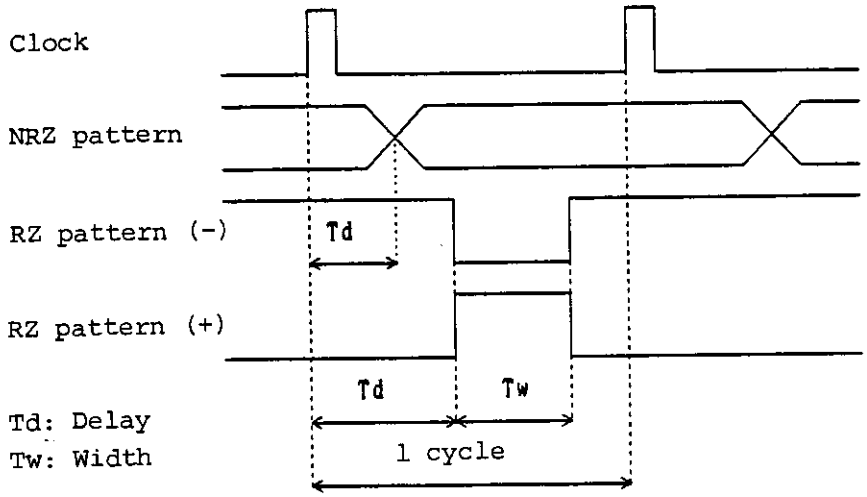


Figure 4-8 NRZ and RZ Patterns

(3) Setting delay time (Delay)
When the delay time is set by using the following keys, the resolution is set to the larger value of 1/10-clock or 10-ns clock:

increment decrement exit

The delay time must be set within the range given in Table 4-1. If the delay time is set out of the range, "Out of range!" appears in "Information:" on the screen.
Each time the key is pressed, its right waveform changes correspondingly.

(4) Setting clock width (Width)
When the clock width is not set to "0", the RZ pattern is defined. Setting procedures and restrictions for the width and its range are the same as for the delay time. See item (3) above.

(5) Assigning output channel (Assignment)
Any channel of the supported 16 channels can be assigned as an output channel. The same channel cannot be assigned more than once. The channel number must be set by using entry keys of 0 thru F.

(6) Selecting and setting timing
The defined timings can be set for each channel. The timing group must be set by using the entry keys of A thru H.

4.2.3 I/O spec (Output Amplitudes, External Control Signals and Strobes Setting)

The "I/O spec" screen is used to select the output voltage source of the pattern generator and sets its amplitude. This screen also defines the input conditions for external control signals, then sets the timing and polarity of strobe output.

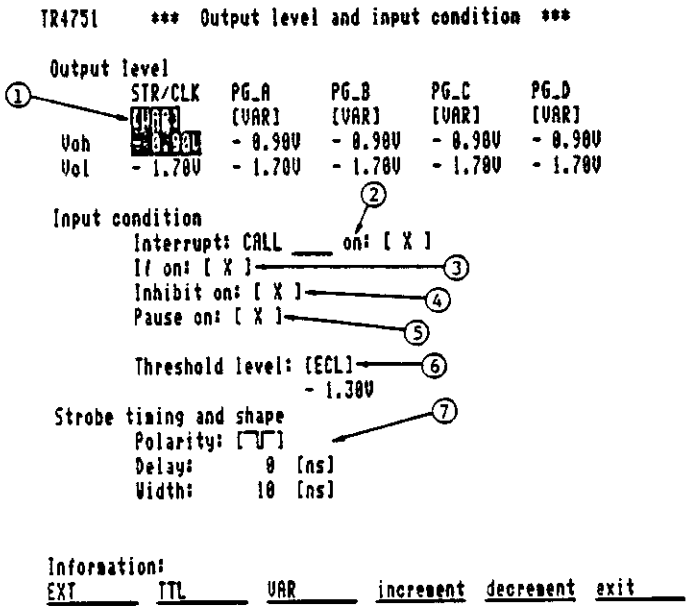


Figure 4-9 "PATT-I/O spec" Screen

(1) Selecting output voltage source and setting its amplitude (Output level)
The output voltage is set by using external and internal voltage sources. The voltage sources are selected by using the following keys:

EXT TTL VAR exit

EXT sets the amplitude according to the voltage applied from the EXT PS connector on the rear panel. The applied voltage must be as follows:

Voh = +6.35 V to -2.00 V, Vol = +2.00 V to -6.35 V, Voh-Vol <= 0.5 V

If the voltage is applied out of the range, "Out of range!" appears in "Information:" on the screen and the values of Voh and Vol are displayed as follows:

```
        [EXT]
Voh     *****
Vol     *****
```

Selecting TTL and VAR specifies an internal voltage source.
Selecting only TTL sets Voh and Vol as follows:

```
        [TTL]
Voh     +2.40 V
Vol     +0.40 V
```

VAR enables setting Voh and Vol to any value. The setting resolution is 50 mV. Voh and Vol cannot be set at the same time. The specified value of the settable voltage of Voh or Vol is displayed in reverse. The voltage is selected by using edit keys, ⇧ and ⇩.

EXT TTL VAR increment decrement exit

STR/CLK on the screen means the strobe and clock output.

(2) Defining interrupt signals and setting jumping address (Interrupt)
An interrupt signal is activated by using "on:". An interrupt signal is selected by moving the cursor with the edit keys and by using the following keys:

positive negative off(X) exit

positive selects [_┌], and interrupts execution when a positive-going edge signal is input.
negative selects [¬└], and interrupts execution when a negative-going edge signal is input.
off(X) selects [x], and disables an interrupt for any input signal.

A jumping address must be entered after CALL within four alphanumeric characters. When the cursor is moved to the left end of the line by using the edit keys and a character is entered, the cursor is automatically moved one space right. If you make an error input, blank the input by using the "x" (don't care) key and reenter the correct characters.

CALL ▮

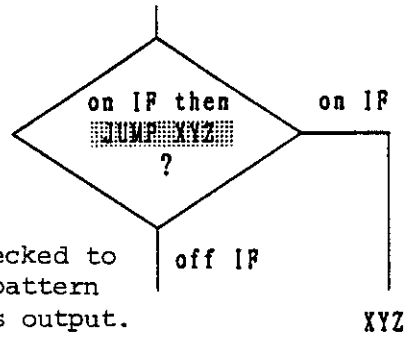
↑

Enter from the right.

(3) Defining IF signal (If on)
The commands following IF are executed when the IF command in a pattern is executed and the level condition for the IF control signal is checked as pass (matching the defined logical level). The IF signal is set in "If on:" as described in item (1). The IF command checks not the signal edge but the signal level.

positive negative off(X) exit

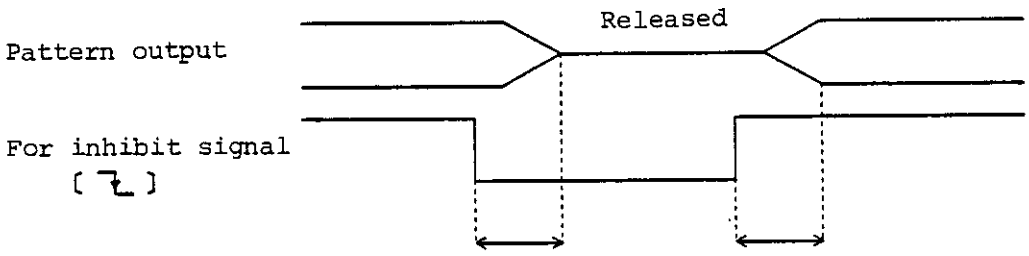
<Example>



• When IF is checked to be true, the pattern labeled XYZ is output.

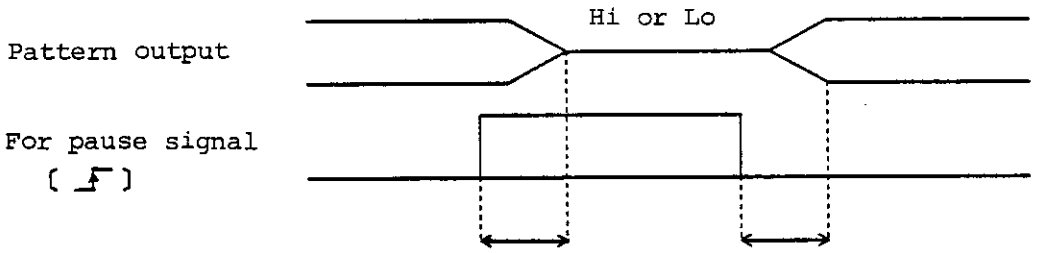
(4) Defining inhibit signals (Inhibit on)
An inhibit signal releases all channels for pattern outputs (setting all channels to the high-impedance mode). Setting procedures for inhibit signals is the same as described in item (3).

positive negative off(X) exit



(5) Defining pause signals (Pause on)
Setting a pause signal terminates (not releases) patterns, and resetting it restarts to output patterns. Setting procedures for pause signals is the same as described in step (3).

positive | negative | off(X) | | | exit



(6) Setting threshold voltage (Threshold level)
"Threshold level" sets all threshold voltages of external control signals to the same level. The voltage level is selected by using the following keys:

TTL | ECL | VAR | | | exit

TTL sets the level to +1.40 V.
ECL sets the level to -1.30 V.
VAR changes the soft keys labels as shown below and sets the level to any value. The setting resolution is 100 mV. The threshold voltage must be within the range of +12.7 V to -12.7 V.

TTL | ECL | VAR | increment | decrement | exit

(7) Setting strobe signals (Strobe timing and shape)
Strobe signals are divided into the systems used for external outputs and those used for sampling the internal acquisition. The strobe signals in the latter system depends not on the polarity and width only but on the delay time.
The polarity is selected by using the following keys:

positive | negative | | | | exit

positive selects [⎍] and issues positive strobe pulses.
negative selects [⎓] and issues negative strobe pulses.

The delay time and width must be set by using numerals and units. Up to three numerals are entered by using the numeric entry keys. Also decimal points are available. The units are entered by using the soft keys. The selectable units are as follows:

ns | µs | ms | | | exit

CAUTION

(1) The delay time and width can be set in four digits without any decimal point. The upper three digits are effective.
(2) The delay time must be set within the range of 0 ns to 10 ms. The setting resolution is 1 ns.
(3) The width must be set within the range of 10 ns to 10 ms. The setting resolution is 10 ns.
(4) The delay time or width can be set less than for each resolution, but these will be ignored at execution.

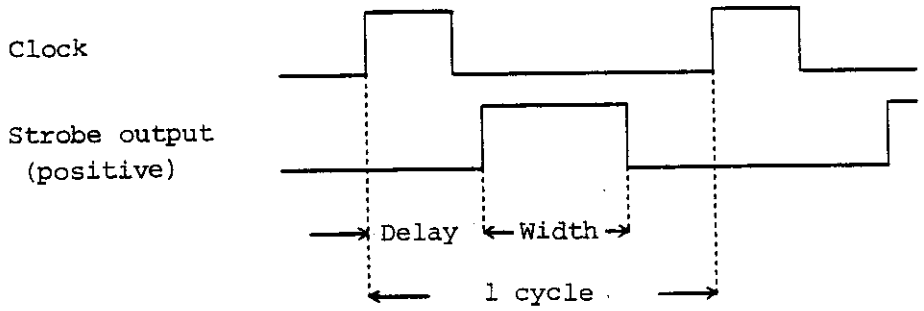


Figure 4-10 Relationship between Clocks and Strobes

4.3 DISPLAY (RESULT DISPLAYING SPECIFICATIONS)

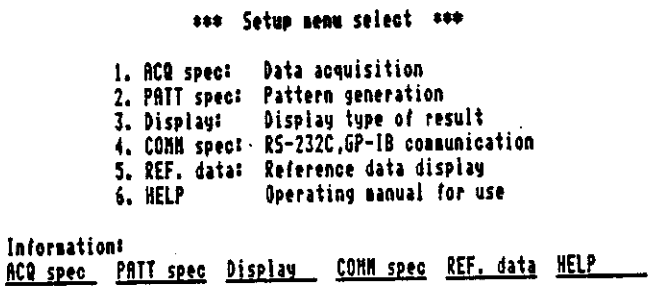
You must specify in advance in which format the acquired data is to be displayed. Data can be displayed in five formats.

When you select Display on the main setup menu screen shown in Figure 4-11 (a), the screen turns as shown in Figure 4-11 (b).

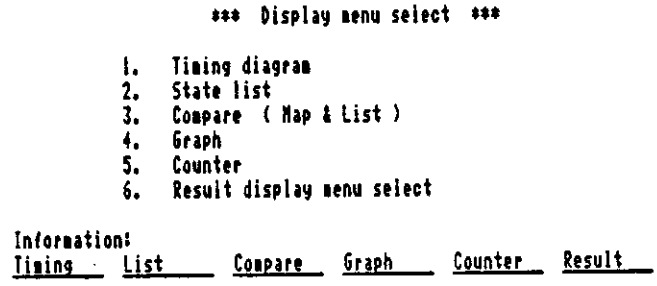
Timing | List | Compare | Graph | Counter | Result

Pressing the Result key displays the screen as shown in Figure 4-11 (c). This screen prompts you to select the format for displaying results. Results are displayed in the format selected on the third screen.

If keys other than the Result key are pressed, the data currently being captured is displayed in the format directly specified.

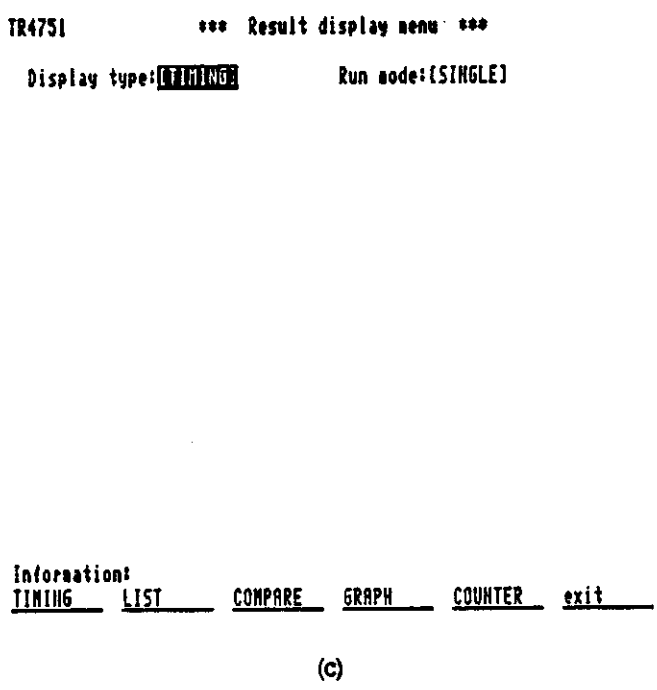


(a)



(b)

Figure 4-11 Result Display Menu Screen



(c)

Figure 4-11 Result Display Menu Screen (Cont'd)

4.3.1 Timing Display (Timing Chart Format Display)

Selecting Timing turns the screen as shown in Figure 4-12.

(1) Only the operating format is selected; the single mode or repeat mode.

SINGLE REPEAT _____ _____ _____ exit

SINGLE executes capturing and displaying data only once when the SYSTEM-RUN or ACQ-RUN key is pressed.
REPEAT repeats capturing and displaying data when the SYSTEM-RUN or ACQ-RUN key is pressed. This operation is stopped by using the STOP key.

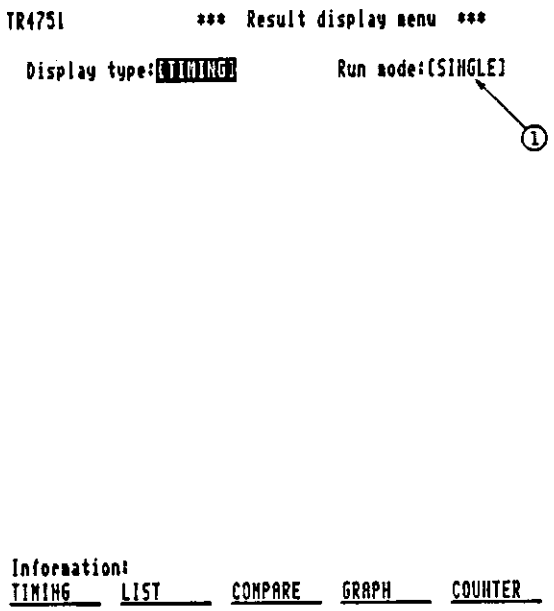


Figure 4-12 Result Display Menu TIMING Screen

4.3.2 List Display (List Format Display)

Selecting LIST turns the screen as shown in Figure 4-13.

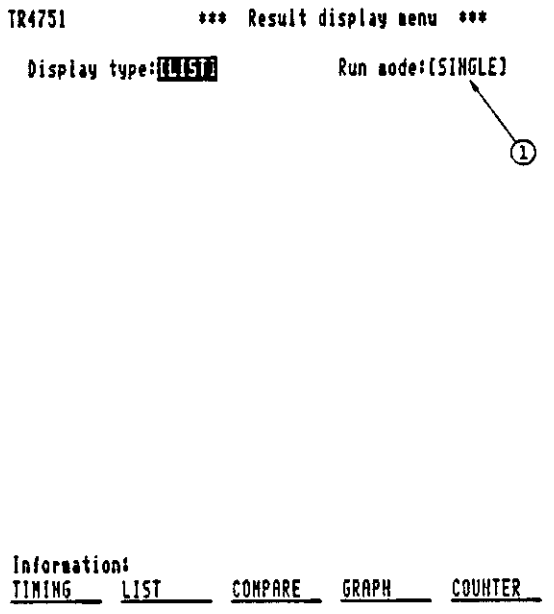


Figure 4-13 Result Display Menu LIST Screen

(1) Only operation format can be selected as same as timing display.

SINGLE REPEAT _____ _____ _____ exit

SINGLE executes capturing and displaying data only once when the SYSTEM-RUN or ACQ-RUN key is pressed.
REPEAT repeats capturing and displaying data when the SYSTEM-RUN or ACQ-RUN key is pressed. This operation is stopped by using the STOP key.

4.3.3 Comparison Display (Comparison of Current Data with the Reference Data and Displaying its Result)

The compared result is displayed in the list format displaying the mismatch data in reverse or in the map format.

Selecting COMPARE turns the screen as shown in Figure 4-14.

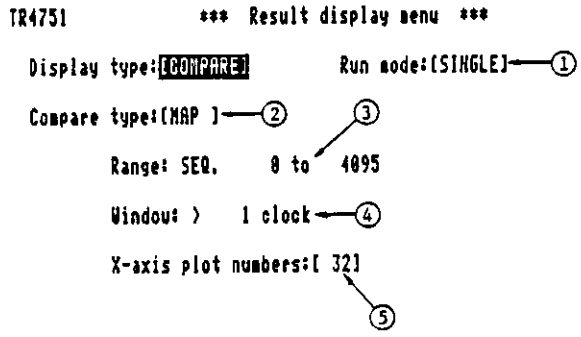


Figure 4-14 Result Display Menu COMPARE Screen

(1) Selecting operation format

SINGLE REPEAT UNTIL= UNTIL ≠ _____ exit

SINGLE executes capturing and displaying data only once when the SYSTEM-RUN or ACQ-RUN key is pressed.

REPEAT repeats capturing and displaying data when the SYSTEM-RUN or ACQ-RUN key is pressed. This operation is stopped by using the STOP key.
UNTIL= repeats capturing and displaying data until all the compared results match.
UNTIL ≠ repeats capturing and displaying data until 1- or more bit mismatch occurs in the compared results.

(2) Selecting display format

MAP LIST _____ _____ _____ exit

MAP displays the compared results in the map format. The captured data is displayed from the upper left to the right, then to the next line sequentially for the number of times specified in "X-axis plot numbers:". Using the map format enables you to see all compared results at one time.
LIST displays the captured data in the list format. In this format, the mismatch data is displayed in reverse. Thus, you can check when and in which channel a mismatch occurs.

(3) Setting comparing range
The comparing range is set by using the sequence numbers of the captured data. The channels not to be compared are specified using the reference data screen.
The comparing range must be set within the range of "0 to memory size" by using the entry keys.

(4) Capacity for the number of mismatch data
The system can ignore comparing operations on the continuous numbers in the mismatch data. For example, data mismatches as shown in Figure 4-15 even if the same data is captured with the asynchronous sampling (sampling caused by internal clocks). This is because the measured data and clocks are not in a synchronous relationship. Consequently, data is not checked as the same. "Window:" filters the mismatch numbers and adds some ambiguity to the comparing operations.

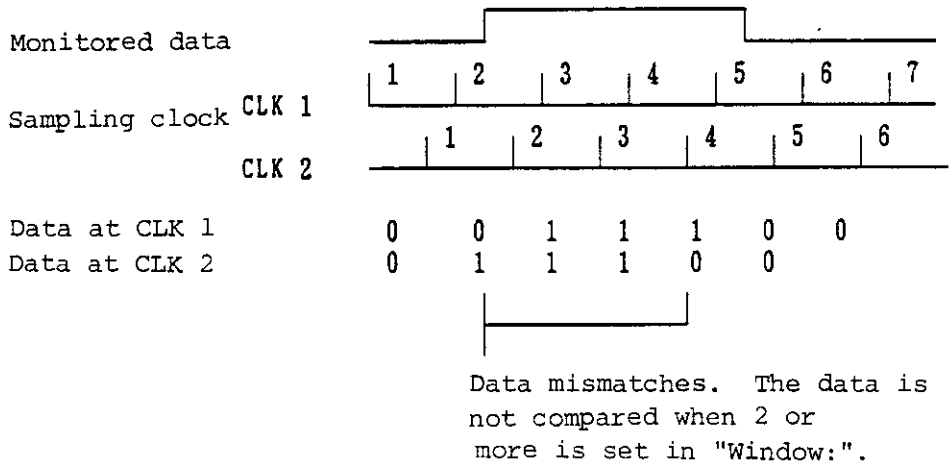


Figure 4-15 Mismatch of Captured Data and Window

(5) Setting the number of data at the horizontal axis in the map format
The number of the data at the horizontal axis in the map format can be set to any of the following four numbers.

32 64 128 256 exit

4.3.4 Graph Display (Comparison Processing with Parameter Change)

The graph display function repeats generating patterns, capturing data, comparing it with the reference data and displaying the results for the specified number, changing some parameters automatically (Shmoo plot).

Selecting GRAPH turns the screen as shown in Figure 4-16.

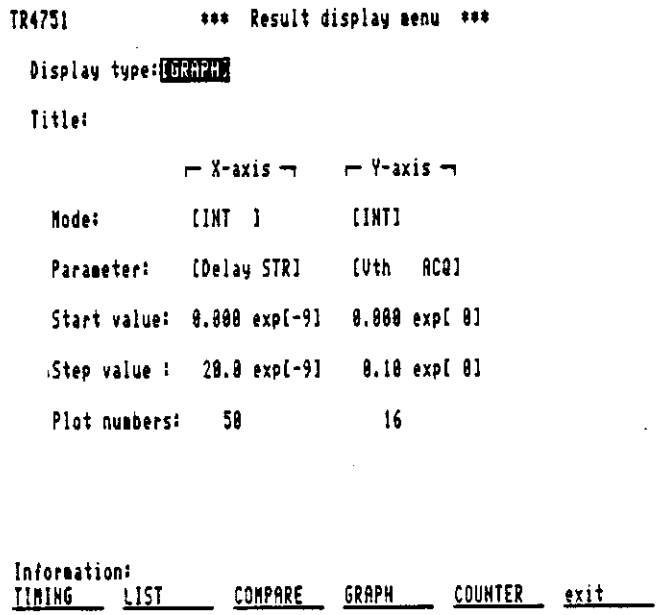


Figure 4-16 Result Display Menu GRAPH Screen

(1) Titling graph
A graph can be titled with up to 40 alphanumeric characters by using the entry keys. A title is displayed on the result display screen.

(2) Selecting parameters
Parameters are divided into internal and external parameters. Internal parameters are processed automatically. Values for external parameters are displayed on the result display screen just as comments. Internal or external parameters are selected by using the BASIC program on a GPIB controller (an internal controller or a controller connected externally).

INT GPIB _____ _____ _____ exit

INT selects internal parameters.
GPIB selects external parameters.

(3) Setting parameters
The following internal parameters can be selected:

Delay STR Voh PG Vol PG Vth ACQ _____ exit

Delay STR sets the strobe delay time as a parameter.
Voh PG sets the high voltage of the pattern generator as a parameter.
Vol PG sets the low voltage of the pattern generator as a parameter.
Vth ACQ sets the threshold voltage of the acquisition section as a parameter.

(4) Setting initial value
The initial mantissa and exponent of each parameter must be set at the start of a measurement. The mantissa must consist of a four-digit numeral. The exponent can be specified in up to nine powers. Both values are set by using the entry keys.

(5) Setting step value
Step values must be set in the same way as the initial mantissa and exponent. For the range, see each parameter specification.

(6) Setting the number of plots
The number of plots can be set individually for the horizontal and vertical axes.
Up to 50 plots can be set on the horizontal axis.
Up to 16 plots can be set on the vertical axis.

4.3.5 Setting Counter Display

The TR4751 can be used as a multichannel counter also. It is used to measure how long the captured data is set to each logic and obtain its maximum, minimum and mean values, then to display each logical cycle and frequency.

Selecting Counter turns the screen as shown in Figure 4-17.

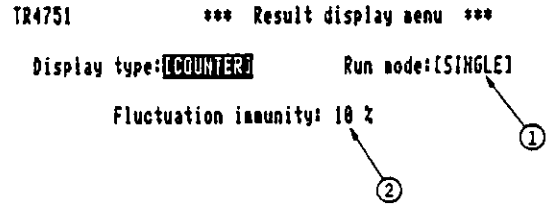


Figure 4-17 Result Display Menu COUNTER Screen

(1) Selecting operation format
Operation formats are as follows:

SINGLE REPEAT _____ _____ _____ exit

(2) Setting range for conversion between cycle and frequency
The range must be within 0% and 50%. When the captured data changes out of the specified range, the conversion between cycle and frequency is not performed and the following indication appears on the screen:

- - - - ns
- - - - clks

The range is set by using the entry keys.

4.4 GENERATING AND EDITING REFERENCE DATA

The reference data is generated by replacing the captured data with the reference data with the store key. The generated reference data can be directly changed.

Pressing the SETUP key and selecting REF data turns the screen as shown in Figure 4-18.

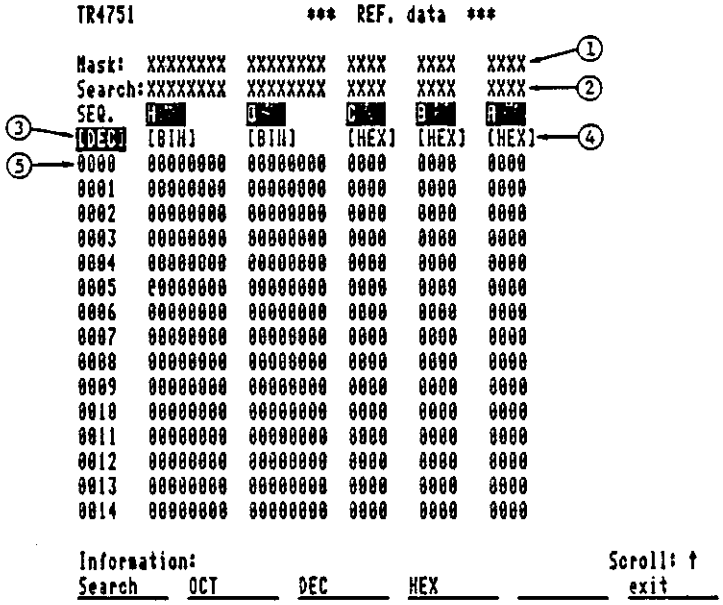


Figure 4-18 REF Data Screen

(1) Mask:
Selects the channel ignoring comparing operations. Using "Compare Range:" along with "Mask:" enables comparing the specific area.

(2) Search:
Describes the words for searching a specific word in data.

(3) Selecting sequence number and display format
The sequence numbers are displayed in the following three formats:

Search OCT DEC HEX exit

(4) Selecting data format
The data format is divided into the following three formats:

Search BIN OCT HEX exit

(5) Changing sequence number
The sequence numbers can be changed by moving the cursor to the beginning of the sequence number to be changed and using the entry keys.

(6) Editing
Use the edit keys to move the cursor to anywhere in data and change the data directly.

4.5 COMM spec (THE COMMUNICATION SECTION SETTING)

The TR4751 is provided with two interfaces, GPIB and RS-232C. This section explains how to set these interfaces.

(1) Press the SETUP key to turn the screen to the Setup menu select mode.

(2) Select COMM spec to set the screen to the Communication menu mode.

4.5.1 Setting GPIB

The TR4751 can be controlled through the GPIB interface by moving the cursor to "Interface:" and selecting GPIB.

Setting procedures of the GPIB parameters are given below according to Figure 4-19.

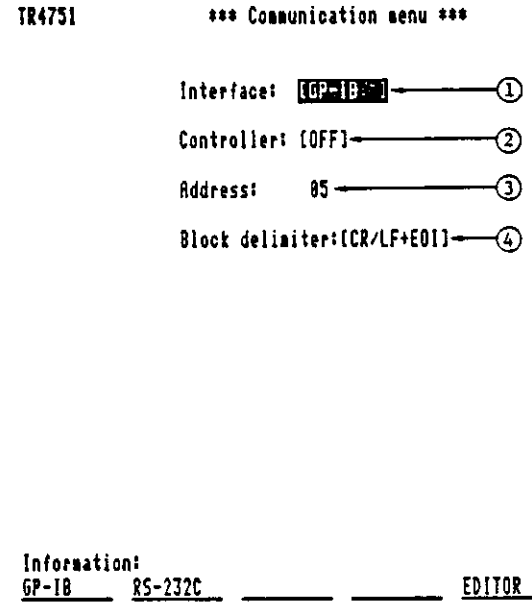


Figure 4-19 GPIB Parameters Setting Screen

(1) Selecting controller (Controller)
"Controller" selects the system controller.

ON OFF EDITOR exit

ON sets the TR4751 as the system controller, and OFF sets the TR4751 as non-system controller. In the non-system controller mode, an external controller is required.

(2) Setting address (Address)
"Address" sets a GPIB address within the range of 00 to 30.
The addresses are set by using the entry keys.
Pressing ON and selecting EDITOR sets the editor screen for the TR4751 GPIB controller program. For editor, see Section 7.2 "GPIB Controller".

(3) Selecting block delimiters (Block delimiter)
The following four items can be selected as delimiters:

CR/LF+EOI LF EOI CR/LF EDITOR exit

4.5.2 Setting RS-232C

The TR4751 can be controlled through the RS-232C interface by moving the cursor to "Interface:" and selecting RS-232C.

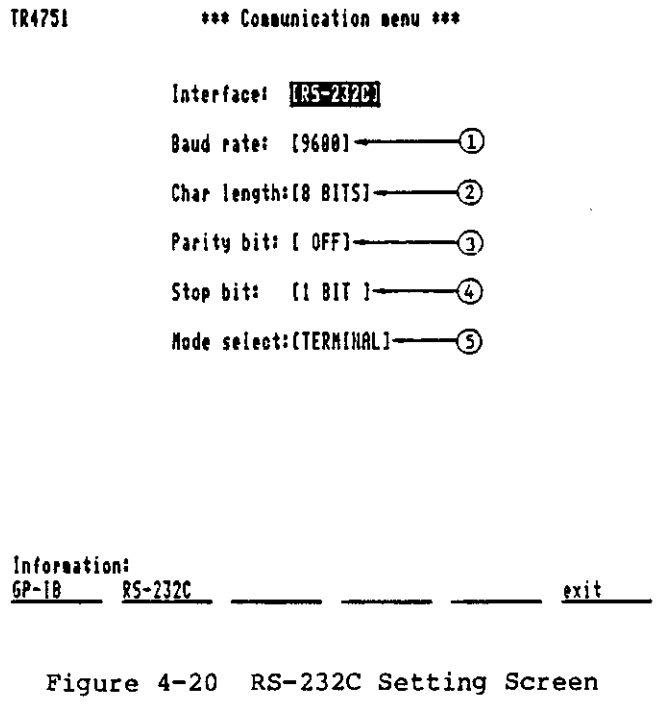


Figure 4-20 RS-232C Setting Screen

(1) Selecting baud rate (Baud rate)
The following numbers can be selected as a baud rate:

9600 4800 2400 1200 600 exit

(2) Selecting character length (Char length)
"Char length" selects the bit length between the start/stop bits.

8 BITS 7 BITS _____ _____ _____ exit

(3) Selecting parity checking format (Parity bit)
The following items can be selected as a parity bit:

OFF EVEN ODD _____ _____ exit

(4) Selecting stop bit (Stop bit)
The stop bit can be selected from the following items:

1 BIT 2 BITS _____ _____ _____ exit

(5) Selecting mode (Mode select)
The communicating mode is divided into two types: TERMINAL and HOST. TERMINAL is set to transfer data to the TR4751 and HOST is set to transfer data from the TR4751.

TERMINAL HOST _____ _____ _____ exit

See Chapter 8 for connecting the RS-232C equipment to the TR4751 and detailed description of RS-232C.

5. READING AND OPERATING RESULT DISPLAY SCREEN

The result display screen appears automatically when a measurement (acquisition) is completed. This screen can be directly selected on the Display menu select screen.

5.1 TIMING DISPLAY

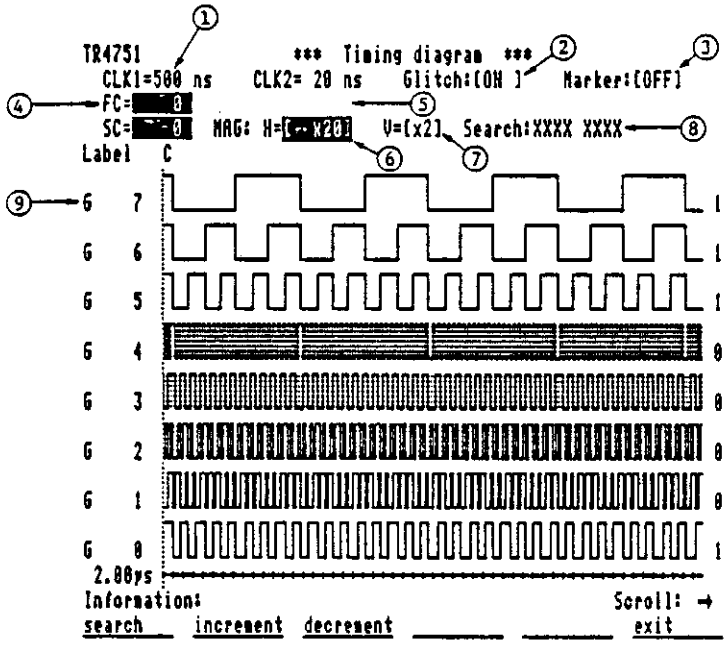
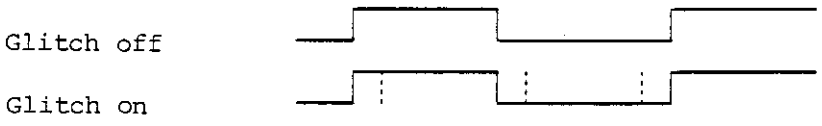


Figure 5-1 Timing Screen

(1) CLK 1=
Displays the sampling cycle for the data currently being displayed. The value specified in Section 4.1.2 (1) is displayed.

(2) Glitch:
Sets ON/OFF of the glitch display.

search on off exit



(3) Marker:
Activates the marker and measures the interval (time) between the cursor and a marker.

search ___ on _____ off _____ _____ _____ exit _____

Selecting on _____ activates the marker over the cursor and displays "C-M=" on the screen. The interval (time) between the cursor and the marker is displayed in "C-M=" by moving the cursor with the scroll keys on the front panel.
Note that the cursor cannot be moved at that time.
Selecting off _____ deletes the marker and "C-M=".

(4) FC=
Displays the interval (clock cycles) between the trigger point and the cursor. The internal value changes whenever the cursor is moved by using the scroll keys on the front panel. The change rate depends on the extension rate.

(5) T-C=
Displays the interval (time) between the trigger point and the cursor. The time is displayed only when the internal sampling clock is selected.

(6) MAG: H=
Sets the horizontal magnification by using the following keys:

search ___ increment decrement _____ _____ exit _____

Data is displayed when X1 is selected. The magnification must be set within the range of 1 to 10000 and in 1-2-5 step. Incrementing the value magnifies the display.

(7) MAG: V=
Sets the vertical magnification by selecting X1 or X2. When X1 is selected, 16 channels are displayed. 8 channels are displayed with X2 selected.

(8) search:
Searches the specific words made up of the displayed channels. "search:" must be 1, 0 or X. (X means a "don't care".)
The right bit of "search:" represents the upper channel for timing display.

(9) Changing display sequence
The display sequence can be changed by using the following keys after moving the cursor with the edit keys on the front panel to the label of the channel to be changed. The sequence indications are deleted by setting the label off.

search ___ preview ___ next _____ _____ _____ exit _____

preview ___ repeats scrolling data up; next _____ scrolls data down. At the time, the "OFF" label is inserted between the top and bottom data.

5.2 LIST DISPLAY

Data is displayed corresponding to the assignments and labels defined in "Chnl spec" given in Section 4.1.1.

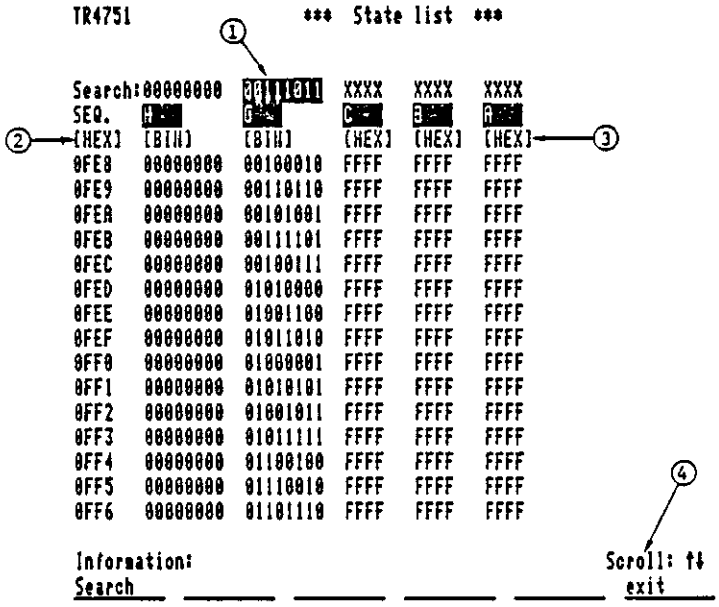


Figure 5-2 State List Screen

(1) Search:
Describes the data to search the specific words. "Search:" is executed by using the search soft key. Data must be described according to the data format given below.

(2) Selecting sequence number and display format
The following three types of numbering is selected with the measurement data:

search | OCT | DEC | HEX | | exit

OCT displays data in octal.
DEC displays data in decimal.
HEX displays data in hexadecimal.

(3) Selecting data format
The following formats can be selected for displaying data:

search | BIN | OCT | HEX | | exit

(4) Scrolling and paging data
Data is searched by using the scroll and page keys on the front panel. Data also can be searched directly through its number by moving the cursor to the data number.

5.3 COMPARED RESULT DISPLAY

Compared results are displayed in the list or map format. The map format is used to check the entire data. The list format is used to check the specific channel in a certain time.

5.3.1 Map Display

You can check the entire data at a glance by displaying compared results in the map format.

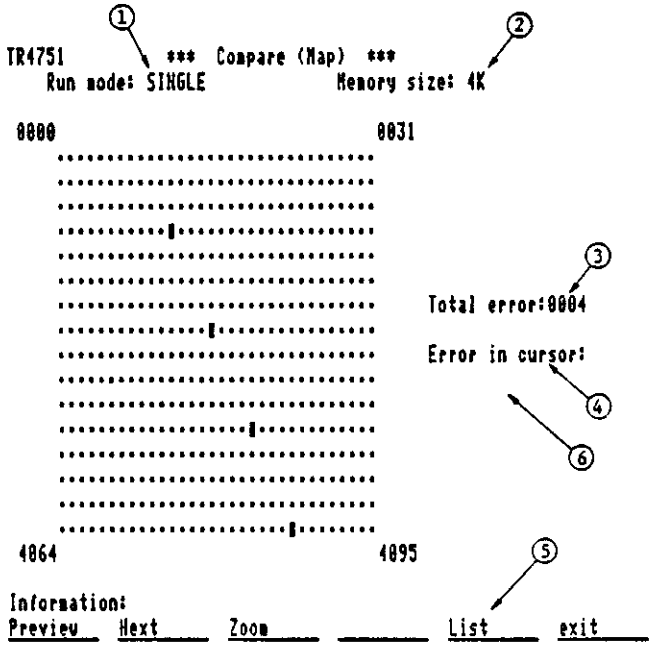


Figure 5-3 Compare MAP Screen

(1) Run mode:
Displays the operation format as specified in Section 4.3.3 (1).

(2) memory size:
Displays the size of the memory currently being used (displaying the value specified in Section 4.1.2 (2)).

(3) Total error:
Displays the number of mismatch words.

(4) Error in cursor:
Displays the sequence numbers of the mismatch words in the area pointed by the cursor.

(5) The soft key labels are displayed as follows:

Preview Next Zoom List exit
(With "Compare" selected)

Preview Next Compress List exit
(With "Zoom" selected)

Preview and Next move the cursor to the mismatch words (.). Zoom magnifies the data display. When Zoom is displayed as a soft key label, one dot contains eight (or 16) words. When Compress is displayed, one dot contains only one word. With Zoom selected, the screen can be scrolled up and down, or right and left by using the scroll keys. To switch the screen scrolling of right/left and up/down, use the middle key of the scroll keys.

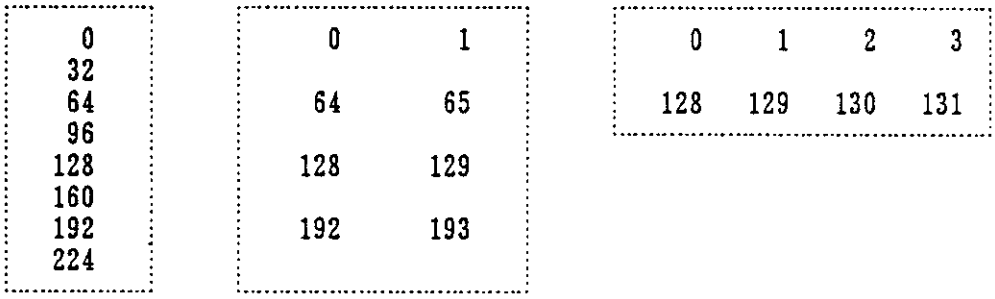


Figure 5-4 Sequence Numbers Corresponding to Dots (at address 0000 in the screen)

Selecting List displays data in the list format.

(6) Pass count: and Fail count:
Displays the operation format when the parameters other than "SINGLE". One count is added to "Fail count:" when 1- or more bit mismatch occur while the checking is repeated. When all bits match, one count is added to "Pass count:".

5.3.2 List Display

To check the entire data at a glance, use the map format. To check channels (bits) in detail, use the list format. Mismatch bits are displayed inversely in the list format.

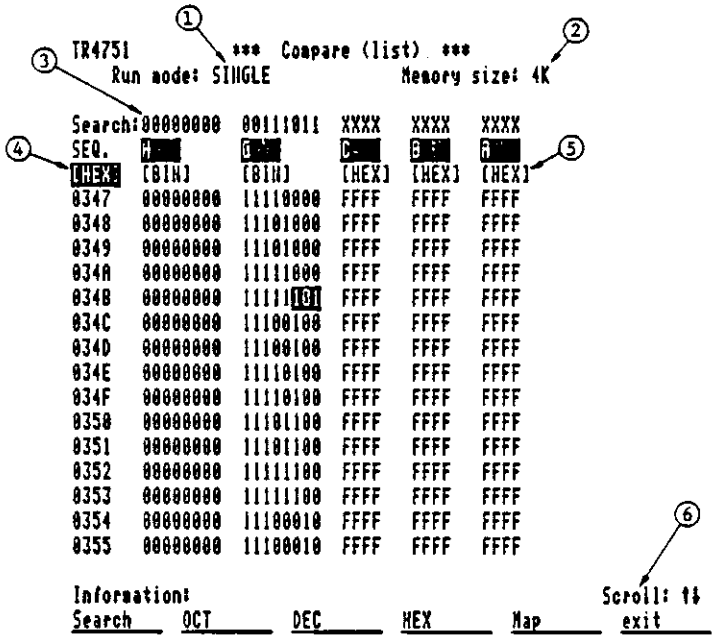


Figure 5-5 Compare List Screen

(1) Run mode:
Displays the operation format as specified in Section 4.3.3 (1).

(2) Memory size:
Displays the value specified in Section 4.1.2 (2).

(3) Search:
Describes the data to search the specific words. "Search:" is executed by using the Search soft key.

(4) Display format for sequence numbers
Sequence numbers are displayed in the following three formats:

OCT DEC HEX exit

(5) Selecting data display format
The following formats can be selected for data display:

BIN OCT HEX exit

(6) Scrolling and paging data
Data can be scrolled up and down by using the scroll and page keys on the front panel. At that time, some effective keys are displayed in "Scroll:". A sequence number can be changed directly by moving the cursor to the top of the number.

5.4 GRAPH DISPLAY

The graph display is used to check whether all words match the reference data when the results of comparing words are displayed. This display is also used to display the results in a graph whenever parameters are changed.

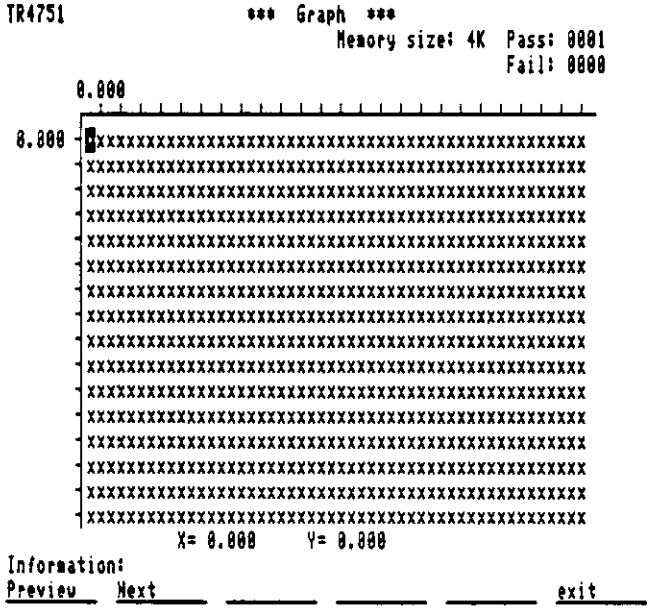


Figure 5-6 Graph Screen

(1) Memory size:
Displays the value specified in Section 4.1.2 (2).

(2) Displaying titles
The title described in Section 4.3.4 (1) is displayed.

(3) Initial value
The initial values on the horizontal and vertical axes specified in Section 4.1.2 are displayed.

(4) Parameter names
The parameter names on the horizontal and vertical axes specified in Section 4.1.2 are displayed.

(5) Parameters pointed to by the cursor
The parameters on the horizontal (X) and vertical (Y) axes pointed to by the cursor are displayed in "X=" and "Y=", respectively.

(6) Pass: and Fail:
Displays each number of match and mismatch points displayed.

(7) Moving the cursor
The cursor can be moved to anywhere on the screen by using the edit keys. The cursor also can be moved to mismatch points by using the following soft keys:

preview	next				exit

preview moves the cursor left and (or) up.
next moves the cursor right and (or) down.

5.5 COUNTER DISPLAY

The logical change of the acquired data is displayed in time, cycle and frequency.

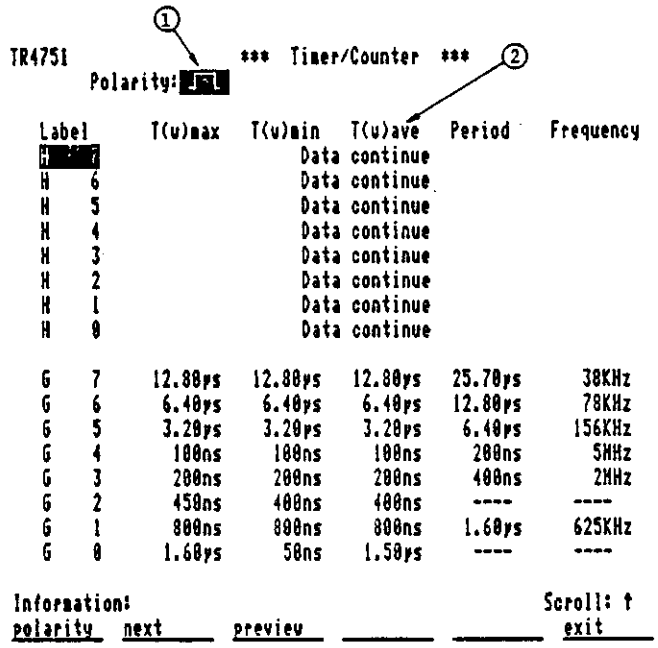


Figure 5-7 Counter Screen

(1) Selecting logical level to be measured
The following soft keys are used to select the logical level to be measured and displayed:

polarity next preview exit

Pressing the polarity key switches the level to high [⎍] or low [⎎].

(2) Display items
T(w)max : Displays the maximum logical level specified in the acquired data.
T(w)min : Displays the minimum logical level specified in the acquired data.
T(w)ave : Displays the average logical level specified in the acquired data.
Period : Displays the average cycle time in the acquired data.
Frequency: Displays the average frequency in the acquired data.

"Period" and "Frequency" display nothing when the cycle time and frequency change out of the range.

6. OPERATING FLOPPY DISK

The TR4751 is equipped with 3.5-inch micro floppy disk drive as standard.

The TR4751 system program is stored in the system disk as described in Chapter 2. To activate the TR4751, use the system disk (see Section 2.1 "Activating the TR4751").

6.1 DISK TYPES

The TR4751 is supplied with two accessory disks; a system disk and a user disk. The user disk is used to store the user created programs, setting parameters or measurement data.

6.2 FILE TYPES

The TR4751 manages data files, sorting them into the following types:

AQD : Acquired data
AQP : Acquisition parameters
REF : Reference data
DIS : Parameters to display results
PGU : Pattern generator programs/parameters
COM : Communication parameters
ALL : Data/parameters/programs in one system

6.3 OPERATING FLOPPY DISK

Pressing the FLOPPY-EXE key displays the Floppy disk operation screen as shown in Figures 6-1 and 6-2.

(1) Disk operations
"Command:" selects a disk operation. Pressing the FLOPPY-EXE key executes the command specified in "Command:".

DELETE	RENAME	FORMAT	STATUS		
DIRECTORY	GET	SAVE	COPY	exchange	exit

The following eight commands are provided for disk operations:

DELETE	:	Deletes the file in the specified drive.
RENAME	:	Renames a file.
FORMAT	:	Initializes a new disk.
STATUS	:	Displays a disk name and the data stored in it.
DIRECTORY	:	Lists the files stored in a disk.
GET	:	Loads the data in a file to the system.
SAVE	:	Saves the data in the system into a disk.
COPY	:	Copies the files in a disk to the specified disk.

(2) Selecting drive
The TR4751 can be connected to an external drive unit (option).
Specify the TR4751 internal drive as F0: and an external drive as F1:.

F0:	F1:				exit

(3) File:
Defines a file name of up to six alphanumeric characters.

(4) Type:
Selects the data type by using the soft key as described in Section 6.2.

AQP	DIS	CDM			
AQD	REF	PGU	ALL	exchange	exit

(5) Disk:
Specifies the disk type at disk format. The disks are divided into the following two types:

SYSTEM	USER				exit

Selecting SYSTEM generates the system disk.

(6) Disk name:
Names a disk at disk format. A disk name must be up to 20 characters.

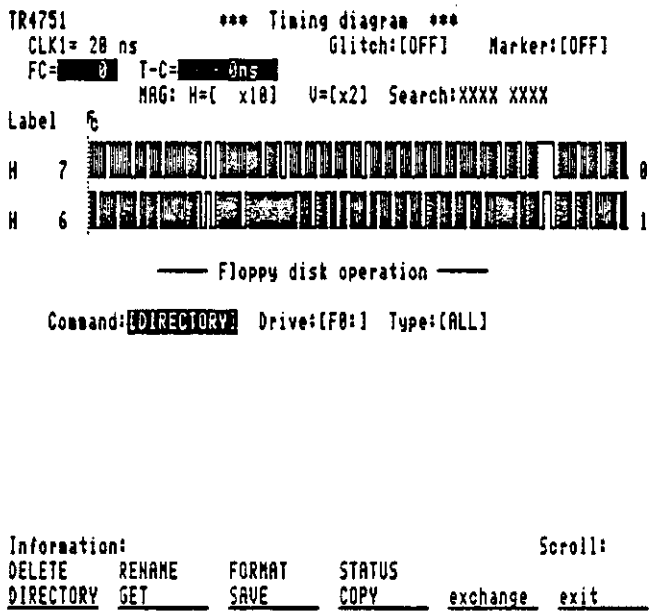


Figure 6-1 Floppy Disk Operating Screen (1)

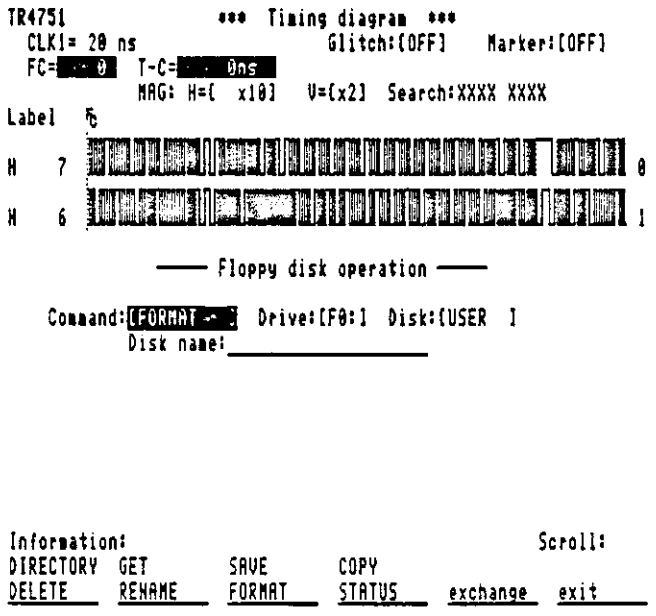


Figure 6-2 Floppy Disk Operating Screen (2)

7. GPIB INTERFACE

7.1 GPIB CONNECTIONS AND PROGRAMMING

7.1.1 Introduction

The TR4751 Logic Analysis System can be connected to IEEE standard 488-1978 measuring bus GPIB* equipment via the GPIB interface provided as a standard feature.

The GPIB interface specifications and functions are described in this chapter.

* GPIB: General Purpose Interface Bus

7.1.2 Outline of the GPIB

The GPIB is an interface system which can be connected by simple cable (bus line) between measuring equipment and controller or other peripheral devices.

The GPIB is far more versatile and easier to use than the more conventional interfacing methods and includes a greater expansion capacity. And since different GPIB equipped brands are electrically, mechanically, and functionally compatible with each other, highly functional automatic measuring systems can be formed from a more simple system by using only a single bus cable.

The first thing to do in GPIB systems is to set the individual addresses of each component device connected to the bus line. Each of these devices may have one or more controller, talker, or listener roles.

During system operation, only one talker can send data to the bus line at any one time, but any number of listeners can receive that data.

The controller specifies the talker and listener addresses, transfers data from talker to listener, and also sets measuring conditions in the listener from the controller itself (acting in this case as a talker).

Data transfer between devices employs eight "bit-parallel byte-serial" format data lines. Bidirectional transfers are executed in asynchronous mode. And since it is an asynchronous system, high-speed and low-speed devices can be connected together in any desired combination.

The data (messages) transferred between devices includes measured data and measuring conditions (programs), and various commands in ASCII code.

In addition to the above eight data lines, the GPIB also includes three handshake lines which control the transfer of asynchronous data between devices, and five control lines which control the information flow on the bus.

- The following signals are used in the handshake lines:
  - DAV (Data Valid) : Signal indicating data vadlidity
  - NRFD (Not Ready for Data) : Signal indicating that any further dasta cadnnot be received
  - NDAC (Not Data Accepted) : Signal indicating that present data reception has not been completed
- The following signals are used in the control lines:
  - ATN (Attention) : Signal used to distinguish whether the data line signal is an address or command, or some other data
  - IFC (Interface Clear) : Signal for clearing the interface system
  - EOI (End or Identify) : Signal used when data trasnsfer is completed
  - SRQ (Service Request) : Signal used to make a request from any device for service from the controller
  - REN (Remote Enable) : Signal used to place the remote programmable device under remote control

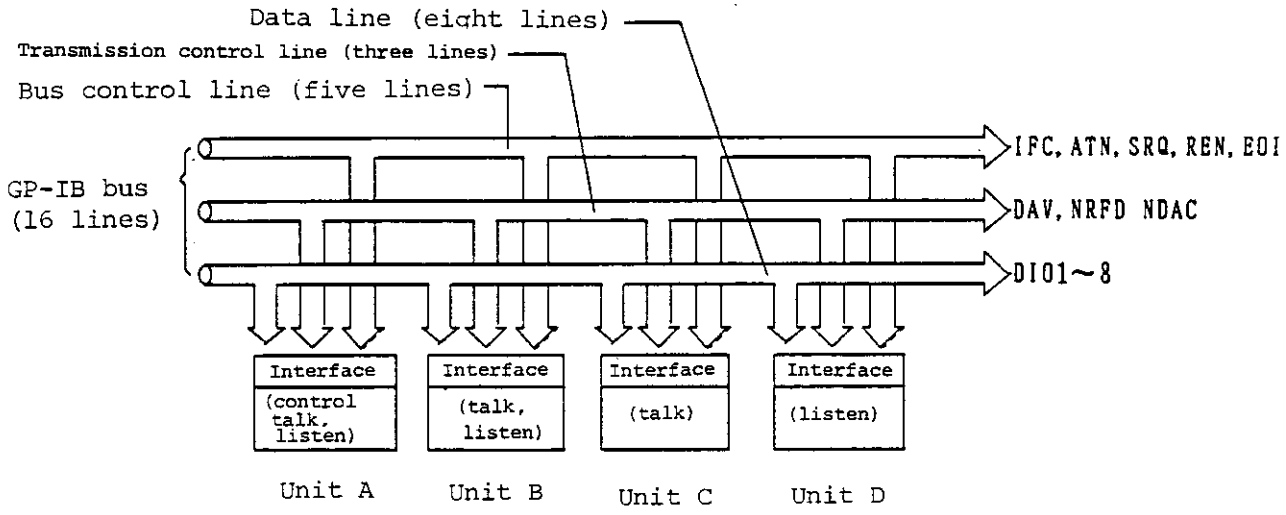


Figure 7-1 GPIB Outline

7.1.3 Specifications

(1) GPIB specifications

Basic standards	:	IEEE standard 488-1978
Codes employed	:	ASCII code, but binary code for packed format.
Logic levels	:	Logic 0 "High" status +2.4 V min. Logic 1 "Low" status +0.4 V max.
Signal line termination	:	The 16 bus lines are terminated as indicated in the following diagram.

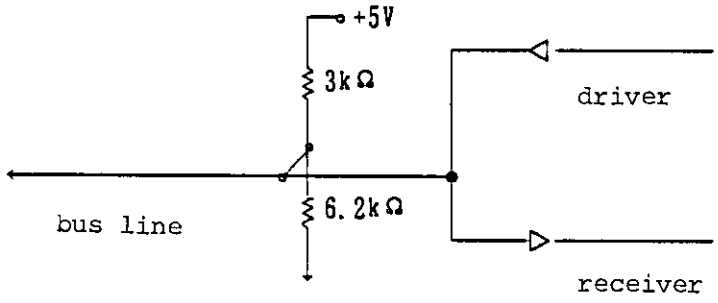
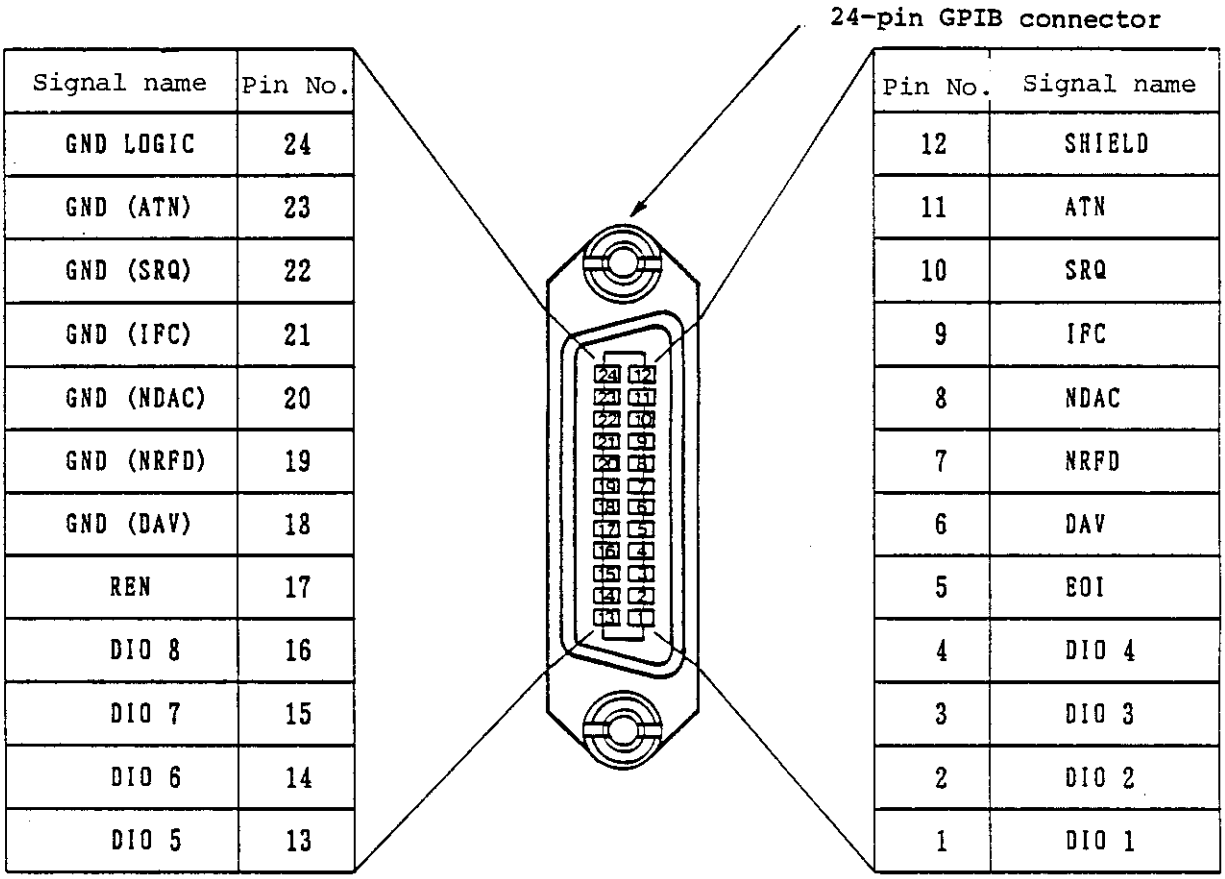


Figure 7-2 Signal Line Termination

Driver specifications	:	Open collector
"Low" status output voltage	:	+0.4 V max, 48 mA
"High" status output voltage	:	+2.4 V min, -5.2 mA
Receiver specifications	:	"Low" status at below +0.6 V "High" status at above +2.0 V
Bus cable length	:	The total bus cable length should not exceed (number of devices connected to the bus) x 2 m, nor should it exceed 20 m.
Address assignment	:	31 separate talker and listener addresses can be set by the rear panel address selector switch. Press the PRESET key after setting the address selector switch.
Connectors	:	24-pin GPIB connector 57-20240-D35A (Amphenol or equivalent)



Signal name	Pin No.
GND LOGIC	24
GND (ATN)	23
GND (SRQ)	22
GND (IFC)	21
GND (NDAC)	20
GND (NRFD)	19
GND (DAV)	18
REN	17
DIO 8	16
DIO 7	15
DIO 6	14
DIO 5	13

Pin No.	Signal name
12	SHIELD
11	ATN
10	SRQ
9	IFC
8	NDAC
7	NRFD
6	DAV
5	EOI
4	DIO 4
3	DIO 3
2	DIO 2
1	DIO 1

Figure 7-3 GPIB Connector Pin Configuration

(2) Interface functions

Table 7-1 Interface Functions

Code	Function and description
SH1	Source handshake capability
AH1	Acceptor handshake capability
T6	Basic talker, serial poll, and unaddressed to talk if addressed to listen
L4	Basic listener and unaddressed to listen if addressed to talk
SR1	Service request capability
RL1	Remote/local capability
PP0	No parallel poll capability
DC0	No device clear capability
DT0	No device trigger capability
E1	Open collector bus driver is used. Note, however, that EOI and DAV are E2 (use of tri-state bus driver).
C1	System controller capability
C2	IFC transmission and "controller-in-charge" capability
C3	SRQ response capability
C12	Interface message transmission and control transfer capability

7.1.4 GPIB Handling Procedures

(1) Connections to component devices
Since GPIB systems consist of a number of devices, attention is drawn to the following precautions when setting up the system.

(a) Refer to the TR4751, controller, and peripheral device instruction manuals to check the condition (preparation) and operation of each device before making any connections.
(b) Connecting cables for the measuring equipment and the bus cables for connecting the controller should not be any longer than necessary. The total bus cable length should be less than (number of devices connected to the bus) x 2 m, and not in excess of 20 m. The standard bus cables available from ADVANTEST are listed below.

Table 7-2 Standard Bus Cables (Optional)

Length	Code name
0.5 m	408JE-1P5
1 m	408JE-101
2 m	408JE-102
4 m	408JE-104

(c) The bus cable connectors are "piggy-back" types with separate plug and socket for each connector to enable stacking connections. Triple and other multiple stacking connections, however, are not recommended. And always secure the connections with the connector securing screws.

(d) Always check the power and grounding requirements, and if necessary, the setting conditions of each component device before switching the power on for the respective device.
The power for all devices connected to the bus must be switched on. If the power for any of the devices is left off, normal operation for the entire system cannot be guaranteed.

(2) Setting GPIB address
Set the GPIB address after "Address:" on the Communication menu screen in decimals by using the entry keys. Table 7-3 lists address codes.

Table 7-3 Address Code

ASCII address codes		Decimal equivalent
LISTEN	TALK	
SP	@	0
!	A	1
"	B	2
#	C	3
$	D	4
%	E	5
&	F	6
'	G	7
(	H	8
)	I	9
*	J	10
+	K	11
'	L	12
-	M	13
.	N	14
/	O	15

ASCII address codes		Decimal equivalent
LISTEN	TALK	
0	P	16
1	Q	17
2	R	18
3	S	19
4	T	20
5	U	21
6	V	22
7	W	23
8	X	24
9	Y	25
:	Z	26
;	[	27
<	\	28
=	]	29
>	∿	30

(3) Setting block delimiters
Match the block delimiters for the connected devices to those for the TR4751 transfer data. Otherwise, data may not be transferred. The TR4751 can receive any type of block delimiters.
Select a block delimiter from the following soft key labels and set it in "Block delimiter:" on the Communication menu screen:

CR/LF+EOI LF EOI CR/LF EDITOR exit

EDITOR sets the controller to the editor mode. exit returns control to the Main setup menu screen. The above four types of block delimiters can be selected.

(4) Selecting built-in controller
TR4751 enables selecting an internal or external controller. Select ON for "Controller:" on the Communication menu screen. This enables to control the external devices connected to the GPIB interface by using the built-in controller. The control is performed by the program described in the unique BASIC language through the editor function. (Selecting EDITOR specifies the editor screen.) Selecting OFF for "Controller:" places the TR4751 to be controlled by using an external controller.

7.1.5 Programming

On the TR4751, most parameters are specified through the GPIB controller. This section explains the commands to specify the parameters corresponding to each screen.

(1) Parameters on "Chnl. spec" screen

(a) Setting threshold voltage

TL<P>0	..... Selects TTL.
1	..... Selects ECL.
2	..... Selects VAR.
LV<p><s><n.nn> unit Volt	..... Enables specifying any voltage (VAR is selected.)

<p> specifies a probe pod name.
<s> specifies a polarity, + (positive) or - (negative).
<n.nn> specifies a voltage. In the H/G pods, ±6.35 V is set (the resolution is 50 mV). In the A/B/C pods, ±12.7 V is set (the resolution is 100 mV).

(b) Defining labels

LN<n><name>	..... Specifies a label.

<n> specifies 0 through 4 labels on the screen.
<name> must be four alphanumeric characters.

(c) Specifying logical polarity of the group labeled

LP<n>0	..... Positive
LP<n>1	..... Negative

<n> specifies 0 through 4 labels on the screen.

(2) Parameters on "Trigger spec" screen

(a) Sampling clock sources

FAST side	SLOW side	
FS0	SS0	..... External clock
FS1	SS1	..... Pattern generator strobe
FS2	SS2	..... Internal clock

(b) Selecting memory size

FAST side	SLOW side	
FM0	SM0	..... 4/8K
FM1	SM1	..... 1/2K

(c) Selecting external clock polarity

FAST side	SLOW side	
FX<nn>	SX<nnn>	..... Selects the external clock polarity.

Set 0: positive edge, 1: negative edge, and 2: off in <nnn> in the order shown on the screen.

(d) Selecting trigger mode

FAST side	SLOW side	
FT0	ST0	..... Match trigger
FT1	ST1	..... Release trigger

(e) Selecting trigger position

FAST side	SLOW side	
FP0	SP0	..... Begin trigger
FP1	SP1	..... Center trigger
FP2	SP2	..... End trigger
FP3	SP3	..... Delay trigger

(f) Setting trigger delay

FAST side	SLOW side
FY<s><nnnnn>	SY<s><nnnnn>

<s> must be + (positive) or - (negative).
<nnnnn> specifies the delay value.

(g) Setting trigger word

FAST side	SLOW side
FW<p><aa .. aa>	SW<p><aa .. aa>

<p> specifies a pod name.
<aa .. aa> must be a binary number (0 and 1) or Xs (don't care).

(h) Setting glitch trigger

FG<p><aa .. aa>

<p> specifies a pod name.
In <aa .. aa>, "1" is described for the specified bit and X for the other bits.

(i) Setting enable-1 word
E1<p><aa .. aa>

<p> specifies a pod name.
<aa .. aa> must be a binary number (0 and 1) or Xs (don't care).

(j) Setting enable-2 word
E2<p><aa .. aa>

<p> specifies a pod name.
<aa .. aa> must be a binary number (0 and 1) or Xs (don't care).

(k) Setting disable word
DW<p><aa .. aa>

<p> specifies a pod name.
<aa .. aa> must be a binary number (0 and 1) or Xs (don't care).

(l) Setting trigger filter

FAST side	SLOW side
FF<nn>	SF<nn>

<nn> must be a clock of 1 through 15.

(m) Setting event count

FAST side	SLOW side
FV<nnnnn>	SV<nnnnn>

<nnnnn> specifies the event count value of 1 through 32769.

(n) Setting link module

FAST side	SLOW side	
FL0	SL0	..... Off
FL1	SL1	..... Pattern generator
FL2	SL2	..... Acquisition

(o) Selecting clock qualifier

FAST side	SLOW side
FQ<nn>	SQ<nn>

The array of <nnn> is the same as of the qualifier selecting items on the screen, and must be 0s (positive), 1s (negative) or Xs (don't care).

(p) Selecting sampling rate

FAST side	SLOW side
FR<n . n><aa>	SR<nnn><aa>

<nnn> specifies the clock value in the 1-2-5 sequence.
<aa> specifies an unit (NS=ns, US=µs and MS=ms).

(3) Parameters on "Pattern program" screen

(a) Selecting pattern mode

PM0	..... Program mode
PM1	..... Memory mode

(b) Selecting run mode

RM0	..... Repeat mode
RM1	..... Single mode
RM2	..... Step mode

(c) Selecting edit mode

EM0	..... Horizontal direction
EM1	..... Vertical direction

(d) Specifying starting position

SA<aaaa>

<aaaa> must be up to four alphanumeric characters.

(4) Parameters on "Timing and channel assignment" screen

(a) Selecting clock rate

PC<nnn><aa>

<nnn> specifies the clock value in the 1-2-5 sequence, and must be 10 through 500.
<aa> specifies a unit (NS=ns, US=μs and MS=ms).

(b) Setting timing

TI<a><p><n><n>

<a> specifies a group name of A to H.
<p> specifies a polarity, + or -.
<n> specifies a delay and its width, and must be 0 through 9. n is restricted depending on the clock rate.

(c) Specifying channel array

AS<a><c><g><c> ... <c><g>

<a> specifies the name of a pattern generator output group.
<c> specifies a channel name.
<g> specifies a timing group name.

(5) Parameters on "I/O spec." screen

(a) Setting output level

OL<a>0	..... EXT
OL<a>1	..... TTL
OL<a>2	..... VAR
VL<a><S><n.nn> unit Volt	..... Specifies Vol.
VH<a><S><n.nn> unit Volt	..... Specifies Vol.
VL<a>INC/DEC	..... Increases/decreases Vol of the specified group.
VH<a>INC/DEC	..... Increases/decreases Voh of the specified group.

<a> specifies a pattern generator output group of A to D.
<s> specifies a polarity, + or -.
<n.nn> must be the specified value.

(b) Setting interrupt label

IL<aaaa>

<aaaa> must be four alphanumeric characters.

(c) Setting external input conditions

IN<p>	..... Specifies the interrupt input polarity.
IF<p>	..... Specifies the IF input polarity.
IH<p>	..... Specifies the inhibit input polarity.
PA<p>	..... Specifies the pause input polarity.

<p> specifies a polarity. 0: Positive, 1: Negative, 2: Invalid

(d) Setting threshold voltage for external inputs

TH0	..... TTL
TH1	..... ECL
TH2	..... VAR
TV<s><n.nn> unit Volt	..... Sets the variable data.
TV<s>INC/DEC	..... Increases/decreases by the resolution voltage.

<s> specifies a polarity.
<n.nn> must be the specified value (±12.7 V, its unit is Volt and its resolution is 100 mV.).

(e) Setting strobes

PO<p>	..... Specifies the strobe polarity.
DS<nnnn><uu>	..... Specifies the strobe delay.
WS<nnnn><uu>	..... Specifies the strobe width.

<nnnn> must be a 3-digit number containing a decimal points (the last digit is cut off when 4-digit number is specified.).
<uu> specifies a unit (NS=ns, US=μs and MS=ms).

(6) Parameters on "Result display menu" screen

(a) Selecting display type

DT0	..... Displays a timing.
DT1	..... Displays a list.
DT2	..... Displays a compared result.
DT3	..... Displays a graph.
DT4	..... Displays a timer/counter.

(b) Selecting run mode

DM0	..... Single mode
DM1	..... Repeat mode
DM2	..... Repeat mode (=)
DM3	..... Repeat mode (≠)

(c) Inserting a graph title
GT<aaa .. aaa>

<aaa .. aa> must be up to 40 alphanumeric characters.

(d) Specifying range (for comparing)

CS<nnnn>	..... Specifies the sequence number to start a comparison.
CP<nnnn>	..... Specifies the sequence number to end a comparison.

<nnnn> specifies a sequence number, and must be a decimal number.

(e) Setting a window (for comparing)
CW<nn>

<nn> specifies a window value within the range of 1 through 15.

(f) Specifying a mask channel (for comparing)
MC<n><cc .. cc>

<n> specifies a probe pod name.
<cc .. cc> specifies a compare bit with 1 and a mask bit with X.

(g) Search
SH<p><nn .. nn>

<p> specifies a probe pod name.
<nn .. nn> must be a binary number or Xs (don't care).

(h) Selecting a polarity on timer/counter display

TP0	..... Positive
TP1	..... Negative

(8) Data I/O commands

(a) Acquisition data outputs
OA<p><s>

<p> specifies a probe pod name.
<s> selects an upper/lower channel. Data is output in eight channel groups.
L: 0 to 7 channels H: 8 to F channels

(b) Compared results

OC<p><s>	..... Specifies a compared result.
OCTE	..... Specifies the numbers of mismatch sequences and total errors on the Compare screen.

<p> specifies a probe pod name.
<s> selects an upper/lower channel. Data is output by eight channel groups.
L: 0 to 7 channels H: 8 to F channels

(c) Pattern data inputs

IP<p><s>

<p> specifies name of a pattern generator group.
<s> selects an upper/lower channel. Data is output in eight channel groups.
L: 0 to 7 channels H: 8 to F channels
Input the entire data according to the length of the group data.

(d) Reference data

IR<p><s>

Data is input in the inverted format for data output in the same way as the acquisition data.

(9) Execution commands

SYS-RUN key	ER0, ER	Loads the pattern data.
	ER1	Loads no pattern data.
ACQ-RUN key	EA	
PATT-RUN key	EP0, EP	Loads the pattern data.
	EP1	Loads no pattern data.
STOP key	ES	
STORE key	DR	
EXE key	DF	

7.2 GPIB CONTROLLER

7.2.1 Introduction

The GPIB Controller is designed to control devices connected to the GPIB (General Purpose interface Bus: conforming with IEEE Standard 488-1978) by BASIC language.

And since this controller can be programmed in general BASIC language, and is equipped with a "function and graphic" function as a standard feature, it can also be used as a personal computer.

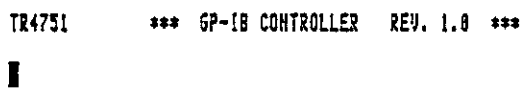
7.2.2 Preparation for GPIB Controller Operations

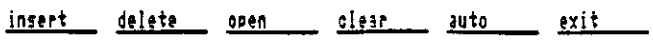
(1) Checking items
Before using the GPIB controller, check the followings.

(a) GPIB address
Check the GPIB address as follows. (See Section 7.1.4-(2))
- The GPIB address is set to "1" at the power-on.
- Check that the current GPIB address is not used for other device. If the GPIB address for the TR4751 is used for other device, normal operations are not guaranteed.
- Check the system controller state. If the TR4751 is in the non-system controller state, an external system controller must be prepared. (Normally, place the TR4751 in the system controller state.)

(b) Interface bus
- If the TR4751 is in the system controller state, check that system controllers such as personal computers are not connected to the GPIB.
  If system controllers are connected to each other on the GPIB bus, an abnormally high current will flow in the GPIB bus and may damage the interface.

(2) Edit mode
When a BASIC program entry is required, the TR4751 must be set to edit mode.
Press the EDITOR key after specifying the "COMM spec." screen and setting GPIB as an interface. This turns the screen as follows:





The TR4751 is thus put into edit mode where entry of programs is possible. Edit mode can be cancelled by pressing the exit key or SETUP key.

(3) Key entry
Programs are entered by operating front panel keys or external keyboard.
After setting the TR4751 to edit mode, press keys A and B. The following display appears on the CRT.

```
TR4751      *** GP-IB CONTROLLER  REV. 1.0 ***

RS
```

└ Cursor(Indicates the position at which the next input character is displayed.)

insert delete open clear auto exit

7.2.3 Fundamentals of Programming

The execution of an actual but simple program is described below to outline the fundamentals of programming in the TR4751.

(1) program deletion
Before executing new programming, the former programs must always be deleted. Execute the "LIST" command to display the list of the previously entered program. Since the TR4751 contains no program when it is powered on, there should be no display when the command is first executed. When this command is executed, the character display disappears, and the cursor moves to the top left hand corner of the CRT. When a program list does appear on the CRT at this time, that program must be deleted.
Press the clear key to initially clear the screen display, then press the N, E, and W panel keys, and eventually press the ENTER key. "NEW" is the command for deleting the previously entered program.

```
NEW
```

Press the ENTER key. Press the ENTER key at the end whenever entering a program or command.

"Delete old text? Yes [Y] or No [any key]" appears on the 22nd line of the CRT when the ENTER key is pressed. The program is actually cleared when the "Y" (Yes) key is then pressed. "Deleted old text" is then displayed instead. The program is not deleted if any other key apart from Y is pressed.

(2) Actual programming
After deleting all of old GPIB controller programs, enter the following program example.

```
10  SCLEAR
20  DISP "HELLO!!"
30  DISP "I am GPIB Controller"
40  DISP "Good-bye!"
50  END
```

The ENTER key must always be pressed at the end of each line. In addition to terminate that line, this operation moves the cursor to the beginning (left end) of the next line.
If an incorrect keystrokes was made, use the BS (Back Space) key to erase the character or to key in again from the beginning of that line. Or if the operator wishes to key in a different character on top of a previously keyed-in character, the former character is deleted and is replaced by the new input character. (See Section 7.2.4 "Program Editing" below for further details on program editing).
After the program has been correctly keyed in, press the GPIB RUN key, or enter the RUN command to execute the program.
When the program is executed, the entire screen display disappears, and is replaced by the following display.

```
TR4751          ***  GP-IB CONTROLLER   REV. 1.0  ***

HELLO !!
I am GPIB CONTROLLER
Good-bye !
```

Then enter the LIST command again to display the program list.

```
10 SCLEAR
20 DISP "HELLO !!"
30 DISP "I am GPIB CONTROLLER"
40 DISP "Good-bye !"
50 END
```

The numbers at the beginning of each line are called line numbers, and must always be keyed when generating a program.
When a program runs, it is executed in sequence from the lower numbered lines. Therefore, an error in the line number may result in the program being executed in the wrong sequence. Line numbers are non-negative integers from 1 thru 32767. Line numbers only need to be in proper sequence, the intervals between numbers does not matter.
The character string following the line number is the instruction sent to the GPIB controller, and is generally called the statement. The statements in the above example are;

SCLEAR : Clear characters displayed on the CRT.
DISP : Display the specified characters or numerical letters on the CRT.
END : End of program.

(For a more detailed description of statements, see Section 7.3 "Command and Statement Syntax and Interpretation").
These line numbers and statements are always written according to specific syntaxial rules. Any infringement of these rules results in the output of an error message, and interruption of the processing. For example, if line 30 in the above example is rewritten as

```
30 WRITE ("I am GPIB CONTROLLER")
```

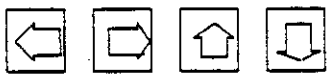
where proper syntax has not been used, an attempt to execute the program results in the generation of an error, and output of the following error message.

```
Syntax error 30
```

7.2.4 Prograsm Editing

A number of different editing functions used to generaste programs are available in edit mode. The keys used in program editing are described below together with the basic editing procedures.

(1) Cursor movement
As a rule, programming is executed in respect to the position of the blinking cursor.

The cursor can be shifted by using  keys.

(a) EDIT  : Single-character shift of the cursor to the right. When the cursor is already in the next position to the right of the end of the statement, however, this key cannot shift it any further. And if the cursor is already at the right hand end of the line, it will only proceed to the next line if the statement on that line has been continued onto the next line. If the statement is not continued, the cursor stops at the right hand end of that line. If EDIT  is pressed when the cursor is in the following position,

```
20 DISP "ADVANTEST CORPORATION"
```

it is shifted to the neighboring position.

```
20 DISP "ADVANTEST CORPORATION"
```

EDIT  : Single-character shift of the cursor to the left. If the cursor is already at the left hand end of the line, it will only shift back to the previous line if the statement on that line has been continued from the previous line. If the statement has not been continued, the cursor stops at the left hand end of that line. If EDIT  is pressed when the cursor is in the following position,

```
20 DISP "ADVANTEST CORPORATION"
```

it is shifted to the neighboring position.

```
20 DISP "ADVANTEST CORPORATION"
```

EDIT  : Shift of cursor to the end of the statement on the previous line. If it is already on the top line, it is shifted to the end of the bottom line.

```
10 SCLEAR
20 DISP "ADVANTEST CORPORATION"
30 BEEP
```

If EDIT  is pressed in this case, the cursor moves like this

```
10 SCLEAR
20 DISP "ADVANTEST CORPORATION"
30 BEEP
```

EDIT  : Shift of cursor to the end of the statement on the next line. If it is already on the bottom line, it is shifted to the end of the top line.

```
10 SCLEAR
20 DISP "ADVANTEST CORPORATION"
30 BEEP
```

If EDIT  is pressed in this case, the cursor moves like this

```
10 SCLEAR
20 DISP "ADVANTEST CORPORATION"
30 BEEP
```

(2) Character insertion

The insert key is used to insert characters into statements on lines which have already been programmed, or on lines which are about to be programmed.

When this key is pressed, all characters to the right of the position where the cursor is blinking on and off are shifted by one character to the right, and a space is generated in the cursor position. A separate character can be inserted into that space.

```
20 DISP "ADVANTEST CORPORATION"
```

Pressing the insert key in this case results in

```
20 DISP "ADVANTEST CORPORATION"
```

(3) Character deletion

The delete key is used to delete characters from statements on lines which have already been programmed, or on lines which are about to be programmed.
When this key is pressed, the character at the position where the cursor is blinking on and off is deleted, and all characters to the right of that position are shifted by one character to the left.

```
20 DISP "ADVANTEST CORPORATION"
```

Pressing the delete key in this case results in

```
20 DISP "ADVANTEST CORPORATION"
```

(4) Clearing the CRT display
The clear key is used to clear the display from the CRT. When this key is pressed, all lines below and including the line where the cursor is blinking on and off are erased from the screen.
The CRT can display up to 19 lines at any one time. If the ENTER key is pressed after completing 19 lines of programming, the entire screen is scrolled up by one line, leaving the last line blank again. Programming can then be continued on that blank line. And if the cursor is shifted to the top line, the entire current display can be cleared by pressing the clear key.
And when the program list is displayed, this key can also be used to clear the display after the display contents have been checked.

```
20 DISP "ADVANTEST CORPORATION"
30 BEEP
40 WAIT 1000
50 OUTPUT 0:"TLH2"
60 OUTPUT 0:"LVH-1.25"
70 OUTPUT 0:"TLG2"
80 OUTPUT 0:"LVG-1.35"
100 OUTPUT 0:"OLA2"
110 OUTPUT 0:"VHA-0.60"
120 OUTPUT 0:"VLA-1.80"
```

If the clear key is pressed in this case, the display is cleared in the following manner.

```
20 DISP "ADVANTEST CORPORATION"
30 BEEP
40 WAIT 1000
```

(5) Single line insertions
Programs which have already been completed can be modified by inserting additional lines between existing lines.

(a) When a new line is to be added, the line number for the new line must be an intermediate number between the line numbers of the previous and following lines.
For example, to insert a new line, "A=SIN(A)", between lines 10 and 20 in the following example,

```
10  A=2*PI
20  DISP A
30  END
```

the line number must be a number between 10 and 20 (such as 15). After keying in

```
15  A=SIN(A)
```

press the ENTER key.
The new program list can then be displayed by entering LIST command.

```
10  A=2*PI
15  A=SIN (A)
20  DISP A
30  END
```

As can be seen from this example, it is not necessary to key in the program in the same order as the line numbers. Additional intermediate lines can be inserted after entry of the initial program has been completed.

(b) In more complicated branched programs, however, clearing the display, keying in new lines, and displaying the list again can be very troublesome. And it is also necessary to take into consideration of the interrelations between previous and following lines.
In this kind of case, keystrokes of open can open a blank line between two existing lines.

```
20 DISP "ADVANTEST CORPORATION"
30 BEEP
40 WAIT 1000
50 OUTPUT 0:"TLH2"
60 OUTPUT 0:"LVH-1.25"
70 OUTPUT 0:"TLG2"
80 OUTPUT 0:"LVG-1.35"
100 OUTPUT 0:"OLA2"
110 OUTPUT 0:"VHA-0.60"
120 OUTPUT 0:"VLA-1.80"
150 OUTPUT 0:"P01"
160 OUTPUT 0:"DS20.0NS"
```

If the open key is pressed in this case, the display is altered in the following way.
A new line can then be keyed in directly on that line.

```
20 DISP "ADVANTEST CORPORATION"
30 BEEP
40 WAIT 1000
50 OUTPUT 0:"TLH2"
60 OUTPUT 0:"LVH-1.25"
70 OUTPUT 0:"TLG2"
80 OUTPUT 0:"LVG-1.35"
100 OUTPUT 0:"OLA2"
110 OUTPUT 0:"VHA-0.60"
120 OUTPUT 0:"VLA-1.80"
150 OUTPUT 0:"P01"
```

(6) List output
The LIST command is entered when the operator wishes to display the program list on the CRT.
Since the CRT can only display 19 lines at any one time, only the first 19 lines of the program can be displayed if the program contains more than 19 lines. To display the next part of the list, the list will have to be "scrolled".

(a) SCROLL  : For scrolling the list upwards.

```
10 SCLEAR
20 DISP "ADVANTEST CORPORATION"
30 BEEP
40 WAIT 1000
50 OUTPUT 0:"TLH2"
60 OUTPUT 0:"LVH-1.25"
70 OUTPUT 0:"TLG2"
80 OUTPUT 0:"LVG-1.35"
100 OUTPUT 0:"OLA2"
110 OUTPUT 0:"VHA-0.60"
```

If pressed SCROLL  in this case, the list is scrolled up a line.

```
20 DISP "ADVANTEST CORPORATION"
30 BEEP
40 WAIT 1000
50 OUTPUT 0:"TLH2"
60 OUTPUT 0:"LVH-1.25"
70 OUTPUT 0:"TLG2"
80 OUTPUT 0:"LVG-1.35"
100 OUTPUT 0:"OLA2"
110 OUTPUT 0:"VHA-0.60"
120 OUTPUT 0:"VLA-1.80"
```

(b) SCROLL  : For scrolling the list downwards.

```
30 BEEP
40 WAIT 1000
50 OUTPUT 0:"TLH2"
60 OUTPUT 0:"LVH-1.25"
70 OUTPUT 0:"TLG2"
80 OUTPUT 0:"LVG-1.35"
100 OUTPUT 0:"OLA2"
110 OUTPUT 0:"VHA-0.60"
120 OUTPUT 0:"VLA-1.80"
150 OUTPUT 0:"PO1"
```

If pressed SCROLL  in this case, the list is scrolled down a line.

```
20 DISP "ADVANTEST CORPORATION"
30 BEEP
40 WAIT 1000
50 OUTPUT 0:"TLH2"
60 OUTPUT 0:"LVH-1.25"
70 OUTPUT 0:"TLG2"
80 OUTPUT 0:"LVG-1.35"
100 OUTPUT 0:"OLA2"
110 OUTPUT 0:"VHA-0.60"
120 OUTPUT 0:"VLA-1.80"
```

(c) List display by LIST command

When the LIST command is used, and a specific line number is specified, the program list can be displayed from that line number. The LIST command format is

```
LIST n
```

where "n" is the line number. The list starting from the specified line number is subsequently displayed. If there is no line number corresponding to the number specified by "n", the list is displayed from the next line number nearest to the number specified by "n".
If the ENTER key is pressed after entering

```
LIST 100
```

the list from line 100 is displayed.

```
100 OUTPUT 0:"OLA2"
110 OUTPUT 0:"VHA-0.60"
120 OUTPUT 0:"VLA-1.80"
150 OUTPUT 0:"PO1"
160 OUTPUT 0:"DS20.0NS"
170 OUTPUT 0:"WS10.0NS"
200 OUTPUT 0:"PM1"
210 OUTPUT 0:"RM0"
220 OUTPUT 0:"PC20.0NS"
250 OUTPUT 0:"TIA+25"
260 OUTPUT 0:"TIB-52"
270 OUTPUT 0:"TIC-26"
300 END
```

When this list is displayed, the cursor is positioned at the right hand end of the top line of the CRT.

(7) List output to printer

Connect a printer to the TR4751 as described in Section 7.2.9 "Printer Connection". Use of the PLIST command enables output of the program list to the printer.
Key in

```
PLIST
```

and then press the ENTER key.
The program list is subsequently printed out in the printer irregardless of the current screen display.

(8) Line replacement

If the operator wishes to rewrite a particular existing line, that line is simply keyed in a second time. The previous line is deleted, and is replaced by the new line.

If the operator wishes to revise only a part of an existing line, or add a part of a line, display the program list containing that line, and rewrite the line using the insert and delete keys (with the cursor positioned on that line).

After completing the revision, always leave the cursor on that line and press the ENTER key. Note that the rewriting is not completed until the ENTER key is pressed.

(9) AUTO function

When the AUTO function is used, line numbers are generated automatically by the GPIB controller irrespective of whether they are keyed in by the operator or not. To use the AUTO function, either key in the AUTO command, or press the auto key.

The AUTO command format is

```
AUTO [start][,step]
```

where "start" specifies the starting line number, and "step" the interval between line numbers. For example, to specify line numbers from 100 in steps of 10, key in

```
AUTO 100, 10
```

and then press the ENTER key. These arguments can be omitted, in which case "10" is automatically set for both arguments. And if the auto key is pressed, the resultant operation is the same as if "AUTO 10, 10" was issued.

When the AUTO function is activated, the starting line number appears on the CRT. And after completing the entry of each line, the next line number appears on the CRT automatically to enable entry of the next statement.

```
AUTO 10, 20
10 DIM A (100)
30
```

This AUTO function can be cancelled by pressing the auto key again, and also by pressing the exit, SET UP or GPIB RUN key.

(10) Program deletion

When a new program is to be entered, or when the current program is no longer required, key in the NEW command to delete the old program. If a new program is entered without deleting the old program, lines unrelated to the new program are left behind, resulting in the generation of a bug (an operation not intended in the original program). It is recommended that the NEW command also be executed.

When

```
NEW
```

is keyed and the ENTER key is pressed, a buzzer sound is generated, and the following message appears on the 22nd line of the CRT.

```
Delete old text? Yes [Y] or No [any key]
```

The purpose of this inquiry is to check whether the program is to be deleted or not.
If Y key is then pressed, the previous program is deleted, and the following display is shown accompanied by a buzzer sound.

```
Deleted old text
```

The program is not deleted if any other key apart from Y is pressed.

(11) Multi-statement

In the basic program entry procedure, there written only one statement per line. In TR4751 BASIC programs, however, more than one statement can be entered on the same line by using a colon (:) to partition the statements.
Example:

```
10 A=10:DISP A:END
```

This is called a multi-statement. The following three statements, however, cannot be included in multi-statements.

```
GOTO, END, and REM
```

Note that multi-statements can also be used in direct mode (see Section 7.2.5-(4)).
Example:

```
FOR I=0 TO 100:DISP I:NEXT I
```

7.2.5 Program Execution

(1) Program execution

Programs can be executed by pressing the GPIB RUN key, or by keying in the RUN command. And if the GOTO statement or the GOSUB statement is used in direct mode (see Section 7.2.5-(4)), the program can be started from an intermediate step.
When a program is executed by pressing the GPIB RUN key or keying in the RUN commadnd, all variables are initialized (replaced by 0), and array declarations in numerical arrasy variables and character string variables are cancelled.
If the program is executed directly from an intermediate step by using the GOTO or GOSUB statement, all variables maintain the values set prior to execution. And if a subroutine is executed by using the GOSUB statement, program execution is terminated at the RETURN statement of that subroutine.

(2) Program execution pause

To stop a program temporarily during execution, press the GPIB RUN key. Although this key is used to start program execution normally, it is also used to stop the program temporarily after execution has been commenced. This program pause is accompanied by a buzzer sound, and "PAUSE" is displayed on the CRT.
This pause status is released by pressing the GPIB RUN key again.

(3) Halting program execution
Program execution can be halted by pressing the STOP key. Execution can be started again by pressing GPIB RUN key, but note that execution is started from the beginning again.

(4) Execution by direct mode
If a statement is keyed in without being given a line number, the GPIB controller executes that program immediately. This is called execution by direct mode.
For example, if

```
DISP 3*9
```

is keyed in without a line number, and the ENTER key is then pressed, the GPIB controller calculates "3 x 9" immediately and displays the result "27" on the CRT.
Execution by direct mode is convenient when certain calculations need to be performed, or when the operator intends to check the operation of a statement which is to be entered while generating a program.

7.2.6 Key Words used in TR4751 Basic

Key words are specific BASIC language words used to express commands, statements, and functions. BASIC programs are composed of combinations of these key words.

Key words used in TR4751 BASIC

- Commands
  AUTO, LIST, NEW, PLIST, PRINTER, RUN, SIZE
- Statements
  BEEP, CURSOR, DATA, DIM, DISABLE INTR, DISP, ENABLE INTR, END, FOR-TO-STEP-NEXT, GET, GOSUB, GOTO, IF-GOTO, IF-THEN, INPUT, LET, OFF ERR, OFF KEY, OFF SRQ, ON ERR, ON KEY, ON SRQ, PAUSE, PRINT, PRINTER, READ, REM, RESTORE, SCLEAR, WAIT
- GPIB control statements
  CLEAR, DELIMITER, ENTER, INTERFACE CLEAR, LISTEN BUFFER, LOCAL, LOCAL LOCKOUT, OUTPUT, PASS CONTROL, REMOTE, REQUEST, RESUME, SEND-DATA-CMD-TALK-LISTEN-UNT-UNL, INTERFACE CLEAR, TALK BUFFER, TRIGGER
- Functions
  ABS, CONV, COS, BIT, EXP, ERR, INT, LOG, LN, PI, SGN, SIN, SPOLL, WCONV, SQR, TAN
- Status functions
  STATUS

7.2.7 Constants and Variables

The use of constants and variables is indispensible in numerical and character string calculations. The display formats and operating procedures are described below.

(1) Constants and variables

(a) Constants

There are two types of constants - character constants and numerical constants.

Constants ─┬─ Character constants
           └─ Numerical constants

① Character constants

A character constant is a group of alphanumeric characters and symbols enclosed between quotation marks ("). Note that quotation marks cannot be handled as part of the character constant.

Examples: "ABCDEF $?*"
"21-JUL-83"

② Numerical constants

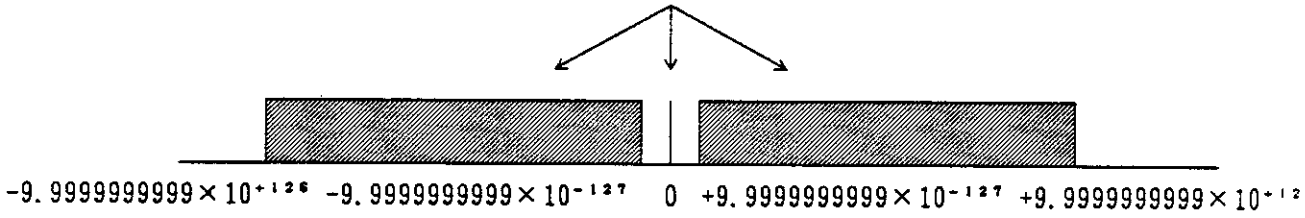
A numerical constant may be a positive or negative real number, or 0. The maximum number of significant digits of the real number is 11, and the exponent can be expressed by E.

Examples: 12345678901
1.2345E6 (equivalent to 1.2345×10^{6})
5.4321E-6 (equivalent to 5.4321×10^{-6})

The ranges of numerical values which can be handled by TR4751 BASIC are

$-9.9999999999 \times 10^{-127}$ to $-9.9999999999 \times 10^{+126}$,
$+9.9999999999 \times 10^{-127}$ to $+9.9999999999 \times 10^{+126}$,
and 0.

The ranges of numerical values which can be handled by TR4751



Although the plus (+) sign can be omitted from positive values, the minus (-) sign cannot be omitted from negative values.

(b) Variables

A variable is a number whose value can be changed during calculations and other processing operations within the BASIC program.

Variables are divided into those which handle numerical values and those which handle character strings. These can be roughly classified into the following three types.

- Variables
  - Character string variables
  - Numerical variables
    - Simple numerical variables
    - Numerical array variables

Variables generally referred to as numerical variables are simple numerical variables.

In BASIC programming, few numerical values and character strings are handled as constants; they are usually calculated after being substituted in variables.

① Character string variables

The names of character string variables are expressed as a single alphabetic character A thru Z, plus a single capital or small alphabetic character, or numerical character with a $ sign appended.

In variable names consisting of more than two characters, however, the third and subsequent characters are disregarded. Also note that character combination identical to BASIC key words cannot be used.

Character string variable name: 1 alphabetic character + 1 small or capital alphabetic or numerical character + $ symbol

Examples: A$, B$, A1$, AA$

When a character string variable is used, the length of the character string which can be handled must be declared in advance by the DIM statement. If a character string variable is used without the DIM statement declaration, a 20-character character string variable length is declared automatically.

● Use of a character string variable

```
10  DIM A$(11),B$(1),C$(2)
20  A$="TR4751␣GPIB"
30  B$="␣"
40  C$="CONTROLLER"
50  DISP A$,B$,C$
60  END
```

If the above program is executed, an error will be generated at line 40.

```
Subscript out of range  40
```

This is because ten characters have been inserted in C$ in line 40 when only two characters have been declared in the character string variable C$ array in line 10. That is, an error has been generated as a result of attempting to insert a character string longer than the declared length.
Correct programming can be made by revising line 10 as follows.

```
10 DIM A$(11),B$(1),C$(10)
```

Insertion of a space ("␣") in B$ in line 30 is permissible. In character strings, a space is counted as one character.
Also note that fetching of partial contents of a character string variable is possible. And example of this is given below.

```
10  DIM A$(10)
20  A$="ABCDEFGHIJ"
30  SCLEAR
40  DISP A$
50  DISP A$(1)
60  DISP A$(3,5)
70  END
```

When the above program is executed, the following display appears on the CRT.

```
ABCDEFGHIJ
A
CDE
```

The A$(1) in line 50 specifies the first character in the character string variable A$, and the A$(3,5) in line 60 specifies the character string from the third to the fifth character in the character string variable A$.
To fetch a character string portion from the character string variable, key in as follows.

A$[([Start point],[End point])]

Examples:
- A$(3) The third character in A$
- A$(2,6) The character string from the second to the sixth character in A$
- A$(5,) The character string from the fifth character to the end of A$
- A$(,8) The character string from the first character up to the eighth character in A$

This procedure can also be applied character string calculations and IF statements.

Example: B$=A$(2,5)+"HZ"
IF C$(,3)="MKR" THEN 100

② Simple numerical variables
The names of simple numerical variables are expressed as a single alphabetic character A thru Z, plus numerical character, or a capital or small alphabetic character.
In variable names consisting of more than two characters, however, the third and subsequent characters are disregarded. Also note that character combination identical to BASIC key words cannot be used.
Simple numerical variable name: 1 alphabetic character + (1 numerical character, or small or capital alphabetic character)
Examples: A, B, A1, NA, Amplitude
In this case, alphabetic symbols cannot be used in the variable name.

③ Numerical array variables
In programs where large volumes of data are handled, the use of separate variable names for each different number results in a very complex program which is very hard to comprehend. For this reason, numerical array variables capable of handling multiple data in a single variable name are used.
Numerical array variable names are expressed by appending numbers enclosed in parentheses ( ) to simple numerical variables.
Numerical array variables: 1 alphabetic character + 1 numerical character, or small or capital alphabetic character + (number)
Examples: A(1), A(I-2), Band(X), B(X,Y)
The number enclosed in parentheses may be a constant, a variable, or an arithmetic expression. The array may be declared in up to two dimensions.
When numerical array variables are used, the array size must be declared in advance by the DIM statement. If this kind of variable is used without the DIM statement declaration, a (10,10) two-dimensional array is declared automatically.

```
5  SCLEAR
10  DIM A(9)
20  FOR I=0 TO 9
30  INPUT A(I)
40  NEXT I
50  FOR I=0 TO 9
60  DISP A(I)
70  T=T+A(I)
80  NEXT I
90  DISP "Total␣=␣",T
100  DISP "Average␣=␣",T/10
110  END
```

In the above example, after the ten items of data are keyed in by panel key operation and are stored in the numerical array called A, the total and average data are displayed.
An array declaration DIM A(9) has been made in line 10, resulting in an array of ten items of data from A(0) thru A(9) being reserved.
Note that a distinction is made between the numerical array variable A and the simple numerical variable A in this case.

(2) Expressions and functions

(a) Expressions

Expressions consist of numerical constants, numerical variables, and functions linked by operators and parentheses. Constants and variables alone are also called expressions. These expressions are also referred to as numerical value expressions.
Operators employed in BASIC programs differ slightly from those used in general expressions. The differences are shown in the following table.

	Operators used in general expressions	Operators used in BASIC
Addition	+	+
Subtraction	-	-
Multiplication	x	*
Division	÷	/

When an expression is operated, the calculating priority order is determined by the operators and parentheses in the following way.

① Parentheses contents
② Functions
③ Multiplication and division
④ Addition and subtraction

Examples of descriptions of arithmetic expressions in BASIC programs are given below.

General arithmetic expressions	Expressions used in BASIC
$\frac{5x+3}{-x+2}$	(5X+3)/(-X+2)
xX(-1)	X*(-1) or X*-1
3x+2y	3*X+2*Y
$\frac{a+b}{2}$	(A+B)/2

(b) Functions

There are two types of functions - arithmetic functions and character string functions; and these are divided into the following four groups.

- Functions
  - Character string functions
  - Numerical functions
    - General numerical functions
    - Status functions

The following description only covers numerical functions. The numerical functions available in TR4751 BASIC are listed below.

Table 7-4 Numerical Functions

Function	Description
ABS(X)	Absolute value of X.
CONV(A$(X))	Conversion of the ASCII code of the Xth character in A$ to a single-byte binary data numerical value.
COS(X)	Cosine of X (with X in radians).
BIT(X,A)	Select the value of X in terms of 16-bit binary code (binary notation), and judge whether bit A is 0 or 1. The value of BIT(X,A) becomes 0 if bit A is 0, and 1 if bit A is 1. The range for the value of X is 0 to 65535, and the range for the value of A is 0 to 15.
INT(X)	Integral number of X obtained by discarding decimal places.
LOG(X)	Common logarithmic of X. $\log_{10}(X)$
LN(X)	Natural logarithmic of X. $\log_e(X)$
PI	The value of the constant π. PI=3.1415926535
SGN(X)	The sign of X. 1 when X is positive, 0 when 0, and -1 when negative.
SIN(X)	Sine of X (with X in radians).
SPOLL(X)	Serial polling of device where the GPIB address is X, and reading of the status byte.
WCONV(A$(X))	Conversion of two characters of ASCII code from the Xth character in A$ to a 2-byte binary data numerical value. The Xth character is taken as the higher order byte, and the (X+1)th character is taken as the lower order byte.
SQR(X)	The square root of X.
TAN(X)	The tangent of X (with X in radians).
BDEC(A$(X))	Binary (character string) to decimal (numeral) conversion. A binary number represented by a character string starting from the X'th position of A$ is converted to a decimal number. Up to a 16-digit binary number can be converted.
HDEC(A$(X))	Hexadecimal (character string) to decimal (numeral) conversion. A hexadecimal number represented by a character string starting from the X'th position of A$ is converted to a decimal number. Up to a 4-digit hexadecimal number can be converted.

- Status functions
  Status functions are used to indicate the GPIB status. Although this function is a type of numerical value, it has a somewhat special feature.
  The status function converts single bytes of status data into decimal numbers ranging from 0 to 255. Details of this status data is given in the following table.

Table 7-5 Status Function

Bit No.	7	6	5	4	3	2	1	0
Decimals	128	64	32	16	8	4	2	1
Contents	SYSTEM CONT-ROLLER	REN	ACTIVE CONT-ROLLER		IFC	TAKE CONTROL	DATA OUT	DATA IN

BIT0: Set to "1" when input data is placed in the listener buffer from the GPIB.
BIT1: Set to "1" when data stored in the talker buffer is passed to the GPIB.
BIT2: Set to "1" when controller function is received.
BIT3: Set to "1" if the interface clear (IFC) command is received when the TR4751 is not specified as the system controller.
BIT4: No function
BIT5: Set to "1" while the TR4751 has the controller function.
BIT6: Set to "1" when the GPIB Remote Enable Line (REN) is true.
BIT7: Set to "1" when cntroller selection of the TR4751 is OFF.

Bits 0 thru 3 of the status data are cleared at the same time when they are read by the status function.
And when status data is read by the status function, the numerical value obtained is the result of adding a decimal value corresponding to each bit of the status data. For example, when "1" is set in BIT6 and BIT2, a value of [64 + 2 = 66] is obtained. When each bit is checked by a bit function, however, the status data bits can be specified directly. For example, to check bit 0 by

```
A=BIT(STATUS,0)
```

the status of bit 0 of the status data is placed in the variable A. (A=1 when bit 0 is "1", and A=0 when "0").
Note, however, that BIT0 thru BIT3 of the status data are also cleared in this case.

(3) Character string expressions and character string functions

(a) Character string expressions
Character string expressions consist of character string constants, variables, and functions linked by the addition sign (+). Character string expressions can also consist of only a character string constant or character string variable.
The addition sign (+) is used to link separate character strings.
Examples: "ABC" + "DEF"
A$+B$(5,4)
A$+CONV$(67)

(b) Character string functions
Character string functions are listed in Table 7-6 below.

Table 7-6 Character string functions

Function	Description
CONV$(X)	Conversion of the numerical value X to a single-byte character (binary code). X can be a numerical value expression (with X satisfying the range requirements $0 \leqq X \leqq 255$).
WCONV$(X,)	Conversion of the numerical value X to a two-byte character (binary code). X can be a numerical value expression (with X satisfying the range requirements $0 \leqq X \leqq 65535$).
MODE$	Corresponds to the TR4751 mode string.
BIN$(X)	Decimal (numeral) to binary (character string) conversion. Numeral X ($0 \leqq X \leqq 65535$) is converted to a 16-digit binary number. The binary number is represented with a character string.
HEX$(X)	Decimal (numeral) to hexadecimal (character string). Numeral X ($0 \leqq X \leqq 65535$) is converted to a 4-digit hexadecimal number. The hexadecimal number is represented with a character string.

The CONV$ and WCONV$ character string functions correspond to the respective CONV and WCONV numerical value functions.
When the CONV$ and WCONV$ functions are used, numerical values correspond to the characters listed in the following table.

Table 7-7 Numerics and Corresponding Characters when Character String Functions are Used

Numerics	Character	Numerics	Character	Numerics	Character	Numerics	Character
0		32	SPACE	64	@	96	`
1		33	!	65	A	97	a
2		34	"	66	B	98	b
3		35	#	67	C	99	c
4		36	$	68	D	100	d
5		37	%	69	E	101	e
6		38	&	70	F	102	f
7		39	'	71	G	103	g
8		40	(	72	H	104	h
9		41	)	73	I	105	i
10		42	*	74	J	106	j
11		43	+	75	K	107	k
12		44	,	76	L	108	l
13		45	-	77	M	109	m
14		46	.	78	N	110	n
15		47	/	79	O	111	o
16		48	0	80	P	112	p
17		49	1	81	Q	113	q
18		50	2	82	R	114	r
19		51	3	83	S	115	s
20		52	4	84	T	116	t
21		53	5	85	U	117	u
22		54	6	86	V	118	v
23		55	7	87	W	119	w
24		56	8	88	X	120	x
25		57	9	89	Y	121	y
26		58	:	90	Z	122	z
27		59	;	91	[	123	
28		60	<	92	\	124	
29		61	=	93	]	125	
30		62	>	94	^	126	
31		63	?	95	_	127	

Values from 128 to 235 correspond to the inverse characters for values 0 to 127.
CONV$ and WCONV$ are also capable of converting decimal numbers into binary code (binary numbers).
CONV$ is capable of handling numerical values from 0 to 255, generating single-byte binary codes. And WCONV$ is capable of handling numerical values from 0 to 65535, generating two-byte binary codes.

```
10  DIM A$(256),B$(512)
20  FOR I=0 TO 255
30  A$(I+1)=CONV$(I)
40  NEXT I
50  FOR I=0 TO 511
60  B$(2*I+1,)=WCONV$(I)
70  NEXT I
80  END
```

In the above program example, single-byte binary code from 0 to 255 is stored in A$, and two-byte binary code from 0 to 511 is stored in B$. The A$(I+1)=CONV$(I) in line 30 means that the decimal value given as numerical variable 1 is converted into a single-byte binary code and is inserted in the I+1 position of the character string variable A$.
And the B$(2*I+1,)=WCONV$(I) in line 60 means that the decimal value given as numerical variable I is converted into a two-byte binary code and is inserted from the 2*I+1 position of the character string variable B$. The binary code input sequence is higher order byte first followed by the lower order. In this case, the higher order byte is placed in the 2*I+1 position, and the lower order byte is placed in the 2*I+2 position.
In TR4751 BASIC, binary code is thus formed directly from decimal numbers.
And to convert binary code into decimal numbers, the CONV and WCONV numerical functions are used.

CAUTION

In TR4751 BASIC, binary code must be handled as character string variables. Binary code cannot be handled as numerical variables.

(4) Numerical conversion
In the BASIC program of the TR4751, numerals are all handled as 11-digit real numbers. For other representations, use the following functions:

```
BIN$   :  Decimal to binary
HEX$   :  Decimal to hexadecimal
CONV$  :  Decimal to 1-byte binary
WCONV$:   Decimal to 2-byte binary

BDEC   :  Binary to decimal
HDEC   :  Hexadecimal to decimal
CONV   :  1-byte binary to decimal
WCONV  :  2-byte binary to decimal
```

For CONV$ and WCONV$, see Section 7.2.7-(3) "(b) Character string functions".

(a) Decimal and binary conversion
Use BIN$ for decimal to binary. Use BDEC for binary for decimal. BIN$ converts a numeral to 16-digit binary. The converted binary number is output as a character string.

Example

```
10  SCLEAR
20  A=100,B=20,C=3
30  A$=BIN$(A+B+C)
40  DISP A$
50  DISP BIN$(4095)
60  END
```

Result

```
0000000001111011
0000111111111111
```

BDEC converts a binary contained in a string variable to decimal.

Example

```
10  SCLEAR
20  DIM A$(16)
30  A$="0010111101011101"
40  A=BDEC(A$)
50  DISP A
60  DISP BDEC(A$(9))
70  END
```

Result

```
12125
93
```

As seen in line 60 in the example, if a character in the middle of a character string variable is specified, binary to decimal conversion is conducted on the specified and succeeding characters. Conversion ends when conversion of a 16 digit string is completed or when a character other than "1" and "0" appears. When a string to be converted begins with space(s) ("␣"), conversion starts from a character next to the space, ignoring the space(s).

"0101010101010101010101"

Conversion is conducted in this range.

"␣␣␣␣011001A00110"

Ignored.

Conversion is conducted in this range.

(b) Decimal and hexadecimal conversion
Use HEX$ for decimal to hexadecimal conversion. Use HDEC for hexadecimal to decimal conversion.
HEX$ converts a numeral to 4-digit hexadecimal. The converted hexadecimal number is output as a character string.

Example

```
10  SCLEAR
20  A=200,B=1000,C=3
30  A$=HEX$(A+B*C)
40  DISP A$
50  DISP HEX$(65535)
60  END
```

Result

```
0C80
FFFF
```

HDEC converts a hexadecimal number contained in a string variable to decimal number.

Example

```
10  SCLEAR
20  A$="1AC0BDEF"
30  A=HDEC(A$)
40  DISP A
50  DISP HDEC(A$(5))
60  END
```

Result

```
6848
48623
```

As seen in line 50 in the example, if a character in the middle of a string variable is specified, hexadecimal to decimal conversion is conducted on the specified and succeeding characters. Conversion ends when conversion of 4-digit string is completed or when a character other than "0" to "9" and "A" to "F" appears. When a string to be converted begins with a space(s) ("␣"), conversion starts from a character next to the space, ignoring the space(s).

"01AB23CD"

Conversion is conducted in this range.

"␣␣␣10AHEB0"

Ignored.

Conversion is conducted in this range.

(c) 1-byte binary to decimal conversion
CONV converts data contained in a string variable to a decimal number as a binary code. CONV handles one character as an 8-bit binary code.

For instance, suppose "11" is contained in A$. Conversion with BDEC(A$) gives "3," whereas CONV(A$) converts it to a decimal number taking the first "1" in A$ as a 1-byte binary code. Since "1" is 49 in ASCII code, CONV(A$) gives 49. (See Table 7-5.)
To convert the n'th character in a string variable, use CONV(A$(n)). To convert a numeral to 1-byte binary code, as opposed to CONV, use function CONV$. (See Section 7.2.7-(3) "(b) Character string functions".)

Example

```
10  SCLEAR
20  A$="123*"
30  FOR I=1 TO 4
40  DISP CONV(A$(I))
50  NEXT I
60  END
```

Result

```
 49
 50
 51
 52
```

(d) 2-byte binary to decimal conversion
WCONV is an CONV extension. WCONV also converts data contained in a string variable as a binary code. WCONV is different from CONV in that two characters are handled as a 16-bit binary code.
For instance, suppose A$ contains "12". WCONV(A$) converts it to a decimal number as 16-bit data. The upper 8 bits represent "1" and the lower 8-bits represent "2".
In ASCII, "1" is 49 and "2" is 50. They are represented by 00110001 and 00110010 in binary notation.
They are converted as a series of 16-bit data to a decimal number of 12594. This WCONV(A$) gives 12594.
WCONV(A$(N)) converts the N'th and (N+1)'th character in A$ to a decimal number.

Example

```
 5  SCLEAR
10  A$="1234AB*/ab"
20  FOR I=1 TO 5
30  DISP WCONV(A$(I*2-1))
40  NEXT I
50  END
```

Result

```
12594
13108
16706
10799
24930
```

To convert a numeral to a 2-byte binary code, as opposed to WCONV$, use WCONV$. (See Section 7.2.7-(3) "(b) Character string functions".)

7.2.8 Error Messages

If the TR4751 BASIC interpreter detects an error in the program text when a program is executed by RUN command, or when a single line is executed in direct mode, an error message is displayed on the CRT.

(1) When executing in direct mode
An error message from the list in Table 7-8 is displayed on the bottom line of the CRT.

(2) When executing by RUN command
In addition to an error message from the list in Table 7-8 being displayed on the bottom line of the CRT, the program list is displayed from the line preceding the line where the error has been generated.
Furthermore, the text on the line where the error has been generated appears as an inverse display, and the line number of the error line is shown to the right of the error message.

Table 7-8 Error Messages

(a) During program execution

Code	Error message	Description
0.1	Syntax error	Syntax error - statement with unclear meaning
0.11	Missing parameter	Parameter error
0.12	Out of Memory	GOSUB statement nesting level is too large. The memory area capable of using numerical array and character string variables has been exceeded. The usable memory area differs according to program length.
0.13	Duplicate Definition	The DIM statement contains more than one variable with the same name.
0.14	Overflow	The calculation result exceeds the range which can be handled in BASIC. In the case of an underflow, 0 is obtained as a result, and no error is generated.
0.15	NEXT without FOR	NEXT statement has been executed prior to the FOR statement, or else NEXT statement has been included without a corresponding FOR statement.
0.16	Subscript out of range	Numerical array variable or character string variable has been used without an array declaration in the DIM statement. Or else, the numerical array variable or character string variable exceeds the range declared for the array.
0.17	Undefined line number	Branching by GOTO, GOSUB, or IF statement to a line number which does not exist.
0.18	Division by zero	Attempt to divide a number by 0.
0.19	Illegal function call	A value which cannot be calculated has been specified as the function argument. ● LOG(0) ● LN(0) ● SQR(x) (x<0)
0.2	RETURN without GOSUB	RETURN statement execution without branching by GOSUB statement.
0.21	Invalid parameter	More than one talker addresses are specified by SEND TALK statement. Setting of a meaningless parameter.
0.22	Out of data	No data to be read by READ statement.
0.23	Type mismatch	The types of variables do not agree.

Table 7-8 Error Messages (Cont'd)

Code	Error message	Description
0.24	Invalid instruction	CLEAR, REMOTE, LOCAL, RESUME, SEND, TRIGGER, LOCAL LOCKOUT statement are executed after transfer of controller function, or when there is no system controller (before reception of controller function).

(b) During program editing

Error message	Description
Missing line number	Setting integer number outside the 1 thru 32767 range as the line number during program entry.
Memory overflow!	A program entry which is too long to fit in the program storage memory.

7.2.9 Printer Connection

Any type of printer which can be controlled by the GPIB may be connected. Before connecting the printer to the TR4751, carefully read the printer instruction manual.

Connect the printer to the TR4751 and the GPIB. It is recommended that the printer GPIB address being set to 0. If the address setting is not very practical, however, it is not essential to set the address to 0.

Output of lists and data to the printer involves the use of the PLIST command and the PRINT statement. Before sending data to the printer, however, take note of the following precautions.

(1) When printer is set to "listen only"
All data input/output to the GPIB by OUTPUT, PRINT, SEND, or ENTER statement is printed by the printer. In this case, data cannot be sent by only addressing the printer.

(2) When printer address is not set to 0
Set the printer address by PRINTER command before executing the PLIST command or the PRINT statement. No printing will be possible if this command is not set.

Example: Printer address set to 2

```
PRINTER=2
```

Once the PRINTER command has been executed, it remains valid until the power is switched off, or until the PRESET key is pressed.
And since the PRINTER command can also be used within a program, it will be found more convenient to input the PRINTER command at the beginning of the program. An example of this kind of program is given below.

Example:

```
10  DIM A$(40)
20  PRINTER=2
30  INPUT "What is your name?",A$
40  PRINT A$
50  GOTO 20
60  END
```

7.3 SYNTAX AND DESCRIPTION OF COMMADND ADND STATEMENT

7.3.1 Introduction

The syuntax of the commands and statements used by the TR4751 are described below with the aid of diagrams to make the syntax easier to understasnd.

7.3.2 Syntax Descripotion Methods

(1) Diagrams
The syntax is divided into basic elements joined together by straight lines.
Statements always proceed in the direction of the arrows. Where there is brasnching, the statement may proceed in either direction. And where a loop is formed, that loop may be entered any number of times in the direction of the arrow.

(2) Text expressions
The following symbols have been used in the text expressions.
[ ]: The enclosed contents may be omitted.
{ }: The enclosed contents may be repeated once or more times.
| : Denotes "or".
Example: <A>|<B> ..... Use <A> or <B>.
The following terms are used in both diagrammatic and textual representastions.

- Numerical expression : Numerical constant, numerical variable, or numerical formula
- Charascter string expression: Character string constant, character string variable, or character string formula
- Device address : Address of device connected to the GPIB

7.3.3 TR4751 GPIB Controller Command Syntax

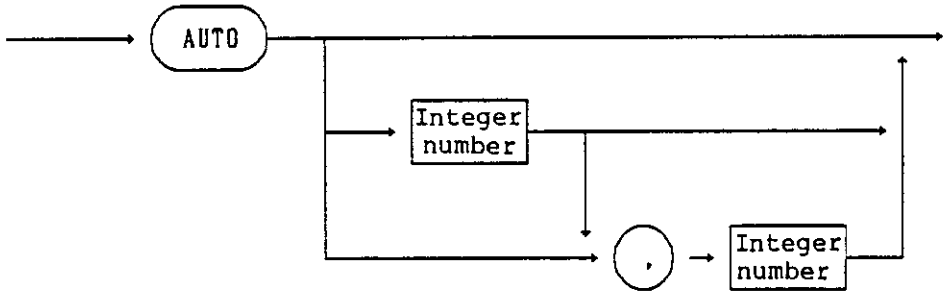
AUTO
LIST
NEW
PLIST
PRINTER
RUN
SIZE

AUTO

Outline

Automatic output of the line numbers on the CRT during program entry.

Syntax



AUTO [Integer number][,integer number]
Specify integer numbers from 1 to 32767.

Description

- When the AUTO command is executed during entry of a BASIC program, line numbers are generated automatically and shown on the CRT.
- The first integer is the starting line number, and the second integer is the line number increment interval.
  AUTO Starting line no., Increment
  Both integers can be omitted, in which case values of 10 are set automatically for the respective items.
- The AUTO function can be released by pressing the auto key again. AUTO function is also released when a program is executed, and when edit mode is cancelled.

Example

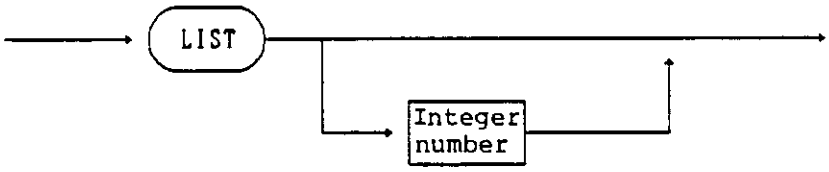
```
AUTO
AUTO 100
AUTO ,15
AUTO 30,5
```

LIST

Outline

Display of the program list on the CRT.

Syntax



LIST [Integer number]

Specify integer numbers from 1 to 32767.

Description

- Display of 19 lines of program list on the CRT.
- Any line number can be specified for display of the program list from that line number.

  LIST Line no.

  If the line number is not specified, the program list is displayed from the beginning of the program.

Example

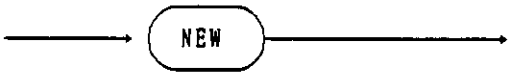
```
LIST
LIST 100
```

NEW

Outline

Deletion of the BASIC program stored in memory.

Syntax



NEW

Description

- Execute this command when the BASIC program already stored in memory is no longer required.
- When the NEW command is executed,
  "Delete old text? Yes [Y] or No [any key]"
  is displayed. The BASIC program is then deleted if the "Y" key is pressed in response, but is retained if any other key is pressed. Once the program has been deleted,
  "Deleted old text"
  is displayed.

Example

```
NEW
```

PLIST

Outline

Output of program list to the printer.

Syntax

PLIST

Description

- The BASIC program list is output in the printer connected to the GPIB.
- The default value for the GPIB address to be sent to the printer is 0. If the GPIB address for the connected printer is not 0, specify the printer's GPIB address by using the PRINTER command.

Example

PLIST

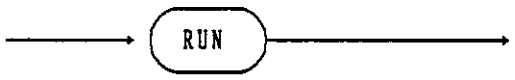
PRINTER

See the PRINTER statement on page 7-88.

RUN

Outline
Execution of the BASIC program.

Syntax



RUN

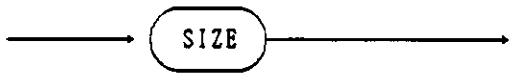
Description
- Execution of the BASIC program from the first line.
- When the RUN command is executed, all variables are cleared before execution of the program is commenced, and array declaration settings are reset.

Example
RUN

SIZE

Outline
Display of remaining program memory area.

Syntax



SIZE

Description
- The number of bytes of remaining memory area is displayed on the 22nd line of the CRT.
- The program memory area is 12277 bytes when no program is stored.

Example
SIZE

7.3.4 TR4751 Basic Statement Syntax

BEEP

CURSOR

DATA

DIM

DISABLE INTR

DISP

ENABLE INTR

END

FOR-TO-STEP
NEXT

GET

GOSUB
RETURN

GOTO

IF GOTO
IF THEN

INPUT

LET

OFF ERR

OFF KEY

OFF SRQ

ON ERR

ON KEY

ON SRQ

PAUSE

PRINT

PRINTER

READ

REM

RESTORE

SCLEAR

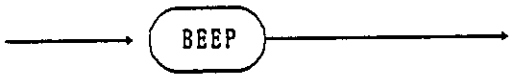
WAIT

BEEP

Outline

Generation of buzzer sound.

Syntax



BEEP

Description

- The TR4751 built-in buzzer is generated when the BEEP statement is executed.

Example

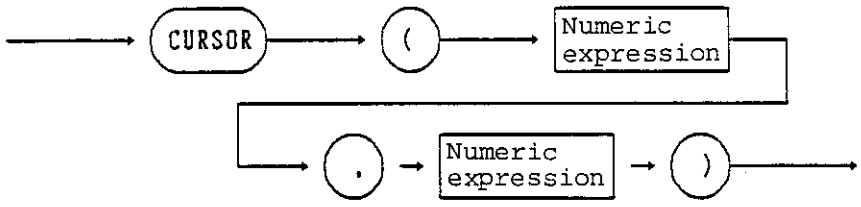
```
10 BEEP
```

CURSOR

Outline

Movement of the cursor to the specified screen co-ordinates.

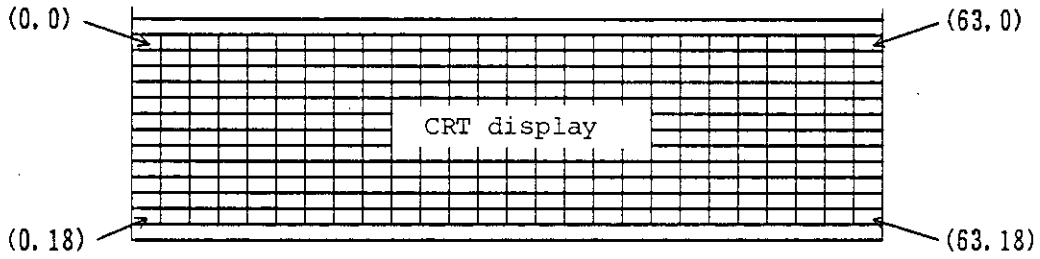
Syntax



CURSOR (Numerical expression, numerical expression)

Description

- The cursor is moved to the position specified on the CRT.
- The first numerical value enclosed in parentheses is the X axis co-ordinate, while the second numerical value is the Y axis co-ordinate.
  CURSOR (X axis co-ordinate, Y axis co-ordinate)
  The X and Y axis co-ordinate values must lie within the following ranges.
  0 ≦ X axis co-ordinate ≦ 63
  0 ≦ Y axis co-ordinate ≦ 18



Example

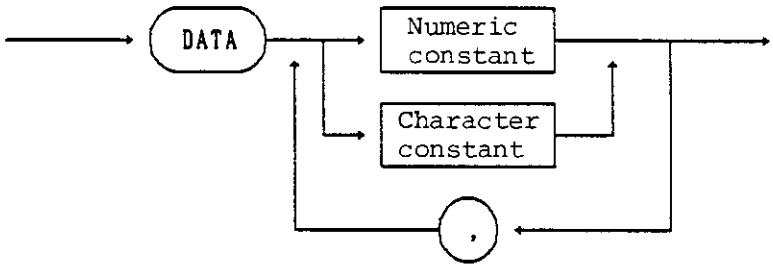
```
10  CURSOR(10,5)
20  CURSOR(X*10,Y+5)
```

DATA

Outline

Preparation of data within a program.

Syntax



DATA Numerical constant | character constant
{,Numerical constant | character constant}

Description

- Preparation of data within a program. The data can handle numerical and character constants.
- Since the DATA statement is a non-executable statement, it has no effect on the program flow, no matter where it is positioned.
- A single DATA statement can contain more than one item of data partitioned by a comma (,), and both numerical and character constants can be handled at the same time.
- Data prepared by the DATA statement is read by the READ statement. (Data is read by the READ statement starting from the DATA statement of the lowest line number).

Example

```
100  DATA 1, 2, 3
110  DATA ABC, DEFG
120  DATA 57.6, HIJK, 4, 32E15
```

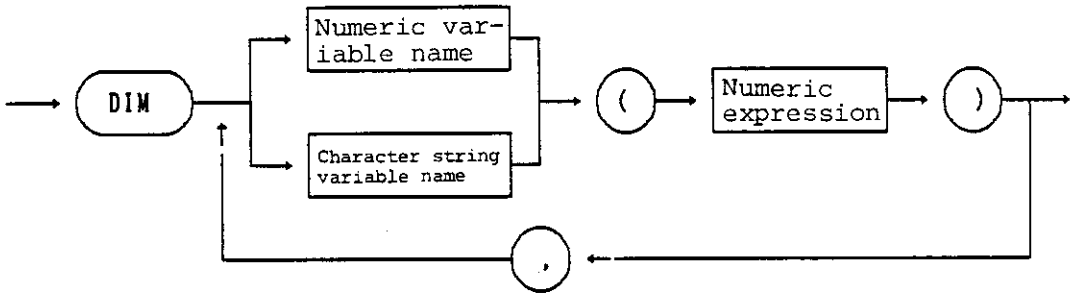
See pages 7-89 and 7-92 for the READ and RESTORE statements.

DIM

Outline

Statement for declaring the definition of the array and character string variables.

Syntax



DIM <A> (Numerical expression){, <A> (numerical expression)}
<A> :: = Numerical variable | character string variable

Description

- When an array variable or character string variable is used, the array variable name and array size must be defined by the DIM statement. Attempt to use an array variable or character string variable which has not be defined results in the generation of an error.
- When the array declaration is made by the DIM statement, array variables of the designated size are reserved in memory. If the array declaration is too large, however, there will not be enough area left for the BASIC program. (An error is generated (Out of memory) if array size is greater than the memory area, and execution of the program is halted).
- The numerical expression indicating the size of the array variable loses any decimal places and becomes an integer, even if the result of calculation is a real number expression.

Example

```
10  DIM A(100)
20  DIM B(20),A$(40)
40  DIM C(I),D(J) B$(X)
```

DISABLE INTR

Outline

Inhibition of interrupts by KEY, KNOB, SRQ, and ERR

Syntax

DISABLE INTR

Description

- Interrupts enabled by the ENABLE INTR, ON ERR, ON KEY, and ON SRQ statements are disabled by this statement.
- To re-enable the interrupts after executing this statement, execute the ENABLE INTR statement. Since the interrupt branching conditions previously set by the ON ERR, ON SRQ statements are saved in this case, there is no need to set them again.
  To change the interrupt branching conditions, however, perform the desired settings with the ON ERR, ON KEY, and ON SRQ statements, or the OFF ERR, OFF KEY, and OFF SRQ statements before executing the ENABLE INTR statement.
- Interrupts remain disabled from immediately after the program has been executed until the ENABLE INTR statement is executed.

Example

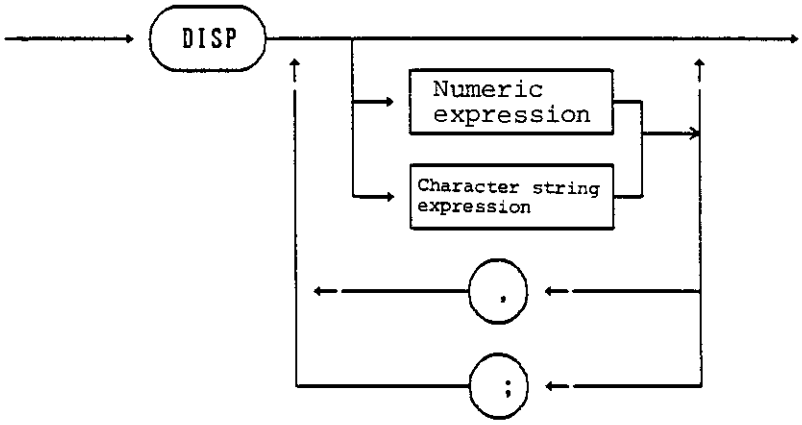
```
10  ON SRQ GOSUB 200
20  ENABLE INTR
30  FOR I=0 TO 100
40  DISP I
50  NEXT I
60  DISABLE INTR
70  FOR I=0 TO 100
80  DISP I
90  NEXT I
100  GOTO 20
110  END
200  DISP SPOLL(1)
210  ENABLE INTR
220  RETURN
```

DISP

Outline

Display of numerical values and character strings on the CRT.

Syntax



DISP [Numerical expression | Character string expression]

Description

- When a numerical value or a numerical value expression is set after DISP, the calculated results for that numerical value or numerical value expression is displayed on the CRT.
- And if a character string is set after DISP, that character string is displayed on the CRT.
- Normally, only a maximum of 21 characters on the bottom line can be displayed by DISP. If the character string to be displayed contains more than 21 characters, the subsequent characters are disregarded. The entire CRT can be used to display the characters or numerical values after the CRT display is cleared by using the SCLEAR statement. In this case, the characters or numerical values to be displayed are output at the cursor position.
- The data displayed can include more than one item of data by partitioning items with a comma (,) or semicolon (;).

Note: If the EXIT statement is executed after executing the SCLEAR statement, the DISP display only appears on the bottom line of the CRT. If the entire CRT display is desired after executing EXIT, the SCLEAR statement must be executed again.

Example

```
10  DISP 123.45
20  DISP 10*I
30  DISP "ABC"
40  DISP A$+"Hz"
50  DISP "START=",M,"Hz"
```

ENABLE INTR

Outline

Release of the DISABLE INTR interrupt disabled status, and the interrupt disabled status applied to interrupts initially enabled by the ON KEY, ON SRQ, or ON ERR statement.

Syntax


```
ENABLE INTR
```

Description

- When branching is generated by an interrupt enabled by the ON KEY, ON SRQ, or ON ERR statement, branching by all interrupts is temporarily disabled. The purpose of this is to prevent nesting of the interrupt processing when an another interrupt is generated while the old interrupt processing is proceeding.
- If this statement is executed when another interrupt is applied after processing of branching generated by an earlier interrupt is completed, the interrupt disabled status is released and the interrupt branching is ready to be processed.
- Processing can be performed smoothly if this ENABLE INTR statement is entered immediately prior to the RETURN statement when the interrupts are processed by subroutine.
- This statement is also executed to enable interrupts again after the DISABLE INTR statement has been executed.
- The interrupt disable status is maintained from immediately after the program is executed up until this statement is executed.

Example

```
10  DIM A$(1)
20  ON KEY GOSUB 100
30  ENABLE INTR
40  GOTO 30
50  END
100  GET A$
110  DISP A$
120  ENABLE INTR
130  RETURN
```

END

Outline

Termination of program execution.

Syntax

END

Description

- When the END statement is executed, execution of the program is halted irrespective of the current program status, and the TR4751 controller is stopped.

Example

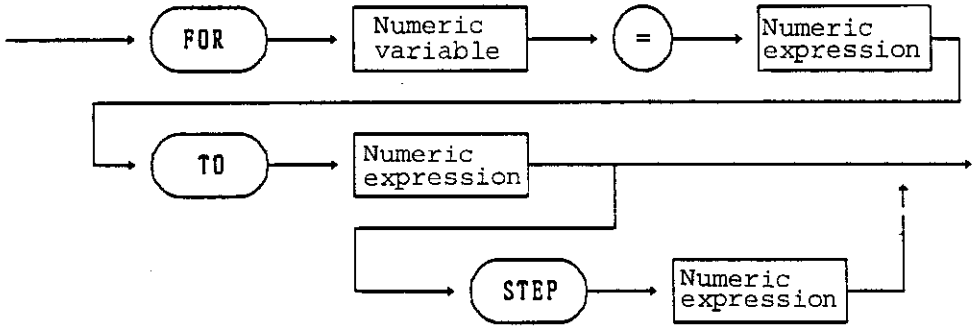
```
10  END
```

FOR-TO-STEP
NEXT

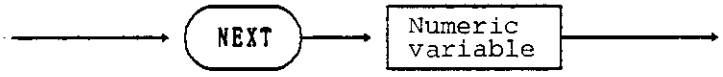
Outline

Formation of a program loop (repeated processing) with a pair of FOR and NEXT statements.

Syntax



FOR Numerical variable = numerical expression
TO Numerical expression [STEP numerical expression]



NEXT Numerical variable

Description

- The specified numerical variable is used as a loop (repetition) counter which is incremented from the initial value to the final value. The loop is terminated when the counter value becomes equal to, or greater than, the final value. Counter changes are executed by the NEXT statement. Hence, the program included between FOR and NEXT statements is processed repeatedly.
- The initial and final values, and the increment are specified in the following way:
  FOR A = (Initial value) TO(Final value) STEP(Increment)
- If STEP (increment) is omitted, the increment is automatically set to +1.
- Nesting of the FOR and NEXT statements is possible.
- The numerical variable names of the loop counter used for a pair of FOR and NEXT statements must be the same. An error is generated if the names are different. (NEXT without FOR)

- If the value of the numerical variable used in the loop counter is changed during loop processing by FOR and NEXT statements, proper loop processing will no longer be possible.

Example

```
10  FOR I=0 TO 100
20  FOR J=A*10 TO B*20 STEP C
30  FOR K=10 TO -10 STEP -1
40  NEXT K
50  NEXT J
60  NEXT I
```

Note: If the following program is generated and a FOR - NEXT loop is executed, the program may appear to be operating without trouble. In fact, however, program operation is not normal. This is because the numerical value of the FOR statement increment is very much smaller than the initial and final values, and is consequently considerably smaller than the TR4751 BASIC significant figures. (TR4751 BASIC significant figures go down to 11 digits). And if the increment is below the significant figures for the initial or final values, the increment is disregarded and the loop counter remains unchanged. The FOR - NEXT statement loop thus becomes a permanent loop without an end.

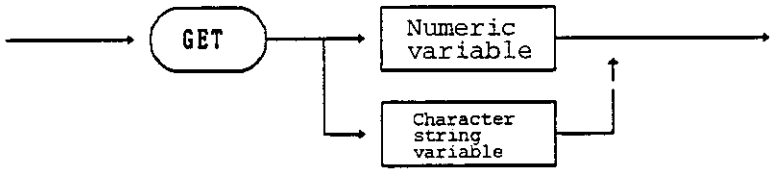
```
10  FOR I=1E12 TO 2E12 STEP 0.01
20  DISP I
30  NEXT I
40  END
```

GET

Outline

Numerical values or characters corresponding to keys pressed just before this statement is executed are obtained as variables.

Syntax



GET Numerical variable | character string variable

Description

- Input of numeric characters from 0 to 9 is enabled by specifying numerical variables.
- Input of all alphabetic characters handled by panel keys is enabled by specifying character string variables.
- Only one numeric or alphabetic character can be input by the GET statement.

Example

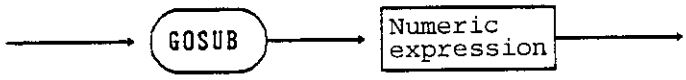
```
10  DIM A$(1)
20  GET A
30  GET A$
```

GOSUB
RETURN

Outline

Branching to and return from the specified subroutine.

Syntax



GOSUB Numerical expression

RETURN

Description

- Shifting control processing to the subroutine beginning from the line number specified by numerical expression, and return by RETURN statement to the next statement following the GOSUB statement.
- Always insert a RETURN statement at the end of the subroutine to enable processing to return to the main program.
- An error is generated if a RETURN statement is executed without first branching to a subroutine. (RETURN without GOSUB)
- Since GOSUB statement - RETURN statement nesting is possible, processing can be branched to another subroutine from within the entered subroutine. Too much nesting, however, will use up the available memory area, resulting in the generation of an error. (Out of memory)

Example

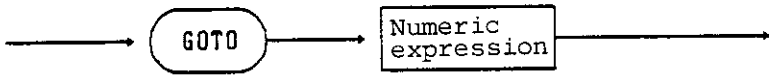
```
10   GOSUB 1000
20   GOSUB 2000
30   END
1000   IF I=0 THEN RETURN
1010   I=I+10
1020   RETURN
2000   GOSUB 3000
2010   A=I*100
2010   RETURN
3000   A=123
3010   RETURN
```

GOTO

Outline

Branching to the specified line number.

Syntax



GOTO Numerical expression

Description

- Statement for unconditional branching to the specified line number.
- An error is generated if the specified line number does not exist in the program. (Undefined line number)

Example

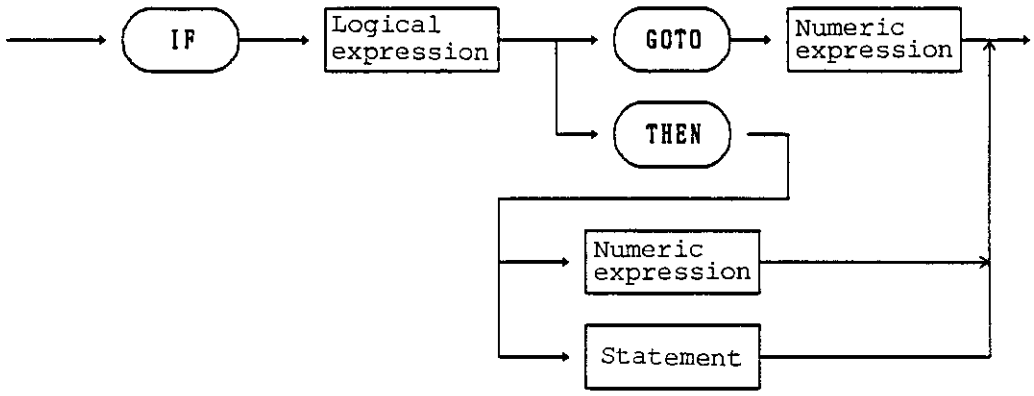
```
10  GOTO 100
20  GOTO I*20
```

IF GOTO
IF THEN

Outline

Branching according to conditions, and execution of specified statement.

Syntax




```
IF  <Logical expression> <A> | <B>
    <A>::=GOTO  Numerical expression
    <B>::=THEN  Numerical expression | <Statement>
```

Description

- Program branching and processing according to logical conditions.
- The THEN statement or GOTO statement is executed after the logical expression relationship is established. The THEN statement can be followed by a line number or a statement. When followed by a line number, the THEN statement has the same meaning as a GOTO statement. And if followed by a statement, that statement is executed.
- If the logical relationship is not established, the processing proceeds to the next line.
- There are six types of logical expressions.

A=B	Relationship is established when A and B are equal.
A>B	Relationship is established when A is greater than B.
A<B	Relationship is established when A is less than B.
A>=B (A=>B)	Relationship is established when A is greater than or equal to B.
A<=B (A=<B)	Relationship is established when A is less than or equal to B.
A<>B (A><B)	Relationship is established when A and B are not equal.

In the above logical expressions, both A and B can be either numerical expressions or character string expressions. The types of expressions, however, cannot be mixed.

Example

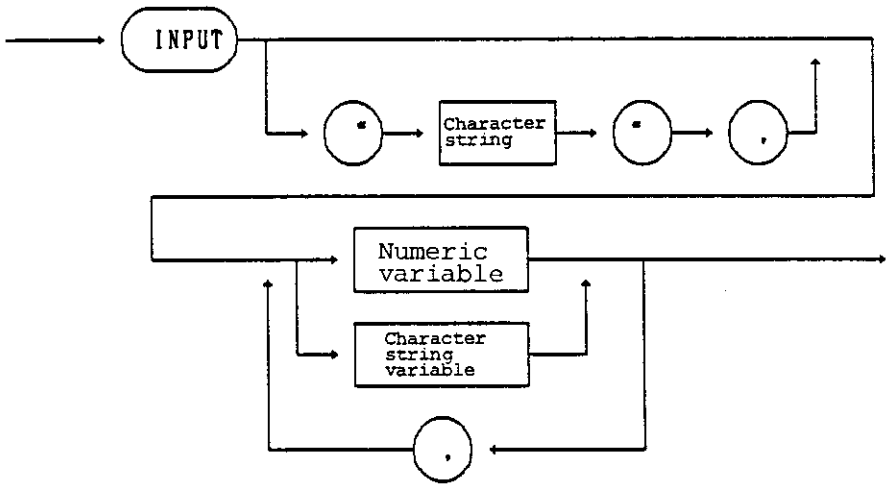
```
10  IF A=10 GOTO 1000
20  IF A+1=B*10 THEN DISP "OK"
30  IF B<>C THEN 200
40  IF A$="ABC" GOTO 1000
50  IF A$>=B$ THEN DISP "READY"
60  IF A$+"DEF"="ABCDEF" THEN 300
```

INPUT

Outline

Substitution of the keyed-in input data in a numerical or character string variable.

Syntax




```
INPUT  ["<Character string>",] numerical variable |
       character string variable{,numerical variable |
       character string variable}
```

Description

- Input of data by panel key operation.
- When the INPUT statement is executed, the program is stopped temporarily and a question mark "?" appears on the CRT to indicate that the program is waiting for a key entry. This wait status is maintained until the ENTER key is pressed. When this ENTER key is pressed, the keyed-in data is substituted in the variable.
- More than one variable can be specified using commas (,) to partition each variable. In this case, the input data may be entered one item at a time, pressing the ENTER key after each item of data has been keyed in, or it may be entered as a block, partitioning each input item with a comma.
- The number of question marks ("?") displayed when more than one variable is specified and keyed in varies according to which variable key input is being waited for in wait status. For example, "??" is displayed when in key-in wait status for the second variable, and "???" is displayed when in key-in wait status for the third variable. In other words, the number of question marks indicates which variable is being waited for key-in wait status.

- Although either numerical or character string variables can be handled by the INPUT statement, a question mark is displayed again if non-numerical characters (alphabetic characters or symbols) are keyed in when a numerical variable is to be entered. Since this question mark denotes a request for re-input of data, key in the correct data.
- It is also possible to display a comment instead of the question mark when a key input request is made by the INPUT statement. If the designation is made by enclosing a character string between quotation marks (") after INPUT, that comment is displayed instead of the question mark ("?").

Example

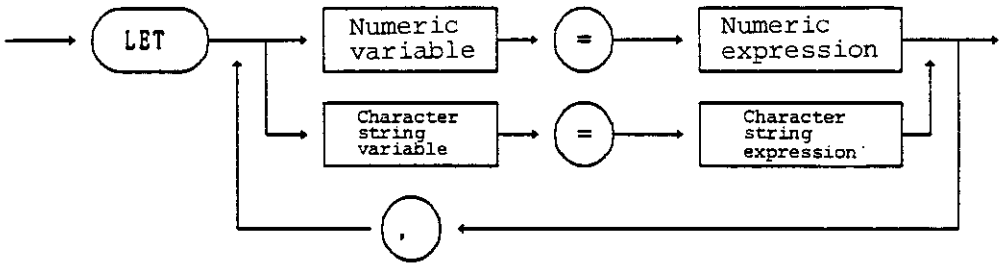
```
10  INPUT A
20  INPUT X,Y,Z,A$
30  INPUT "Please input",B
```

LET

Outline

Substitution of variables.

Syntax




```
LET  <A> | <B>{,<A> | <B>}
     <A>::=Numerical variable = numerical expression
     <B>::=Character string variable = character string expression
```

Description

- Substitution of numerical or character string expression into numerical or character string variables. In this case, the equal sign "=" denotes substitution, and differs in meaning from the mathematical equal sign.
- LET can be omitted from the program.

Example

```
10  DIM A$(10)
20  LET A=10
30  LET B=A*3, C=123
40  LET A$="ABC"
50  A=10, B=A*3, C=123
60  A$="ABC", A=100
```

OFF ERR

Outline

Inhibition of branching by error detection interrupt.

Syntax

OFF ERR

Description

- Inhibition of branching by error detection interrupts enabled by the ON ERR statement.

Example

```
10  OFF ERR
```

OFF KEY

Outline
Inhibition of branching by key input interrupt.

Syntax

OFF KEY

Description
- Inhibition of branching by key input interrupts enabled by the ON KEY statement.

Example
```
10  OFF KEY
```

OFF SRQ

Outline

Inhibition of branching by service request (SRQ) interrupt.

Syntax

OFF SRQ

Description

- Inhibition of branching by GPIB SRQ signal interrupts enabled by the ON SRQ statement.

Example

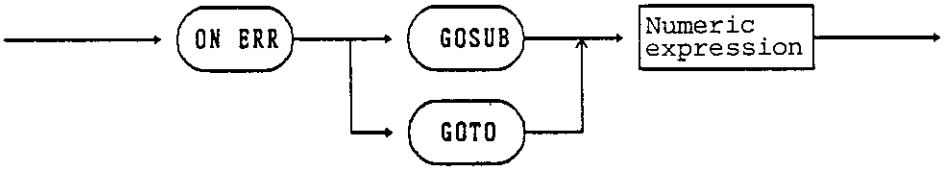
```
10  OFF SRQ
```

ON ERR

Outline

Enabling of branching by error detection interrupt.

Syntax



ON ERR GOSUB | GOTO Numerical expression

Description

- An interrupt is generated when the BASIC interpreter detects a program error during execution of the program. Branching is executed as a result of the interrupt.
- The branching is executed from the line where the error is detected.
- When returning from a subroutine, processing resumes from the next statement after the statement being executed when the interrupt was generated.
- Branching by error detection interrupt can be inhibited by executing the OFF ERR statement.
- When branching is generated by the ON ERR statement, the error contents are indicated by numbers 0.1 thru 0.24 set by the ERR function. Details of these contents are given in the ERR function correspondence table.

Error Types and Corresponding ERR Functions

ERR function value	Error message	ERR function value	Error message
0.1	Syntax error	0.18	Division by Zero
0.11	Missing parameter	0.19	Illegal function call
0.12	Out of memory	0.2	RETURN without GOSUB
0.13	Duplicate Definition	0.21	Invalid parameter
0.14	Overflow	0.22	Out of data
0.15	NEXT without FOR	0.23	Type mismatch
0.16	Subscript out of range	0.24	Invalid instruction
0.17	Undefined line number		

Example

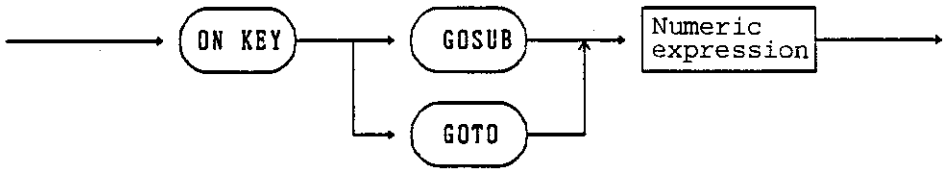
```
10  ON ERR GOTO 100
20  ENABLE INTR
30  ABCDEF
100  DISP ERR*100
110  END
```

ON KEY

Outline

Enabling of branching by key input interrupt.

Syntax



ON KEY GOSUB | GOTO Numerical expression

Description

- An interrupt is generated when TR4751 panel keys are pressed. Branching is executed as a result of this interrupt.
- The branching is executed after processing of the statement being executed when the interrupt was generated is completed. In other words, the statement where branching is executed during program execution depends on the timing of the key-in interrupt.
- When returning from a subroutine, processing resumes from the next statement after the statement being executed when the interrupt was generated.
- Branching by key-in interrupt can be inhibited by executing the OFF KEY statement.

Example

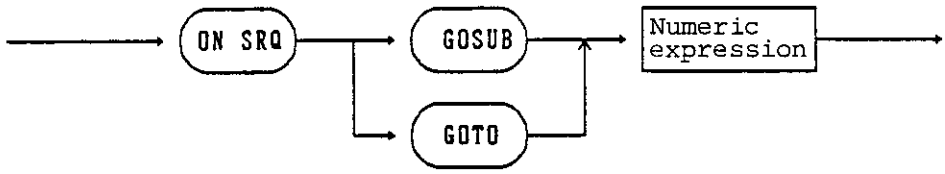
```
10  ON KEY GOSUB 2000
20  ENABLE INTR
```

ON SRQ

Outline

Enabling of branching by service request (SRQ) interrupt.

Syntax



ON SRQ GOSUB | GOTO Numerical expression

Description

- An interrupt is generated when the GPIB single line signal SRQ (service request) is "true" (low level). Branching is executed as a result of this interrupt.
- The branching is executed after processing of the statement being executed when the interrupt was generated is completed. In other words, the statement where branching is executed during program execution depends on the timing of the generation of the SRQ signal.
- When returning from a subroutine, processing resumes from the next statement after the statement being executed when the interrupt was generated.
- Branching by SRQ interrupt can be inhibited by executing the OFF SRQ statement.

Example

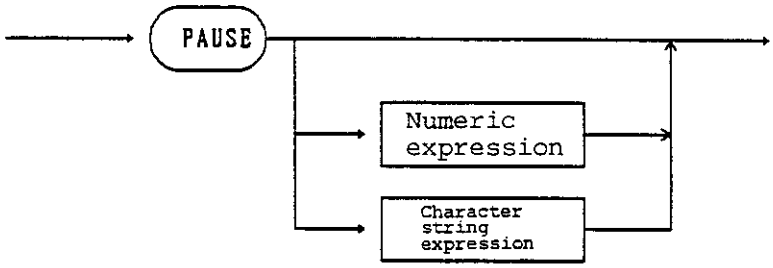
```
10  ON SRQ GOSUB 2000
20  ENABLE INTR
```

PAUSE

Outline

Temporary halting of program execution.

Syntax



PAUSE [Numerical expression | character string expression]

Description

- Program execution is stopped until a panel key is pressed.
- When the PAUSE statement is executed, "PAUSE" is displayed on the CRT to indicate that processing has been temporarily stopped. Program execution can be resumed from the next statement by pressing an arbitrary key (apart from the PRESET key).
- If a numerical or character string expression is specified after the PAUSE statement, a numerical value or character string can be displayed instead of "PAUSE".

Example

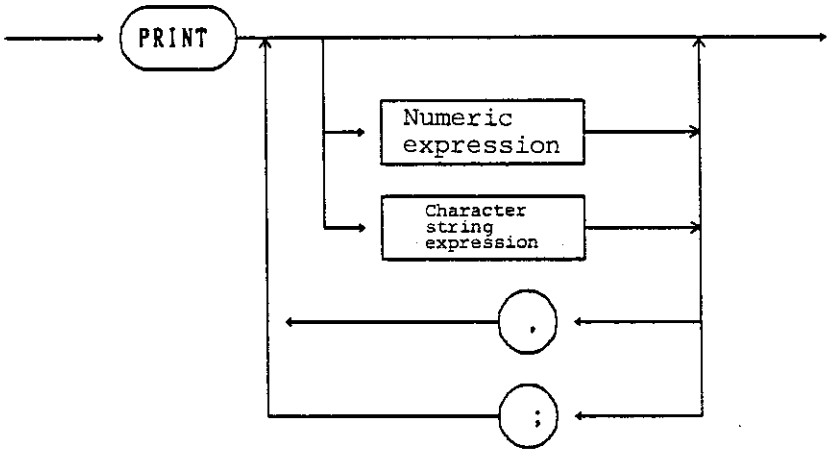
```
10  PAUSE
20  PAUSE A*10
30  PAUSE "ABCD"
40  PAUSE A$
```

PRINT

Outline

Output of numerical value or character string to a printer.

Syntax



PRINT [Numerical expression | character string expression
{,|;Numerical expression | character string expression}]

Description

- Output of the specified numerical value or character string to a printer connected to the GPIB.
- If more than one numerical value or character string is partitioned by commas (","), successive numerical values or character strings are output one after another without carriage return.
- And if a comma (",") or semicolon (";") is placed at the end of the PRINT statement, the end of the print output is not followed by a carriage return. That is, when the next PRINT statement is executed, printing is continued on the same line as the previous print output.
- The default value is 0 for the device address sent from the TR4751 to the printer connected to the GPIB. Therefore, if the printer device address is already 0, the printer may be connected without any further modifications. If the printer address is not 0, on the other hand, the TR4751 must be informed of that fact. The device address sent from the TR4751 to the printer can be changed by the PRINTER statement. (See the PRINTER statement on page 0-00).

Example

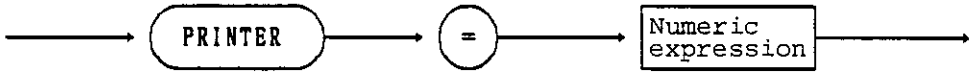
```
10  PRINT 123*456
20  PRINT "ABC"
30  PRINT "Freq.=", A, "Hz"
40  PRINT I,
```

PRINTER

Outline

Designation of the device address to be sent to the printer.

Syntax



PRINTER=Numerical expression

Description

- Command for conveying the device address of the printer connected to the GPIB to the TR4751. The TR4751 must always be informed of the printer device address by this PRINTER statement before the PRINT statement is executed.
- The default value is 0 for the printer device address sent from the TR4751. If the printer device address is already 0, the PRINTER statement is not required to be executed.
- The device address must be integer of 0 thru 31.

Example

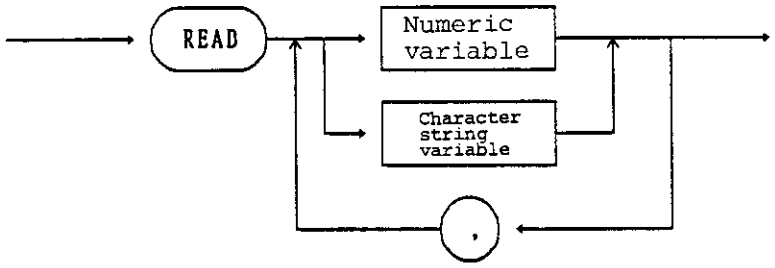
```
10  PRINTER=1
```

READ

Outline

Read out of data prepared by the DATA statement.

Syntax



READ Numerical variable | character string variable
{, Numerical variable | character string variable}

Description

- Sequential reading of data prepared by the DATA statement, and substitution in specified variables.
  For example, where the following program is executed,

```
10  READ A
20  READ B,C
30  DATA 1,2,3
```

  numerical values are substituted in successive variables in the following way:

```
A=1, B=2, C=3
```

- More than one variable can be specified in a single READ statement by partitioning each variable by commas (",").
- The number of data items read by the READ statement must correspond on a 1-to-1 basis with the number of data items prepared by the DATA statement.
  If insufficient read data is obtained during READ statement execution, an "Out of data" error is generated.
- And if the type of variable where data is to be substituted does not match the data to be read by the READ statement, a "Type mismatch" error is generated.
- The DATA statement data can be read again from the beginning by using the RESTORE statement.

Example

```
10  DIM A$(4),B$(4)
20  READ AA
30  READ B,C
40  READ A$
50  READ D,B$,E
60  DATA 1,2,3,ABCD,4,EFGH
70  DATA 5
```

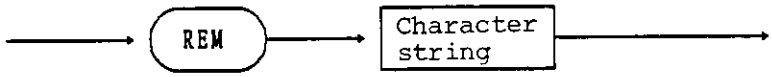
See pages 7-61 and 7-92 for details on the DATA and RESTORE statements.

REM

Outline

Program comments.

Syntax



REM <Character string>

Description

- Statement used to include comments in programs.
- Since the REM statement is not executed, the character string following REM may be of any form. All alphanumeric characters can be used.
- The REM statement can be substituted by an exclamation mark ("!").
- Note that multi-statements cannot be used (that is, a REM statement cannot be followed by a colon ":"). Anything following a colon will be taken as a comment.

Example

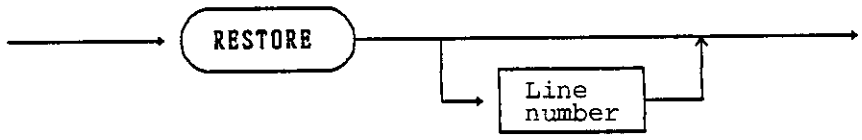
```
10  REM "PROGRAM 1"
20  ! 1983-JUN-02
30  A=A+1:! INCREMENT A
```

RESTORE

Outline

Reading of data by READ or DATA statement is recommenced from the beginning by this statement.

Syntax



RESTORE [Line number]

Description

- This statement is used when reading of the DATA statement from the beginning again is desired. And if the RESTORE statement is executed while reading the READ statement data, reading is recommenced from the beginning again. Therefore, this statement is useful in reading one type of data many times over.
- And if a line number is specified in the RESTORE statement, data reading is recommenced from the data on the specified line.

Example

```
10  READ A,B,C
20  RESTORE
30  READ D,E,F
40  RESTORE
50  READ H,I,J
60  DATA 10,20,30
70  DATA 40,50,60
80  END
```

SCLEAR

Outline
Clearing of the CRT.

Syntax

SCLEAR

Description
- Clearing of all characters and graphic lines currently displayed on the CRT.
- Clearing of the CRT display is also accompanied by return of the cursor to the home position.
- Execution of the SCLEAR statement can be followed by a return to display mode by using the DISP statement. Usually, the DISP statement enables a maximum of 21 characters to be displayed on the bottom line of the CRT. If the SCLEAR statement is used, however, display in the entire CRT is possible.
- When the SCLEAR statement is executed, the synthesized signal generator parameters are no longer displayed. If display of the parameters is again desired, execute the EXIT statement.

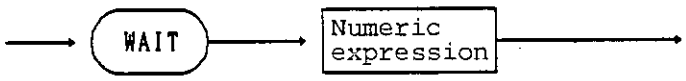
Example
```
10  SCLEAR
```

WAIT

Outline

Program execution is stopped for the specified period of time.

Syntax



WAIT Numerical expression

Description

- Program execution is stopped for the period of time specified by the numerical value following WAIT.
- The unit of the numerical value is the milli-seconds (ms). Values from a minimum of 0 seconds to 65.535 seconds can be set.
- The delay set by the WAIT statement is subject to a certain amount of difference due to various conditions during program execution. Also note that the WAIT specified delay time does not include the time taken to execute the WAIT statement.

Example

```
10  WAIT 1000
20  WAIT I*10-5
```

7.3.5 TR4751 Basic GPIB Control Statement Syntax and Applications

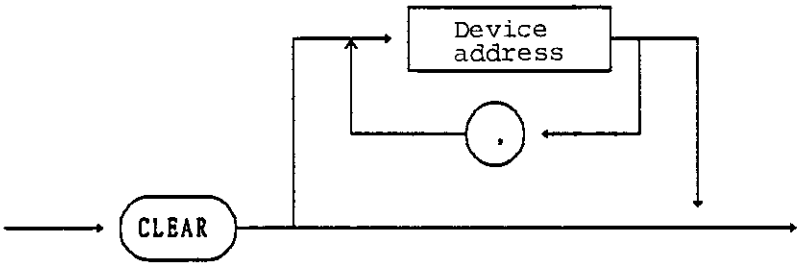
CLEAR
DELIMITER
ENTER
LISTEN BUFFER
LOCAL
LOCAL LOCKOUT
OUTPUT
PASS CONTROL
REMOTE
REQUEST
RESUME
SEND
TALK BUFFER
TRIGGER

CLEAR

Outline

Initialization of selected or all devices connected to the GPIB.

Syntax



CLEAR [Device address{,device address}]

Description

- If the CLEAR statement is executed without specifying any device addresses, the DCL (device clear) universal command is sent to the GPIB, resulting in all devices connected to the GPIB being initialized.
- When a device address is specified after CLEAR, only the device specified by device address is addressed and the SDC (select device clear) address command is sent to that device. In this way, it is possible to initialize only specific devices. More than one device address can be specified at any one time.
- When the CLEAR statement is executed, the GPIB attention (ATN) line is set to true (Low level). And if a device address is specified, the specified device is automatically placed to listener mode. If ATN is set to false (High level), or when remote status is released, the respective RESUME or LOCAL statement must be executed.
- Specifying an asterisk ("*") instead of a device address results in one's own address being specified.

Example

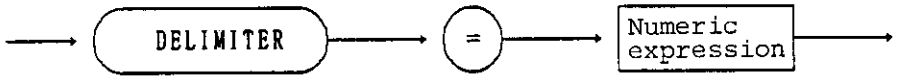
```
10  CLEAR
20  CELAR 2
30  CLEAR 1,3,5,7
40  CLEAR A*10+I
```

DELIMITER

Outline

Statement for selecting and setting four different delimiters.

Syntax



DELIMITER=Numerical expression

Description

- Setting of the delimiter corresponding to the number indicated by numerical expression. The delimiter selection numbers and corresponding types are listed in the following table.

Selection number	Delimiter type
3	Output of "CR" and "LF" 2-byte codes.
0	Output of "CR" and "LF" 2-byte codes. The "LF" output is also accompanied by a uniline signal "EOI" output.
1	Output of the "LF" 1-byte code.
2	Output of the uniline signal "EOI" at the same time as the final byte of data.

- An error is generated if the numerical expression result is outside the 0 to 3 range. All decimal positions are disregarded and the results handled as integer numbers.
- "DELIMITER=0" is set automatically when the power is switched on.

Example

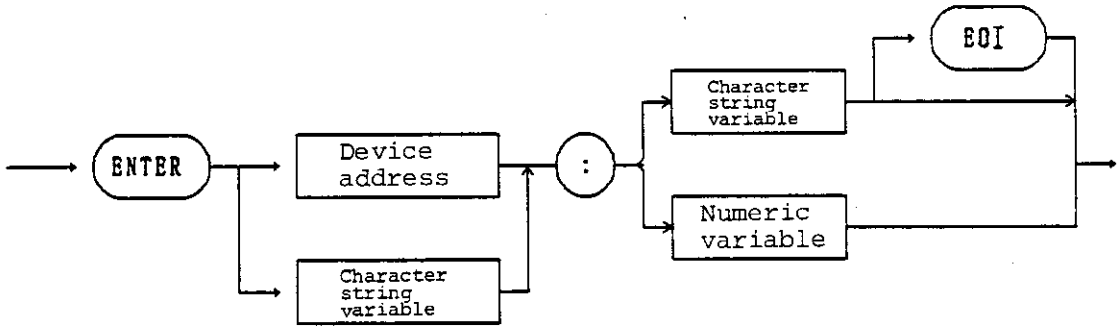
```
10  DELIMITER=0
20  DELIMITER=1
30  DELIMITER=A*10
```

ENTER

Outline

Data acceptation from the GPIB.

Syntax



ENTER Device address | character string variable :
numerical variable | character string variable [EOI]

Description

- Data from the device specified by device address is input via the GPIB, and is stored in the BASIC variable as a numerical value or character string. Note, however, that if the device specified by device address does not include a talker capability, the controller will be brought to a halt without completing the handshake.
  And if a character string variable is used, the character string variable must be declared in advance by DIM statement.
- If "EOI" is specified after the character string variable name, the data input from the GPIB will be terminated by an EOI signal. In this case, "CR" and "LF" inputs will not be regarded as delimiters. Use this function for input of binary data.
- Also note that if the character string variable used in the destination for character string inputs is not sufficiently long enough, an input data overflow may be generated, resulting in the data which cannot fit into the character string variable being disregarded.
- If a character string variable is specified in place of a device address, the contents of that character string variable are substituted in the destination variable.
  This procedure is used for data input when the controller capability is transferred from the TR4751 to another device.
- Specifying an asterisk ("*") instead of a device address results in one's own address being specified.

Example

```
10  ENTER 1:A
20  DIM A$(20),B$(1000),C$(40)
30  ENTER 2:A$
40  ENTER 3:B$ EOI
50  LISTEN BUFFER C$
60  ENTER C$:A
```

See the LISTEN BUFFER statement on page 7-101.

INTERFACE CLEAR

Outline

Initialization of all GPIB interfaces connected to the TR4751.

Syntax

INTERFACE CLEAR

Description

- Execution of this statement results in output of the GPIB single line signal IFC for about 100 µs.
  Listener or talker status is released for all GPIB interfaces of devices connected to the TR4751 GPIB when the IFC signal is received.

Example

```
10  INTERFACE CLEAR
```

LISTEN BUFFER

Outline

Designation of the buffer to be used as listener.

Syntax

LISTEN BUFFER Character string variable

Description

- Direct input of data from the GPIB by ENTER statement is not possible after the control function has been delivered to another device by the PASS CONTROL statement, nor while the TR4751 is not the system controller (that is, until the controller capability is received) since the bus cannot be controlled while there is no controller capability.
  While the controller capability is not delivered to the TR4751, data input is executed via the listener buffer.
  When the LISTEN BUFFER statement is executed, the specified character string variable is handled as the listener buffer, and when the TR4751 is addressed to listen, input data is received by the listener buffer.
  The status function judges whether or not data has been input in the listener buffer (see Section 7.2.7-(2) (b) "Functions").
- If the size of the input data is greater than the array declaration value for the character string variable used as the listener buffer, any overflow data is disregarded.

Example

```
10  DIM A$(100)
20  LISTEN BUFFER A$
30  IF BIT(STATUS, 0)<>1 THEN 30
40  ENTER A$:A
50  DISP A
```

CAUTION

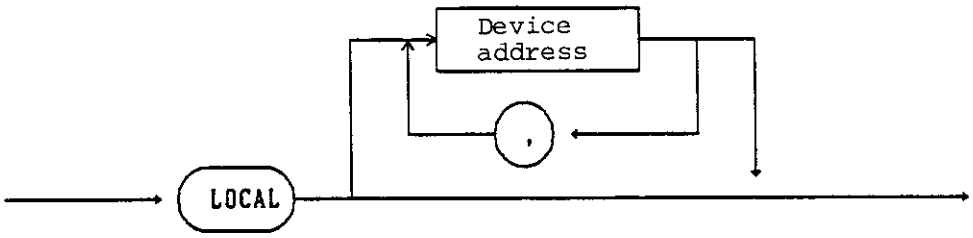
The uniline signal EOI is used as the delimiter for data acceptance. Data input is continued until reception of the EOI signal. Note that the "CR" and "LF" codes are not used as delimiters.

LOCAL

Outline

Release of the specified device from remote control status. And the REN (remote enable) line is made false.

Syntax



LOCAL [Device address{,device address}]

Description

- If the LOCAL statement is executed without a device address designation, the GPIB REN (remote enable) line is set to false (High level), and all devices connected to the GPIB are switched to local control status. Note that GPIB devices cannot be set by OUTPUT command when REN is false (since control by GPIB is no longer disabled). REN is returned to true (Low level) by executing the REMOTE statement.
- When a device address is specified after the LOCAL statement, only the device specified by the device address is addressed and released from remote control status.
- Since the GPIB ATN (attention) line is set to true (Low level) when the LOCAL statement is executed, it may sometimes be necessary to execute the RESUME statement to set ATN line to false (High level).
- If an asterisk "*" is specified in place of the device address, one's own address is specified.

Example

```
10  LOCAL
20  LOCAL 1
30  LOCAL 1,2,3
40  LOCAL A*10+J
```

LOCAL LOCKOUT

Outline

Inhibition of the function whereby devices connected to the GPIB are normally placed to local control status by device's own panel key operation.

Syntax

LOCAL LOCKOUT

Description

- When each device connected to the GPIB has been placed into remote control status (that is, when remotely controlled by the GPIB), the panel keys of each device are locked, and data cannot be set by panel key operation. The LOCAL key (usually written LCL), however, remains unlocked. Pressing this key returns the respective device to local control status where data setting is again enabled. As a result, some problems can arise during remote control operation which prevent proper control. If the LOCAL LOCKOUT statement is executed, however, even the LOCAL key of each device connected to the GPIB is also locked, thereby inhibiting local control of all devices completely.
- When the LOCAL LOCKOUT statement is executed, the LLO (local lockout) universal command is sent to the GPIB. Since the ATN (attention) line remains at true (Low level), it may be necessary to execute the RESUME statement to set the ATN line to false (High level).

Example

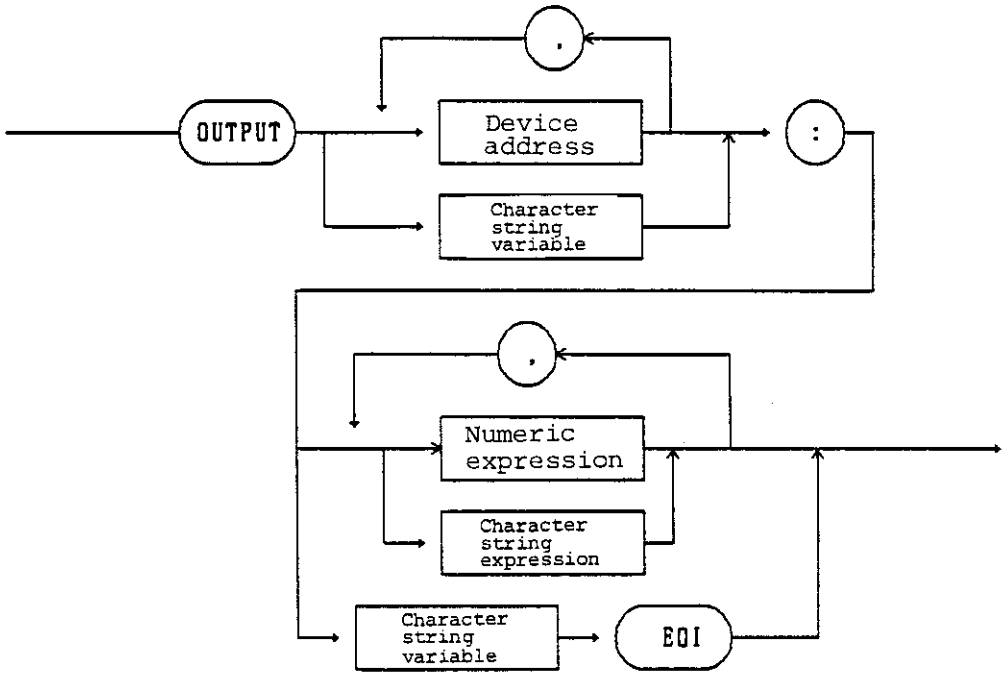
```
10  LOCAL LOCKOUT
```

OUTPUT

Outline

Output of data to the GPIB.

Syntax




```
OUTPUT  Device address{,device address}|
   <A>::=Numerical expression | character string expression
         {,numerical expression | character string expression}
   <B>::=character string variable  EOI
```

Description

- Numerical values and character strings are sent as ASCII data to devices specified by device address.
  Device addresses can be partitioned by comma (,) to specify more than one device at a time. It is also possible to use mixed numerical and character string expressions by partitioning by commas.
- If the result of operations on numerical expressions exceed ±99.999.999.999, the output is expressed as an exponential value.
  For example, the output of the result of the
  ```
  OUTPUT  1:1234*56E10
  ```
  operation is expressed as
  "691.04E+12"

- If there connected no devices specified by device address on the GPIB, the program proceeds to the next line without anything being executed.
- If the OUTPUT statement is executed with the REN line at true (Low level), devices specified by device address are automatically turned to remote control status. To release remote control status by program, execute the LOCAL statement.
- If a character string variable is specified in the output data, and "EOI" is attached to the end of the statement, data stored in the character string variable is output. The data transfer is completed by output of the single line signal EOI together with the last byte. The "CR" and/or "LF" delimiters are not output in this case. This method is used when binary data is transferred.
- If a character string variable is specified instead of the device address destination, the data to be output in that character string variable is stored in ASCII code. This method is used to output data after the TR4751 has delivered the controller capability to another device.
  This method is also used when converting numerical values to character string numerical values.
- If an asterisk "*" is specified in place of the device address, one's own address is specified.

See the TALK BUFFER statement on page 7-112.

Example

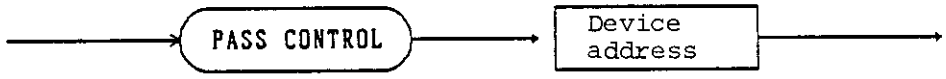
```
10  OUTPUT 3:1234
20  OUTPUT 2:A,B+1/I
30  OUTPUT 4:"ABCD"
40  OUTPUT 5:A$,"ABCD",A$+B$+"EF",C$(3,6)
50  OUTPUT 1,2,3:A*30,1.387,A$(5),"AB"
60  OUTPUT N*100+I:"CF",A,B$
70  OUTPUT 1:A$ EOI
80  TALK BUFFER D$
90  OUTPUT D$:A*B+10
```

PASS CONTROL

Outline

Delivery of controller capability to the specified device.

Syntax



PASS CONTROL Device address

Description

- Talk address and the TCT (take control command) are sent to the device specified by device address to deliver the control capability.
- The GPIB ATN (attention) line turns to false (High level) after this statement is executed.

Example

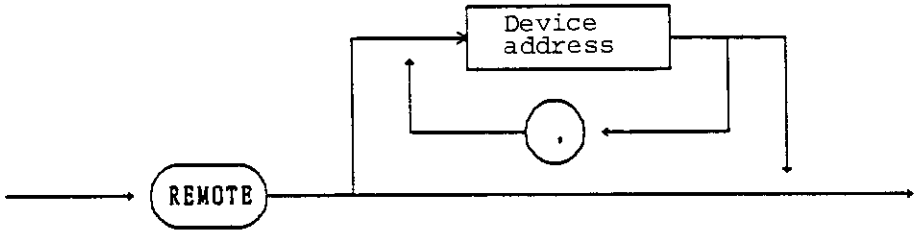
```
10  PASS CONTROL 12
50  PASS CONTROL A*B-10
```

REMOTE

Outline

Specified devices are placed to remote control status, or else GPIB REN (remote enable) is set to false.

Syntax



REMOTE [Device address{,device address}]

Description

- If the REMOTE statement is executed without a device address designation, the GPIB REN (remote enable) line is set to true (Low level) to enable all devices connected to the GPIB to be controlled remotely. The REN line is returned to false (High level) by executing the LOCAL statement.
- When a device address is specified after the REMOTE statement, the device specified by the device address is put into remote control status (but only when the REN line is true (Low level)). More than one device address can be specified at this time. The remote control status can be released by executing the LOCAL statement.
- Although the REMOTE statement is used to put the selected devices into remote control status, specified devices are put into remote control status automatically when the following statements are executed without the REMOTE statement (note that remote control status can only be achieved with the REN line true (Low level)).

```
CLEAR  [Device address{,device address}]
OUTPUT  Device address{,device address}:
        <output data>{,<output data>}
REMOTE  [Device address{,device address}]
SEND LISTEN  Device address{,device address}
TRIGGER  Device address{,device address}
```

- If an asterisk "*" is specified in place of the device address, one's own address is specified.

Example

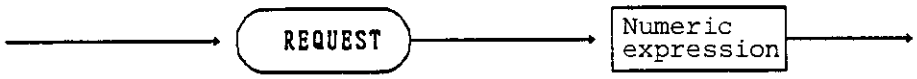
```
10  REMOTE
20  REMOTE 5
30  REMOTE 1,2,3,4
40  REMOTE A*100+I
```

REQUEST

Outline

Setting of the serial poll status byte.

Syntax



REQUEST Numerical expression

Description

- Setting of the bits of the status byte specified by numerical expression. (The result of operations on the numerical expression is to lie within the 0 to 255 range).
- Numerical values and the corresponding status byte bits are given in the following table.

Bit	7	6	5	4	3	2	1	0
Numerical value	128	64	32	16	8	4	2	1

Setting bits 4 and 6 of the status byte in this case results in

```
REQUEST  64+16
```

or

```
REQUEST  80
```

Example

```
10  REQUEST 128+64+1
20  REQUEST A+B*10
```

RESUME

Outline

Setting the GPIB ATN (attention) line to false.

Syntax

RESUME

Description

- If the RESUME statement is executed when the GPIB ATN (attention) line is true (Low level), the ATN line turns to false (High level). If the ATN line is already false, there is no change.
- The ATN line turns to true by the following statements.
  CLEAR
  LOCAL
  LOCAL LOCKOUT
  REMOTE
  SEND
  TRIGGER

  Since these statements leave the ATN line in true status, it may sometimes be necessary to set the ATN line to false by the RESUME statement.

Example

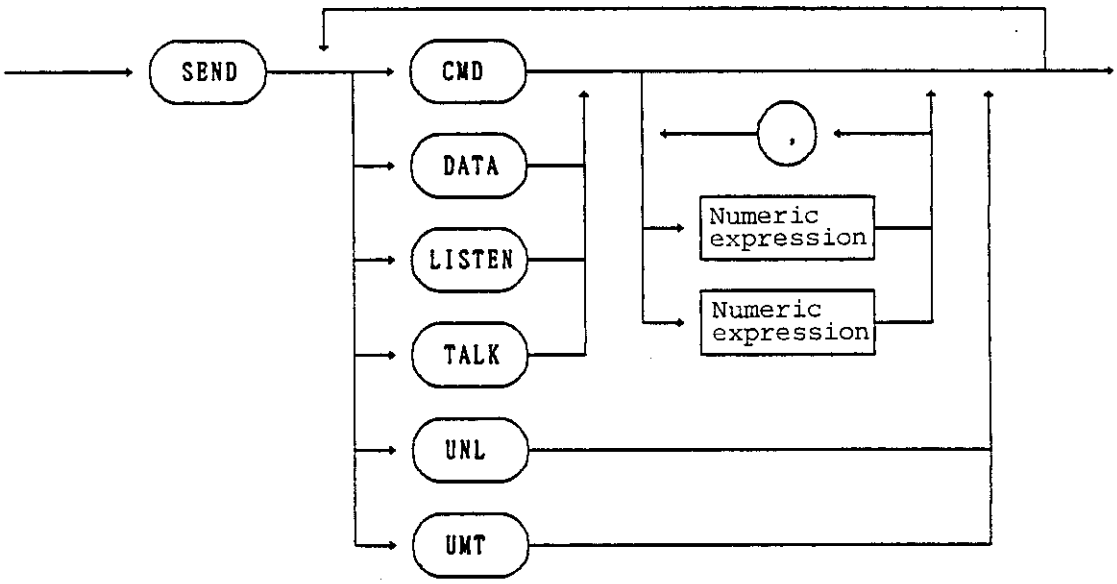
```
10  RESUME
```

SEND

Outline

Output of commands and data to the GPIB.

Syntax




```
SEND  <A>|<B>{,<A>|<B>}
<A>::=CMD | DATA | LISTEN | TALK[<C>{,<C>}]
<B>::=UNT | UNL
<C>::=Numerical expression | character string expression
```

Description

- Statement for independent sending of universal commands, address commands, and data to the GPIB.

  CMD : With the ATN (attention) line set to true (Low level), the given numerical value is sent to the GPIB. The numerical value, however, is converted to 8-bit binary data before being passed to the GPIB. Therefore, the range of values which can be handled must be 0 to 255. And numerical values with decimal places are automatically converted to integer numbers.

  DATA : With the ATN (attention) line set to false (High level), the given numerical values are sent to the GPIB. The numerical values handled in this case are the same as for "CMD".

LISTEN: The given numerical values are sent to the GPIB as LAG (listener address group). More than one numerical value can be specified.

TALK : The given numerical values are sent to the GPIB as TAG (talker address group). Only one numerical value can be specified.

UNT : The "UNT" (untalk) command is sent to the GPIB. The devices addressed to talk before this command is executed are unaddressed to talk.

UNL : The "UNL" (unlisten) command is sent to the GPIB. Devices addressed to listen before this command is executed are unaddressed to listen.

- Since the GPIB ATN (attention) line is left in true (Low level) status when the SEND statement is executed, it may sometimes be necessary to set the ATN line to false by executing the RESUME statement. Note, however, that ATN is false while DATA- is being executed.

Example

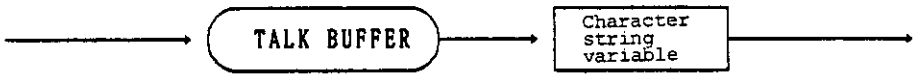
```
10  SEND UNT UNL LISTEN 1,2,3 TALK 4
20  SEND UNT CMD 10,200 DATA 30,54
30  SEND DATA "ABC",3,4
```

TALK BUFFER

Outline

Designation of the buffer to be used as talker.

Syntax



TALK BUFFER Character string variable

Description

- Direct input of data from the GPIB by OUTPUT statement is not possible after the control function has been delivered to another device by the PASS CONTROL statement, and while the TR4751 is not the system controller (that is, until the controller capability is received) since the bus cannot be controlled while there is no controller capability.
  While the controller capability is not delivered to the TR4751, data input is executed via the talker buffer.
  When the TALK BUFFER statement is executed, the specified character string variable is handled as the talker buffer, and when the TR4751 is addressed to talk, data stored in the talker buffer is output.
  The status function judges whether or not data has been output from the talker buffer (see Section 7.2.7-(2), (b) "Functions").
  If the size of the data to be stored in the talker buffer is greater than the array declaration value of the character string variable used as the talker buffer, any overflow data is disregarded.

Example

```
10 DIM A$(100)
20 TALK BUFFER A$
30 OUTPUT A$:A*B/C
40 IF BIT(STATUS,1)=0 THEN 40
50 DISP "DATA OUT"
```

CAUTION

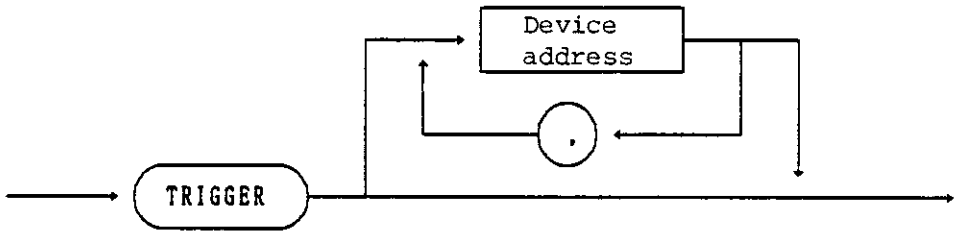
The uniline signal EOI is output together with the last byte as the delimiter for data output. Note that the "CR" and "LF" codes are not used as delimiters.

TRIGGER

Outline

Output of the GET (group execute trigger) command of the address command group (ACG) to specifically selected devices, or all devices connected to the GPIB.

Syntax



TRIGGER [Device address{,device address}]

Description

- If only the TRIGGER statement is executed without specifying a device address, only the GET (group execute trigger) address command is sent to the GPIB. In this case, the device where a trigger is to be applied is to be set in advance.
- If a device address is specified after TRIGGER, the GET command is sent only to the device specified by the device address.

Example

```
10  TRIGGER 1
20  TRIGGER
30  TRIGGER 2,A*100-J,30
```

List of Commands and Statements

Commands

AUTO[n],[m]	:	Automatic generation of line numbers
LIST[n]	:	Display of program list on the CRT
NEW	:	Erasure of previously entered program
PLIST	:	Output of program list to the printer
PRINTER	:	Designation of printer GPIB address
RUN	:	Program execution
SIZE	:	Display of remaining program area size

Statements

BEEP	:	Buzzer generation
CURSOR(X,Y)	:	Cursor control
DATA	:	Data storage
DIM	:	Array declaration
DISABLE INTR	:	Inhibition of interrupts
DISP	:	Display on the CRT
ENABLE INTR	:	Permitting interrupts
END	:	End of program
FOR-TO-STEP-NEXT	:	Loop processing
GET	:	Key inputs (without key input wait)
GOSUB	:	Branching to a subroutine
GOTO	:	Unconditional branching
IF-GOTO	:	Conditional branching
IF-THEN	:	Conditional judgement
INPUT	:	Key inputs (with key input wait)
LET	:	Substitution
OFF ERR	:	Inhibition of branching for interrupt generated by error generation
OFF KEY	:	Inhibition of branching for interrupt generated by key input
OFF SRQ	:	Inhibition of branching for interrupt generated by service request
ON ERR	:	Enabling of branching for interrupt generated by error generation
ON KEY	:	Enabling of branching for interrupt generated by key input
ON SRQ	:	Enabling of branching for interrupt generated by service request
PAUSE	:	Temporary halt of program execution
PRINT	:	Printing in printer
PRINTER	:	Designation of printer GPIB address
READ	:	Reading of data
REM (or !)	:	Program comment
RESTORE	:	Re-reading of data
SCLEAR	:	Erasure of entire CRT display
WAIT	:	Time delay

GPIB control statements

CLEAR	:	Device clearing
DELIMITER	:	Block delimiter designation
ENTER	:	Input from GPIB
INTERFACE CLEAR	:	Interface clearing
LISTEN BUFFER	:	Listener buffer designation
LOCAL	:	Remote control release
LOCAL LOCKOUT	:	Local lockout
OUTPUT	:	Output to GPIB
PASS CONTROL	:	Delivery of controller capability
REMOTE	:	Remote control
REQUEST	:	Service request output
RESUME	:	Uniline signal ATN switched OFF
SEND-DATA-CMD-TALK-LISTEN-UNT-UNL	:	Output of commands and data to the GPIB
TALK BUFFER	:	Talker buffer designation
TRIGGER	:	Output of group execute trigger

MEMO

8. CONNECTING RS-232C

The TR4751 is provided with not only the GPIB interface (see Chapter 7) but also the RS-232C interface. This enables the acquired data to be output on the RS-232C printer. The RS-232C interface has the mechanical and electric characteristics for connecting the data terminals to data communication devices standardized by the U.S. electronic industry association (EIA). For these characteristics, refer to the EIA specification book.

8.1 CONNECTOR AND SIGNAL TABLE

(1) Connector: 25-pin D-Sub connector (male type)

(2) Signal table

Pin number	Pin name	Pin number	Pin name
1	AA Security ground	14	
2	BA Sending data	15	
3	BB Receiving data	16	
4	CA Sending request	17	
5	CB Enabling signal sending	18	
6	CC Data set ready	19	
7	AB Signal ground	20	CD Data terminal ready
8		21	
9		22	
10		23	CI/CH Data signal speed selection
11			
12		24	
13		25	

8.2 CONNECTOR LOCATION

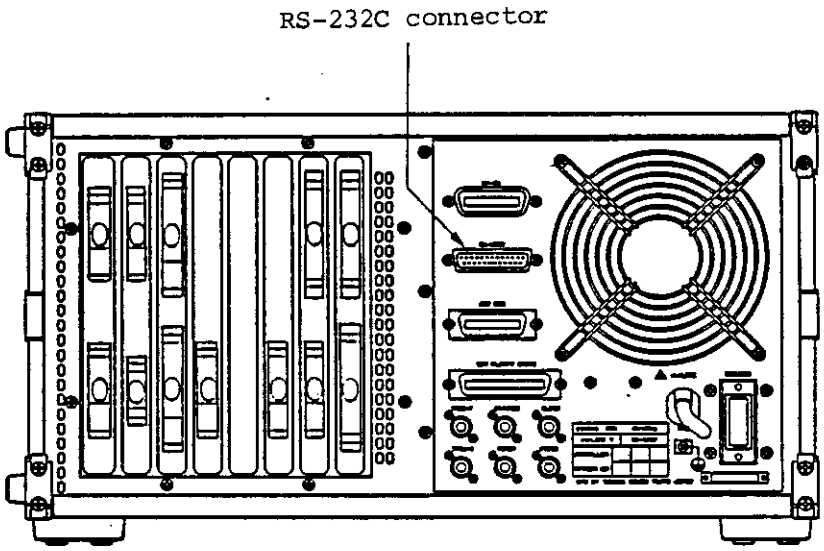


Figure 8-1 RS-232C Interface Connector

9. USING OPTIONS

The TR4751 is supported by mounting the optional low-speed acquisition section to extend the acquisition channels to 64 channels.

9.1 OPTIONAL CONFIGURATION

Table 9-1 lists the optional configurations. These articles are added to the standard configurations.

Table 9-1 Optional Configurations

Item	Model name	Remarks
Data probe	TR14701-01	Only for option 73
Data probe	TR14701-02	Only for option 73
Data probe	TR14701-03	
Clock qualifier probe	TR1472-04	

9.2 CONNECTING DATA PROBE AND CLOCK/QUALIFIER PROBE

Connect each probe to the connectors on the TR4751 rear panel as shown in Figure 9-1.

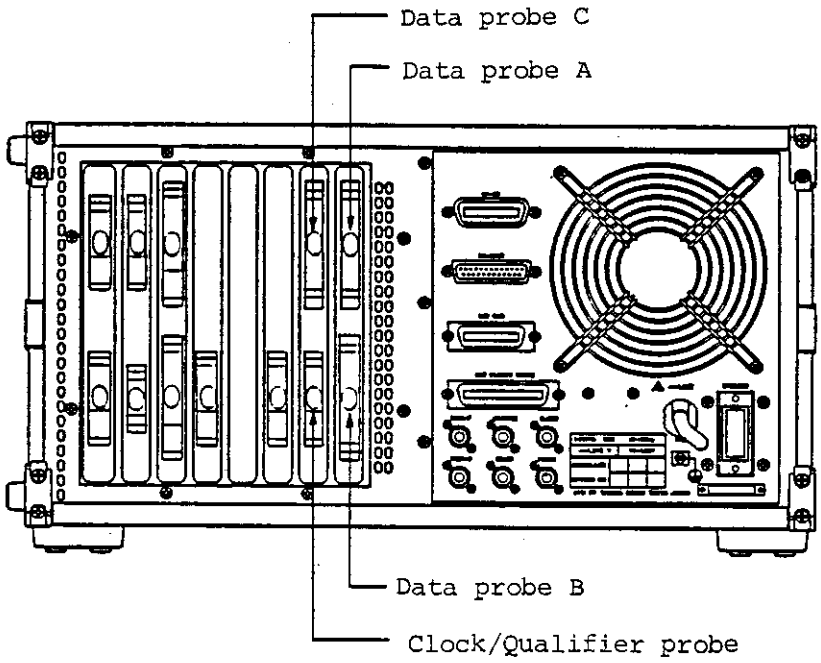


Figure 9-1 Connecting Probes

9.3 OPERATING LOW-SPEED ACQUISITION SECTION

Basically, the low-speed acquisition section operates in the same way as the high-speed acquisition section of a standard configuration. See Section 4.1 "ACQ spec (Acquisition Section Setting)".

9.3.1 Chnl spec (Input Channels Definition)

(1) Pod names
Each pod name corresponds to ACQ-C, B or A on the screen. One pod name contains 16 channels. CLK/CQ represents the clock/qualifier pod. This pod contains 3-clock and 3-qualifier channels.

(2) Selecting threshold voltage type
The following items can be selected as the threshold voltage:

TTL ECL VAR _____ _____ exit

TTL sets the threshold voltage to +1.40 V. ECL sets -1.30 V.
VAR enables setting the threshold voltage to any value by using the increment and decrement keys. The setting resolution is 100 mV. The threshold voltage must be within the range of -12.7 V to +12.7 V. The values on the lower column are displayed in reverse.

TTL ECL VAR increment decrement exit

(3) Threshold voltage value
Displayed in three digits.

(4) Labels
Specified by using the entry keys. A label must be up to four alphanumeric characters.

(5) Changing logical polarity
positive sets the positive logic (displaying "+"). negative sets the negative logic (displaying "-"). In the negative logic, the input logic is replaced with that displayed on the screen.

positive negative _____ _____ _____ exit

(6) Channel assignments
The channel assignments can be defined by using a pair of a pod name and a channel name. When the cursor is on the POD side, the following soft key labels are displayed:

insert delete POD-H POD-G exit
(When the channel assignment for the H/G probe is defined:)

insert delete POD-C POD-B POD-A exit
(When the channel assignment for the A/B/C probe is defined:)

POD-H POD-G POD-C POD-B POD-A exit
(When no channel assignment is defined:)

When the cursor is on the CH side, the following labels are displayed:

insert delete exit

insert moves the right channels of the cursor to the right and inserts a space at that position. This enables defining a new channel. delete deletes the channel pointed by the cursor and moves the right channels to the left.

CAUTION

(1) If you operate the soft keys too fast, they may overlap and an error occur.
(2) "POD-H, G" and "POD-C, B, A" cannot be combined in the channel assignments.

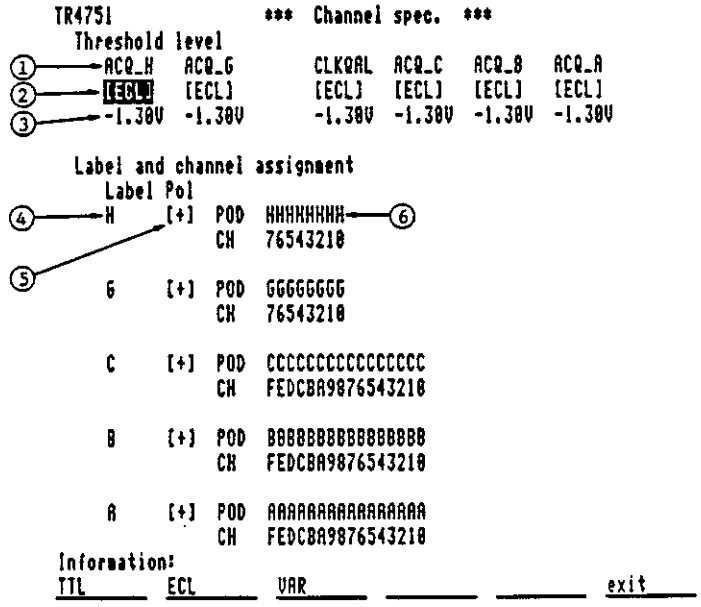


Figure 9-2 Chnl spec Screen

9.3.2 Trig spec (Trigger Conditions Setting)

"Trig spec" sets the trigger conditions, sampling clock and memory size.

(1) Setting sampling clock CLK2 (Clock)
The following items can be selected as the sampling clock:

EXIT	PG strobe	INT			exit

EXT specifies an external clock as the sampling clock. At this time, three clocks C1, C2 and C3 are input to the optional clock/qualifier probe (displaying as C3, C4 and C5 on the screen, respectively) and the logical sum and logical polarity of each clock can be selected.
PG strobe performs a sampling at the strobe (delay) timing defined in the "I/O spec" screen.
INT samples data asynchronously. The sampling rate must be within the range of 20 ns to 50 ms in 1-2-5 step.
At that time, the soft keys labels are displayed as follows:

EXT	PG strobe	INT	increment	decrement	exit

(2) Selecting memory size (Memory size)
The memory size is divided into long size and short size. When the cursor is moved to "Memory size:", the following soft key labels are displayed:

4K	1K				exit

(3) Selecting trigger mode (Trigger mode)
The trigger mode is divided into the following two types:

MATCH	RELEASE				exit

(4) Setting trigger position (Position)
"Position" sets the position of the trigger word defined in "Trigger mode:" and "Trigger:". The following four labels can be selected in "Position".

BEGIN	CENTER	END	DELAY		exit

BEGIN positions the trigger word at the first address.
CENTER positions the trigger word at the center address.
END positions the trigger word at the end address.
DELAY positions the trigger word at the specified value behind the first address.

Specifying a positive value positions the trigger word on the former half of memory. Specifying a negative value positions the trigger word on the latter half of memory. "Position" must be within the range of -4091 to 28659. If you made an error entry, press the "x" (don't care) key to blank it and reenter the correct value.

(5) Selecting trigger word display format
The trigger word can be displayed in the following three formats:

BIN OCT HEX exit

The defined format has no affect on the result display format.

(6) Setting trigger word
The desired trigger word can be set in the specified display format by using the entry keys.
The optional trigger format is of a 4-level sequential structure. In this format, the sequence of four words, "Enable1:", "Enable2:", "Trigger:" and "Disable:" is traced and a trigger occurs when the sequence is established. Figure 9-3 shows the flowchart of this sequence.

(7) Setting filter (Filter)
The TR4751 is provided with the capacity to filter the trigger word and identify the longer trigger word. The value in "Filter" corresponds to the number of clocks. The value must be within the range of 1 to 15.

(8) Setting the number of times for repeating trigger word (Event)
The TR4751 is equipped with the event counter. This repeats a trigger for the number specified in the trigger word. "Event" must be within the range of 1 to 32769.

(9) Linkage with other module
The trigger condition defined in this screen can be armed through the high-speed acquisition section or pattern generator.
ACQ activates the trigger condition for the low-speed acquisition section after the trigger condition for the high-speed acquisition section is established.
PAT activates the trigger condition for the low-speed acquisition section after the pattern generator executes "/TRG".
When OFF is selected, the trigger condition is not affected by the operations of other modules.

OFF PAT ACQ exit

(10) Setting clock qualifier (Clock qualifier)
Three qualifiers can be linked by using AND. A clock qualifier must be entered by using "1", "0" or "X". "X" has no affect on clock.

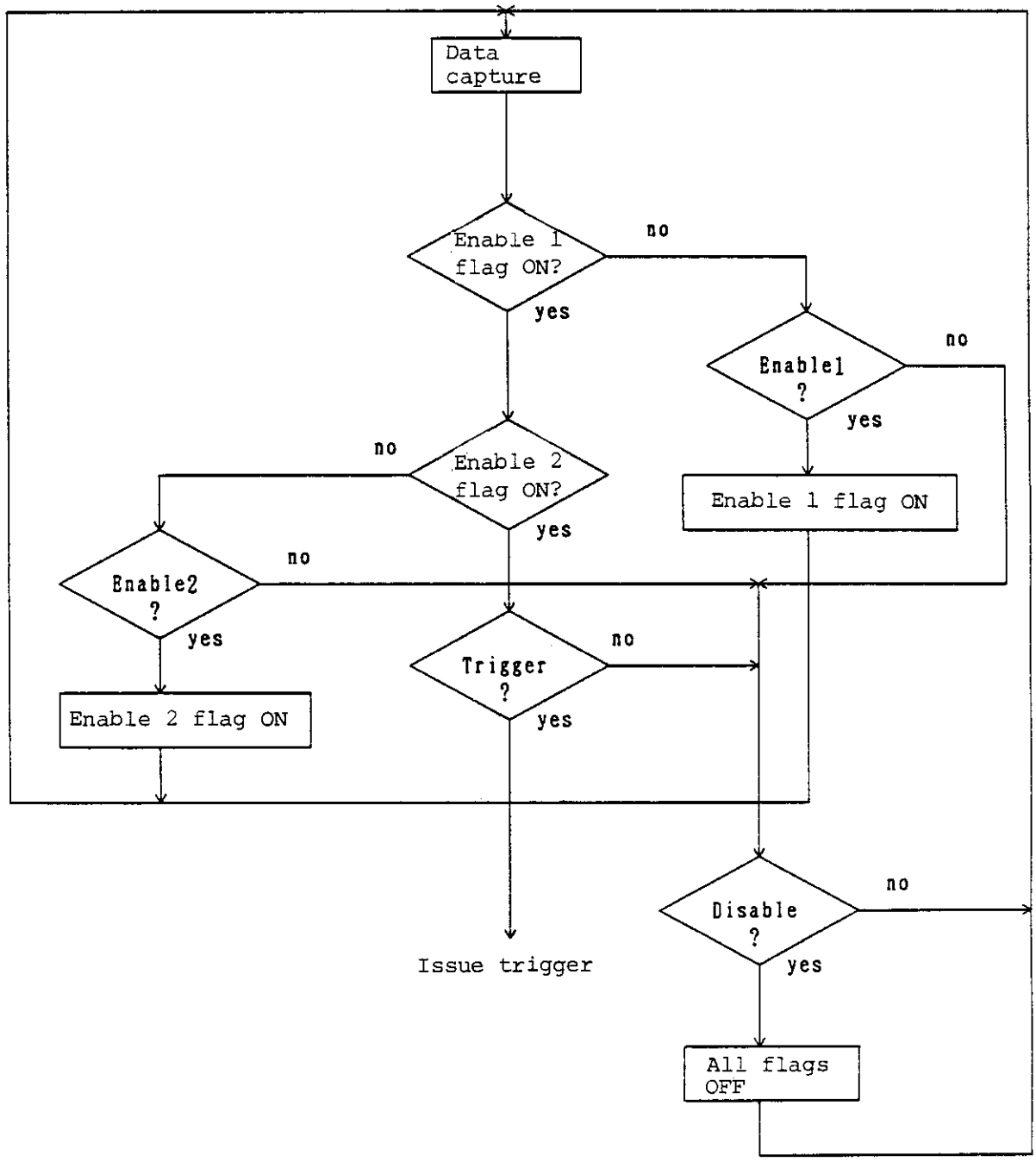


Figure 9-3 Sequential Trigger

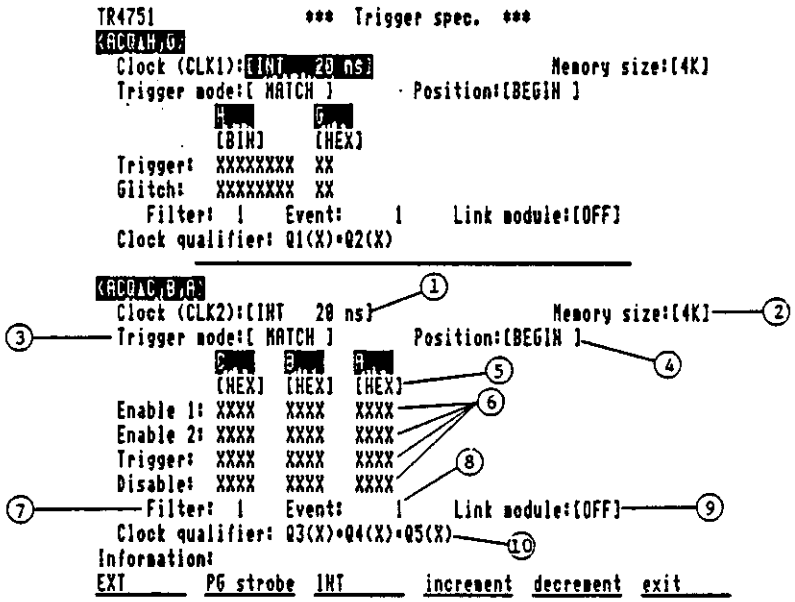


Figure 9-4 Trig spec Screen

10. TR47501 OPERATION

The pattern generating channels can be extended to 64 channels by using the TR47501 Extended Pattern Generator along with the TR4751.

The TR47501 is controlled by the TR4751. The TR47501 functions and the operating procedure are the same as the TR4751. For operating the TR47501, see Section 4.2 "PATT spec".

10.1 TR47501 CONFIGURATIONS

Table 10-1 lists the TR47501 configurations.

Table 10-1 TR47501 Configurations

No.	Item	Stock No.	Stan-dard	Option		Remarks
				71	72	
1	Extended pattern generator	TR47501	1			
2	Pattern generator cable B (in pairs for 0 to 7 chs and 8 to F chs)	A04703-12	1			For PG-B
3	Pattern generator cable C (in pairs for 0 to 7 chs and 8 to F chs)	A04703-13		1	1	For PG-C
4	Pattern generator cable D (in pairs for 0 to 7 chs and 8 to F chs)	A04703-14			1	For PG-D
5	Cable conversion adapter (in a set of ten cables) UM-QJ connector type	A04701-95	2	2	4	For cable B, C and D
6	PG external power cable B	A04702-32	1			For PG-B
7	PG external power cable C	A04702-33		1	1	For PG-C
8	PG external power cable D	A04702-34			1	For PG-D
9	Connection cable	MO-28	1			For EXT BUS
10	Connection cable	A04703-52	1			For PG CONT

10.2 CONNECTING THE TR47501 TO THE TR4751

The TR47501 is connected to the TR4751 by using two connection cables as shown in Figure 10-1.

CAUTION

(1) To connect the TR47501 to the TR4751, use the supplied accessory cables. Otherwise, the TR47501 may not operate normally.
(2) To activate the system, turn the TR47501 power on before turning the TR4751 power on. At the system activation, the TR4751 checks whether the TR47501 is connected correctly and also checks its configurations. If you turn the TR4751 power on before the TR47501, the TR47501 may not operate normally.

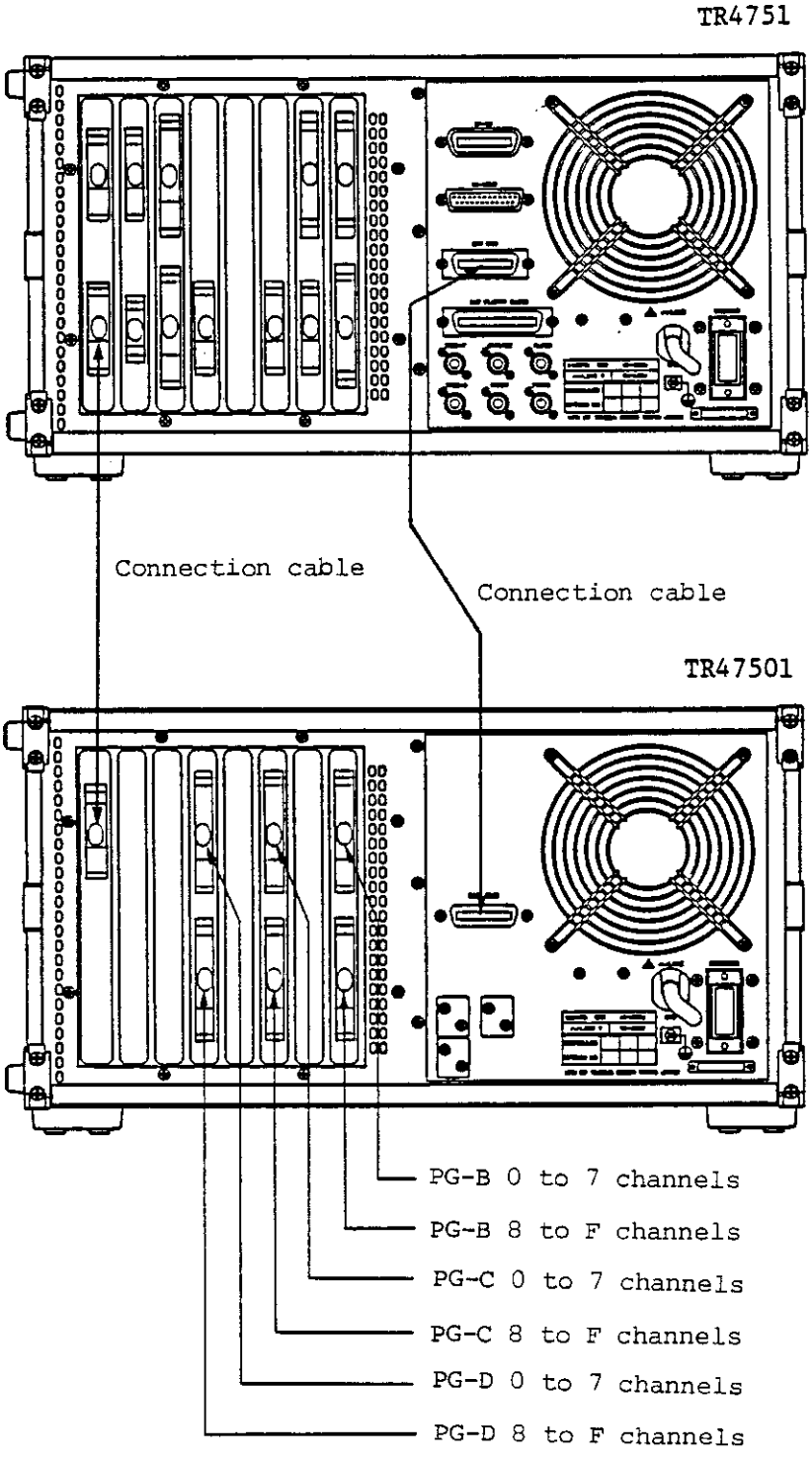


Figure 10-1 Connecting the TR47501 to the TR4751

MEMO

11. CAUTIONS ON STORAGE AND TRANSPORTATION

11.1 STORAGE

The storage temperature for the TR4751 is -10°C to +60°C. When the TR4751 is not used for a long time, protect it with a vinyl cover, or store the TR4751 in a cardboard box, and place it away from dumpness and direct sunlight.

The storage temperature for floppy disks is +10°C to +60°C.

11.2 TRANSPORTATION

When the TR4751 is transported elsewhere, use the original packaging material or a similar one. During transportation, insert the safety sheet into the floppy drive installed on the TR4751 so that it is not damaged by oscillation.

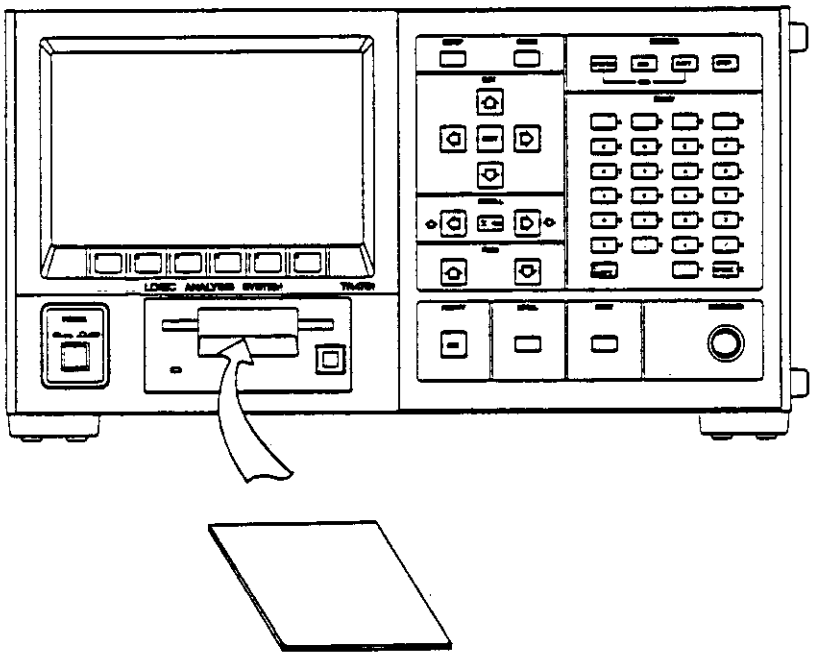


Figure 11-1 Inserting Safety Sheet

12. SPECIFICATIONS

12.1 TR4751 SPECIFICATIONS

12.1.1 Acquisition Section Performance

(1) High-speed acquisition section

Input channels	:	16 channels
Maximum sampling rate:	:	Internal 2.5 ns (400 MHz)
	:	External 10 ns (100 MHz)
Memory size:	:	Data 4K bit/channel
	:	Glitch 4K bit/channel (At 2.5 ns sampling, 8K bit data is allowed per channel and any glitch cannot be detected.)
Data setup time	:	0 ns
Data holding time	:	7 ns
Glitch minimum pulse width:		3 ns (At the sampling rate less than 200 MHz)
Clock qualifier	:	2 ch (Switches the polarity and selects the logical sum.)
Setup time	:	0 ns
Holding time	:	6 ns
Sampling clock	:	Internal 50 ms to 2.5 ns (in 1-2-5 step) External 2 channels (Switches the polarity and selects the logical sum.)
Trigger function	:	Release match pattern or mismatch pattern and sets the glitch mode. (The glitch mode is not set at 400 MHz sampling.)
Link module	:	Low-speed acquisition section, pattern generator
Filter	:	1 to 15 clocks
Event	:	1 to 32769 events
Delay	:	-4084 to 28666 clocks
Probe (Two TR14703s)		
Channels	:	16 channels for data, 2 channels for clock qualifiers, 2 channels for clocks
Impedance	:	1 MΩ//5 pF or less
Bandwidth	:	DC to 150 MHz
Sensitivity	:	500 mVp-p or less
Threshold voltage	:	TTL (+1.40 V), ECL (-1.30 V), VAR (+6.35 V to -6.35 V) ... Can be set in probe pod units.
Resolution	:	50 mV
Accuracy	:	±4% of the setting value ± 140 mV
Dynamic range	:	Threshold voltage ± 5 V
Maximum input voltage	:	± 30 V
Skew	:	3 ns or less (including probes)

12.1.2 Pattern Generator Performance

(1) 16-channel pattern generator

Output channels	:	16 channels
Maximum output rate	:	Program mode 20 ns (50 MHz) Memory mode 10 ns (100 MHz)
Clock		
Internal	:	50 ms to 10 ns (in 1-2-5 step)
External	:	DC to 10 ns ("Td" and "Tw" cannot be set. Those for strobe signals also cannot be set.)
Timing	:	8 phases
Resolution	:	Larger value of 1/10 rate or 10 ns
Memory size	:	1K bit/channel
Program size	:	1K step
Program instructions	:	8 instructions (CALL, RETURN, JUMP, HOLD, INC/DEC, REPEAT, HALT, IF)
External control instructions	:	Four instructions (INTERRUPT, INHIBIT, PAUSE, IF)
Pattern modes	:	Program mode, memory mode
Run modes	:	Repeat, single and step modes
Strobe output		
Delay	:	10 ms to 0 ns
Resolution	:	Larger value of three digits or 1 ns
Width	:	10 ms to 10 ns
Resolution	:	Larger value of three digits or 10 ns
Clock output (Duty rate)	:	1:1 at 10 ns, 1:3 at 20 ns, 3:7 at 50 ns, 2:3 at 100 ns, 1:4 at 200 ns, 2:23 at 500 ns At 1 µs or more, 1:4 with coefficient 1 1:9 with coefficient 2, 1:24 with coefficient 5
Pattern output (Data, strobes, clocks)		
Output range	:	Voh +6.35 V to −2 V (16 channels parallel) Vol +2 V to +6.35 V (16 channels parallel)
Output amplitude	:	0.6 V to 7 Vp-p
Resolution	:	50 mV
Accuracy	:	±3% ± 50 mV
Rising/falling time	:	5 ns or less (at 5 Vp-p output) 2 ns or less (at 1 Vp-p output)
Output impedance	:	Approx. 50 Ω (back termination)
Load impedance	:	≧ 1 kΩ
Connection format	:	In-axis connector
Output guard	:	Fuse
Skew	:	1 ns or less

12.1.3 Analysis Section Performance

(1) Timing display

Display channels	:	16 or 8 channels
Labels	:	5 labels (Channels whose labels are the same are automatically numbered.)
Changing display sequence	:	Display position can be changed in the label defined by the same clock.
Display channel scrolling	:	Scrolling or paging can be performed by every 16, 8 or 1 channel.
Magnification	:	Display can be magnified by 1 to 10000 in 1-2-5 step.
Cursor	:	Channel data pointed at by the cursor is displayed in binary. Interval or the number of clocks between the cursor and marker can be displayed.
Search function	:	Searches the specified word in the displayed channel.
Glitch mode	:	Displays the glitch for the channel sampled on the high-speed acquisition section.
Multi-clock display	:	Displays the data of the low/high-speed acquisition sections at the specified timing (only when they are sampled by internal clocks.).

(2) List display

Display channels	:	Up to 64 channels
Label	:	5 labels (displayed in each group)
Display format	:	Data is displayed in binary, octal or hexadecimal.
Search function	:	Searches the specified word in the displayed channel.
Scroll function	:	Scrolls data up and down.
Run mode	:	The measurement repeat mode also is provided.

(3) Compare display

Compare channels	:	Up to 64 channels
Compare data	:	Any acquisition data can be compared with any channel.
Window function	:	Masks the mismatch data in channel 1 to 15 against compare processing.
Display modes		
Map display	:	Displays the compare results of all data in the map format. The total number of error data also is displayed.
List display	:	Displays the reference data and inverts the mismatch data between the reference and acquisition data.

Reference data	:	Acquired data can be processed as the reference data and edited.
Run mode	:	Repeats a measurement depending on compare results.

(4) Graph display

Plots	:	Up to 50 plots for X axis Up to 16 plots for Y axis
Title	:	Up to 40 characters
Label on X/Y axes	:	Up to 6 characters
Setting range	:	Floating decimal portion; mantissa: four digits, exponent: ±9 powers
Compare function	:	Conform to compare display.
Parameters		
Internal	:	Vth, Voh, Vol, Td (Strobe)
External	:	Controlled by GPIB.

(5) Counter display

Measurement items	:	Maximum, minimum and average time, and cycle and frequency of each channel data are displayed in four digits (only when data is acquired by internal clocks.).
Cycle/frequency changing range	:	Cycle and frequency can be changed by up to 0 to 50% for instability in a cycle.

12.1.4 Interface

(1) GPIB

Interface functions	:	SH1, AH1, T6, L4, SR1, RL1, PP0, DC1, DT0, C1, C2, C3, C12, E1 (EOI; E2 for DAV)
Controller commands	:	AUTO, LIST, NEW, PLIST, PRINTER, RUN, SIZE
Statements	:	BEEP, CURSOR, DATA, DIM, DISABLE, INTR, DISP, ENABLE INTR, END, FOR-TO-STEP-NEXT, GET, GOSUB, GOTO, IF-GOTO, IF-THEN, INPUT, LET, OFF ERR, OFF KEY, OFF SRQ, ON ERR, ON KEY, ON SRQ, PAUSE, PRINT, PRINTER, READ, REM, RESTORE, SCLEAR, WAIT
GPIB control statements	:	CLEAR, DELIMITER, ENTER, LISTEN, BUFFER, LOCAL, LOCAL LOCKOUT, OUTPUT, PASS CONTROL, REMOTE, REQUEST, RESUME, SEND-DATA-CMD-TALK-LISTEN-UNT-UNL, TALK BUFFER, TRIGGER, INTERFACE CLEAR
Functions	:	ABS, CONV, COS, BIT, EXP, ERR, INT, LOG, LN, PI, SGN, SIN, SPOLL, WCONV, SQR, TAN
Statement function	:	STATUS

(2) RS-232C

Transfer speed	:	600, 1200, 2400, 4800, 9600 bps Full-duplex and synchronous start-stop system
Transfer data	:	Acquisition data, compare data, reference data, pattern data (only data)
RS-232C printer	:	Prints acquisition data list, reference data list, compare data list and pattern program list
Connector	:	Dsub connector

(3) Composite video output

Von	:	1.4 to 1.6 V
Voff	:	0.3 to 0.7 V
Vsync	:	0 to 0.1 V
Horizontal cycle	:	63.5 µs ± 0.1 µs
Vertical cycle	:	16.5 ms ± 0.1 ms
Connector	:	BNC connector

12.1.5 Floppy Drive

Capacity	:	632.5K bytes at formatting (1M byte at unformatting)
Disk type	:	Two types are provided (System disk and user disk).
File type	:	Eight types are provided (for acquisition parameters, acquisition data, display parameters, reference data, pattern generator programs/parameters, GPIB controller programs/parameters, ALL and SYS).
Operations	:	COPY, SAVE, DELETE, RENAME, FORMAT GET, STATUS, DIRECTORY

12.1.6 General Specifications

CRT

Size	:	8 inches
Display method	:	Raster scan display
Number of characters	:	64 characters x 24 lines per screen
Phosphor	:	B39

Operation environment (excluding floppy disks)

Ambient temperature	:	+5°C to +40°C
Relative humidity	:	80% or less

Storage temperature (excluding floppy disks)

Ambient temperature	:	-10°C to +60°C

Operation environment for floppy disks

Ambient temperature	:	+10°C to +60°C
Relative humidity	:	80% or less
Storage temperature	:	+10°C to +60°C

Power source		
Voltage	:	90 V to 132 Vac (198 V to 250 Vac: Option 40)
Frequency	;	48 to 63 Hz
Power consumption	:	700 VA or less (Normal) 1000 VA or less (Maximum)
Dimensions (mm)	:	Approx. 420 (W) x 220 (H) x 550 (D)
Weight	:	33 kg or less (Maximum)

12.2 OPTIONS AND ACCESSORIES

12.2.1 Options (Low-speed Acquisition Module)

Input channels	:	16 ch. (OPT. 4751+71)
	:	48 ch. (OPT. 4751+73)
Maximum sampling rate	:	Internal 20 ns (50 MHz) External 20 ns (50 MHz)
Memory size	:	4K bit/channel
Data setup time	:	12 ns
Data holding time	:	3 ns
Clock qualifiers	:	3 channels are provided. (Polarity can be switched and AND can be selected.)
Setup time	:	2 ns
Holding time	:	12 ns
Clocks		
Internal	:	50 ms to 20 ns (in 1-2-5 step)
External	:	0 to 20 ns (DC to 50 MHz). 3 channels are provided. (Polarity can be switched and AND can be selected.)
Trigger	:	Releases match or mismatch patterns. Four-level sequential trigger is allowed by combining enable2 and reset1. Trigger can be linked to high-speed acquisition section and pattern generator.
Filter	:	1 to 15 clocks
Event	:	1 to 32769 events
Delay	:	-4091 to 28659 clocks
Probe		
Input channels	:	Data 16 ch. (OPT. 4751+71) 48 ch. (OPT. 4751+73) Clock qualifier 3 channels Clock 3 channels
Input impedance	:	1 MΩ//8 pF or less
Bandwidth	:	DC 50 MHz
Sensitivity	:	200 mVp-p
Threshold voltage	:	TTL (+1.40 V), ECL (-1.30 V), VAR (+12.0 V to -12.0 V) ... for every 16 channels
Resolution	:	100 mV
Accuracy	:	±4% ± 140 mV
Dynamic range	:	Threshold voltage ± 10 V
Maximum input voltage	:	±50 V
Skew	:	5 ns or less (including probes)

12.2.2 TR47501 (Extended Pattern Generator)

Channels	:	16 ch. 32 ch. (OPT. 47501+71) 48 ch. (OPT. 47501+72)
Maximum output rate	:	20 ns (50 MHz) for program mode 10 ns (100 MHz) for memory mode
Timing	:	8 phases
Resolution	:	Larger value of 1/10 rate or 10 ns
Memory size	:	4K bit/channel (1K step/channel for program mode)
Program instructions	:	Same as those for TR4751.
External instructions	:	Same as those for TR4751.
Strobe outputs	:	Same as those for TR4751.
Clock outputs	:	Same as those for TR4751.
Pattern outputs (data)		
Output range	:	Voh +7 V to -2 V Vol +2 V to -7 V
Output amplitude	:	0.6 V to 7 Vp-p
Resolution	:	50 mV (16 channels parallel)
Accuracy	:	±3% ± 50 mV
Rising/falling time	:	5 ns or less (at 5 Vp-p output) 2 ns or less (at 1 Vp-p output)
Output impedance	:	Approx. 50 Ω (Back termination)
Load impedance	:	1 kΩ
Connection format	:	In-axis connector
Output guard	:	Fuse
Skew	:	1 ns or less
General specifications		
Operation environment		
Ambient temperature	:	+5°C to +40°C
Relative humidity	:	85% or less
Storage temperature		
Ambient temperature	:	-20°C to +60°C
Power source		
Voltage	:	90 V to 132 Vac (198 V to 250 Vac: Option 40)
Frequency	:	48 to 63 Hz
Power consumption	:	900 VA or less (Normal) 1000 VA or less (Maximum)
Dimensions (mm)	:	Approx. 420 (W) x 220 (H) x 550 (D)
Weight	:	28 kg or less (Maximum)

LIST OF FIGURES

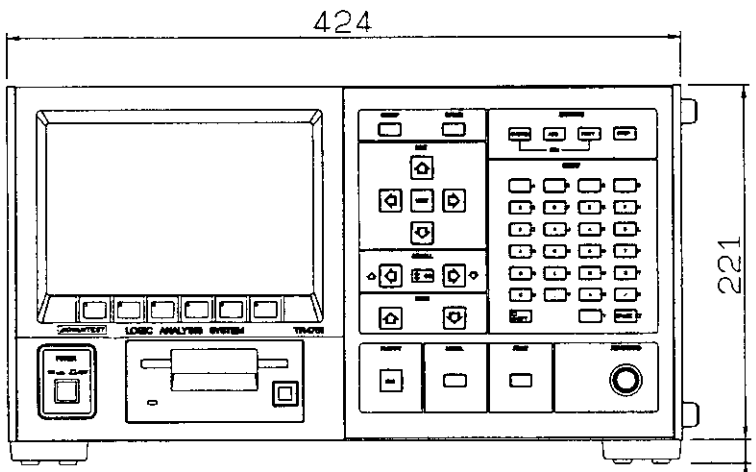
LIST OF FIGURES (Cont'd)

LIST OF TABLES

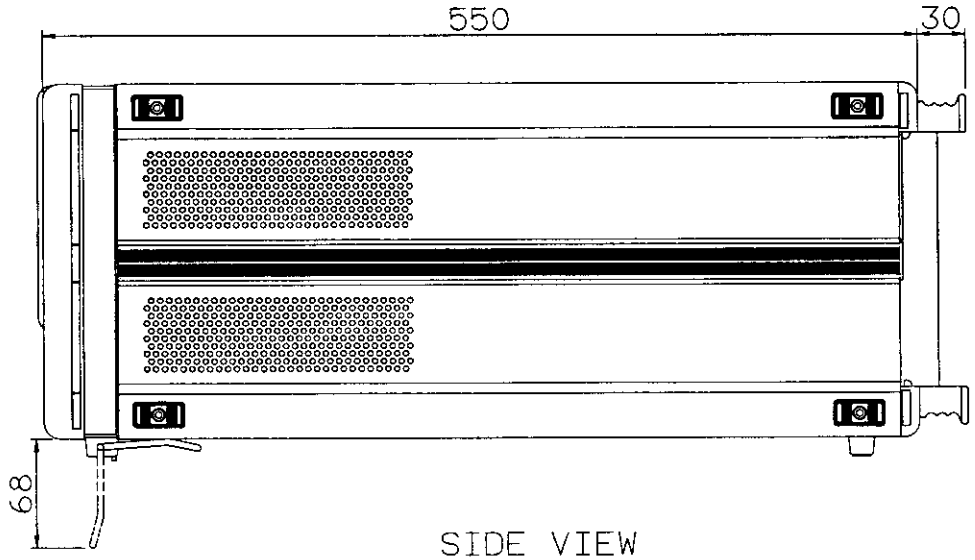
LIST OF TABLES

Example No.	Title	Page

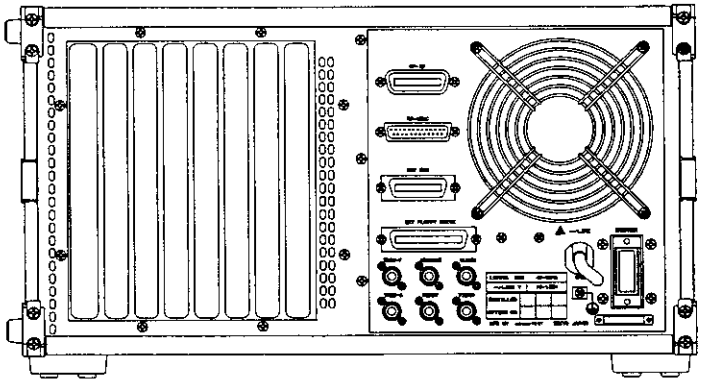
(No example numbers are assigned in this manual.)



FRONT VIEW



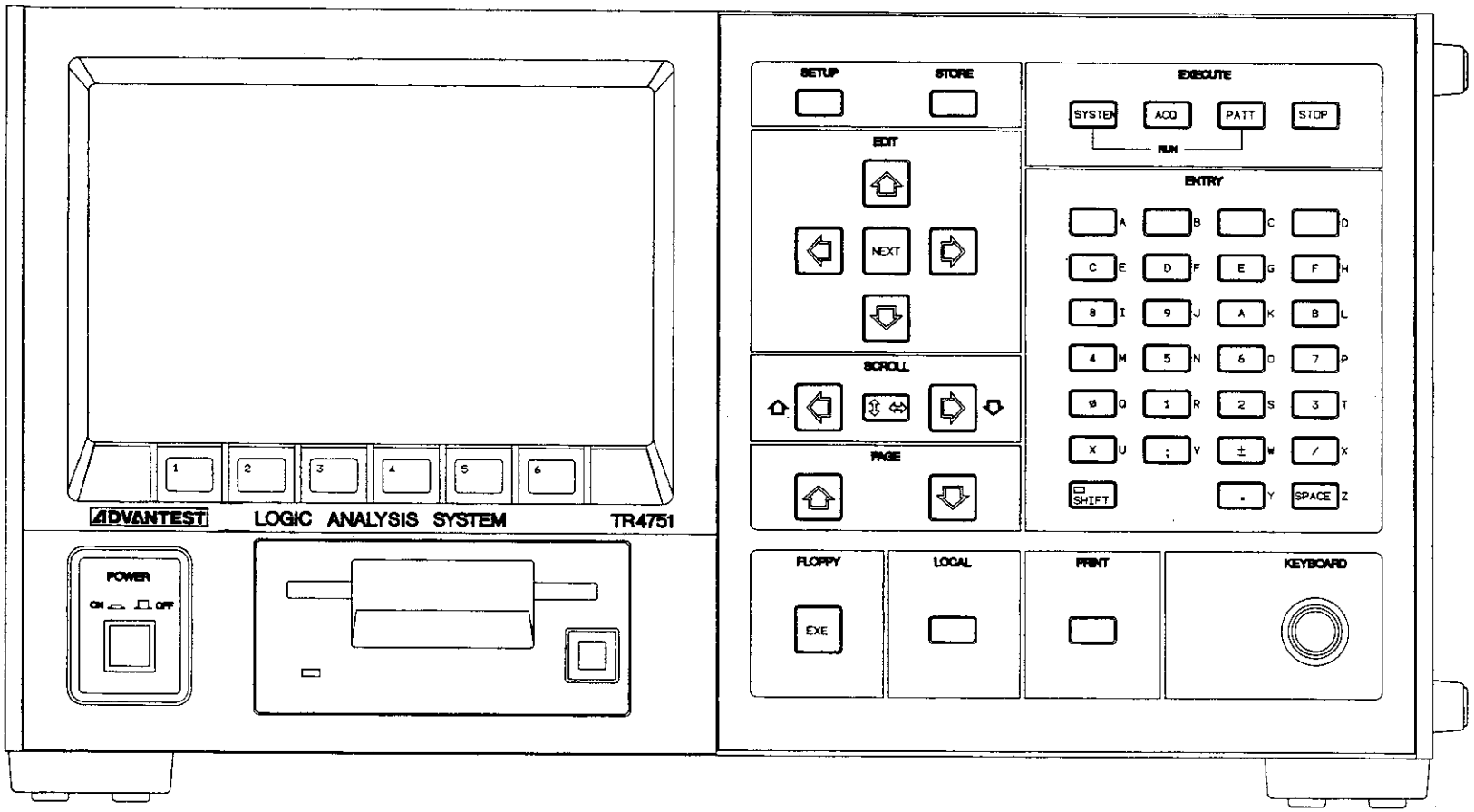
SIDE VIEW



REAR VIEW

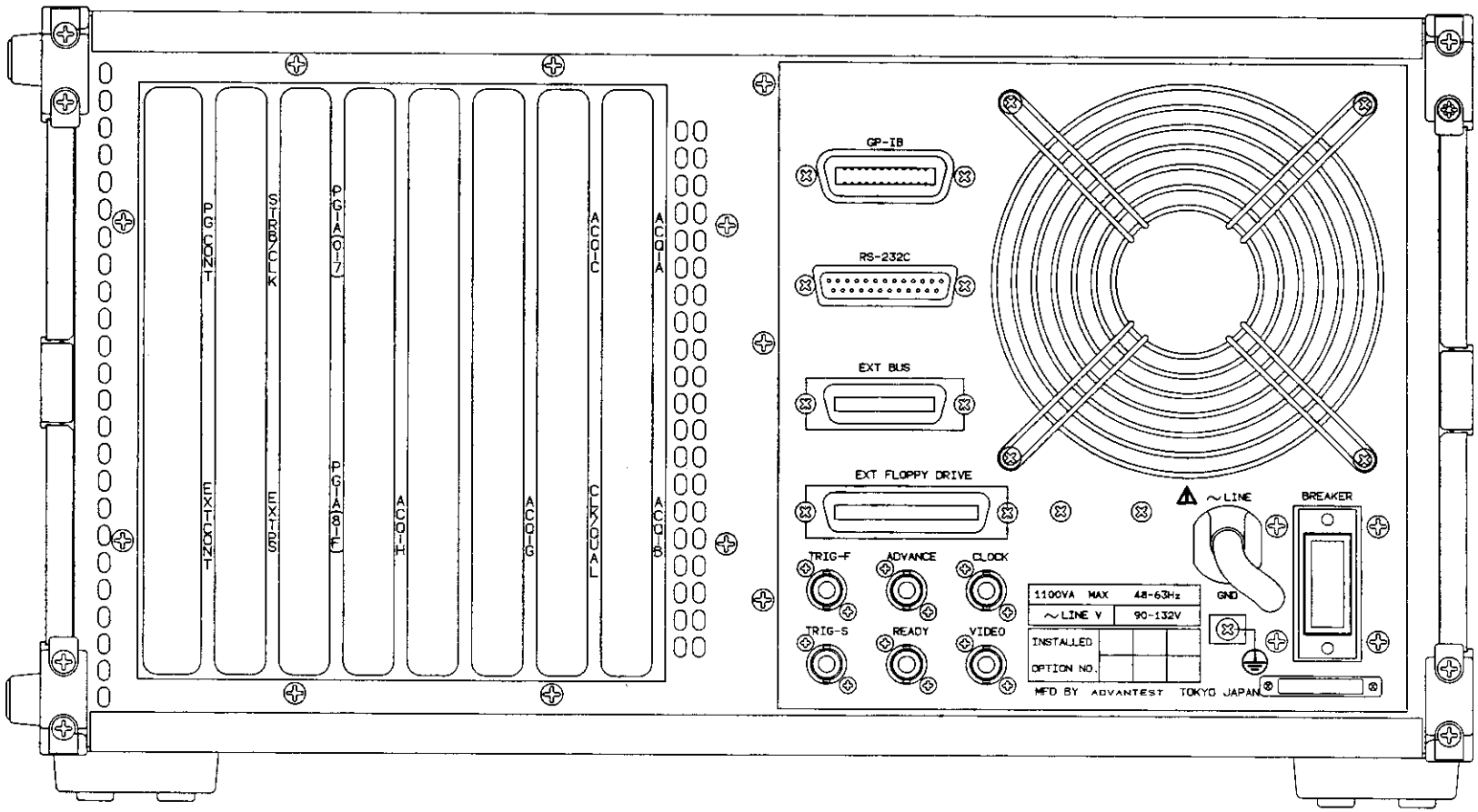
4751EXT1-508-A

TR4751
EXTERNAL VIEW



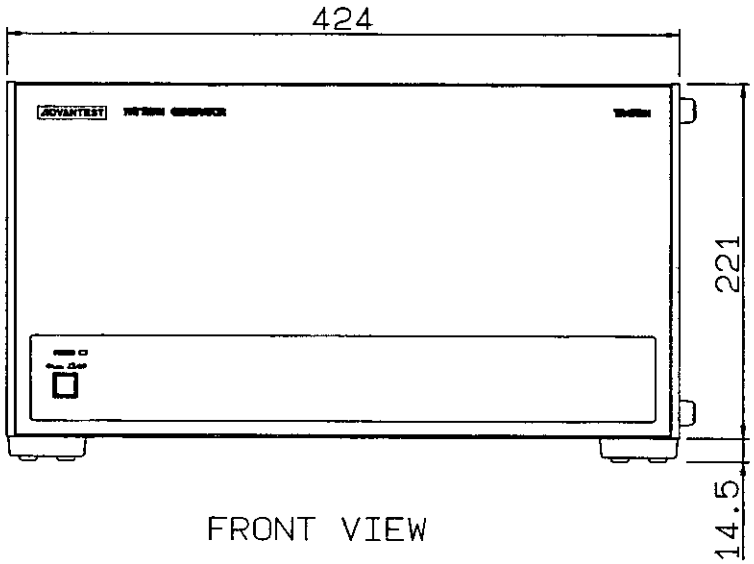
TR4751 FRONT VIEW

4751EXT2-508-A

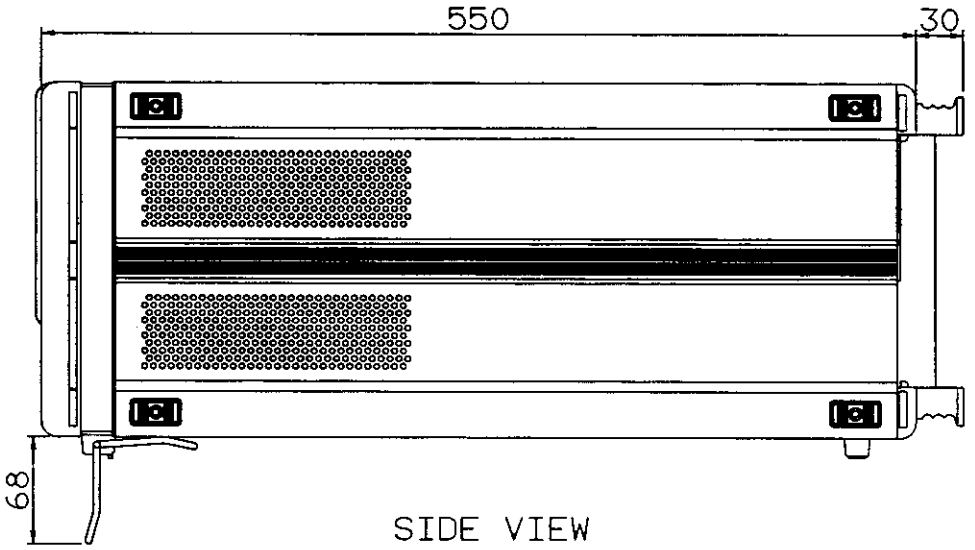


TR4751 REAR VIEW

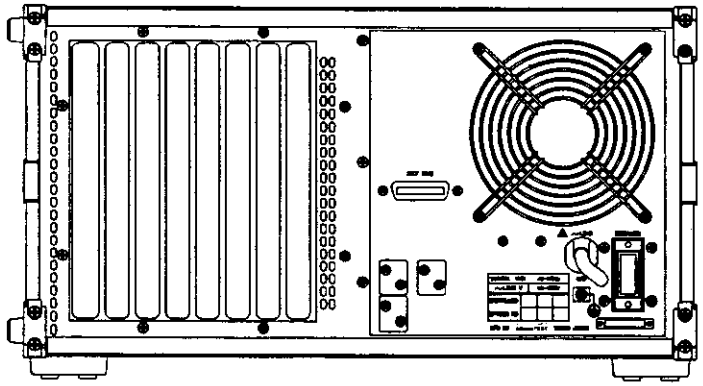
4751EXT3-508-A



FRONT VIEW



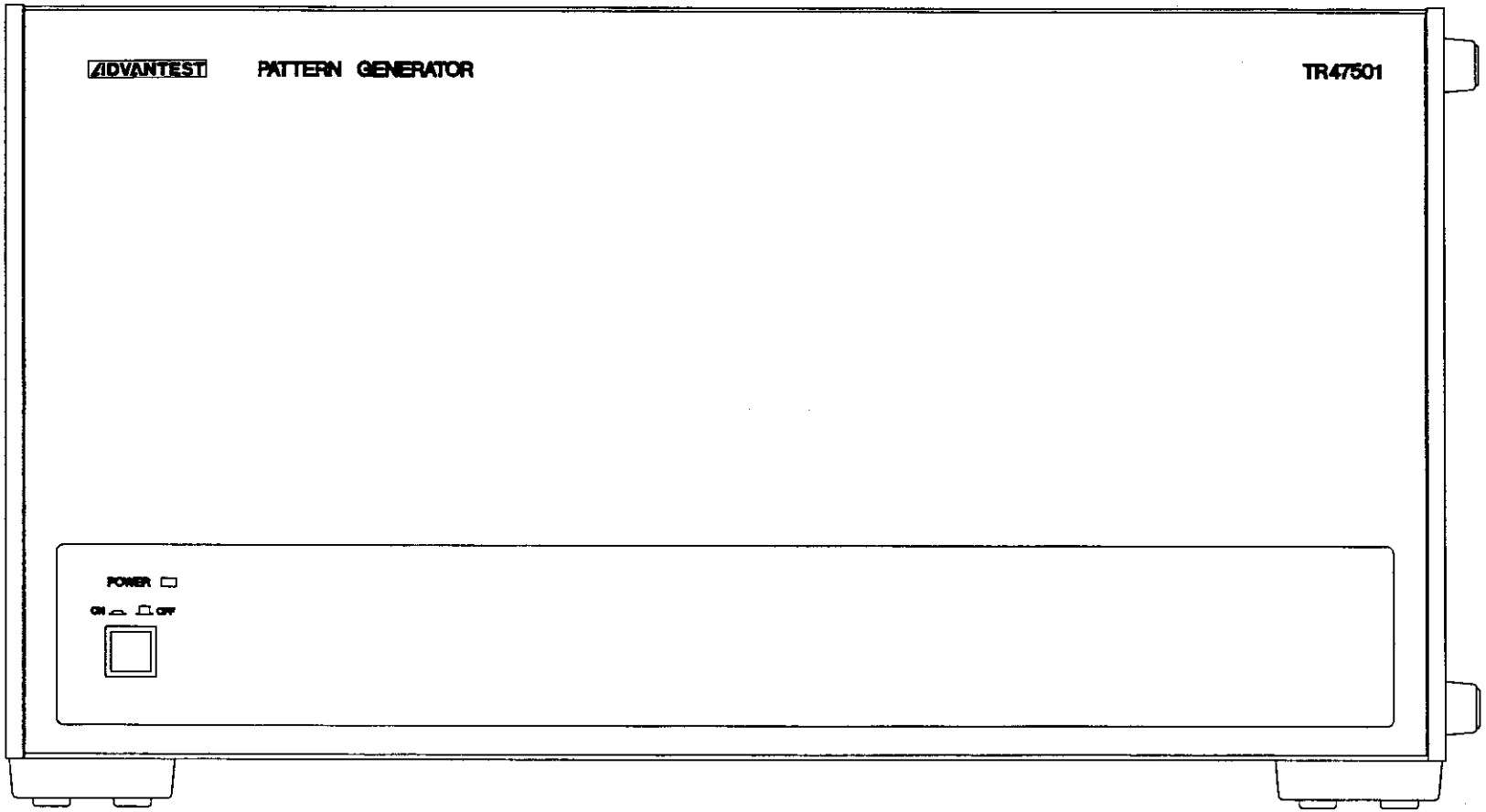
SIDE VIEW



REAR VIEW

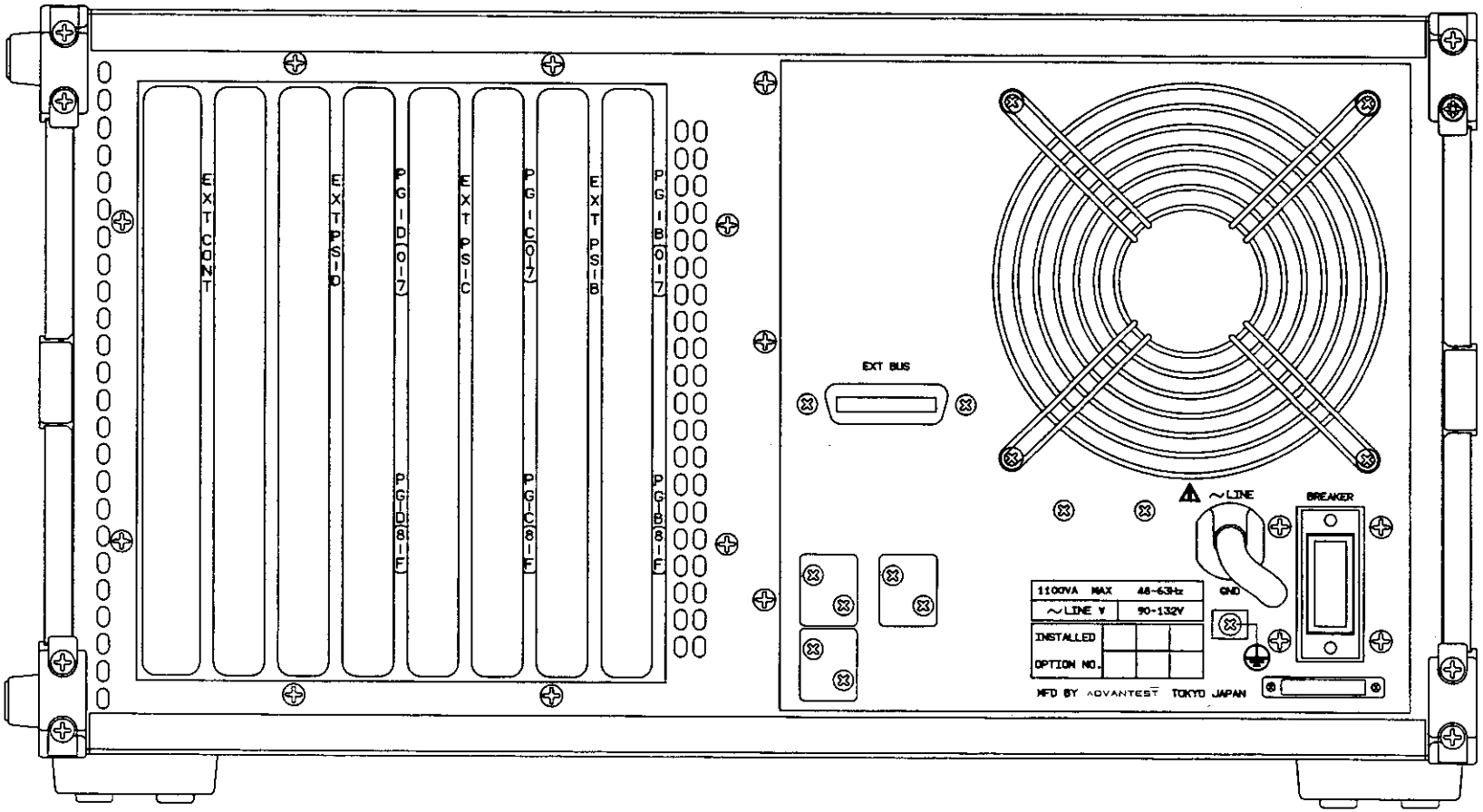
47501EXT1-508-A

TR47501
EXTERNAL VIEW



TR47501 FRONT VIEW

47501EXT2-508-A



TR47501 REAR VIEW

47501EXT3-508-A

IMPORTANT INFORMATION FOR ADVANTEST SOFTWARE

PLEASE READ CAREFULLY: This is an important notice for the software defined herein. Computer programs including any additions, modifications and updates thereof, operation manuals, and related materials provided by Advantest (hereafter referred to as "SOFTWARE"), included in or used with hardware produced by Advantest (hereafter referred to as "PRODUCTS").

SOFTWARE License

All rights in and to the SOFTWARE (including, but not limited to, copyright) shall be and remain vested in Advantest. Advantest hereby grants you a license to use the SOFTWARE only on or with Advantest PRODUCTS.

Restrictions

(1) You may not use the SOFTWARE for any purpose other than for the use of the PRODUCTS.
(2) You may not copy, modify, or change, all or any part of, the SOFTWARE without permission from Advantest.
(3) You may not reverse engineer, de-compile, or disassemble, all or any part of, the SOFTWARE.

Liability

Advantest shall have no liability (1) for any PRODUCT failures, which may arise out of any misuse (misuse is deemed to be use of the SOFTWARE for purposes other than it's intended use) of the SOFTWARE. (2) For any dispute between you and any third party for any reason whatsoever including, but not limited to, infringement of intellectual property rights.

LIMITED WARRANTY

1. Unless otherwise specifically agreed by Seller and Purchaser in writing, Advantest will warrant to the Purchaser that during the Warranty Period this Product (other than consumables included in the Product) will be free from defects in material and workmanship and shall conform to the specifications set forth in this Operation Manual.

2. The warranty period for the Product (the "Warranty Period") will be a period of one year commencing on the delivery date of the Product.

3. If the Product is found to be defective during the Warranty Period, Advantest will, at its option and in its sole and absolute discretion, either (a) repair the defective Product or part or component thereof or (b) replace the defective Product or part or component thereof, in either case at Advantest's sole cost and expense.

4. This limited warranty will not apply to defects or damage to the Product or any part or component thereof resulting from any of the following:

   (a) any modifications, maintenance or repairs other than modifications, maintenance or repairs (i) performed by Advantest or (ii) specifically recommended or authorized by Advantest and performed in accordance with Advantest 's instructions;

   (b) any improper or inadequate handling, carriage or storage of the Product by the Purchaser or any third party (other than Advantest or its agents);

   (c) use of the Product under operating conditions or environments different than those specified in the Operation Manual or recommended by Advantest, including, without limitation, (i) instances where the Product has been subjected to physical stress or electrical voltage exceeding the permissible range and (ii) instances where the corrosion of electrical circuits or other deterioration was accelerated by exposure to corrosive gases or dusty environments;

   (d) use of the Product in connection with software, interfaces, products or parts other than software, interfaces, products or parts supplied or recommended by Advantest;

   (e) incorporation in the Product of any parts or components (i) provided by Purchaser or (ii) provided by a third party at the request or direction of Purchaser or due to specifications or designs supplied by Purchaser (including, without limitation, any degradation in performance of such parts or components);

   (f) Advantest's incorporation or use of any specifications or designs supplied by Purchaser;

   (g) the occurrence of an event of force majeure, including, without limitation, fire, explosion, geological change, storm, flood, earthquake, tidal wave, lightning or act of war; or

   (h) any negligent act or omission of the Purchaser or any third party other than Advantest.

5. **EXCEPT TO THE EXTENT EXPRESSLY PROVIDED HEREIN, ADVANTEST HEREBY EXPRESSLY DISCLAIMS, AND THE PURCHASER HEREBY WAIVES, ALL WARRANTIES, WHETHER EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, (A) ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE AND (B) ANY WARRANTY OR REPRESENTATION AS TO THE VALIDITY, SCOPE, EFFECTIVENESS OR USEFULNESS OF ANY TECHNOLOGY OR ANY INVENTION.**

6. **THE REMEDY SET FORTH HEREIN SHALL BE THE SOLE AND EXCLUSIVE REMEDY OF THE PURCHASER FOR BREACH OF WARRANTY WITH RESPECT TO THE PRODUCT.**

7. **ADVANTEST WILL NOT HAVE ANY LIABILITY TO THE PURCHASER FOR ANY INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR PUNITIVE DAMAGES, INCLUDING, WITHOUT LIMITATION, LOSS OF ANTICIPATED PROFITS OR REVENUES, IN ANY AND ALL CIRCUMSTANCES, EVEN IF ADVANTEST HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES AND WHETHER ARISING OUT OF BREACH OF CONTRACT, WARRANTY, TORT (INCLUDING, WITHOUT LIMITATION, NEGLIGENCE), STRICT LIABILITY, INDEMNITY, CONTRIBUTION OR OTHERWISE. TORT (INCLUDING, WITHOUT LIMITATION, NEGLIGENCE), STRICT LIABILITY, INDEMNITY, CONTRIBUTION OR OTHERWISE.**

8. **OTHER THAN THE REMEDY FOR THE BREACH OF WARRANTY SET FORTH HEREIN, ADVANTEST SHALL NOT BE LIABLE FOR, AND HEREBY DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW ANY LIABILITY FOR, DAMAGES FOR PRODUCT FAILURE OR DEFECT, WHETHER ARISING OUT OF BREACH OF CONTRACT, TORT (INCLUDING, WITHOUT LIMITATION, NEGLEGENCE), STRICT LIABILITY, INDEMNITY, CONTRIBUTION OR OTHERWISE.**

CUSTOMER SERVICE DESCRIPTION

In order to maintain safe and trouble-free operation of the Product and to prevent the incurrence of unnecessary costs and expenses, Advantest recommends a regular preventive maintenance program under its maintenance agreement.

Advantest's maintenance agreement provides the Purchaser on-site and off-site maintenance, parts, maintenance machinery, regular inspections, and telephone support and will last a maximum of ten years from the date the delivery of the Product. For specific details of the services provided under the maintenance agreement, please contact the nearest Advantest office listed at the end of this Operation Manual or Advantest 's sales representatives.

Some of the components and parts of this Product have a limited operating life (such as, electrical and mechanical parts, fan motors, unit power supply, etc.). Accordingly, these components and parts will have to be replaced on a periodic basis. If the operating life of a component or part has expired and such component or part has not been replaced, there is a possibility that the Product will not perform properly. Additionally, if the operating life of a component or part has expired and continued use of such component or part damages the Product, the Product may not be repairable. Please contact the nearest Advantest office listed at the end of this Operation Manual or Advantest's sales representatives to determine the operating life of a specific component or part, as the operating life may vary depending on various factors such as operating condition and usage environment.

SALES & SUPPORT OFFICES

Advantest Korea Co., Ltd.
22BF, Kyobo KangNam Tower,
1303-22, Seocho-Dong, Seocho-Ku, Seoul #137-070, Korea
Phone: +82-2-532-7071
Fax: +82-2-532-7132

Advantest (Suzhou) Co., Ltd.
Shanghai Branch Office:
Bldg. 6D, NO.1188 Gumei Road, Shanghai, China 201102 P.R.C.
Phone: +86-21-6485-2725
Fax: +86-21-6485-2726

Shanghai Branch Office:
406/F, Ying Building, Quantum Plaza, No. 23 Zhi Chun Road,
Hai Dian District, Beijing,
China 100083
Phone: +86-10-8235-3377
Fax: +86-10-8235-6717

Advantest (Singapore) Pte. Ltd.
438A Alexandra Road, #08-03/06
Alexandra Technopark Singapore 119967
Phone: +65-6274-3100
Fax: +65-6274-4055

Advantest America, Inc.
3201 Scott Boulevard, Suite, Santa Clara, CA 95054, U.S.A
Phone: +1-408-988-7700
Fax: +1-408-987-0691

ROHDE & SCHWARZ Europe GmbH
Mühldorfstraße 15 D-81671 München, Germany
(P.O.B. 80 14 60 D-81614 München, Germany)
Phone: +49-89-4129-13711
Fax: +49-89-4129-13723

ADVANTEST®

http://www.advantest.co.jp

ADVANTEST CORPORATION
Shin-Marunouchi Center Building, 1-6-2 Marunouchi, Chiyoda-ku, Tokyo 100-0005, Japan
Phone: +81-3-3214-7500