

---

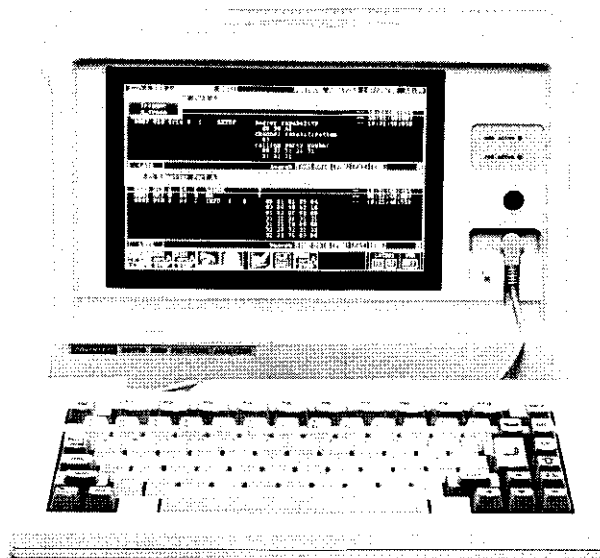
**ADVANTEST®**  
ADVANTEST CORPORATION

---

**INSTRUCTION  
MANUAL**  
**D5111B**  
**ISDN PROTOCOL ANALYZER**

MANUAL NUMBER OEA00 9102

---



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. Govern-  
ment under its Export Control Law.



---

## TABLE OF CONTENTS

1. OUTLINE .....	1-1
1.1 How to use This Operation Manual .....	1-2
1.2 Product Outline .....	1-3
1.3 Before Use .....	1-4
1.3.1 Confirmation of accessories supplied .....	1-4
1.3.2 Operating environment .....	1-5
1.3.3 Supply voltage .....	1-6
1.3.4 Power cable .....	1-7
1.3.5 Fuse replacement .....	1-8
1.3.6 Keyboard attachment or detachment .....	1-9
1.3.7 Power cable storage .....	1-10
1.4 Panels, and Location and Functions of Each Part .....	1-11
1.4.1 Front panel .....	1-11
1.4.2 Right side panel .....	1-12
1.4.3 Left side panel .....	1-13
1.4.4 Rear panel .....	1-15
1.4.5 Keyboard .....	1-16
1.5 Basic Operation .....	1-18
1.5.1 Function configuration .....	1-19
1.5.2 Screen configuration and functions .....	1-20
1.5.3 Function module selection .....	1-22
1.5.4 Popup menu .....	1-23
1.5.5 File name specification .....	1-25
1.6 Operation Example .....	1-27
1.6.1 Translation display .....	1-27
1.6.2 Simulation .....	1-31
1.7 Start-Up File .....	1-36
1.7.1 Command description .....	1-36
1.7.2 Start-up file creation .....	1-38
1.8 Connection with Equipment to be Measured .....	1-41
1.8.1 Pin number and functions .....	1-41
1.8.2 Connector pin number .....	1-42
1.8.3 Feeding .....	1-42
1.9 Performance .....	1-43

---

2. MONITOR .....	2-1
2.1 Preparation for Monitoring .....	2-2
2.1.1 Configuration setting .....	2-2
2.1.2 Monitor loading .....	2-4
2.1.3 Trigger condition setting .....	2-5
2.1.4 Selective trace setting .....	2-10
2.1.5 Translation screen setting .....	2-13
2.2 Real-Time Display .....	2-14
2.2.1 Real-time trace operation .....	2-14
2.2.2 Display omission .....	2-14
2.3 History Display .....	2-15
2.3.1 History display method .....	2-15
2.3.2 Search function .....	2-15
2.3.3 Redisplay by translation layer selection .....	2-18
2.4 Translation Screen .....	2-19
2.4.1 Translation screen description (common section) .....	2-19
2.4.2 Layer-2 (LAY2) translation display .....	2-20
2.4.3 Layer-3 (LAY3) translation display .....	2-21
2.4.4 Layer-23 (LAY23) translation display .....	2-22
2.4.5 Layer-2 full (LAY2 FULL) translation display .....	2-23
2.4.6 Layer-3 full (LAY3 FULL) translation display .....	2-23
2.4.7 Layer-23 full (LAY23 FULL) translation display .....	2-24*
2.4.8 Mark in translation display .....	2-24*

---

3. SIMULATION .....	3-1
3.1 Simulation language .....	3-2
3.1.1 PSL51 .....	3-2
3.1.2 PSL51 language specification .....	3-2
3.2 Programming structure .....	3-3
3.2.1 PSL51 programming structure example .....	3-3
3.2.2 Specification statement of simulation .....	3-3
3.3 Simple Program Creation .....	3-5
3.3.1 Character display on screen .....	3-5
3.3.2 Simulation using program .....	3-8
3.3.3 Error code using compiler .....	3-11
3.3.4 Error message and error list .....	3-12
3.3.5 Program execution .....	3-16
3.3.6 Message creation by message builder .....	3-17
3.3.7 Object program save/load .....	3-26
3.3.8 Message data save/load .....	3-28
3.4 Execution of Simulation Function without Program .....	3-30
3.5 Syntax of Simulation Language (PSL51) .....	3-32
3.6 Common Function .....	3-45
3.7 Function for Transparent Mode .....	3-55
3.8 Function for Layer-2 Automatic Mode .....	3-76
4. EDITOR .....	4-1
4.1 Editor Activation and Release .....	4-2
4.2 Editor Command .....	4-5
4.3 Keyboard Macro .....	4-9*
5. CONSOLE .....	5-1
5.1 Disk I/O Related Command .....	5-2
5.2 Environment Set Command .....	5-7
5.3 Other Commands .....	5-8
5.4 Error Message .....	5-9
6. EXTERNAL CONTROL .....	6-1
6.1 Remote Control Execution .....	6-2
6.2 Remote Control Command .....	6-3
6.3 Interface .....	6-4
6.4 Command Explanation .....	6-5

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**Table of Contents**

---

<b>7. PRINTER AND HARD DISK</b> .....	<b>7-1</b>
<b>7.1 Printer Output</b> .....	<b>7-2</b>
7.1.1 Screen print .....	7-2
7.1.2 Translation data print .....	7-3
7.1.3 Print on console .....	7-4
<b>7.2 Hard Disk</b> .....	<b>7-5</b>
7.2.1 System installation .....	7-5
7.2.2 SCSI-compatible hard disk expansion .....	7-6*
<b>Appendix 1 ERROR MESSAGE LIST</b> .....	<b>A1-1</b>
<b>Appendix 2 HELP MENU</b> .....	<b>A2-1</b>

## LIST OF ILLUSTRATIONS

No.	Title	Page
1-1	Operating environment	1-5
1-2	Power cable plug and adaptor	1-7
1-3	Detachment of keyboard from D5111B	1-9
1-4	Attachment of keyboard to D5111B	1-9
1-5	Power cable storage	1-10
1-6	Front panel	1-11
1-7	Right side panel	1-12
1-8	Left side panel	1-13
1-9	Rear panel	1-15
1-10	Keyboard	1-17
1-11	Flowchart for function module selection	1-22
1-12	Flowchart for parameter setting on popup menu screen	1-23
1-13	Flowchart for change of window position and size	1-24
1-14	Specification of file name created using editor	1-25
1-15	Specification of file name for loaded data	1-26
1-16	Pin number	1-42
1-17	Feeding	1-42
2-1	System Configuration screen	2-2
2-2	Monitor loading	2-4
2-3	Flowchart for trigger function	2-5
2-4	Trigger switch setting	2-7
2-5	Selective trace switch setting	2-10
2-6	Translation screen setting	2-13
2-7	Time search	2-15
2-8	Frame number search	2-16
2-9	Pattern search	2-16
2-10	Translation screen	2-19
2-11	Layer-3 translation display	2-21
2-12	Layer-23 translation display	2-22
2-13	Layer-2 full translation display	2-23
2-14	Layer-3 full translation display	2-23
2-15	Layer-23 full translation display	2-24*

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**List of Illustrations**

---

<u>No.</u>	<u>Title</u>	<u>Page</u>
3-1	User of editor .....	3-5
3-2	Procedure for simulation execution .....	3-8
3-3	Program creation .....	3-9
3-4	Program creation .....	3-11
3-5	Error code .....	3-11
3-6	Simulation screen .....	3-16
3-7	Message creation (message builder) .....	3-17
3-8	Use of message builder .....	3-18
3-9	Frame setting .....	3-20
3-10	Object program saving .....	3-27
3-11	Object program loading .....	3-27
3-12	Message data saving .....	3-29
3-13	Message data loading .....	3-29
4-1	When a character count in one line exceeds the maximum length .....	4-4
4-2	When the display scrolls in the right direction .....	4-4
A2-1	HELP-1 in the EDITOR menu .....	A2-1
A2-2	HELP-2 in the EDITOR menu .....	A2-1
A2-3	HELP-3 in the EDITOR menu .....	A2-2
A2-4	HELP-4 in the EDITOR menu .....	A2-2
A2-5	HELP-5 in the EDITOR menu .....	A2-3
A2-6	HELP-6 in the EDITOR menu .....	A2-3
A2-7	HELP-1 in the Simulator Port A menu .....	A2-4
A2-8	HELP-2 in the Simulator Port A menu .....	A2-4
A2-9	HELP-3 in the Simulator Port A menu .....	A2-5



**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**List of Tables**

---

**LIST OF TABLES**

<u>No.</u>	<u>Title</u>	<u>Page</u>
1-1	Standard accessories supplied .....	1-4
1-2	Fuse specification .....	1-8
1-3	Pin number and functions .....	1-41
3-1	PSL51 language specification .....	3-2
3-2	Error message (1/4) .....	3-12
3-3	Simulation functions .....	3-30
3-4	Common table .....	3-45
3-5	Function for transparent mode .....	3-55
3-6	Function for layer-2 automatic mode (1/2) .....	3-76



## 1. OUTLINE

This chapter describes the configuration of this operation manual, the outline of D5111B's functions, precautions on use, and the preparations for measurement. Read this manual before starting measurement.

Configuration of chapter 1			
Outline	1.1	How to use this operation manual	
	1.2	Product outline	
	1.3	Before use	Confirmation of accessories supplied Operating environment Supply voltage Power cable Fuse replacement Keyboard attachment or detachment Power cable storage
	1.4	Panels, and location and functions of each part	Front panel Right side panel Left side panel Rear panel Keyboard
	1.5	Basic operation	Function configuration Screen configuration and functions Function module selection ( <input type="checkbox"/> F1 key function) Popup menu
	1.6	Operation example	Translation display and simulation
	1.7	Start-up file	Command explanation Start-up file creation
	1.8	Connection with equipment to be measured	Pin number and functions Connector pin number Feeding
	1.9	Performance	

## 1.1 How to Use This Operation Manual

This manual describes each item sequentially from the basic operation as explained below. You who are unfamiliar with such a measuring instrument can also use the D5111B's plentiful functions by reading this manual.

- |                          |   |  |
|--------------------------|---|--|
| 1. Outline               | : | Supposes a concrete objective with the prompt operability as an aim and describes the simple operation. Proceed to the chapters below according to the step required after operating actually in this chapter. |
| 2. Monitor               | : | Describes the details of a monitor function.   |
| 3. Simulation            | : | Describes the preparation and language required to execute a simulation function.  |
| 4. Editor/compiler       | : | Describes the environment of the program creation required for simulation.   |
| 5. Console               | : | Describes the details of a command on the file management.   |
| 6. External control      | : | Describes the command on the external control.   |
| 7. Printer and hard disk | : | Describes how to use a printer and the hard disk.  |

## 1.2 Product Outline

The D5111B can evaluate the communication equipment and line that are connected to an S/T reference point at the basic rate of ISDN.

The D5111B employs a multiwindow and popup menu, and has an excellent operability. In the communication system that puts some channels in one line as in ISDN, the D5111B improves the user-friendly by multiwindow.

### <Example >

- The multiwindow can observe two channels simultaneously.
- It enables several operations on the screen.

During simulation, an ISDN command is used for the original language (near a general-purpose language). A layer-2 protocol is automatically supported during layer-3 simulation. This facilitates programming.

### <Features >

- Conforms to the ISDN basic interface (I.430).
- Monitor functions
  - Real-time translation function
  - Replay function
  - Trigger function
  - Filter function
  - Search function
- Simulation functions
  - Editor
  - Compiler
  - PSL51 language
- Console function
- Communication function
- File system
  - MS-DOS compatible
  - 40M-byte hard disk
  - 1M-byte floppy disk
- Flat display
- Compact and lightweight

### 1.3 Before Use

#### 1.3.1 Confirmation of accessories supplied

Confirm the following after you received the D5111B ISDN protocol analyzer:

< Confirmation >

- ① Check that the external appearance of the product is not damaged.
- ② Check the quantity and specification of the standard accessories supplied while referring to Table 1-1.

If the product is damaged or the standard accessories supplied are insufficient, consult the sales and the support offices. The addresses and phone numbers are described at the end of this manual.

Table 1-1 Standard accessories supplied

Description	Specification		Quantity	Remarks
	Model name	Stock No.		
Power cable	—	DCB-DD313X01-1	1	
Power fuse	Slow-blow fuse 2A (EAWAK 2A)	DFT-AA2A	1	
System disk	D5111B system disk	PD511100-PJ	1	
Operation manual	English	ED5111B	1	

Note: Order the supplied accessories by the model name (or stock No.).

**Caution**

The D5111B ISDN protocol analyzer has an internal hard disk. For the transportation of the D5111B, see step (5) on the next page.

### 1.3.2 Operating environment

- (1) Do not leave the set in a location subject to excessive dust or vibration, direct sunlight, or corrosive gas. Use the set at an ambient temperature of +5°C to +40°C and a relative humidity of less than 80%.
- (2) The D5111B has an internal hard disk. A shock with the power set to on damages the disk. Be careful not to shock the D5111B.
- (3) The D5111B uses a cooling suction fan for internal cooling. Be careful not to block the intake or the ventilating slit at the top of the D5111B. This prevents ventilation.
- (4) The storage temperature of the D5111B is -10°C to +60°C. If the D5111B is not to be used for an extended period of time, cover it with vinyl or put it in a corrugated cardboard box and keep in a dry place not subject to direct sunlight.
- (5) Use a supplied packing material when you transport the D5111B. If you lost the packing material, pack the D5111B as follows:
  - ① Wrap the D5111B in vinyl.
  - ② Cover the D5111B with a shock absorbing material and put it in a corrugated cardboard box of more than 5mm in thickness.
  - ③ Put accessories supplied in the corrugated cardboard box, then a shock absorbing material closely. Close the box and fix it with packing tape.
  - ④ Close the cover of the corrugated cardboard box and fix it with packing tape.

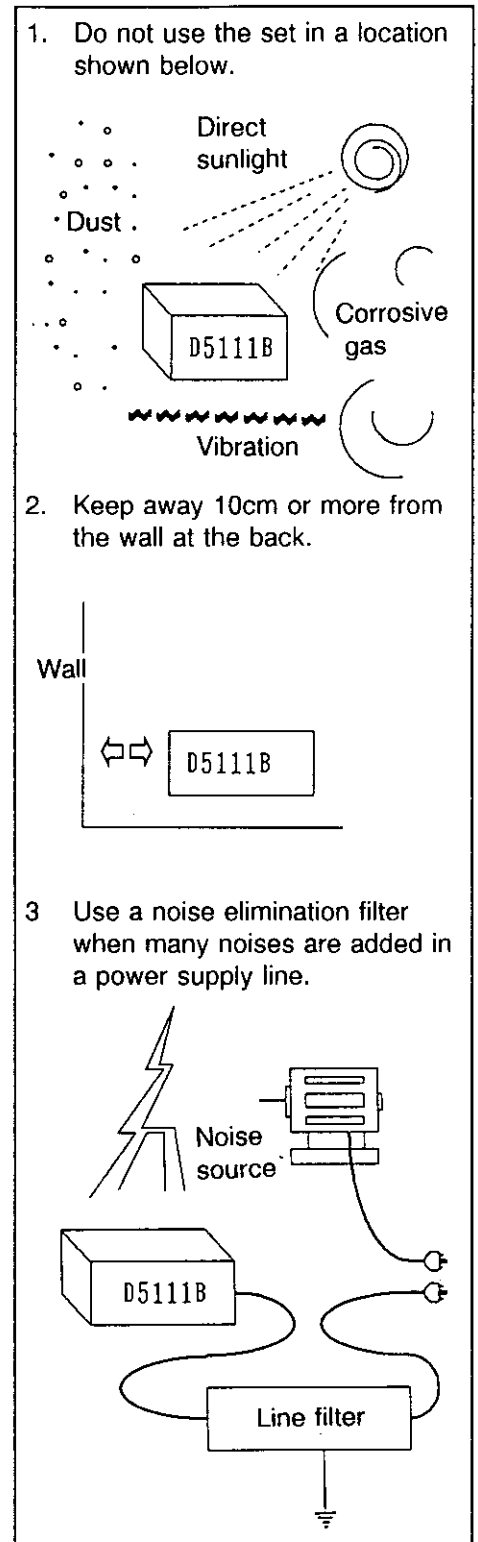


Figure 1-1 Operating environment

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**1.3.3 Supply voltage**

Turn the power switch off before connecting the power cable. The supply voltage is 90V to 132V AC/180V to 250V AC, and the power frequency is 48 to 440Hz.



### 1.3.4 Power cable

The plug of the power cable has three pins. The round pin is used for ground.

Use a grounded wall outlet. Use an adaptor (A09034) supplied for the plug when using only the two pins. In this case, connect the ground wire leading from the adaptor securely to ground.

The two electrodes of the adaptor (A09034) differ in width. Confirm the direction of the plug and wall outlet before inserting the adaptor into the wall outlet. Use an optional adaptor (KPR-13) when the adaptor (A09034) cannot be connected to the wall outlet used.

#### Caution

Be careful not to touch the AC power supply when the ground wire leading from the adaptor is connected to ground.

This may damage the D5111B or other connected equipment.

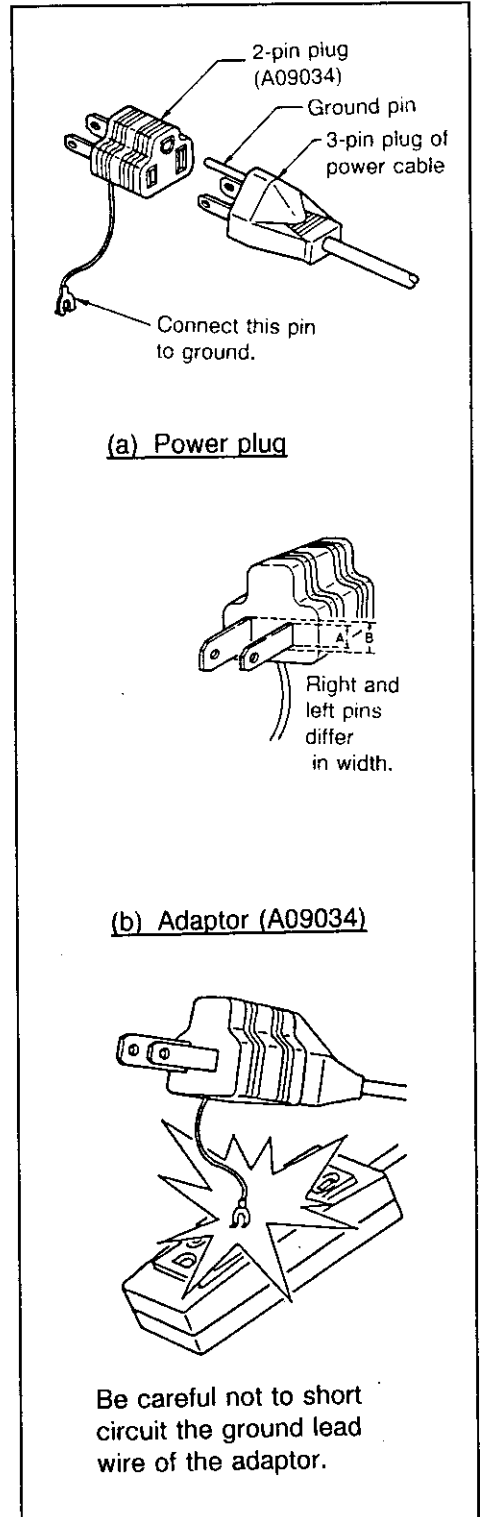


Figure 1-2 Power cable plug and adaptor

### 1.3.5 Fuse replacement

<Operation procedure>

- ① Loosen the fuse holder cap with a flat-bladed screwdriver, then pull out the fuse and cap.
- ② Replace the fuse on the cap by a new one.
- ③ Insert the cap in which a new fuse is installed into the holder.

Table 1-2 Fuse specification

Fuse name	Model name	Part code
Power fuse	Slow-blow fuse 2A	DFT-AA2A

Caution

1. Turn the power off and pull out the power cable from the wall outlet before replacing a fuse.
2. Do not check a fuse only visually. Measure the resistance value and check that it is less than 15 ohms.
3. To avoid a fire, use a fuse of same type and rating during replacement.

### 1.3.6 Keyboard attachment or detachment

The keyboard of the D5111B is attached on the front panel. It functions to protect the front panel part during carrying.

To detach the keyboard from the D5111B, slide the knobs on both sides of the top inwards and pull out the keyboard toward you. At that time, take care not to drop the keyboard.

To attach the keyboard, put the cable near the D5111B's connector in a hook as shown in the figure and store the extra cable in the rack on the front of the panel. Then, align the projections on the keyboard with the grooves on the D5111B and push the keyboard to the D5111B side. Insert so that the projections on both side are clicked.

#### Caution

Do not attach the keyboard with the power turned on. Cooling of the display section on the D5111B is obstructed. This may cause a trouble. Be sure to turn the power switch off before attaching the keyboard.

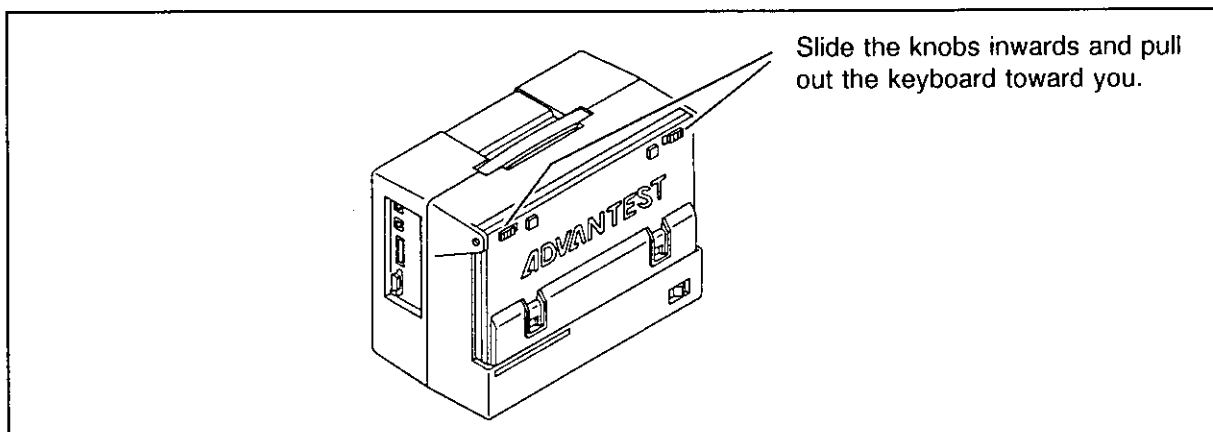


Figure 1-3 Detachment of keyboard from D5111B

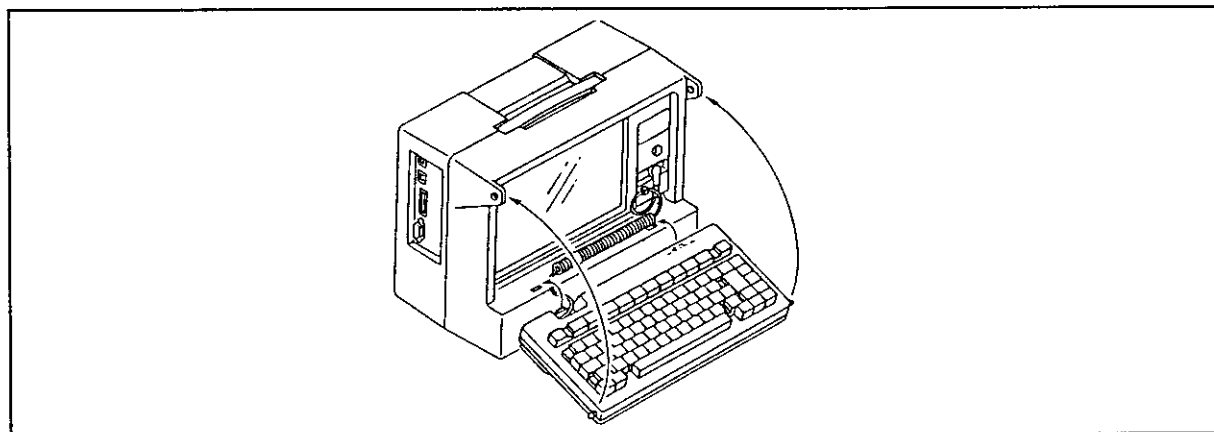


Figure 1-4 Attachment of keyboard to D5111B

### 1.3.7 Power cable storage

The power cable is packed together with other accessories supplied at the factory. It can be stored in the bottom of the D5111B during transport. Slide the bottom of the D5111B toward you. The lid on the bottom is then opened. Fold the power cable properly and put it in the bottom.

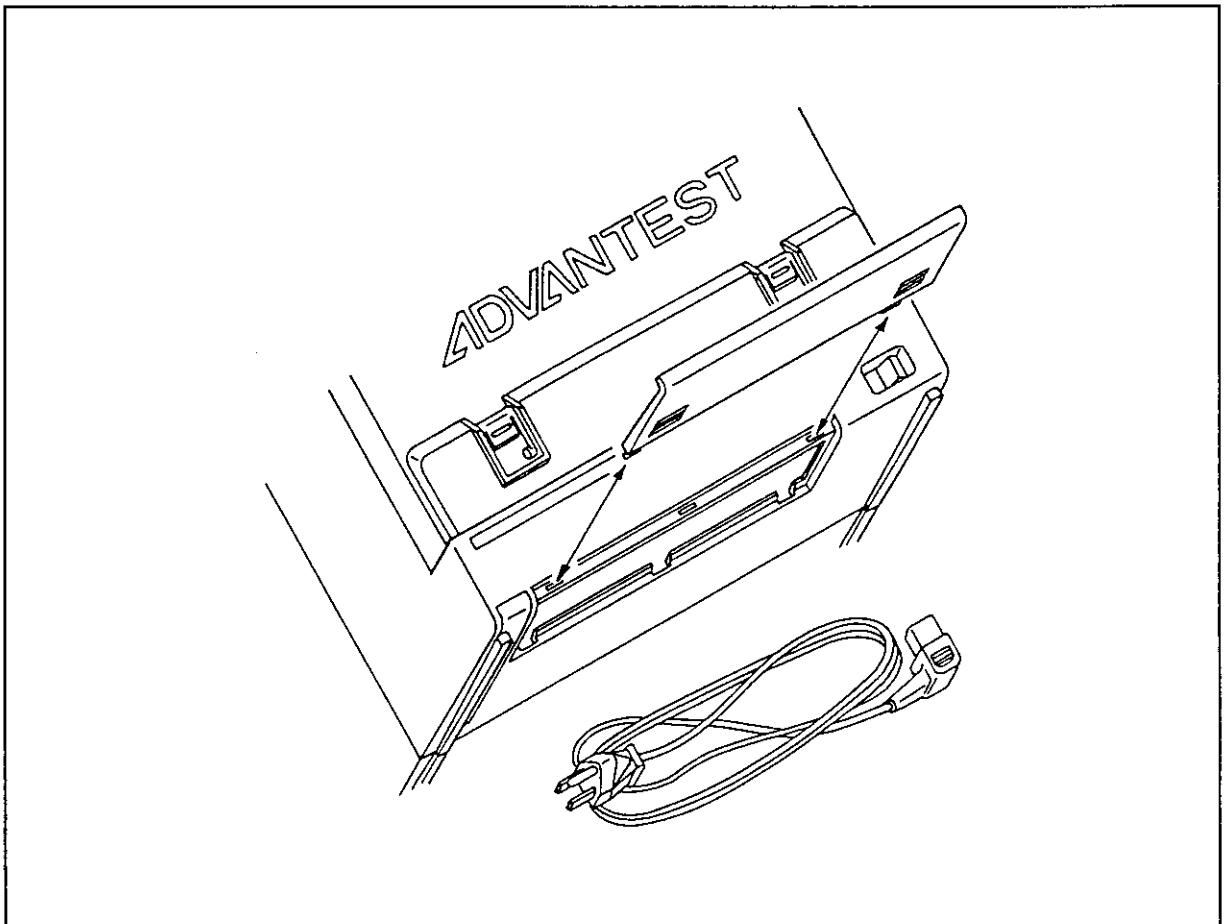


Figure 1-5 Power cable storage

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**1.4 Panels, and Location and Functions of Each Part**

**1.4 Panels, and Location and Functions of Each Part**

**1.4.1 Front panel**

- |   |                                  |   |   |
|---|----------------------------------|---|---|
| ① | Fan filter                       | : | Cooling fan filter. Eliminate dust from the filter per month.   |
| ② | Power cable storage section      | : | Stores a power cable when it is not used.   |
| ③ | Power switch                     | : | Used as a power switch of the D5111B. Press the power switch to the left to turn the power off. Press it to the right to turn the power on. |
| ④ | Keyboard connector               | : | Used for keyboard connection.   |
| ⑤ | FDD lamp                         | : | Lights when the floppy disk drive is activated. (Red LED)   |
| ⑥ | HDD lamp                         | : | Lights when the hard disk drive is activated. (Red LED)   |
| ⑦ | Electroluminescence (EL) display | : | The EL display is an 9-inch flat display.   |

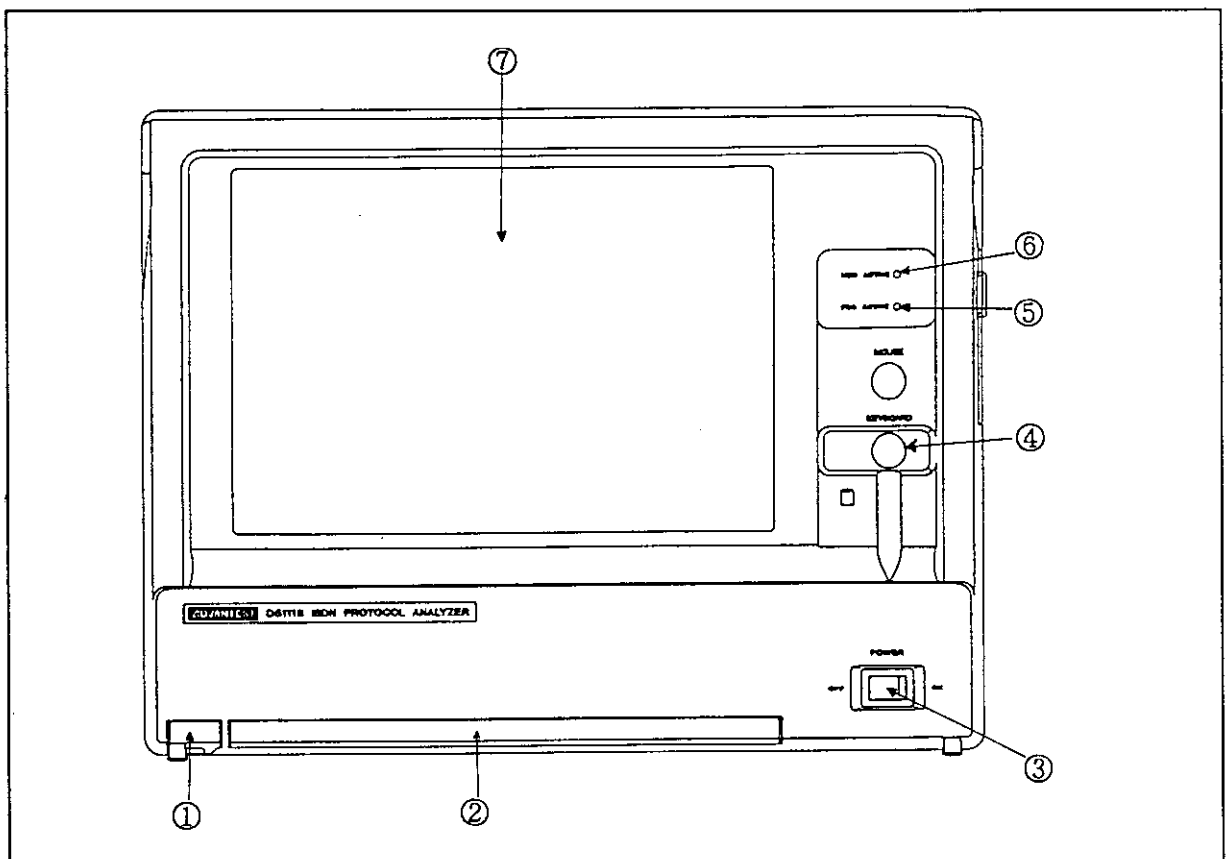


Figure 1-6 Front panel

### 1.4.2 Right side panel

① FD drive

: The FD drive is a 3.5-inch floppy disk drive.

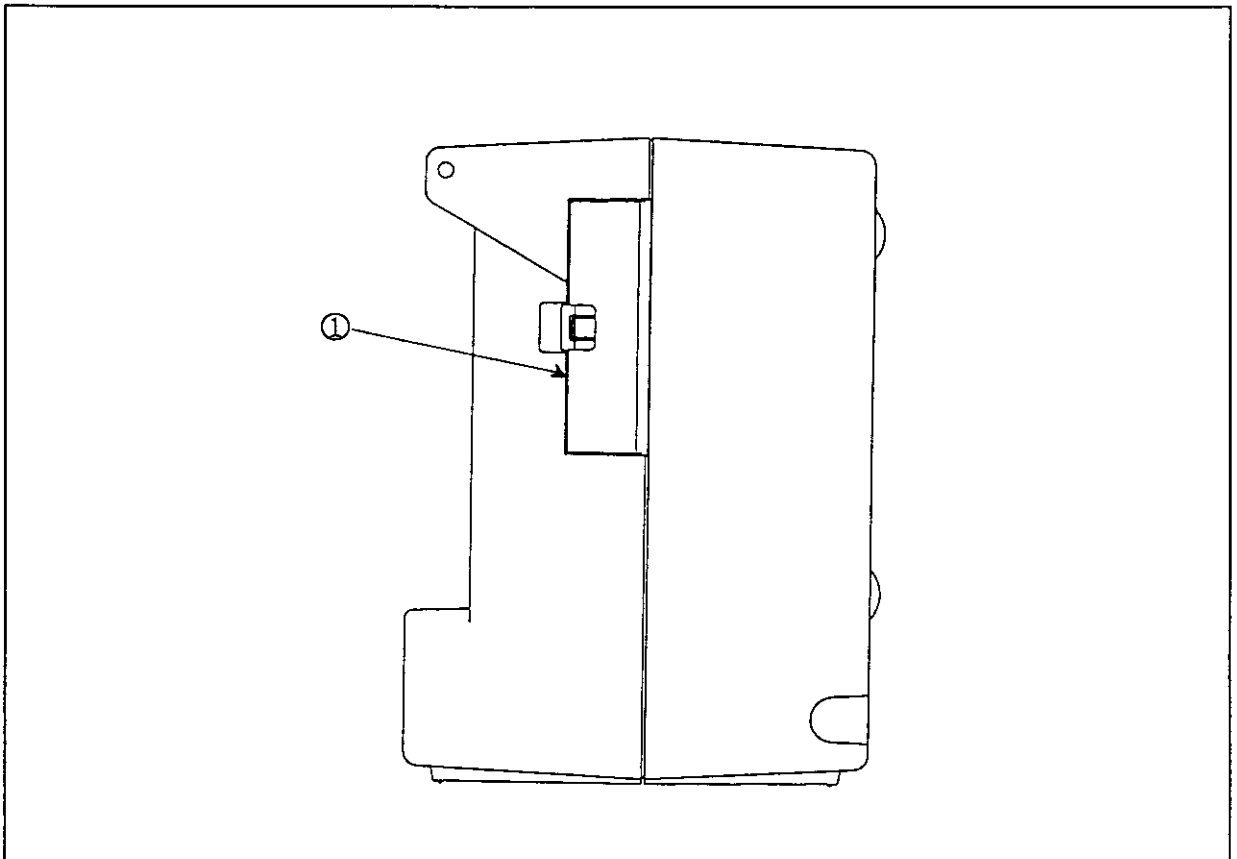
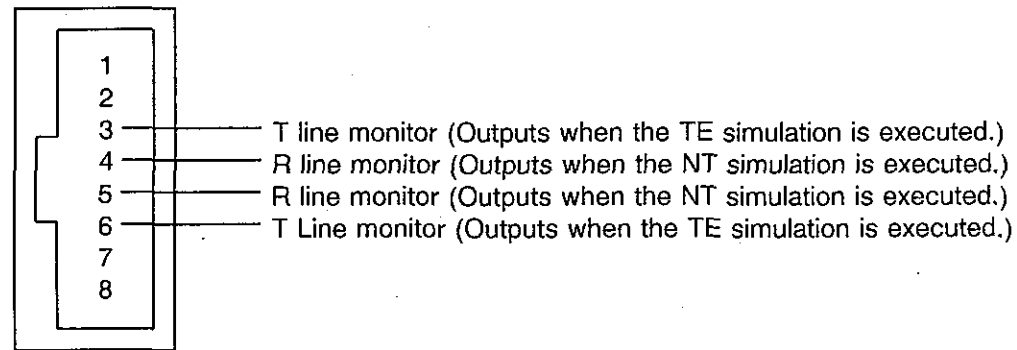


Figure 1-7 Right side panel

1.4.3 Left side panel

①, ② S-BUS connectors : Used to connect to an S/T reference point at the basic rate of an ISDN line.  
Each 3 to 6 pins of S-BUS connectors.



③ MONITOR OUT connector : Outputs the frame clock and the clock data.

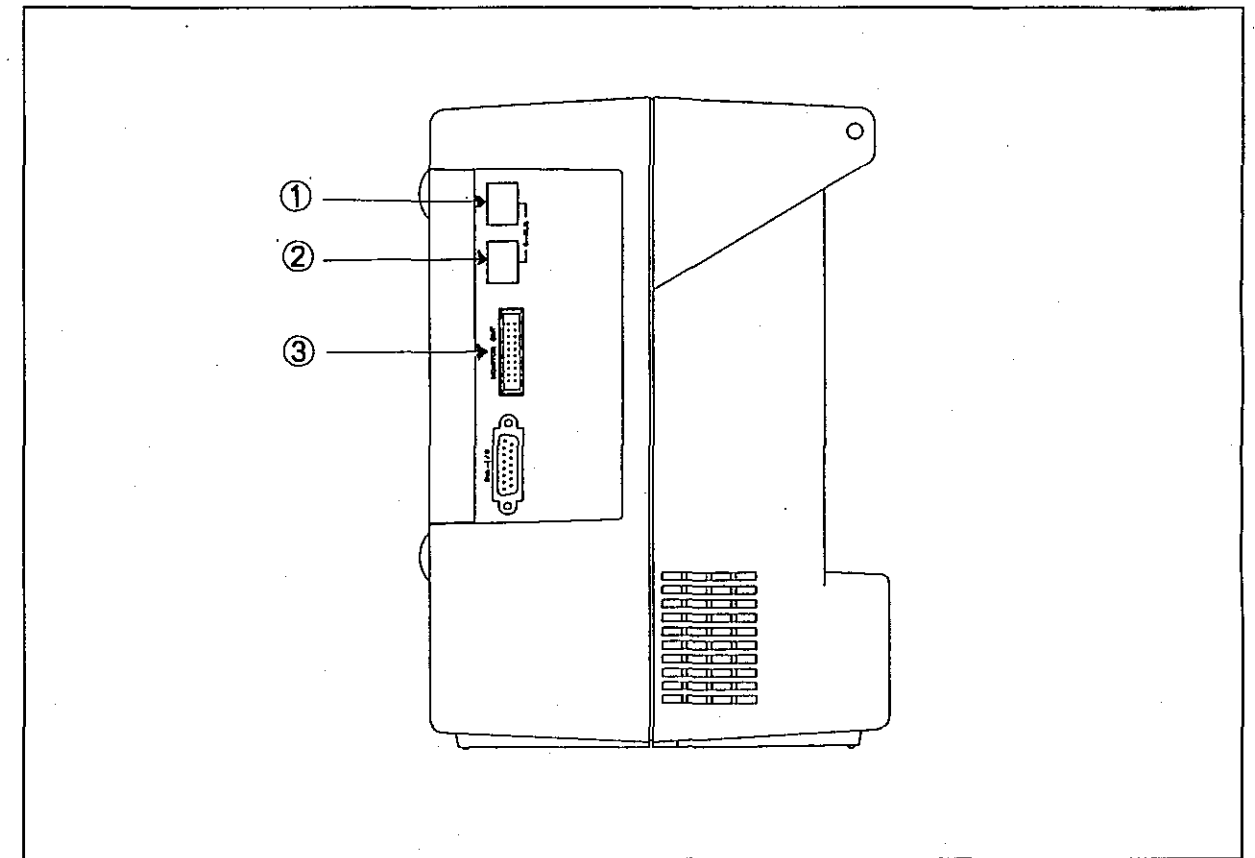
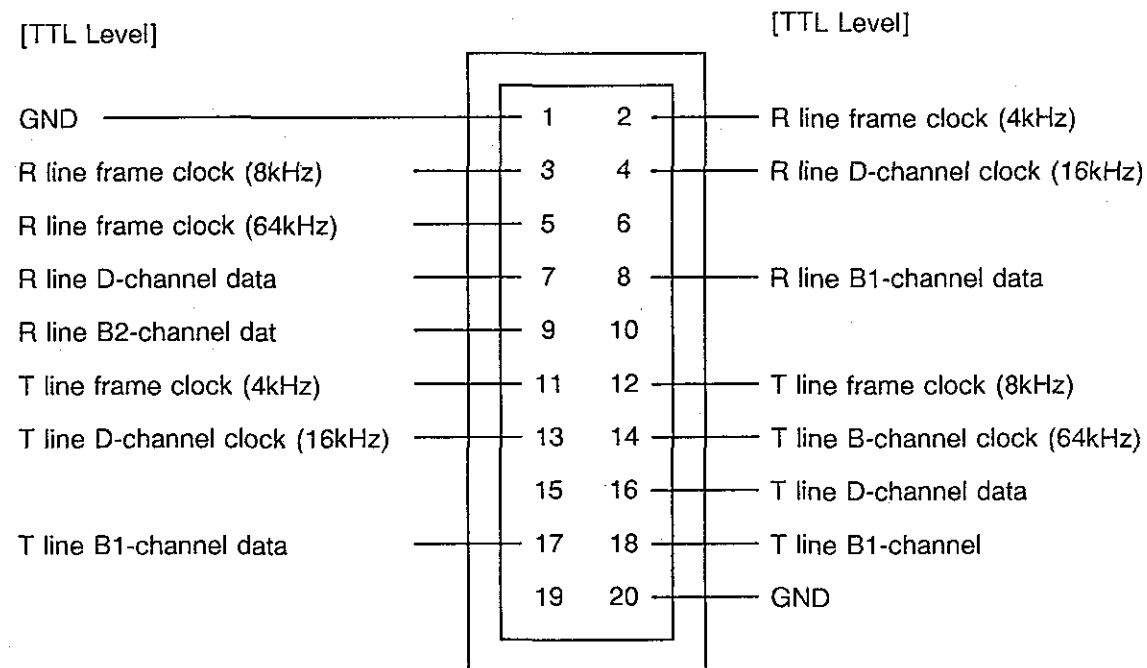


Figure 1-8 Left side panel

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**1.4 Panels, and Location and Functions of Each Part**

**1.4.4 Rear panel**

- ① ~LINE : Power input connector of D5111B. The supply voltage that can be input is 90 to 132V AV/180 to 250V AC (48 to 440Hz). Use a T2A-type fuse.
- ② SCSI : Used as a connection connector of the external hard disk.
- ③ Centronics I/F : Printer interface. Use a printer that conforms to the Centronics standard.
- ④ EIA-232D : Conforms to the EIA-232D interface standard. The EIA-232D is used to transfer measurement data to the external unit or to activate or stop the D5111B remotely.

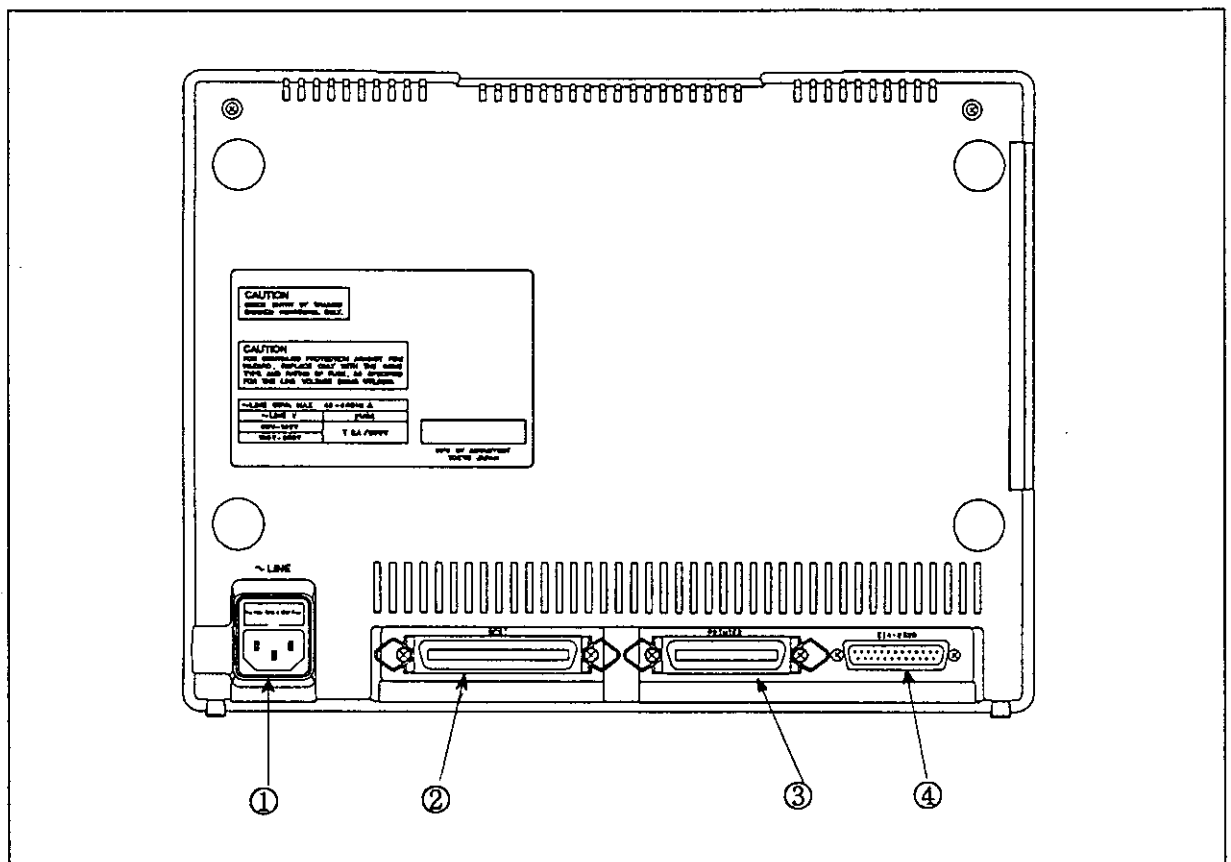


Figure 1-9 Rear panel



### 1.4.5 Keyboard

① Connection cable : Used for connection with the D5111B.

② Major keys

F1	:	Selects the function module.														
F2	:	Controls the window.														
F3	:	Activates the control function for the editor screen.														
F9	:	Starts the screen print by pressing the measurement start (START)/ <span style="border: 1px solid black; padding: 2px;">CTRL</span> + <span style="border: 1px solid black; padding: 2px;">F9</span> keys.														
F10	:	Stops the screen print by pressing the measurement stop (STOP)/ <span style="border: 1px solid black; padding: 2px;">CTRL</span> + <span style="border: 1px solid black; padding: 2px;">F10</span> keys.														
SPACE	:	Activates or sets the popup menu.														
CR	:	Activates or sets the popup menu.														
ESC	:	Moves the cursor to the home position. Cancels the popup menu.														
ALT	:	Activates the function module that is loaded using the <span style="border: 1px solid black; padding: 2px;">ALT</span> + Fn keys*.														
	(* Fn keys:	<table border="1" style="display: inline-table; border-collapse: collapse; text-align: center;"> <tr> <td style="padding: 2px;">F1</td> <td style="padding: 2px;">F2</td> <td style="padding: 2px;">F3</td> <td style="padding: 2px;">F4</td> <td style="padding: 2px;">F5</td> <td style="padding: 2px;">F6</td> <td style="padding: 2px;">F7</td> </tr> <tr> <td style="padding: 2px;">F8</td> <td style="padding: 2px;">F9</td> <td style="padding: 2px;">F10</td> <td colspan="4"></td> </tr> </table> )	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10				
F1	F2	F3	F4	F5	F6	F7										
F8	F9	F10														
HELP	:	Displays the help screen. (In EDITOR and SIMULATION PORT A screen) Cancels the help screen. (In EDITOR and SIMULATION PORT A screen)														
←	:	Deletes a previous character of the cursor. Inputs the tab code. (In EDITOR screen)														
DEL	:	Deletes a character at the cursor point. Inputs the tab code. (In EDITOR screen)														
↗	:	Moves the next input field. (It all screens except for EDITOR)														
☐ LOCK	:	Selects the cap or the low. (In EDITOR screen)														
↑ ↓	:	Scrolls the monitor display and the history display.														
▲ ◀ ▼ ▶	:	Moves the cursor.														

D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL

1.4 Panels, and Location and Functions of Each Part

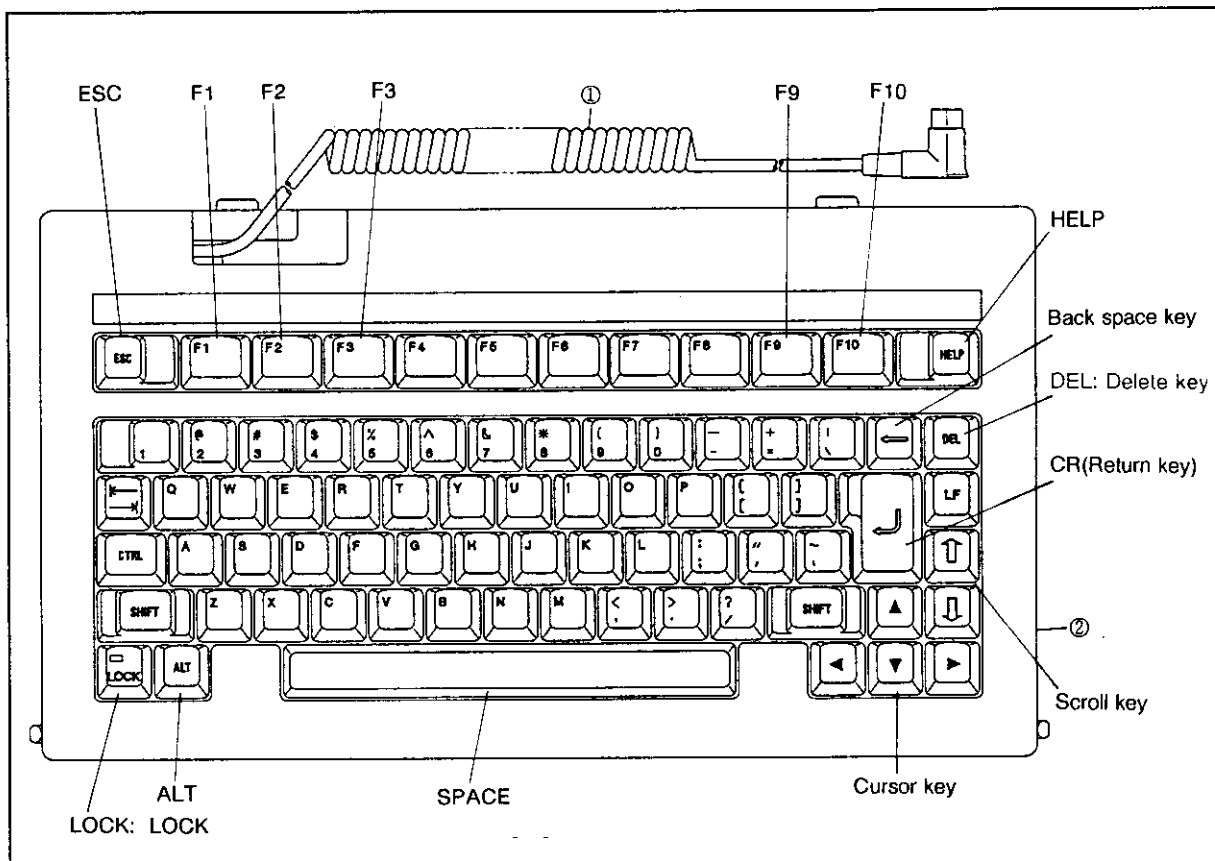


Figure 1-10 Keyboard

## 1.5 Basic Operation

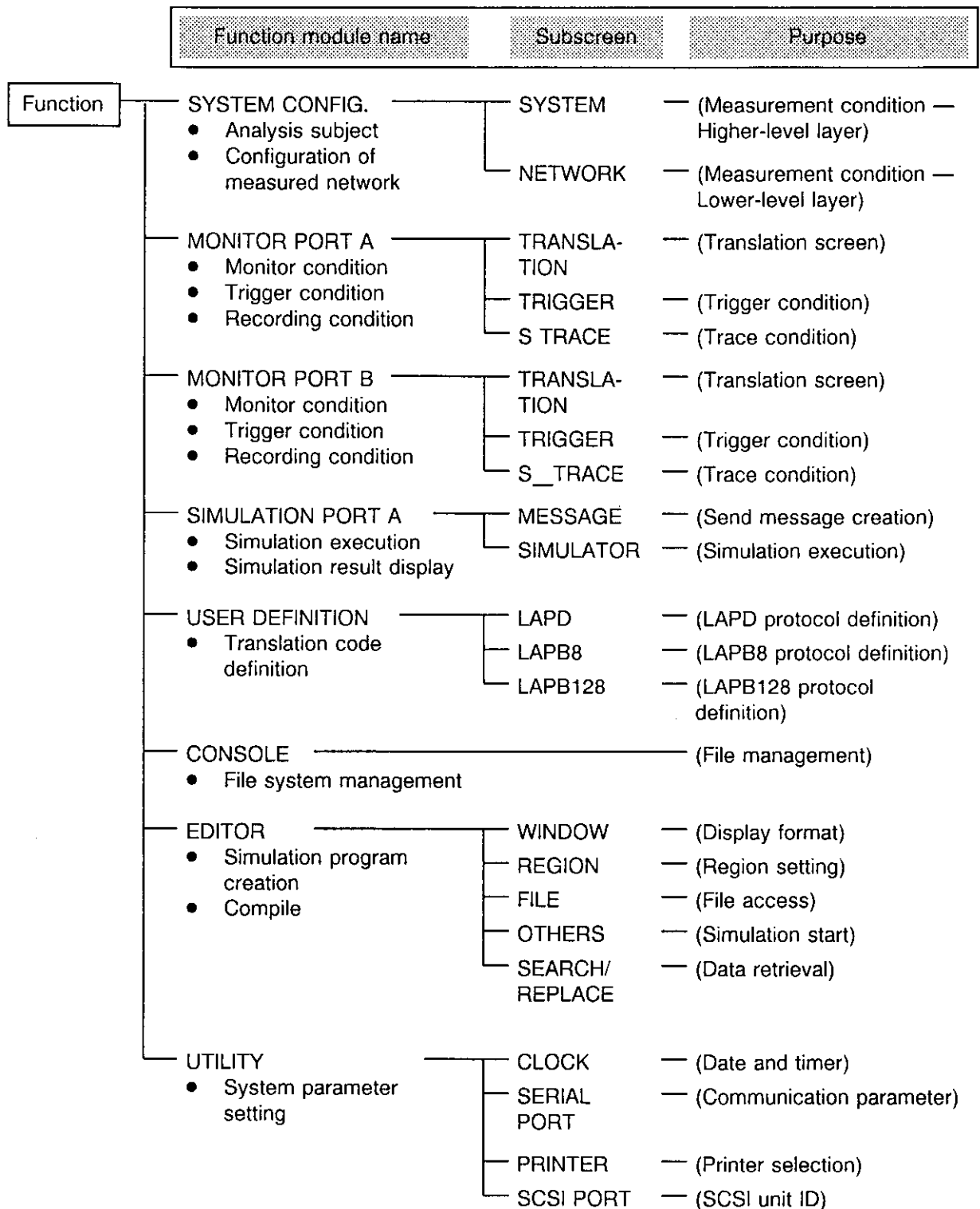
This manual describes each function in detail, but this D5111B has a man-machine interface much more excellent than the conventional man-machine interface. Therefore, you can operate without reading the operation manual thoroughly. To master the operation more quickly, concrete examples are given for explanation. You do not require understanding each operation exactly. Comprehend the thought of the whole operation.

The D5111B is constituted by eight functions as described on the next page. Each function is called a function module and does not exist always in the system. A user loads the system to execute the functions as required. This D5111B installs a window system, so you can use each function without having a sense of incompatibility. During system start, you can also load a necessary function in the system according to your applications using a start-up file. Setting, analysis (translation), and programming are performed using each function module. The popup menu is often used in the window system of a workstation. The popup menu of this D5111B can also be handled in the same way as in the workstation.

This section describes the function configuration, popup menu, and window system.

### 1.5.1 Function configuration

The function configuration and module functions are shown below.





1.5.2 Screen configuration and functions

The screenshot displays the D5111B ISDN Protocol Analyzer interface. At the top, it shows 'ADVANCED D5111' and 'PROTOCOL ANALYZER 08/30/90 21:08'. The main window is divided into three sections:

- System Configuration:** Includes settings for SOURCE, LINE, RECORD, OFF, MONITOR (Port A/B), Channel (D/B1), Protocol (LAPD/LAPB8), Memory Mode (SINGLE), and Frame Len. (0264).
- LAPD User Definition:** A table for defining LAPD patterns. The 'Define' field is set to 'LAYER2'. The table has columns for No, TYP, C/R, FRAME, and PATTERN.
- Monitor Port A LAPD:** A data capture table with columns for NO, D, S, C, R, I, FRAME, NR, AS, DE, LIE, and a final column with values 08, 02, 08, 02, 08, 02, 08, 02, 08, 02, 08, 02, 08, 02, 08, 03.

At the bottom, there is a control bar with buttons for CONFIG, MON port A/B, SIM port A, USERDEF, and a search bar with options TIME, TRIG, FRM, PAT, ERR. A 'LAYER1' section has buttons for TE, NT, and PWR OFF.

- ① Function module selection area
- ② Active window display selection area
- ③ Calendar timer display area
- ④ Function module title display area
- ⑤ Subscreen selection area
- ⑥ File save/load selection area
- ⑦ Function module icon display area
- ⑧ Layer-1 status display area
- ⑨ Feed status display area
- ⑩ Translation display format selection area
- ⑪ Data search mode selection area
- ⑫ Print-out specification area

### 1.5.3 Function module selection

( F1 key function )

The function module can be accessed on the screen at all times. The same function module can be loaded at the same time. This enables the translating protocol in layers 2 and 3 to be displayed simultaneously using same data. The data monitor can also be displayed at the same time while the simulation function is executed.

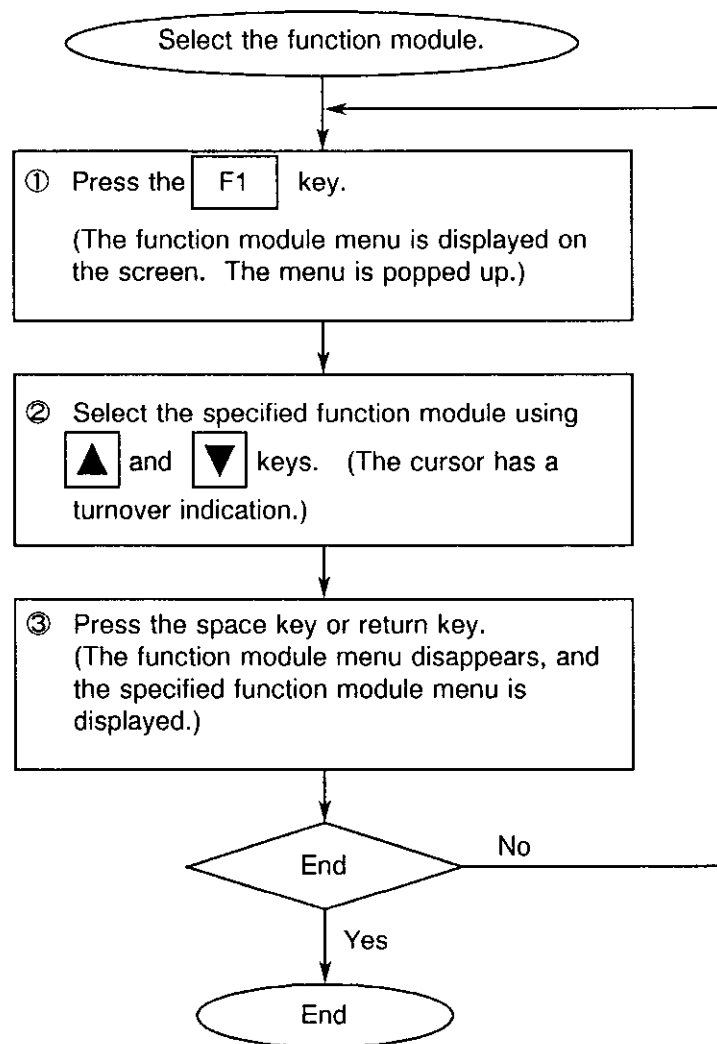


Figure 1-11 Flowchart for function module selection

When several function modules are loaded (several function module icons are displayed in the lower part of the screen), press the ALT key and the function keys (F1 through F10) corresponding to each icon position at the same time. Any screen can then be specified in an active window.

The active window indicates the screen on which a key can be entered. The outer frame of the screen has a turnover indication.

#### 1.5.4 Popup menu

Parameters can be set on the popup menu screen in accordance with the procedure below. The meaning of each parameter is explained in the corresponding function.

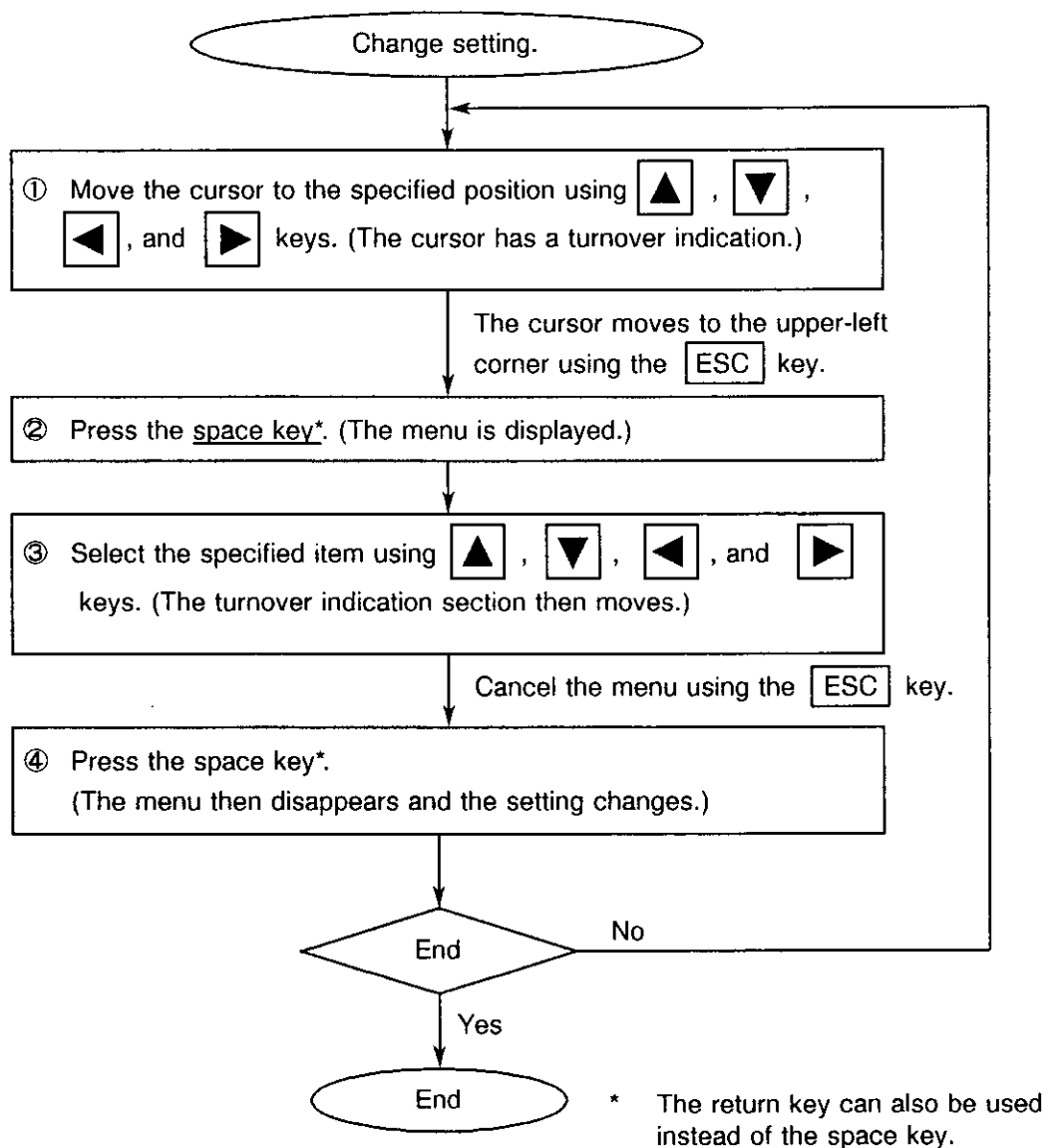


Figure 1-12 Flowchart for parameter setting on popup menu screen



<Display position and size (  key function )>

The D5111B installs a multiwindow system. This system enables two or more functions to be executed at a time and displayed at the same time. The specification of the active window position and size is described below.

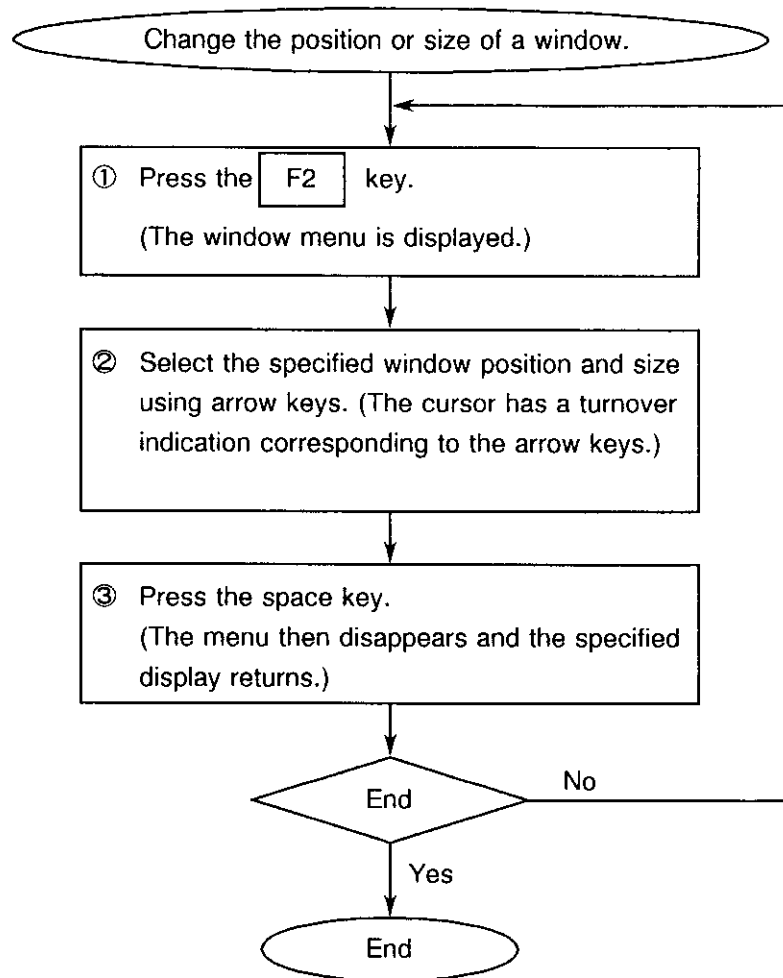


Figure 1-13 Flowchart for change of window position and size

Note: The size that can be displayed is determined corresponding to the menu screen. Therefore, the window menu cannot display all screen sizes and positions.

HIDE: Moves the current active window to the lowermost position of the overlapped screen. A part or all of the display cannot often be viewed when two or more screens are displayed.

QUIT: Erases the current active window in the system memory of the D5111B. The icon display in the lowermost position also disappears.

### 1.5.5 File name specification

To specify a name for the file name created using an editor and save it in a disk or specify a name for the data loaded from a line and save it in a disk, specify the file name using the popup menus shown below.

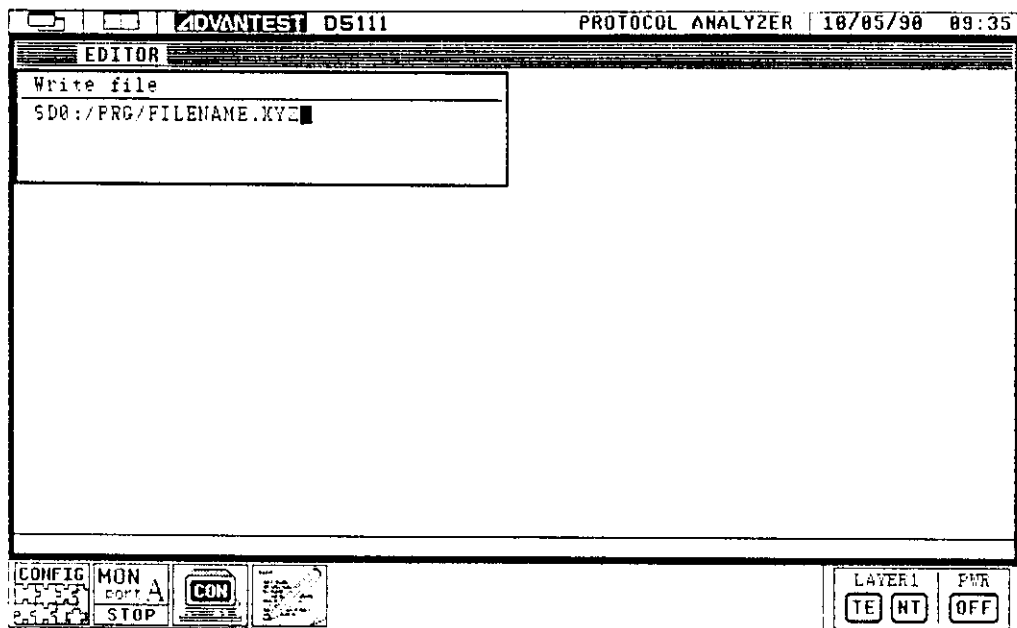


Figure 1-14 Specification of file name created using editor

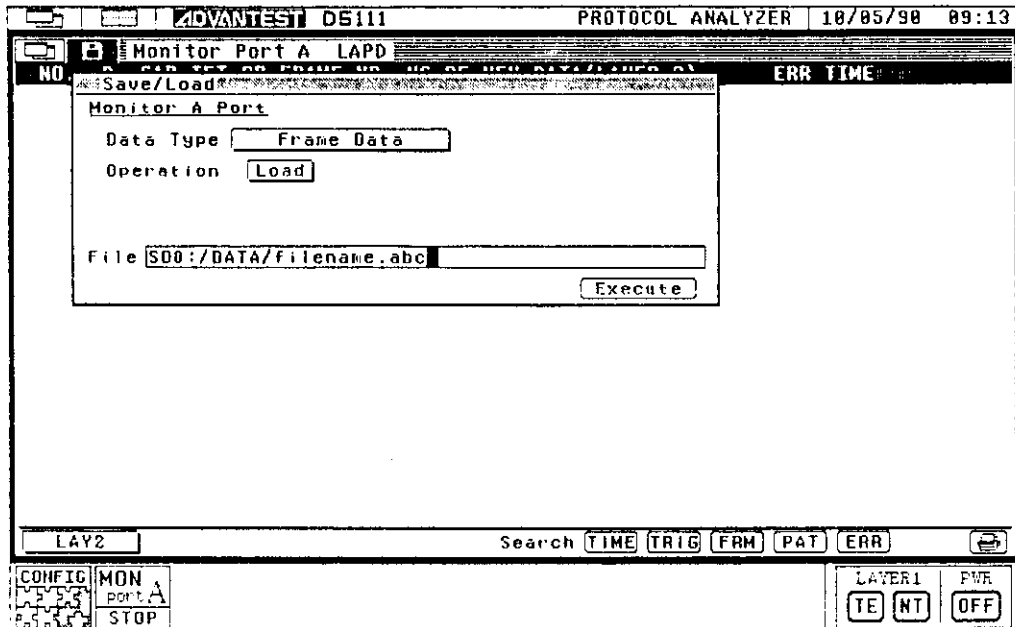
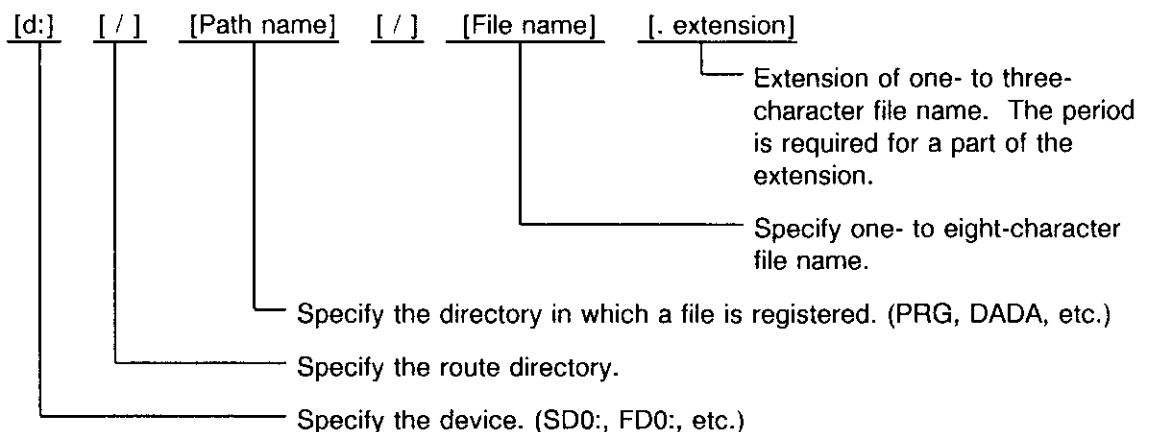


Figure 1-15 Specification of file name for loaded data

The specified file name at that time is limited as in an MS-DOS system. Specify a file name in accordance with the format below.



The files in upper- and lower-case characters are not distinguished.

SD0:/PRG/FILENAME. XYZ    and    sd0:/prg/filename. xyz

are handled as a same file name.

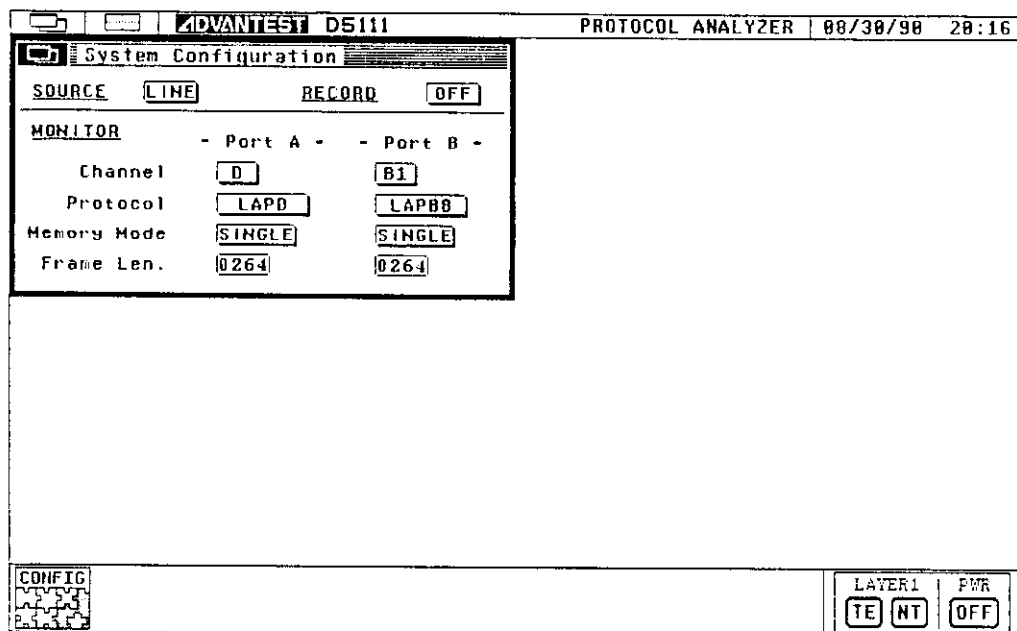
## 1.6 Operation Example

### 1.6.1 Translation display

This section describes the operation procedure for a translation display using sample data. Actual line data can be monitored through this explanation.

Begin with step ③ or ④ when the system is already started.

- ① Insert the plug.  
Check that the D5111B power is turned on, then insert the plug into the wall outlet.
- ② Power on  
The D5111B starts a self diagnosis when the power is turned on. If the self-diagnosis result is normal, the system is loaded.
- ③ Initial screen  
The "System Configuration" screen is displayed after the system loading is completed.

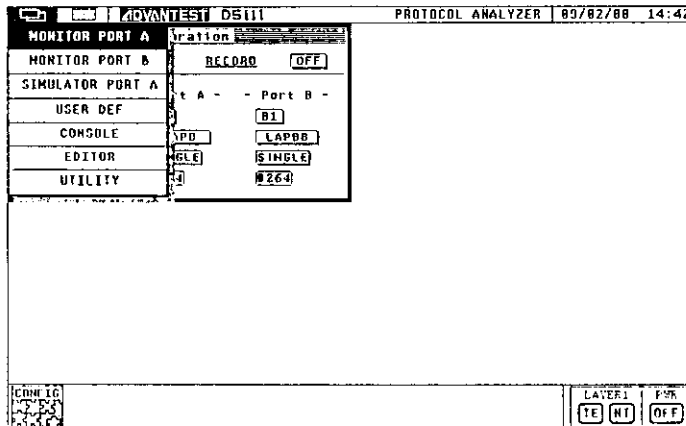


**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

1.6 Operation Example

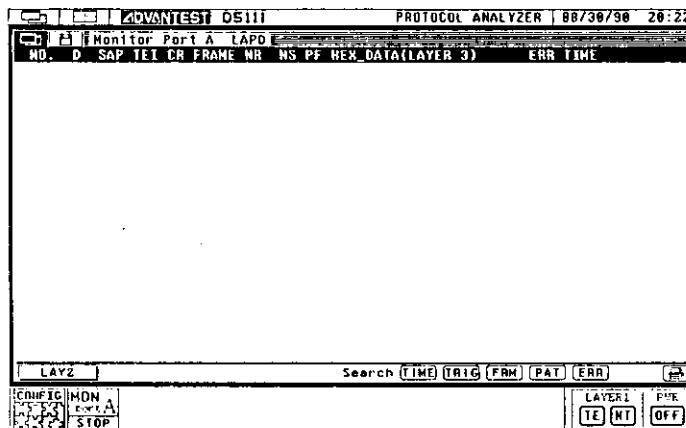
④ Function selection screen

Press the F1 key; the function module selection menu is displayed.



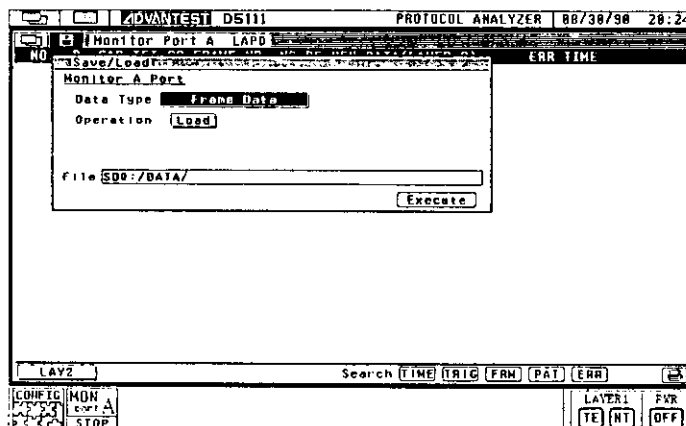
⑤ Translation screen

Select the "MONITOR PORT A" using ▲ and ▼ keys. Press the space key or return key; the translation screen is displayed. No data exists yet. (An actual line is not monitored on this screen. The translation display is performed using saved data.)



⑥ Preparation for data read

Move the cursor to the file SAVE/LOAD selection area using the ▶ key and press the space key (to read the data in a disk).




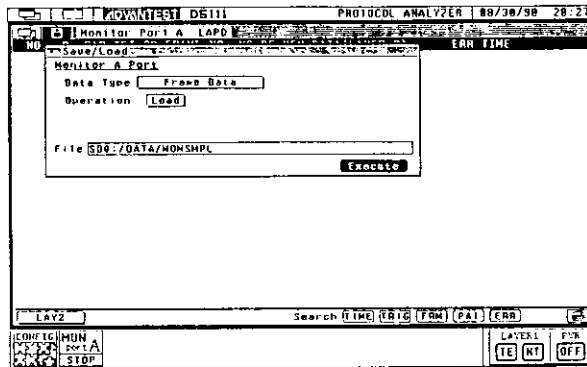
**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**





**1.6 Operation Example**

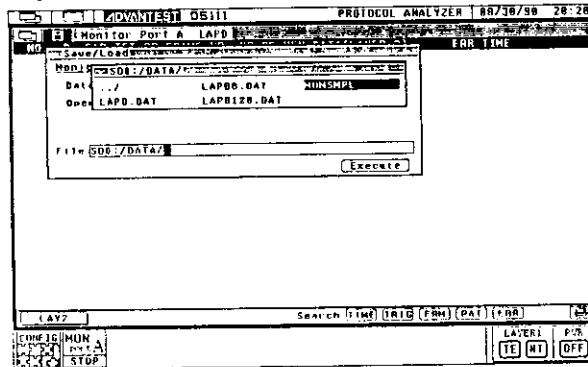
⑦ Data read

There are two methods below to read data.



- (a) On the "Save/Load" screen, enter "DATA/MONSMPL" in the "File" using a  key and move the cursor to the next "Execute". Then, press the space key. (The "Save/Load" screen disappears when loading is completed. The translation data is then displayed.)

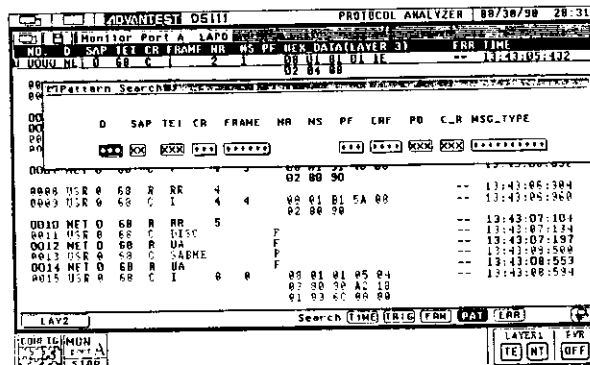


- (b) Press the space key with the cursor in the "File" position; the file name in an SD0:/DATA/ directory is displayed. Select "MONSMPL" using , , , and  keys and press the space key. Then, move the cursor to the "Execute" and press the space key.



⑧ Data retrieval



Move the cursor to the "PAT" in the lower-right position of the screen using  and  keys and press the space key. Next, retrieve specified pattern data.

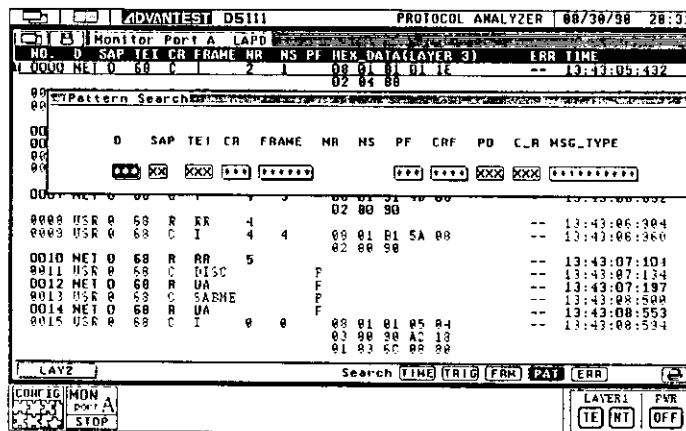


**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

1.6 Operation Example

⑨ Retrieval pattern registration

Register a specified pattern in the pattern registration menu using a popup menu and key. Press the  key after the registration is completed; the pattern data is retrieved upward from the current cursor (underlined) position. Press the  key; the pattern data is retrieved downward. If no registered pattern frame exists, a message is displayed.



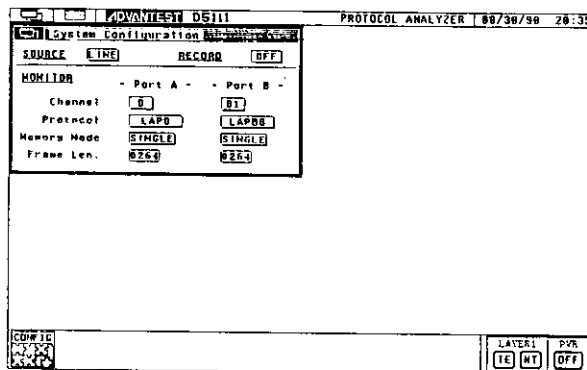
Note: For more information on the translation, see 2.4, "Translation Screen."

### 1.6.2 Simulation

The simulation requires a target for concrete operation. This section describes the simulation procedure. For more information, see chapter 3, "Simulation" and chapter 4, "Editor."

To execute the simulation, create the program using an editor and create the execute-form file using a compiler. This example uses the "SIMSMPL" program that already exists in a disk.

- ① Insert the plug.  
Check that the D5111B power is turned off, then insert the plug into the wall outlet.
- ② Power on  
The D5111B starts a self diagnosis when the power is turned on. If the self-diagnosis result is normal, the system is loaded.
- ③ Initial screen  
The "System Configuration" screen is displayed when the system loading is completed.



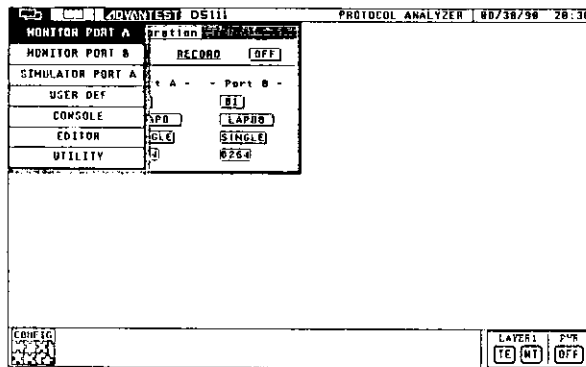


**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

1.6 Operation Example

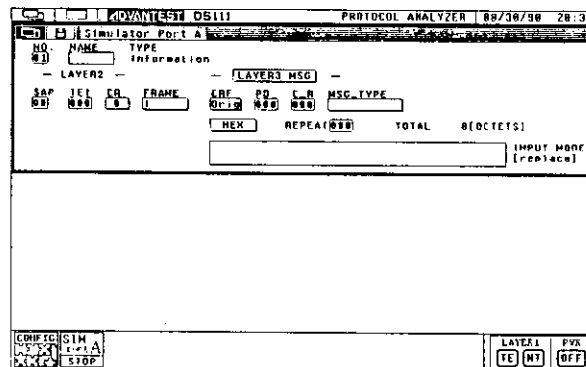
④ Function selection screen

Press the F1 key; the function module selection menu is displayed.



⑤ Simulation screen

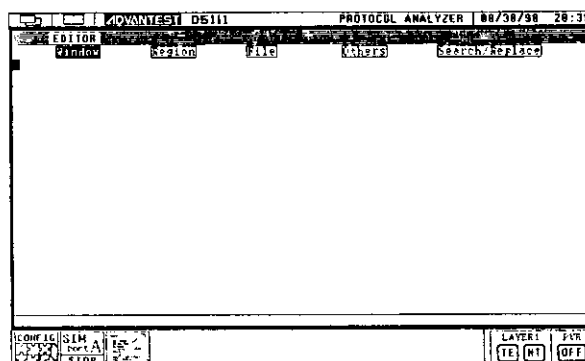
Select "SIMULATOR PORT A" using ▲ and ▼ keys and press the space key or return key. The simulation screen is then displayed.



⑥ Program preparation

Select the editor of a function module using the F1 key.



Then, press the F3 key; the editor's function selection menu is displayed.

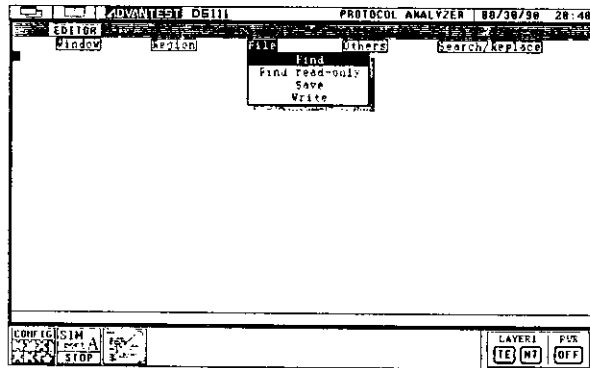


**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**1.6 Operation Example**

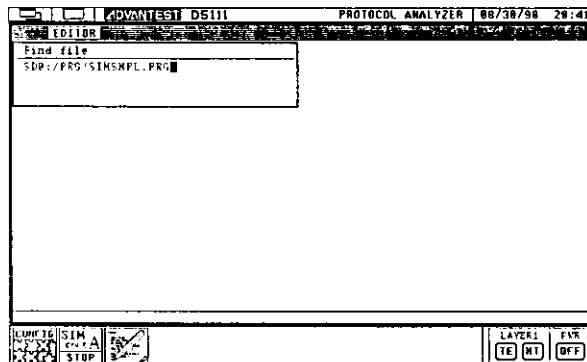
⑦ Program read

Select "File" using  and  keys and press the space key. Then, select "Find."




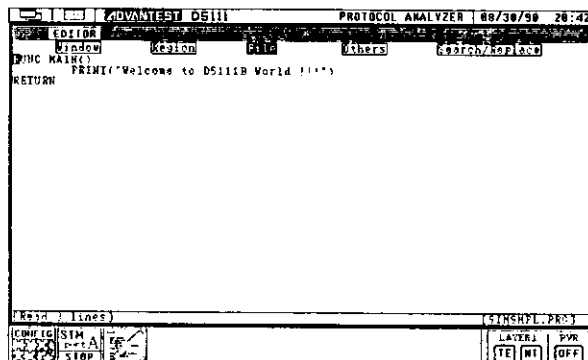
⑧ Read the program.

Enter the "Find file" name as "/PRG/SIMSMPL.PRG."



⑨ Program edit





The program that is read is displayed on the screen. Press the  key again; the function label disappears. Each key can then be entered.

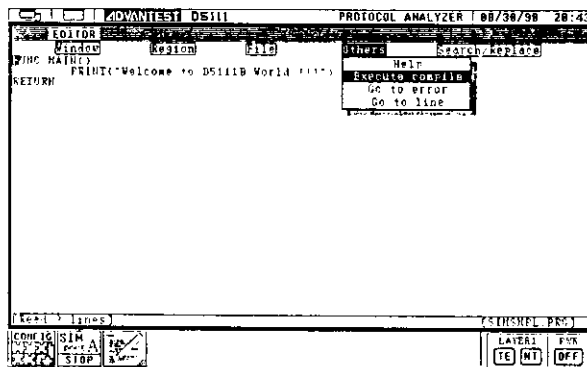


**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

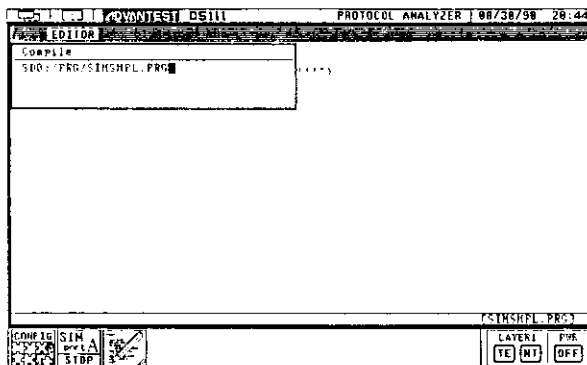
1.6 Operation Example

⑩ Compile

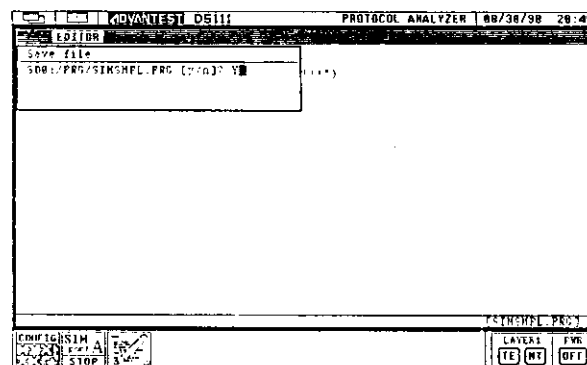
Press the **F3** key when the edit is completed. Then, select "Others" using  and  keys, select "Execute compile" using  and  key, and press the space key.



⑪ The file name to be compiled is confirmed. Press the return key at that time. To compile other files, enter the corresponding file name.



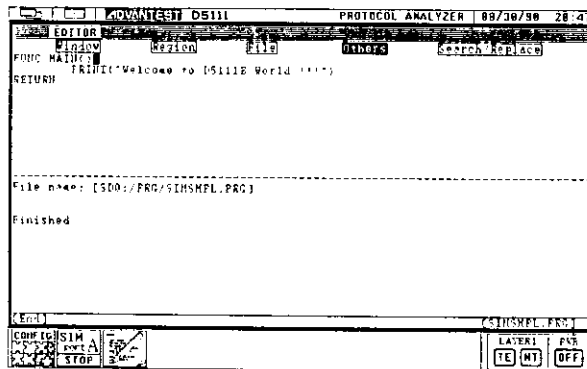
⑫ Save the program file.  
Whether the modified program should be saved is confirmed. Enter "Y (Yes)," then press the return key. The compile is started.



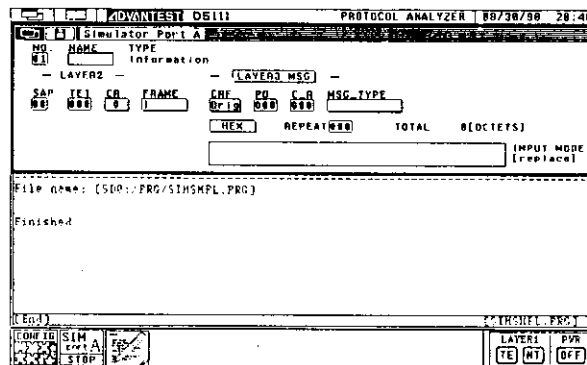
**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

1.6 Operation Example

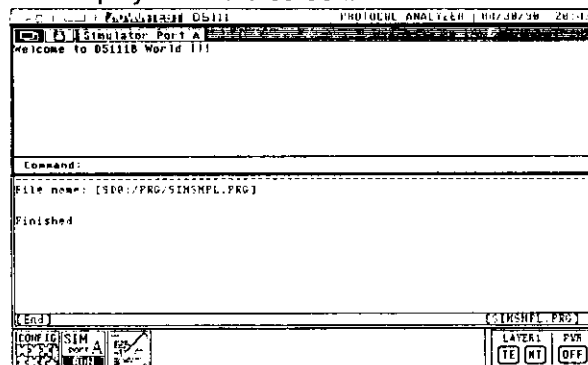
- ⑬ Compile completion  
 "Finished" is displayed when the compile is completed. This indicates that no error exists. If an error exists, the edit is repeated to eliminate the error.



- ⑭ Simulation preparation  
 Display "SIMULATOR PORT A" again using **ALT** and **F2** keys.  
 Move the cursor to the upper-left position and select "SIMULATOR."



- ⑮ Simulation execution  
 Press the **F9** key to activate the simulation. The content described in a print statement is then displayed on the screen.



Note: For details of the simulation program, see chapter 3, "Simulation." For details of the editor operation, see chapter 4, "Editor."

## 1.7 Start-Up File

The D5111B loads "System Configuration" after the power is turned on and terminates the system start. After that, a user loads the function module as required and executes it. The same operation is required each time even if a same module is used.

The function module can be executed at any time by entering a command so that the function module or data required for a start-up file is loaded during the power-on sequence. This section describes the start-up file command and how to create the start-up file.

### 1.7.1 Command description

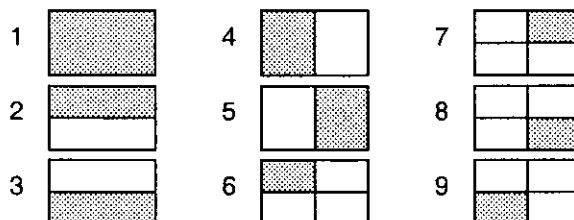
① OPEN command

Function : Loads the function module specified by a function module name and displays it in the position size specified by a window number.

Format : OPEN [Function module name] [Window No.]

Parameter : Function module name  
 MONA (Monitor port A)  
 MONB (Monitor port B)  
 SIMA (Simulation port A)  
 EDITOR (Editor)  
 CONSOLE (Console)  
 UTILITY (Utility)  
 USERDEF (User definition)

Window number



<Example > OPEN MONA 3

② PATH command

Function : Changes the path name of the system area specified by a type name.

Format : PATH [Type] [Path name]

Parameter : Type  
DAT (Measurement data)  
SET (Setting data)  
PRG (Program)  
MSG (Message)  
Path name;  
FD0: Floppy disk  
SD0: Hard disk

<Example> PATH DAT SD0:/DATA/

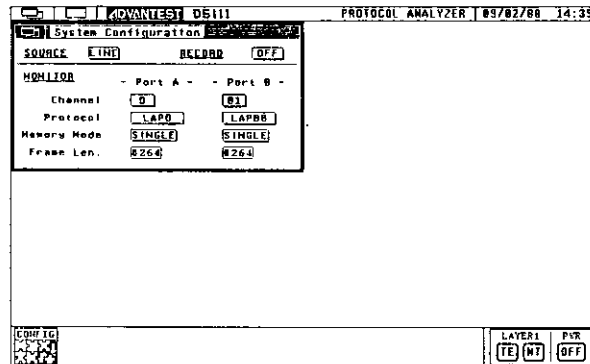
**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**1.7 Start-Up File**

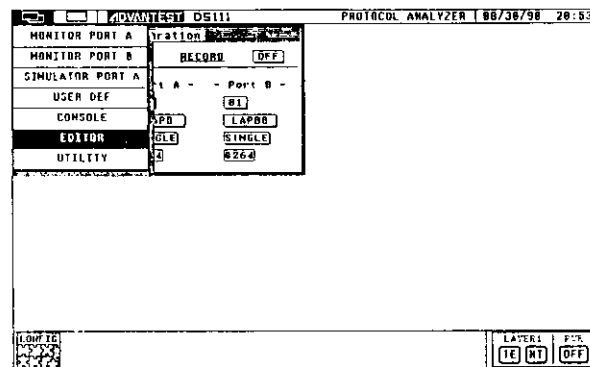
**1.7.2 Start-up file creation**

Use an editor to create the start-up file. The procedure to activate the editor is described below.

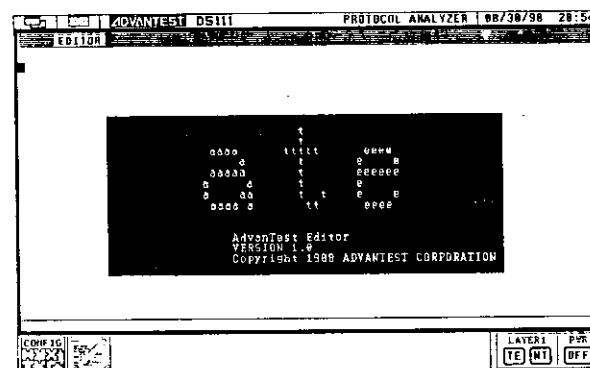
- ① Turn the power on and start the system. (The figure below shows the default screen.)



- ② Press the **F1** key to display the function module selection menu and select "EDITOR" using **▲** and **▼** keys.



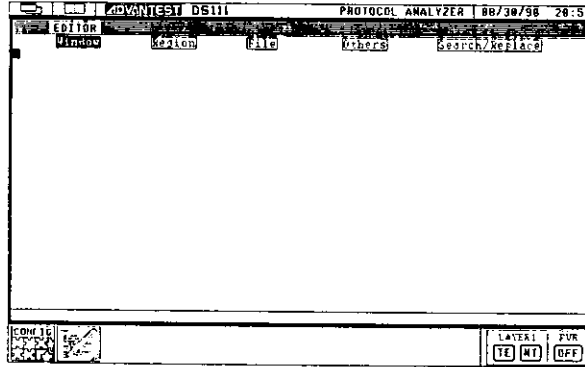
- ③ Press the space key; the editor screen is displayed.



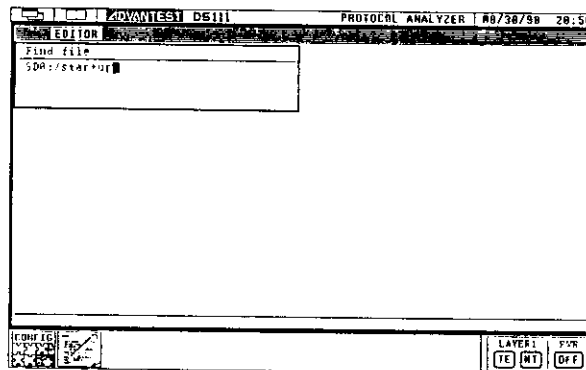
**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**1.7 Start-Up File**

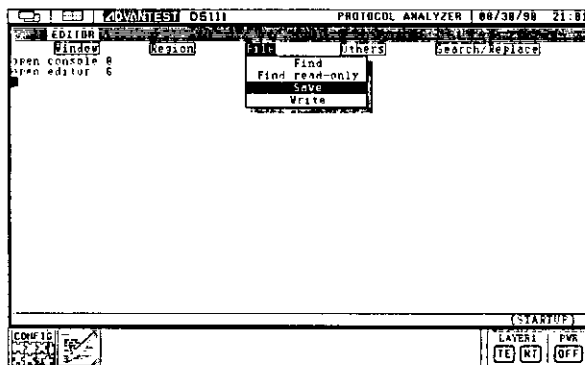
- ④ Press the F3 key; the edit function selection menu is displayed in the upper position of the screen.



- ⑤ Move the cursor to "File" and press the space key. The save/load function selection popup menu is then displayed. Select "Find" and enter "startup" in "Find file."



- ⑥ Press the return key. The content of a "STARTUP" file is displayed on the screen. (The "STARTUP" file does not exist in the hard disk at the factory. "New file" is thus displayed in the lowermost position.)

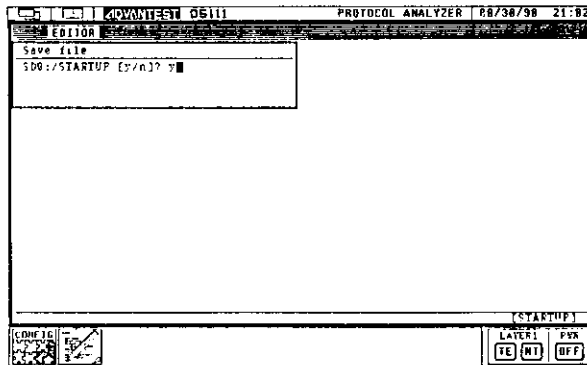




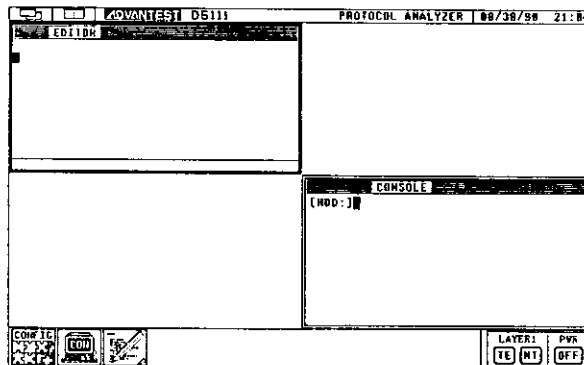
**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**1.7 Start-Up File**

- ⑦ Create the start-up file using a specified command.
- ⑧ Press the F3 again and move the cursor to "File." Then, press the space key.
- ⑨ Select "Save" and enter "STARTUP" as a file name. Then, press the return key.



- ⑩ The creation of the start-up file is completed. Turn the power off, then on. The start operation is then initiated. The function module described in the start-up file is finally read by the system. This can be confirmed using a function module icon in the lower position of the screen.



## 1.8 Connection with Equipment to be Measured

This section describes the connection of the D5111B and the equipment to be measured. This D5111B can evaluate and verify layer-2 and -3 protocols via an interface that is prescribed by CCITT I.430.

### 1.8.1 Pin number and functions

The D5111B has a ① monitor function, ② TE simulation function, and ③ NT simulation function. The functions of signals for each function are shown in Table 1-3.

For the two measurement connectors (S-BUS connectors) on the side panel, same pins numbers are connected each other.

Table 1-3 Pin number and functions

Pin No.	Function		
	① Monitor	② TE simulation	③ NT simulation
1	N.C	N.C	N.C
2	N.C	N.C	N.C
3	T line receive +	Transmission +	Reception +
4	R line receive +	Reception +	Transmission +
5	R line receive -	Reception -	Transmission -
6	T line receive -	Transmission -	Reception -
7	N.C	N.C	N.C
8	N.C	N.C	N.C

### 1.8.2 Connector pin number

The pin number of a measurement connector is shown in the figure below.

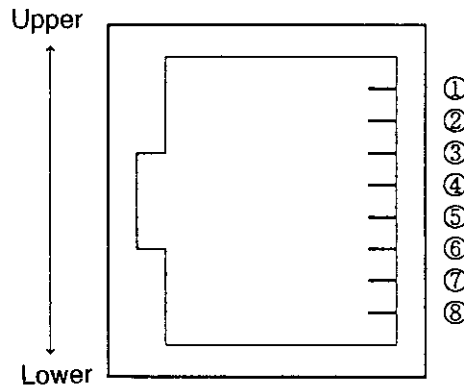


Figure 1-16 Pin number

### 1.8.3 Feeding

The phantom voltage during NT simulation is applied between the T and R lines as shown in Figure 1-17. For normal feeding, the T line becomes positive (+), and the R line becomes negative (-). For reverse feeding, the T line becomes negative (-) and the R line becomes positive (+).

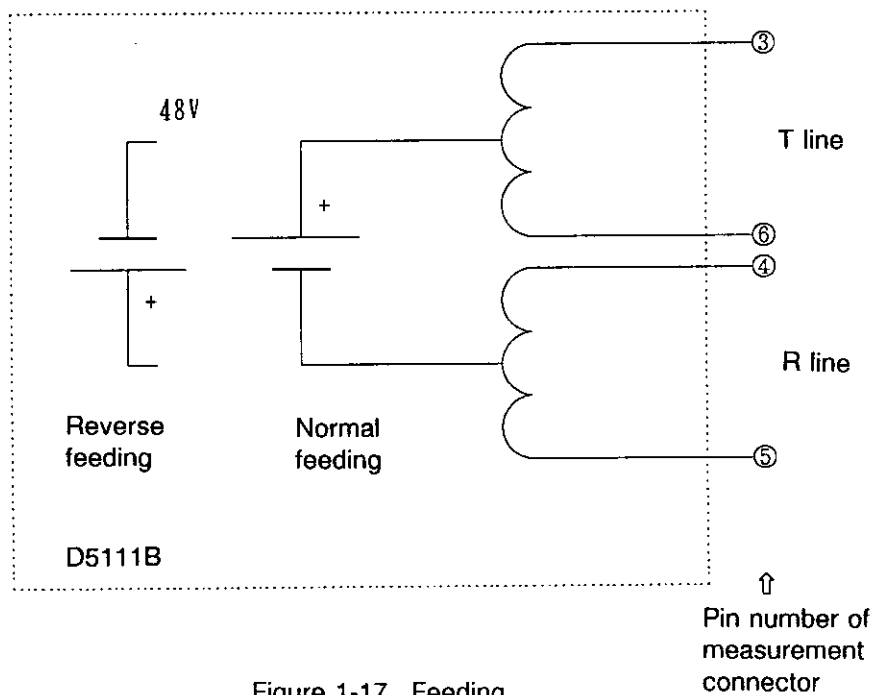


Figure 1-17 Feeding

## 1.9 Performance

### <Major functions>

- Applicable protocol : Layer 1 I.430  
Layer 2 Q.921 (Channel D)  
LAP-B (Channel B) Monitor only  
Layer 3 Q.931 (Channel D)  
X.25 (Channel B) Monitor only
- Layer-1 function : Start/stop state display  
Feeding polarity display (Off/normal/reverse)  
Wiring form ; Short-distance passive bus/extension passive  
bus/point-point  
Terminating resistor; Off/50 $\Omega$  /100  $\Omega$   
Feeding ; Normal/reverse  
(NT simulation: 40 V, 1W (maximum))
- Monitor : Trace function
- Real-time trace ; Traces any two channels simultaneously, and translates and displays them in a real time.
  - Recording in hard disk ; Records D-, B1-, and B2-channel data simultaneously.
  - Replay display ; Replays the D-, B1- and B2-channel data recorded in a hard disk in a real time.
- Translation function
- Layer-2 display
  - Layer-2/3 discrete display and simultaneous display
  - Layer-3 detailed translation display
  - Translation display by user definition
- Trigger function
- Two-type pattern combination and four-stage sequence
  - Delay value; Frame (0 to 9999) or time (0 to 65000msec)
- Filter function
- Records only a four-type pattern frame.
- Recording capacitance
- RAM ; 300K bytes per channel
  - Disk ; About 20M bytes

	Time stamp	
	<ul style="list-style-type: none"><li>● Resolution; 1msec</li><li>● Maximum recording time; 429,496sec (about 4.9 days)</li></ul>	
	Search function	
	<ul style="list-style-type: none"><li>● Time/trigger/frame/pattern/error</li></ul>	
User definition	:	Definition form ; Layer 2/layer 3 (Header block) Definition count; 64
Simulation (D channel only)	:	Language; PSL51 (Protocol simulation language for D5111) Program capacitance ; 64K bytes Timer ; Software timers (5)(10msec resolution) Function ; Common/transparent mode/automatic layer-2 Mode ; Layer-2 mode/layer-2 automatic execution mode Message creation ; Created by an original builder. Message type ; 64 Command ; Frame send, etc.
Editor	:	Edit function (using a function key) <ul style="list-style-type: none"><li>● Window ; Single window/split display</li><li>● Region ; Mark setting/delete/copy</li><li>● File ; Save/load/write/read</li><li>● Others ; Help/compile/error search/line search</li><li>● Search ; Forward/backward/replace/batch replace</li></ul>
Console	:	File management ; Save/load/copy/delete Environment setting ; Date/printer/communication parameter Others ; Help/screen erase
Utility	:	Date and timer/communication parameter/printer
External control	:	Interface specification ; Conforms to EIA-232D. Command ; Recording data output/monitor start and stop/simulation start and stop/disk operation/time setting read/delimiter setting

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**1.9 Performance**

---

Printer : Conforms to Centronics' specifications.

Recommendatory  
model : Corresponding to FP-80 (by EPSON)  
Corresponding to PR-201 (by NEC)

Hard disk : Conforms to SCSI specifications.

<General specifications >

Display : 9-inch/electroluminescence (EL) display (yellow orange)  
640 × 400 dot, bit-map display

Floppy disk unit : 3.5-inch 2DD/2HD. About 720K bytes (2DD) and 1.2M bytes (2HD)

Hard disk unit : 3.5-inch, about 40M bytes. The seek time is 23msec. An SCSI-compatible hard disk can be extended.

Dedicated keyboard : 76-key, ASCII-array

Operating  
environment : Ambient temperature ; +5 to +40°C  
Relative humidity ; 80% or less

Storage  
environment : Ambient temperature ; -10 to +60°C  
Relative humidity ; 80% or less

Power requirement : 90 to 132VAC/180 to 250VAC, 48 to 440Hz

Power consumption : About 90 VA (maximum)

Dimensions : About 340 (W) × 260 (H) × 180 (D)mm

Weight : 9.5kg or less

Accessories supplied : Power cable 1  
System disk 1  
Fuse 1  
Operation manual 1

## 2. MONITOR

This chapter describes the details of a monitor function.

Configuration of chapter 2		
Monitor	2.1 Preparation for monitoring	Configuration setting Monitor loading Trigger condition setting Selective trace setting Translation display screen setting
	2.2 Real-time display	Real-time trace operation Display omission
	2.3 History display	History display method Search function Redisplay by translation layer selection
	2.4 Result display screen	Display screen description (common section) Layer-2 translation display Layer-3 translation display Layer-23 translation display Layer-2 full-translation display Layer-3 full-translation display Layer-23 full-translation display

## 2.1 Preparation for Monitoring

### 2.1.1 Configuration setting

The D5111B can monitor any two channels from among D, B1, and B2 channels at the same time. Ports A and B are activated independently. The channel, protocol, and display screen can also be set independently.

The monitor function can select and monitor the frame at an S/T point or the frame recorded in a hard disk. (The frame data at the S/T point can be directly recorded in the hard disk.)

Press the ALT and F1 keys to display the System Configuration shown in Figure 2-1.

(This screen is displayed during system start.)

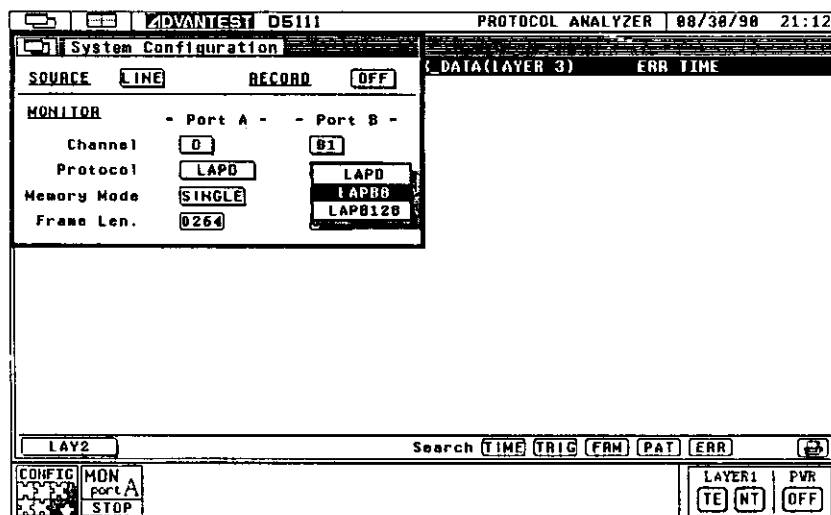
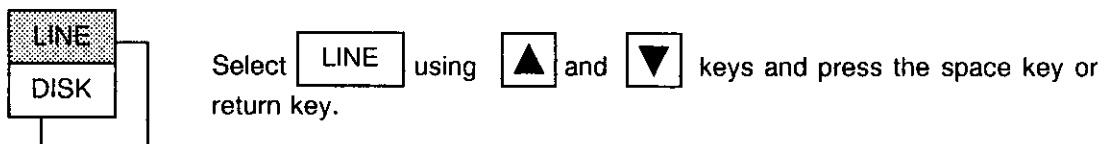


Fig. 2-1 System Configuration screen

#### (1) Source selection

Move the cursor to "SOURCE" (displayed as LINE or DISK ) and press the space key or return key. The popup menu is then displayed.



**Note:** The D5111B displays the popup menu using the space key or return key and select the required item using the ▲ or ▼ key. The popup menu is closed when the item selection is determined using the space key or return key.



(2) Record selection

Move the cursor to "RECORD" and select  OFF or  2B + D .

When the source is set to  LINE , the record selection is able to set.

OFF : It is not recorded in a hard disk.

2B + D : It is recorded in a hard disk.

(3) Channel selection

Move the cursor to "CHANNEL" and press the space key or return key.

The popup menu is then displayed. Select the channel to be monitored for ports A and B using the  ▲ and  ▼ keys.

(4) Protocol selection

Move the cursor to "PROTOCOL" and press the space key or return key.

The popup menu is then displayed. Select the translation protocol for ports A and B using the  ▲ and  ▼ keys.

(5) Memory mode setting

Move the cursor to "Memory Mode" and select  REPEAT or  SINGLE .

REPEAT : Repeat mode  
Acquisition memory is used as the ring memory. New data is overwritten on the old data even if the acquisition memory is filled. Acquisition is thus continued.

SINGLE : Single mode  
A message is displayed when the acquisition memory is filled. The monitor then stops.

(6) Maximum frame length setting

Moves the cursor to "Max Frame Length" and enter the maximum frame length for acquisition (loads line data in the internal memory) (in the range of 3 to 4096).

The length when a flag and frame check sequence (FCS) are excluded is a frame length. The frame with the length exceeding a value that is set here is not left in the acquisition memory.

### 2.1.2 Monitor loading

Load the monitor in port A to drive a monitor.

Press the  key to display the popup menu shown in Figure 2-2.

Next, select  using the  and  keys and press the space key or return key. The popup menu is then closed. When the monitor is loaded, the icon (monitor port A) at the bottom of Figure 2-2 is displayed to indicate that the monitor in port A is loaded.

To drive the monitor in port B, select  in the same way as for port A and load the monitor in port B.

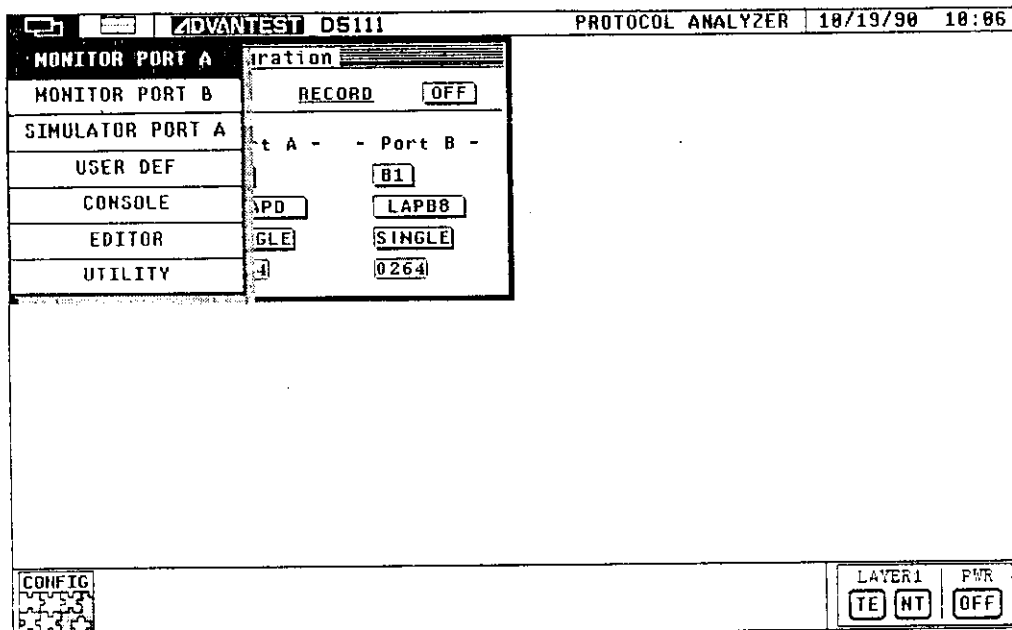


Fig. 2-2 Monitor loading

### 2.1.3 Trigger condition setting

#### (1) Trigger function

The D5111B has a trigger function that terminates acquisition (loads line data in the internal memory) when all setting conditions are satisfied. The trigger consists of four blocks (1 to 4). Two frame patterns based on an OR condition can be registered each in these blocks.

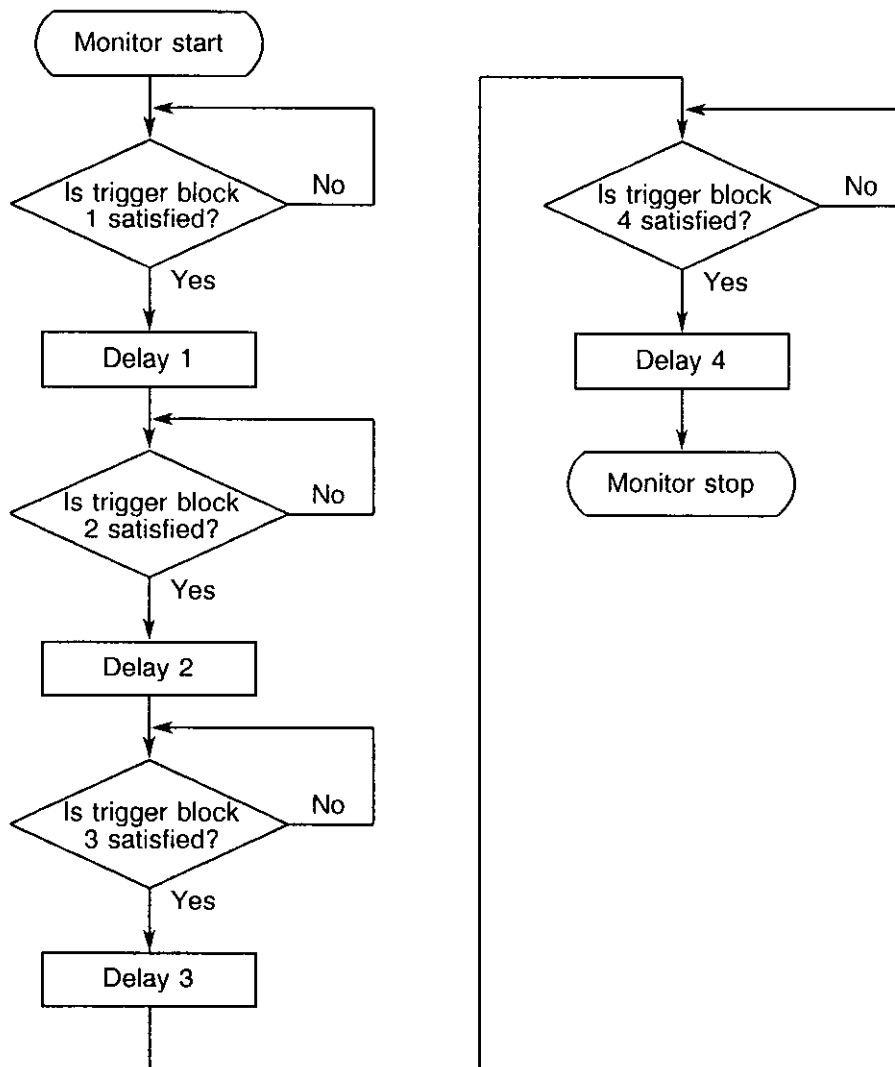


Fig. 2-3 Flowchart for trigger function

Whether the acquisition frame satisfies the condition of trigger block 1 when the monitor starts is judged. When the condition of trigger block 1 is satisfied, the condition of trigger block 2 is judged after a delay. If the condition of trigger block 4 is satisfied, the monitor stops after a delay.

If one of the two patterns that can be registered each in the blocks coincides with the pattern of an acquisition frame, the trigger block is satisfied. If the two patterns in one trigger block are set to  OFF  , the condition of the trigger block is regarded to have already been satisfied. The condition of the next trigger block is then judged without delay.

If one of the two patterns in one trigger block is set to  OFF  , only the condition of the pattern that is set to  ON  is judged.

If an error occurs in the acquisition frame when the ERROR mode is set to  ENABLE  , the trigger block is regarded to have been satisfied, irrespective of whether the pattern coincides or not. The condition of the next trigger block is then judged after a delay. The trigger condition is not judged when the whole trigger switch is set to  OFF  .

The trigger is thus established, and the monitor operation does not stop.

(2) Trigger switch setting

The third-line  OFF or  ON display from the top of the screen is used for a whole trigger switch (see Figure 2-4).

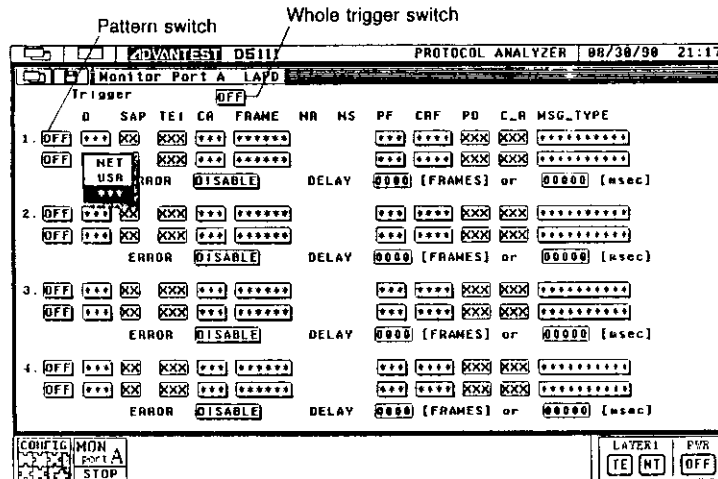


Fig. 2-4 Trigger switch setting

When the trigger switch is set to  ON , the trigger is activated (see the flowchart in Figure 2-3).

When the trigger switch is set to  OFF , the trigger is not activated. The monitor operation does not stop.

(3) Pattern switch setting

The  OFF or  ON display on the left end of each pattern is a switch that sets whether the pattern should be used for the judgment of the trigger condition. The block when the two patterns of each block are set to  OFF differs in handling from the block when only one pattern is set to  OFF . (For more details, see (1), "Trigger Function.")

(4) Direction setting

The "D" in Figure 2-4 is used to set the frame direction.

- NET : Frame transferred by NT
- USR : Frame transferred by TE
- \*\*\* : Don't care. Frame transferred by TE and NT.

(5) SAP setting

The "SAP" in Figure 2-4 is used to set the frame SAPI.

through  : SAPI value (displayed in decimals).  
Cannot be set in ranges other than this range.

: Don't care.

(6) TEI setting

The "TEI" in Figure 2-4 is used to set the frame TEI.

through  : TEI value (displayed in decimals).  
Cannot be set in ranges other than this range.

: Don't care.

(7) CR setting

The "CR" in Figure 2-4 is used to set the frame command or response.

: The TE send frame is a command, and the NT send frame is a response.

: The TE send frame is a response, and the NT send frame is a command.

: Don't care.

(8) Frame type setting

The "FRAME" in Figure 2-4 is used to set the type of a frame.

(9) NR/NS setting

The "NR" and "NS" in Figure 2-4 are used to set the receive sequence number N (R) and send sequence number N (S).

(10) PF bit setting

The "PF" in Figure 2-4 is used to set the pole and final bits.

: The pole and final bits are set to "0."

: The pole bit is used for a command, and the final bit for a response.

: Don't care.

(11) CRF setting

The "CRF" in Figure 2-4 is used to set the call number flag of layer 3.

Orig	:	Activation side of call. (Call-out side)
Dest	:	Call-in side
XXX	:	Don't care.

(12) PD setting

The "PD" in Figure 2-4 is used to set the protocol identifier.

0	through	256	:	"8" (Q.931) is usually set.
XXX	:			Don't care.

(13) C R setting

The "C R" in Figure 2-4 is used to set the call number of layer 3.

0	through	127	:	Call number
XXX	:			Don't care.

(14) MSG TYPE setting

The "MSG TYPE" in Figure 2-4 is used to set the type of a message in layer 3.

(15) ERROR mode setting

The "ERROR" in Figure 2-4 is used to set the error mode.

ENABLE	:	The condition of the trigger block is regarded to have been satisfied when errors (abort detection, short frame error, FCS error, and nonoctet error) are detected in the frame.
DISABLE	:	No frame error is detected.

(16) Setting of delay frame count and delay timer

The "DELAY" in Figure 2-4 is used to set the number of frames and the timer time.

Frame count	:	0	through	9999	(FRAMES)
Timer time	:	0	through	65000	(msec)

In delay mode, the trigger block does not enter the next trigger block until a frame is set to the acquisition memory proportionally to the number set in the number of delay frames or until the time set in a delay timer passes. If the number of frames of the trigger block or the delay time at that time is set to 0 no delay occurs.

### 2.1.4 Selective trace setting

(1) Selective trace function

The selective trace is used when you wish to acquisition a specified frame (e.g., two or more TEs are connected to the S/T reference point to set the communication of a specified TE and NT as a target).

Up to four frame patterns can be registered in the selective trace. Patterns other than frame pattern that is set are not left in the acquisition memory when the selective trace is set to  ON . All frames that are passed through a line are set to the acquisition memory when the selective trace is set to  OFF .

Any pattern is regarded to disaccord if four pattern switches are all set to  OFF  when the selective trace is set to  ON . No frame is then set to the acquisition memory.

(2) Selective trace switch setting

The  OFF  or  ON  display in the third line from the top (in Figure 2-5) is used for a whole selective trace switch.

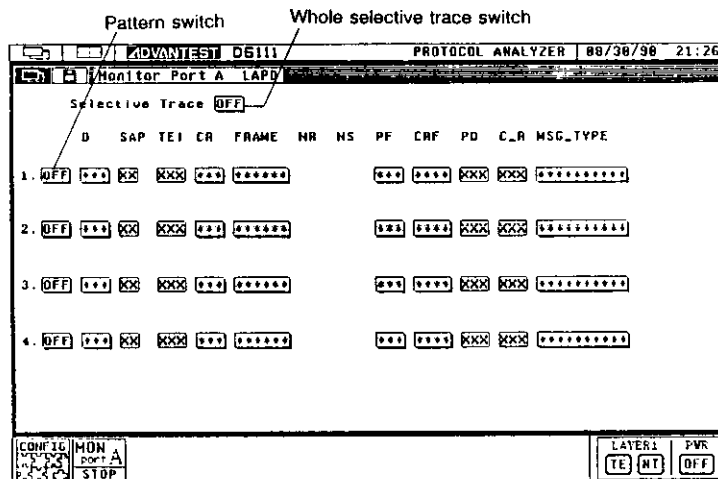


Fig. 2-5 Selective trace switch setting

ON  : Loaded in the acquisition memory when the frame that coincides with the frame pattern of which pattern switch is set to  ON  is received.

OFF  : All frames that are passed through a line are loaded in the acquisition memory.



(3) Pattern switch setting

See 2.1.3 (3), "Pattern Switch Setting."

(4) Direction setting

See 2.1.3 (4), "Direction Setting."

(5) SAP setting

See 2.1.3 (5), "Direction Setting."

(6) TEI setting

See 2.1.3 (6), "TEI Setting."

(7) CR setting

See 2.1.3 (7), "CR Setting."

(8) Frame type setting

See 2.1.3 (8), "Frame Type Setting."

(9) NR and NS setting

See 2.1.3 (9), "NR and NS Setting."

(10) PF bit setting

See 2.1.3 (10), "PF Bit Setting."

(11) CRF setting

See 2.1.3 (11), "CRF Setting."

D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL

2.1 Preparation for Monitoring

---

(12) PD setting

See 2.1.3 (12), "PD Setting."

(13) C\_\_R setting

See 2.1.3 (13), "C\_\_R Setting."

(14) MSG\_\_TYPE setting

See 2.1.3 (14), "MSG\_\_TYPE Setting."

### 2.1.5 Translation screen setting

The translation screen is displayed when  Translation  is selected in Figure 2-6.

The monitor start can be executed on only this screen. The monitor stop can be executed on any screen.

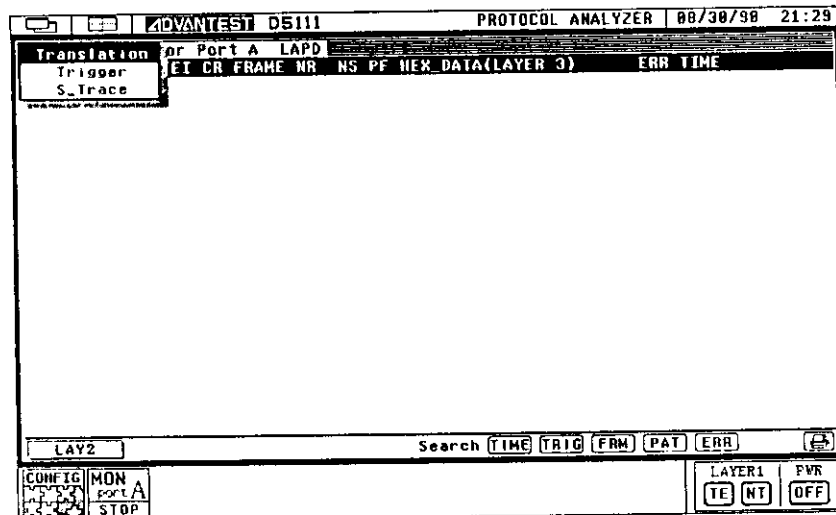


Fig. 2-6 Translation screen setting

#### (1) Translation layer selection

Move the cursor to  LAY2  (  LAY3  ,  LAY23  ,  LAY2FULL  ,  LAY3FULL  ,  L23FULL  ,  HEX or  HEXFULL ) in the lower-left position of Figure 2-6 and select the translation layer using a popup menu. This setting can also be performed before monitor start. If the setting is switched with the translation screen displayed after monitor stop, the display is also changed.

## 2.2 Real-Time Display

### 2.2.1 Real-time trace operation

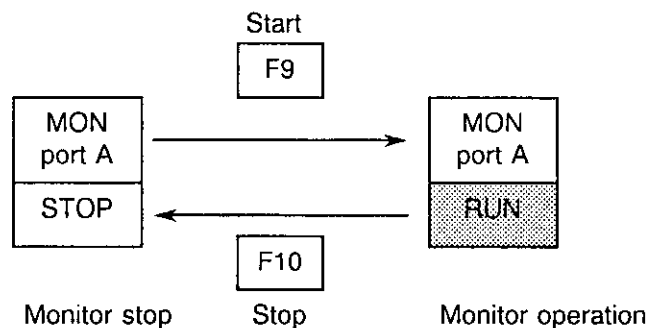
In the real-time trace, the frame passed through a line is displayed on the screen in real time when the D5111B monitor is in the run state.

If the desired setting is completed in 2.1, "Preparation for Monitoring", the subsequent operation is easy to perform. (If it is not, set in accordance with section 2.1.)

Switch the screen to the monitor screen (load it in accordance with section 2.1.2 if the monitor is not loaded) and select Translation using the popup menu to display the translation screen (see Figure 2-6).

Press the F9 key on this screen. The monitor then starts. The icon displayed in the lower position of the screen is changed when the monitor starts.

Press the F10 key. The monitor then stops.



### 2.2.2 Display omission

When a frame is received during monitor operation, the frame content is displayed on the translation screen. However, the frame display is omitted when the number of frames to be received increases and the display is late for the frames. The frames are translated at intervals of a few frames to several ten frames. The omitted frame is displayed when the monitor stops.

## 2.3 History Display

### 2.3.1 History display method

The history display appears when the monitor stops in real-time trace state. The history display appears again in accordance with the set translation layer when the translation layer is switched with the history displayed. Press the  and  keys.

The screen is then scrolled up and down. (For more details, see 2.1.5 (1), "Translation Layer Selection.")

### 2.3.2 Search function

#### (1) Time search

The popup menu is displayed when Search  is selected with the history displayed as shown in Figure 2-7. Enter the acquisition time of a frame you wish to search.

Press the  or  key. The frame of a time to be set or the next frame is then displayed (if the frame of a time to be set does not exist, the next frame is displayed).

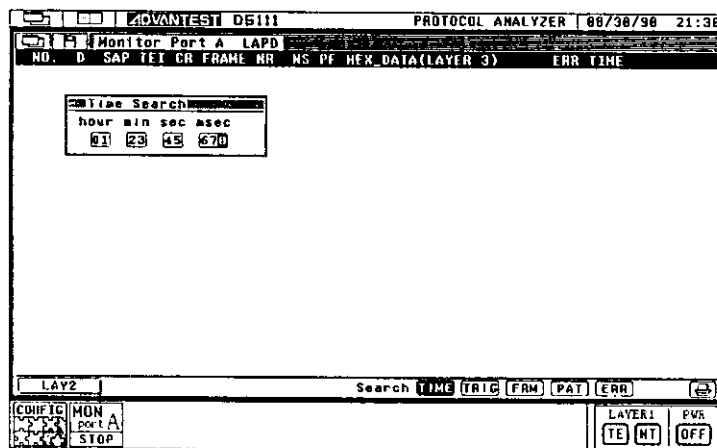


Fig. 2-7 Time search

#### (2) Trigger search

The popup menu is displayed when Search  is selected with the history displayed. Press the  or  key. The trigger-point frame is then displayed. If no trigger point exists, the display does not change.

#### (3) Frame number search

The popup menu is displayed when Search  is selected with the history displayed as shown in Figure 2-8. The frame number you wish to search can be set.

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

2.3 History Display

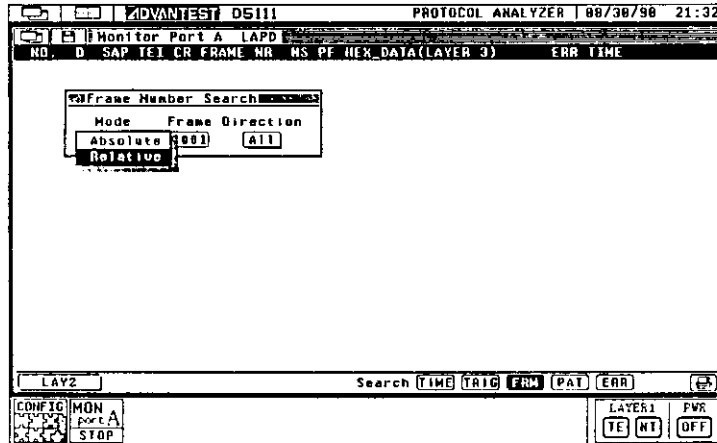


Fig. 2-8 Frame number search

<Mode setting>

- Absolute : Searches the frame number that is set, irrespective of the cursor position.
- Frame number : Sets the frame number to be searched.
- Direction : Sets the direction of a frame to be searched.
- Relative : Searches the line that is relatively located to the current cursor line.

(4) Pattern search

The popup menu is displayed when Search  PAT is selected with the history displayed as shown in Figure 2-9.

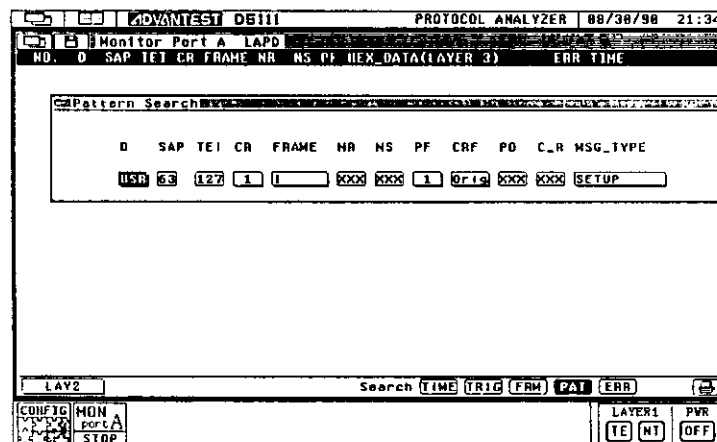




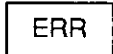


Fig. 2-9 Pattern search

Set the frame pattern you wish to search. The frame pattern is set in the same way as for the trigger condition.

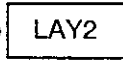

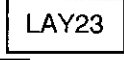
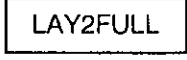
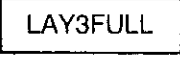
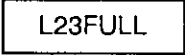


- ① Direction setting  
See 2.1.3 (4), "Direction Setting."
- ② SAP setting  
See 2.1.3 (5), "SAP Setting."
- ③ TEI setting  
See 2.1.3 (6), "TEI Setting."
- ④ CR setting  
See 2.1.3 (7), "CR Setting."
- ⑤ FRAME setting  
See 2.1.3 (8), "Frame Type Setting."
- ⑥ NR and NS setting  
See 2.1.3 (9), "NR and NS Setting."
- ⑦ PF setting  
See 2.1.3 (10), "PF Bit Setting."
- ⑧ CRF setting  
See 2.1.3 (11), "CRF Setting."
- ⑨ PD setting  
See 2.1.3 (12), "PD Setting."
- ⑩ C\_R setting  
See 2.1.3 (13), "C R Setting."
- ⑪ MSG\_TYPE setting  
See 2.1.3 (14), "MSG TYPE Setting."

Press the  key after the pattern is set. The search is then started upward from the current cursor position. If a frame that coincides with the pattern to be set is detected, the frame is displayed. Press the  key. The search is started downward from the current cursor position.

(5) Error search

Move the cursor to Search  with the history displayed and press the  key. The search is started upward from the current cursor position. If a frame in which an error occurs is detected, the frame is displayed. Press the  key. The search is started downward from the current cursor position.

### 2.3.3 Redisplay by translation layer selection

Move the cursor to  ( , , ,  , , , or  ) and display the popup menu.

The history redisplay is switched each time the translation layer is selected. Translate in accordance with the selected layer.



## 2.4 Translation Screen

### 2.4.1 Translation screen description (common section)

The display on the translation screen varies depending on the selection of the translation screen

(  ,  ,  ,  ,  ,  ,  , or  ).

The display common to all translation screens is shown in the figure below.

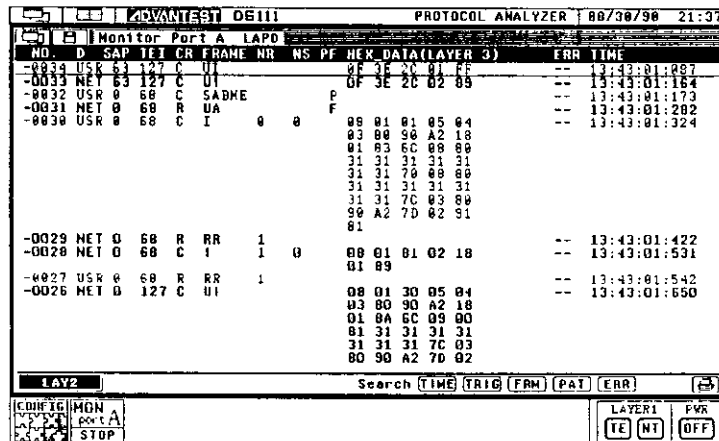


Figure 2-10 Translation screen

< Items in uppermost line of the screen >

- |       |   |   |   |  |
|-------|---|---|---|--|
| No.   | : | Frame numbers 0 through 2047. The frame that was set to the acquisition memory has a lower-frame number. Used when a search is made by the search frame number. |   |  |
| D     | : | Direction. Identifies whether the frame is generated by TE (displayed as "USR" or "U") or NT (displayed as "NET" or "N").                                       |   |  |
| SAP   | , | SA  | : | SAPI. Displayed by 0 through 63 (in decimals). |
| TEI   | : | Displayed by 0 through 127 (in decimals).   |   |  |
| CR    | : | Displays C (command) and R (response).  |   |  |
| FRAME | : | Displays the type of a layer-2 frame ("SABME" or "I").  |   |  |
| NR    | : | Displays N (R).   |   |  |
| NS    | : | Displays N (S).   |   |  |
| PF    | : | Displays P (pole) and F (final) bits.   |   |  |

CRF	:	Identifies Orig (call out) and Dest (call in).
PD	:	Displays the protocol identifier in decimals.
C_R	:	Displays the call number.
MSG_TYPE	:	Displays the message type for a frame in which layers 1 through 3 exist.
ERR	:	Displays the type of the error if an error exists. A: Abort error S: Short frame error F: Frame check sequence (FCS) error N: Nonoctet error
TIME	:	Time stamp. Displays the time when a frame is set to the acquisition memory.

#### 2.4.2 Layer-2 (LAY2) translation display

Translation is performed with layer 2 in the center. Layer 3 is displayed in hexadecimal.

The translation display of one frame is limited to 12 lines. For the frame exceeding 12 lines, a ▶ mark is displayed between the frame number and direction displays. To display the whole frame content, set the layer-2 full translation display.

### 2.4.3 Layer-3 (LAY3) translation display

Only layer 3 of a frame in which layer 3 exits is translated and displayed. The frame ("SAMBE" or "DISC") in only layer 2 is not displayed. The message type, information element name, and time stamp in layer 3 are displayed. For the information element, only the name is translated. The content of the information element is only displayed in hexadecimal.

The translation display of one frame is limited to 12 lines. For the frame exceeding 12 lines, a ▶ mark is displayed between the frame number and direction displays. To display the whole frame content, set the layer-3 full translation display.

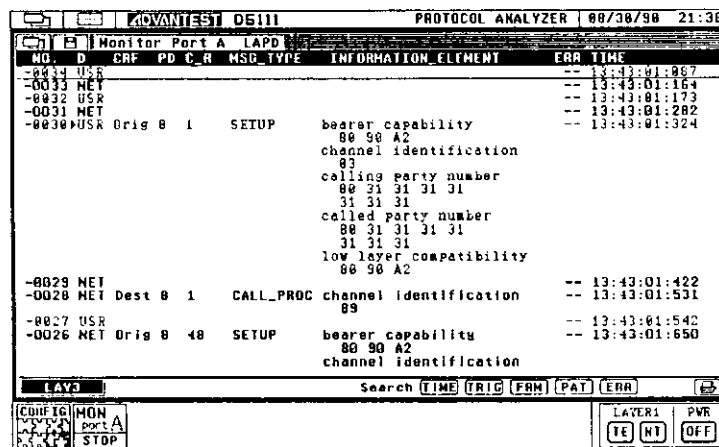


Figure 2-11 Layer-3 translation display

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**2.4 Translation Screen**

**2.4.4 Layer-23 (LAY23) translation display**

The layer-2 display, and layer-3 message type and information element name are displayed. The time stamp is not displayed.

The translation display of one frame is limited to 12 lines. For the frame exceeding 12 lines, a ▶ mark is displayed between the frame number and direction displays. To display the whole frame content, set the layer-23 full translation display.

NO.	D	SA	TET	CR	FRAME	NR	NS	PF	CRF	PD	C	R	MSG. TYPE	INFORMATION_ELEMENT
-0031	U	63	127	C	UI									
-0032	U	63	127	C	UI									
-0031	M	0	68	C	SABME				P	0F	3E	2C	02	89
-0031	M	0	68	R	UA				F					
-0030	U	0	68	C	I		0	0		0rg	0	1	SETUP	bearer capability 80 90 A2 CH identification 83 calling party number 80 31 31 31 31 31 31 31 called party number 80 31 31 31 31 31 31 31 low layer compati. 60 90 A2
-0029	M	0	68	R	RR	1								
-0028	M	0	68	C	I	1	0			Dst	B	1	CALL_PROC	CH identification 89
-0027	U	0	68	R	RR	1								
-0026	M	0	127	C	UI					0rg	B	48	SETUP	bearer capability 80 90 A2 CH identification

Figure 2-12 Layer-23 translation display

### 2.4.5 Layer-2 full (LAY2 FULL) translation display

The same in content as the layer-2 translation display except that the whole frame content is displayed.

NO.	D	SAP	TEI	CR	FRAME	NR	NS	PF	HEX DATA (LAYER 3)	ERR TIME
-0030	USR	03	127	C	UI				0F 3E 2C 01 FF	-- 13:43:01:087
-0032	NET	0	68	C	UI				0F 3E 2C 02 89	-- 13:43:01:164
-0031	USR	0	68	C	SABME					-- 13:43:01:173
-0030	USR	0	68	R	UA					-- 13:43:01:202
-0030	USR	0	68	C	I	0	0		00 01 01 05 04 03 80 90 A2 10 01 83 6C 00 80 31 31 31 31 31 31 31 70 00 80 31 31 31 31 31 31 31 7C 03 80 90 A2 7D 02 91 81	-- 13:43:01:324
-0029	NET	0	68	R	RR	1				-- 13:43:01:422
-0028	NET	0	68	C	I	1	0		00 01 01 02 10 01 89	-- 13:43:01:531
-0027	USR	0	68	R	RR	1				-- 13:43:01:542
-0026	NET	0	127	C	UI				00 01 30 05 04 03 80 90 A2 10 01 8A 6C 00 80 01 31 31 31 31 31 31 31 7C 03 80 90 A2 7D 02	-- 13:43:01:650

Figure 2-13 Layer-2 full translation display

### 2.4.6 Layer-3 full (LAY3 FULL) translation display

The same in content as the layer-3 translation display except that the whole frame content is displayed.

NO.	D	CRF	PD	C	R	MSG TYPE	INFORMATION ELEMENT	ERR TIME
-0030	USR					SETUP	bearer capability 80 90 A2	-- 13:43:01:087
-0032	NET						channel identification 83	-- 13:43:01:164
-0031	USR						calling party number 80 31 31 31 31 31 31 91	-- 13:43:01:173
-0030	USR	Orig	0	1			called party number 80 31 31 31 31 31 31 31	-- 13:43:01:202
-0030	USR	Orig	0	1			low layer compatibility 80 90 A2	-- 13:43:01:324
-0029	NET						high layer compatibility 91 81	-- 13:43:01:422
-0028	NET	Dest	0	1		CALL_PROC	channel identification 89	-- 13:43:01:531
-0027	USR							-- 13:43:01:542
-0026	NET	Orig	0	48		SETUP	bearer capability	-- 13:43:01:650

Figure 2-14 Layer-3 full translation display

### 2.4.7 Layer-23 full (LAY23 FULL) translation display

The same in content as the layer-23 translation display except that the whole frame content is displayed.

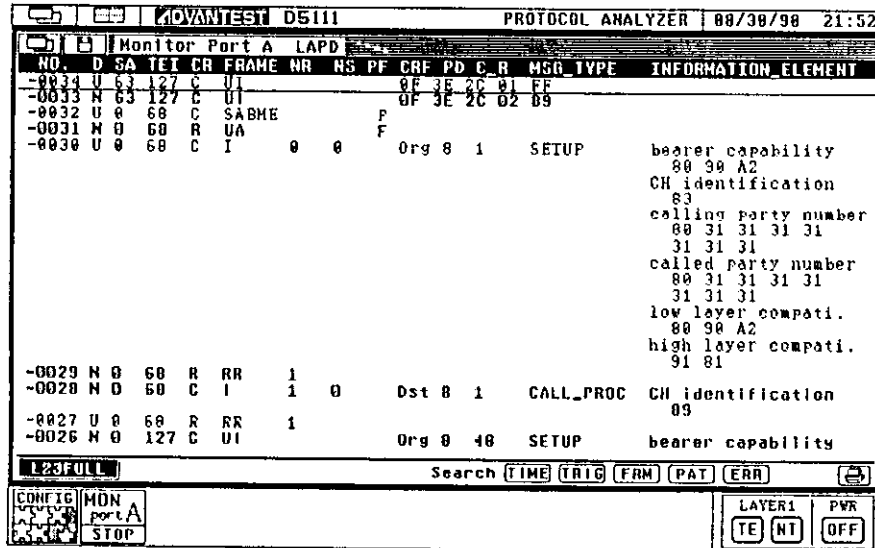


Figure 2-15 Layer-23 full translation display

### 2.4.8 Mark in translation display

Mark	Meaning
<b>I</b>	It indicates the frame is set to trigger stare.
➔	It indicates this length of a frame exceeds the configuration menu's. The data is acquired within the limits of the frame length to be set and is displayed within the limits of the setting length.
▶	It indicates the contents of a frame are displayed only part way. Selects the layer in <input type="text" value="LAY2FULL"/> , <input type="text" value="LAY3FULL"/> , <input type="text" value="LAY23FULL"/> , or <input type="text" value="HEXFULL"/> when the whole frame content is displayed.

### 3. SIMULATION

This chapter describes the preparation and languages required for simulation execution.

Configuration of chapter 3		
Simula- tion	3.1 Simulation language	PSL51 PSL51 language specification
	3.2 Programming structure	PSL51 programming structure example Specification statement of simulation
	3.3 Simple program creation example	Character display on screen Simulation using program Error code using compiler Error message and error list Program execution Message creation by message builder Object program save/load Message data save/load
	3.4 Execution of simulation function without program	
	3.5 Syntax of simulation language (PSL51)	
	3.6 Common function	
	3.7 Function for transparent mode	
	3.8 Function for layer-2 automatic mode	

### 3.1 Simulation Language

#### 3.1.1 PSL51

The simulation language (PSL51: Protocol simulation language for D5111) has the functions described below.

- (1) Sending of a frame created by user
- (2) Judgment of the frame content to be received

#### 3.1.2 PSL51 language specification

Table 3-1 PSL51 language specification

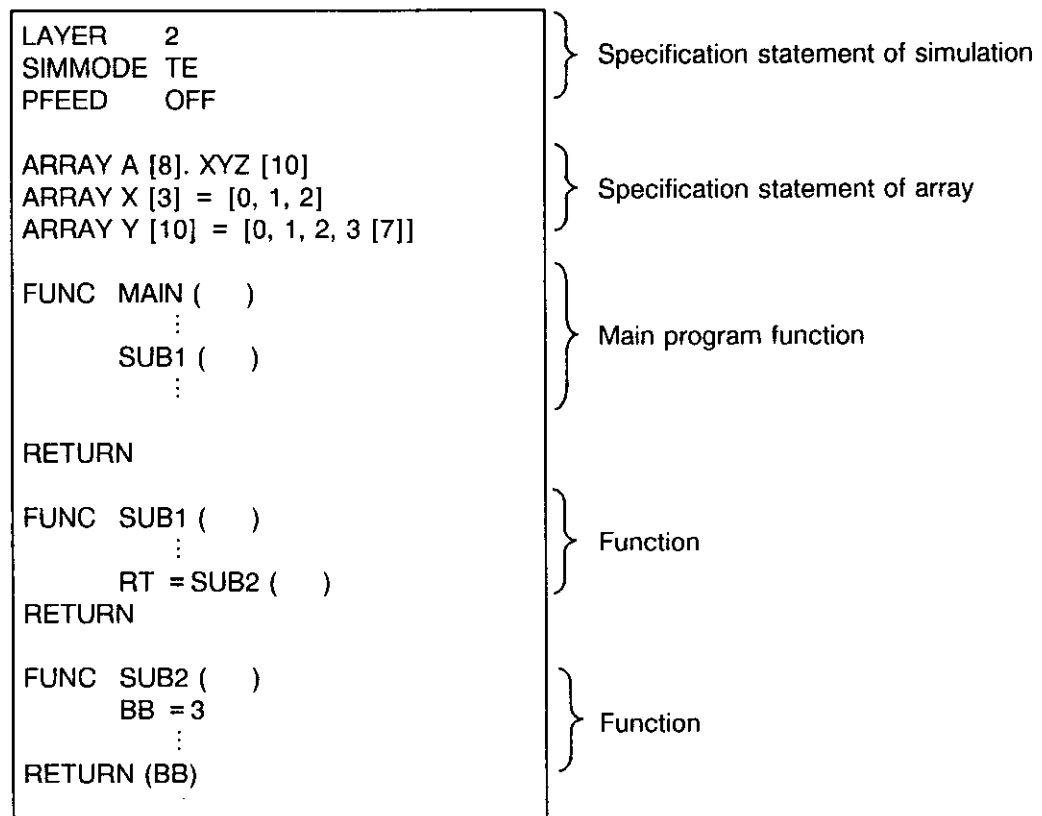
Data type	Integer type	Sign binary bits (-2147483648 to +2147483647)
Array	One-dimensional array	
Operator	+ - * /	Addition Subtraction Multiplication Division
Relational operator	> < >= <= == !=	Greater than Less than Greater than or equal to Less than or equal to Equal to Not equal to
Logical operator	!	NOT
Logical operator per bit	& 	AND per bit OR per bit
Function	FUNC	This function has an integer-type value.
Statement	Assignment statement IF statement FOR statement WHILE statement CASE statement EXIT statement RETURN statement	val = al + BB*XYZI  IF a=1 THEN X=0 ELSE X=1 END FOR i=0 TO 100 DO..... END WHILE x= =1 ..... END CASE XY+1 OF '1' '2'..... END Exits from the loop of a WHILE statement. Returns from the function.



## 3.2 Programming Structure

### 3.2.1 PSL51 programming structure example

A basic programming structure example of PSL51 is shown below.



The program is executed from main program function MAIN ( ). A MAIN ( ) function must thus exist in the program.

### 3.2.2 Specification statement of simulation

To perform the simulation in channel D, the mode to be used must be specified in advance by a specification statement. The following are defined by the specification statement:

- (1) LAYER 2/3
  - 2: Executed in transparent mode.
  - 3: Executed in layer-2 automatic mode.

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**3.2 Programming Structure**

---

(2) SIMMODE TE/NT

TE: Executed in TE mode.

NT: Executed in NT mode.

(3) PFEED OFF/NORM/RVS

OFF: No feeding

NORM: Performs normal feeding.

RVS: Performs reverse feeding.

Note: This specification statement is valid only when the D5111B is set to NT.

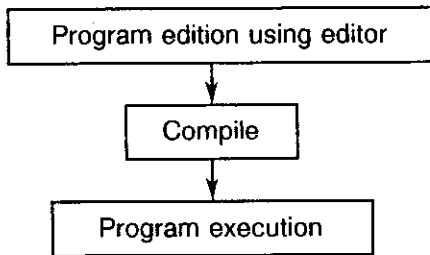
<Declaration example by specification statement >

LAYER	3	...	Layer-2 automatic mode (Layer-3 mode)
SIMMODE	NT	...	NT mode
PFEED	RVS	...	Reverse feeding

### 3.3 Simple Program Creation

#### 3.3.1 Character display on screen

This section describes how to display characters on the screen using a program. The program is executed in accordance with the procedure below.



(1) Use of editor

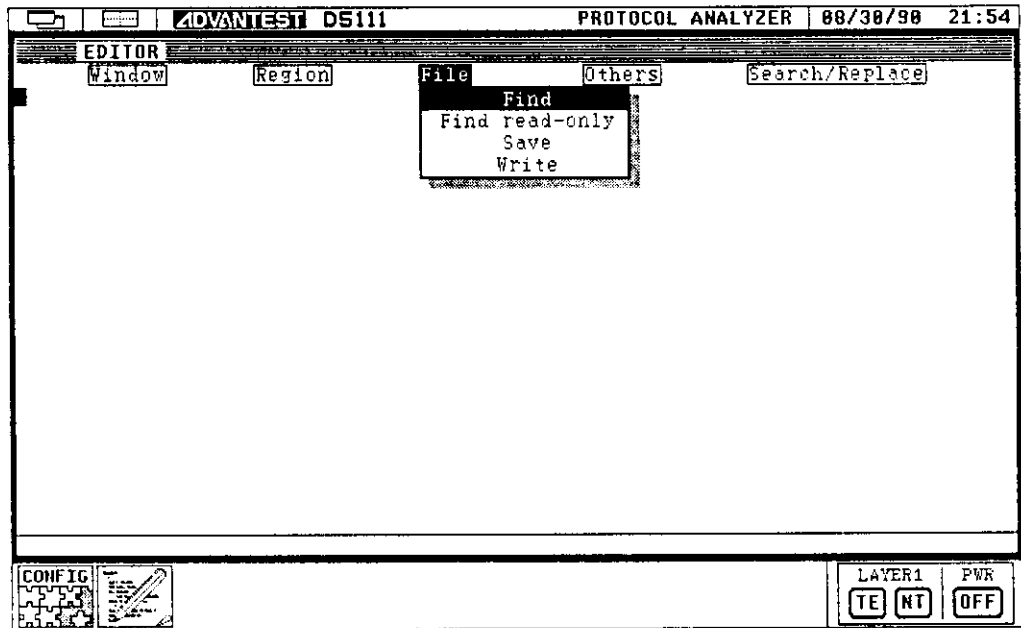


Figure 3-1 Use of editor

- ① Press the  key to display the popup menu.  
Move the cursor to the  position using the  and  keys and press the space key or return key.
- ② Confirm that "ate" is displayed on the screen.
- ③ Press the  key to display the popup menu.  
Move the cursor to the  position using the  and  keys and press the space key (or return key). Then, move the cursor to the  position using the  and  keys and press the space key (or return key).
- ④ "Find File" is displayed, and the file name is asked. Enter "moji.prg" (not exceeding 8 characters + 3 characters) as the file name and press the return key.
- ⑤ A program can be created in the "moji.prg" file. Press the  key again.  
The popup menu then disappears. Next, key in the program below.





```
FUNC MAIN ( )  
  PRINT ( "Welcome to D5111B " )  
RETURN
```

(Enter the command by upper-case characters.)

(2) Compile






- ① Press the  key to display the popup menu again.  
Move the cursor to the  position using the  and  keys and press the space key (or return key). Then, move the cursor to the  position using the  and  keys and press the space key (or return key).
- ② The popup menu for the compile is displayed. The file name to be edited is then displayed. Press the return key.
- ③ Whether a created file should be saved in the disk or not is asked ("SD0: /PRG/MOJI.PRG [y/n]?). Key in "Y" and press the return key.
- ④ The screen is split horizontally, and the compile result is displayed. If no error occurs, the display below appears.

```
File name : [SD0: /PRG/MOJI.PRG]  
Finished
```

- ⑤ If an error is displayed, move the cursor to the **Window** position using  and  keys and press the space key or return key. Then, move the cursor to the **Only** position using the  and  keys and press the space key (or return key). The pre-compile state returns. Modify the content of the line number in which an error occurs in accordance with the error message and compile again.
- ⑥ Press the F3 key to close the popup menu.

(3) Program execution

Execute the program after the compile is terminated normally.

- ① Press the F1 key to display the popup menu. Select **SIMULATOR PORT A** using the  and  keys and press the return key or space key.
- ② Move the cursor to the  position and press the space key or return key. The popup menu is then displayed. Select **SIMULATOR** using the  and  keys and press the space key or return key.
- ③ Press the **F9** key. The program is then executed. Confirm that the display below appears on the screen.

Welcome to D5111B

The **F10** key is used to stop a program. The program above is stopped automatically.

Pressing of the **F10** key is thus required.

### 3.3.2 Simulation using program

The simulation using a program must be executed in accordance with the flowchart below.

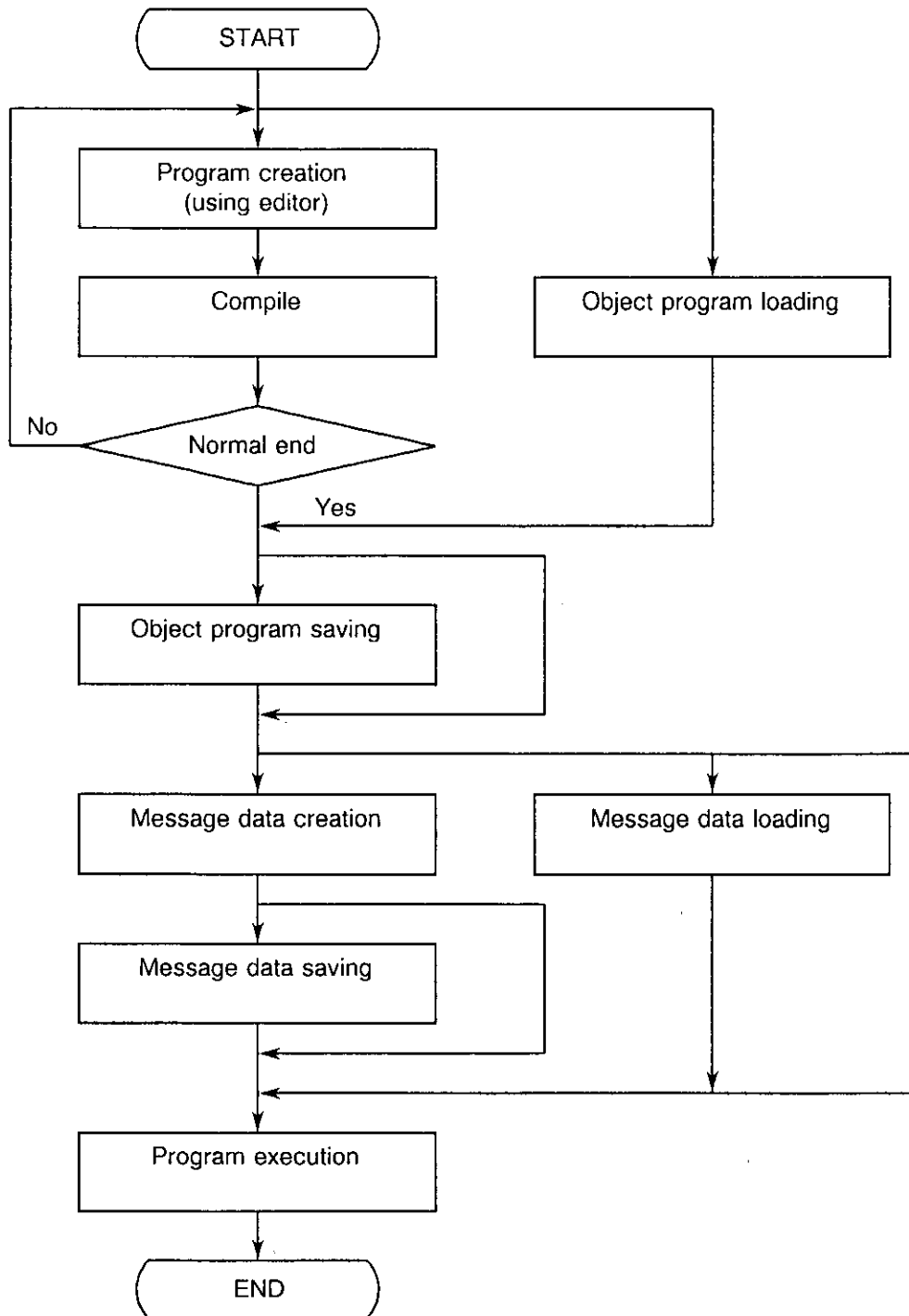


Figure 3-2 Procedure for simulation execution

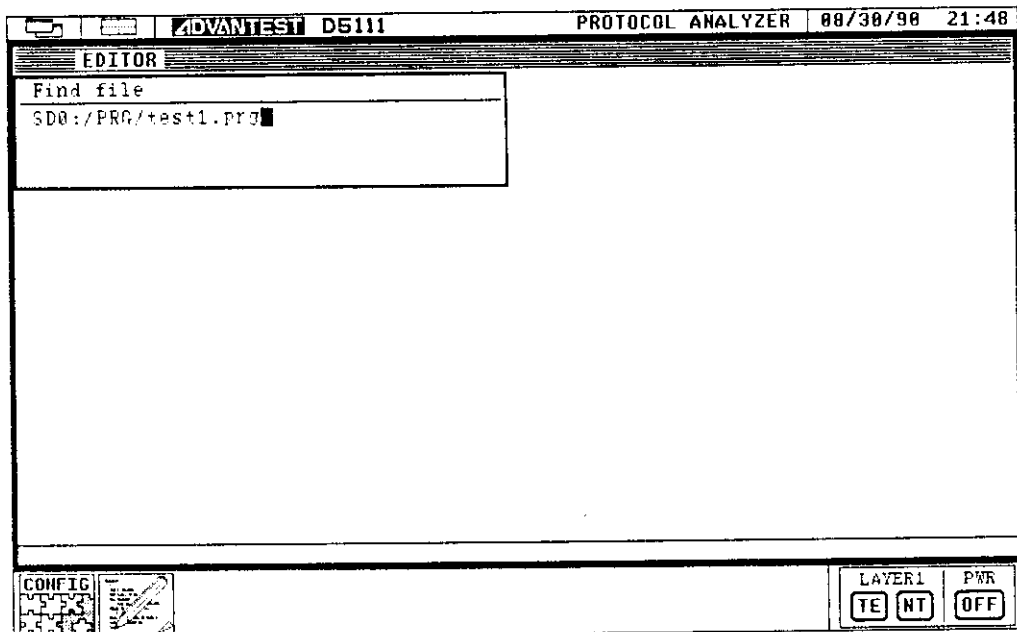


Figure 3-3 Program creation

(1) Program creation

The program is created using an editor.

- ① Press the **F1** key to display the popup menu. Move the cursor to the EDITOR position using the **▲** and **▼** keys and press the return key.
- ② Press the **F3** key to display the popup menu. Move the cursor to the **File** position using the **◀** and **▶** keys and press the return key. Then, move the cursor to the Find position using the **▲** and **▼** keys and press the space key.  
(The operation above can also be performed by pressing the C-x and C-f keys.)  
C-x (Press the **CTRL** and **X** key simultaneously.)  
C-f (Press the **CTRL** and **F** key simultaneously.)
- ③ Key in the file name to be edited and press the return key.
- ④ Press the **F3** key again. The popup menu then disappears.
- ⑤ Create a program.

(2) Program compile

Compile the program after the program creation is completed.

- ① Press the  key to display the popup menu. Move the cursor to the  position using the  and  keys and press the space key (or return key). Then, move the cursor to the  position using the  and  key and press the space key (or return key). (The operation above can also be performed by pressing the C-x and C-e keys.)

C-x (Press the  and  key simultaneously.)

C-e (Press the  and  key simultaneously.)

- ② Key in the file name to be compiled. The file name is displayed when a file is edited in advance using an editor. If the displayed file name is OK, press the return key. If you wish to compile another file, change the file name to the desired file name and press the return key.
- ③ If the file to be compiled is modified using an editor, but not saved in a disk, the file saving is asked. Enter "Y" to save the file and press the return key (the file you wish to compile is then saved). If you enter "N" and press the return key, the file before modified using an editor is compiled.
- ④ The screen is split, and the compile result is displayed. Confirm that no error occurs. If an error occurs, modify the file again using an editor and execute the compile. The popup screen appears or disappears each time the  key is pressed. To use the editor again, press the  key to display the popup menu. Then, select  in the  menu and press the space key. The former display state returns. (The operation above can also be performed by pressing the C-X 1 key.)

C-x (Press the  and  key simultaneously.)



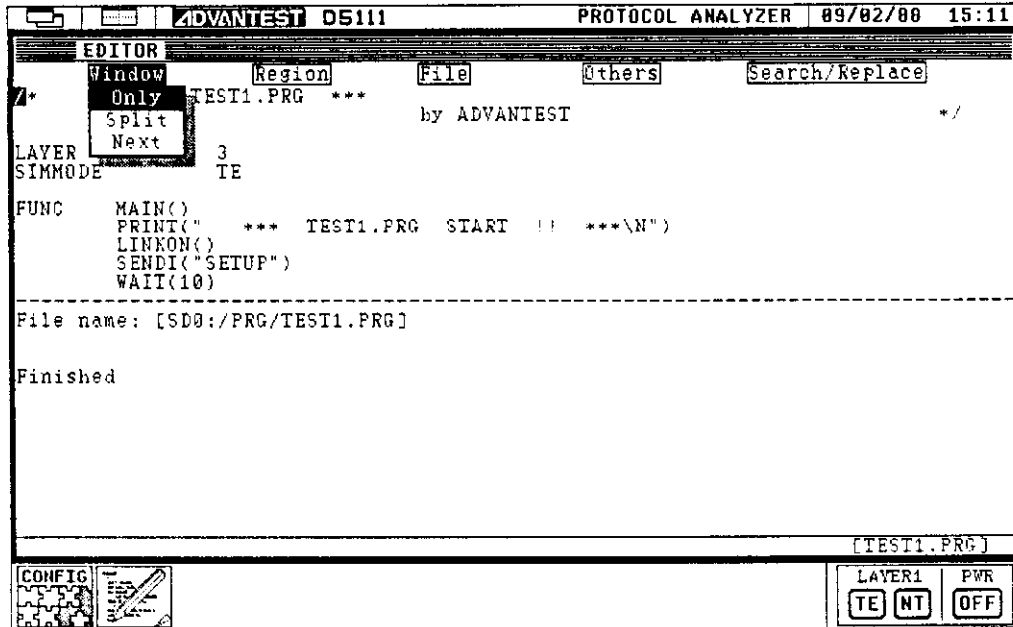


Figure 3-4 Program creation

### 3.3.3 Error code using compiler

The error code below for an error is displayed on the screen when a PSL51 language is compiled.

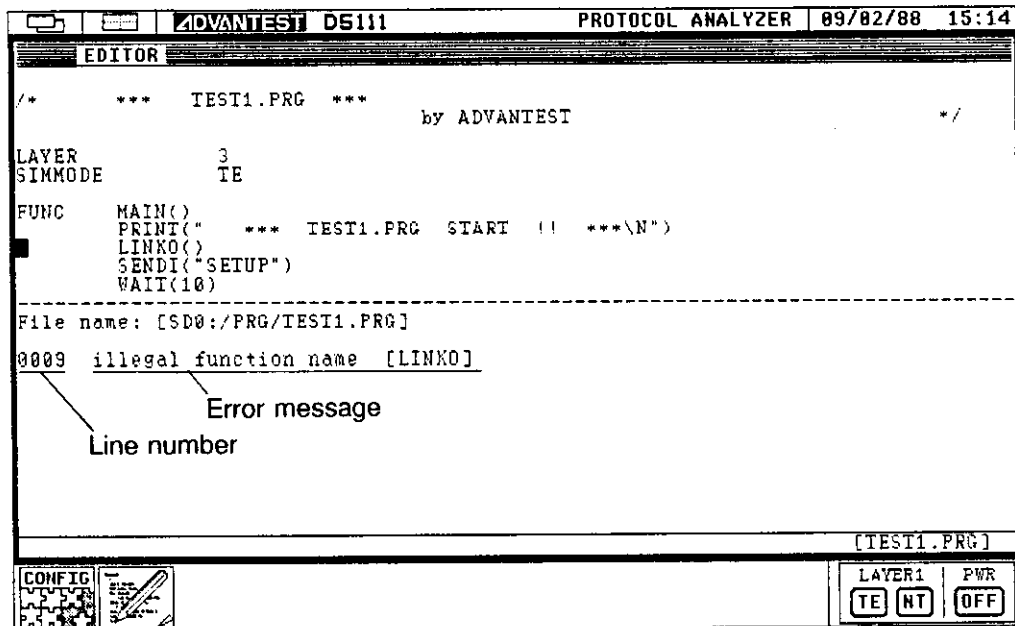


Figure 3-5 Error code

### 3.3.4 Error message and error list

Table 3-2 Error message (1/4)

Error message	Description
missing (,) or illegal character for operator	The parentheses are not balanced normally. An inappropriate character [ ] is used as the operating element.
illegal use for RESERVED word	The reserved word [ ] is used as a name.
illegal operator	An operating element exists continuously. The operating element [ ] exists in the position where an operated element should exist.
illegal function name or label	The function name and label are used as a variable name [ ].
PRINT format error	The notation of a PRINT statement is not proper.
SYNTAX error	Syntax error
invalid value	The value is invalid.
illegal function name	An inappropriate name [ ] is used as the function name in the form of a function designator.
redeclared function name	The function name [ ] has already been declared.
the left side is illegal	The left part of assignment is not correct.
ARRAY needs suffix	There is no subscript in the array name.
illegal ARRAY use	Operation is performed for the array name. The array name is used as a subscript. An expression with only the array name is used for the condition part of an IF or WHILE statement.
illegal ARRAY suffix	The subscript of an array is not correct. [ ] exists in the position where ")" should exist.

Note: The characters in brackets [ ] are displayed on the screen.

Table 3-2 Error message (2/4)

Error message	Description
illegal ARRAY size	The array size is specified incorrectly.
illegal separator	A symbol [ ] is used to delimit the initial-value list of an array.
illegal ARRAY name	The name [ ] is inappropriate as an array name.
needs more initial values	The initial value of an array is too low.
too many initial values	The initial value of an array is too high.
illegal arguments	The form of an argument is not correct. ")" does not exist. A character [ ] is used instead of "," or ")".
illegal separator	A symbol [ ] is used to delimit the argument.
illegal argument	The name [ ] is not correct as an argument.
illegal function name	The name [ ] is inappropriate as a function name.
illegal argument number	The number of arguments in a function [ ] is not proper.
undefined function	The function [ ] is undefined.
SYNTAX error for IF	The form of an expression following the IF statement is not correct. [ ] exists in the THEN position. A syntax error occurs in the IF statement. [ ] exists in the END or ELSE position.
illegal control variable for FOR	The control variable of a FOR statement is not correct.
SYNTAX error for FOR	The = position in a FOR statement is not correct. = does not exist.

Note: The characters in brackets [ ] are displayed on the screen.

Table 3-2 Error message (3/4)

Error message	Description
SYNTAX error for FOR	The incremental expression in a FOR state is not correct. TO does not exist. The final-value expression in a FOR statement is not correct. DO does not exist. The statement list in a FOR statement is not correct.
SYNTAX error for WHILE	The form of an expression following the WHILE statement is not correct. The statement list in a WHILE statement is not correct.
illegal label	There is a label in a non-WHILE statement and [ ]. The label [ ] is written incorrectly. [ ] is inappropriate as a label.
redefined label	The label [ ] is defined two or more times.
undefined label	The label [ ] is undefined.
needs label after EXIT	The EXIT "(" is followed not by a label, but [ ].
illegal label	The EXIT label [ ] is written incorrectly.
illegal use for CASE	Thee condition [ ' ' ] in a CASE statement is not correct.
SYNTAX error for CASE	The condition expression in a CASE statement is not correct. OF does not exist. The condition in a CASE statement is not initiated by " ".
illegal RETURN format	The expression in a RETURN statement is not correct. ")" does not exist.
MAIN not found	The function MAIN ( ) in a main program does not exist.

Note: The characters in brackets [ ] are displayed on the screen.

Table 3-2 Error message (4/4)

Error message	Description
variable & array memory overflow	The variable and array exceed the allowable capacitance.
program memory overflow	The program capacitance exceeds the memory capacitance.
temporary memory overflow	The temporary capacitance exceeds the memory capacitance.
print statement area overflow	The capacitance in a PRINT statement exceeds the memory capacitance.
parameter not found	No argument is found.
missing return	No RETURN statement exist.

Note: The characters in brackets [ ] are displayed on the screen.

### 3.3.5 Program execution

Execute the program after the program compile is terminated normally.

- ① Press the  key to display the popup menu. Select "SIMULATOR PORT A" using the  and  keys and press the return key.  
If "SIMULATOR PORT A" has already been selected, press the  and function (  through  ) keys at the same time and call "Simulator Port A."
- ② Press the space key with the cursor put in the  position.  
The popup menu is then displayed. Select "SIMULATOR" using the  and  keys and press the space key (or return key).
- ③ Press the  key on this screen to run the program. Press the  key.  
The program then stops. If the function (SENDF, SENDI, or SENDUI) requiring a message is used by the message builder in a program, the program operates normally only when the message is created by the message builder.

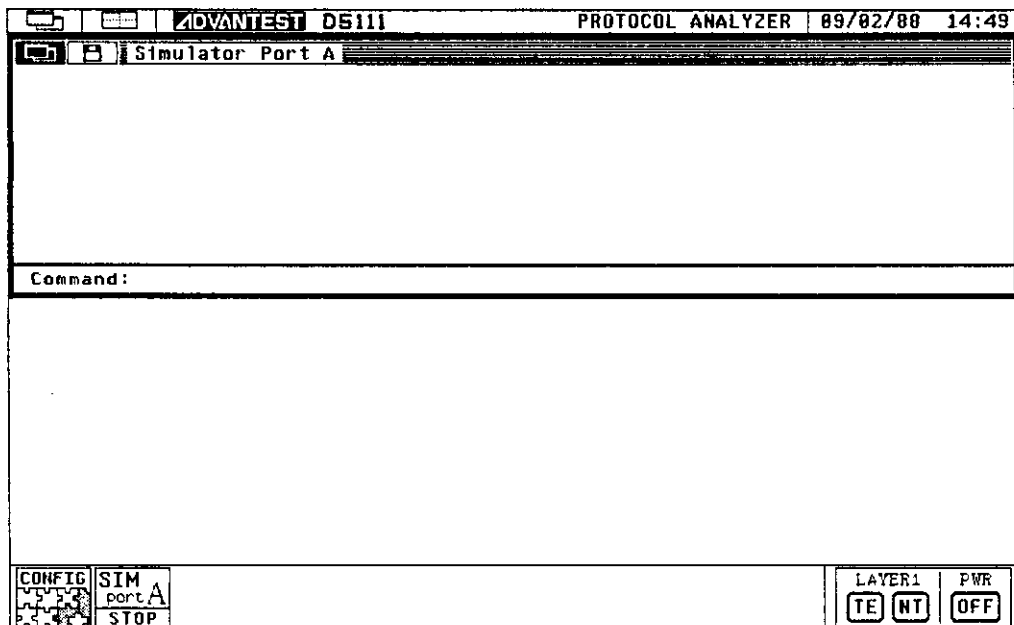


Figure 3-6 Simulation screen

### 3.3.6 Message creation by message builder

The data of a frame to be sent is created by a message builder.

- ① Press the  key to display the popup menu. Select "SIMULATOR PORT A" using the  and  keys and press the space key (or return key).  
If "SIMULATOR PORT A" has already been loaded, reselect it by pressing the  and function keys at the same time.
- ② Press the space key with the cursor put in  position.  
The popup menu is then displayed. Select "MESSAGE" using the  and  keys and press the space key (or return key).
- ③ Messages of Nos. 1 to 64 can be created. Pay attention to the following when using a message builder:

The LAYER2 information created by a message builder is ignored in layer-2 automatic mode. N (S) and N (R) used in transparent mode is automatically set referring to the V (S) and V (R) values set in a program. The P/F bit value is valid when it is specified by an argument.

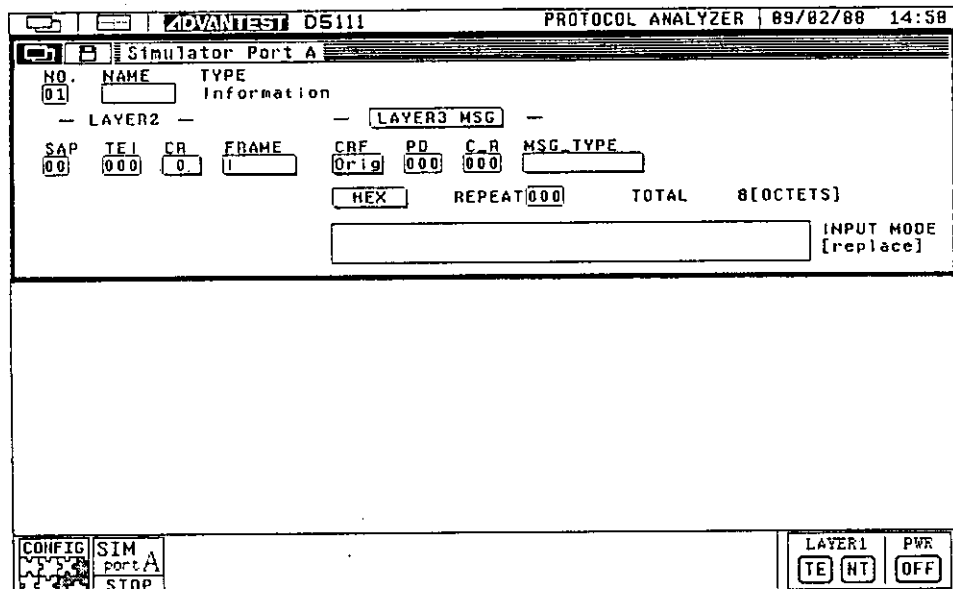
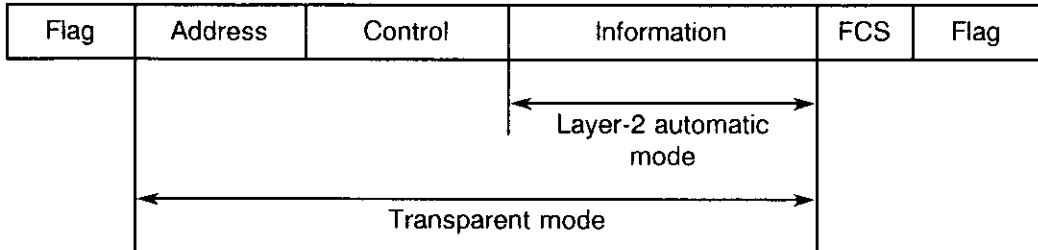


Figure 3-7 Message creation (message builder)

- (1) Use of message builder

The frame shown on the next page can be created using a message builder.



Frames of Nos. 1 to 64 can be created using a message builder. The name of the created frame must be given. The created frame is validated when it is named. (Up to six characters can be entered.)

In the address and control fields, the SAPI, TEI, C/R bit, and frame type can be set. The N (R) and N (S) values cannot be set using a message builder. The N (R) and N (S) values are automatically added referring to the V (S) and V (R) values set using simulation functions (INCVS, INCVR, SETVS, and SETVR). The P/F bit cannot also be set using a message builder. This bit must be set by an argument when a SENDF function is executed.

The data in the information field is set by three methods. Move the cursor to the top of the screen center. Press the space key. Then, you can select one of the three formats below to create a message.

- ① LAYE3 MSG : Sets the layer-3 common block in detail.
- ② LAYE2 INFO : Enters the information field directly.
- ③ NOTHING : No information field

Move the cursor to the position of the format you wish to select using the and keys. Press the return key. The creation format can then be changed.

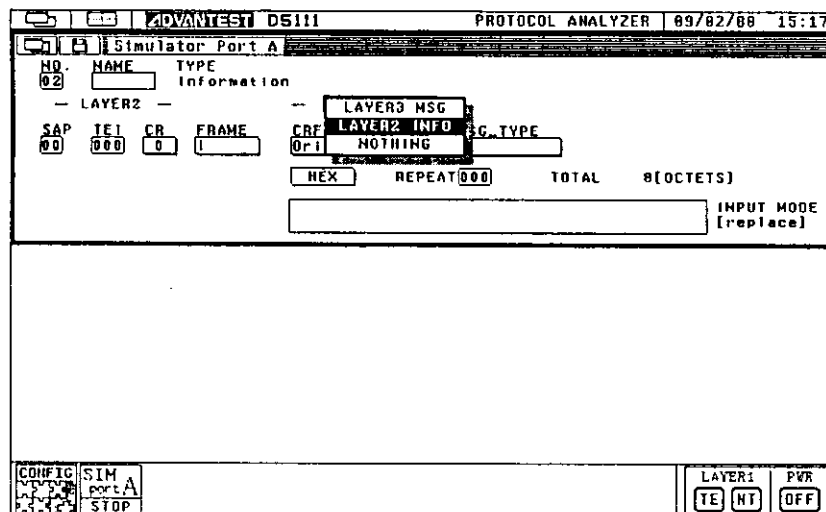
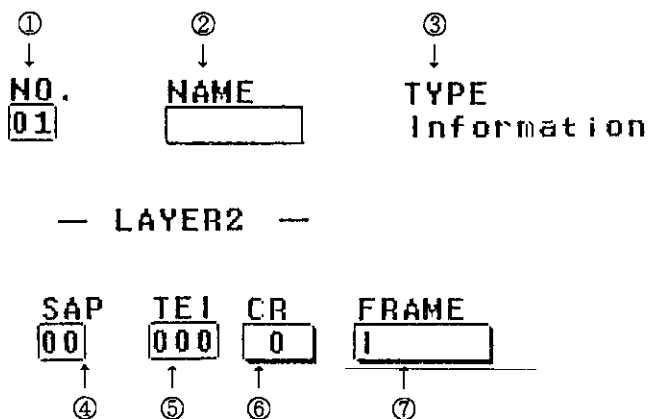


Figure 3-8 Use of message builder



(2) Creation of address and control fields using message builder

Select the message builder screen. The format shown below appears on the left of the message builder screen.



Each item on the screen above is described below.

- ① NO.  
Displays the frame number. Messages of Nos. 1 to 64 can be created. Directly enter the numeric you wish to set. The numeric that is set can be changed using the  and  keys.
- ② NAME  
Enter the frame name. Up to six characters can be entered.
- ③ TYPE  
Displays the type of a frame that is being created currently. The type to be displayed is as follows:
  - Information
  - Supervisory
  - Unnumbered
- ④ SAP  
Enter the SAPI value. Value 0 through 63 can be set.
- ⑤ TEI  
Enter the TEI value. Value 0 through 127 can be set.

⑥ CR

Sets the command/response value. Move the cursor to the CR position and press the space key (or return key). The popup menu below is then displayed.





0
1

Select the value you wish to set using the  and  keys. Press the space key (or return key) to set the desired value.

⑦ FRAME

Sets the type of a frame. Move the cursor to the FRAME position and press the space key (or return key). The popup menu below is then displayed.

I	RR	RR
RNR	RNR	REJ
REJ	SABME	DISC
PM	UA	FRMR
UI	XID	XID

Move the cursor to the type position of a desired frame using the , , , and  keys. Press the space key (or return key) to set the desired frame. The TYPE display may then change.

For the type of a frame displayed on the popup menu screen, the frame defined by a user is displayed sequentially.

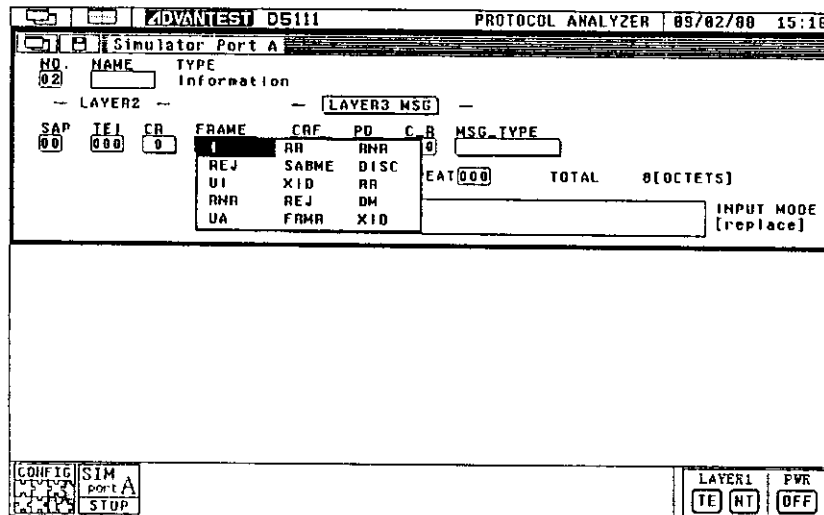


Figure 3-9 Frame setting

(3) Creation of information field using message builder

Move the cursor to the top of the screen center and press the space key. The popup menu below is then displayed.

LAYER3 MSG
LAYER2 INFO
NOTHING

The setting of the information field value varies according to the selection of the three items above.

(a) LAYER3 MSG format

How to set a layer-3 common block in detail and enter a message is described below.

<LAYER3 MSG setting screen >

The screenshot shows the following fields and options:

- ① LAYER3 MSG (highlighted)
- ② CRF (Call-Number Flag) with a sub-menu showing 'Orig' and 'Dest'
- ③ PD (Protocol Discriminator) set to '000'
- ④ C\_R (Call Reference) set to '000'
- ⑤ MSG\_TYPE (Message Type) field
- ⑥ HEX (Hexadecimal format)
- ⑦ REPEAT (Repeat count) set to '000'
- ⑧ A large empty text input field
- ⑨ TOTAL (Total length) set to '8[OCTETS]'
- ⑩ INPUT MODE [replace]

The data above is described sequentially.

① LAYER3 MSG

Move the cursor to the LAYER3 MSG position and press the space key (or return key). The popup menu below is then displayed.

LAYER3 MSG
LAYER2 INFO
NOTHING

To change the setting format of the information field, move the cursor to the position of the format you wish to change and press the space key (or return key). An example when LAYER3 MSG is selected is explained below.

② CRF

Sets the call-number flag. Move the cursor to the CRF position and press the space key (or return key). The popup menu below is then displayed.

Orig
Dest

Select the flag you wish to set using the and keys and press the space key (or return key). The new flag is then set.

- ③ PD  
Sets the protocol identifier. Enter the numeric. Enter numeric "8" when a Q931 protocol is used.
- ④ C\_R  
Sets the call number.
- ⑤ MSG\_TYPE  
Sets the message type. Move the cursor to the MSG\_TYPE position and press the space key (or return key). The popup menu is then displayed. Move the cursor to the desired message position and press the space key (or return key). The message is then selected.
- ⑥ HEX  
Specifies the input mode when transmission data is directly created in the area of step ⑧. The input mode is classified into HEX, ASCII, JIS, and EBCDIC. The input mode is set in accordance with the procedure below.

Move the cursor to the position of step ⑥. Press the space key to display the popup menu below.

HEX
ASCII
JIS
EBCDIC

Move the cursor to the position of the input mode you wish to set and press the space key. The popup menu is then closed, and the input mode that is set is displayed in the position of step ⑥.

- ⑦ REPEAT  
Sets the number of repetition times in the area of step ⑧. The REPEAT area to be set is repeated proportionally to the number of repetition times. If numeric "0" is set, no repetition is performed. If numerics other than "0" are set, the font of the REPEAT area set in the area of step ⑧ is changed.
- ⑧ Direct input area of data  
Enter data directly. The input mode is classified into HEX, ASCII, JIS, and EBCDIC. The input mode can be set in the area of step 6). If the REPEAT area is set, the corresponding input data is repeatedly sent in proportion to the number of repetition times.

In the area of step ⑧, data can be edited by pressing the keys below.

Reset of repeat position

C-r (Press the 

CTRL
------

 and 

R
---

 keys at the same time.)

Specification of repeat start position

C-b (Press the  and  keys at the same time.)

Specification of repeat end position

C-e (Press the  and  keys at the same time.)

Deletion of data following the cursor position

C-k (Press the  and  keys at the same time.)

All data clear

C-c (Press the  and  keys at the same time.)

Input mode selection

C-i (Press the  and  keys at the same time.)

Move the cursor to the head in line

C-← (Press the  and  keys at the same time.)

Move the cursor to the last in line

C-→ (Press the  and  keys at the same time.)

⑨ TOTAL

Displays the total number of octets when a frame is assembled from the created data. A flag and frame check sequence (FCS) are not then counted.

⑩ INPUT MODE

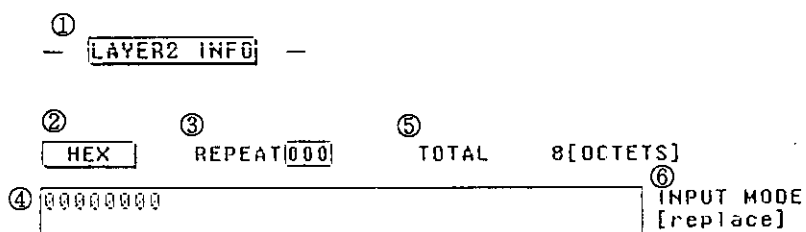
Displays the setting of a data's input mode. The input mode can be selected by pressing the C-i key (pressing the  and  keys at the same time) in the area of step ⑧.

[replace] : Replace mode  
[insert] : Insert mode

(b) LAYER2 INFO format

How to directly enter the information field is described below.

<LAYER2 INFO setting screen >



① LAYER2\_INFO



Move the cursor to the LAYER2 INFO position and press the space key. The popup menu below is then displayed.

LAYER3 MSG
LAYER2 INFO
NOTHING

To change the setting format of the information field, move the cursor to the position of the format you wish to change and press the space key. An example when LAYER INFO is selected is explained below.

② HEX

Specifies how to enter the data when transmission data is input to the area of step 4). The input mode is classified into HEX, ASCII, JIS, and EBCDIC. To specify the input mode, press the space key and display the popup menu below.

Move the cursor to the position of the desired input mode using the  and  keys and press the space key. The desired input mode is then selected.

HEX
ASCII
JIS
EBCDIC

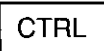

③ REPEAT

Sets the number of repetition times. The REPEAT area that is set is repeated proportionally to the number of repetition times. If numeric "0" is set, no repetition is performed. If numerics other than "0" are set, the font of the REPEAT area set in the area of step ③ is changed.

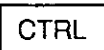

④ Direct input area of data

Enter data directly. The input mode is classified into HEX, ASCII, JIS, and EBCDIC. The input mode can be set in the area of step 2). If the REPEAT area is set, the corresponding input data is repeatedly send in proportion to the number of repetition times.



Reset of repeat position

C-r (Press the  and  keys at the same time.)

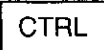

Specification of repeat start position

C-b (Press the  and  keys at the same time.)



Specification of repeat end position

C-e (Press the  and  keys at the same time.)

Deletion of data following the cursor position

C-k (Press the  and  keys at the same time.)

All data clear

C-c (Press the  and  keys at the same time.)

Input mode selection

C-i (Press the **CTRL** and **I** keys at the same time.)

Move the cursor to the head in line

C-← (Press the **CTRL** and **←** keys at the same time.)

Move the cursor to the last in line

C-→ (Press the **CTRL** and **→** keys at the same time.)

⑤ TOTAL

Displays the total number of octets when a frame is assembled from the created data. The total number of octets when REPEAT is set is also be displayed. A flag and frame check sequence (FCS) are then not counted.

⑥ INPUT MODE

Displays the setting of a data's input mode. Enter "I" while pressing the control (CTRL) key in the area of step ④. The input mode can then be selected.

[replace] : Replace mode  
[insert] : Insert mode

(c) NOTHING format

The NOTHING format is selected when a message not including an information field is created.

<NOTHING setting screen >

① — **NOTHING** —

① NOTHING

Move the cursor to the NOTHING position and press the space key. The popup menu below is displayed.

LAYER3 MSG
LAYER2 INFO
<b>NOTHING</b>

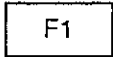


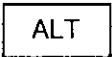



To change the setting format, move the cursor to the desired position and press the space key. The message not including an information field is created when NOTHING is selected.

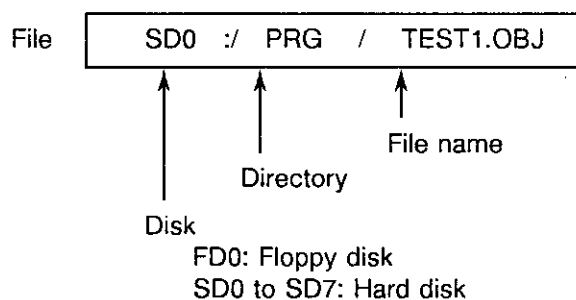
### 3.3.7 Object program save/load

An object program for simulation execution is created when a compile is completed. If the object program does not exist, no program can be executed.

Create the object program after the compile is completed or load an object file from a disk. The object program for simulation execution that can be created simultaneously is only one. The previous object program is erased when a new object program is created. To leave the object program, save it in a disk (object file).

The procedure for saving or loading the object program is described below.

- ① Press the  key to display the popup menu. Select "SIMULATOR PORT A" using the  and  keys and press the return key or space key. If "SIMULATOR PORT A" has already been loaded, press the  and function keys at the same time and select "Simulator Port A" again.
- ② Press the space key (or return key) with the cursor put in the  position. The save/load screen is then displayed.
- ③ Move the cursor to the Data Type position and press the space key (or return key). Select "Object Program" using the  and  keys and press the space key (or return key).
- ④ Move the cursor to the Operation position and press the space key (or return key). Move the cursor to the desired "Save" or "Load" position and press the space key.
- ⑤ Move the cursor to the File position and enter the file name to be saved or loaded. The format in the position where the file name is entered is shown below.



- ⑥ Move the cursor to the Execute position and press the space key (or return key). The object program is then saved or loaded.



D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL

3.3 Simple Program Creation

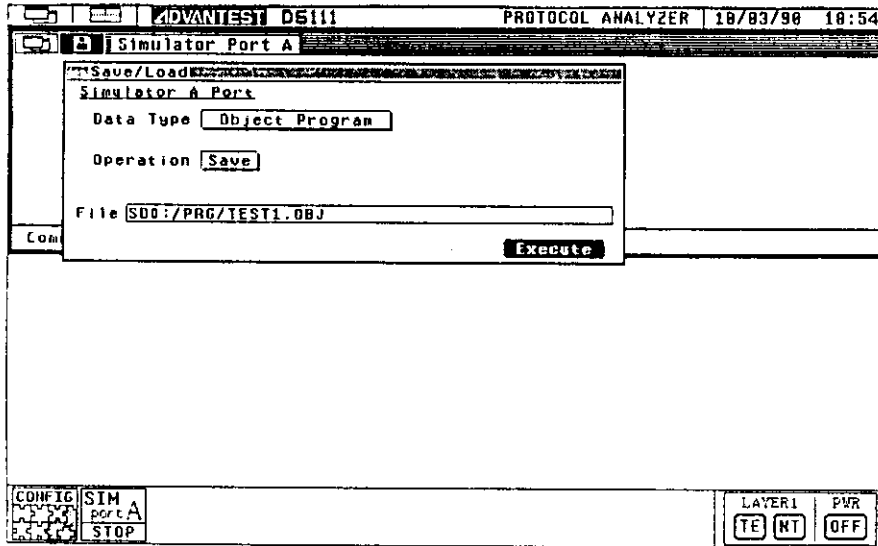


Figure 3-10 Object program saving

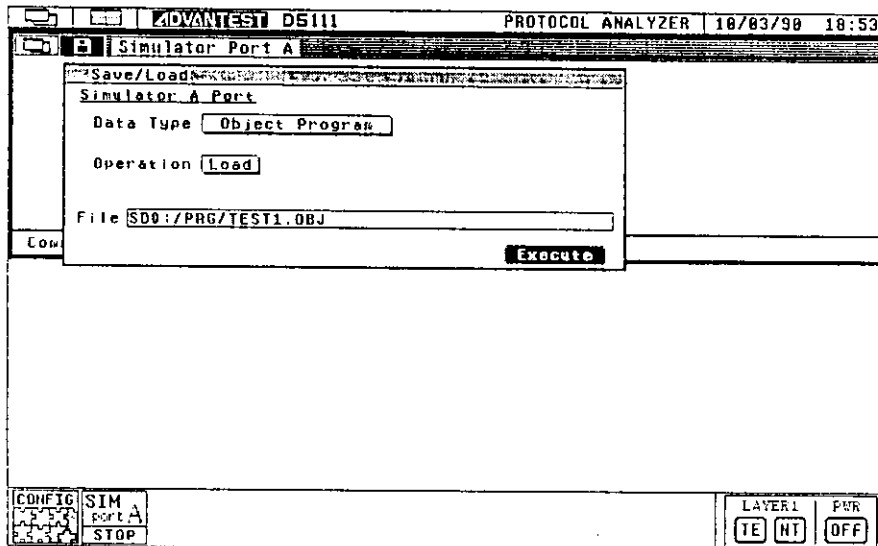


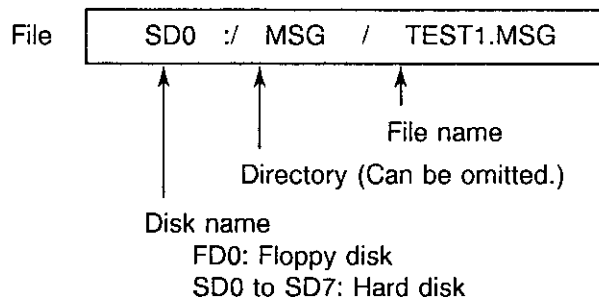
Figure 3-11 Object program loading

### 3.3.8 Message data save/load

To execute a saved program, load the object program or compile the source program, as described previously. Message data is required when a program of which message data is used is executed. The message data can be prepared by creating data again using a message builder or loading message data from a disk.

The procedure for saving or loading the message data is shown below.

- ① Press the  key to display the popup menu. Select "SIMULATOR PORT A" using the  and  keys and press the return key or space key. If "SIMULATOR PORT A" has already been loaded, press the  and function keys at the same time and select "Simulator Port A" again.
- ② Press the space key (or return key) with the cursor put in the  position.  
The save/load screen is then displayed.
- ③ Move the cursor to the Data Type position and press the space key (or return key). Select "Message Data" using the  and  keys and press the space key (or return key).
- ④ Move the cursor to the Operation position and press the space key. Move the cursor to the "Save" position when you wish to save, and move it to the "Load" position when you wish to load. Then, press the space key.
- ⑤ Move the cursor to the File position and enter the file name to be saved or loaded. The format in the position where the file name is entered is shown below.



- ⑥ Move the cursor to the Execute position and press the space key (or return key). The message data is then saved or loaded.

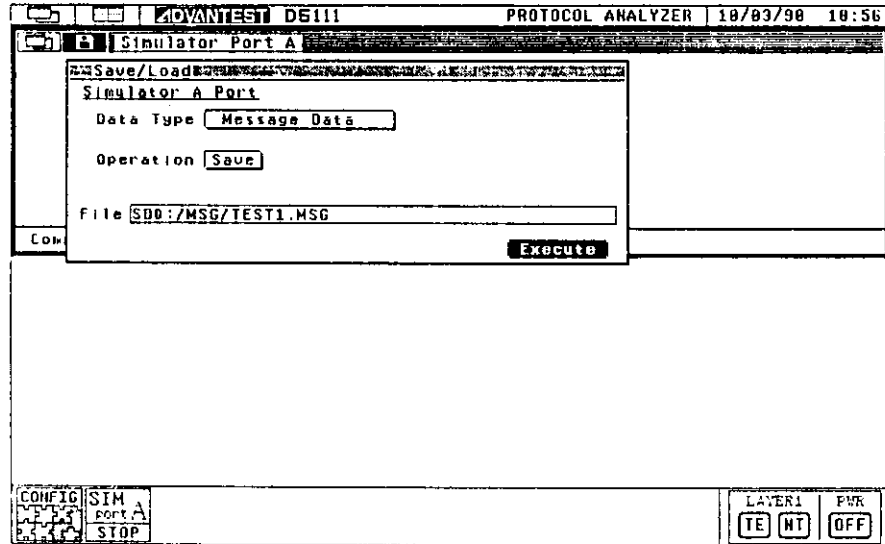


Figure 3-12 Message data saving

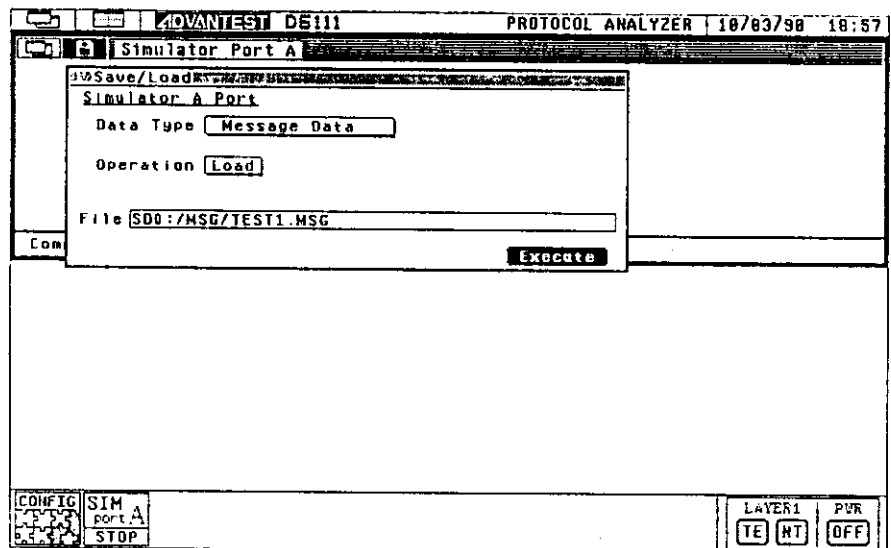


Figure 3-13 Message data loading

### 3.4 Execution of Simulation Function without Program

The simulation functions that can be used in this section are shown in Table 3-3.

[Operation procedure]







- ① Select the simulation port A screen.
- ② Move the cursor to the  position using the  and  keys and press the space key (or return key).
- ③ Select "SIMULATOR" using the  and  keys and press the space key (or return key).
- ④ Press the  key to move the cursor to the command column.
- ⑤ Enter the function name of a simulation you wish to execute.

Table 3-3 Simulation functions

	Argument 1	Argument 2	
① INCVS	-	-	Increments the V (S) value.
② INCVR	-	-	Increments the V (R) value.
③ SIMMODE	TE	-	Set to TE mode.
	NT	-	Set to NT mode.
④ PFEED	NORM	-	Set to normal feeding.
	RVS	-	Set to reverse feeding.
	OFF	-	Set to feeding OFF state.
⑤ SENDF	Name	0/1	Transmits the frame.
	Frame number	0/1	Transmits the frame.
⑥ SETVS	0 to 127	-	Sets the V (S) value.
⑦ SETVR	0 to 127	-	Sets the V (R) value.
⑧ PH_ACT	-	-	Activates layer 1.
⑨ PH_DEACT	-	-	Stops layer 1.
⑩ SIMSTATUS	-	-	Displays the setting state in steps ③ and ④.
⑪ PH_DEFAULT	-	-	Stops layer 1 and cancels the setting state in steps ③ and ④.

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**3.4 Execution of Simulation Function without Program**

---

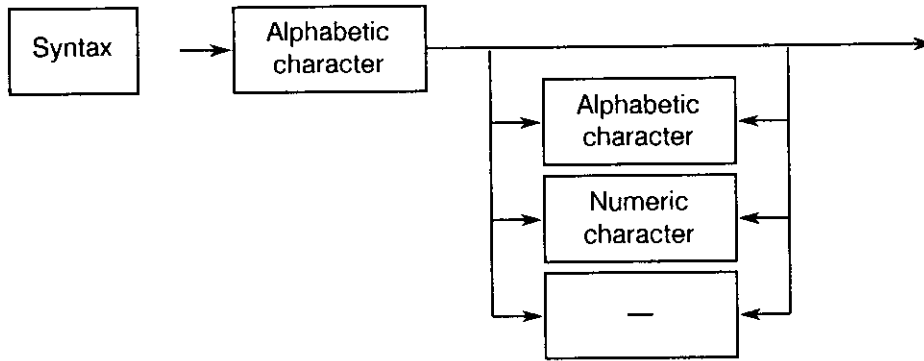
[Example] The procedure for transmitting the SABME frame registered in a message builder with the D5111B used as NT simulation is described below.

[Operation procedure]

- ① SIMMODE\_NT  (Set to NT mode.)
- ② PFEED\_RVS  (Set to reverse feeding.)
- ③ PH\_ACT  (Activates layer 1.)
- ④ SENDF\_SABME\_1  (Transfers the SABME frame of which P/F bit is "1.")
- ⑤ PH\_DEFAULT  (Returns to the previous state in which a simulation function is executed.)

### 3.5 Syntax of Simulation Language (PSL51)

(1) Variable

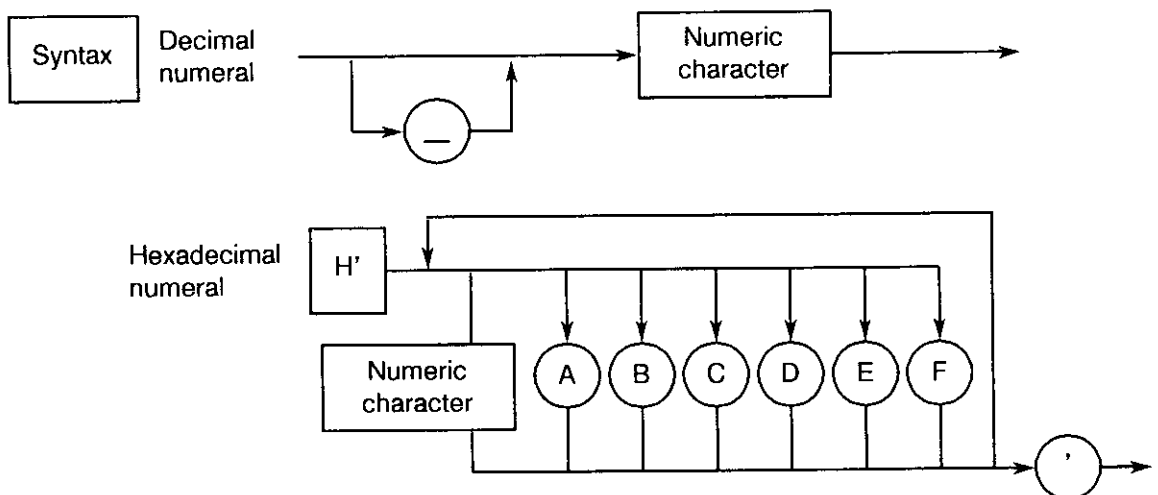


**Description** The variable consists of an alphabetic character, numeric character, and underline ('\_'). Up to ten characters can be recognized.

**Example** ABC  
X1234  
Z\_XY  
A123456789X (The same as A123456789)

\* The whole variable is defined in the suitable variable at entire area.

(2) Constant



**Description** The constant is classified into decimal and hexadecimal constants. It has a value of -2147483648 to 2147483647 (signed 32 bits). The hexadecimal numeral is represented in the form of H'F34' (3892 in decimals).

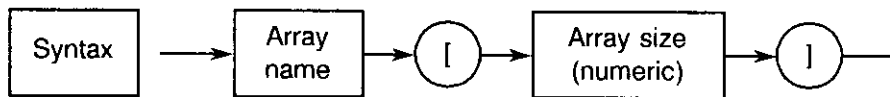
**Example**    ABC = 32767  
              X = H'F6D'

**Caution**

- The PSL51 data type is signed 32 bits. Therefore, the decimal numeral exceeding 2147483648 (H'80000000' in hexadecimals) indicates a negative value.

Decimal numeral	Hexadecimal numeral	Actual value
2147483648	80000000	2147483648
2147483649	80000001	2147483647
2147483650	80000002	2147483646
⋮	⋮	⋮
4294967294	FFFFFFFE	-2
4294967295	FFFFFFF	-1

(3) **Array**



**Description**    Only a one-dimensional array can be used. The array has a value of -2147483648 to 2147483647 (signed 32 bits) as in the variable. The array can also be used when it is declared at the beginning of a program. (For more information, see 3.2, "PSL51 Programming Structure.")

Declare the array at the beginning of a program as given by the example below. Array XYZ is also initialized as shown below.

```

XYZ [0] = 1
XYZ [1] = 2
XYZ [2] = 3
XYZ [3] = 4
XYZ [4] = XYZ [5] = XYZ [6] = 5
XYZ [7] = XYZ [8] = XYZ [9] = 0
  
```

**Example** ARRAY A [10], B [20], C [4]  
ARRAY XYZ [10] = [1, 2, 3, 4, 5 [3], 0 [3]]

Caution

- The subscript of an array begins with "0."
- To initialize an array, initialize all elements.
- The size of the whole array must be less than 32 K bytes in all total including a variable. The size exceeding 32 K bytes may cause malfunction.

(4) Operator

**Description** The operator has an addition (+), subtraction (-), multiplication (\*), and division (/).

**Example** AB1 = X1 (I) \*6 - XYZ + a / ab12

Caution

- In the division, the result is rounded down so that it is an integer (-2147483647 to 2147483647).  
(Example) 123 / 12 = 10  
The value becomes (-1) when a divisor is "0."

(5) Relational operator

**Description** The relational operator has six elements (>, <, >=, <=, ==, and !=).

	Relational operator	Function
(1)	>	Greater than
(2)	<	Less than
(3)	>=	Greater than or equal to
(4)	<=	Less than or equal to
(5)	==	Equal to
(6)	!=	Not equal to

**Example** IF A == B THEN C = 3 \* A ELSE C = A END  
IF X != Y THEN Z = 5 / W ELSE Z = W END



(6) Logical operator

**Description** The logical operator has two elements (NOT and (!)). The value when a logical operator is used with AB1 as variable is shown in the table below.

Value of variable AB1	Value of !AB1
0	1
Values except "0"	0

**Example** IF (!AB1) THEN a = 1 ELSE a = 0 END

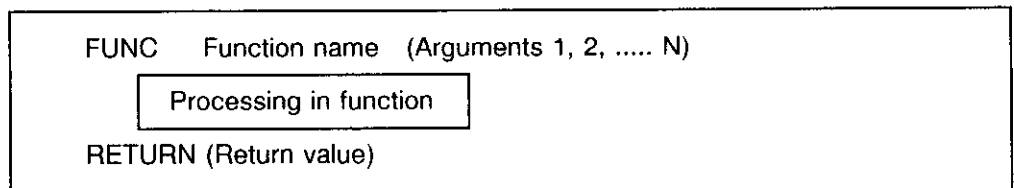
(7) Logical operator per bit

**Description** The logical operator per bit has two elements (& (AND per bit) and (OR per bit)).

**Example**  
A = 5  
B = 6  
C = A & B (Value C is 4.)  
D = A | B (Value D is 7.)

(8) Function

**Description** The program consists of an aggregate called multiple functions. The function structure must be as shown below.



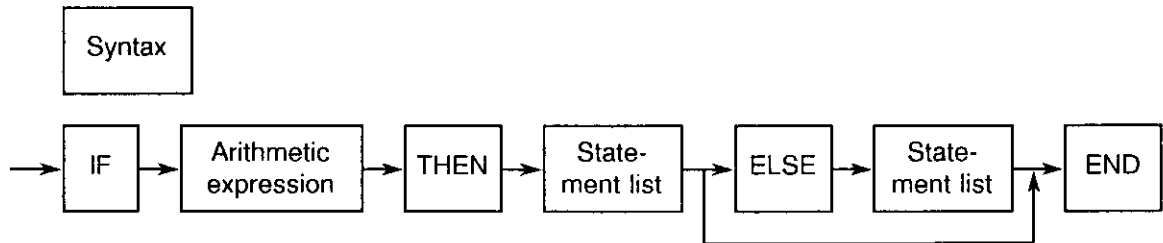
The program is also executed from a MAIN ( ) function. The MAIN ( ) function must thus be created. The argument or return value can be omitted.

**Example**

```

FUNC  MAIN ( )
    A = 123
    B = 256
    C = SUB1 (A, B)
RETURN
FUNC  SUB1 (ARG1, ARG2)
    VAL = ARG1 + 3 x ARG2
RETURN (VAL)
    
```

(9) IF statement



Description If the condition written in an arithmetic expression is true (except "0"), the statement just after THEN is executed. If it is false and there is ELSE, the statement just after ELSE is executed. If there is not ELSE, the statement following END is executed.

Example

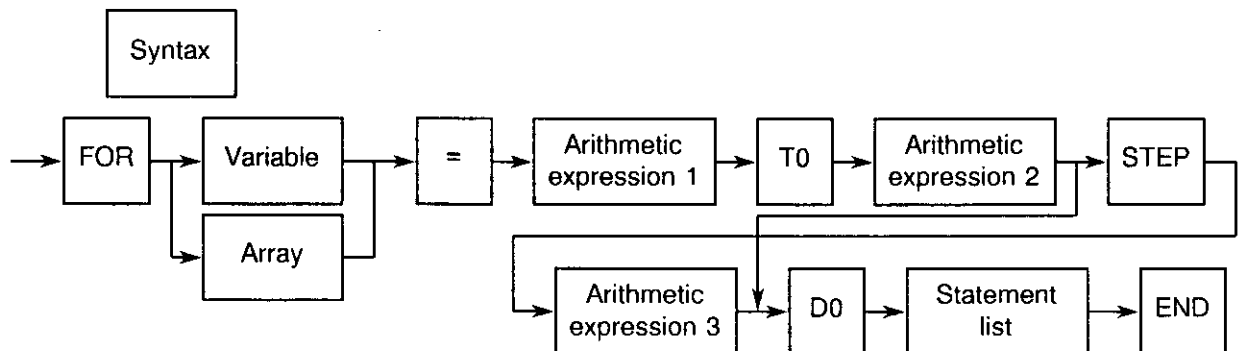
```

IF (! check (arg1, arg2)) THEN ok = 0
                               ELSE ok = 1
END
  
```

```

FUNC check (X, Y)
  IF MODE = = 0 THEN ret = X + Y
                    ELSE ret = X - Y
  END
RETURN (ret)
  
```

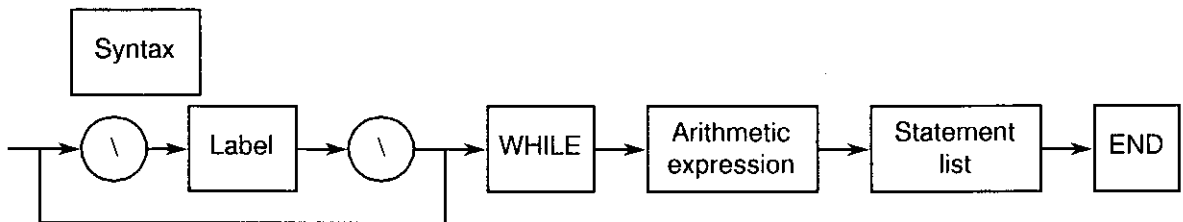
(10) FOR statement



Description The statement list enclosed by DO through END is repeatedly executed in accordance with the conditional expression following a FOR statement. In the conditional expression following a FOR statement, the initial value of arithmetic expression 1 is assigned to a variable or array, and the variable or array is added proportionally to the value of arithmetic expression 3 each time the list of DO through END statements is executed. The list of DO through END statements is repeatedly executed until the variable or array equals arithmetic expression 2. "STEP arithmetic expression 3" can be omitted. If it is omitted, the increment is automatically set to "1."

Example	<pre>FOR I = 0 TO 13 STEP 1 DO   A[I] = 1 + 2   PRINT ("A[%D] = %D\n", I, A[I]) END</pre>
---------	---

(11) WHILE statement



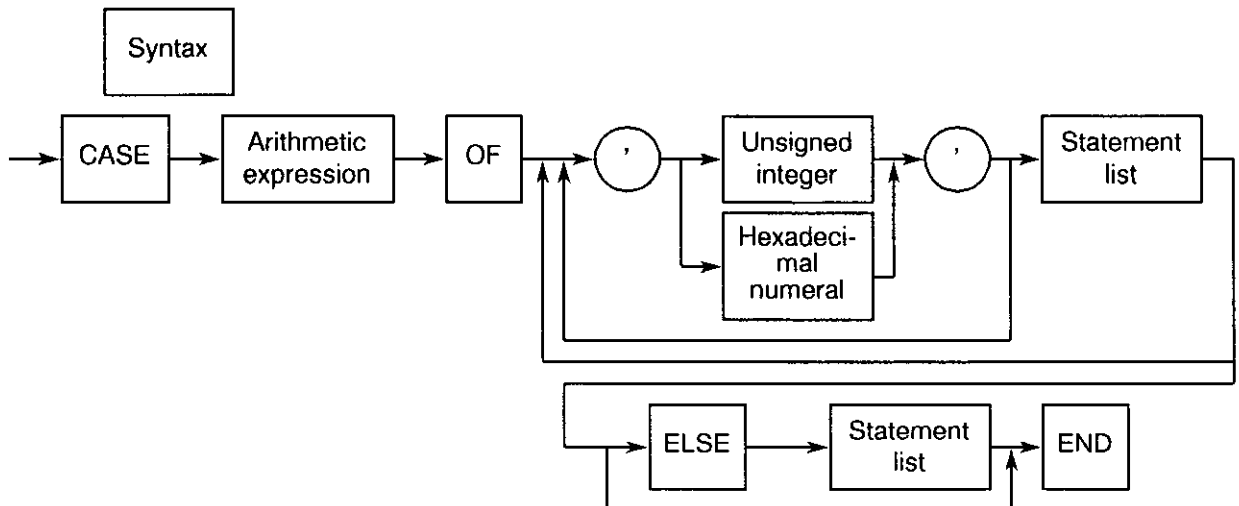
Description	The statement list enclosed by END is repeatedly executed from the arithmetic expression while an arithmetic expression is true (except "0").
-------------	---

Example	<pre>I = 10 WHILE I   AB[I] = xyz + I   I = I - 1   PRINT ("AB[%D] = %D\n", I, AB[I]) END</pre>
---------	---

Caution

- To exit the loop of a WHILE statement, use the EXIT statement and label in common. For more details, see step (14), "EXIT Statement."

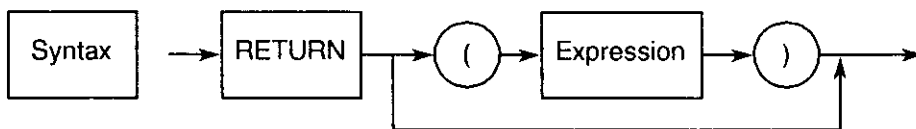
(12) CASE statement



Description	<p>In the example below, the values of variable xyz are executed respectively.</p> <p>For 0, 1, or 3, function procl (0)</p> <p>For 2 or 4, function procl (1)</p> <p>For 5, function procl (2)</p> <p>For except the above, function procl (3)</p>
-------------	---

Example	<pre> CASE xyz OF '0' '1' '3' procl (0) '2' '4'   procl (1) '5'      procl (2) ELSE     procl (3) END         </pre>
---------	--

(13) RETURN statement

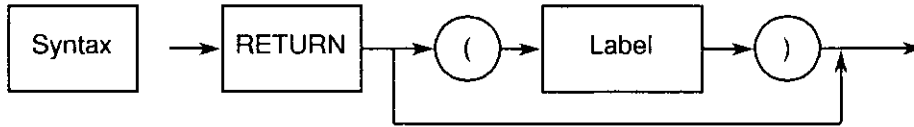


Description	<p>The RETURN statement is used to return a control to the calling function when function execution is completed. The use of the RETURN statement is shown in the table below.</p>
-------------	--

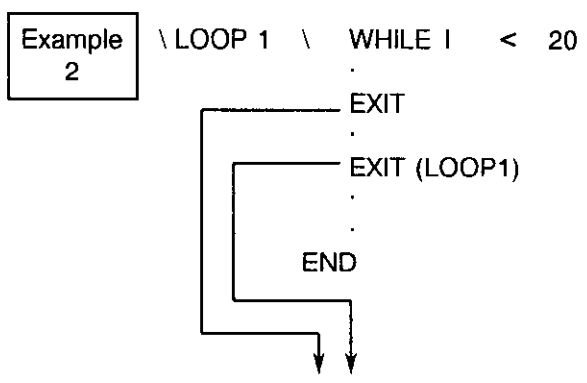
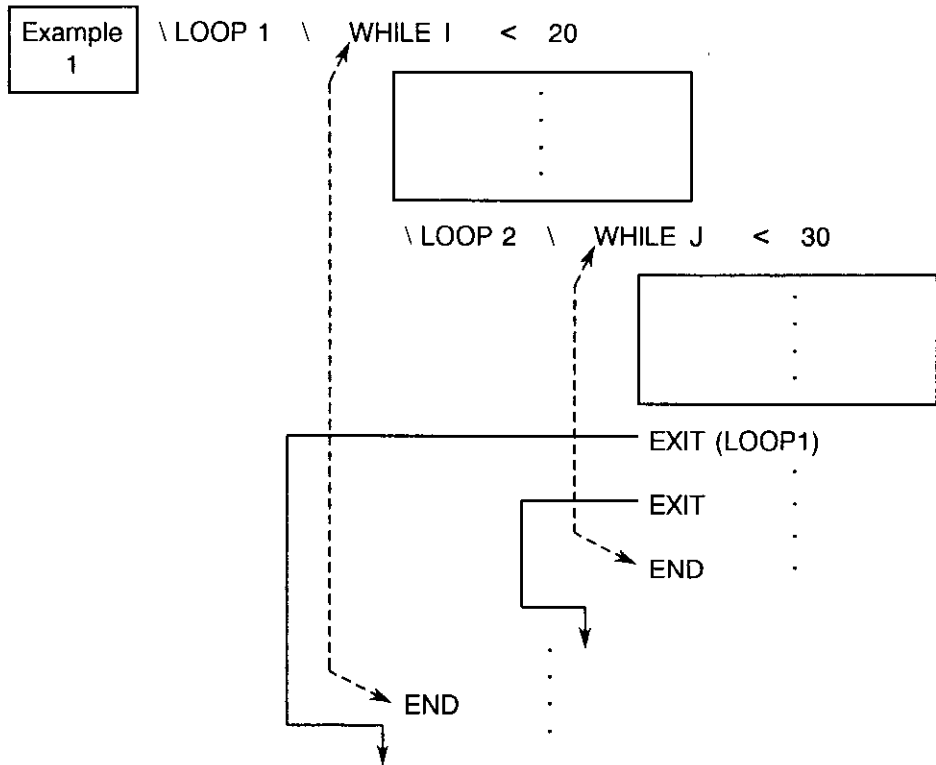
Function execution is complete (for return value available)	RETURN (Return value)
Function execution is complete (for no return value)	RETURN

Example	<pre> FUNC func1 (arg1, arg2) ... RETURN (xyz + 2) /* Function completion (for return value available) */ FUNC func1 (arg1, arg2) ... RETURN          /* Function completion (for not return value available)*/         </pre>
---------	--

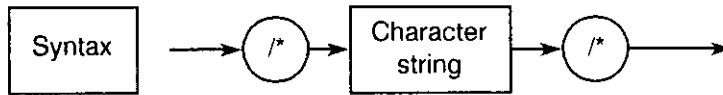
(14) EXIT statement



**Description** The EXIT statement is used to forcibly terminate the loop of an innermost WHILE statement executed currently and exit the loop. If a label is specified, exit the WHILE loop that is constituted by a WHILE statement having the label.



(15) Comment statement



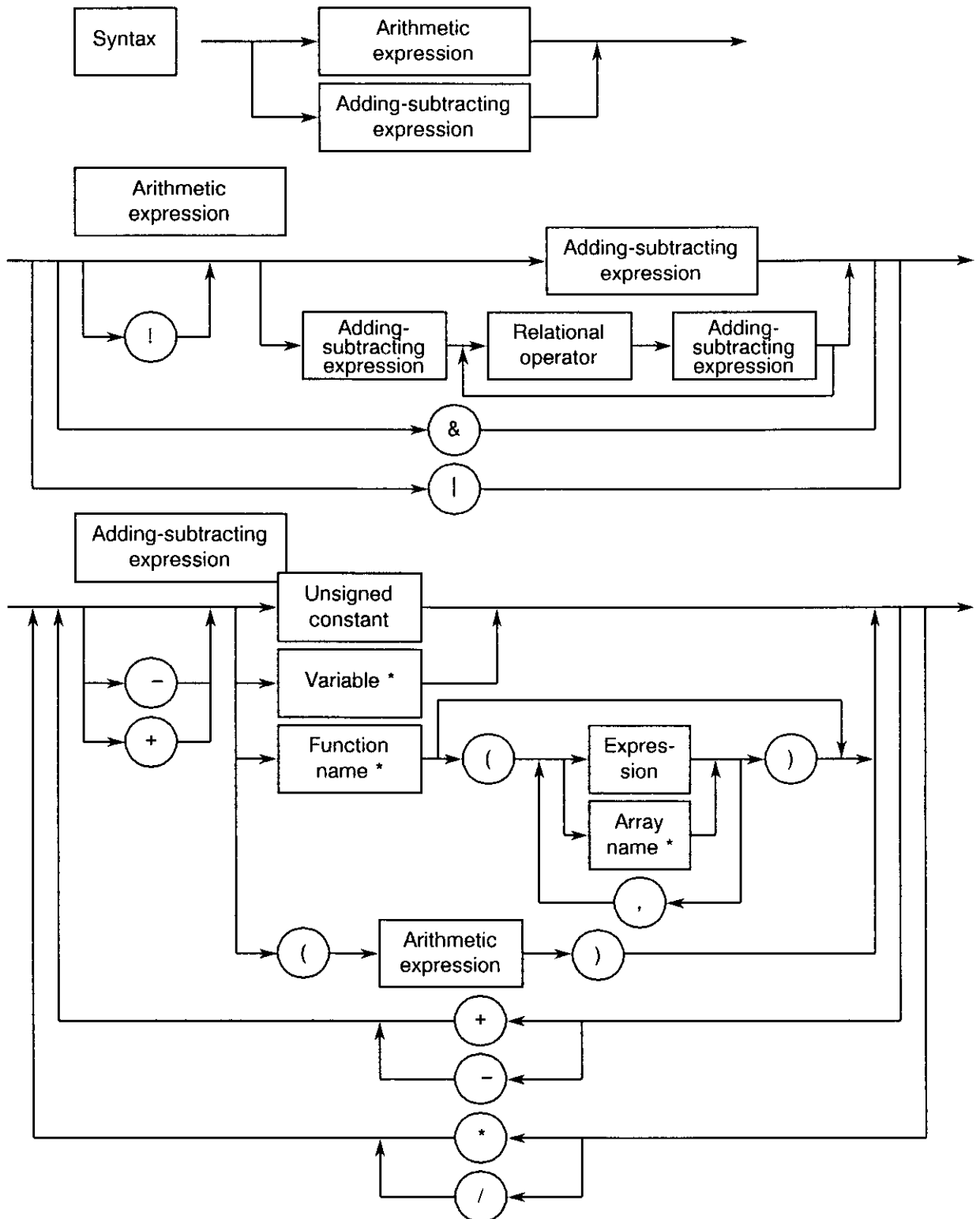
**Description** The character string between /\* and /\* in a program is interpreted as a comment and not compiled.

**Example 1**

```
/*.....*/
This is COMMENT example !!!
/*.....*/

IF ab == xy /* same value ? */
  THEN PRINT (" sb == xy / n")
  ELSE PRINT ("sb != xy / n")
END
```

(16) Expression



\* See the corresponding caution.

Caution

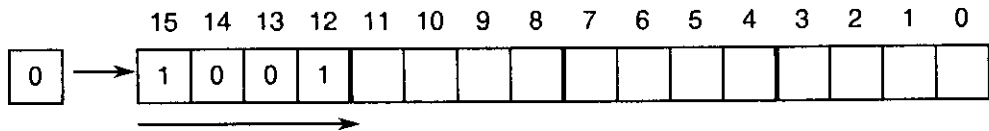
The words below are used as a reserved word, so cannot be used as a variable, function name, and array name.

ARRAY	IF	SIMMODE
CASE	OF	PFEED
DO	RETURN	
ELSE	STEP	
END	THEN	
EXIT	TO	
FOR	WHILE	
FUNC	LAYER	

(17) R\_SHIFT (val, times)

Function

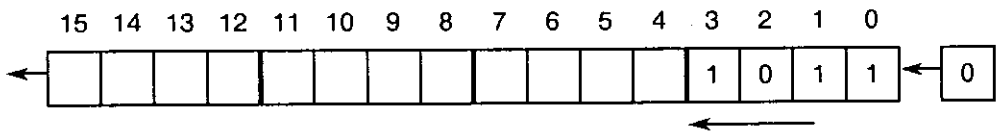
Returns the value when the value of argument val is shifted to the right as a function value proportionally to the number of times specified by argument times. The logical shift is executed as shown below.



(18) L\_SHIFT (val, times)

Function

Returns the value when the value of argument val is shifted to the left as a function value proportionally to the number of times specified by argument times. The logical shift is executed as shown below.



Example

```
IF data >= 20
THEN data = R_SHIFT (data, 4)
ELSE data = L_SHIFT (data, 4)
```



(19) PRINT ("format", arg1, arg2, ...)

Function
----------

Displays data on the screen in accordance with the format specified by an argument. If there are no arguments other than "format", the character string enclosed by " " is directly displayed.

A variable is assigned to arg1 and arg2. The display indication conforms to the character code beginning with % in a format. The display format is roughly classified into the following:

- ① % [ + ] [ 0 ] [ n ] d: Decimal display conversion
- + : Displayed at the beginning if the numeric is positive.
  - 0 : Padded at the beginning of a field.
  - n : Specifies the length of a field that stores a replaced character by the number of digits.
  - d : Converts in a decimal numeral.

The specification enclosed by brackets [ ] can be omitted.

Example    %D    → 123  
             %5D   → 123  
             %05D → 00123  
             % + 5D → + 123

- ② % [ 0 ] [ n ] x: Hexadecimal display conversion
- 0 : Padded at the beginning of a field.
  - n : Specifies the length of a field that stores a replaced character by the number of digits.
  - x : Converts in a hexadecimal numeral.

The specification enclosed by brackets [ ] can be omitted.

Example    %x    → 123  
             %5x   → 123  
             %05x → 00123  
             % + 5x → + 123

- ③ % [ 0 ] [ n ] u
- 0 : Padded at the beginning of a field.
  - n : Specifies the length of a field that stores a replaced character by the number of digits.
  - u : Converts in an unsigned decimal numeral.

The specification enclosed by brackets [ ] can be omitted.

Example    %u    → 123  
             %5u   → 123  
             %05u → 00123  
             % + 5u → + 123

- ④ % [ n ] b
- n : Specifies the length of a field that stores a replaced character by the number of digits.
  - b : Display the blank.

Example    %5b   →

\* A line feed is performed when "\n" or "\N" is specified in the format.

Example	<pre>A = 123 B = -71 C = 51 PRINT ("SAMPLE") PRINT ("PROGRAM" !!\N") PRINT ("A = %D(%3x) ",A,A) PRINT ("B = %5D\N",B) PRINT ("C = %U B = %05D\N",C,B)</pre>
---------	---

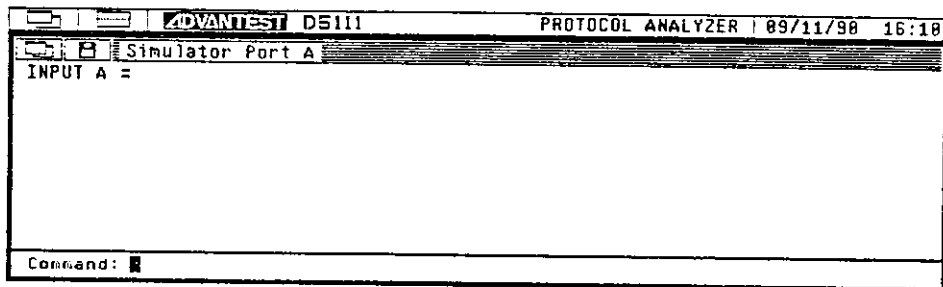
Specification for line feed

Result	<pre>SAMPLE PROGRAM !! A = 123( 7B) B = -71 C = 51 B = -0071</pre>
--------	--

(20) INPUT ( )

Function	Returns the keyed-in value as a function value. The key-in operation can be executed only when the cursor is in the "Command" position (see the figure below).
----------	--

The key-in operation is classified as a decimal-code entry and hexadecimal-code entry. For more information, see the description of a constant.



Result	<pre>PRINT ("INPUT A =") A = INPUT() PRINT ("%D\N",A)</pre>
--------	---

### 3.6 Common Function

Table 3-4 shows the common function.

Table 3-4 Common table

(1) INSERT	Changes any octet value of a send frame.
(2) RECEIVE	Enters the frame reception wait mode.
(3) T_START	Activates a timer.
(4) T_STOP	Stops a timer.
(5) EXTRACT	Extracts any octet value of a receive frame.
(6) RXFRLN	Returns the data length of a receive frame as a function value.
(7) WAIT	Stops a program for a specified period of time.
(8) PH_ACT	Activates layer 1.
(9) PH_DEACT	Stops layer 1.

Each function is explained on the next or later pages.

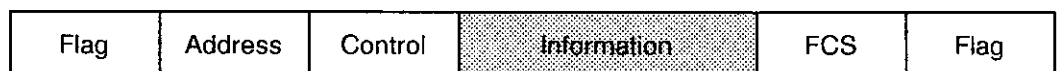
(1) INSERT

Calling format	INSERT("NAME", n, DT)
Argument description	<p>"NAME" : Name of a frame that changes data</p> <p>n : Octet value that changes data</p> <p>DT : Value of data change (in the range of 0 to 255)</p>
Function value	<p>0 : Normal termination</p> <p>- 60 : Argument error</p> <p>- 61 : Frame name error</p>
Example of use	INSERT("SABME", 2, 81)
Function description	Changes any octet data of the frame name specified by an argument.
Caution	The changed value is valid until simulation stops. The data area that can be changed in transparent and layer-2 automatic modes differs each other (see the figure below). The diagonally striped-portion area in the figure below can be changed. The change position is an octet that is sequentially counted from "1" on the left of the diagonally striped portion.
Item to be limited	If there is no frame with frame name specified by an argument, a frame name error occurs. If you try to change the place including no data, an argument error occurs.

① Transparent mode



② Layer-2 automatic mode



(2) RECEIVE

Calling format RECEIVE(TIMER)

Argument description TIMER : Timer number of a timer to be used

Function value 0 : Termination by time-out  
1 : Termination by frame reception

Example of use RECEIVE(202)  
T201 = 201  
RECEIVE(T201)

Function description When this RECEIVE function is executed, a program is interrupted until a frame is received or a time-out is performed. Data is transferred to the read buffer when this function is terminated by frame reception. A receive frame read function then enters the execution enable state.

Caution <<Details of frame reception>>

A receive frame is transferred to the receive data buffer when it is received by D5111B. The receive data buffer consists of 32 ring buffers.

To read the content of the receive frame, the receive data in the receive data buffer must be sent to the read buffer using the RECEIVE function. If there is no receive data in the receive data buffer, the program is interrupted until a frame is received. However, if the timer with timer number specified by an argument is timed out at that time, this function is terminated after returning the time-out function value without transferring data to the read buffer.

The RECEIVE function is also used to release one receive data buffer that is put into the enable state during frame reception. The RECEIVE function must be executed proportionally to the number of frames to be received. If the RECEIVE function is not executed when many frames are received, some receive frames may not be received or the TEI management procedure may not be performed in layer-2 automatic mode.

(3) T\_START

Calling format	T_START(TIMER, SEC)
Argument description	TIMER : Timer number of a timer to be activated SEC : Time-out value (in one-second units)
Function value	0 : Normal termination - 60 : Argument error - 120 : Exceeding maximum number of timers to be activated simultaneously
Example of use	T_START(201, 1) T202 = 202 S = 2 T_START (T202, S)
Function description	Activates the timer with timer number specified by an argument. If a timer with same timer number has already been activated when this function is executed, the timer stops to activate the new timer.
Item to be limited	If a value exceeding 65535 is set to the time-out value, an argument error occurs. If timers exceeding the maximum number of timers to be activated simultaneously are activated, an overtimer error occurs. (Up to five timers can be activated at the same time.)

(4) T\_STOP

Calling format	T_STOP(TIMER)
Argument description	TIMER : Timer number of a timer that stops
Function value	0 : Normal termination - 121 : No timer is activated.
Example of use	T_STOP(201) T202 = 202 T_STOP(T202)
Function description	Stops the timer with timer number specified by an argument.
Item to be limited	If the timer with timer number specified by an argument is not activated, an error code is returned as the function value. Nothing else is done.

(5) EXTRACT

Calling format	EXTRACT(n)
Argument description	n : Octet value of data to be extracted The value of <1 to receive frame length> can be set.
Function value	0 to 255 : Value of data to be extracted - 60 : Extraction error - 80 : Frame non-receive error
Example of use	CF1 = EXTRACT (3) AA = 4 CF2 = EXTRACT(AA)
Function description	Extracts any octet value of a frame to be received.
Caution	The data area that is extracted in transparent and layer-2 automatic modes differs each other (see the figure below). The diagonally-striped portion area in the figure below can be extracted. The extraction position is an octet that is sequentially counted from "1" on the left of the diagonally striped portion.
Item to be limited	The octet data set by an argument is returned as the function value. The argument value of 1 to receive frame length can be set. If values other than the above are set, an argument error occurs. If there is no receive frame in the read buffer when this function is executed, a frame non-receive error occurs. The frame is loaded in the read buffer using a RECEIVE function.

① Transparent mode



② Layer-2 automatic mode





(6) RXFRLN

Calling format                      RXFRLN( )

Argument description

Function value                      0 to 65535 : Data length of receive frame  
   - 80        : Frame non-receive error

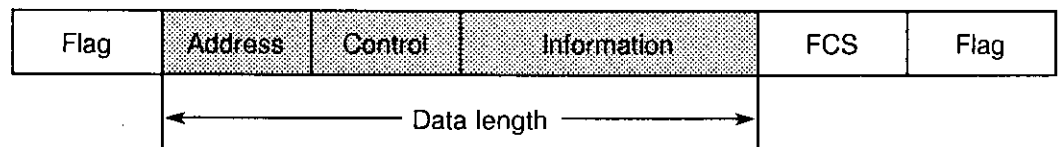
Example of use                      LENGTH = RXFRLN( )

Function description                Returns the data length of a receive frame as the function value.

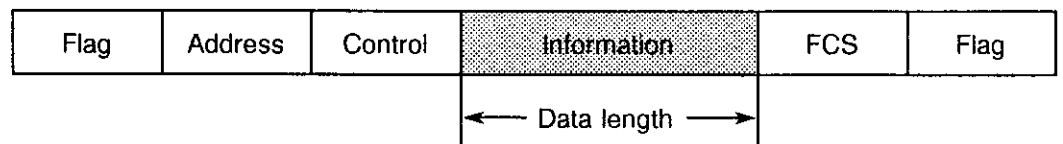
Caution                              The data length that is defined in transparent and layer-2 automatic modes differs each other (see the figure below).

Item to be limited                  If there is no receive data in the read buffer, a frame non-receive error occurs. For more information, see the description of a RECEIVE function.

① Transparent mode



② Layer-2 automatic mode



(7) WAIT

Calling format	WAIT(SEC)
Argument description	SEC : Wait time The value of 0 to 65535 can be set in 100 msec units.
Function value	0 : Normal termination - 60 : Argument error
Example of use	WAIT(10) SEC = 5 WAIT(SEC)
Function description	Stops the program execution for a specified period of time.

(8) PH\_ACT

Calling format PH\_ACT( )

Argument description

Function value 0 : Normal termination  
-160 : Layer-1 error

Example of use PH\_ACT( )

Function description Activates layer 1.

Caution If layer 1 cannot be activated, the function value indicating a layer-1 error is returned.

(9) PH\_DEACT

Calling format PH\_DEACT( )

Argument description

Function value 0 : Normal termination  
- 53 : NT mode error

Example of use PH\_DEACT( )

Function description Stops layer 1.

Item to be limited The PH\_DEACT function cannot be executed in modes other than NT.

### 3.7 Function for Transparent Mode

Table 3-5 shows the function list for the transparent mode.

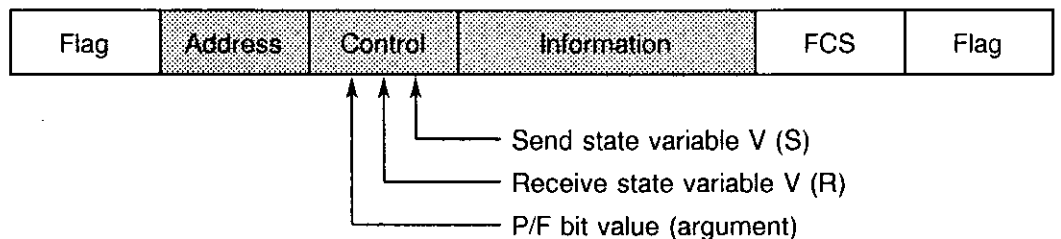
Table 3-5 Function for transparent mode

Frame transmission related function	
(1) SENDF	Transmits a frame.
(2) INCVS	Increments send state variable V (S) by +1.
(3) INCVR	Increments receive state variable V (R) by +1.
(4) SETVS	Sets the value of send state variable V (S).
(5) SETVR	Sets the value of receive state variable V (R).
Frame reception related function	
(6) RXSAPI	Returns the SAPI value of a receive frame as a function value.
(7) RXTEI	Returns the TEI value of a receive frame as a function value.
(8) RXCR	Returns the C/R bit value of a receive frame as a function value.
(9) RXNR	Returns the N(R) value of a receive frame as a function value.
(10) RXNS	Returns the N(S) value of a receive frame as a function value.
(11) RXPF	Returns the P/F bit value of a receive frame as a function value.
(12) RXTYPE	Returns the frame type of a receive frame as a function value.
(13) RXCF1	Returns the first octet in the control field of a receive frame as a function value.
(14) RXCF2	Returns the second octet in the control field of a receive frame as a function value.
(15) RXFRCF1	Returns the first octet in the information field of a receive FRMR frame as a function value.
(16) RXFRCF2	Returns the second octet in the information field of a receive FRMR frame as a function value.
(17) RXFRVS	Returns the V(S) value in the information field of a receive FRMR frame as a function value.
(18) RXFRVR	Returns the V(R) value in the information field of a receive FRMR frame as a function value.
(19) RXFRCR	Returns the C/R bit value in the information field of a receive FRMR frame as a function value.
(20) RXFRWXYZ	Returns the WXYZ bit value in the information field of a receive FRMR frame as a function value.

Each function is described on the next or later pages.

(1) SENDF

Calling format	SENDF("NAME", PF)
Argument description	"NAME" : Frame name to be transferred PF : P/F bit value of a frame to be transferred
Function value	0 : Normal termination - 50 : Transparent mode error - 61 : Frame number error - 110 : Transmit frame length error
Example of use	SENDF("ID_REQ", 0) PF = 1 SENDF("SABME", 1)
Function description	Sends a frame in transparent mode. The frame with frame name specified by an argument is sent with the argument-specified P/F bit value, transmission sequence number N (S), and reception sequence number N (R) added. Send state variable V (S) can be changed using an INCVS OR SETVS function. Receive state variable V (R) can also be changed using an INCVR or SETVR function. The data that is added during frame transmission using this SENDF function is shown in the figure below.
Item to be limited	The SENDF function cannot be executed in layer-2 automatic mode.



(2) INCVS

Calling format                      INCVS(    )

Argument description

Function value                      0            : Normal termination  
   - 50        : Transparent mode error

Example of use                      INCVS(    )

Function description                Increments send state variable V (S) by + 1. This value is added to the frame as an N (S) value only when a number-system information frame (I frame) is sent.

Item to be limited                    The INCVS function cannot be executed in layer-2 automatic mode.

(3) INCVR

Calling format                   INCVR(    )

Argument description

Function value                0           : Normal termination  
                              - 50       : Transparent mode error

Example of use                INCVR(    )

Function description           Increments receive state variable V (R) by + 1. This value is added to the send frame as an N (R) value when a number-system information frame (I frame) and number-system supervisory frame (S frame) are sent.

Item to be limited            The INCVR function cannot be executed in layer-2 automatic mode.



(4) SETVS

Calling format	SETVS(VS)
Argument description	VS : Send state variable V (S) The value of 1 to 127 can be set.
Function value	0 : Normal termination -50 : Transparent mode error -60 : Argument error
Example of use	SETVS(5) VS = 6 SETVS(VS)
Function description	Sets the value of send state variable V (S). This value is added to the send frame as an N (S) value only when a number-system information frame (I frame) is sent.
Item to be limited	The SETVS function cannot be executed in layer-2 automatic mode. If values other than 0 to 127 are set to an argument, an argument error occurs.

(5) SETVR

Calling format	SETVR(VR)
Argument description	VR : Receive state variable V (R)
Function value	0 : Normal termination -50 : Transparent mode error -60 : Argument error
Example of use	SETVR(5) VR = 6 SETVR(VR)
Function description	Sets the value of receive state variable V (R). This value is added to the send frame as an N (R) value only when a number-system information frame (I frame) and number-system supervisory frame (S frame) are sent.
Item to be limited	The SETVR function cannot be executed in layer-2 automatic mode. If values other than 0 to 127 are set to an argument, an argument error occurs.

(6) RXSAPI

Calling format                      RXSAPI (   )

Argument description

Function value                      0 to 63     :   SAPI value of receive frame  
   -80         :   Frame non-receive error  
   -82         :   SAPI does not exist in the receive frame.

Example of use                      SAPI = RXSAPI (   )

Function description                Returns the SAPI value of a receive frame loaded in the read buffer as a function value.

Item to be limited                   If there is no receive frame in the read buffer, a frame non-receive error occurs. An error also occurs when SAPI does not exist in the receive frame. The receive frame is loaded in the read buffer using a RECEIVE function.

(7) RXTEI

Calling format RXTEI ( )

Argument description

Function value

0 to 127	:	TEI value of receive frame
-80	:	Frame non-receive error
-82	:	TEI does not exist in the receive frame.

Example of use TEI = RXTEI ( )

Function description Returns the TEI value of a receive frame loaded in the read buffer as a function value.

Item to be limited If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when TEI does not exist in the receive data of the read buffer. The receive frame is loaded in the read buffer using a RECEIVE function.

(8) RXCR

Calling format RXCR ( )

Argument description

Function value

0, 1	:	C/R bit value of receive frame
-50	:	Transparent mode error
-80	:	Frame non-receive error
-82	:	No C/R bit exists in the receive frame.

Example of use CR = RXCR ( )

Function description Returns the C/R bit value of a receive frame loaded in the read buffer as a function value.

Item to be limited If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when no C/R bit exists in the receive data of the read buffer. The receive frame is loaded in the read buffer using a RECEIVE function. The RXCR function cannot be executed in layer-2 automatic mode.

(9) RXNR

Calling format RXNR ( )

Argument description

Function value

0 to 127	:	N (R) value of receive frame
-50	:	Transparent mode error
-80	:	Frame non-receive error
-82	:	N (R) does not exist in the receive frame.

Example of use NR = RXNR ( )

Function description Returns the N (R) value of a receive frame loaded in the read buffer as a function value.

Item to be limited If there is no receive frame in the read buffer, a frame non-receive error occurs. An error also occurs when N (R) does not exist in the receive frame. The receive frame is loaded in the read buffer using a RECEIVE function. The RXNR function cannot be executed in layer-2 automatic mode.

(10) RXNS

Calling format RXNS ( )

Argument description

Function value

0 to 127	:	N (S) value of receive frame
-50	:	Transparent mode error
-80	:	Frame non-receive error
-82	:	N (S) does not exist in the receive frame.

Example of use NS = RXNS ( )

Function description Obtains the N (S) value of a receive frame loaded in the read buffer as a function value.

Item to be limited If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when N (S) does not exist in the receive data of the read buffer. The receive frame is loaded in the read buffer using a RECEIVE function. The RXNS function cannot be executed in layer-2 automatic mode.

(11) RXPF

Calling format                      RXPF (    )

Argument description

Function value                      0, 1        : P/F bit value of receive frame  
   -50        : Transparent mode error  
   -80        : Frame non-receive error  
   -82        : No P/F bit exits in the receive frame.

Example of use                      PF = RXPF (    )

Function description                Returns the P/F bit value of a receive frame loaded in the read buffer as a function value.

Item to be limited                   If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when no P/F bit exits in the receive data of the read buffer. The receive frame is loaded in the read buffer using a RECEIVE function. The RXPF function cannot be executed in layer-2 automatic mode.





(13) RXCF1

Calling format	RXCF1 ( )
Argument description	
Function value	0 to 255 : First octet value of receive frame's control field -50 : Transparent mode error -80 : Frame non-receive error -82 : No control field exists in the receive frame.
Example of use	CF1 = RXCF1
Function description	Returns the first octet value of a receive frame's control field in the read buffer as a function value.
Item to be limited	If there is no receive frame in the read buffer, a frame non-receive error occurs. An error also occurs when no control field exists in the receive frame. The receive frame is loaded in the read buffer using a RECEIVE function. The RXCF1 function cannot be executed in layer-2 automatic mode.

(14) RXCF2

Calling format	RXCF2 ( )
Argument description	
Function value	0 to 255 : Second octet value of receive frame's control field -50 : Transparent mode error -80 : Frame non-receive error -82 : The second octet of a control field does not exist in the receive frame.
Example of use	CF2 = RXCF2 ( )
Function description	Returns the second octet value of a receive frame's control field in the read buffer as a function value.
Item to be limited	If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when the second octet of a control field does not exist in the receive data of the read buffer. The receive frame is loaded in the read buffer using a RECEIVE function. The RXCF2 function cannot be executed in layer-2 automatic mode.

(15) RXFRFC1

Calling format RXFRFC1 ( )

Argument description

Function value

0 to 255	:	First octet value of receive FRMR frame information field
-50	:	Transparent mode error
-80	:	Frame non-receive error
-81	:	The receive frame is not an FRMR frame.
-82	:	The first octet of an information field does not exist in the receive frame.

Example of use FRFC1 = RXFRFC1 ( )

Function description Returns the first octet value of a receive FRMR frame information field in the read buffer.

Item to be limited If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when the receive frame in the read buffer is not an FRMR frame or the first octet of the information field does not exist. The receive frame is loaded in the read buffer using a RECEIVE function. The RXFRFC1 function cannot be executed in layer-2 automatic mode.

(16) RXFRCF2

Calling format RXFRCF2 ( )

Argument description

Function value

0 to 255	:	Second octet value of receive FRMR frame information field
-50	:	Transparent mode error
-80	:	Frame non-receive error
-81	:	The receive frame is not an FRMR frame.
-82	:	The second octet of an information field does not exist in the receive frame.

Example of use FRCF2 = RXFRCF2 ( )

Function description Returns the second octet value of a receive FRMR frame information field in the read buffer.

Item to be limited If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when the receive frame in the read buffer is not an FRMR frame or the second octet of the information field does not exist. The receive frame is loaded in the read buffer using a RECEIVE function. The RXFRCF2 function cannot be executed in layer-2 automatic mode.

(17) RXFRVS

Calling format RXFRVS ( )

Argument description

Function value

0 to 127	:	V (S) value of receive FRMR frame information field
-50	:	Transparent mode error
-80	:	Frame non-receive error
-81	:	The receive frame is not an FRMR frame.
-82	:	V (S) does not exist in the receive FRMR frame information field.

Example of use FRVS = RXFRVS ( )

Function description Returns the V (S) value of a receive FRMR frame information field in the read buffer.

Item to be limited If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when the receive frame in the read buffer is not an FRMR frame or V (S) does not exist in the information field. The receive frame is loaded in the read buffer using a RECEIVE function. The RXFRVS function cannot be executed in layer-2 automatic mode.

(18) RXFRVR

Calling format RXFRVR ( )

Argument description

Function value

0 to 127	:	V (R) value of receive FRMR frame information field
-50	:	Transparent mode error
-80	:	Frame non-receive error
-81	:	The receive frame is not an FRMR frame.
-82	:	V (R) does not exist in the receive FRMR frame information field.

Example of use FRVR = RXFRVR ( )

Function description Returns the V (R) value of a receive FRMR frame information field in the read buffer.

Item to be limited If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when the receive frame in the read buffer is not an FRMR frame or V (R) does not exist in the information field. The receive frame is loaded in the read buffer using a RECEIVE function. The RXFRVR function cannot be executed in layer-2 automatic mode.

(19) RXFCR

Calling format	RXFCR ( )
Argument description	
Function value	0, 1 : C/R bit value of receive FRMR frame information field -50 : Transparent mode error -80 : Frame non-receive error -81 : The receive frame is not an FRMR frame. -82 : No C/R bit exists in the receive FRMR frame information field.
Example of use	FCR = RXFCR ( )
Function description	Returns the C/R bit value of a receive FRMR frame information field in the read buffer.
Item to be limited	If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when the receive frame in the read buffer is not an FRMR frame or no C/R bit exists in the information field. The receive frame is loaded in the read buffer using a RECEIVE function. The RXFCR function cannot be executed in layer-2 automatic mode.



(20) RXFRWXYZ

Calling format                      RXFRWXYZ (   )

Argument description

Function value                      0 to 15     :    WXYZ bit value of receive FRMR frame information field  
    -50         :    Transparent mode error  
    -80         :    Frame non-receive error  
    -81         :    The receive frame is not an FRMR frame.  
    -82         :    No WXYZ bit exists in the receive FRMR frame information field.

Example of use                      FRWXYZ = RXFRWXYZ (   )

Function description               Returns the WXYZ bit value of a receive FRMR frame information field in the read buffer. The WXYZ bit is configured as shown in the table below.

Item to be limited                 If there is no receive data in the read buffer, a frame non-receive error occurs. An error also occurs when the receive frame in the read buffer is not an FRMR frame or no WXYZ bit exists in the information field. The receive frame is loaded in the read buffer using a RECEIVE function. The RXFRWXYZ function cannot be executed in layer-2 automatic mode.

Bit	7	6	5	4	3	2	1	0
Assignment	0	0	0	0	Z	Y	X	W

### 3.8 Function for Layer-2 Automatic Mode

Table 3-6 shows the function list for the layer-2 automatic mode.

Table 3-6 Function for layer-2 automatic mode (1/2)

Frame transmission related function	
(1) SENDUI	Sends a UI frame.
(2) SENDXIDC	Sends an XID command.
(3) SENDXIDR	Sends an XID response.
(4) SENDI	Sends an I frame.
Frame reception related function	
(5) RXSAPI	Returns the SAPI value of a receive frame as a function value.
(6) RXTEI	Returns the TEI value of a receive frame as a function value.
(7) RXTYPE	Returns the frame type of a receive frame as a function value.
(8) RXPD	Returns the protocol identifier of a receive frame as a function value.
(9) RXCRL	Returns the call number length of a receive frame as a function value.
(10) RXCRF	Returns the call number flag of a receive frame as a function value.
(11) RXCRV	Returns the call number value of a receive frame as a function value.
(12) RXMSG	Returns the message type of a receive frame as a function value.
(13) RXINFO_NUM	Returns the information element count (JT-Q931) of a receive frame as a function value.
(14) RXINFO_ELM	Returns the information element identifier (JT-Q931) of a receive frame as a function value.
(15) RXINFO_LEN	Returns the information element content length (JT-Q931) of a receive frame as a function value.
(16) RXINFO_VAL	Returns the information content (JT-Q931) of a receive frame as a function value.
TEI management procedure related function	
(17) REQ_TEI	Activates the TEI assignment procedure. (TE mode)
(18) CHKREQ_TEI	Performs the TEI check procedure. (NT mode)
(19) REMOVE_TEI	Performs the TEI release procedure. (NT mode)
(20) VERIFY_TEI	Performs the TEI verify procedure. (TE mode)

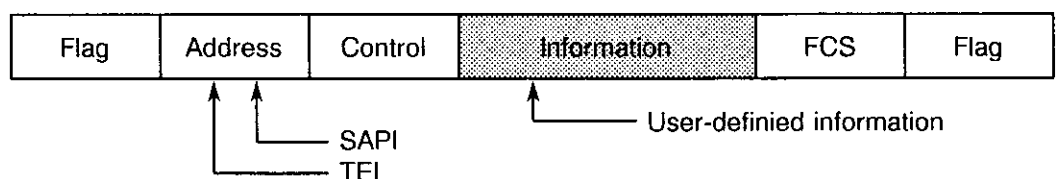
Table 3-6 Function for layer-2 automatic mode (2/2)

Link related function	
(21) LINKON	Sets a link
(22) LINKOFF	Releases a link.
(23) L_STATUS	Confirms that the line is set.
(24) SET_BUSY	Sets the source station to the busy state.
(25) REL_BUSY	Releases the busy state of the source station.
(26) PROHIBIT_L	Prohibits the setting of a new link.
(27) PERMIT_L	Releases the link setting non-permission state.
SAPI and TEI management related function	
(28) REG_TEI	Registers the TEI value in D5111B.
(29) REL_TEI	Releases the TEI value from D5111B.
(30) NEXT_TEI	Declares that a new TEI value is used. (TE mode)
(31) ACT_SAPI	Sets and changes the SAPI value.
(32) ACT_TEI	Sets and changes the TEI value.
(33) LOCK_SAPI	Prohibits the setting and change of an SAPI value.
(34) LOCK_TEI	Prohibits the setting and change of a TEI value.
(35) FLEX_SAPI	Releases the unchangeable state of an SAPI value.
(36) FLEX_TEI	Releases the unchangeable state of a TEI value.

Each function is described on the next or later pages.

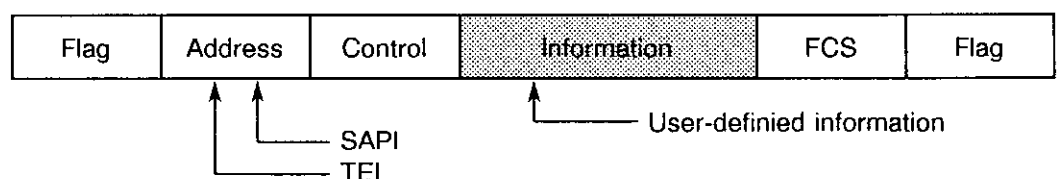
(1) SENDUI

Calling format	SENDUI("NAME")
Argument description	"NAME" : Name of a frame to be sent
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -61 : Frame name error -70 : SAPI error -71 : TEI error -110 : Send frame length error
Example of use	SENDUI("IDCHK")
Function description	Sends a UI frame in layer-2 automatic mode. The UI frame with frame name specified by an argument with the preset SAPI and TEI values used in an address (see the figure below).
Caution	The initial value of the SAPI value is 0, and the initial value of the TEI value 127. The SAPI value is set using an ACT_SAPI function, and the TEI value using an ACT_TEI function. The TEI value is also set when the TEI assignment procedure or link setting is performed. The SAPI value is also set when the link setting is performed. Setting of new SAPI and TEI values can be prohibited using LOCK_SAPI and LOCK_TEI functions, respectively.
Item to be limited	If this function is executed when values other than 0, 16, and 63 are set to the SAPI value, an SAPI error occurs. If this function is executed when an unregistered TEI value is set to the TEI value, a TEI error occurs. The TEI value can be registered using an REG_TEI function. If there is no frame with frame name specified by an argument or the frame length of a send frame is short, an error occurs. The SENDUI function cannot be executed in transparent mode.



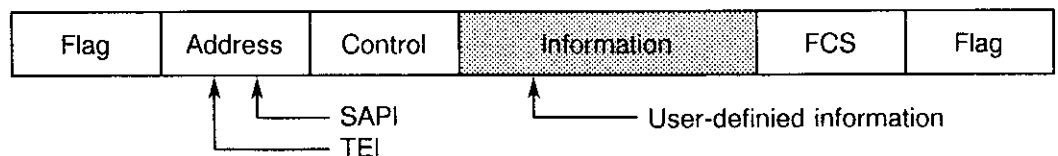
(2) SENDXIDC

Calling format	SENDXIDC("NAME")
Argument description	"NAME" : Name of a frame to be sent
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -61 : Frame name error -70 : SAPI error -71 : TEI error -110 : Send frame length error
Example of use	SENDXIDC("XID1")
Function description	Sends an XID command in layer-2 automatic mode. The XID command with frame name specified by an argument is sent with the preset SAPI and TEI values used in an address (see the figure below).
Caution	The initial value of the SAPI value is 0, and the initial value of the TEI value 127. The SAPI value is set using an ACT_SAPI function, and the TEI value using an ACT_TEI function. The TEI value is also set when the TEI assignment procedure or link setting is performed. The SAPI value is also set when the link setting is performed. Setting of new SAPI and TEI values can be prohibited using LOCK_SAPI and LOCK_TEI functions, respectively.
Item to be limited	If this function is executed when values other than 0, 16, and 63 are set to the SAPI value, an SAPI error occurs. If this function is executed when an unregistered TEI value is set to the TEI value, a TEI error occurs. The TEI value can be registered using an REG_TEI function. If there is no frame with frame name specified by an argument or the frame length of a send frame is short, an error occurs. The SENDXIDC function cannot be executed in transparent mode.



(3) SENDXIDR

Calling format	SENDXIDR("NAME")
Argument description	"NAME" : Name of a frame to be sent
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -61 : Frame name error -70 : SAPI error -71 : TEI error -110 : Send frame length error
Example of use	SENDXIDR("XID2")
Function description	Sends an XIR response in layer-2 automatic mode. The XID response with frame name specified by an argument is sent with the preset SAPI and TEI values used in an address (see the figure below).
Caution	The initial value of the SAPI value is 0, and the initial value of the TEI value 127. The SAPI function is set using an ACT_SAPI function, and the TEI value using an ACT_TEI function. The TEI value is also set when the TEI assignment procedure or the link setting is performed. The SAPI value is also set when the link setting is performed. Setting of new SAPI and TEI values can be prohibited using LOCK_SAPI and LOCK_TEI functions, respectively.
Item to be limited	If this function is executed when values other than 0, 16, and 63 are set to the SAPI value, an SAPI error occurs. If this function is executed when an unregistered TEI value is set to the TEI value, a TEI error occurs. The TEI value can be registered using an REG_TEI function. If there is no frame with frame name specified by an argument or the frame length of a send frame is short, an error occurs. This SENDXIDR function cannot be executed in transparent mode.



(4) SENDI

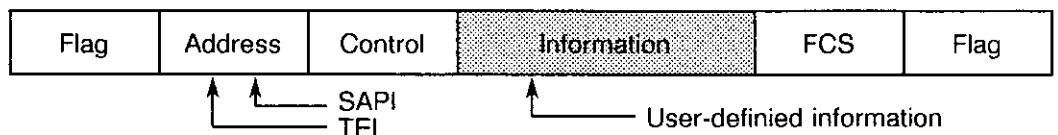
Calling format	SENDI("NAME")
Argument description	"NAME" : Name of a frame to be sent
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -61 : Frame number error -70 : SAPI error -71 : TEI error -72 : Link non-setting error -110 : Send frame length error -200 : Other errors

Example of use                      SENDI("SETUP")

Function description                Sends an I frame in layer-2 automatic mode. The I frame with frame name specified by an argument is sent with the preset SAPI and TEI values used in an address (see the figure below).

Caution                                A link must be set to send the I frame. The link is set using a LINKON function or set after accepting a link set request from the destination. If this function is executed using SAPI and TEI of which link is not set, a link non-setting error occurs. The I frame is usually sent using SAPI and TEI of which link is set most lately. SAPI and TEI that are set can be fixed using LOCK\_SAPI and LOCK\_TEI functions respectively as required.

Item to be limited                    To send the frame of a link that was set previously, set the link correctly using an ACT SAPI or ACT TEI function. If this function is executed when values other than 0, 16, and 63 are set to the SAPI value, an SAPI error occurs. If this function is executed when an unregistered TEI value is set to the TEI value, a TEI error occurs. The TEI value can be registered using an REG TEI function. If there is no frame with frame name specified by an argument or the frame length of a send frame is short, an error occurs. The SENDI function cannot be executed in transparent mode.



(5) RXSAPI

Calling format                      RXSAPI (    )

Argument description

Function value                      0 to 63     :    SAPI value of receive frame  
   -80         :    Frame non-receive error

Example of use                      SAPI = RXSAPI (    )

Function description                Returns the SAPI value of a receive frame loaded in the read buffer as a function value.

Item to be limited                  If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function.



(6) RXTEI

Calling format RXTEI ( )

Argument description

Function value 0 to 127 : TEI value of receive frame  
-80 : Frame non-receive error

Example of use TEI = RXTEI ( )

Function description Returns the TEI value of a receive frame loaded in the read buffer as a function value.

Item to be limited If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function.

(7) RXTYPE

Calling format RXTYPE ( )

Argument description

Function value 0 to 7 : Type of receive frame  
-80 : Frame non-receive error

Example of use TYPE = RXTYPE ( )

Function description Returns the type of a receive frame loaded in the read buffer as a function value.

Item to be limited The relationship between the function value and frame type is shown in the table below. If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function.

Frame type	Function value
I command	0
UI command	4
XID command	6
XID response	7

(8) RXPDP

Calling format RXPDP ( )

Argument description

Function value

0 to 255	:	Protocol identifier of receive frame
-51	:	Layer-2 automatic mode error
-80	:	Frame non-receive error
-82	:	No protocol identifier exists.

Example of use PD = RXPDP ( )

Function description Returns the protocol identifier of a receive frame loaded in the read buffer as a function value.

Item to be limited This function can be used in layer-2 automatic mode, but not in transparent mode. If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function. If the receive frame loaded in the read buffer has no protocol identifier, an error occurs.

(9) RXCRL

Calling format                      RXCRL (   )

Argument description

Function value                      0 to 15     :   Call number length of receive frame  
   -51         :   Layer-2 automatic mode error  
   -80         :   Frame non-receive error  
   -82         :   No call number length exists.

Example of use                      CRL = RXCRL (   )

Function description                Returns the call number length of a receive frame loaded  
   in the read buffer as a function value.

Item to be limited                   This function can be used in layer-2 automatic mode, but  
   not in transparent mode. If there is no receive frame in  
   the read buffer, a frame non-receive error occurs. The  
   receive frame is loaded in the read buffer using a  
   RECEIVE function. If the receive frame loaded in the  
   read buffer has no call number length, an error occurs.

(10) RXCRF

Calling format	RXCRF ( )
Argument description	
Function value	0, 1 : Call number flag value of receive frame -51 : Layer-2 automatic mode error -80 : Frame non-receive error -82 : No call number flag exists.
Example of use	CRF = RXCRF ( )
Function description	Returns the call number flag of a receive frame loaded in the read buffer as a function value.
Item to be limited	This function can be used in layer-2 automatic mode, but not in transparent mode. If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function. If the receive frame loaded in the read buffer has no call number flag, an error occurs.

(11) RXCRV

Calling format	RXCRV ( )
Argument description	
Function value	1 to 8388607 : Call number value of receive frame -51 : Layer-2 automatic mode error -80 : Frame non-receive error -82 : No call number value exists.
Example of use	CRV = RXCRV ( )
Function description	Returns the call number value of a receive frame loaded in the read buffer as a function value.
Item to be limited	This function can be used in layer-2 automatic mode, but not in transparent mode. If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function. If the receive frame loaded in the read buffer has no call number value, an error occurs.

(12) RXMSG

Calling format                      RXMSG (   )

Argument description

Function value                      0 to 255    :    Message type of receive frame  
   -51         :    Layer-2 automatic mode error  
   -80         :    Frame non-receive error  
   -82         :    No message type exists.

Example of use                      MSG = RXMSG (   )

Function description                Returns the message type of a receive frame loaded in the read buffer as a function value.

Item to be limited                   This function can be used in layer-2 automatic mode, but not in transparent mode. If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function. If the receive frame loaded in the read buffer has no message type, an error occurs.

(13) RXINFO\_NUM

Calling format                      RXINFO\_NUM ( )

Argument description

Function value                      0 to 255 : Number of information elements (JT-Q931) included in layer-3 message of receive frame  
   -51 : Layer-2 automatic mode error  
   -80 : Frame non-receive error  
   -82 : No JT-Q931 user and network call control message exists.

Example of use                      NUM = RXINFO\_NUM ( )

Function description                Returns the information element count (JT-Q931) of a receive frame loaded in the read buffer as a function value.

Item to be limited                 This function can be used in layer-2 automatic mode, but not in transparent mode. If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function. If the receive frame loaded in the read buffer has no JT-Q931 user and network call control message, an error occurs.



(14) RXINFO\_\_ELM

Calling format	RXINFO__ELM (n)
Argument description	n : Position of information element that extracts an information element identifier
Function value	0 to 225 : Information element identifier (JT-Q931) included in layer-3 message of receive frame -51 : Layer-2 automatic mode error -60 : Argument error -80 : Frame non-receive error -82 : No JT-Q931 user and network call control message exists.
Example of use	ELM1 = RXINFO__ELM(1) NUM = 2 ELM = RXINFO__ELM(NUM)
Function description	Returns the information element count identifier (JT-Q931) of a receive frame loaded in the read buffer as a function value.
Caution	The position of the information element identifier is specified by the value that is sequentially counted from "1" on the first information element identifier side.
Item to be limited	This function can be used in layer-2 automatic mode, but not in transparent mode. The information element identifier in the position specified by an argument is returned as a function value. If no information element identifier exists in the position, an argument error occurs. The argument error is caused when the value exceeding the number of information elements is specified. If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function. If the receive frame loaded in the read buffer has no JT-Q931 user and network call control message, an error occurs.

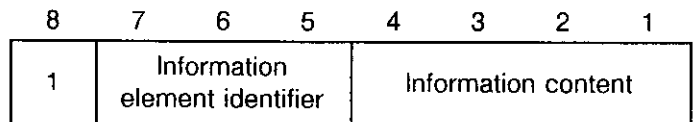
(15) RXINFO\_LEN

Calling format	RXINFO_LEN (n)
Argument description	n : Position of information element that extracts the length of an information element content
Function value	0 to 255 : Length of information element content (JT-Q931) included in layer-3 message of receive frame -51 : Layer-2 automatic mode error -60 : Argument error -80 : Frame non-receive error -82 : The content of the information indicated by JT-Q931 does not exist.
Example of use	LEN1 = RXINFO_LEN(1) NUM = 2 LEN = RXINFO_LEN(NUM)
Function description	Returns the information element content length (JT-Q931) of a receive frame loaded in the read buffer as a function value (see the figure below).
Caution	The position of the information element is specified by the value that is sequentially counted from "1" on the first information element side.

Item to be limited

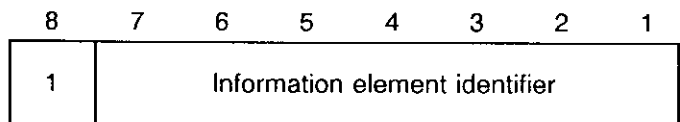
This function can be used in layer-2 automatic mode, but not in transparent mode. The length of the information element content with information element specified by an argument is returned as a function value. If no information element exists in the position specified by the argument or no information content exists even if an information element exists, an argument error occurs. The argument error is caused when the value exceeding the number of information elements is specified as an argument (see the figure below). If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function. If the receive frame loaded in the read buffer has no JT-Q931 user and network call control message, an error occurs.

a) Format of single fixed-length information element (type 1)



RXINFO\_LEN (n) = 0

b) Format of single fixed-length information element (type 2)



RXINFO\_LEN (n) = -82 (Error)

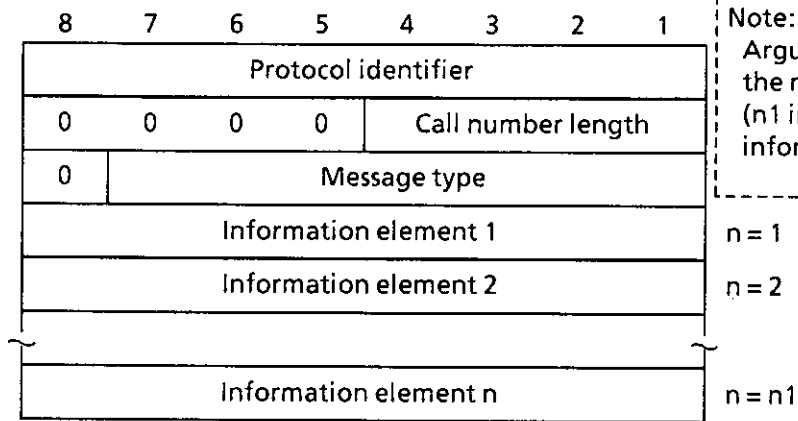
c) Variable-length information element

RXINFO\_LEN (n) = Length of information  
 element content

(16) RXINFO\_VAL

Calling format	RXINFO VAL (n, m)
Argument description	n : Position of information element that extracts an information element content m : Octet value of data that is extracted from the information content
Function value	0 to 255 : Information content (JT-Q931) included in layer-3 message of receive frame -51 : Layer-2 automatic mode error -60 : Argument error -80 : Frame non-receive error -82 : The content of the information indicated by JT-Q931 does not exist.
Example of use	VAL = RXINFO_VAL(1, 1) N = 1 M = 2 VAL2 = RXINFO_VAL(N, M)
Function description	Returns the information content (JT-Q931) of a receive frame loaded in the read buffer as a function value.
Item to be limited	This function can be used in layer-2 automatic mode, but not in transparent mode. The information content in the position specified by an argument is returned as a function value. The argument is set as shown in the figure below. If the argument is set in ways other than shown below, an argument error occurs. If there is no receive frame in the read buffer, a frame non-receive error occurs. The receive frame is loaded in the read buffer using a RECEIVE function. If the receive frame loaded in the read buffer has no JT-Q931 user and network call control message, an error occurs.

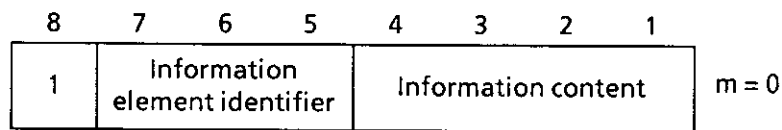
① Specification of argument n in RXINF VAL (n, m) function



Note:  
Argument n is specified in the range of  $1 \leq n \leq n1$ . (n1 indicates the number of information elements.)

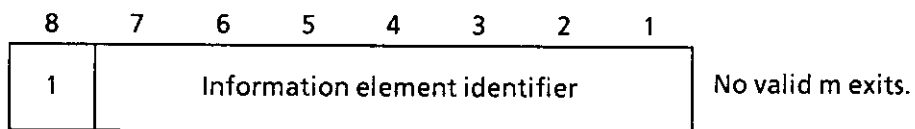
② Specification of argument m in RXINFO\_\_ VAL (n, m) function

(a) Format of single fixed-length information element (type 1)



RXINFO\_LEN (n) = 0  
RXINFO\_\_VAL (n, 0) = "Information content"

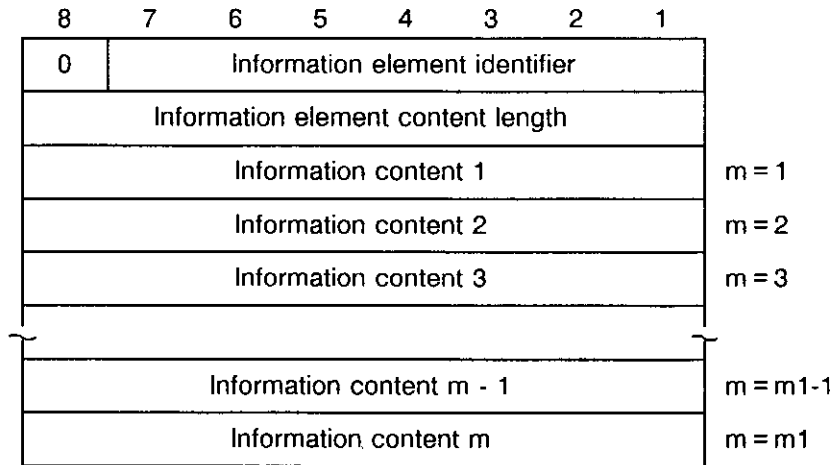
(b) Format of single fixed-length information element (type 2)



RXINFO\_LEN (n) = -82 (Argument error)  
RXINFO\_\_VAL (n, m) = -82 (Argument error)

Note: An RXINFO LEN or RXINFO VAL function cannot be executed in an information element of this type.

(c) Format of variable-length information element



Note: Argument  $m$  is specified in the range of  $1 \leq m \leq m1$ .  
 ( $m1$  indicates the length of an information element content.)

(17) REQ\_\_TEI

Calling format                      REQ\_\_TEI (    )

Argument description

Function value                      0 to 126    :    TEI value to be assigned  
   -51            :    Layer-2 automatic mode error  
   -52            :    TE mode error  
   -90            :    TEI assignment error

Example of use                      TEI=REQ\_\_TEI (    )

Function description                Activates the TEI assignment procedure in layer-2 automatic mode. This function is used when D5111B is in TE mode. The assigned TEI value is returned as a function value when the TEI assignment is terminated normally. The TEI value that is assigned at that time is also registered.

Item to be limited                    The REQ TEI function cannot be used in transparent and NT modes.

(18) CHKREQ\_TEI

Calling format	CHKREQ_TEI(TEI)
Argument description	TEI : TEI value for ID check
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -53 : NT mode error -60 : Argument error
Example of use	CHKREQ_TEI (64) TEI = 65 CHKREQ_TEI (TEI)
Function description	Performs the TEI check procedure in layer-2 automatic mode. This function is used when D5111B is in NT mode. A TEI check procedure is performed for the TEI value specified by an argument. If the argument is "127", the TEI check procedure for all TEI values is performed.
Item to be limited	If values other than 0 through 127 are specified as an argument, an argument error occurs. The CHKREQ TEI function cannot be executed in a transparent or TE mode.



(19) REMOVE\_\_TEI

Calling format	REMOVE__TEI(TEI)
Argument description	TEI : TEI value to be released
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -53 : NT mode error -60 : Argument error
Example of use	REMOVE__TEI(64) TEI = 64 REMOVE__TEI(TEI)
Function description	Performs the TEI release procedure in layer-2 automatic mode. This function is used when D5111B is in NT mode. When this function is executed, an ID release message of which TEI value specified by an argument is added in an AI field is continuously sent two times.
Item to be limited	If values other than 0 through 127 are specified as an argument, an argument error occurs. The REMOVE TEI function cannot be executed in a transparent or TE mode.

(20) VERIFY\_\_TEI

Calling format                      VERIFY\_\_TEI (    )

Argument description

Function value                      0            :   Normal termination  
                                     -51         :   Layer-2 automatic mode error  
                                     -52         :   TE mode error  
                                     -71         :   TEI error

Example of use                      VERIFY\_\_TEI (    )

Function description                Activates the TEI verification procedure in layer-2 automatic mode. This function is used when D5111B is in TE mode. An ID verification request message is sent when this function is executed. The preset TEI value is then added in AI field.

Item to be limited                   If the preset TEI value is other than 0 through 126, a TEI error occurs. Setting of the TEI value can be changed using an ACT\_\_TEI function. The VERIFY\_\_TEI function cannot be executed in a transparent or NT mode.

(21) LINKON

Calling format LINKON ( )

Argument description

Function value

0	:	Normal termination
-51	:	Layer-2 automatic mode error
-70	:	SAPI error
-71	:	TEI error
-100	:	Link setting rejection
-101	:	Exceeding the maximum number of simultaneous links

Example of use LINKON ( )

Function description

Sets a link in layer-2 automatic mode.  
Preset SAPI and TEI are usually used for link setting.  
If D5111B is in TE mode and the function execution is a first one after simulation, a TEI assignment procedure is performed before link setting and the TEI value assigned at that time is used for the link setting.  
However, if an REQ\_\_TEI function is executed before the LINKON function is executed, no TEI assignment procedure is performed even if the execution is a first execution after simulation, and the preset TEI value is used for the link setting.  
If a NEXT\_\_TEI function is executed before the LINKON function is executed, a TEI assignment procedure is performed even if the execution is not a first one after simulation, and the TEI value set at that time is used for the link setting.  
The example above is valid when D5111B is in TE mode.  
Preset SAPI and TEI values are used when D5111B is in NT mode.  
Setting of the SAPI value is changed using an ACT\_\_SAPI function. Setting of a new SAPI value can be inhibited using a LOCK\_\_SAPI function. An SAPI value is changed to the setting of the corresponding SAPI value when D5111B receives a link setting request from the destination.

However, if a LOCK\_\_SAPI function is executed before D5111B receives a link setting request from the destination, setting of the SAPI value is not changed.

An ACT\_\_TEI function is used for TEI value setting. Setting of a new TEI value can be inhibited using a LOCK\_\_TEI function. The TEI value is set when an REQ\_\_TEI function as well as the ACT\_\_TEI function is executed or link setting is performed. It is also set when D5111B assigns the TEI value in NT mode. If a LOCK\_\_TEI function is executed before these events, setting of the TEI value is not changed by a link setting request from the destination.

Item to be limited

If this function is executed when the SAPI value that is set is other than 0, 16, and 63, an SAPI error occurs. If an unregistered TEI value is used in the TEI value that is set, a TEI error occurs. The TEI value is registered using an REG\_\_TEI function. This D5111B can set a maximum of eight links at the same time. If nine or more links are set at the same time, a maximum link count error occurs.

(22) LINKOFF

Calling format LINKOFF ( )

Argument description

Function value           0           : Normal termination  
                         -51           : Layer-2 automatic mode error  
                         -102          : Link non-setting error

Example of use LINKOFF ( )

Function description Releases a link in layer-2 automatic mode. The link with SAPI and TEI values that are set in advance is released using this function.

Item to be limited If a link that is expected to be released is not set, a link non-setting error occurs. The SAPI value is set using an ACT\_SAPI function, and the TEI value using an ACT\_TEI function. The LINKOFF function cannot be executed in transparent mode.

(23) L\_\_STATUS

Calling format	L__STATUS(SAPI,TEI)
Argument description	SAPI : SAPI value that inspects the link status TEI : TEI value that inspects the link status
Function value	0 : A link has not been set. 1 : A link has already been set. -51 : Layer-2 automatic mode error
Example of use	L__STATUS(0,64) SAPI = 16 TEI = 65 L__STATUS(SAPI,TEI)
Function description	Inspects whether the link with SAPI and TEI values specified by an argument is set in layer-2 automatic mode. The result is returned by a function value. If "1" is returned as a function value, it indicates that the link has already been set. If "0" is returned as a function value, it indicates that the link has not been set.
Item to be limited	The L__STATUS function cannot be executed in transparent mode.

(24) SET\_BUSY

Calling format                      SET\_BUSY ( )

Argument description

Function value                      0            : Normal termination  
                                     -51         : Layer-2 automatic mode error  
                                     -70         : SAPI error  
                                     -71         : TEI error  
                                     -72         : Link error

Example of use                      SET\_BUSY ( )

Function description                Sets the source station to the busy state in layer-2 automatic mode. Preset SAPI and TEI values are used for the busy state. If the TEI value is "127", all links with SAPI value that is set in advance are set to the busy state.

Item to be limited                   If this function is executed when the SAPI value that is set is other than 0, 16, and 63, an SAPI error occurs. If an unregistered TEI value is used in the TEI value that is set, a TEI error occurs.  
The TEI value is registered using an REG\_TEI function. If this function is executed for the SAPI and TEI values of which link is not set, a link error occurs except when the TEI value is "127." The SET\_BUSY function cannot be executed in transparent mode.

(25) REL\_\_BUSY

Calling format REL\_\_BUSY ( )

Argument description

Function value

0	:	Normal termination
-51	:	Layer-2 automatic mode error
-70	:	SAPI error
-71	:	TEI error
-72	:	Link error

Example of use REL\_\_BUSY ( )

Function description Releases the busy state of the source station in layer-2 automatic mode. The busy state is released for the SAPI and TEI values that are set in advance. If the TEI value that is set is "127", the busy state of all links with SAPI value that is set are released using this function.

Item to be limited If this function is executed when the SAPI value that is set is other than 0, 16, and 63, an SAPI error occurs. If an unregistered TEI value is used in the TEI value that is set, a TEI error occurs.

The TEI value is registered using an REG\_\_TEI function. If this function is executed for the SAPI and TEI values of which link is not set, a link error occurs except when the TEI value is "127." The REL BUSY function cannot be executed in transparent mode.



(26) PROHIBIT\_L

Calling format                      PROHIBIT\_L ( )

Argument description

Function value                      0            : Normal termination  
                                      -51         : Layer-2 automatic mode error  
                                      - 70        : SAPI error

Example of use                      PROHIBIT\_L ( )

Function description                Prohibits the setting of a new link with preset SAPI value in layer-2 automatic mode. Setting of a new link from the destination is prohibited when this function is executed. However, a link from the source can be set.

Item to be limited                   If this function is executed when the SAPI value that is set is other than 0, 16, and 63, an SAPI error occurs. The PROHIBIT function cannot be executed in transparent mode.

(27) PERMIT\_L

Calling format PERMIT\_L ( )

Argument description

Function value            0            : Normal termination  
                             -51           : Layer-2 automatic mode error  
                             -71           : SAPI error

Example of use PERMIT\_L

Function description Releases the link setting non-permission state set using a PROHIBIT\_L function in layer-2 automatic mode. The link non-permission state is released for the SAPI value that is set in advance.

Item to be limited This PERMIT\_L function cannot be executed in transparent mode.

(28) REG\_\_TEI

Calling format	REG__TEI (TEI)
Argument description	TEI : TEI value to be registered
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -60 : Argument error
Example of use	REG__TEI (64) TEI = 65 REG__TEI (TEI)
Function description	Registers the TEI value in layer-2 automatic mode. The TEI value specified by an argument is registered using this function.
Item to be limited	If the TEI value that has already been registered is registered again, an argument error occurs. The argument error is caused when the argument is other than 0 through 126. The REG__TEI function cannot be executed in transparent mode.

(29) REL\_\_TEI

Calling format	REL__TEI (TEI)
Argument description	TEI : TEI value to be released
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -60 : Argument error
Example of use	REL__TEI (64) TEI=65 REL__TEI (TEI)
Function description	Releases the TEI value registered in layer-2 automatic mode. The value of 0 to 127 can be set in an argument. If "127" is specified as the argument, all TEL values that are registered are released.
Item to be limited	If values other than 0 though 127 are set in the argument, an argument error occurs. The REL__TEI function cannot be executed in transparent mode.

(30) NEXT\_TEI

Calling format                      NEXT\_TEI ( )

Argument description

Function value                      0            : Normal termination  
                                      -51         : Layer-2 automatic mode error

Example of use                      NEXT\_TEI ( )

Function description                Used when a link is set in layer-2 automatic mode using a new TEI value.  
                                      This function is used together with a LINKON function when D5111B is in TE mode.  
                                      The LINKON function is used for link setting. For only the first execution after simulation, a TEI assignment procedure is activated before link setting. In the second or later LINKON function execution, a link with preset SAPI and TEI values is set.  
                                      The TEI assignment procedure is then not activated.  
                                      The NEXT\_TEI function is used before LINKON function execution when a link is set using a new TEI value by reactivating the TEI assignment procedure.  
                                      An REQ\_TEI function can be used in stead of this function.

Item to be limited                    The NEXT\_TEI function cannot be executed in transparent mode. Nothing occurs even if the function is used in NT mode.

(31) ACT\_\_SAPI

Calling format	ACT__SAPI (SAPI)
Argument description	SAPI : SAPI value to be set
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -60 : Argument error
Example of use	ACT__SAPI (0) SAPI = 16 ACT__SAPI (SAPI)
Function description	Used when setting of the SAPI value is changed in layer-2 automatic mode.
Item to be limited	If the SAPI value specified by an argument is other than 0, 16, and 63, an argument error occurs. The ACT__SAPI function cannot be executed in transparent mode.

(32) ACT\_TEI

Calling format	ACT_TEI (TEI)
Argument description	TEI : TEI value to be set
Function value	0 : Normal termination -51 : Layer-2 automatic mode error -60 : Argument error
Example of use	ACT_TEI (64) TEI = 65 ACT_TEI (TEI)
Function description	Used when setting of the TEI value is changed in layer-2 automatic mode.
Item to be limited	If the TEI value specified by an argument is an unregistered one, an argument error occurs. The ACT_TEI function cannot be executed in transparent mode.

(33) LOCK\_SAPI

Calling format                      LOCK\_SAPI ( )

Argument description

Function value                      0            : Normal termination  
                                      -51         : Layer-2 automatic mode error

Example of use                      LOCK\_SAPI ( )

Function description               Prohibits that setting of the SAPI value is changed by the link setting from the destination in layer-2 automatic mode.

However, setting of the SAPI value can be changed using an ACT\_SAPI function after this function is executed.

Item to be limited

The SAPI value is set to the unchangeable state by executing this LOCK\_SAPI function. This state is released using an FLEX\_SAPI function.

The LOCK\_SAPI function cannot be executed in transparent mode.



(34) LOCK\_TEI

Calling format                      LOCK\_TEI (    )

Argument description

Function value                      0            : Normal termination  
                                      -51         : Layer-2 automatic mode error

Example of use                      LOCK\_TEI (    )

Function description                Prohibits that setting of the TEI value is changed by the link setting from the destination in layer-2 automatic mode.  
                                      However, setting of the TEI value can be changed using an ACT\_TEI function after this function is executed.

Item to be limited                   The TEI value is set to the unchangeable state by executing this LOCK\_TEI function. This state is released using an FLEX\_TEI function.  
                                      The LOCK\_TEI function cannot be executed in transparent mode.

(35) FLEX\_\_SAPI

Calling format FLEX\_\_SAPI ( )

Argument description

Function value 0 : Normal termination  
-51 : Layer-2 automatic mode error

Example of use FLEX\_\_SAPI ( )

Function description Releases the unchangeable state of the SAPI value in layer-2 automatic mode.

Item to be limited The FLEX\_\_SAPI function cannot be executed in transparent mode.

(36) FLEX\_TEI

Calling format FLEX\_TEI ( )

Argument description

Function value 0 : Normal termination  
-51 : Layer-2 automatic mode error

Example of use FLEX\_TEI ( )

Function description Releases the unchangeable state of the TEI value in layer-2 automatic mode.

Item to be limited The FLEX\_TEI function cannot be executed in transparent mode.

*MEMO*



A large, empty rectangular area with rounded corners, enclosed by a dashed border, intended for writing the memo's content.

## 4. EDITOR

This chapter describes how to use an editor (ate (AdvanTest Editor)). The editor of the D5111B is used for creation of a simulation program or start-up file. This editor that is a full-screen editor has various functions to ensure the input efficiency improvement and easy edition.

Configuration of chapter 4		
Editor	4.1 Editor activation and release	
	4.2 Editor command	File Point movement Text erase Mark Retrieval and replacement Window Text conversion Buffer Keyboard macro Compiler Line/column information
	4.3 Keyboard macro	

## 4.1 Editor Activation and Release

The editor is activated and released in accordance with the procedure below.

- ① Press the F1 key to display the function module menu.
- ② Move the cursor to the "EDITOR" position using the ▲ and ▼ keys.
- ③ Press the space key or return key. The "EDITOR" screen is then displayed.
- ④ Press the F3 key to display a menu at the top of the screen.  
The functions defined by this menu are described below.

### "Window"      Screen control

	Only	:	Single screen
	Split	:	Upper/lower two screens
	Next	:	Moves the cursor to another screen.

### "Region"      Region

	Set mark	:	Mark
	Kill region	:	Deletes the specified range.
	Copy region	:	Copies the specified range.
	Yank	:	Loads the deleted text.

### "File"          File operation



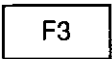

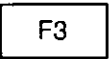
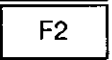


	Find	:	Specifies a file.
	Find read-only	:	Reads a file (read only).
	Save	:	Saves a file.
	Write	:	Writes a file.

### "Others"        Compiler related operation

	Help	:	Help screen (sequence command directory)
	Execute compile	:	Activates a compiler.
	Go to error	:	Moves the cursor to a compile error line.
	Go to line	:	Moves the cursor to a specified line.

### "Search/Replace

	Forward search	:	Retrieves forward.
	Backward search	:	Retrieves backward.
	Query replace	:	Replaces (in interactive mode).
	Replace string	:	Replaces (in automatic mode).

- ⑤ Name the program to be edited. Move the cursor to the "File" position using the  and  keys. Press the space key to display the popup menu for "Find/Find read-only/Save/Write." Select "Find." The file name is inquired. Enter a proper name (e.g., "TEST.PRG"). Then, press the return key.
- ⑥ Press the  key. The menu screen disappears and the system enters the key-in state. Enter a proper program.
- ⑦ After the program entry is completed, press the  key to display the menu screen. To save data as a file, move the cursor to the "File" position, display the popup menu, then select "Save." To compile it, display the popup menu for "Others" and select "Execute compile."
- ⑧ Press the  key to release the edit. The menu screen then disappears. Press the  key to display the window menu. Select "Quit" using the  and  keys and press the space key or return key. Saving of the created file is inquired. Enter "Y" to save the file or "N" to avoid saving it, and press the return key. "Quit [y/n]?" (for releasing the editor) is then displayed. Enter "Y" to release or "N" to edit,

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**4.1 Editor Activation and Release**

**Note for executing the editor function**

A maximum of 256 characters is specified in one line.

When the character count in one line exceeds the maximum length, indicates \$ at the right rim in a display. Indicates \$ at the left rim in a display when the display scrolls and the character that is existent at the left side from this point cannot indicate.

Two type of displays is as follows.

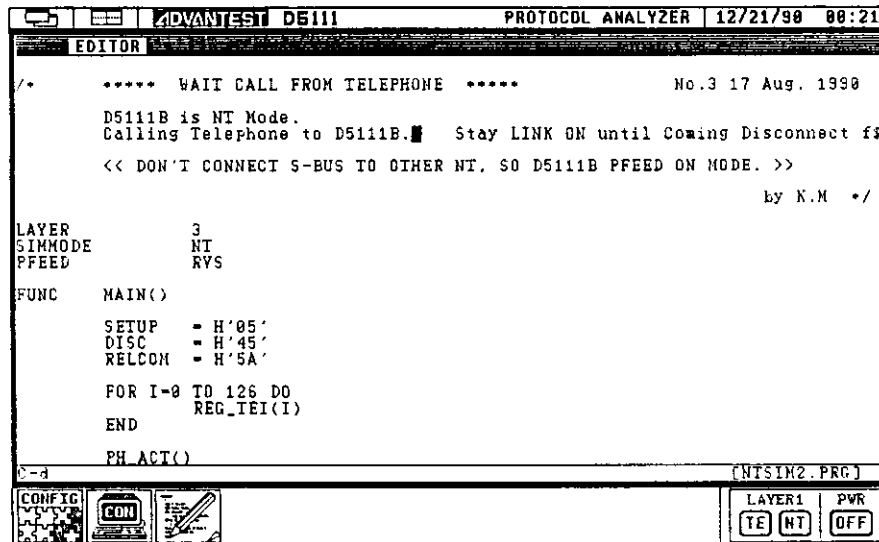


Figure 4-1 When a character count in one line exceeds the maximum length

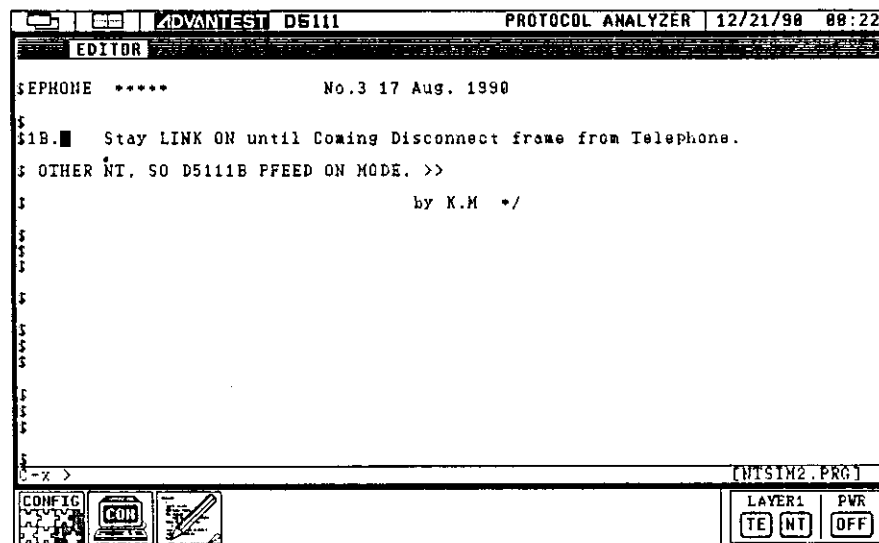


Figure 4-2 When the display scrolls in the right direction



## 4.2 Editor Command

This editor has various edit commands. Function keys are assigned in the same way as in a general-purpose editor.

### ① File

- CTRL-x CTRL-f : Reads a file (Can be read/written.)
- CTRL-x CTRL-r : Reads a file (Read only)
- CTRL-x CTRL-s : Writes a file during edition in the disk.
- CTRL-x CTRL-w : Specifies a file name and writes it in the disk.
- CTRL-x & : Displays the current file name.

### ② Point movement

- CTRL-f : Moves to the right by one character.
- CTRL-b : Moves to the left by one character.
- ESC-f : Moves to the right by one word.
- ESC-b : Moves to the left by one word.
- CTRL-n : Moves downward by one line.
- CTRL-p : Moves upward by one line.
- CTRL-a : Moves to the beginning of a line.
- CTRL-e : Moves to the end of a line.
- ESC-< : Moves to the beginning of a buffer.
- ESC-> : Moves to the end of a buffer.
- CTRL-x CTRL-g : Specifies the line of the movement destination.
- CTRL-x CTRL-n : Moves to the error line (after compile).
- CTRL-v : Moves forward by one page.
- ESC-v : Moves backward by one page.
- CTRL-x -- : Scrolls the screen downward by one line.
- CTRL-x \* : Scrolls the screen upward by one line.
- ESC-& : Moves the point position to the uppermost position of the window.

③ Text erase

- CTRL-d or DEL : Erases a character just after the cursor.
- CTRL-h or BS : Erases a character just before the cursor.
- ESC-d : Deletes to the end of the next word in the forward direction.
- ESC-BS : Deletes to the top of the preceding word in the reverse direction.
- CTRL-k : Deletes to the end of a line.
- CTRL-w : Deletes from the mark position to the cursor.
- CTRL-x CTRL-o : Deletes a continuous blank line.
- ESC-w : Copies an area to the kill buffer.

④ Text insertion

- CTRL-i or TAB : Inserts a tab.
- CTRL-q : Inserts a nongraphic character.
- CTRL-m or RET : Feeds a line.
- CTRL-j or LF : Performs a line feed and indent.
- CTRL-o : Inserts one blank line.
- CTRL-y : Inserts the stored character string before the cursor.

⑤ Mark

- CTRL-spc or @ : Sets a mark to the point position.
- CTRL-x CTRL-x : Replaces a mark by a point.

⑥ Retrieval and replacement

- CTRL-s : Retrieves in the forward direction.
- CTRL-r : Retrieves in the reverse direction.
- ESC-% : Replaces the related words while confirming them one by one.
- ESC-\* : Replaces the related words collectively.
- CTRL-t : Replaces the cursor by the preceding character.

⑦ Window

- CTRL-x 1 : Erases windows other than the selected window.
- CTRL-x 2 : Splits the selected window horizontally.
- CTRL-x o : Selects other windows.
- CTRL-x CTRL-z : Narrows the selected window upward or downward.
- CTRL-x ^ : Widens the selected window upward or downward.
- CTRL-x < : Scrolls the selected window to the left.
- CTRL-x > : Scrolls the selected window to the right.

⑧ Text conversion

- ESC-1 : Converts the word to an lower-case character.
- ESC-u : Converts the word to a upper-case character.
- ESC-c : Converts the beginning of a word to an upper-case character.
- CTRL-x CTRL-l : Converts the region to a lower-case character.
- CTRL-x CTRL-u : Converts the region to an upper-case character.

⑨ Buffer

- CTRL-x b : Selects another buffer.
- CTRL-x CTRL-b : Displays a buffer list.
- CTRL-x k : Specifies the buffer name and deletes the buffer.

⑩ Keyboard macro

- CTRL-x ( : Starts the keyboard macro definition.
- CTRL-x ) : Ends the keyboard macro definition.
- CTRL-x e : Executes the keyboard macro.

⑪ **Complier**

CTRL-x CTRL-e : Executes a compile

⑫ **Line/column information**

CTRL-x ! : Displays the line number of the cursor position.

CTRL-x 1 : Displays the number of lines per page.

CTRL-x = : Display the cursor position.

⑬ **Others**

CTRL-1 : Redisplays the screen information.

CTRL-g : Cancels a command (also cancels a menu).

ESC-# : Displays a popup menu (The popup menu is also displayed using an F3 key.)

HELP : Displays the help screen.

Enter a -o flag after the source file name; the object is saved in a file. The default's object file name is —. SIM.

**Example**

- TEST.PRG -o [RETURN]  
→ The object is saved in TEST.SIM.
- TEST.PRG -o TEST.OBJ [RETURN]  
→ The object is saved in TEST.OBJ.

### 4.3 Keyboard Macro

The keyboard macro is used to define a macro through combined use of some functions of an editor and execute it as one command. It is valid when several command operations are performed continuously.

The start of macro definition is reported to the editor using the "CTRL-x (" key. Enter each key code corresponding to the functions you wish to execute. Press the "CTRL-x )" key to end the macro definition after the code entry is completed. To execute the macro as a command, press the "CTRL-x e" key. Some of the macro-defined functions are then executed.

The defined macro is valid until it is defined again.

(Example) To enter tabs at the beginning of all lines.

- ① Move the cursor to the beginning of a buffer.
- ② CTRL-x (  
↓ [Defining in keyboard macro....]  
"TAB"  
↓  
CTRL-n  
↓  
CTRL-a  
↓  
CTRL-x ( [Keyboard macro was defined.]  
Macro definition is completed.
- ③ All tabs can be entered using the "CTRL-x e" key.

*MEMO*








A large, empty rectangular area with rounded corners, enclosed by a dashed border, intended for writing the memo's content.

## 5. CONSOLE

This chapter describes the console commands of the D5111B.







Configuration of chapter 5		
Console	5.1 Disk I/O related command	chdir chkdsk copy delete directory dump erase format hformat label mkdir more print rmdir rename renedir load save scopy type
	5.2 Environment set command	date sptr stty sfont
	5.3 Other commands	batch cls echo help
	5.4 Error message	






## 5.1 Disk I/O Related Command

- ① **chdir (cd)**
- Function : Changes or displays the current directory. The current directory is displayed if the path name is omitted.
- Format : `chdir [<Path name>]`
- Another notation : `cd`
- Example of use : When the directory A, B, C exists  
> `chdir ABC` 
- ② **chkdsk**
- Function : Checks the disk state of a specified drive. The disk state of the current drive is checked if the drive specification is omitted.
- Format : `chkdsk [<Device name>]`
- Example of use : When you wish to know the state of the hard disk  
> `chkdsk sd0` 
- ③ **copy**
- Function : Creates a file copy.
- Format : `copy [<File name>]{<File name>}`
- Example of use : When the file AAA is copied to the file BBB  
> `copy AAA BBB` 
- ④ **delete**
- Function : Deletes the specified file. A wild card (\*, ?) can be used in the file name.
- Format : `delete [<File>]`
- Example of use : When the file AAA is deleted  
> `Delete AAA` 
- ⑤ **directory**
- Function : Displays the directory content. The details of the directory are displayed when a mode switch is set to -L. Only the file name in the directory is displayed if the mode switch is omitted. The file name in the current directory is displayed if the mode switch and file name are omitted.
- Format : `directory [-L]{<File name>}`
- Example of use : When you wish to know the filename of the current directory.  
> `dir` 



- 
- ⑥ dump
- Function : Displays the file content in hexadecimal and ASCII codes in one-screen units.
- Format : dump [ <File name > ]
- Example of use : When the file AAA is displayed in hexadecimal  
> dump AAA
- ⑦ erase
- Function : Erases the specified file. A wild card (\*, ?) is supported for the file name. The erase command is the same in function as a delete command.
- Format : erase [ <File name > ]
- Example of use : When the file AAA is deleted  
> erase AAA
- ⑧ format
- Function : Initializes a floppy disk. The drive to be used is an internal drive.
- Format : format [-S/-U]  
[-S]: Soft format  
[-U]: Creation of user data directory (after format)
- Example of use : When the floppy disk is formatted  
> format
- ⑨ hformat
- Function : Initializes the hard disk specified by a drive ID. Only a soft format is executed when the mode switch is set to -S. A user data directory (default value) is created after format when it is set to -U.
- Format : hformat [-S/-U][ <Drive ID > ]  
Drive ID: 0 to 7 [SCSI ID number]
- Example of use : When the internal hard disk is formatted  
> hformat 0
- ⑩ label
- Function : Register (changes) a disk label. The label name of the current drive is displayed if the label name is omitted.
- Format : label [ <Label name > ]
- Example of use : When HDD is specified in the label name of internal hard disk  
> Label HDD

- ⑪ mkdir (md)
- Function : Creates a new directory.
- Format : mkdir [ <Path name > ][ <Directory name > ]
- Another notation : md
- Example of use : When the directory AAA is created in the current directory  
> mkdir AAA 
- ⑫ more
- Function : Displays the file content in one-screen units. In the key-in wait state, the system can be interrupted using the "CTRL-c" key.
- Format : more [ <File name > ]
- Example of use : When the file "AAA" content is displayed  
> more AAA 
- ⑬ print
- Function : Outputs a file to the printer. Up to ten files can be output at the same time. A wild card can be used for specification of a file name.
- Format : print [ <File name > ]
- Example of use : When the file AAA is output  
> print AAA 
- ⑭ rmdir (rd)
- Function : Deletes the specified directory.
- Format : rmdir [ <Directory name > ]
- Another notation : rd
- Example of use : When the directory ABC is deleted  
> rmdir ABC 
- ⑮ rename
- Function : Changes the source file name to the destination file name.
- Format : Format: rename [ <File name > ][ <File name > ]
- Example of use : When the file name AAA changes to the BBB  
> rename AAA BBB 
- ⑯ rendir
- Function : Changes the source directory name to the destination directory name.
- Format : Format: rendir [ <Directory name > ][ <Directory name > ]
- Example of use : When the directory name ABC changes to the DEF  
> rendir ABC DEF 

- ①7 load
- Function : Loads a system set file.
- Format : load [-TA/-TB][ <File name > ] : Trace data loading  
[-SA/-SB][ <File name > ] : Setting data loading  
[-MA/-MB][ <File name > ] : Message data loading  
[protocol][ <File name > ] : User-defined data loading
- Example of use : When the trace data of port A, the file name is DATA, is loaded  
> load -ta DATA   
When the user-defined data of LAPD, the file name is BBB, is loaded  
> load lapd BBB 
- ①8 save
- Function : Writes the system setting state as a file (system setting file).
- Format : save [-TA/-TB][ <File name > ] : Trace data save  
[-SA/-SB][ <File name > ] : Setting data save  
[-MA/-BM][ <File name > ] : Message data save  
[protocol][ <File name > ] : User-defined data save
- Example of use : When the trace data of port A 10 through 50 line is written as a file name "DATA".  
> save -ta 10 50 DATA   
When the user-defined data of LAPB8, the file name is CCC, is written  
> load lapb8 CCC 
- ①9 scopy
- Function : Copies from the floppy disk to the floppy disk in a single disk drive.
- Format : scopy [ <Source file name > ][ <Destination file name > ]
- Example of use : Example of use: When the file AAA in the floppy disk 1 is copied to the floppy disk 2  
> scopy AAA AAA 

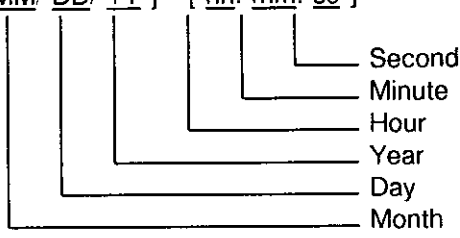
D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL

5.1 Disk I/O Related Command





---

- ② type
- Function : Displays the content of a file.
- Format : type [ <File name > ]
- Example of use : When the content of the file AAA is displayed  
> type AAA

## 5.2 Environment Set Command

- ②1 date
- Function : Sets the date and time. The current date and time are displayed if the date and time are omitted.
- Format : date [ MM/ DD/ YY ] [ hh: mm: ss ]
- 
- Example of use : When the date is set to November 8, 1990 and the time is set to P.M. 3:25
- ```
> date 11/8/90 3:25 ↵
```
- ②2 sprt
- Function : Selects the type of a printer.
- Format : sprt [ <Printer name> ]  
Printer name: NEC/EPSON/OTHERS
- Example of use : When the printer to be used is produced by NEC
- ```
> sprt nec ↵
```
- ②3 stty
- Function : Sets a communication port.
- Format : stty [Parameter group]
- Baud rate: 300/600/1200/2400/4800/9600  
Bit length: 7/8  
Stop bit: 1/2  
Parity: Odd/even/none  
x control: Enable/disable
- Example of use : For baud rate 9600, bit length 8, stop bit 1, parity none, x control enable
- ```
> stty 9600 8 1 n en ↵
```
- ②4 sfont
- Function : Sets the display font on the console screen. The display font has S (Sm-all) and M (Middle) types.
- Format : sfont [S/M]
- Example of use : When the Small is selected as a display font.
- ```
> sfont s ↵
```

### 5.3 Other Commands

- ②5 batch
- Function : Executes a batch file.
- Format : batch [ <File name > ]
- Example of use : When a batch file "AAA" is executed  
> batch AAA 
- ②6 cls
- Function : Erases the screen.
- Format : cls
- Example of use : When the display is deleted  
> cls 
- ②7 echo
- Function : Outputs a message on the screen.
- Format : echo [ <Message > ]
- Example of use : When the message "THIS IS D5111B" is displayed  
> echo "THIS IS D5111B" 
- ②8 help
- Function : Displays the help for the command group supported by a console.
- Format : help
- Example of use : When the command group to be usable by a console  
> help   
To press the help key executes the same function as the upper way.

## 5.4 Error Message

### Batch Command In Batchfile

- Cause : A batch command is included in the batch file.
- Action : Releases the batch command in the batch file.

### Bad Command Length

- Cause : An error occurs in the length of a specified command.
- Action : Reenter correctly while referring to the command list.

### Command Error

- Cause : An error occurs in the command entry.
- Action : Reenter correctly while referring to the command list.

### Command Not Found

- Cause : An error occurs in the command entry.
- Action : Reenter correctly while referring to the command list.

### Destination File Error

- Cause : An error occurs in the name of a specified destination file.
- Action : Enter a correct destination file name.

### Directory Not Found

- Cause : No specified directory exists.
- Action : Enter a correct directory name.

### File Not Found

- Cause : No specified file exists.
- Action : Enter a correct file name.

### Illegal"

- Cause : An error occurs in the use of (").
- Action : Use (") correctly.

### Illegal '

- Cause : An error occurs in the use of (').
- Action : Use (') correctly.

**Memory Allocation Error**

- Cause :** No work area can be secured in memory.
- Action :** Quit an unused window so that memory can be secured.

**Memory Free Error**

- Cause :** The work area in memory cannot be released.
- Action :** Memory is assumed to be abnormal. Restart the system.

**No Support Printer**

- Cause :** The printer is not supported.
- Action :** Select a printer that can be supported.

**Not Digit**

- Cause :** Characters other than numerics are specified for a numeric parameter.
- Action :** Enter using numerics.

**Not Option**

- Cause :** The option is not supported.
- Action :** Select an option that can be supported.

**Parameter Error**

- Cause :** No specified parameter is used.
- Action :** Use a specified parameter.

**Unset Command**

- Cause :** The input command cannot be used.
- Action :** Select a command that can be used while referring to the command list.

**Where Is Directory Name?**

- Cause :** The directory name is unknown.
- Action :** Reenter a correct directory name.

**Disk Operation Error**

- Cause :** The disk operation is abnormal.
- Action :** If a same message is displayed after retry, the media or disk drive is abnormal.



**Filename Too Long**

**Cause :** The file name is longer than a prescribed one.

**Action :** Enter a file name not exceeding the prescribed length.

**Illegal Function Call**

**Cause :** An error occurs in the function call.

**Action :** Enter a correct function call.

*MEMO*



A large, empty rectangular area with rounded corners, enclosed by a dashed border, intended for writing the memo's content.

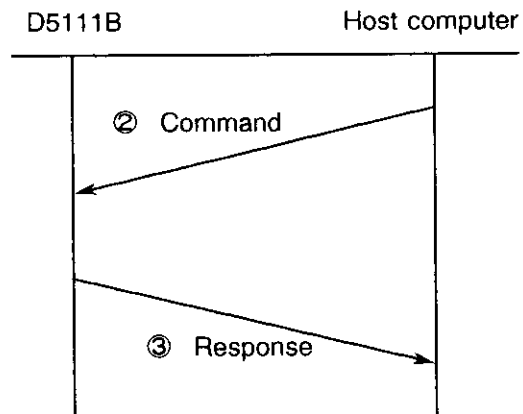
## 6. EXTERNAL CONTROL

This D5111B can exchange data in series with the external computer or system via the EIA-232D connector on the rear panel. The EIA-232D interface condition can be set on the utility menu or console screen.

Configuration of chapter 6			
External control	6.1	Remote control execution	
	6.2	Remote control command	
	6.3	Interface	
	6.4	Command explanation	Read of frame count in trace data Frame information read Frame data read Monitor operation activation Monitor operation stop Monitor operation state check Simulation activation Simulation stop Simulation state Simulation message read Setting file loading Message file loading Simulation object file loading Trace data save Time setting Date setting Time read Date read Output data delimiter setting

## 6.1 Remote Control Execution

- ① To control this D5111B remotely, make the EIA-232D interface condition of a host computer same as that of the D5111B.
- ② Send a command from the host computer.
- ③ This D5111B interprets and executes the received command, then sends the result data as a response.



## 6.2 Remote Control Command

The remote control command is sent from a host computer to this D5111B to control the D5111B operation from the external unit.

(1) The command functions are as follows:

- Trace data read
- Monitor operation control
- Simulation control
- Disk operation
- Time setting/read
- Output data delimiter setting

(2) The command conditions are as follows:

- All commands conform to the ASCII code.
- All alphabetic characters are upper-case characters.
- The numeric is represented in decimals.
- The command ends in CR or CR + LF.
- The response ends in CR or CR + LF. You can specify CR or CR + LR.
- The maximum length of a command is 199 characters including CR. If the command exceeds 199 characters, response message "OVER" is returned.

### 6.3 Interface

(1) Connector

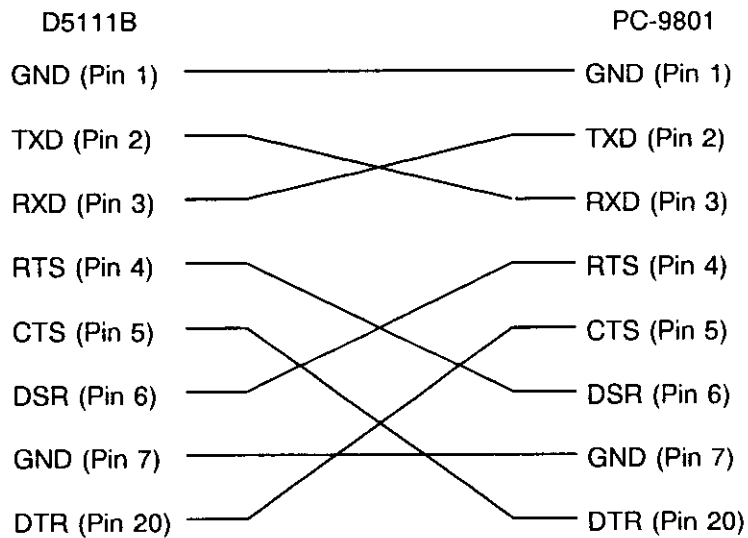
Pin No	Signal name	Pin No	Signal name
1	GND	14	NC
2	TXD	15	NC
3	RXD	16	NC
4	RTS	17	NC
5	CTS	18	NC
6	DSR	19	NC
7	GND	20	DTR
8	NC	21	NC
9	NC	22	NC
10	NC	23	NC
11	NC	24	NC
12	NC	25	NC
13	NC		

(2) Connection

This D5111B operates as data terminal equipment (DTE).

(Connection example)

To connect the D5111B to PC-9801 (NEC), connect as shown in the figure below.



## 6.4 Command Explanation

### (1) Read of frame count in trace data

(Command)

?MON\_RANGE,port

(Response)

OK, number : Normal termination  
BUSY : A monitor is in operation.  
NG : Error  
(An error occurs in the port specification.)

(Parameter)

port : Specify the port of data you wish to read.  
A, B  
number : The number of frames is returned in decimals.

(Description)

Reads the number of frames included in data that is loaded in this D5111B using a monitor function. The frame cannot be read when the monitor function is being used.

(Example)

Store the number of frames included in data that is loaded in port A in variable NUMBER. (The program is written by N88-BASIC.)

```
OPEN"COM:N81XN" AS #1
*loop
PRINT #1, "?MON_RANGE,A"
INPUT #1,HDR$
IF HDR$ = "BUSY" THEN GOTO *loop
IF HDR$ = "NG" THEN GOTO *fail
INPUT #1,NUMBER
PRINT "THE NUMBER OF FRAME IS";NUMBER
GOTO *prgend
*fail
PRINT"ERROR!!"
*prgend
CLOSE #1
```

(2) Frame information read

(Command)

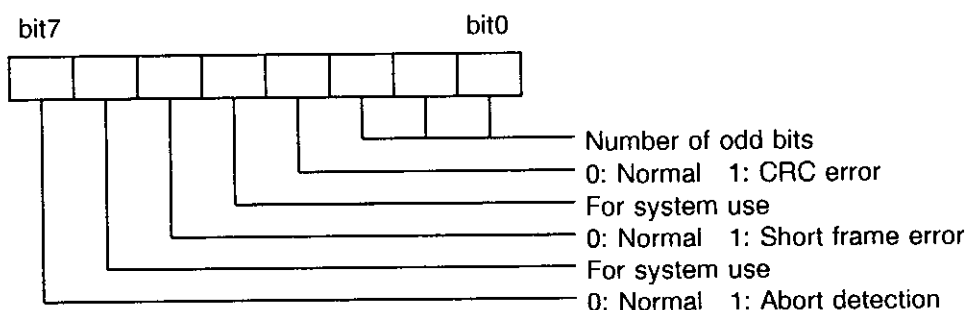
?MON\_\_INFO,port,seq

(Response)

- OK,size,line,err,time : Normal termination
- EOF,size,line,err,time : Normal termination (Last frame data)
- BUSY : A monitor is in operation.
- NOTHING : No data exists.
- NG : Error (No specified frame exists.)

(Parameter)

- port : Specify the port of data you wish to read.  
A, B
- seq : Specify the frame you wish to read.  
0,1,... (Frame number)  
\$ (Last frame)  
+ (Next frame)  
. (Same frame)
- size : Represents the size of loaded data in octet units.
- line : Represents the line in which a frame is loaded.  
USR, NET
- err : Frame error information. Eight-bit information is represented in hexadecimal by two digits. (Example: &H80 abort detection)



- time : Represents the time in which a frame is loaded. The value is a time elapsed from when a monitor function is activated. The unit is 0.1 msec.



(Description)

Reads the information (size, line, error, and time) on a frame to be loaded. This information cannot be read when a monitor function is operating.

(Example)

Display the frame information at the beginning of data that is loaded in port A.

```
PRINT #1,"?MON__INFO,A,O"  
INPUT #1,HDR$  
INPUT #1,SIZE,NET$,ERRINFO,TIMESTAMP  
PRINT "SIZE = ";SIZE  
PRINT "LINE = ";NET$  
PRINT "TIMESTAMP = ";TIMESTAMP;"ms"  
AB = ERRINFO AND &H80  
SH = ERRINFO AND &H20  
CRC = ERRINFO AND &H08  
BIT = ERRINFO AND &H07  
IF AB = &H80 THEN PRINT "ABORT FRAME"  
IF SH = &H20 THEN PRINT "SHORT FRAME"  
IF CRC = &H08 THEN PRINT "CRC ERROR"  
IF BIT > 0 THEN PRINT "NONOCTET FRAME,";BIT;"BIT"
```

(3) Frame data read

(Command)

?MON\_DATA,port,seq,offset,max

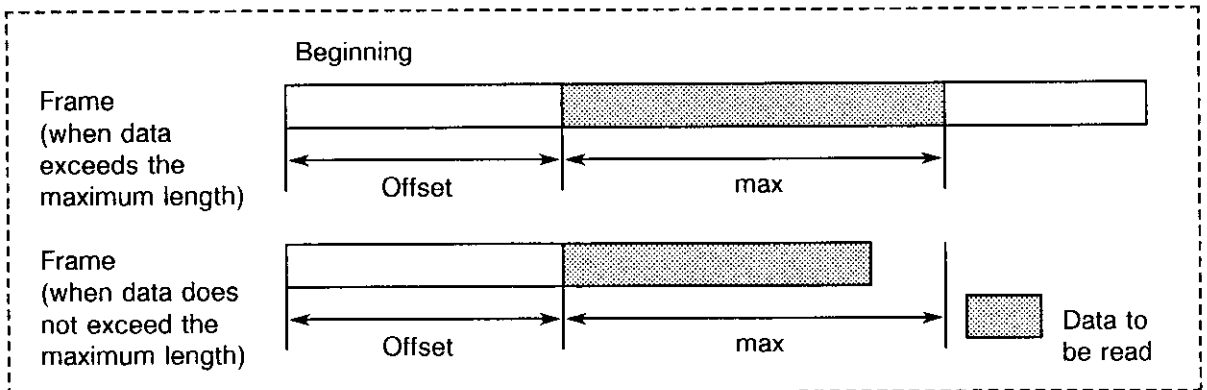
(Response)

- OK,data : Normal termination
- EOF,data : Normal termination (Last frame data)
- BUSY : A monitor is in operation.
- NOTHING : No data exists.
- NODATA : The offset is longer than the frame length.
- NG : Error (No specified frame exists.)

(Parameter)

- port : Specify the port of data you wish to read.  
A, B
- seq : Specify the frame you wish to read.  
0,1,.. (Frame number)  
\$ (Last frame)  
+ (Next frame)  
. (Same frame)
- offset : Specifies the offset value from the beginning of the frame of data to be read in octet units (see the figure below).
- max : Specifies the maximum length of data to be read in octet units (see the figure below).

Note: The maximum length of data is 100 octets (200 characters). Only 100 octets are read if 100 octets or more are specified.



data : Data string

(Description)

Reads the data of a frame to be loaded. The data cannot be read when a monitor function is operating.

(Example)

The statistics for the TEI value of a frame at port A are taken. The result data is stored in the STEI array.

```
DIM STEI (127)
PRINT #1,"?MON__DATA,A,0,0,1"
INPUT #1,HDR$
INPUT #1,TEIDATA$
TEIVALUE$ = "&H" + TEIDATA$
TEI = INT(VAL(TEIVALUE$)/2)
STEI(TEI) = STEI(TEI) + 1
IF HDR$ = "EOF" THEN GOTO *finish
*loop
PRINT #1,"?MON__DATA,A, +,0,1"
INPUT #1,HDR$
INPUT #1,TEIDATA$
TEIVALUE$ = "&H" + TEIDATA$
TEI = INT(VAL(TEIVALUE$)/2)
STEI(TEI) = STEI(TEI) + 1
IF HDR$ = "EOF THEN GOTO *finish ELSE GOTO *loop
*finish
    ?Processing routine for display
```

(4) Monitor operation activation

(Command)

IMON\_\_RUN

(Response)

OK

(Description)

Activates the monitor operation. All monitors at a port that are installed using this command are activated.

(Example)

Activate a monitor operation.

```
PRINT #1,"IMON RUN"  
INPUT #1,RSP$
```

(5) Monitor operation stop

(Command)

ISTOP

(Response)

OK

(Description)

Stops the monitor/simulation operation. All monitors at a port that are installed using this command stop.

(Example)

Monitor for 7:00 to 8:00.

```
MONSTATUS = 0
ON TIME$ = "07:00:00" GOSUB *monstart
*loop1
IF MONSTATUS = 0 THEN GOTO *loop1
ON TIME$ = "08:00:00" GOSUB *monstop
*loop2
IF MONSTATUS = 1 THEN GOTO *loop2
```

'Processing after monitor stop

```
*monstart
PRINT #1, "IMON_RUN"
INPUT #1,RSP$
MONSTATUS = 1
RETURN
```

```
*monstop
PRINT #1, "ISTOP"
INPUT #1,RSP$
MONSTATUS = 0
RETURN
```

(6) Monitor operation state check

(Command)

?MON\_STAT,port

(Response)

RUNNING : In-operation  
STOPPED : Stop  
NG : Error (An error occurs in the port name.)

(Parameter)

port : Specify the port you wish to read.  
A, B

(Description)

Reads the monitor operation state.

(Example)

Wait that the monitor is stopped by a trigger to be set after monitor activation.

```
PRINT #1,"!MON_RUN"  
INPUT #1,RSP$  
  
*loop  
PRINT #1,"?MON_STAT,A"  
INPUT #1,STATES$  
IF STATES$ = "RUNNING" THEN GOTO *loop  
IF STATES$ = "STOPPED" THEN GOTO *finish  
  
*finish  
  'Processing after monitor stop
```

(7) Simulation activation

(Command)

!SIM\_\_RUN,port

(Response)

OK : Normal termination

BUSY : A monitor is in operation.

NG : Error (An error occurs in the port name.)

(Parameter)

port : Specify the port for simulation you wish to execute.  
A

(Description)

Activates the simulation function. The function cannot be activated when a simulation menu is not loaded.

(Example)

Activate the simulation at port A.

```
PRINT #1,"!SIM__RUN,A"  
INPUT #1,RSP$
```

(8) Simulation stop

(Command)

ISTOP

(Response)

OK

(Description)

Stops the simulation/monitor operation.

(Example)

Stop the simulation at port A.

PRINT #1,"ISTOP"

INPUT #1,RSP\$



(9) Simulation state

(Command)

?SIM\_\_STAT,port

(Response)

RUNNING : In-operation

STOPPED : Stop

NG : Error (An error occurs in the port name.)

(Parameter)

port : Specify the port in the state where you wish to read.  
A

(Description)

Reads the operation state of a simulation function.

(Example)

Wait that a simulation program is completed after simulation is activated at port A.

```
PRINT #1,"ISIM__RUN"  
INPUT #1,RESPONSE$
```

```
*loop  
PRINT #1,"?SIM__STAT A"  
INPUT #1,STATE$  
IF STATE$ = "RUNNING" THEN GOTO *loop  
IF STATE$ = "STOPPED" THEN GOTO *finish
```

```
*finish  
'Processing after simulation stop
```

(10) Simulation message read

(Command)

?SIM\_MSG,port

(Response)

OK,message

NOTHING : No message exists.

NG : Error (An error occurs in the port name.)

(Parameter)

port : Specify the port of a message you wish to read.  
A

message : Message

(Description)

Reads the message that is output by a PRINT statement during simulation execution.

(Example)

Read and display a simulation message in sequence  
(when the last message of a simulation program is "SIMULATION END").

```
*loop
PRINT #1,"!SIM_MSG,A
INPUT #1,RSP$
IF RSP$ = "NOTHING" GOTO *loop
INPUT #1,MESSAGE$
PRINT MESSAGE$
IF MESSAGE$ = "SIMULATION END" THEN GOTO *finish
GOTO *loop

*finish
'Processing after simulation end
```

(11) Setting file loading

(Command)

!LOAD\_\_SET,port,file

(Response)

OK : Normal termination

NG : Error (An error occurs in the file name.)

(Parameter)

port : Port name (A or B)

file : File name (including a path name)

(Description)

Sets the content of a setting file to be specified.

(Example)

Set the content of a test.set file to port A, then monitor.

```
PRINT #1,"!LOAD__SET,A,FD0:/SETUP/TEST.SET"  
INPUT #1,RSP$  
PRINT #1,"!MON__RUN"  
INPUT #1,RSP$
```

(12) Message file loading

(Command)

ILOAD\_\_MSG,port,file

(Response)

OK : Normal termination

NG : Error (An error occurs in the file name.)

(Parameter)

port : Specify the port of a message you wish to set.  
A

file : File name (including a path name)

(Description)

Sets the send message used in simulation by a specified file.

(Example)

Set the content of a test.msg file, then execute the simulation at port A.

```
PRINT #1,"ILOAD__MSG,A,FD0:/MESSAGE/TEST.MSG"  
INPUT #1,RSP$  
PRINT #1,"ISIM__RUN,A"  
INPUT #1,RSP$
```

(13) Simulation object file loading

(Command)

ILOAD\_\_SIM,port,file

(Response)

OK : Normal termination

NG : Error (An error occurs in the file name.)

(Parameter)

port : Specify the port of a message you wish to read.  
A

file : File name (including a path name)

(Description)

Loads the execution-form file of a specified simulation program in this D5111B.

(Example)

Execute the simulation of a test.obj execution file at port A.

```
PRINT #1,"ILOAD__SIM,A,FD0:/OBJECT/TEST.OBJ"  
INPUT #1,RSP$  
PRINT #1,"ISIM__RUN,A"  
INPUT #1,RSP$
```

(14) Trace data save

(Command)

ISAVE\_\_MON,port,start,stop,file

(Response)

OK	:	Normal termination
BUSY	:	A monitor is in operation.
NG	:	Error (An error occurs in the file name.)
NOTHING	:	No data is loaded, so data cannot be saved in a disk.

(Parameter)

port	:	Specify the port of a message you wish to read. A, B
start	:	Specify the beginning frame number by specifying the specified range to be saved.
stop	:	Specify the last frame number by specifying the specified range to be saved.
file	:	File name (including a path name)

(Description)

Saves the data loaded in the D5111B on a disk.

(Example)

Save the 100 frames (with frame numbers 0 to 99) of the trace data at port A on a floppy disk as file name TEST.DAT. If the traced frames are under 100, only the traced frame is saved.

```
PRINT #1,"ISAVE__MON,A,0,99,FD0:TEST.DAT"  
INPUT #1,RSP$
```

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**6.4 Command Explanation**

---

(15) Time setting

(Command)

!TIME, hh:mm:ss

(Response)

OK : Normal termination

NG : Error (The time is specified incorrectly.)

(Parameter)

hh : Hour

mm : Minute

ss : Second

(Description)

Sets the internal timer of the D5111B.

(Example)

Set the timers in a host computer and this D5111B to the same time.

```
PRINT #1,"!TIME,";TIME$  
INPUT #1,RSP$
```

(16) Date setting

(Command)

IDATE,mm/dd/yy

(Response)

OK : Normal termination

NG : Error (The date is specified incorrectly.)

(Parameter)

mm : Month

dd : Day

yy : Year

(Description)

Sets the internal timer of the D5111B.

(Example)

Set the date to March 20, 1990.

```
PRINT #1,"IDATE,3/20/90"  
INPUT #1,RSP$
```



(17) Time read

(Command)

?TIME

(Response)

OK,hh:mm:ss

(Parameter)

hh : Hour

mm : Minute

ss : Second

(Description)

Reads the time of a timer in the D5111B.

(Example)

Read the time, then display.

```
PRINT #1,"?TIME"  
INPUT #1,HDR$  
INPUT #1,VALUE$  
PRINT "TIME IS";VALUE$
```

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**6.4 Command Explanation**

---

(18) Date read

(Command)

?DATE

(Response)

OK,mm:dd:yy

(Parameter)

mm : Month

dd : Day

yy : Year

(Description)

Reads the date of a timer in the D5111B.

(Example)

Read the date, then display.

```
PRINT #1,"?DATE"  
INPUT #1,HDR$  
INPUT #1,VALUES$  
PRINT"DATE IS";VALUES$
```

(19) Output data delimiter setting

(Command)

IDEL,code

(Response)

OK

(Parameter)

code : 1 (CR + LF)  
2 (CR only)

(Description)

Sets a delimiter that is output at the end of a response. The default is CR + LF.

(Example)

Change the termination delimiter from "CR + LF" to "CR."

```
PRINT #1,"IDEL 2"  
INPUT #1,RESPONSE$
```

*MEMO*



A large, empty rectangular area with rounded corners, enclosed by a dashed border, intended for writing the memo's content.

## 7. PRINTER AND HARD DISK

This chapter describes how to use a printer and a hard disk.

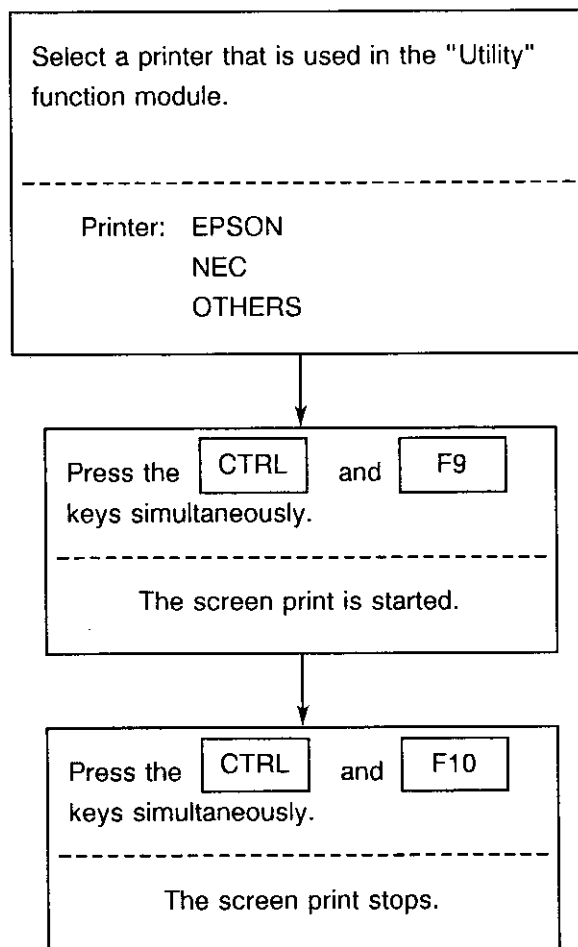
Configuration of chapter 7		
Printer and hard disk	7.1 Printer output	Screen print Translation data print Print on console
	7.2 Hard disk	System installation SCSI-compatible hard disk expansion

## 7.1 Printer Output

This D5111B can output the screen directly to a printer, and can print the translation data or simulation program and record it.

### 7.1.1 Screen print

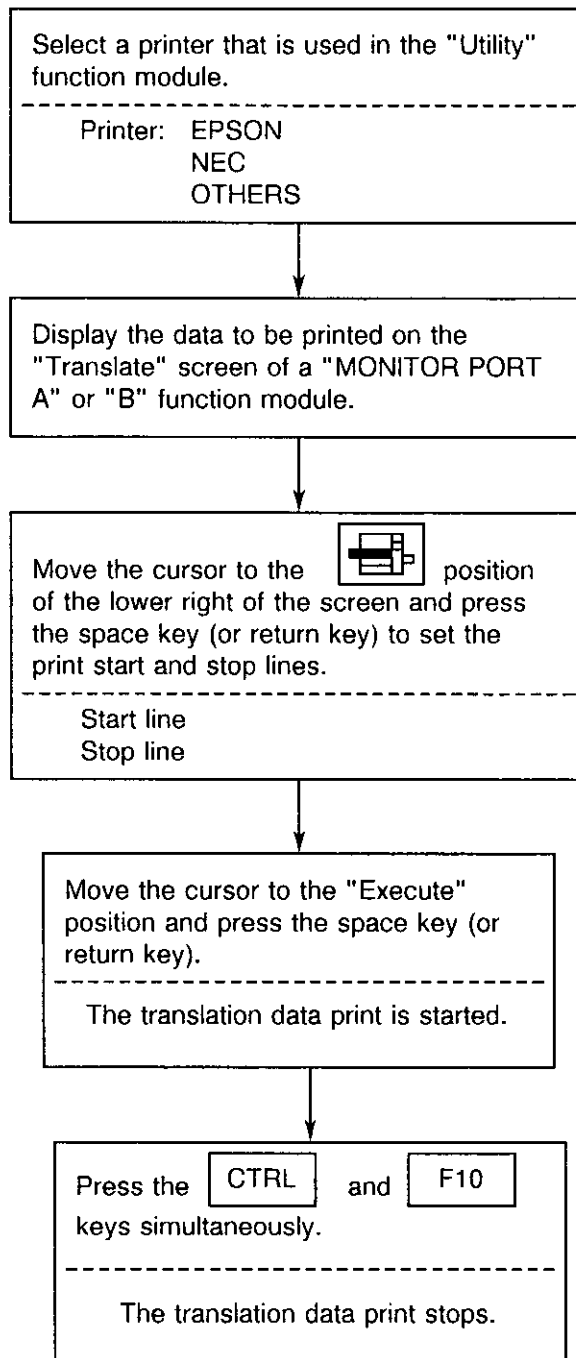
The screen can be printed at any time.



Note: For OTHERS, the screen cannot be printed.

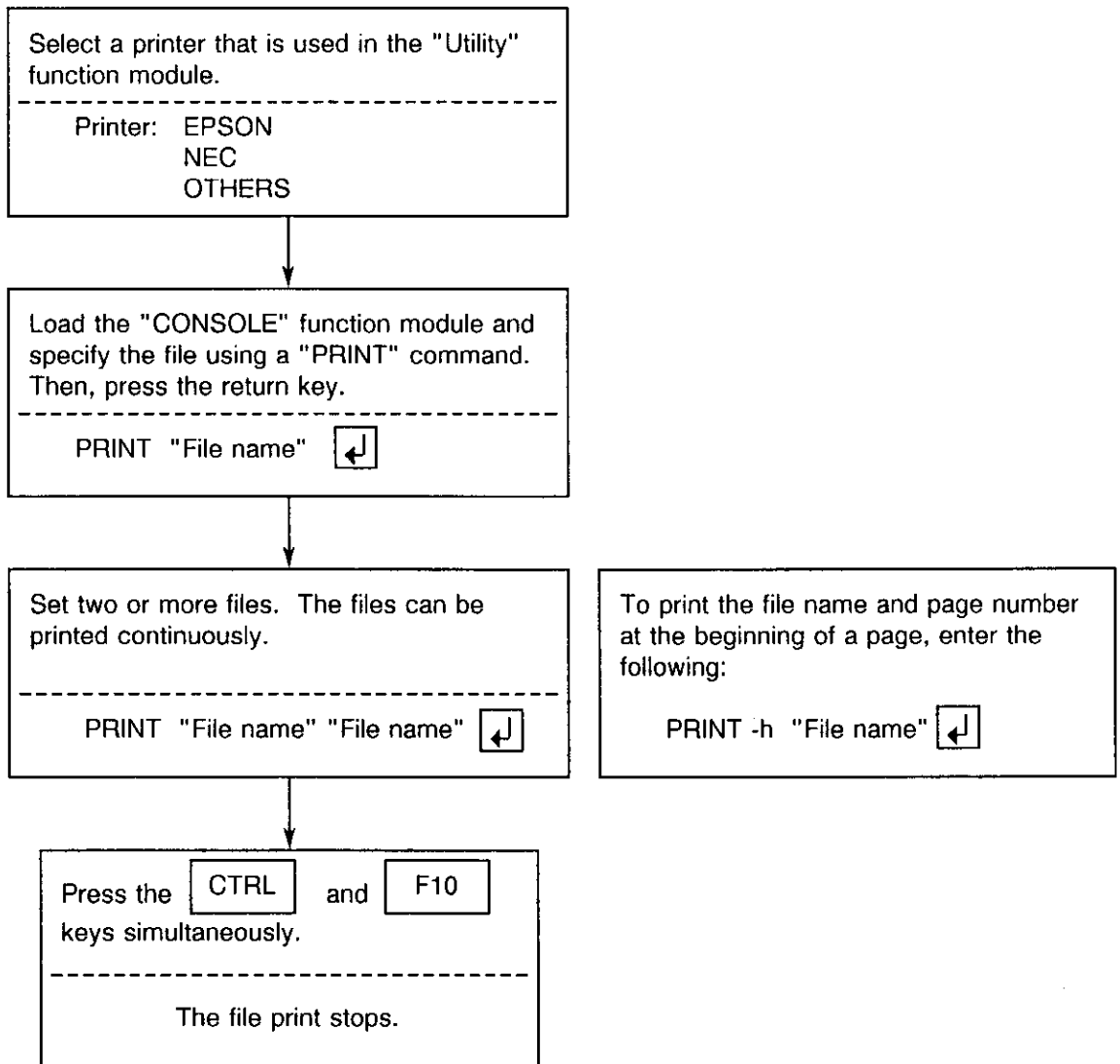
### 7.1.2 Translation data print

For translation data, only the number of lines to be required can be printed. You can stop the print during print. However, the data buffered in a printer is printed, so the stop operation requires a little time.



### 7.1.3 Print on console

An MS-DOS file is output for print from the console. All files are then printed.





## 7.2 Hard Disk

### 7.2.1 System installation

This D5111B has a system program installed in the hard disk at the factory. Therefore, it can be used immediately the power is turned on. However, reinstall the system program if a system program in the disk is destroyed for some reason or if an I/O error message is often displayed when a file in the hard disk is read or written. How to load a system program from the floppy disk to the hard disk is explained below.

- ① Insert a supplied D5111B system disk (PD511100-PJ) into the floppy disk drive and turn on the power. → The "SYSTEM CONFIGURATION" screen is displayed when the system is started.
- ② Press the F1 key to display the function module selection menu and select "CONSOLE." → The system enters the command wait state.
- ③ Enter "HFORMAT" to initialize the hard disk. Press the return key; message "Y/N" is displayed to confirm the initialization. Press the return key again. Message "LABEL?" is displayed. Enter a proper name (not exceeding 11 characters) and press the return key. The message below is then displayed.

"[SD0] Format then hit RETURN KEY (Abort SPACE key)"

- ④ Press the return key. Message "FORMATING..." → "finished" is displayed, and the hard disk format is completed (about 3 minutes and 45 seconds).
- ⑤ Execute the command below on the console screen.

BATCH FD0:INSTALL

- ⑥ Installation of the system program in the hard disk is completed.

Caution

All files in the hard disk are destroyed when "HFORMAT" is entered and executed.



## Appendix 1 ERROR MESSAGE LIST

Error message	Description
bad attribute type	A system file, volume label, and subdirectory cannot be opened.
bad file revision	The system revision of a file differs.
bad file type    type = [xxx]	The type of a file differs. The file type is displayed in [xxx].
bad port type    type = [xxx]	The port type of a file differs. The port type of the file is displayed in [xxx].
BINARY entry field (use 0, 1 key)	Field in which a binary numeric is entered. Use <input type="text" value="0"/> , and <input type="text" value="1"/> keys.
BINARY entry field (use 0, 1, X key)	Field in which a binary numeric is entered. Use <input type="text" value="0"/> , <input type="text" value="1"/> , and <input type="text" value="X"/> keys.
Buffer is being displayed	One buffer is killed. (Editor)
Buffer is read-only	Data is written in the read-only buffer. (Editor)
Bug: lost mark	The mark set position is deleted. (Editor)
Busy search !	Translation display data is being searched. (Monitor)
Busy Translate !	Data is being translated.
can't modify directory	No data can be written in the directory.
Cannot compile for no memory	Data cannot be compiled because there is no memory. (Editor)
Cannot create compiler	A compiler cannot be activated. (Editor)
Cannot create edit buffer	No memory exists. (Editor)
Cannot create window because of no memory	A window cannot be created because there is no memory. (Editor)
Cannot edit because of no memory	Memory is lost during edit. (Editor)
Cannot load compiler	A compile cannot be loaded. (Editor)
Cannot open file for reading	A file cannot be opened for reading. (Editor)

D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL

Appendix 1 ERROR MESSAGE LIST

Error message	Description
Cannot open file for writing	A file cannot be opened. (Monitor)
Cannot open file for writing	A file cannot be opened for writing. (Editor)
Cannot split a __ line window	A small window is split. (Editor)
Compiling other	Other programs are compiled during compile. (Editor)
Couldn't goto line	A constant is specified incorrectly. (Editor)
data end	A replay is complete.
Data Not Exist! Can't Saving.	Trace data cannot be saved because there is no data. (Monitor)
Data not Exist! Can't Print	Data cannot be printed because it does not exist. (Monitor)
DECIMAL entry field (use 0 to 9 key)	Field in which a decimal numeric is entered. Use <input type="text" value="0"/> through <input type="text" value="9"/> keys.
device not found	No corresponding device exists.
Directory not found	No corresponding directory exists.
disk full	Up to the last sector of a hard disk is recorded.
DRIVE NOT READY !   retry-SPACE continue-ESC	A floppy disk is not ready for operation. After preparation (media insertion), press the space key to retry and press the <input type="text" value="ES"/> key to ignore and continue.
Extention too long	An extension is too long.
File has long line	The one line read from a file exceeds the maximum number of characters. (256 characters) (Editor)
File is read-only	A file to be written is read-only mode. (Editor)
file not found	No corresponding file exists.
File not found	A specified file or director cannot be found.
File read error	A file cannot be read.

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**Appendix 1 ERROR MESSAGE LIST**

Error message	Description
Frame not found	A frame cannot be searched.
Help data file NOT FOUND	A help menu cannot be displayed because no specified help message file exists. (On this screen, the help message is not supported.)
HEXADECIMAL entry field (use 0 to 9, A to F key)	Field in which a hexadecimal numeric is entered. Use <input type="text" value="0"/> through <input type="text" value="9"/> and <input type="text" value="A"/> through <input type="text" value="F"/> keys.
Hit and key!	Press any key.
Illegal entry	Invalid keys are entered.
Impossible change	The scaling in a window reaches the limit. (Editor)
Invalid character	The character cannot be used.
Invalid Frame No.	The frame number is set incorrectly. (Monitor)
Invalidate for small screen size	A popup menu is displayed when the window width is small. (Editor)
layer 1 error	An error occurs during real-time recording and replay. This operation can be performed no longer.
Line length is long	Characters exceeding the maximum number of characters in one line are entered. (Editor)
LOAD ERROR (XXX)	An error occurs when a function module (XXX) is loaded because of a disk read error or insufficient memory capacity.
Loading Now!	In-loading (Monitor)
Maximum of total length is 512 octets	The message length reaches 512 octets. (Message builder)
MEDIA CHANGED? please check the DISK, then hit any key	A floppy disk may have been replaced. Check the disk and press any key for confirmation. This message is displayed when the continued processing may destroy the disk.

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**Appendix 1 ERROR MESSAGE LIST**

Error message	Description
Monitor Running! Can't Loading.	Trace data cannot be loaded because a monitor is operating. (Monitor)
Monitor Running! Can't Print	Print cannot be performed because a monitor is operating. (Monitor)
Monitor Running! Can't Saving.	Trace data cannot be saved because a monitor is operating. (Monitor)
Monitor Running!	A monitor is operating. (Monitor)
No buffer	A frame moves to the line exceeding the maximum number of lines in the buffer. (Editor)
No changes	The content is the same as during previous saving. (Editor)
no data	Data to be replayed does not exist.
no disk space	A disk has no space to save a file.
No file name	The file name of a buffer to be saved does not exist. (Monitor)
No Frame!	A frame to be monitored does not exist. (Monitor)
No mark in this window	A frame moves to the mark position without setting a mark. (Editor)
No mark set in this window	No mark is set when a region function is used. (Editor)
No more (XXX)	A function module (XXX) can be loaded no longer. Two of a monitor A/B, editor, and console can be loaded. For other modules, only one module can be loaded.
No more applications	No more function modules can be loaded. The number of function modules that can be loaded at the same time is limited by the display memory. The number of modules is 9 to 10 (varies depending on the combination of the module to be loaded).

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

**Appendix 1 ERROR MESSAGE LIST**

Error message	Description
No more errors	A frame moves to the error line in spite of the absence of an error. (Editor)
No Port!	A specified port does not exist. (Monitor)
No such file or directory	A specified file or directory cannot be found.
not D5111 file	The file is not a D5111B save/load file.
Not found	No character string to be retrieved exists. (Editor)
Not implemented	No execution
Not now	A macro is executed without keyboard macro definition. (Editor)
Only one window	Other windows are activated in spite of only one window. (Editor)
Option error	The argument during compile activation is not correct. (Editor)
printer active now	The screen output overlaps with a file and translation output. (Printer)
printer error	An error occurs in the printer. (Printer)
printer not ready	Power is off and paper is out. (Printer)
PROTOCOL mismatch! xxx data loading now	A file protocol differs, but the function module is loaded. The file protocol is displayed in [XXX].
read only volume	A disk is write protected.
root directory full	A root directory has no space to save a file.
SDO: Not Active	No hard disk can be used. No real-time recording and replay can be performed.
Soon can't allocate!!	Alarm message that memory is lost soon. (Editor)
Stop compile	A compiler is stopped using the CTRL-G key. (Editor)

D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL

Appendix 1 ERROR MESSAGE LIST

Error message	Description
Stop compile for compiler error	A compiler stops due to an error during compile. (Editor)
Stop compile for no file	No compile can be performed because there is no specified file. (Editor)
Stop compile for no memory	Memory is lost during compile. (Editor)
Too large region	A large region is specified. (Editor)
too many dots	Excessive dots exist in the file name.
Type SPACE or RETURN key to SET	Pressing the space key or return key enables setting.
Type SPACE or RETURN key	Field in which the space key or return key is entered
Unknown module ID (nn)	The function module with specified number (nn) is not registered in DISPMNGR.
Unknown module name (XXX)	The function module with specified number (XXX) is not registered in DISPMNGR.
valid keys: SPACE, # @, -, 0 to 9, A to Z	An invalid key is entered in the message name field. (Message builder)
Write I/O error	A file cannot be written. (Editor)
Write I/O error.	Any error occurs in a file during write. (Monitor)
XXX can't execute (NO PORT)	A function module (XXX) cannot be executed because the corresponding communication port (hardware) is not installed.
XXX can't execute (NO PROTOCOL TABLE)	A function module (XXX) cannot be executed because there is no data table of the corresponding protocol.
[xxx] open error	File open error during file output. (xxx indicates the file name.)

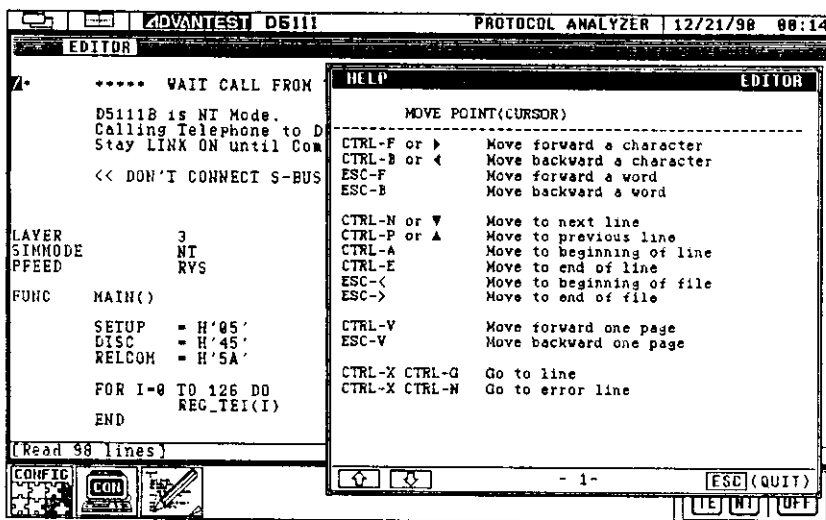


## Appendix 2 HELP MENU

The HELP menu are provided in the EDITOR and Simulator Port A menu in the D5111B.

- EDITOR

To press the HELP key displays the HELP menu that operates by using the CTRL key.



To press the HELP key displays the HELP menu display. The page changes to press the ↑ and ↓ keys.

Figure A2 - 1 HELP-1 in the EDITOR menu

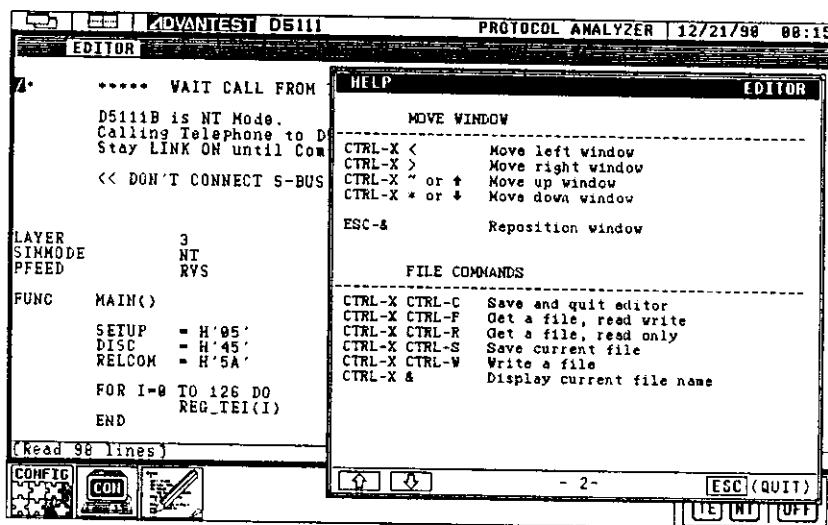


Figure A2 - 2 HELP-2 in the EDITOR menu

**D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL**

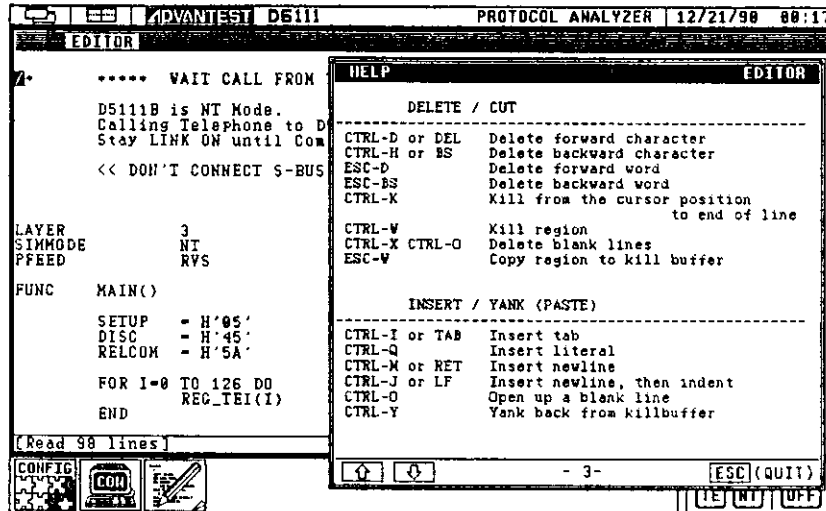


Figure A2 - 3 HELP-3 in the EDITOR menu

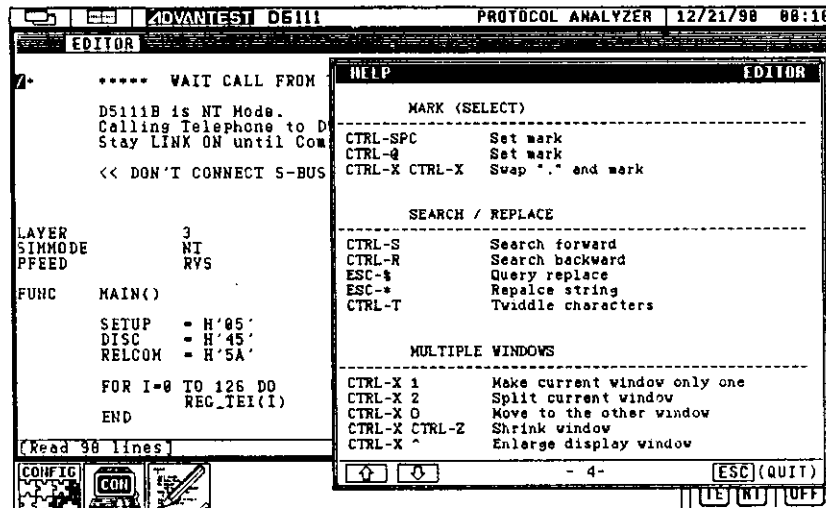


Figure A2 - 4 HELP-4 in the EDITOR menu

**D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL**

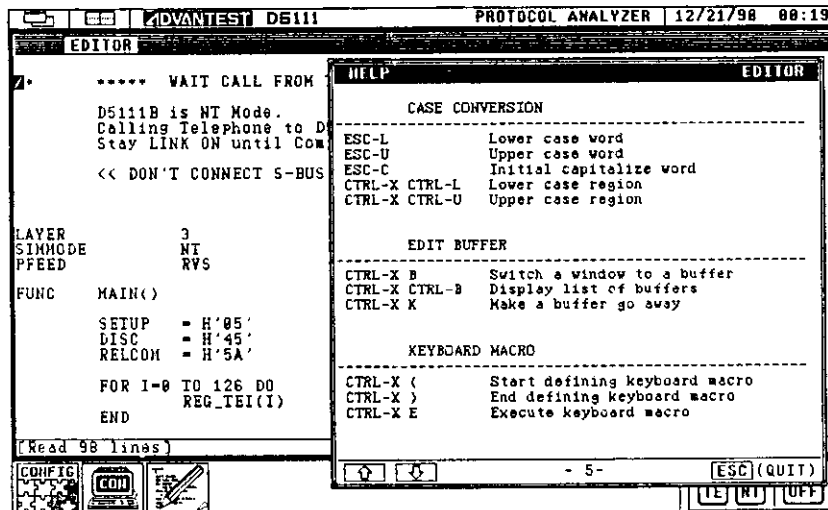


Figure A2 - 5 HELP-5 in the EDITOR menu

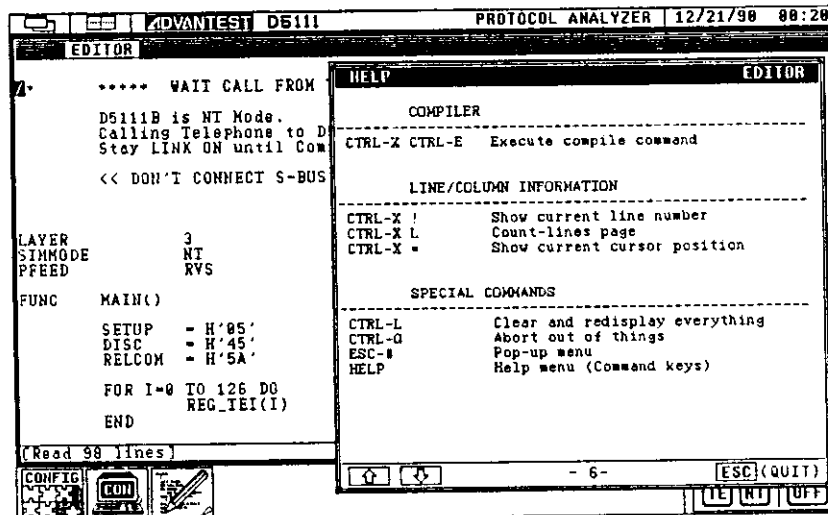


Figure A2 - 6 HELP-6 in the EDITOR menu

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

- Simulator Port A

To press the **HELP** key displays the use of a common by using the console command of a simulation program.

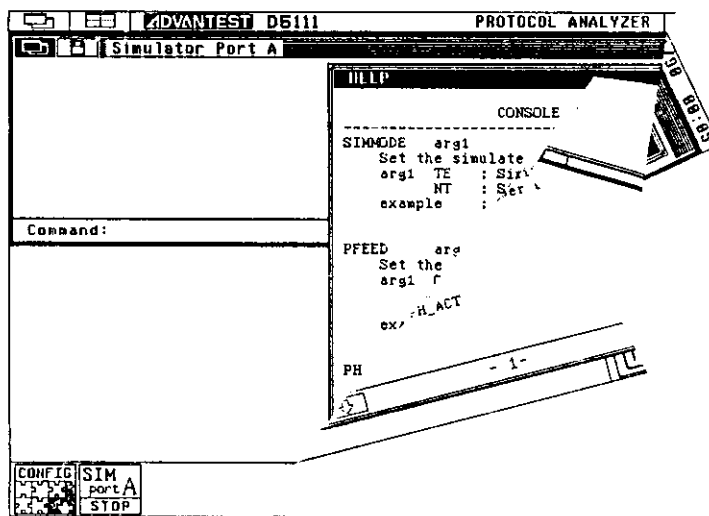


Figure A2 - 7 HELP-1 in the Simulator Port A menu

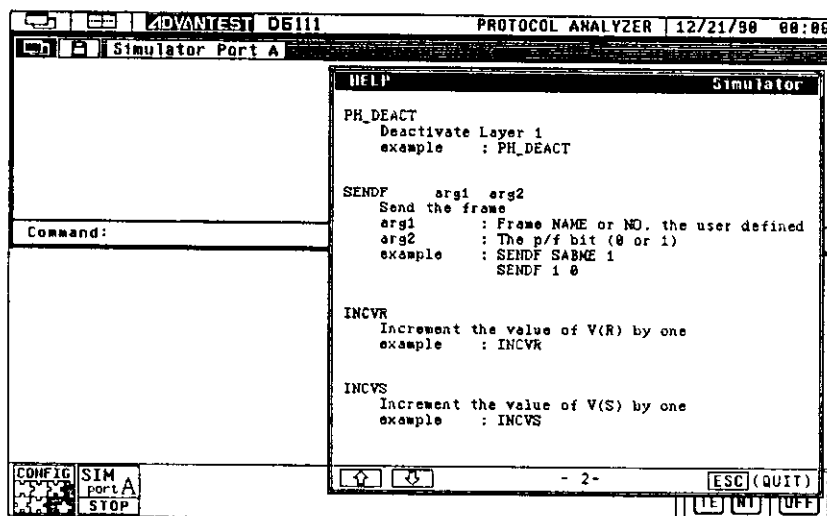


Figure A2 - 8 HELP-2 in the Simulator Port A menu

D5111B  
ISDN PROTOCOL ANALYZER  
INSTRUCTION MANUAL

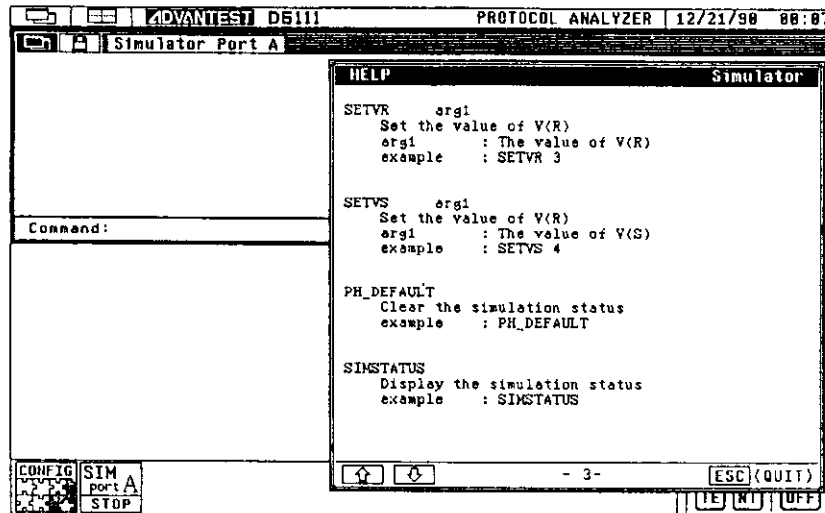


Figure A2 - 9 HELP-3 in the Simulator Port A menu

- \* The HELP menu can be scrolled to press the  ,  keys. The HELP menu is deleted to repress the  key (or  key).

*MEMO*



A large, empty rectangular area with rounded corners, enclosed by a thin black border, intended for writing the memo's content.

## ALPHABETICAL INDEX

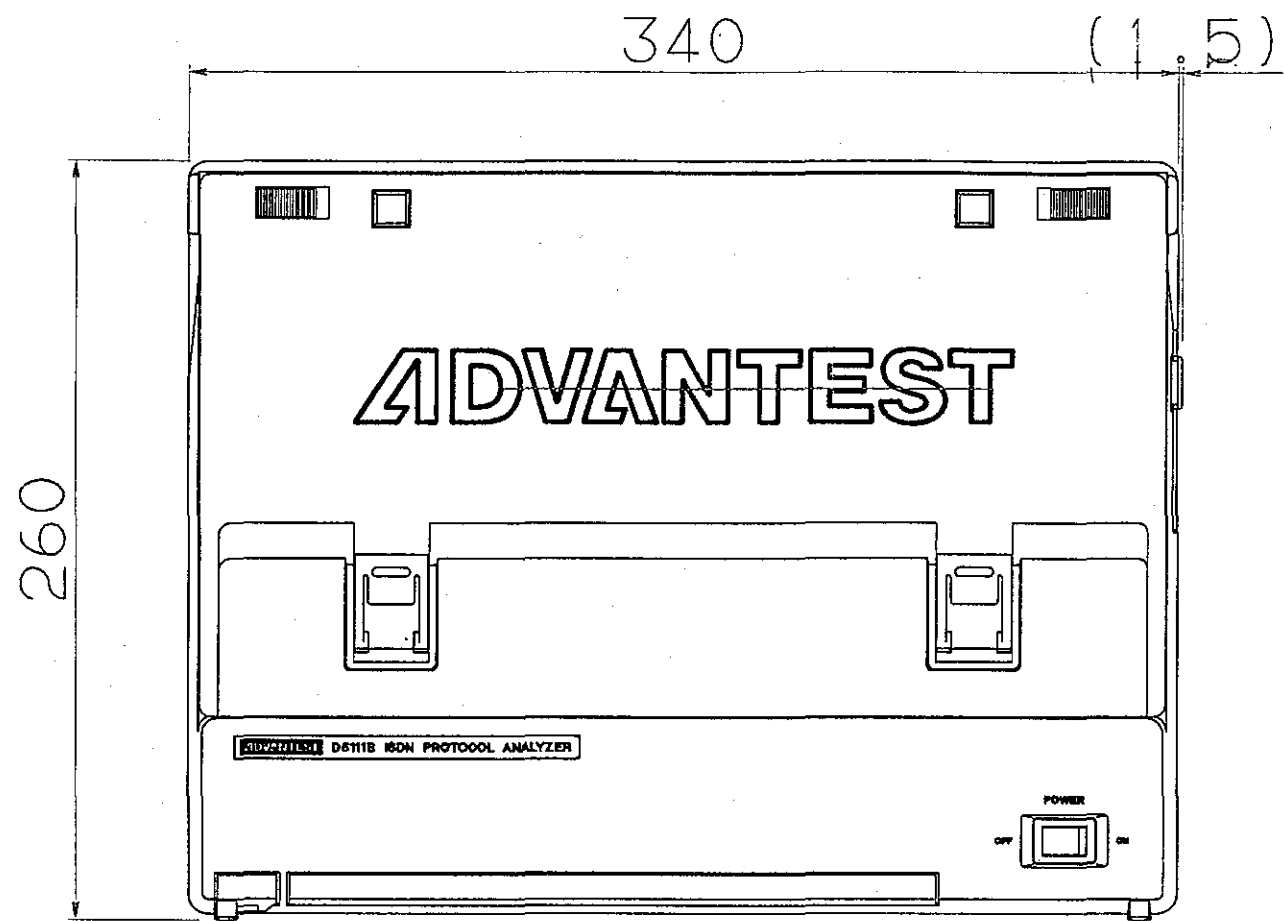
<p style="text-align: center;">[A]</p> <p>acquisition ..... 2-5</p> <p>active window ..... 1-23</p> <p>Active window display selection area .. 1-20</p>	<p style="text-align: center;">[F]</p> <p>Feeding ..... 1-42</p> <p>Feeding ..... 1-43</p> <p>Feeding polarity display ..... 1-43</p> <p>Feed status display area ..... 1-20</p> <p>File save/load selection area ..... 1-20</p> <p>Frame number search ..... 2-15</p> <p>function keys ..... 1-22</p> <p>function module icons ..... 1-22</p> <p>Function module icon display area .... 1-20</p> <p>Function module name ..... 1-19</p> <p>Function module selection ..... 1-22</p> <p>Function module selection area ..... 1-20</p> <p>Function module title display area ..... 1-20</p>
<p style="text-align: center;">[C]</p> <p>Calendar timer display area ..... 1-20</p> <p>Channel selection ..... 2-3</p> <p>Compile ..... 1-34</p> <p>Compile ..... 3-6</p> <p>Complier ..... 4-8</p> <p>Configuration setting ..... 2-2</p> <p>Console ..... 5-1</p> <p>cursor (underlined) ..... 1-30</p>	<p style="text-align: center;">[D]</p> <p>data read ..... 1-28</p> <p>Data search mode selection area ..... 1-20</p> <p>date ..... 5-7</p> <p>delay ..... 2-6</p> <p>Disk I/O related Command ..... 5-2</p> <p>Display position and size ..... 1-24</p>
<p style="text-align: center;">[E]</p> <p>EDITOR ..... 4-1</p> <p>Editor Activation and Release ..... 4-2</p> <p>Editor Command ..... 4-5</p> <p>editor's function selection menu ..... 1-32</p> <p>Environment Set Command ..... 5-7</p> <p>Error search ..... 2-18</p> <p>Execution of Simulation Function without Program ..... 3-30</p> <p>EXTERNAL CONTROL ..... 6-1</p> <p>extension passive bus ..... 1-43</p>	<p style="text-align: center;">[H]</p> <p>HARD DISK ..... 7-1</p> <p>History Display ..... 2-15</p>
	<p style="text-align: center;">[L]</p> <p>layer-2 automatic mode ..... 3-3</p> <p>Layer-2 full (LAY2 FULL) translation display ..... 2-23</p> <p>Layer-3 full (LAY3 FULL) translation display ..... 2-23</p> <p>Layer-23 full (LAY23 FULL) translation display ..... 2-24*</p> <p>Layer-2 (LAY2) translation display .... 2-20</p> <p>Layer-3 (LAY3) translation display .... 2-21</p> <p>Layer-23 (LAY23) translation display ... 2-22</p> <p>Layer-1 status display area ..... 1-20</p>

**D5111B**  
**ISDN PROTOCOL ANALYZER**  
**INSTRUCTION MANUAL**

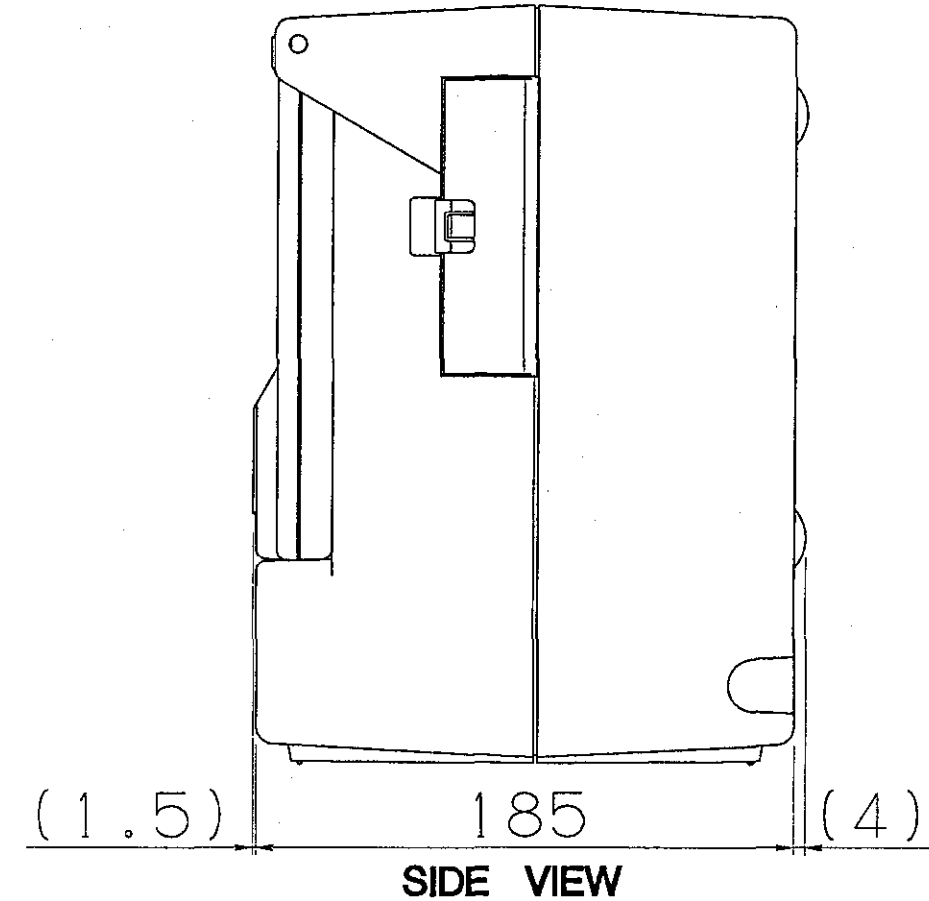
**Alphabetical Index**

<b>[M]</b>		<b>[S]</b>	
Maximum frame length setting . . . . .	2-3	Save the program file . . . . .	1-34
Memory mode setting . . . . .	2-3	Screen print . . . . .	7-2
message builder . . . . .	3-16	self diagnosis . . . . .	1-27
message builder . . . . .	3-17	Short-distance passive bus . . . . .	1-43
Message creation . . . . .	3-17	Simulation . . . . .	1-31
Message data . . . . .	3-28	simulation function . . . . .	3-1
monitor starts . . . . .	2-14	Simulation Language . . . . .	3-2
monitor stops . . . . .	2-14	Specification statement	
		of simulation . . . . .	3-3
<b>[O]</b>		Start/stop state display . . . . .	1-43
Object program . . . . .	3-26	Start-Up File . . . . .	1-36
		Subscreen . . . . .	1-19
<b>[P]</b>		Subscreen selection area . . . . .	1-20
Pattern search . . . . .	2-16	Syntax of Simulation	
Popup menu . . . . .	1-23	Language (PSL51) . . . . .	3-32
point-point . . . . .	1-43		
Preparation for Monitoring . . . . .	2-2	<b>[T]</b>	
PRINTER . . . . .	7-1	Translation data print . . . . .	7-3
Program execution . . . . .	3-7	Translation display . . . . .	1-27
Program compile . . . . .	3-10	Translation display format	
Program creation . . . . .	3-9	selection area . . . . .	1-20
Printer Output . . . . .	7-2	Translation layer selection . . . . .	2-13
Print-out specification area . . . . .	1-20	Translation screen . . . . .	1-28
Program read . . . . .	1-33	Translation Screen . . . . .	2-19
Protocol selection . . . . .	2-3	Translation screen setting . . . . .	2-13
PSL51 . . . . .	1-44	transparent mode . . . . .	3-3
PSL51 language specification . . . . .	3-2	Time search . . . . .	2-15
PSL51 programming structure example . . . . .	3-3	trigger function . . . . .	2-5
		Trigger search . . . . .	2-15
<b>[R]</b>			
Real-Time Display . . . . .	2-14		
Real-time trace . . . . .	2-14		
Remote Control . . . . .	6-2		
reserved word . . . . .	3-42		

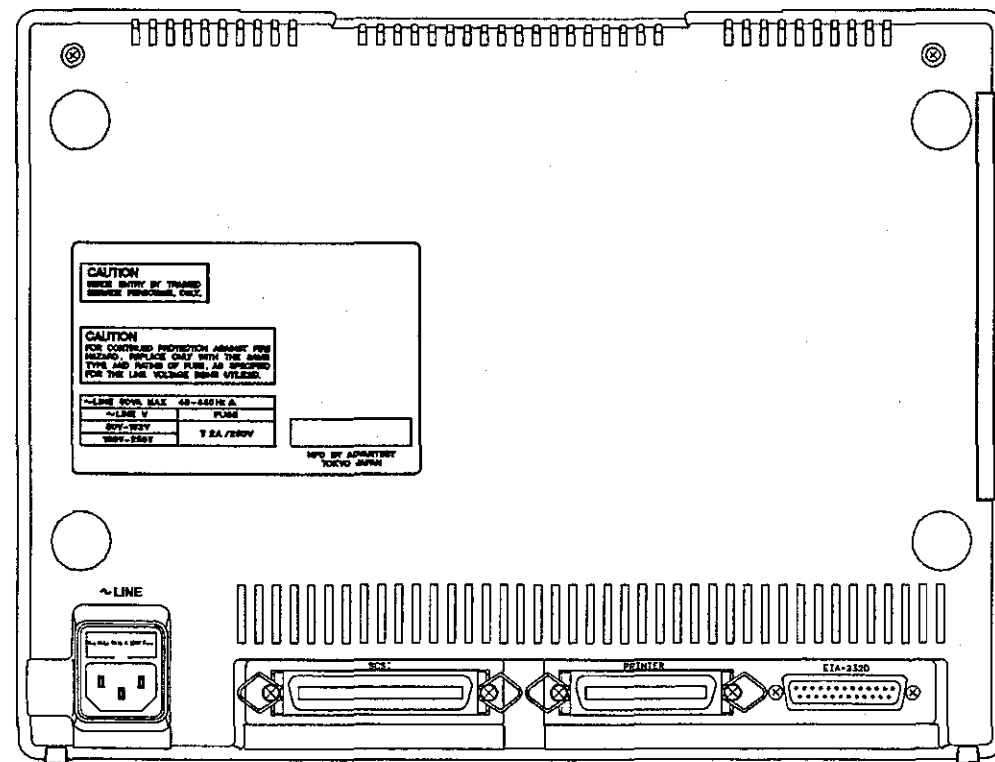




FRONT VIEW



SIDE VIEW

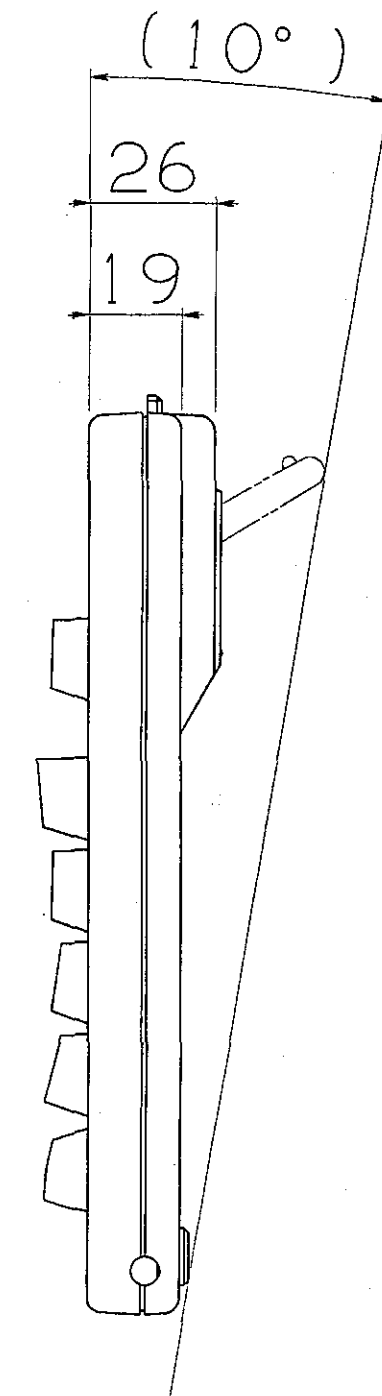
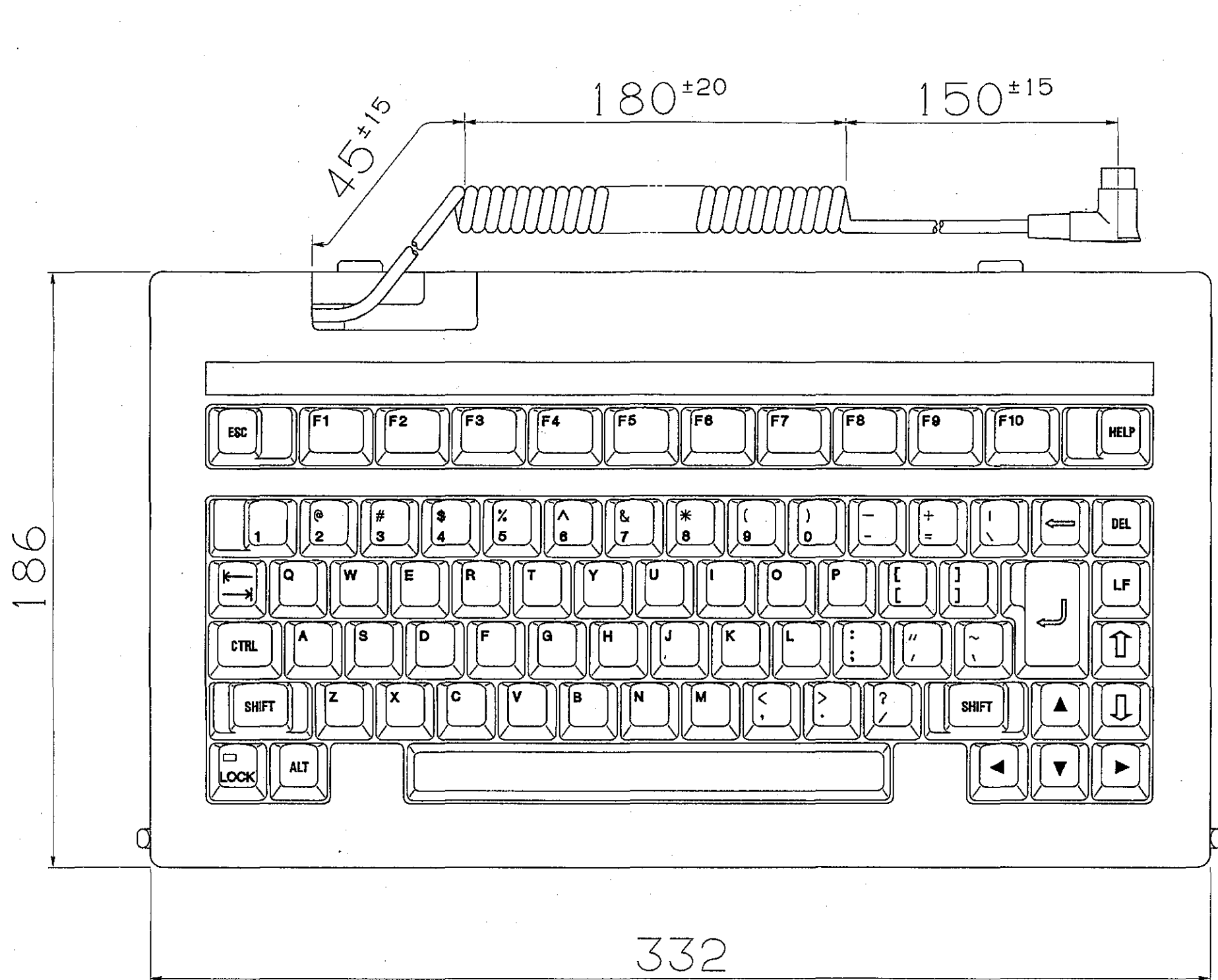


REAR VIEW

Unit : mm

D5111BEXT1-9009-A

D5111 B  
EXTERNAL VIEW



Unit : mm



D5111BEXT2-9009-A

**D5111B**  
**KEYBOARD**  
**EXTERNAL VIEW**

## **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 its 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, NEGLIGENCE), 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>