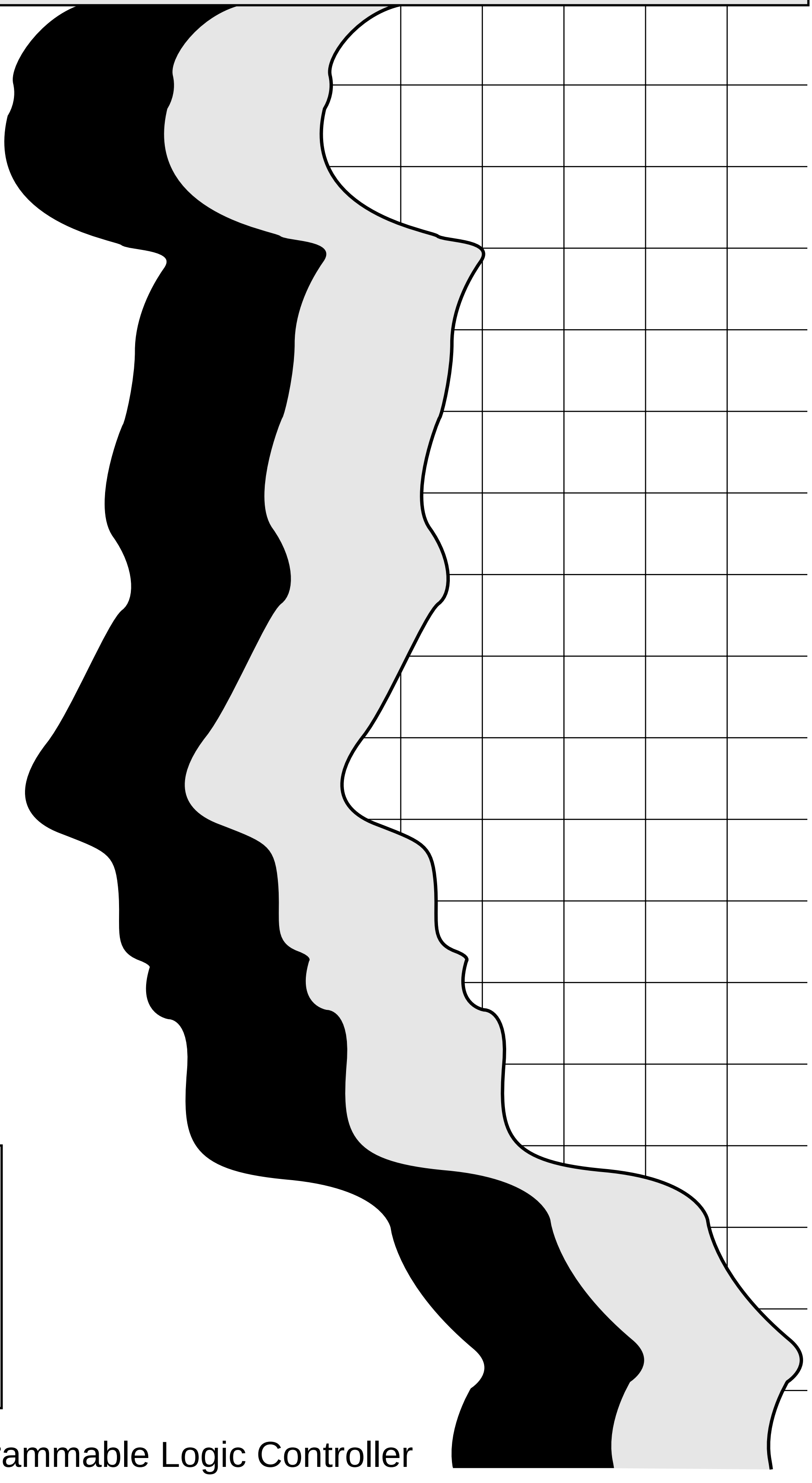
MITSUBISHI

Intelligent Communication Module Type AD51H-S3

User's Manual



A

MELSEC

Mitsubishi Programmable Logic Controller

• SAFETY PRECAUTIONS •

(Always read these instructions before using this equipment.)

Before using this product, please read this manual and the relevant manuals introduced in this manual carefully and pay full attention to safety to handle the product correctly.
The instructions given in this manual are concerned with this product. For the safety instructions of the programmable controller system, please read the CPU module user's manual.
In this manual, the safety instructions are ranked as "DANGER" and "CAUTION".

◇ DANGER	Indicates that incorrect handling may cause hazardous conditions, resulting in death or severe injury.
⚠ CAUTION	Indicates that incorrect handling may cause hazardous conditions, resulting in medium or slight personal injury or physical damage.

Note that the ⚠CAUTION level may lead to a serious consequence according to the circumstances. Always follow the instructions of both levels because they are important to personal safety.

Please save this manual to make it accessible when required and always forward it to the end user.

[DESIGN PRECAUTION]

◇ DANGER

- Please refer to the manual for each data link concerning the operating status of each station when communications errors occur in the data link. There is danger of accidents due to wrong outputs or wrong operations.
- When connecting a peripheral device to the CPU module or performing control of a PLC which is being run through a BASIC program, configure an interlock circuit in the sequence program so that the system overall is operating on the safe side at all times. Also, when performing other control (program modifications, changing the operating status (status control)) of a PLC that is currently running, read the manual thoroughly and proceed only after taking adequate safety precautions. Particularly in cases where the above control is performed toward PLC in remote locations from the opposite party's device, there may be occasions where it is impossible to respond immediately to trouble on the PLC side resulting from data communications errors. At the same time as you are configuring an interlock circuit in the sequence program, decide methods for the system to use to treat data communications errors between the opposite party's device and the PLC CPU.

[DESIGN PRECAUTION]

DANGER

- Do not write data to the "System Area" in the intelligent function module's buffer memory.
  Also, do not turn the output for signals for "Use Prohibited" signals among the output signals from the PLC CPU to the intelligent function module (ON).
  If data are written to the "System Area" or output in response to "Use Prohibited" signals, there is danger that the PLC system will malfunction.

CAUTION

- Do not bunch the control wires or communication cables with the main circuit or power wires, or install them close to each other. They should be installed 100mm (3.94 in.) or more from each other.
  Not doing so could result in noise that may cause malfunction.
- If a BASIC program is registered in the EEP-ROM in the module, do not perform a power OFF or PLC CPU reset operation at the station where the module is installed during registration.
  If a power OFF or PLC CPU reset operation is performed at the station where the module is installed during registration, the data contents in the EEP-ROM will be indefinite and it will be necessary to reset the setting values, etc. in the buffer memory and register them again in the EEP-ROM. It could also cause the module to break down or malfunction.

[INSTALLATION PRECAUTIONS]

CAUTION

- Use the PLC in an environment that meets the general specifications contained in the user's manual of the CPU moudle to use.
  Using this PLC in an environment outside the range of the general specifications may cause electric shock, fire, malfunction, and damage to or deterioration of the product.
- Insert the tabs at the bottom of the module into the holes in the base unit before installing the module.
  Improper installation may cause erroneous operation, accidents, or the module to fall out.
  When using the PLC in the environment of much vibration, tighten the module with a screw.

[INSTALLATION PRECAUTIONS]

⚠ CAUTION

- Tighten the screws within the range of specified torque.
  If the screws are loose, it may cause the module to fallout, short circuits, or malfunction.
  If the screws are tightened too much, it may cause damage to the screw and/or the module, resulting in fall out, short circuits or malfunction.
- Switch all phases of the external power supply off when mounting or removing the module.
  Not doing so may cause failure or malfunction of the module.
- Do not directly touch the conductive area or electronic components of the module.
  Doing so may cause malfunction or failure in the module.

[WIRING PRECAUTIONS]

⚠ CAUTION

- If the module is turned on and operated after installation or wiring operations, etc., be sure install the terminal cover provided with this product. If the terminal cover is not installed, there is danger of malfunction.
- When connecting wires to the connector used for external connections, be sure to crimp, pressure weld or solder the wires correctly using the tool specified by the manufacturer.
  If connections are not perfect, it could result in short circuits, fire or malfunction.
- Securely insatll the connector to the module.
- Be sure to fix communication cables leading from the module by placing them in the duct or clamping them.
  Cables not placed in the duct or without clamping may hang or shift, allowing them to be accidentally pulled, which may cause a module malfunction and cable damage.
- When connecting cables, be sure to do so correctly after confirming the type of interface you are connecting to. If connection is made to a different interface or if wiring is faulty, it could cause the module or external device to break down.
- Tighten the terminal screws within the range of specified torque.
  If the terminal screws are loose, it may result in short circuits or malfunction.
  If the screws are tightened too much, it may cause damage to the screw and/or the module, resulting in the module falling out, short circuits or malfunction.
- When disconnecting the communications cable or power cable that is connected to the module, do not disconnect it by grasping the cable with your hand and pulling it.
  Disconnect cables with connectors attached by taking hold of the connector at the connection with the module and pulling the connector. For cables connected to a terminal block, remove the cable after loosening the terminal block screws.
  If the cable is pulled while it is connected to the module, it could cause malfunction or damage the module or the cable.

[WIRING PRECAUTIONS]

⚠ CAUTION

- Be careful not to let foreign objects such as swarf or wire chips get inside the module. They may cause fires, failure or malfunction.

[STARTING AND MAINTENANCE PRECAUTIONS]

⚠ CAUTION

- Do not disassemble or modify the each module. Doing so could cause failure, malfunction injury or fire.
- Switch all phases of the external power supply off when mounting or removing the module. Not doing so may cause failure or malfunction of the module.
- Do not touch the connector while the power is on. Doing so may cause malfunction.
- Switch all phases of the external power supply off when cleaning or retightening terminal screws and module installation screws.
  Not doing so may cause failure or malfunction of the module.
  If the screws are loose, it may cause the module to fallout, short circuits, or malfunction.
  If the screws are tightened too much, it may cause damages to the screws and/or the module, resulting in fall out, short circuits or malfunction.
- Always make sure to touch the grounded metal to discharge the electricity charged in the electricity charged in the body, etc., before touching the module.
  Failure to do say cause a failure or malfunctions of the module.

[OPERATING PRECAUTIONS]

⚠ CAUTION

- When performing control (in particular, changing data, changing a program or changing the operation status (status control)) of the PLC during operation using a BASIC program, do so only after reading the user's manual thoroughly and taking adequate safety precautions.
  If there are errors when changing data, changing a program or in status control, it could result in system malfunction, or cause mechanical damage or accidents.

[DISPOSAL PRECAUTIONS]

⚠ CAUTION

- When disposing of this product, treat it as industrial waste.

REVISIONS

*The manual number is given on the bottom left of the back cover.

Print Date	*Manual Number	Revision
Jan., 1993	IB (NA) 66401-A	First edition
Sep., 2004	IB (NA) 66401-B	Addition About Manual, Section 3.5 to 3.15.2, WARRANTY AD51H STARTUP PROCEDURE, Section 6.5 Correction SAFETY INSTRUCTIONS, About Manual, Chapter 1, Section 1.1, 1.2, 2.1, 2.1.1, 2.1.2, 2.2, Chapter 3, Section 3.1.1, 3.2.1, 3.2.2, 3.2.3, 3.2.4, 3.2.5, 3.2.6, Section 3.3, 3.5, 3.7.3, 3.11.1, 3.12.3, 3.13.4, 3.14.1, Chapter 4, Section 4.1, 4.2, 4.3, 4.3.2, 4.4.1, 4.4.2, Section 4.5.4, 4.6, Chapter 5, Section 5.1, 5.2.2, 5.2.3, Section 5.3.1, 6.1, 6.2, 6.3, 6.4, APPENDIX 1, 2 Deletion Section 1.3 to 1.5.2, Chapter 7 to 14, APPENDIX 3, APPENDIX 4

Japanese Manual Version IB-68350-E

INTRODUCTION

Thank you for choosing the Mitsubishi MELSEC-A Series of General Purpose Programmable Controllers. Please read this manual carefully so that the equipment is used to its optimum. A copy of this manual should be forwarded to the end User.

CONTENTS

About Manuals

The following manuals are related to the AD51H intelligent communication module.

Related Manual

Manual name	Manual number (type code)
AD51H-BASIC Programming Manual (Command) **This manual explains the programming methods, commands, and error codes of the AD51H-BASIC. (Purchased separately.)**	SH-080090 (13JF63)
AD51H-BASIC Programming Manual (Program edit, Compile) **This manual explains how to use the system commands and debugging commands, and how to create and compile programs with a general-purpose editor. (Purchased separately.)**	SH-080091 (13JF64)
SW1IX-AD51HPE Operating Manual **This manual explains how to operate the software package for the PC/AT. (Packaged with the software package.)**	IB-66402 (13J733)

Conformation to the EMC Directive and Low Voltage Instruction

For details on making Mitsubishi PLC conform to the EMC directive and low voltage instruction when installing it in your product, please see Chapter 3, "EMC Directive and Low Voltage Instruction" of the User's Manual (Hardware) of the CPU module to use.

The CE logo is printed on the rating plate on the main body of the PLC that conforms to the EMC directive and low voltage instruction.

1. GENERAL DESCRIPTION

This manual explains the system configurations, performance specifications, and functions of the AD51H-S3 intelligent communications module (hereafter called the AD51H).

The AD51H has the following functions.

(1) Functions with BASIC programs

- Sub-CPU function: Performs complex calculations and functional calculus.
- Monitor display function: Displays production output status, operation status, and details of faults.
- Key input function: Allows the user to input data such as production schedules, production quantities, operations, and set values.
- Printer function: Allows the user to print data such as production plans, actual production, daily reports, details of faults, planning data, inspection results, and test records.
- Data input function: Allows the user to enter data by using devices such as bar-code readers and magnetic card readers.
- External device connection function: Allows data communications using a BASIC program, with the computer connected to the AD51H via the RS-232C or RS-422 interface.
- Clock function: Allows the user to access the clock data of a PC CPU.

(2) Offline programming function

Allows the user to create and modify programs, save created programs in the user's floppy disk, and print programs with a peripheral device alone.

(3) Online programming function

Allows the creation, execution, and modification of programs from a console.

(4) Multi-task debugging function

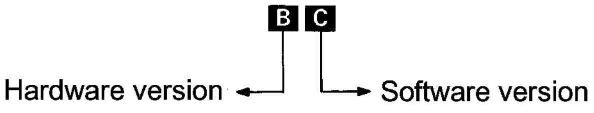
Allows debugging while a BASIC program is being executed.

CAUTION

The following functions are available for the AD51H whose software version is E or later.

- Access to AnUCPU (PCRD, PCWT instructions)
- Access to MELSECNET/10 (PCRD, PCWT instructions)
- Programming by line No. tool or general-purpose editor

The software version is indicated on the seal at the module front.



1.1 General Operation Description of the AD51H

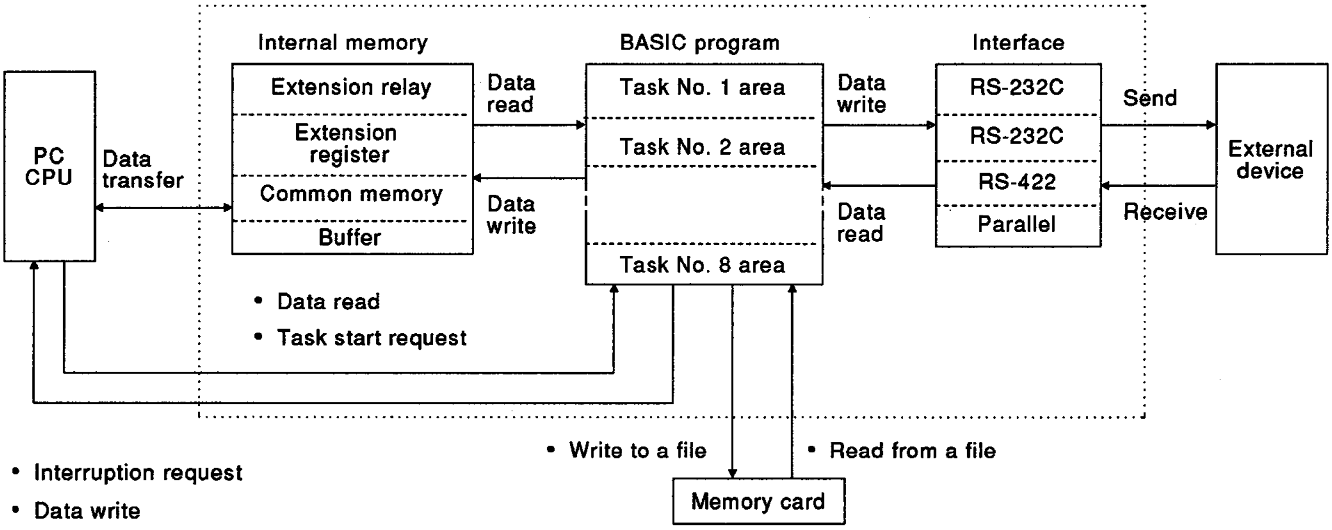
The AD51H operates with a BASIC program created by the user.

AD51H-BASIC can be used with the AD51H, and multitask processing of up to 8 tasks can be executed.

The following operations can be executed using created BASIC programs:

- Data communications with an external device
- Data transfer from/to a PC CPU
- File management using a memory card

The following shows operations when performing data communications with an external device, data transfer from/to a PC CPU, and file management using the AD51H:



(1) Data communications with an external device

The AD51H has two channels for the RS-232C interface, one channel for the RS-422 interface, one channel for the parallel interface, and two channels for the memory card interface.

Data of production orders and actual production output can, without using protocols, be sent/received to/from a console or an external device such as a personal computer (which is connected to an RS-232C or RS-422 interface).

In addition, actual production output data can be printed by a printer connected to the RS-232C, RS-422, or parallel interface.

(2) Data transfer from/to a PC CPU

Device data can be read/written from/to a PC CPU by using the PC access functions (PCRD/PCWT) which transfer data from/to a PC CPU.

In addition, data can be read/written from/to a special-function module by using the PC access functions.

(3) File management using a memory card

Programs and data can be read/written from/to memory cards installed in the memory card interface (MEMORY CARD 1 and 2) of the AD51H by using (a) file operation commands (LOAD, SAVE), and (b) file input/output commands (PRINT#, INPUT#).

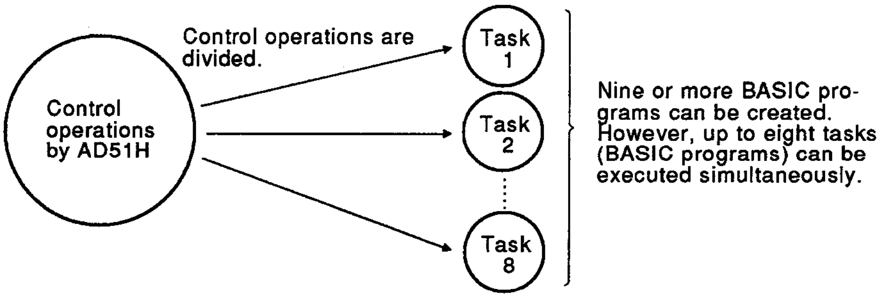
1.2 Features

The AD51H system has the following features:

(1) Up to eight BASIC programs (tasks) can be executed simultaneously.

Since multitask processing is enabled, BASIC programs can be created by dividing the control operations to be executed by the AD51H into the unit of processing.

Tasks required for each control operation can be started according to the contents of processing, and communications timing and execution between tasks can be controlled.


Control operations by AD51H
Control operations are divided.
Task 1
Task 2
Task 8
Nine or more BASIC programs can be created. However, up to eight tasks (BASIC programs) can be executed simultaneously.

(2) Data communications with a connected device is enabled.

<Data communications with an external device>

Data communications with an external device connected to an RS-422 or RS-232C interface can be controlled without using protocols.

<Data communications with a console and a terminal>

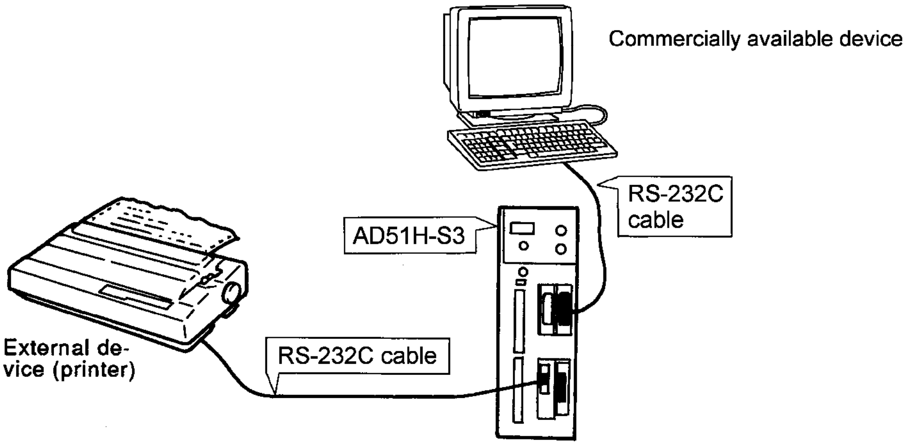
Data can be displayed on, and input through, a console and a terminal connected to an RS-422 or RS-232C interface.

A PC/AT or other devices available on the market can be used as a console or a terminal.

<Printouts using a printer>

Data can be printed out by using a printer connected to the RS-232C or parallel interface.

Printers available on the market can be used.



(3) Data communications with a PC is enabled.

Data communications can be executed from/to a PC CPU to which the AD51H is connected, or a PC CPU on MELSECNET(Ⅱ), MELSECNET/B, MELSECNET/10.

(a) The following data communications can be executed with a PC CPU:

1) Read and write of device data in the PC CPU (max. 64 words/time)

2) Read and write of buffer data in a special-function module (max. 128 words/time)

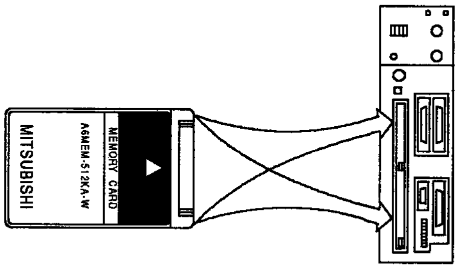
3) Remote RUN/STOP of a PC CPU

4) Read and write of a sequence program and a parameter

5) Interruption to a PC CPU

(4) File management is enabled.

Sequential files and random files can be managed by using an FD or HD of a PC/AT, or memory card(s) installed in the memory card interface of the AD51H.



When managing a file in a memory card, use the file area in the memory card.

[Sequential file]

A sequential file is efficient when reading or writing data sequentially.

This file is used mainly for work instruction data files and production output achievement files.

[Random file]

Data can be read/written from/to any part in this file.

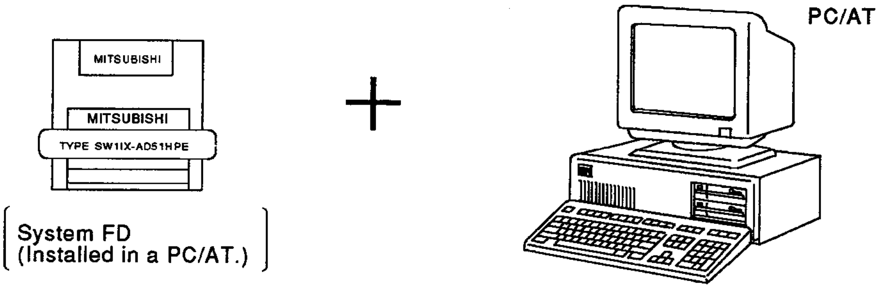
This file is suitable for storing large-capacity inventories, display data for the operator I/F, and master data.

The data size of a one-time read/write is (max.) 256 bytes.

(5) Offline programming using a PC/AT is enabled.

BASIC programs can be edited offline by using a PC/AT.

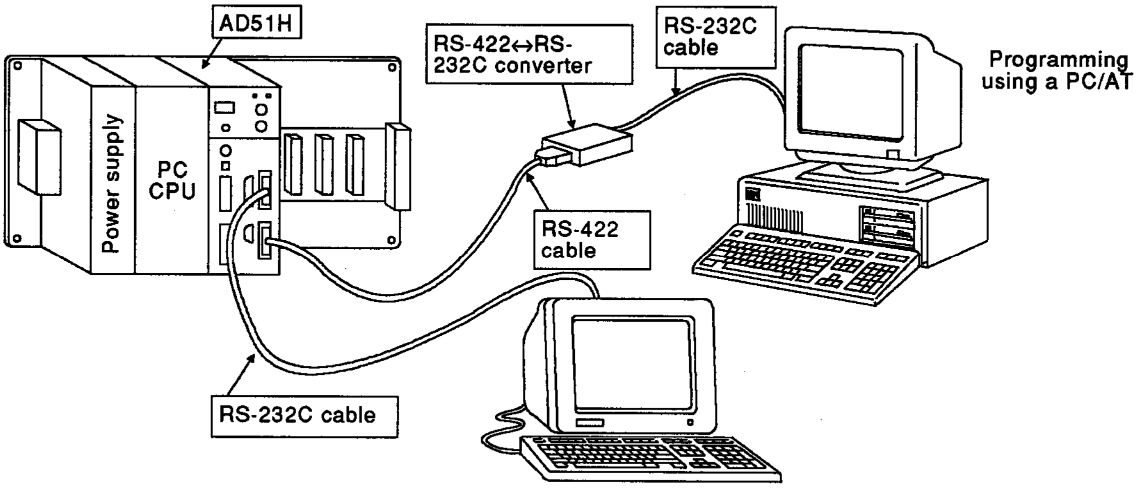
An edited program cannot be executed by using offline programming.



System FD (Installed in a PC/AT.)

(6) Online programming is enabled.

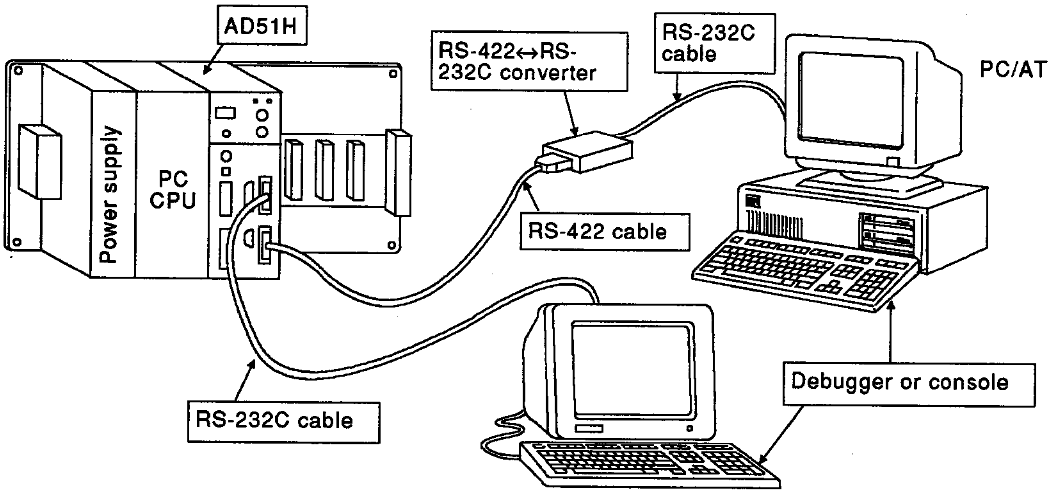
BASIC programs can be edited and debugged by connecting a PC/AT or device available on the market to the AD51H and setting it as the console.



Programming using a VT-382 or VG-620

(7) Multitask debugging is enabled.

Debugging during multitask processing can be executed by connecting a PC/AT or general-purpose device to the AD51H and setting it as the debugger.



VT- 382 or VG-620

Multitask debugging is executed by inputting a debug command from the debugger. The following operations can be executed by using a debug command:

- Executing and stopping the BASIC program in a specified task number area
- Read/write of the value of a specified variable from/to the BASIC program in a specified task number area
- Read/write of data from/to the internal memory of the AD51H
- Display of the state of use of events and message ports being shared between BASIC programs

(8) Both interpreter BASIC and compile BASIC can be used.

BASIC processing can be executed at high speed by compiling BASIC programs created by an interpreter BASIC by using a BASIC compiler.

Interpreter BASIC: Translates programming language into machine language one program statement at a time and then causes the AD51H-S3 to execute that program statement in machine language.

Compile BASIC : Translates programming language into machine language before the program is excuted. The AD51H executes programs in machine language.

MEMO

2. SYSTEM CONFIGURATIONS

This section explains the system configurations and system devices for use with the AD51H.

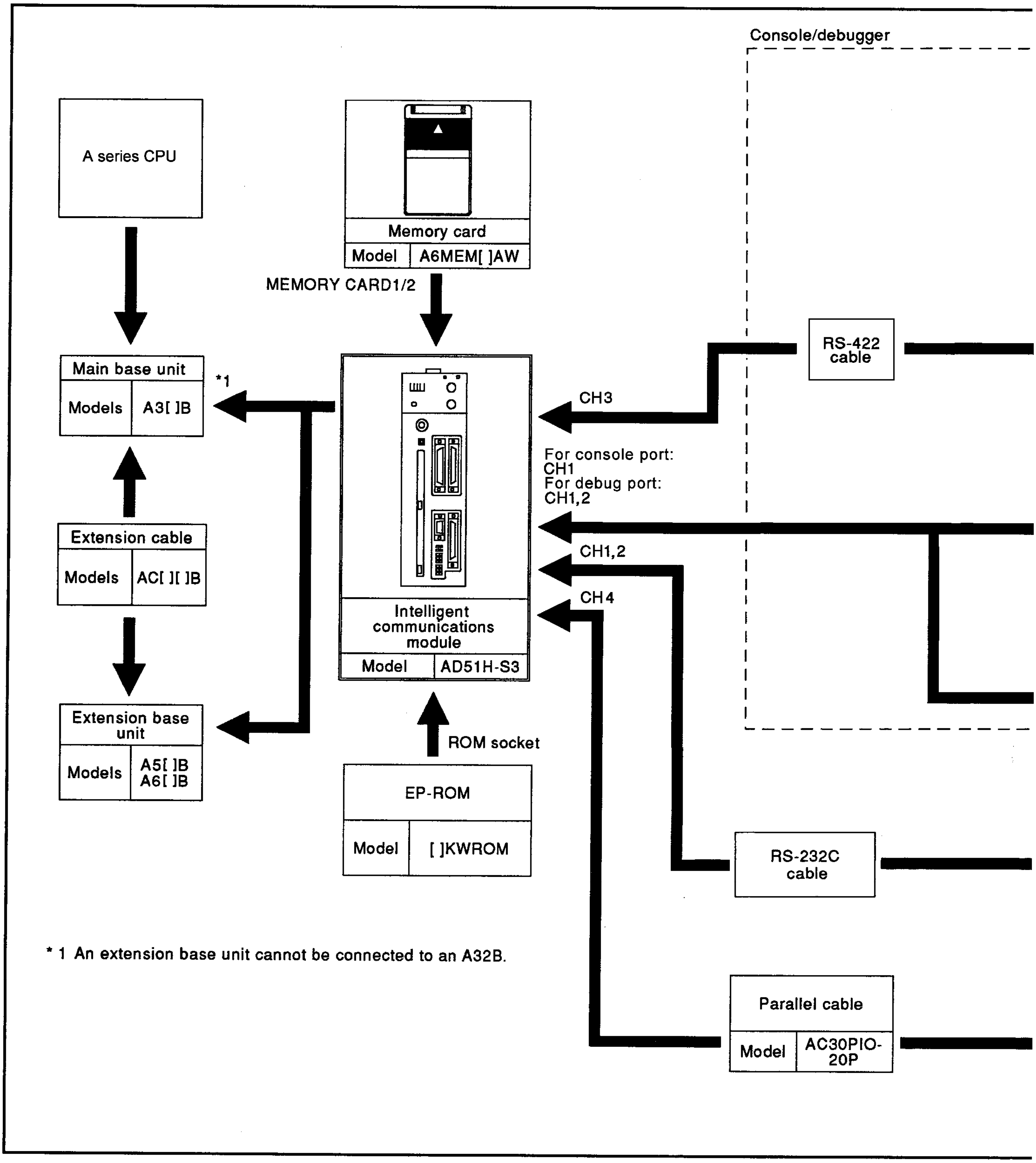
2.1 PC CPU Systems Used With the AD51H

This section explains PC CPU systems, applicable systems, and precautions which must be taken when using the AD51H.

Either a building block-type CPU or a compact-type CPU can be used as a PC CPU.

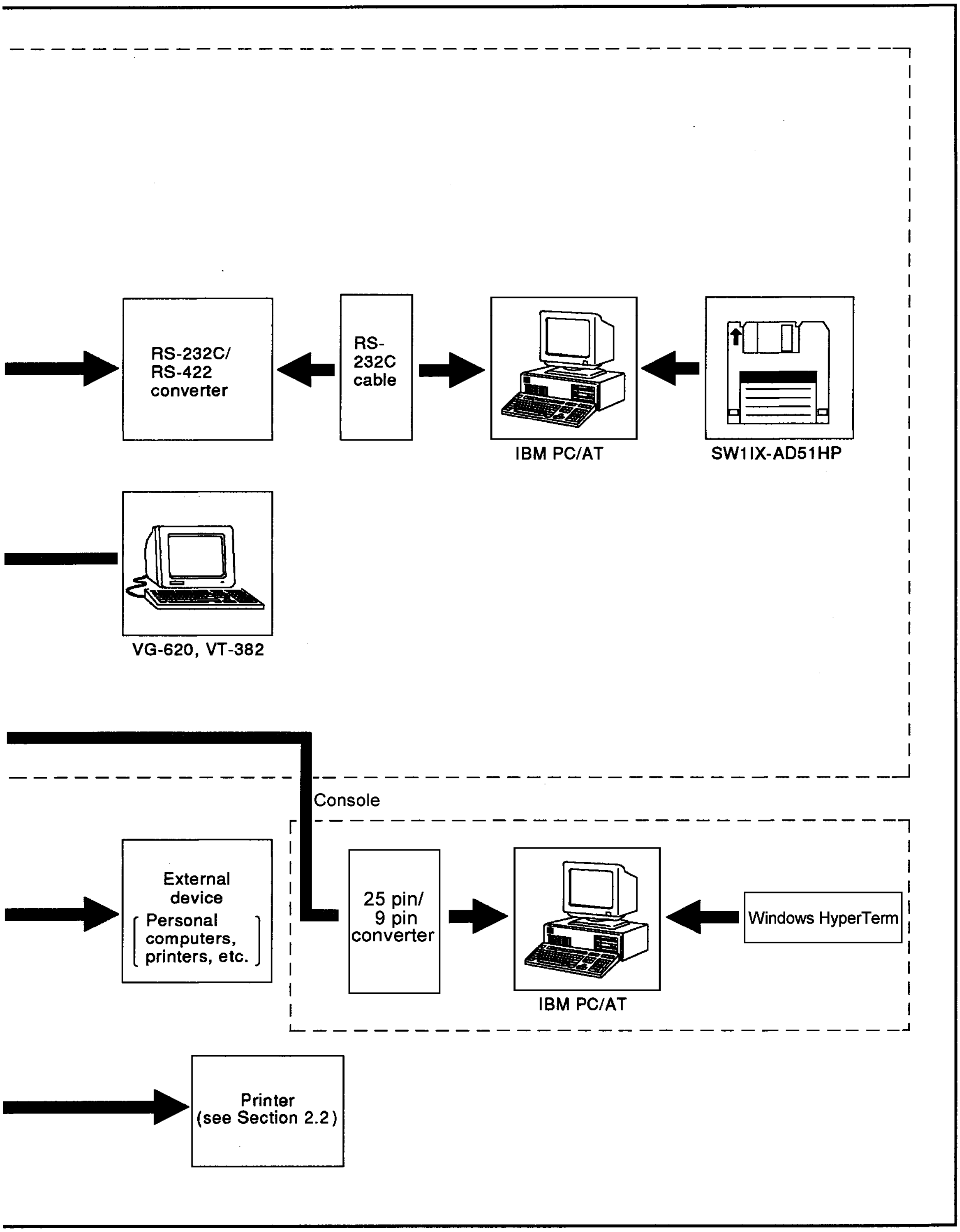
2.1.1 Overall configurations

(1) When using a A series CPU



* 1 An extension base unit cannot be connected to an A32B.

Fig. 2.1 A series CPU Overall Configuration


RS-232C/
RS-422
converter
RS-
232C
cable
IBM PC/AT
SW1IX-AD51HP
VG-620, VT-382
Console
External
device
Personal
computers,
printers, etc.
25 pin/
9 pin
converter
IBM PC/AT
Windows HyperTerm
Printer
(see Section 2.2)

(2) When using a AnS series CPU

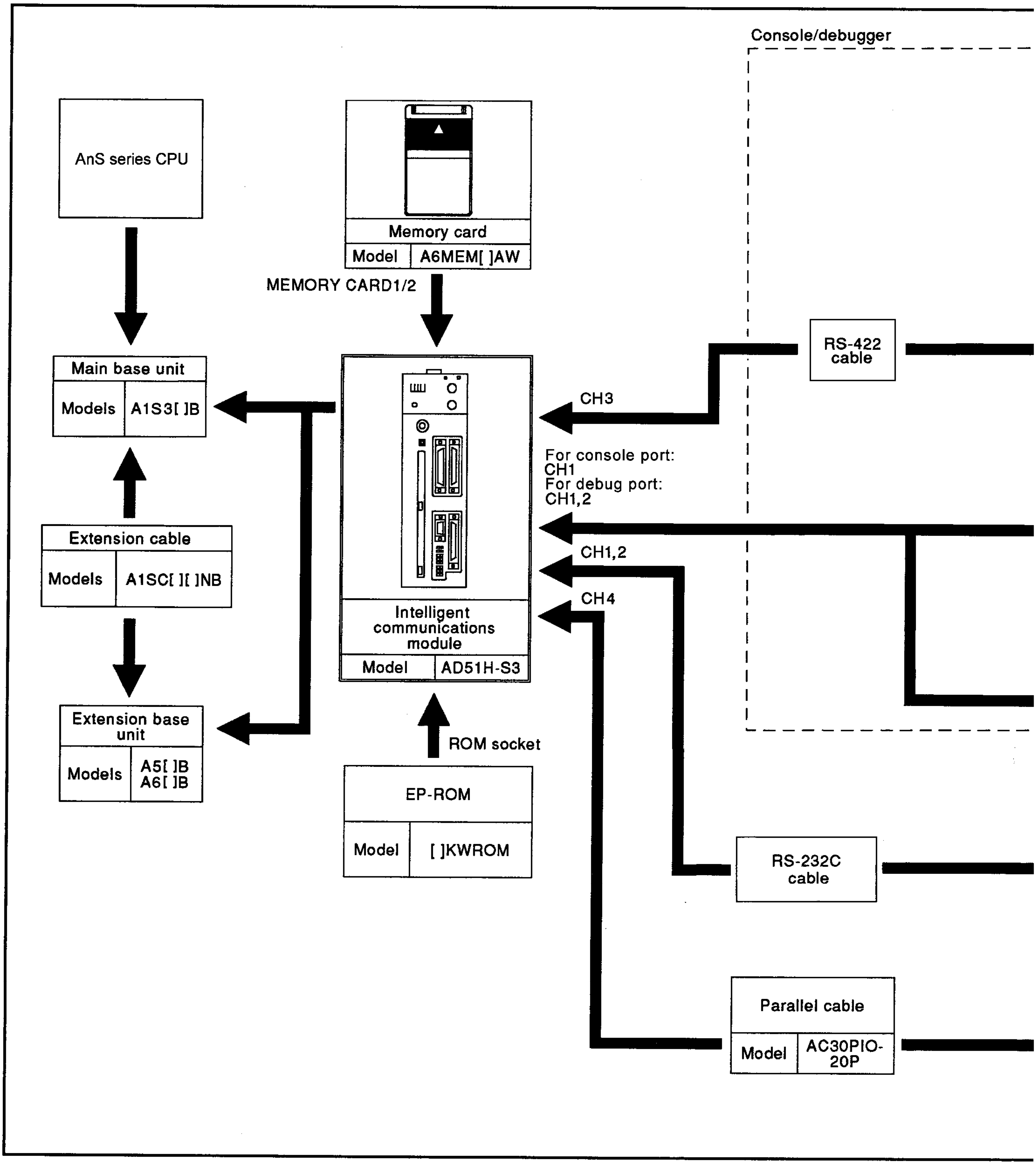
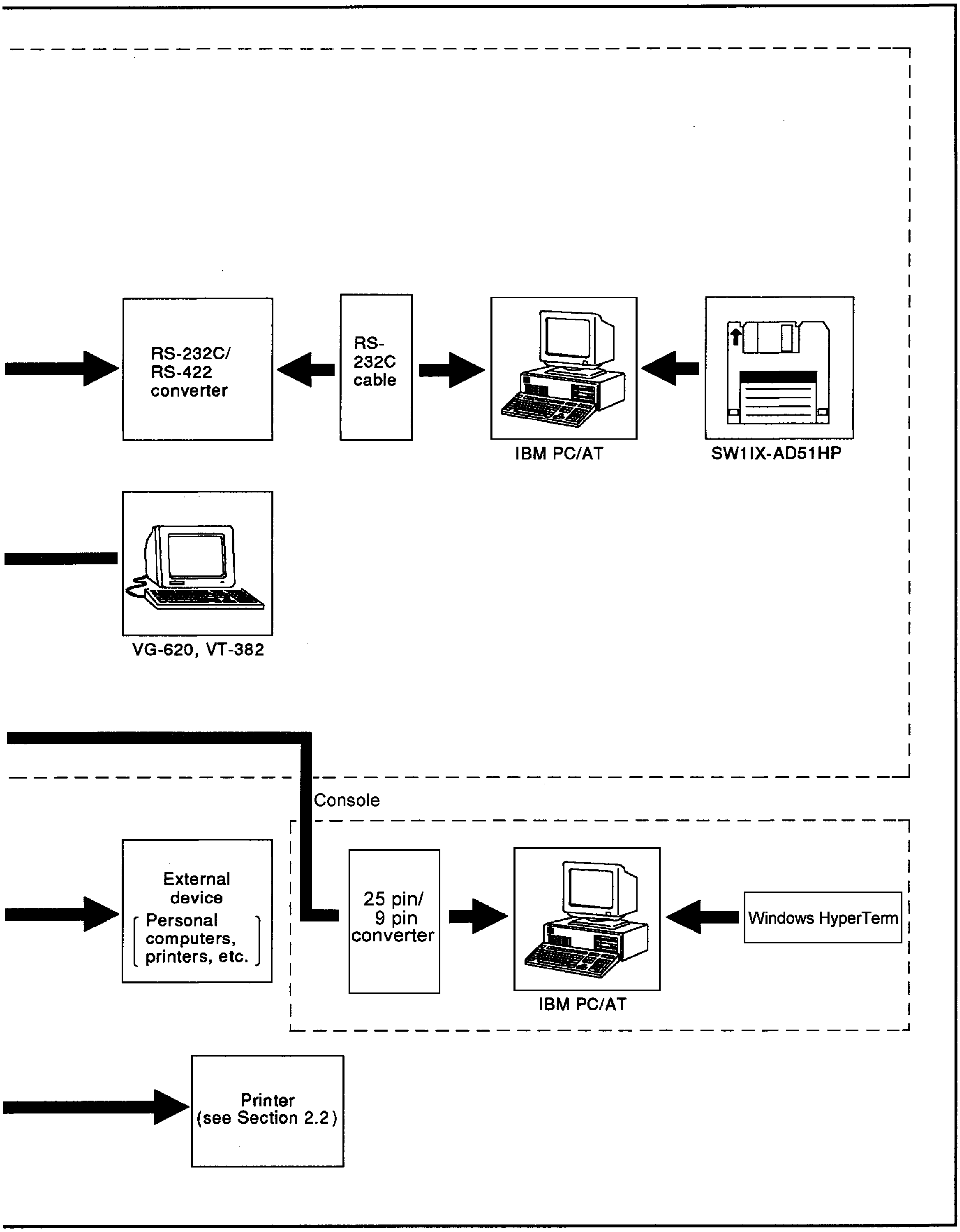
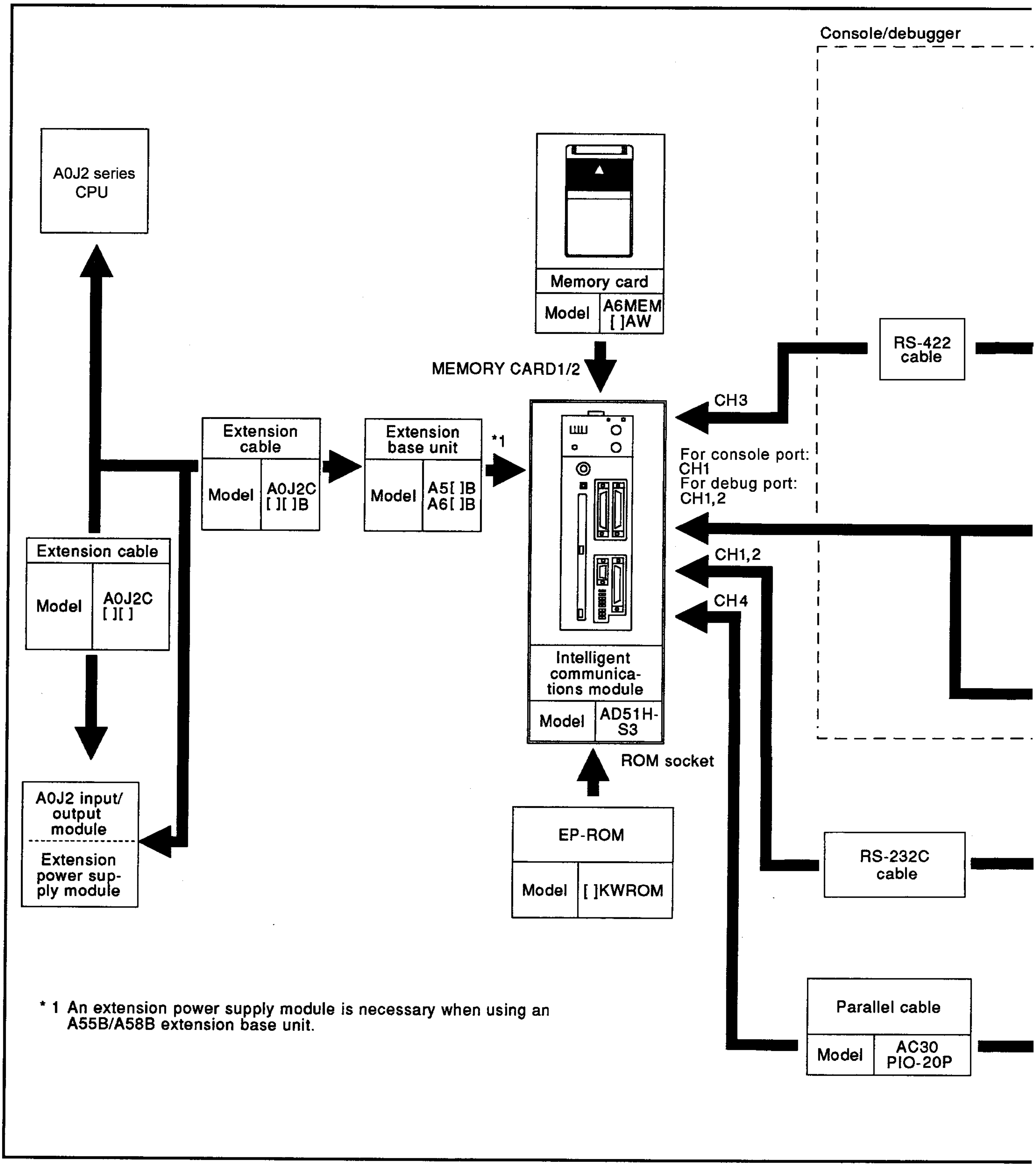


Fig. 2.2 Building AnS series CPU Overall Configuration

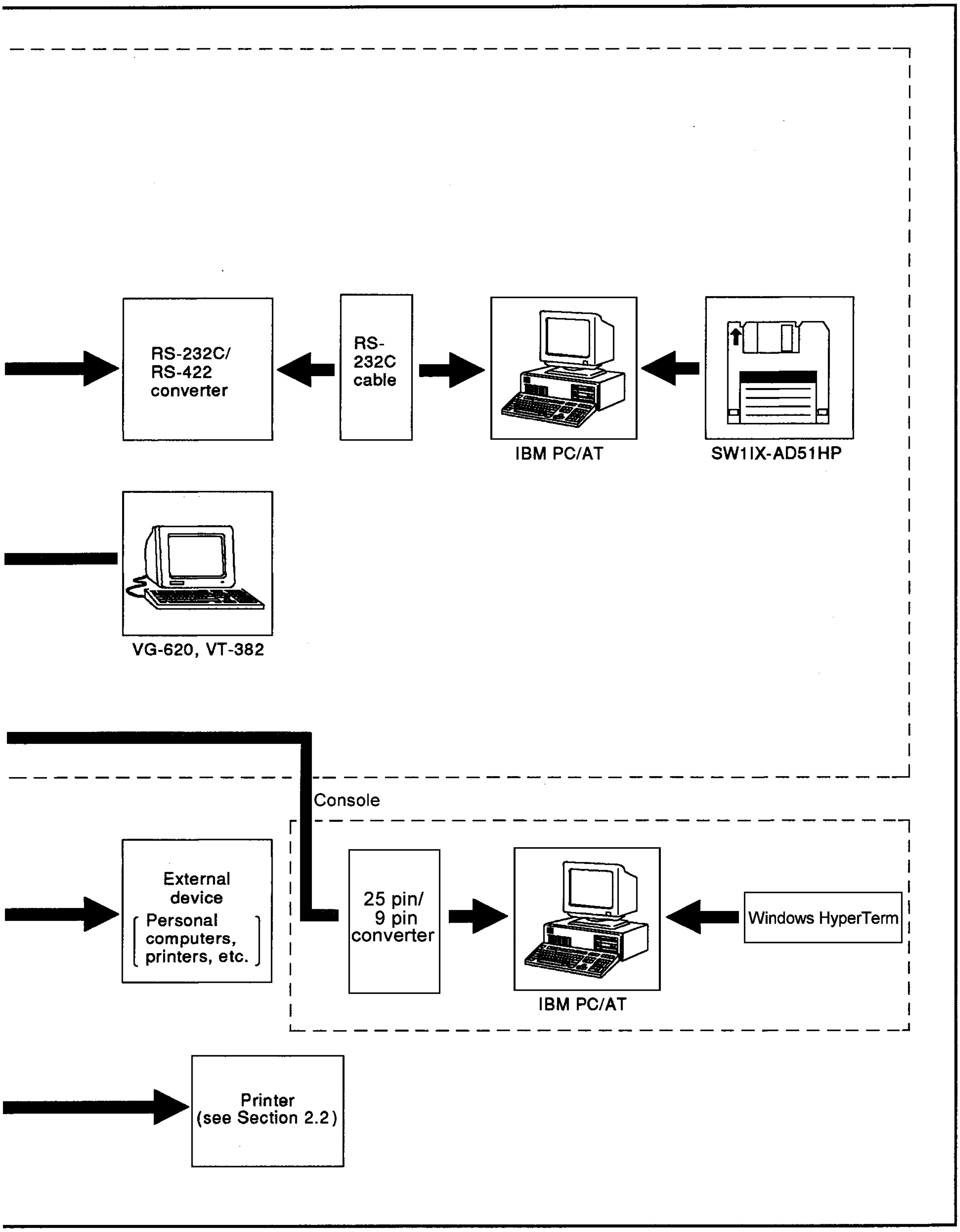

RS-232C/
RS-422
converter
RS-
232C
cable
IBM PC/AT
SW1IX-AD51HP
VG-620, VT-382
Console
External
device
Personal
computers,
printers, etc.
25 pin/
9 pin
converter
IBM PC/AT
Windows HyperTerm
Printer
(see Section 2.2)

(3) When using a A0J2 series CPU



* 1 An extension power supply module is necessary when using an A55B/A58B extension base unit.

Fig. 2.3 A0J2 series CPU Overall Configuration


RS-232C/
RS-422
converter
RS-
232C
cable
IBM PC/AT
SW1IX-AD51HP
VG-620, VT-382
Console
External
device
Personal
computers,
printers, etc.
25 pin/
9 pin
converter
IBM PC/AT
Windows HyperTerm
Printer
(see Section 2.2)

2.1.2 Applicable CPU modules

The AD51H can be used with the CPUs given below.

(1) Applicable CPU modules and the number of the AD57G modules

The table below shows the PC CPU modules to which the AD51H is applicable and the number of the AD51H modules connectable to the PC CPU modules.

CPU Module	Number of Connectable AD51Hs	Note
A0J2	1	The following types of modules should be taken into the number when they are used together with AD51H. • A0J2-C214S1 Computer Link/Multidrop Link Module (only when the computer link function is used) • AD51H(-S3) Intelligent Communication Module • AJ71C22S1 Multidrop Link System Module • AJ71C23-S3 Higher Controller High-Speed Link Module • AJ71E71 N(B2,B5T) Ethernet Interface Module • A1SJ71UC24-R2 Computer Link Module • A1SJ71UC24-PRF Computer Link/Printer function module • A1SJ71UC24-R4 Computer Link/Multidrop Link Module • A1SJ71E-71N(B2,B5T) Ethernet Interface Module
A0J2H A1, A1N, A1S(H) A1SJ(H), A2(-S1) A2N(-S1), A2S(H) A3, A3N A3H, A3M, A73	2	
A2(-S1), A2U(-S1) A2US(H)(-S1), A3A, A3U, A4U	6	
Q2AS(H)(-S1)*		

(A2CCPU is not applicable.)

*Access is allowed to only the internal devices within the A2US(H)CPU(-S2) range.

(2) Applicable base unit

The AD51H can be loaded to any slot of the main base unit or extension base unit with these two exceptions:

(a) The power supply capacity may be insufficient to load the AD51H to an extension base unit with no built-in power supply (A55B or A58B).

Wherever possible, avoid loading an AD51H module into this type of extension base unit.

If it is necessary to use an AD51H module on an extension base unit with no built-in power supply, it is important to consider (a) the power supply capacity of the main base unit and (b) the voltage drop along the extension cables when selecting the power supply module and the extension cables.

(For details, see the User's Manual of respective CPU module to be used.)

(b) The AD51H must not be loaded to the last slot of the A3CPU's 7th extension base unit.

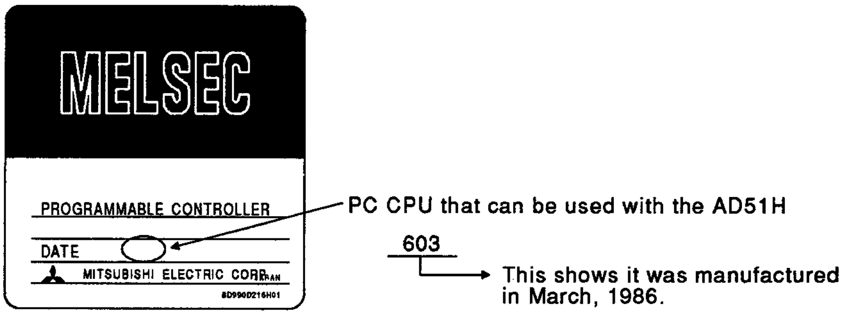
(c) When an A0J2 and an A0J2H are used with an A55B or A58B, an extension power supply module (A0J2PW or A0J2PW-DC24) is required.

POINT

Only PC CPU manufactured after February, 1986 can be used with the AD51H.

Applicability can be confirmed by checking the date area of the PC CPU name plate.

The following shows how to check the PC CPU name plate:


MELSEC
PROGRAMMABLE CONTROLLER
DATE
MITSUBISHI ELECTRIC CORP. JAPAN
BD990D216H01
PC CPU that can be used with the AD51H
603
This shows it was manufactured in March, 1986.

2.2 List of System Devices

Table 2.1 gives the devices needed to use the AD51H.

The particular device manual gives details about each device.

(1) Device that can connected to the AD51H

Table 2.1 shows the MELSEC-A series standard devices, recommended devices, and other devices which can be connected to the AD51H.

Table 2.1 shows the applicability of each device in regard to the following.

Table 2.1 List of System Devices

Product Name	Model	Remarks		
Intelligent communication module	AD51H-S3	• Main module		
Personal computer	IBM PC/AT	• Used as a console, a debugger, or a terminal		
General-purpose consoles	VG-620			
	VT-382			
Software package	SW1IX-AD51HPE	• PC/AT compatible system software for starting up the A1SD51S		
Memory card	A6MEM-256KAW A6MEM-512KAW A6MEM-1024KAW	• Memory card for program creation and storage (Required to start up the AD51H)		
Communication software	Windows HyperTerm	• Terminal emulator software when PC/AT is used as the console.		
General-purpose editors	MIFES	• Used for offline programming.		
	FINAL			
	EDLIN			
User's floppy disk	————	• A 2HD disk for storing data		
Printer	ESC/P is supported.	• Used to print out BASIC programs and data.		
Converters	EL-LINE-II	• An RS-232C/RS-422 converter when using the PC/AT		
	KS-10P			
	FA-CBL25S9P	• 25 pin/9 pin converter for AC30R2		For A1SD51S side
	FA-CBL25S9S			For PC/AT side
Connection cable	————	Classification	Cable length	Connection device
	AC30R2	• RS-232C cable	3 m	VG-620/VT-382
				PC/AT
				Printer

2.3 Compatibility

(1) Memory card compatibility

Model to use Model used for preparing the card	AD51H-S3	AD51H
AD51H-S3	–	x*1
AD51H	o	–

*1 : Use caution because the contents of the memory card can be destroyed.
If a memory card prepared by using an AD51H is once modified by using an AD51H-S3, it cannot be used with an AD51H.

(2) User ROM compatibility

Model to use Model used for preparing the ROM	AD51H-S3	AD51H
AD51H-S3	–	x
AD51H	o	–

(3) Operability when the following hardware and software versions are combined:

	SW1IX-AD51HPE	SW0IX-AD51HPE
AD51H-S3	o	Δ*2
AD51H	Δ*1	o

*1 : The following instruction must be executed after entering online programming:
"CONSOLE 24"
Inoperable if an execution file prepared by compiling is stored in the installed memory card or user ROM.

*2 : The following instruction must be executed after entering online programming:
"CONSOLE 25"

3. PERFORMANCE SPECIFICATIONS OF THE AD51H

This section explains and lists the performance specifications of the AD51H.

3.1 Performance Specifications

This section gives the performance specifications and internal memory maps of the AD51H.

3.1.1 Performance specifications list

Table 3.1 gives the performance specifications of the AD51H.

Table 3.1 Performance Specifications of the AD51H

Item		Specifications	Reference Section
CPU device		80C186 (15MHz)	—
Program language		AD51H-BASIC	—
Number of tasks		(Max.) eight tasks	—
Conditions for starting a task		• Started by power ON • Started by an interrupt from the PC CPU (Not for Compile BASIC) • Started by the start command from other task	3.13
Internal memory		Program memory: (Max.) 384K bytes/eight tasks (Task size can be set to 16K, 32K, 48K, or 64K bytes)	3.7.2
		Common memory: 8K bytes	3.7.4
		Extension register: 1024 points (2K bytes)	3.7.5
		Extension relay: 1024 points	3.7.7
General-purpose I/O		• Input: 27 points • Output: 17 points	3.9
Buffer		3K words (6K bytes)	3.7.3
Memory-protected		0H to 7FFFFH (Execution program in the memory card	5.2.9
Built-in interface	RS-422 I/F	• Conforms to EIA. RS-422 • Used with a connector • Channel 3 • Transmission distance: within 500 m	3.2.5
	RS-232C I/F	• Conforms to EIA. RS-232C • Used with a connector • Channels 1 and 2 • Transmission distance: within 15 m	3.2.3 to 3.2.4
	Centronics I/F	• Conforms to Centronics • Used with a connector • Channel 4 • Transmission distance: within 3 m	3.2.6
	Memory loading I/F	Applicable memory card: A6MEM-[ ]AW	3.3
Memory backup		No backup (However, the data in the memory card is battery-backed.)	5.5
Writing a user program to the ROM		Enabled (execution programs only) (Write to the ROM using the ROM writer connected to a PC/AT.)	—
Console		• PC/AT • VG-620 (manufactured by Victor Datu Systems) • VT-382 (manufactured by Japan Digital Equipment Inc.)	—
Number of occupying I/O points		48 points (vacant 16 points + X/Y 32 points)	3.9
5VDC Internal current consumption		1.0 A	—
Outside dimensions		250 (9.84)(H) x 76 (2.99)(W) x 120 (4.72)(D) mm	—
Weight		0.9 Kg (1.98 lb)	—

3.2 Specifications and Connections of Interfaces for External Devices

The AD51H has the following interfaces for connecting with external devices:

- RS-232C interface : 2 channels
- RS-422 interface : 1 channel
- Parallel interface : 1 channel

Sections 3.2.3 to 3.2.6 give details about the specifications and connections of each interface.

3.2.1 Interface uses

Interfaces connected to external device differ in use according to the device setting and the settings of SW1 to SW5 of mode setting switch 2.

Table 3.2 gives interface uses.

Table 3.2 Interface Uses

o: Available x: Not available

Interface Names	Uses			
	Console Port	Debug Port	General-purpose Port	Printer
RS-232C (CH1)	o *2	o *3	o	o
RS-232C (CH2)	x	o *3	o	o
RS-422 (CH3)	o *1	o *1	o	o
Parallel (CH4)	x	x	x	o

*1 The PC/AT can be used.

*2 The VT-382, VG-620 or PC/AT can be used.

*3 The VT-382, VG-620 or PC/AT can be used.

(1) Console port

An interface set to a console by SW1 to SW5 of mode setting switch 2 is set to a console port. (See Section 4.2.2)

If execution mode 03H ([Ctrl] + [C] key code) is received by a console connected to the console port, execution of multitasking is stopped.

(2) Debug port

An interface set to a debugger by SW1 to SW5 of mode setting switch 2 is set to a debug port. (See Section 4.2.2)

Multitask debugging can be executed by a console connected to the debug port.

(3) General-purpose port

An interface where console setting by SW1 to SW5 of mode setting switch 2 is not done is set to a general-purpose port.

(See Section 4.2.2.)

Execution of multitasking cannot be stopped even if 03H ([Ctrl] + [C] key code) is received by a peripheral device connected to the general-purpose port.

(4) Printer

BASIC programs and data can be printed out by connecting a printer.

3.2.2 Available baud rates

AD51H can set up to 19200 bps for each channel.

However, if several channels are used, the total (maximum) of the set values of the Transmission speed is 28.8 KBPS (28800 BPS). (Except for the parallel interface (CH.4))

(28800BPS) > (Baud rate of CH.1) + (Baud rate of CH.2) + (Baud rate of CH.3)

POINTS

(1) When the RS-232C interface is set to a console or debugger, the RS-232C interface of CH.1 or CH.2 is set to the following specifications.

(a)	Baud rate	: 9600 bps
(b)	Data length	: 8 bits
(c)	Stop bit	: 2 bits
(d)	Parity	: None
(e)	Receive buffer size	: 1024 bytes

If a VG-620, VT-382, or PC/AT is used as a console or debugger, set the VG-620, VT-382, or PC/AT as given above.

(2) Use the ZCNTL command to set the baud rate and others with BASIC programs.

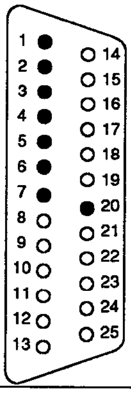
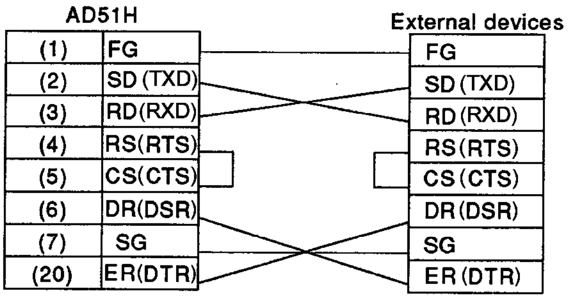
The AD51H-BASIC Programming Manual tells how to use the ZCNTL command.

3.2.3 RS-232C interface (CH1: channel 1)

(1) Interface specifications

Table 3.3 gives the specifications of the RS-232C interface (CH1: channel 1) of the AD51H.

Table 3.3 RS-232C Interface (CH1) Specifications of the AD51H

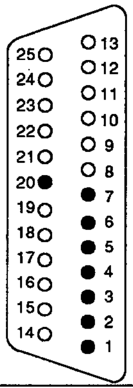
Item		Specifications				
Device to be Connected	Debug port	VG-620, VT-382, PC/AT				
	Console port					
	General-purpose port	With RS-232C interface				
Transmission method		Conforms to EIA and RS-232C.				
Transmission speed (bps)		300, 600, 1200, 2400, 4800, or 9600 can be selected.				
Synchronous method		Asynchronous system				
Transmission specifications (USART mode)	Transmission speed	300, 600, 1200, 2400, 4800, or 9600 can be selected.				
	Parity bit	Without or with parity (odd parity/even parity)				Depending on selection
	Stop bit	1, 1.5, 2 bits				
	Character data bit	5, 6, 7, 8 bits				
	Communications control	DTR/DSR (ER/DR) control or DC code control				
Connector (AD51H side) used	Model name	17L-10250-27-D9A (DDK Product)				
	Pin arrangement and pin numbers	Appearance of connector	Pin number	Signal abbreviation	Signal directions Inside ↔ Outside	General information
		 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	1	FG		Frame grounding maintenance
			2	SD (TXD)	→	Sent data
			3	RD (RXD)	←	Received data
			4	RS (RTS)	→	Request to send
			5	CS (CTS)	←	Sending is enabled.
			6	DR (DSR)	←	Data equipment ready
			7	SG		Signal ground
			20	ER (DTR)	→	Terminal READY notification
Connection example		 AD51H External devices (1) FG (2) SD (TXD) (3) RD(RXD) (4) RS(RTS) (5) CS(CTS) (6) DR(DSR) (7) SG (20) ER(DTR) FG SD (TXD) RD (RXD) RS(RTS) CS (CTS) DR (DSR) SG ER (DTR)				

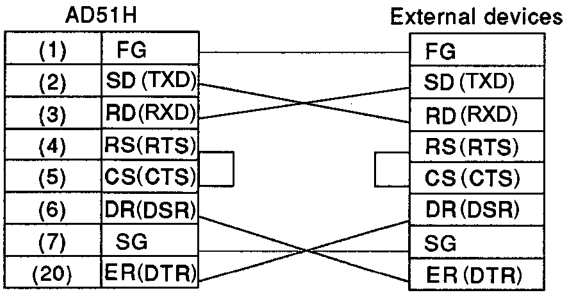
3.2.4 RS-232C interface (CH2: channel 2)

(1) Interface specifications

Table 3.4 gives the specifications of the RS-232C interface (CH2: channel 2) of the AD51H.

Table 3.4 RS-232C Interface (CH2) Specifications of the AD51H

<table>
<tr><th colspan="2">Item</th><th colspan="5">Specifications</th></tr>
<tr><td rowspan="2">Device to be Connected</td><td>Debug port</td><td colspan="5">VG-620, VT-382, PC/AT</td></tr>
<tr><td>General-purpose port</td><td colspan="5">With RS-232C interface</td></tr>
<tr><td colspan="2">Transmission method</td><td colspan="5">Conforms to EIA and RS-232C.</td></tr>
<tr><td colspan="2">Transmission speed (bps)</td><td colspan="5">300, 600, 1200, 2400, 4800, or 9600 can be selected.</td></tr>
<tr><td colspan="2">Synchronous method</td><td colspan="5">Asynchronous system</td></tr>
<tr><td rowspan="5">Transmission specifications (USART mode)</td><td>Transmission speed</td><td colspan="5">300, 600, 1200, 2400, 4800, or 9600 can be selected.</td></tr>
<tr><td>Parity bit</td><td colspan="3">Without or with parity (odd parity/even parity)</td><td colspan="2" rowspan="3">Depending on selection</td></tr>
<tr><td>Stop bit</td><td colspan="3">1, 1.5, 2 bits</td></tr>
<tr><td>Character data bit</td><td colspan="3">5, 6, 7, 8 bits</td></tr>
<tr><td>Communications control</td><td colspan="5">DTR/DSR (ER/DR) control or DC code control</td></tr>
<tr><td rowspan="10">Connector (AD51H side) used</td><td>Model name</td><td colspan="5">17L-10250-27-D9A(DDK Product)</td></tr>
<tr><td rowspan="9">Pin arrangement and pin numbers</td><td>Appearance of connector</td><td>Pin number</td><td>Signal abbreviation</td><td>Signal directions Inside ↔ Outside</td><td>General information</td></tr>
<tr><td rowspan="8">
25○ ○13
24○ ○12
23○ ○11
22○ ○10
21○ ○9
20● ○8
19○ ●7
18○ ●6
17○ ●5
16○ ●4
15○ ●3
14○ ●2
●1

</td><td>1</td><td>FG</td><td></td><td>Frame grounding maintenance</td></tr>
<tr><td>2</td><td>SD (TXD)</td><td>→</td><td>Sent data</td></tr>
<tr><td>3</td><td>RD (RXD)</td><td>←</td><td>Received data</td></tr>
<tr><td>4</td><td>RS (RTS)</td><td>→</td><td>Request to send</td></tr>
<tr><td>5</td><td>CS (CTS)</td><td>←</td><td>Sending is enabled.</td></tr>
<tr><td>6</td><td>DR (DSR)</td><td>←</td><td>Data equipment ready</td></tr>
<tr><td>7</td><td>SG</td><td></td><td>Signal ground</td></tr>
<tr><td>20</td><td>ER (DTR)</td><td>→</td><td>Terminal READY notification</td></tr>
<tr><td colspan="2">Connection example</td><td colspan="5">
AD51H
External devices
(1) FG
(2) SD (TXD)
(3) RD(RXD)
(4) RS(RTS)
(5) CS(CTS)
(6) DR(DSR)
(7) SG
(20) ER(DTR)
FG
SD (TXD)
RD (RXD)
RS(RTS)
CS (CTS)
DR (DSR)
SG
ER (DTR)

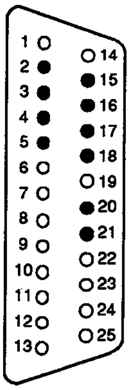
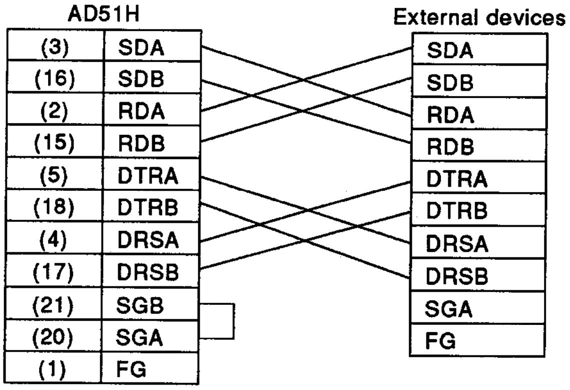
</td></tr>
</table>

3.2.5 RS-422 interface (CH3: channel 3)

(1) Interface specifications

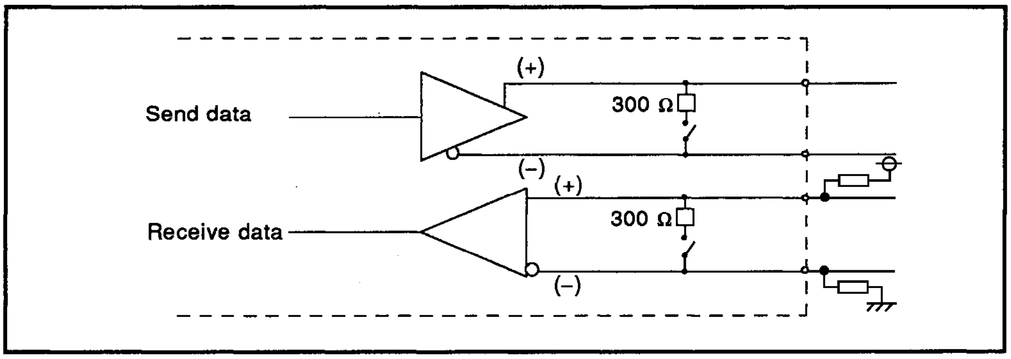
Table 3.5 gives the specifications of the RS-422 interface (CH3: channel 3) of the AD51H.

Table 3.5 RS-422 Interface (CH3) Specifications

<table>
<tr><th colspan="2">Item</th><th colspan="5">Specifications</th></tr>
<tr><td rowspan="3">Device to be Connected</td><td>Debug port</td><td colspan="5" rowspan="2">PC/AT</td></tr>
<tr><td>Console port</td></tr>
<tr><td>General-purpose port</td><td colspan="5">With RS- 422 interface</td></tr>
<tr><td colspan="2">Transmission method</td><td colspan="5">Conforms to EIA and RS-422.</td></tr>
<tr><td colspan="2">Synchronous method</td><td colspan="5">Asynchronous system</td></tr>
<tr><td rowspan="5">Transmission specifications (USART mode)</td><td>Baud rate (bps)</td><td colspan="5">300, 600, 1200, 2400, 4800, or 9600 can be selected.</td></tr>
<tr><td>Parity bit</td><td colspan="4">Without or with parity (odd parity/even parity)</td><td rowspan="3">Depending on selection</td></tr>
<tr><td>Stop bit</td><td colspan="4">1, 1.5, 2 bits</td></tr>
<tr><td>Character data bit</td><td colspan="4">5, 6, 7, 8 bits</td></tr>
<tr><td>Communications control</td><td colspan="5">DTR/DSR (ER/DR) control or DC code control</td></tr>
<tr><td rowspan="11">Connector (AD51H side) used</td><td>Model name</td><td colspan="5">17L-10250-27-D9A (DDK Product)</td></tr>
<tr><td rowspan="10">Pin arrangement and pin numbers</td><td>Appearance of connector</td><td colspan="2">Signal abbreviation</td><td>Pin number</td><td>Signal direction</td></tr>
<tr><td rowspan="9">
</td><td rowspan="2">Send data</td><td>SDA</td><td>(3)</td><td rowspan="2">→Outside</td></tr>
<tr><td>SDB</td><td>(16)</td></tr>
<tr><td rowspan="2">Received data</td><td>RDA</td><td>(2)</td><td rowspan="2">←Outside</td></tr>
<tr><td>RDB</td><td>(15)</td></tr>
<tr><td rowspan="2">READY stop notification</td><td>CSA</td><td>(5)</td><td rowspan="2">→Outside</td></tr>
<tr><td>CSB</td><td>(18)</td></tr>
<tr><td rowspan="2">Data equipment READY</td><td>RSA</td><td>(4)</td><td rowspan="2">←Outside</td></tr>
<tr><td>RSB</td><td>(17)</td></tr>
<tr><td>Signal ground</td><td>SGA</td><td>(21)</td><td></td></tr>
<tr><td colspan="2">Connection example</td><td colspan="5">
</td></tr>
</table>

(2) Function block diagram

The following diagram illustrates the functional blocks of the A1SD51S.



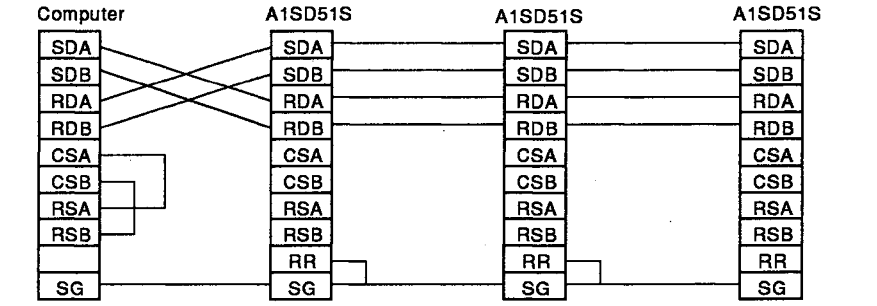
(3) 1:n (multidrop) connection

By using the RS-442 interface, a single computer can be connected with multiple A1SD51S modules (1:n connection).

(a) Specifications

Maximum connectable stations: 32 stations
Maximum extension distance : 500 m

(b) Wiring



Terminal resistor required

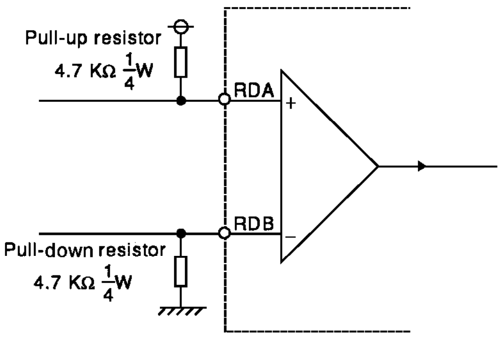
Terminal resistor not required for an
(Set SW7, 8 of Switch 2 to OFF)

Terminal resistor not required for an
(Set SW7, 8 of Switch 2 to OFF)

Terminal resistor required
(Set SW7, 8 of Switch 2 to ON)

(c) Notes on wiring

- Do not make connections between the CSA/CSB and RSA/RSB terminals of different AD51H modules.
- Disable RS/CS control with Procedure Code 18 of the ZCNTL command.
- If both terminal stations are AD51H modules, connect a terminal resistor to the RS-422 or the RS-485 interface.
- If data receive errors are likely to occur due to noise along transmission lines, connect pull-up and pull-down resistors as shown below.



(d) Notes on programming

- When performing 1:n connection, enable high-impedance control with Procedure Code 48 of the ZCNTL command.
  For details of high-impedance control, refer to the AD51H-BASIC Programming Manual (Command).

3.2.6 Parallel interface (CH4: channel 4)

(1) Interface specifications

Table 3.6 gives the specifications of the parallel interface (CH4: channel 4) of the AD51H.

Table 3.6 Parallel Interface (CH4) Specifications

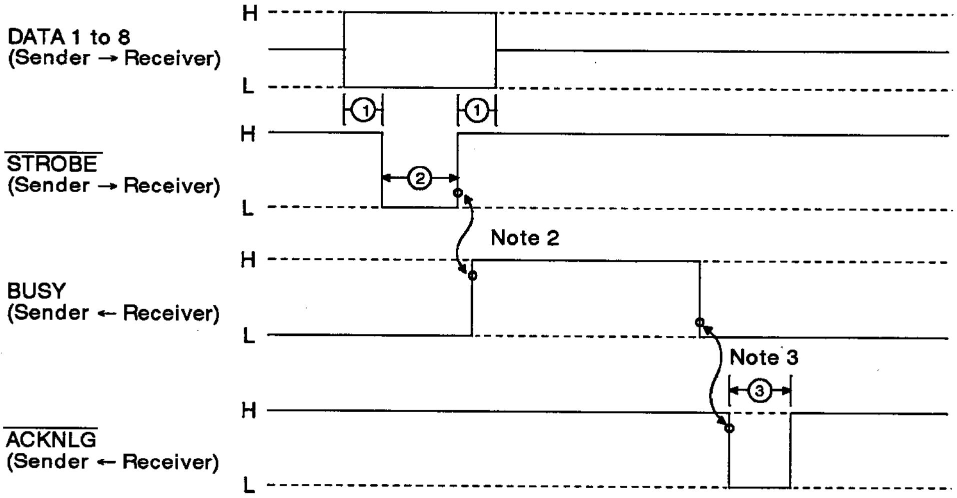
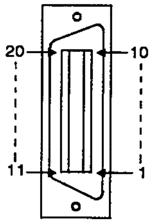
<table>
<tr><th colspan="2">Item</th><th colspan="9">Specifications</th></tr>
<tr><td colspan="2">Conformity standard</td><td colspan="9">Conforms to Centronics.</td></tr>
<tr><td colspan="2">Insulation method</td><td colspan="9">Photocoupler insulation</td></tr>
<tr><td colspan="2" rowspan="2">Signal level</td><td>Input</td><td colspan="8">V_{IH} = 2V, V_{IL} = 0.8V</td></tr>
<tr><td>Output</td><td colspan="8">V_{OH} = 2.4V, V_{OL} = 0.5V</td></tr>
<tr><td colspan="2">Timing chart</td><td colspan="9">
(Note 1) 1)Min. 1.0μS
2)Min. 1.0μS, max. 500μS 3) Min. 2μS
(Note 2) The BUSY signal goes ON at the leading or falling edge of $\overline{\text{STROBE}}$.
(Note 3) At the falling edge of the BUSY signal, $\overline{\text{ACKNLG}}$ is transmitted within 100μS.
The BUSY signal is canceled after $\overline{\text{ACKNLG}}$ is received.</td></tr>
<tr><td rowspan="7">Connector (AD51H side) used</td><td>Model name</td><td colspan="9">10220-52A2JL</td></tr>
<tr><td rowspan="6">Pin arrange-ment</td><td>Appearance of connector</td><td>No.</td><td>Signal name</td><td>No.</td><td>Signal name</td><td>No.</td><td>Signal name</td><td>No.</td><td>Signal name</td></tr>
<tr><td rowspan="5"> </td><td>1</td><td>CHASIS GND</td><td>6</td><td>NC</td><td>11</td><td>DATA8</td><td>16</td><td>DATA3</td></tr>
<tr><td>2</td><td>$\overline{\text{ACKNLG}}$</td><td>7</td><td>INIT</td><td>12</td><td>DATA7</td><td>17</td><td>DATA2</td></tr>
<tr><td>3</td><td>DATA6</td><td>8</td><td>DATA1</td><td>13</td><td>PE</td><td>18</td><td>GND</td></tr>
<tr><td>4</td><td>DATA5</td><td>9</td><td>$\overline{\text{STROBE}}$</td><td>14</td><td>SLCT</td><td>19</td><td>$\overline{\text{ERROR}}$</td></tr>
<tr><td>5</td><td>DATA4</td><td>10</td><td>BUSY</td><td>15</td><td>GND</td><td>20</td><td>GND</td></tr>
</table>

Table 3.6 Parallel Interface Specifications (continued)

Item	Specifications			
Connection	AD51H		External device	
	(1)	CHASIS GND	CHASIS GND	(17)
	(2)	$\overline{\text{ACKNLG}}$	$\overline{\text{ACKNLG}}$	(10)
	(3)	DATA6	DATA6	(7)
	(4)	DATA5	DATA5	(6)
	(5)	DATA4	DATA4	(5)
	(6)	NC	NC	(36)
	(7)	INIT	INIT	(31)
	(8)	DATA1	DATA1	(2)
	(9)	$\overline{\text{STROBE}}$	$\overline{\text{STROBE}}$	(1)
	(10)	BUSY	BUSY	(11)
	(11)	DATA8	DATA8	(9)
	(12)	DATA7	DATA7	(8)
	(13)	PE	PE	(12)
	(14)	SLCT	SLCT	(13)
	(15)	GND	GND	(22)
	(16)	DATA3	DATA3	(4)
	(17)	DATA2	DATA2	(3)
	(18)	GND	GND	(24)
	(19)	$\overline{\text{ERROR}}$	$\overline{\text{ERROR}}$	(32)
	(20)	GND	GND	(19)
	• For the external device, set the pin numbers for other than the above pins to NC (NONE CONNECTION)			

3.3 Memory Card Interface Specifications

This section explains memory card interface specifications.

(1) Memory card interface specifications

Table 3.7 gives the specifications of the memory card interface of the AD51H.

Table 3.7 Memory Card Interface Specifications

Item	MEMORY CARD	
	1	2
Data that can be stored	• File data • Execution program	• File data
Applicable memory card (Section 6.2.1 gives memory card specifications.)	Model name	Memory capacity
	A6MEM-256KAW	256K bytes
	A6MEM-512KAW	512K bvtes
	A6MEM-1024KAW	1024K bytes

3.4 ROM Socket Specifications

The following explains the specifications of the ROM socket and applicable EP-ROMs used with the AD51H.

(1) Specifications of ROM sockets

Table 3.8 ROM Socket Specifications

Item	Specifications
Number of pins	40 pins
EP-ROM that can be installed	64KWROM (128K bytes)
	128KWROM (256K bytes)
	256KWROM (512K bytes)

(2) Specifications of applicable EP-R0Ms

Table 3.9 EP-ROM Specifications

Item \ Model Name		64KWROM	128KWROM	256KWROM
Memory specifications		EP-ROM (Only read is enabled.)		
Memory capacity		128K bytes	256K bytes	512K bytes
Structure		40 pin IC package		
Storage data	Name	Execution program		
	Capacity	128K bytes	256K bytes	384K bytes

REMARK

Storing execution programs to the EP-ROM:

Use a ROM writer connected to a PC/AT to store execution programs to the EP-ROM.

Section 2.2 and Appendix 3 give details about ROM writers that can be connected to a PC/AT.

3.5 AD51H-BASIC commands

Table 3.10 gives the AD51H-BASIC commands that can be used with the AD51H. (The AD51H-BASIC Programming Manual gives details about commands and functions.)

Table 3.10 AD51H-BASIC Commands and Functions

Classification		Commands and Functions	Operation Description
Command to create and modify a program	Preparation for creating a program	NEW	Deletes the program in the program area and initializes all variables.
		AUTO	Automatically displays the line number at the beginning of a line.
		CLEAR	Initializes a variable and sets a memory area.
	Editing a program	DELETE	Clears the designated program range.
		RENUM	Renumbers the line numbers of a program.
		LIST	Displays all (or part) of a program in the program area.
		ZBAS	Returns the BASIC task number area number of the program currently being written or executed.
Function to read memory capacity	Reading memory capacity	FRE	Sets the size of an unused area in the program area by the number of bytes.
Command to execute and control a program	Executing a program	RUN	Executes a program in the program area. Reads and executes a program from an FD, HD, or memory card.
		CONT	Restarts execution of a program stopped by pressing the [BREAK], or [CTRL] + [C] keys. Restarts the program execution stopped by a STOP or END command.
	Controlling a program	CHAIN	This (a) clears all or part of the currently executed program, and (b) reads and executes a designated program.
		FOR ~ NEXT	Repeatedly executes a series of commands a designated number of times.
		STOP	When in the edit mode, temporarily stops execution of a program and makes the processing state wait for a command input. When in the execution mode, stops execution of a program and makes the processing state pause.
		END	When in the edit mode, stops execution of a program and makes the processing state wait for a command input. When in the execution mode, stops execution of a program and makes the processing state pause.
		WHILE ~ WEND	Repeatedly executes a series of commands between the WHILE and WEND commands while a designated condition is satisfied.
	Branching a program	GOTO	Unconditionally jumps the program execution to a designated line.
		GOSUB ~ RETURN	Branches a program to the subroutine and returns it to a designated line number.
		ON GOTO	Branches a program to one of designated line numbers according to a designated value.
		ON GOSUB	Branches the program to one of designated line numbers according to a designated value.

		IF GOTO ELSE
		IF THEN ELSE
			Selects a target branch according to the result of an operation formula.

Command to debug a program	Debugging a program	TRON	Starts tracing a program.
		TROFF	Resets the program trace set by the TRON command.
Command to convert type	Type conversion	CDBI	Converts a double-precision real number into a two-word (32-bit) integer used by a PC CPU.
		CSNI	Converts a single-precision real number into a two-word (32-bit) integer used by a PC CPU.

Table 3.10 AD51H-BASIC Commands and Functions (continued)

Classification		Commands and Functions	Operation Description
Command to write to a designated bit	Writing to a designated bit	WTSET	Writes 0 or 1 to a designated bit of integer type array variable.
Command to read from a designated bit	Reading from a designated bit	RDSET	Reads one-bit data from a designated bit of integer type array variable.
Character-string function	Character↔ASCII conversion	ASC	Converts the first character in a character-string formula to a corresponding character code.
		CHR$	Converts the value of an integer expression into character codes.
	Character→ numeric conversion	STR$	Regards a designated numeric value as a decimal and converts it into a character string.
		VAL	Converts a character string into a numeric value.
	Numeric value→ character string conversion	HEX$	Converts a decimal number into a character string of hexadecimal numbers.
		OCT$	Converts a decimal number into a character string of octal numbers.
		BIN$	Converts a decimal number into a character string of binary numbers.
	Character extraction	LEFT$	Extracts and designates a character string (with a designated number of characters) beginning with the left end of the character string.
		MID$ (part 2)	Designates a character string beginning with a designated position in a character string.
		RIGHT$	Extracts and designates a character string (with a designated number of characters) beginning with the right end of the character string.
		STRING$	Designates a character for designated number of characters.
		SPACE$	Designates a blank character string for designated number of strings.
	Character search	INSTR	Searches a designated part of a character string and designates the position found first.
	Character-string length search	LEN	Designates the number of characters that constitute a character string.
Format control function	Format control	SPC	Designates a blank space in the designated number of spaces.
		TAB	Advances a present character display position to a designated position.
Screen control command	Screen control	CONSOLE	Designates the number of display columns on the console screen.
		CLS	Clears the display from the screen.
		LOCATE	Designates the display position on the console screen.
I/O control command	Board allocation	ZOPEN	Opens a communications port for an external device.
		ZCLOSE	Closes the channel of the port used for communications with an external device.
	Buzzer output	BEEP	Sounds a buzzer from the built-in speaker in the console.
	Printer output	LLIST	Outputs all programs, part of a program, or any program in a designated range of the program area to the printer.
		LPRINT	Outputs data to the printer.
		LPRINT USING	Outputs data to the printer in a designated format.
		LFILES	Outputs the name and size of a file to the printer.
		ZLDV	Switches the port of the printer.
		WIDTH	Sets the output width to the printer.
	Output to the screen	PRINT	Displays data on the screen.
		PRINT USING	Displays a character string or numeric value in a designated format.
		KEY LIST	Displays a character string defined by a function key on the screen.
	Output to the port	ZSEND	Transmits data from a designated port.
	Input from the keyboard	INPUT	Enables data input from the keyboard.
		LINE INPUT	Stores an input line to the character-string variables area without dividing the input line (within 255 characters).
	Input from the port	ZRECEIVE	Receives data from a communications port.
	Port setting change	ZCNTL	Changes setting of a port to an open communications port or reads the state of a communications port.

Table 3.10 AD51H-BASIC Commands and Functions (continued)

Classification		Commands and Functions	Operation Description
Input function	Input from the keyboard	INKEY$	When a character is input from the keyboard, this designates the character. If no character is input, this designates an empty character string.
		INPUT$	Reads and designates a designated-length character string from the keyboard of a console. Reads and designates a designated-length character string from a sequential file. Reads and designates a designated-length character string from a communications port of the AD51H.
Port control function	Controls port state	COM ON/OFF/STOP	Controls interruption-enabled, interruption-disabled, and stop from a communications line.
		ON COM GOSUB	This defines the starting line of a branch processing routine when an interruption is given through a communications line.
Error control function	Error control	ON ERROR GOTO	When an error occurs, this executes a program from a designated line number.
		ERROR	Makes an error with a designated error code occur.
Error information function	Error information	ERR	Designates a detected error code.
		ERL	Designates the line number where an error is detected.
Repetition command to programming mode	Repetition to the programming mode	SYSTEM	Stops control of the BASIC interpreter and switches the mode to the AD51H system mode. Stops offline programming and returns the screen to the main menu.
File operation command	BASIC program I/O	LOAD	Reads a program in an FD, HD, or memory card to the program area.
		MERGE	Combines a program in the program area with a program in an FD or HD.
		SAVE	Stores a program in an FD, HD, or memory card.
	File maintenance	FILES	Displays the name of a file in an FD, HD, or memory card
		FORMAT	Initializes (formats) the file area in a memory card.
		KILL	Clears a file in an FD, HD, or memory card. Clears a system name in an FD, HD, or memory card.
		NAME	Changes the file name of a program or data in an FD, HD, or memory card.
I/O command to the file	Allocation of file buffer	OPEN	Opens a file, and enables I/O processings.
		CLOSE	Ends file I/O processing.
	I/O of a sequential file	PRINT#	Writes data to a sequential file.
		PRINT# USING	Designates the data format and writes data to a sequential file.
		INPUT#	Reads data from a sequential file.
		LINE INPUT#	Reads a character line (up to 265 characters) from a sequential file without dividing the line to the character string variables area.
	I/O of a random file	FIELD	Allocates a designated variable area to buffer of the random file.
		LSET	To prepare for the PUT command, this transfers data to the left end of buffer of a random file.
		RSET	To prepare for the PUT command, this transfers data to the right end of buffer of a random file.
		PUT	Writes one-record data from the buffer to a random file.
		GET	Reads one-record data from a random file to the buffer.

Table 3.10 AD51H-BASIC Commands and Functions (continued)

Classification		Commands and Functions	Operation Description
File information function	File information	EOF	Designates -1 when the end of a sequential file is detected.
		LOF	Designates the size of a file in the number of sectors.
		LOC	Designates the present end position in a file.
File data	Character string → numeric value	CVI	Returns a character string converted by the MKI$ function to an integer.
		CVS	Returns a character string converted by the MKS$ function to a single-precision real number.
		CVD	A character converted by the MKD$ function is returned to a double-precision real number.
		CVSMBF	Converts a character string, that is converted from a single-precision real number of the AD51H-BASIC using an MKS$ function, into the internal representation (single-precision internal representation of the IEEE format) of a floating decimal point real number used for AnA, AnU, QnU, AnUS and QnAS.
		CVDMBF	Converts a character string, that is converted from a double-precision real number of the AD51H-BASIC using an MKD$ function, into the double-precision internal representation of the IEEE format.
	Numeric value → character string	MKI$	Converts an integer type numeric value into a character string.
		MKS$	Converts a single-precision type numeric value into a character string.
		MKD$	Converts a double-precision type numeric value into a character string.
		MKSMBF$	Converts internal representation data (single-precision internal representation of the IEEE format) of a floating decimal point real number used for AnA, AnU and AnUS into a character string that can be expressed as a numeric value by using the CVS function.
		MKDMBF$	Converts data of the double-precision internal representation of the IEEE format into a character string that can be expressed as a numeric value by using the CVD function.
Definition and declaration command	Definition and declaration	DEF FN	Defines a user function and gives it a name.
		DEFINT	Defines a variable (that begins with a character in a designated range) as an integer type variable.
		DEFSNG	Defines a variable (that begins with a character in a designated range) as a single-precision real number type variable.
		DEFDBL	Defines a variable (that begins with a character in a designated range) as a double-precision real number type variable.
		DEFSTR	Defines a variable (that begins with a character in the designated range) as a character type variable.
		KEY	Defines a character string to each function key of a console.
		DIM	Designates the size of an array variable and allocates the memory capacity needed for the array.
		ERASE	Clears an array defined by the DIM command from the memory.
		REM	Designates a comment in a program.
		COMMON	Sets the variables to be transferred to a program which is executed by the CHAIN command.
Figure and character-string operation command	Data operation	DATA	Designates numeric values and character strings to be read by the READ command.
		READ	Reads a value defined by the DATA command and assigns it to a variable.
		RESTOR	Reads the value of a DATA command which is to be read by a READ command from the beginning of the DATA command of a designated line number.
	Assignment to variables	LET	Assigns the value of a formula to a variable.
		SWAP	Exchanges the values of two variables.
		ZMOVE	Transfers a value between two variables.
		BSWAP	Exchanges two values in a byte unit.
	Replacement of a character string	MID$ (part 1)	Replaces part of a character string with another character string.

Table 3.10 AD51H-BASIC Commands and Functions (continued)

Classification		Commands and Functions	Operation Description
Numeric value function	Arithmetical function	ABS	Designates the absolute value of a formula.
		ATN	Designates the inverse tangent (tan-1) of a formula.
		COS	Designates the value of the cosine in a trigonometric function.
		EXP	Designates the value of an exponential function that has e as its base.
		FIX	Truncates the decimal part of a numeric value and designates it as an integer.
		INT	Designates the integer value of a formula.
		LOG	Designates the value of a natural logarithm.
		RND	Designates a random number.
		ROT	Designates a value processed by bit rotation.
		SGN	Designates a numeric value code.
		SIN	Designates the value of the sine in a trigonometric function.
		SQR	Designates the value of a square root.
		TAN	Designates the value of a trigonometric function.
		SHA	Designates a value when the memory capacity of a designated value was processed by an arithmetical shift.
		SHT	Designates a value when the memory capacity of a designated value was processed by a logical shift.
	Numeric value search	SEARCH	Searches for a value designated from the element of any array variable, and designates the position of the element.
	Type conversion function	CDBL	Converts an integer and a single-precision real number into a double-precision real number.
		CSNG	Converts an integer and a double-precision real number into a single-precision real number.
		CINT	Converts a single-precision real number value and a double-precision real number value into an integer.
		CIDB	Converts a two-word (32-bit) integer used by a PC CPU into a double-precision real number.
		CISN	Converts a two-word (32 bits) integer used by a PC CPU into a single-precision real number.
PC access function	PC access	PCRD	Reads various data from a PC CPU.
		PCWT	Writes various data to a PC CPU. Operates a PC CPU.
Port switching command	Port switching	ZIDV	Sends the INPUT command from a designated console.
		ZODV	Outputs the PRINT command to a designated console.
Buffer access function	Buffer access	GETMEM	Reads data from the buffer, common memory, and extension register (ED) of the AD51H.
		PUTMEM	Writes data to the buffer, common memory, and extension register (ED) of the AD51H.
Clock function command	Clock function	TIME$	Sets and reads a PC CPU time (hour, minute, and second).
		DATE$	Sets and reads the PC CPU date (year, month, day, and day of the week).
Real-time control command	Real-time control	ZEVENT	Sets whether an event occurrence is valid/invalid.
		DEF ZEVENT	Defines an event to synchronize executions between programs. Defines an event by the extension relay (EM) of the AD51.
		ZRELEASE	Permits other programs to use a source that was allocated a source number.
		ZRESERVE	Prohibits other programs from using a source that was allocated a source number.
		ZSIGNAL	Makes an event designated by a program occur.
		ZSTART	Starts a designated program.
		ZURGENCY	Changes the priority of a program.

Table 3.10 AD51H-BASIC Commands and Functions (continued)

Classification		Commands and Functions	Operation Description
Command for real-time control	Real-time control	ZWAIT DELAY	Suspends execution of a program until a designated time passes.
		ZWAIT EVENT	Suspends execution of a program until a designated event occurs.
Real-time control function	Real-time control	ZMESSAGE	Defines a message port.
		ZMESSAGE CLOSE	Closes a message port.
		ZMESSAGE GET	Reads a message from a message port.
		ZMESSAGE KILL	Clears a defined message port.
		ZMESSAGE OPEN	Opens a message port.
		ZMESSAGE PUT	Writes a message to a message port.

3.6 AD51H internal memory and memory map

Table 3.11 shows the memory and memory map that can be used with the AD51H.

Table 3.11 AD51H Internal Memory and Memory Map

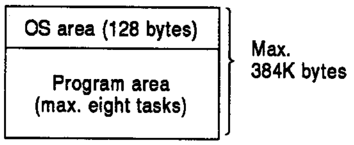
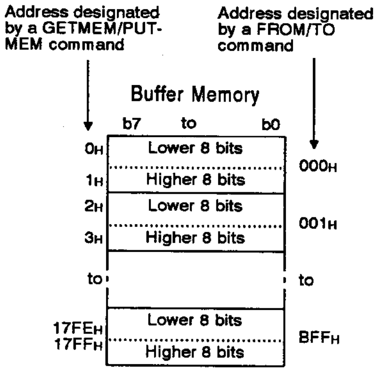
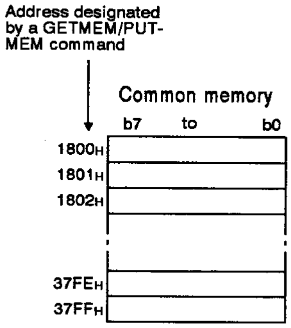
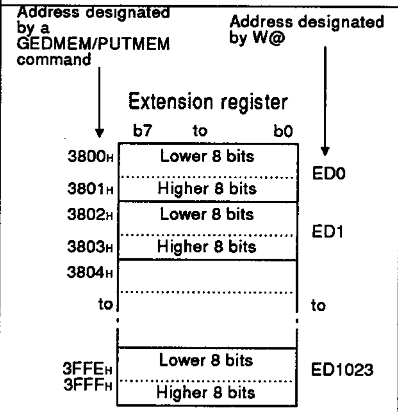
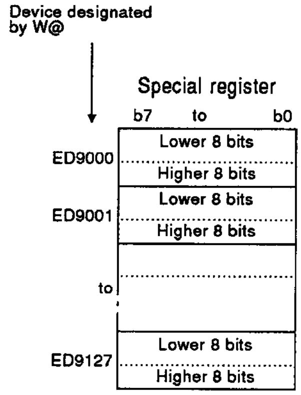
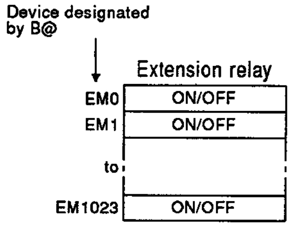
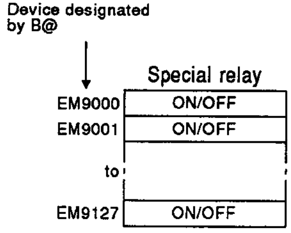
Internal Memory	Description	Memory Map	Reference Section
Execution program area	(1) Memory where a BASIC program executed by the AD51H is stored. (2) The BASIC programs are read from the EP-ROM or a memory card to the execution program area. (The copy target is set by mode setting switch 2.) (3) 64K bytes to 384K bytes can be selected. (Set in units of 64K bytes) (4) 16K, 32K, 48K, or 64K bytes can be set to the task size.	 OS area (128 bytes) Program area (max. eight tasks) Max. 384K bytes	3.7.2
Buffer memory	(1) Memory that can communicate 16 bits of data from/to a PC CPU (2) Read/write of data from/to each task is enabled. (3) Read/write of data is executed in 16-bit units by a FROM/TO command from the PC CPU. (Use X1B as the interlock signal for the execution of the FROM/TO command.) (4) Read/write of data is executed in 8-bit units by a GETMEM/PUTMEM command from a BASIC program. (5) The buffer size is 6K bytes.	 Address designated by a GETMEM/PUT-MEM command Address designated by a FROM/TO command Buffer Memory b7 to b0 0H Lower 8 bits 1H Higher 8 bits 000H 2H Lower 8 bits 3H Higher 8 bits 001H to to 17FEH Lower 8 bits 17FFH Higher 8 bits BFFH	3.7.3
Common memory	(1) Memory used in the AD51H (Read/write from/to a PC CPU is disabled.) (2) Read/write of data from/to each task is enabled. (3) Read/write of data is executed in 8-bit units by a GETMEM/PUTMEM command from/to a BASIC program. (4) The size of the common memory area is 8K bytes.	 Address designated by a GETMEM/PUT-MEM command Common memory b7 to b0 1800H 1801H 1802H 37FEH 37FFH	3.7.4

Table 3.11 AD51H Internal Memory and Memory Map (continued)

Internal Memory	Description	Memory Map	Reference Section
Extension register (ED)	(1) Memory used in the AD51H (Read/write from/to a PC CPU is disabled.) (2) Read/write of data is enabled from/to each task. (3) Read/write of data is executed by a GETMEM/PUTMEM command or special variable W@ from a BASIC program. • GETMEM/PUTMEM : 8-bit unit • W@ : 16-bit unit (4) The number of points in an extension register is 1024 (2048 bytes).	 Address designated by a GEDMEM/PUTMEM command Address designated by W@ Extension register b7 to b0 3800H Lower 8 bits 3801H Higher 8 bits ED0 3802H Lower 8 bits 3803H Higher 8 bits ED1 3804H to to 3FFEH 3FFFH Lower 8 bits Higher 8 bits ED1023	3.7.5
Special register (ED)	(1) An area with a designated use in the internal memory of the AD51H (Read/write from/to a PC CPU is disabled.) (2) Data can be read from each task. (3) Data read is executed in 16-bit units by special variable W@ from a BASIC program. (4) The number of points in the special register is 128.	 Device designated by W@ Special register b7 to b0 ED9000 Lower 8 bits Higher 8 bits ED9001 Lower 8 bits Higher 8 bits to ED9127 Lower 8 bits Higher 8 bits	3.7.6
Extension relay (EM)	(1) Relay used in the AD51H (Read/write from/to a PC CPU is disabled.) (2) Read/write of ON/OFF data from/to each task is enabled. (3) Read/write of ON/OFF data from/to a BASIC program is executed by special variable B@. (4) The number of points in the extension relay is 1024.	 Device designated by B@ Extension relay EM0 ON/OFF EM1 ON/OFF to EM1023 ON/OFF	3.7.7
Special relay (EM)	(1) An area with a designated use in the internal memory of the AD51H (Read/write from/to a PC CPU is disabled.) (2) Data can be read from each task. (3) Read of ON/OFF data from/to a BASIC program is executed by special variable B@. (4) The number of points in the extension relay is 128.	 Device designated by B@ Special relay EM9000 ON/OFF EM9001 ON/OFF to EM9127 ON/OFF	3.7.8

3.7 AD51H Internal Memory Areas

The memory areas of the AD51H, given below, are available to the user.

(1) Memory area where the execution program is stored

- Program area .... (see Section 3.7.2)

  This memory stores BASIC programs executed by the AD51H.

  This program area can be divided into eight task areas and has a total of 384K bytes.

(2) Memory area used for read/write of ON/OFF data

- Extension relay (EM) : 1024 points (EM0 to EM1023) .... (see Section 3.3.7)

(3) Memory area used to read/write 16-bit data

- Extension register (ED) : 1024 points (ED0 to ED1023) .... (see Section 3.7.5)
- Buffer : 6K bytes .... (see Section 3.7.3)
- Common memory : 8K bytes .... (see Section 3.7.4)

Since the memory areas of (1) to (3) above are not battery-backed, if the AD51H power goes OFF, written BASIC programs and all data will be cleared.

Make sure to store all created BASIC programs in the memory card in MEMORY CARD1 before turning OFF the power.

Store all necessary data to the memory card(s) in MEMORY CARD1 or MEMORY CARD2 before turning OFF the power.

3.7.1 Memory map

When executing read/write of data by using extension register (ED), a buffer, or a common memory, designate addresses by using a BASIC program GETMEM/PUTMEM command.

Figure 3.1 shows the memory maps of a memory designated by a GETMEM/PUTMEM command.

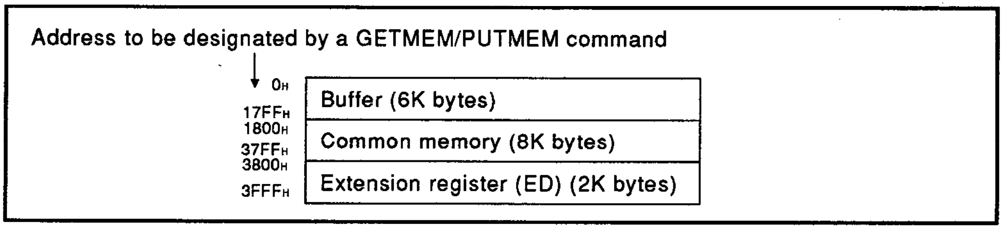

Address to be designated by a GETMEM/PUTMEM command
0H
17FFH
1800H
37FFH
3800H
3FFFH
Buffer (6K bytes)
Common memory (8K bytes)
Extension register (ED) (2K bytes)

Fig. 3.1 Memory Map

REMARK

The AD51H-BASIC Programming Manual gives details about GETMEM/PUTMEM commands.

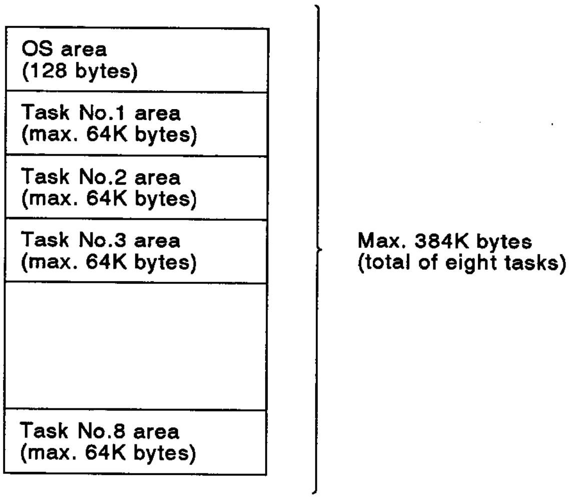
3.7.2 Program area : This is not backed up by a battery

BASIC programs created by a console and an execution program read from the memory card or the EP-ROM are stored in the program area.

(1) Memory capacity

The program area is (max.) 384K bytes.

384K bytes are divided into eight BASIC program areas (task 1 to task 8).

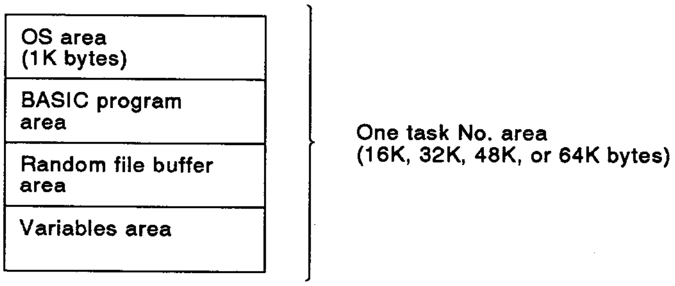


(2) BASIC program capacity

The BASIC program capacity can be set to 16K, 32K, 48K, or 64K bytes for each program.

(3) Composition of each task number area

Each task number area includes a BASIC program area, an OS area, a variables area, and a random file buffer area.



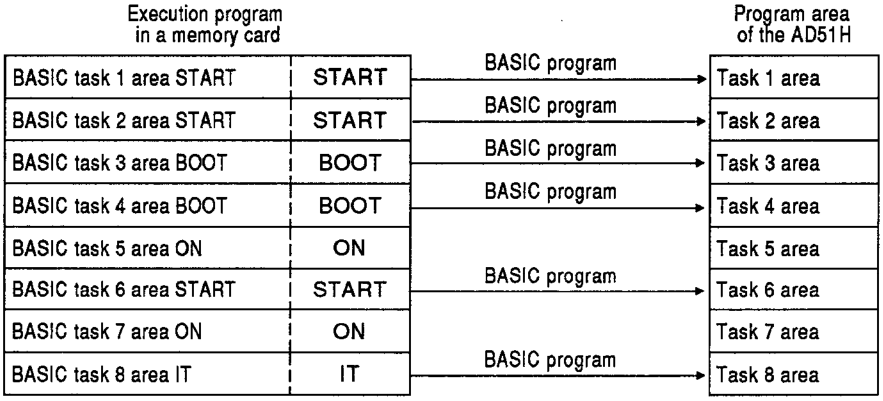
POINT

The execution program of a memory card installed in MEMORY CARD1 is read to the program area of the AD51H in the following cases:

(a) Program mode:
When the power is turned ON or the AD51H is reset

(b) Multitask debugging mode
(c) Execution mode
} When the RUN keyswitch is set to RUN after power is turned ON or the AD51H is reset

However, data in an area where multitask setting is ON will not be read to the program area of the AD51H.



Execution program in a memory card			Program area of the AD51H
BASIC task 1 area START	START	BASIC program →	Task 1 area
BASIC task 2 area START	START	BASIC program →	Task 2 area
BASIC task 3 area BOOT	BOOT	BASIC program →	Task 3 area
BASIC task 4 area BOOT	BOOT	BASIC program →	Task 4 area
BASIC task 5 area ON	ON		Task 5 area
BASIC task 6 area START	START	BASIC program →	Task 6 area
BASIC task 7 area ON	ON		Task 7 area
BASIC task 8 area IT	IT	BASIC program →	Task 8 area

3.7.3 Buffer for communications with a PC CPU : This is not backed up by a battery

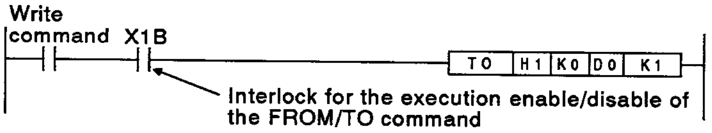
The buffer is used for data communications between the AD51H and a PC. Data written to the buffer from a PC CPU can be read during multitasking. Data written to the buffer during multitasking can be read from a PC CPU.

(1) Read/write of data

(a) Read/write of data from/to the buffer is executed by using a FROM/TO command from a PC CPU.

POINT

To read/write data from/to the buffer from a PC CPU, use the multitask execution start signal (X1B) as an interlock.


Write
command
X1B
TO
H1
K0
D0
K1
Interlock for the execution enable/disable of the FROM/TO command

(b) Read/write of data from/to the buffer of the AD51H is executed by using a GETMEM/PUTMEM command with a BASIC program.

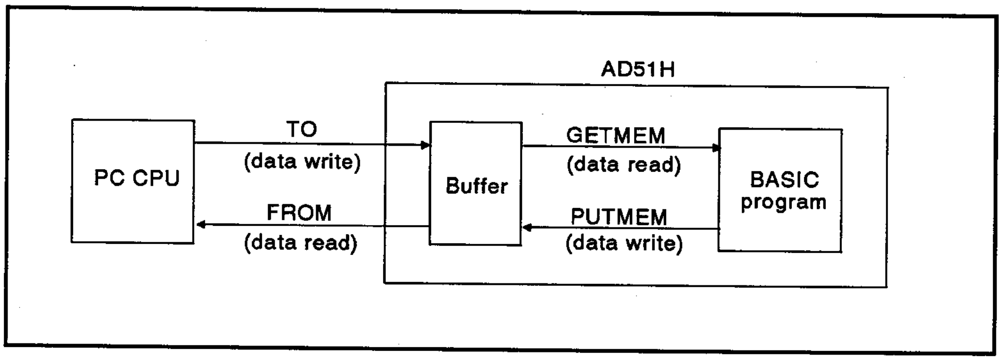


Fig. 3.2 Buffer Data Communications

(2) Buffer capacity

The AD51H buffer consists of 3K words (6K bytes).

(3) Buffer structure

A buffer address consists of 16 bits.

All areas can be used.

REMARK

The AD51H-BASIC Programming Manual gives details about GETMEM/PUTMEM commands.

(4) Buffer addresses

The buffer addresses designated by a PC CPU are different from those designated by a BASIC program, as explained below.

(a) When designating addresses from a PC CPU

Designate addresses from a PC CPU in one-word units.

Use hexadecimal numbers to designate buffer addresses 0H to BFFH.

(b) When designating addresses by a BASIC program

Designate buffer addresses by a BASIC program in one-byte units.

Use hexadecimal numbers to designate buffer addresses 0H to 17FFH.

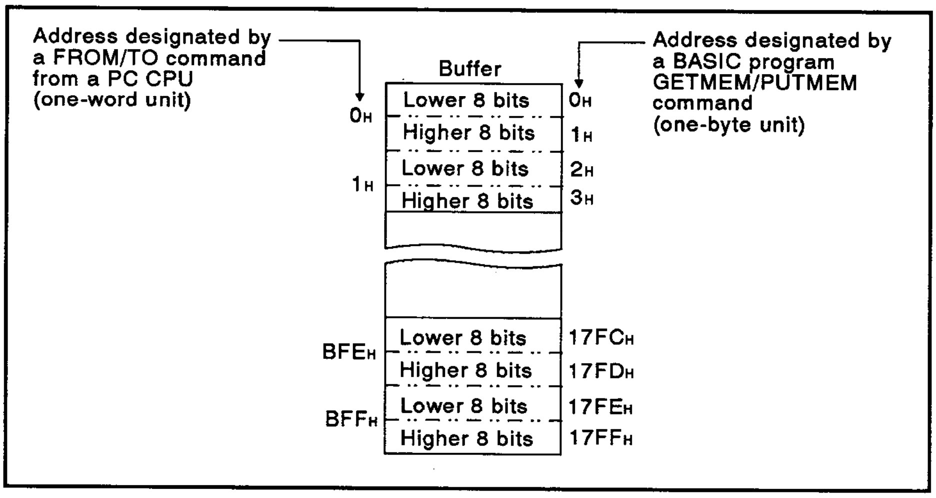


Fig. 3.3 Buffer Addresses

POINT

Among various processings of the special function module, access from the PLC CPU is processed with priority.
Therefore, making frequent access from the PLC CPU to the buffer memory of the special function module will not only increase the scan time of the PLC CPU but also delay various processings of the special function module.
Access from the PLC CPU to the buffer memory by the FROM/TO instruction or like should be made only when it is necessarv.

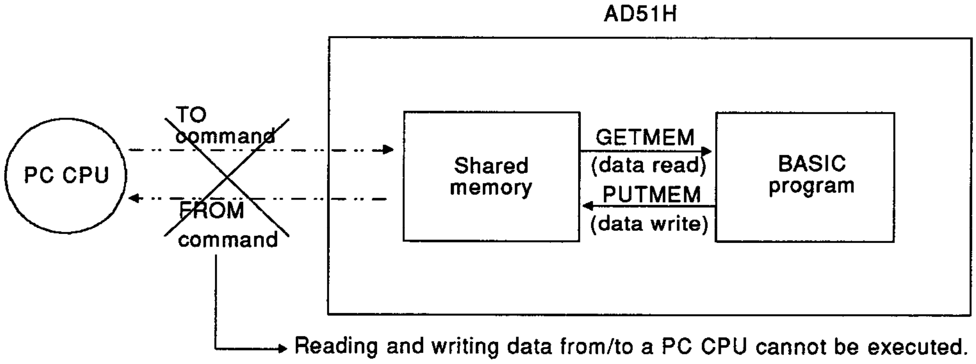
3.7.4 Common memory shared between tasks : This is not backed up by a battery

The common memory can be used for data communications between tasks in the AD51H.

Read/write of data cannot be executed from/to a PC CPU.

(1) Read/write of data from/to the common memory

(a) Use a BASIC program GETMEM/PUTMEM command to read/write data from/to the common memory.



(2) Common memory capacity

The AD51H common memory consists of 8K bytes.

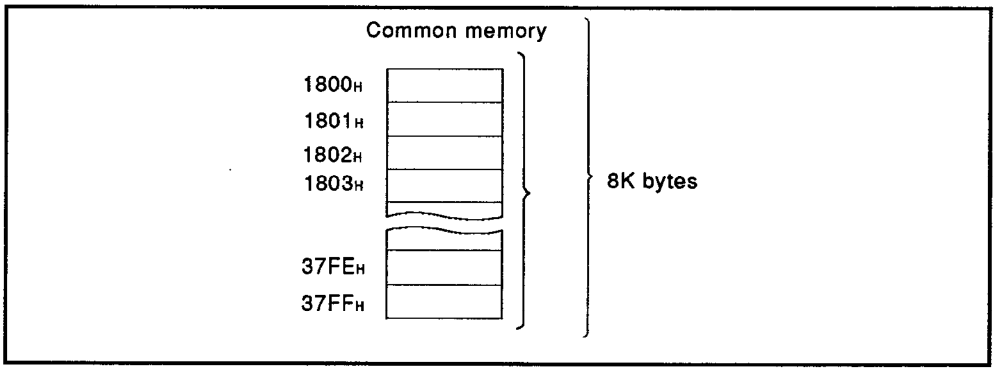
(3) Common memory structure

A common memory address consists of 1 byte (8 bits).

All areas can be used.

(4) Common memory addresses

A common memory address begins with the address (1800H) next to the buffer.



REMARK

The AD51H-BASIC Programming Manual gives details about GETMEM/PUTMEM commands.

3.7.5 Extension register (ED) : This is not backed up by a battery

The extension register is the internal memory in which one-word (16-bit) data can be stored. *1

Data can be stored in the AD51H and one-word data can be communicated between tasks by using an extension register. *2

Communications between a PC CPU and the extension register cannot be executed by using a FROM/TO command.

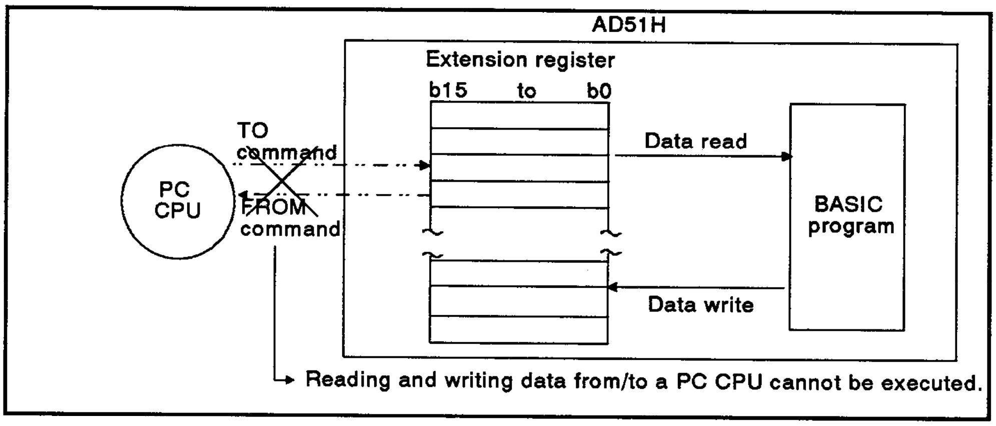


Fig. 3.4 Reading and Writing Data

(1) Reading and writing data from/to the extension register

Reading and writing of data from/to the extension register can be executed by using special variable W@ or a GETMEM/PUTMEM command.

This section gives how to designate the extension register by using special variable W@ or a GETMEM/PUTMEM command.

(a) Reading and writing of data by special variable W@

Designate the extension register from ED0 to ED1023 by using special variable W@.

Designated in one-word units.

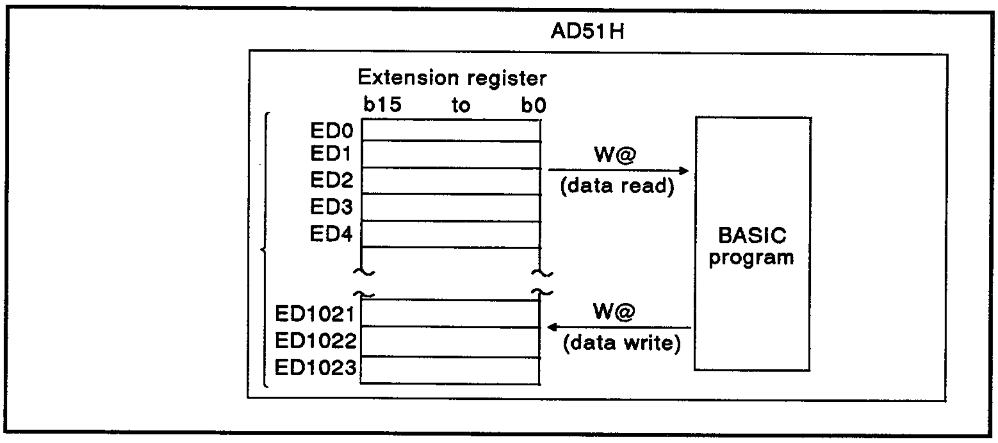


Fig. 3.5 Reading and Writing Data by Using Special Variable W@

REMARKS

(1) *1 : Since extension register data is not battery-backed, if the PC power supply is turned OFF or the PC is reset, any written data will be cleared.

Store data in a memory card before turning OFF the power or resetting the PC.

(2) *2 : The buffer (see Section 3.7.3) and common memory (see Section 3.7.4) can be used for communicating one-word data between tasks.

(b) Reading and writing data by a PUTMEM/GETMEM command

Designate an extension register address by using a PUTMEM/GETMEM command.

Designate addresses from 3800H to 3FFFH in one-byte units.

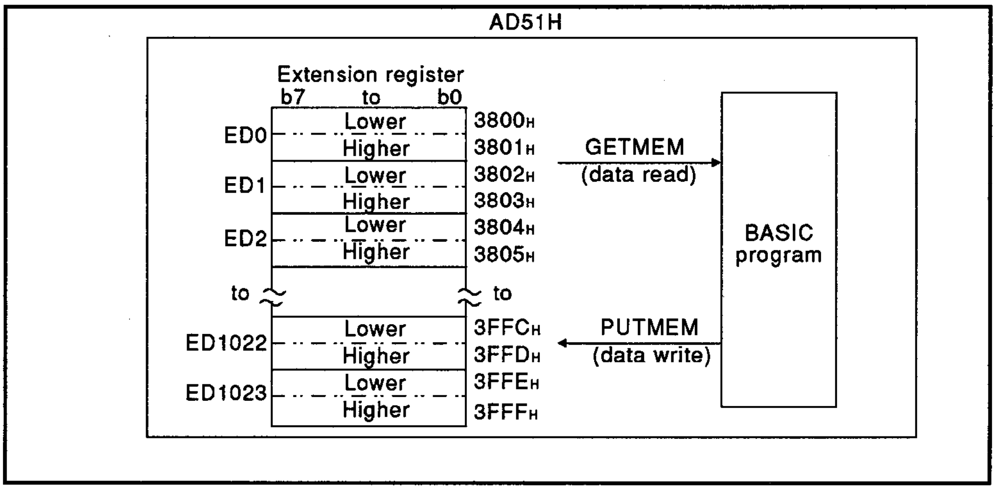


Fig. 3.6 Reading and Writing Data by Using a PUTMEM/GETMEM Command

(2) While executing BASIC programs by multitasking, data can be read or written from/to any task.

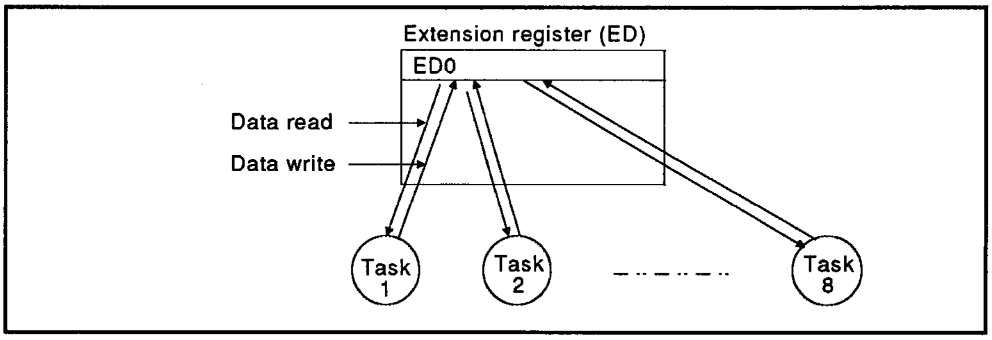


Fig. 3.7 Using an Extension Register with a BASIC Program

(3) Number of extension register points

The extension register of the AD51H has 1024 points (from ED0 to ED1023).

(Extension register numbers ED0 to ED1023 are displayed in decimal.)

3.7.6 Special Registers (ED9000 to ED9127)

Special registers are registers which can be set for specific purposes in the AD51H.

Since the OS of the AD51H stores data, such data can be read and utilized by using special variable W@ with the BASIC program.

Table 3.12 Special Registers

<table>
<tr><th>Special Register Number</th><th>Name</th><th>Description</th><th>Details</th><th>Related Special Relay</th><th>Related Special Register</th></tr>
<tr><td>ED9000</td><td>Error number</td><td>0 to 8</td><td>The error number registered by the AD51H is stored.</td><td>EM9000</td><td>ED9001 to ED9008</td></tr>
<tr><td>ED9001 to ED9008</td><td>Error code</td><td>0 : Normal
Except 0 : Error code</td><td>• Error codes displayed by the LED are stored sequentially from ED9001 to ED9008.
• If an error with more than seven factors occurs, the error code is not stored.
• An error code displayed by the LED can be cleared by the indicator reset switch.</td><td>EM9000</td><td>ED9000</td></tr>
<tr><td>ED9009 to ED9016</td><td>Error line number</td><td>0 : No line number
1 to FFFF: Line number</td><td>• The line number of an error detected by the BASIC is stored.
• Error codes and the line number storage special registers correspond as shown below.
<table>
<tr><th>For Storing an Error Code</th><th>For Storing a Line Number</th></tr>
<tr><td>ED9001</td><td>ED9009</td></tr>
<tr><td>ED9002</td><td>ED9010</td></tr>
<tr><td>ED9003</td><td>ED9011</td></tr>
<tr><td>ED9004</td><td>ED9012</td></tr>
<tr><td>ED9005</td><td>ED9013</td></tr>
<tr><td>ED9006</td><td>ED9014</td></tr>
<tr><td>ED9007</td><td>ED9015</td></tr>
<tr><td>ED9008</td><td>ED9016</td></tr>
</table>
{When an error is detected by the BASIC, the error code and the line number are alternately displayed by the LED.}
• A line number displayed by the LED can be cleared by the indicator reset switch.</td><td>EM9000</td><td>ED9001 to ED9008</td></tr>
<tr><td>ED9020</td><td>Execution modes</td><td>0 to 4</td><td>• The setting number of the mode setting switch is stored.
0, 1 : Execution mode
2, 3 : Multitask debug mode
4 : Programming mode</td><td>—</td><td>—</td></tr>
</table>

Table 3.12 Special Registers (continued)

Special Register Number	Name	Description	Details	Related Special Relay	Related Special Register
ED9021	Memory card state	0 : Memory card access switch ON 1 : Memory cards cannot be inserted or removed. 2 : Memory cards can be inserted or removed.	• ON/OFF states of the memory card access switch and memory card insertion/removal are stored. • 0 : When the memory card access switch is ON and a memory card can be accessed • 1 : The memory card access switch is turned OFF, or the memory card cannot be inserted/removed because the memory card is being accessed or opened. • 2 : The memory card can be inserted/removed after "1" is stored.	EM9021	—
ED9022	BASIC task states	0 : The BASIC task is in the STOP state. 1 : The BASIC task is in the execution, STOP, or wait state.	• The states of the BASIC tasks 1 to 8 are stored as follows: b15 to b8 b7 to b0 0 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 BASIC task 8 BASIC task 7 BASIC task 6 BASIC task 5 BASIC task 4 BASIC task 3 BASIC task 2 BASIC task 1 • 0 : The BASIC task is in the STOP state. (BASIC program is not being executed.) • 1 : The BASIC task is in the execution state. (BASIC program is being executed.) • BASIC task is in the STOP state. (All BASIC tasks are not being executed.) • BASIC task is in the wait state. (The BASIC task is waiting for a command (PRINT command, etc.) to be completed, or the execution interruption command was executed.) • When the bit of a designated task is "0", start the BASIC task using the ZSTART command.	—	—

3.7.7 Extension relay (EM) : This is not backed up by a battery

Since the extension relay is the relay used for communicating ON/OFF data between tasks in the AD51H, it cannot output ON/OFF data to an external device.

(1) ON/OFF state of the extension relay and reading of ON/OFF data

(a) The BASIC program reads the ON/OFF state of the extension relay and turns ON/OFF the extension relay using special variable B@.

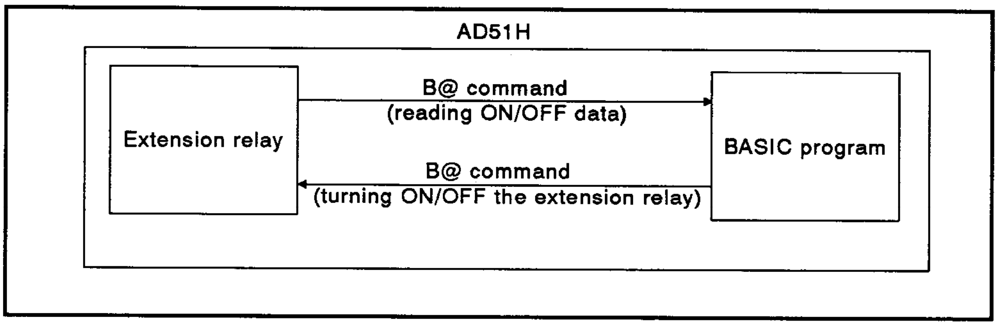


Fig. 3.8 Turning ON/OFF the Extension Relay and Reading ON/OFF Data

(b) ON/OFF data can be read from any task and the data can be used with a BASIC program while doing multitasking of the BASIC program when the BASIC programs are being executed by multitasking.

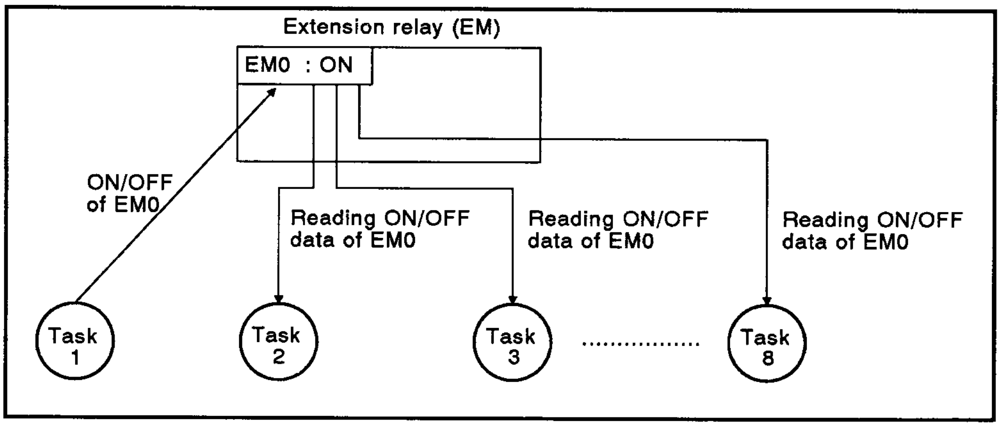


Fig. 3.9 ON/OFF State of the Extension Relays Controlled by the BASIC Programs

(2) Number of extension relay points

The AD51H extension relay has 1024 points.

REMARK

The AD51H-BASIC Programming Manual gives details about special variable B@.

3.7.8 Special Relays (EM9000 to EM9127)

Special relays are relays which can be set for specified purposes in the AD51H.

Since the OS of the AD51H executes ON/OFF control, the ON/OFF state can be read and utilized by using special variable B@ with a BASIC program.

Table 3.13 Special Relays

Special Relay Number	Name	Description	Details	Related Special Register
EM9000	Error registration signal	ON : Error registration OFF: No error registration	This goes ON when a warning error, error message, or system-down error message is displayed on the LED.	ED9000 to ED9016
EM9020	Operation signal	ON : Multitasking OFF: Except multitasking	• Turned ON when the AD51H is multitasking. • Turned OFF when the AD51H is not multitasking.	—
EM9021	Memory card access permission signal	ON : Access is enabled. OFF: Access is disabled.	• This goes ON when the memory card access switch is turned ON. Accesses the memory card when EM9021 is ON. • This goes OFF when the memory card access switch is turned OFF.	—

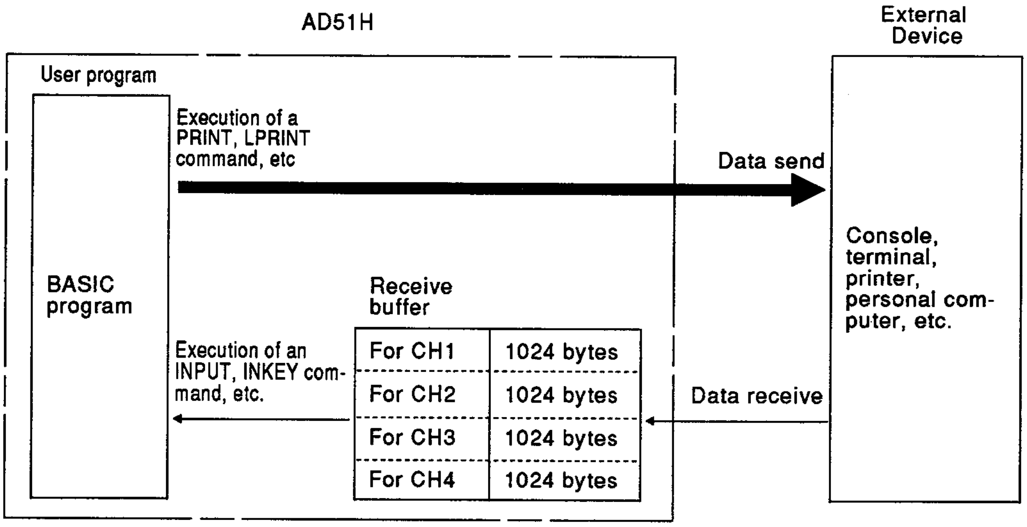
3.8 Receive Buffer

The receive buffer is used to receive and transmit data from/to an external device connected to the RS-232C interface, the RS-422 interface, or the parallel interface of the AD51H.

The OS of the AD51H controls the communications buffer when these interfaces are used.

Each interface is assigned 1024 bytes of receive buffer.

The receive buffer stores data transmitted from the external device to the AD51H. Then, buffer data is stored in a designated variables area by an INPUT or INKEY command.



POINT

The receive buffer area can be changed by the ZCNTL command of the AD51H-BASIC.

3.9 I/O for a PC CPU

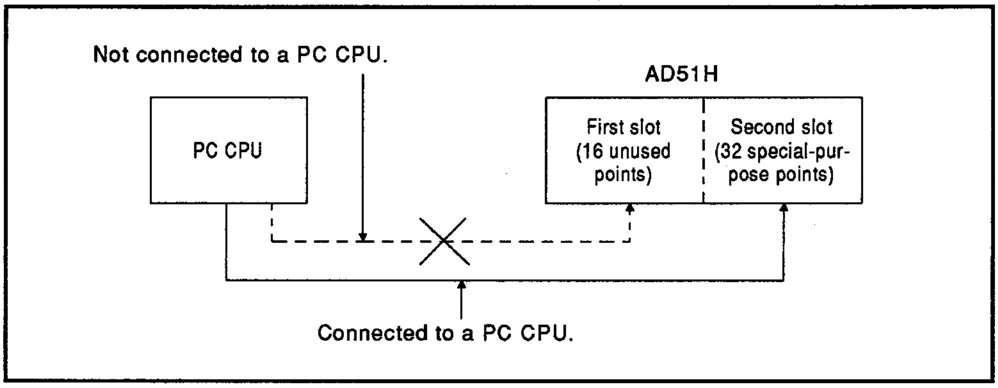
The AD51H I/O is used to (a) communicate ON/OFF data between a PC CPU and the AD51H, and (b) start a designated task from the PC.

Use device X to send an input signal from the AD51H to a PC CPU.

Use device Y to send an output signal from a PC CPU to the AD51H.

The AD51H uses two slots - the first slot has 16 unused points and the second slot has 32 special-purpose points.

Communications between the AD51H and a PC CPU can be executed by using these 32 points.


Not connected to a PC CPU.
AD51H
PC CPU
First slot (16 unused points)
Second slot (32 special-purpose points)
Connected to a PC CPU.

3.9.1 I/O (X, Y) list

Fig. 3.10 shows how to determine a device when an AD51H is installed in slot number 1 or 0 of a main base unit.

- The I/O number used for a sequence program differs depending on the slot in which the AD51H is installed.
- The value designated by special variable B@ of the BASIC program of the AD51H is specified to &H00 to &H1F.

Table 3.14 shows device numbers and signal names.

Sections 3.9.2 to 3.9.3 give details about these signal names.

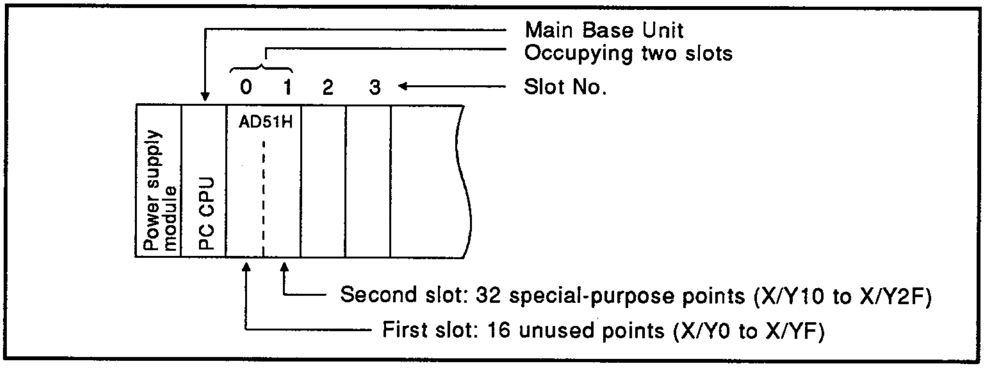

Main Base Unit
Occupying two slots
0
1
2
3
Slot No.
Power supply module
PC CPU
AD51H
Second slot: 32 special-purpose points (X/Y10 to X/Y2F)
First slot: 16 unused points (X/Y0 to X/YF)

Fig. 3.10 Device Number set by a PC CPU When the AD51H is Installed in Slot 0 or 1

Table 3.14 I/O List of the AD51H

Signal Direction : AD51H → PC CPU			Signal Direction : PC CPU → AD51H			
Input Number Used by a PC CPU	Value Designated by Special Variable B@	Signal Name	Input Number Used by a PC CPU	Value Designated by Special Variable B@	Signal Name	
X00 to X0F	—	Unusable	Y00 to Y0F	—	Unused (Available as the substitution for internal relay (M).)	
X10 to X1A	&H00 to &H0A	General-purpose input	Y10 to Y20	&H00 to &H10	General-purpose output	
X1B	&H0B	Start of multitasking				
X1C	&H0C	Stop of multitasking				
X1D	&H0D	AD51H system down				
X1E X1F	&H0E &H0F	Unusable				
X20 to X2F	&H10 to &H1F	General-purpose input	Y21	&H11	Designation of a general-purpose output/start BASIC task No. *	BASIC task 1
			Y22	&H12		BASIC task 2
			Y23	&H13		BASIC task 3
			Y24	&H14		BASIC task 4
			Y25	&H15		BASIC task 5
			Y26	&H16		BASIC task 6
			Y27	&H17		BASIC task 7
			Y28	&H18		BASIC task 8
			Y29	&H19	Program start signal (unusable for a general-purpose output)	
			Y2A to Y2F	&H1A to &H1F	Unusable	

REMARK

* Designate the start task number if a task is set to "start by the interrupt from PC CPU" by multitask setting.
Designate a general-purpose output if the task is not set to "start by the interrupt from PC CPU".

3.9.2 Detailed explanation of input (X)

This section explains input (X) of the AD51H.

The input numbers (Xn) used in this section are the numbers set by a PC CPU.

(1) General-purpose inputs (X10 to X1A, X20 to X2F: 27 points)

The user can use general-purpose inputs for transmitting ON/OFF data from the AD51H to a PC CPU.

Use special variable B@ to turn general-purpose inputs ON or OFF.

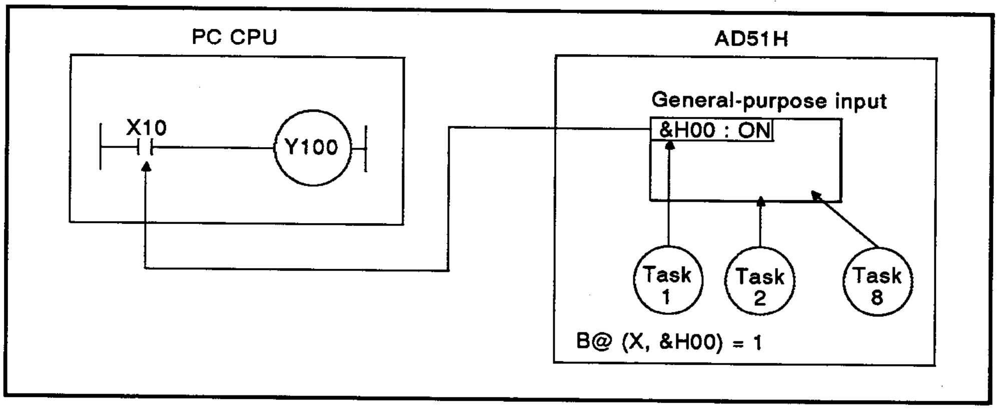


Fig. 3.11 Communications with a PC CPU Using a General-Purpose Input

(2) Multitasking start flag (X1B)

(a) When mode setting switch 1 *1 of the AD51H is set to "execution mode (0 or 1)", this flag becomes valid (i.e., ON/OFF control is executed.).

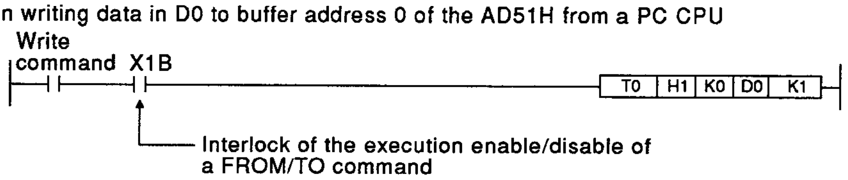
(When the AD51H is set to other modes, this is in the OFF state.)

(b) It goes ON if (a) the AD51H is in the execution mode, or (b) the AD51H goes into the multitask execution enable state after the power is turned ON or the AD51H is reset.

Use this flag as the execution enable/disable interlock signal of the FROM/TO instruction for data communications with the AD51H buffer (see Section 3.7.3) using a sequence program.

When a FROM/TO instruction is executed from a PC CPU, if the AD51H is not in the multitask execution enable state, an "SP UNIT DOWN" error occurs, and the PC CPU stops operating. *2

When writing data in D0 to buffer address 0 of the AD51H from a PC CPU


Write
command
X1B
TO
H1
K0
D0
K1
Interlock of the execution enable/disable of a FROM/TO command

(c) This flag goes OFF in either of the following cases when the AD51H is in the execution mode.

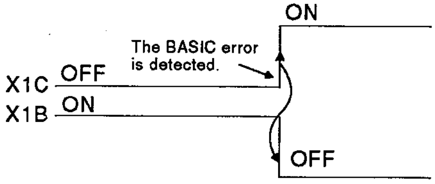
1) RUN keyswitch of the AD51H is in the STOP position.

2) RUN keyswitch of the AD51H was reset.

(When the AD51H completes reset processing and returns to the multitask execution enable state, this signal goes ON.)

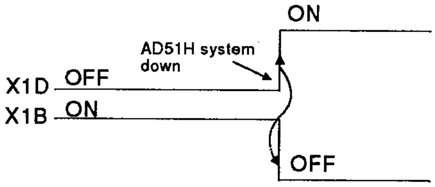
3) A BASIC error is detected.

(Multitask execution stop flag : X1C turns ON.)



4) AD51H system down

(AD51H down flag : X1D goes ON.)



REMARKS

*1 : Sections 5.1 and 5.2.1 give details about mode setting switch 1.

*2 : When the multitask execution start flag (X1B) is turned OFF by resetting the AD51H, if a FROM/TO command is executed within the access enable time of a set FROM/TO command, an error does not occur.

(3) Multitask execution stop flag (X1C)

(a) When mode setting switch 1 of the AD51H is set to "execution mode (0 or 1)", this flag becomes valid. (ON/OFF control is executed.)

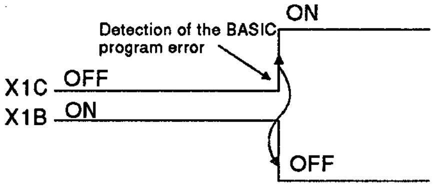
(If the AD51H is set to another mode, this flag goes OFF.)

(b) If a BASIC program error is detected during multitasking, this signal goes ON.

(At this time, error step and error code * numbers are displayed on an LED, and the mode is switched to the system mode.)

(Correct the BASIC program error.)

(c) If the multitask execution stop flag (X1C) goes ON, the multitask execution start flag (X1B) goes OFF.



(4) AD51H system-down flag (X1D)

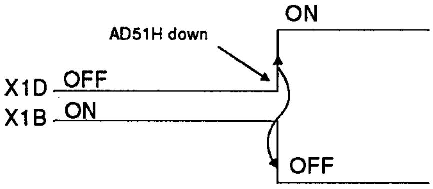
(a) When the execution mode setting switch 1 of the AD51H is set to "execution mode (0 or 1)", this flag becomes valid. (ON/OFF control is executed.)

(b) If the error that caused the system-down message during multitasking is detected, this goes ON.

At this time, an error code signal is displayed on the LED. *

If the AD51H is reset by setting the RUN keyswitch to position of RESET, the AD51H system-down error signal is cleared.

(c) When the AD51H system-down flag goes ON, the multitask execution start flag (X1B) goes OFF.



REMARK

* Section 5.6 gives details about displayed error codes.

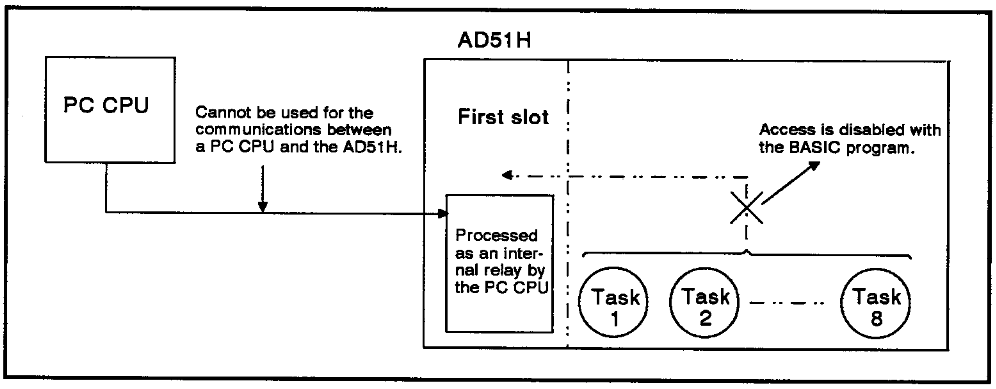
3.9.3 Detailed explanation of output (Y)

This section explains output (Y) of the AD51H.

Output numbers (Yn) used in this Section are the numbers set by a PC CPU.

(1) Unused area (16 points)

These 16 points are in the first slot of the AD51H. These outputs cannot be accessed by a BASIC program.


AD51H
PC CPU
Cannot be used for the communications between a PC CPU and the AD51H.
First slot
Access is disabled with the BASIC program.
Processed as an internal relay by the PC CPU
Task 1
Task 2
Task 8

When output (Y) is turned ON or OFF by a PC CPU, this area can be used as a substitute for internal relay (M).

(2) General-purpose outputs (Y10 to Y20 : 17 points)

A general-purpose output can be used for transmitting ON/OFF data from a PC CPU to the AD51H.

A PC CPU can turn ON/OFF general-purpose outputs using a sequence program or a test function of a peripheral device.

The AD51H uses special variable B@ to read general-purpose output data turned ON/OFF by a PC CPU.

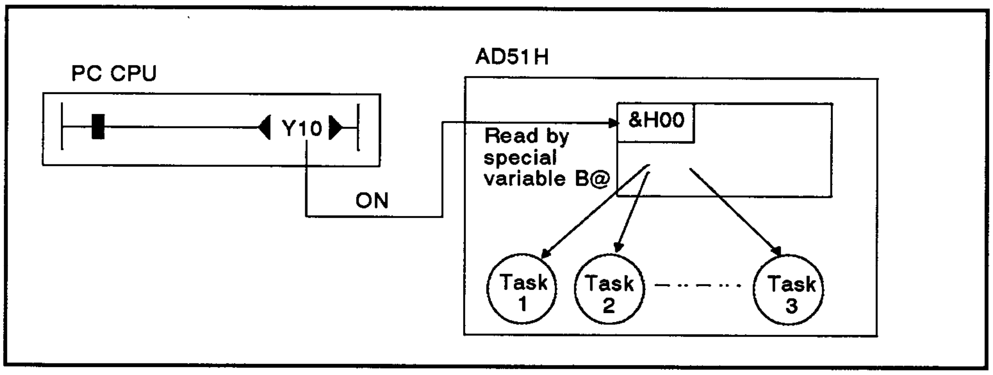


Fig. 3.12 Reading of General-Purpose Output Data Turned ON/OFF by a PC CPU

(3) Start task number designation signals (Y21 to Y28: 8 points)

(a) These signals designate tasks to be started by an interrupt from a PC CPU.

Each signal is allocated to each task from task numbers 1 to 8.

(b) Only a task set to "IT (start by the interrupt from a PC CPU)" by multitask setting can be a start task number designation signal.

Device number of the task that is not set by multitask setting becomes the same with the allocation of a general-purpose output.

Device No.	Multitask Setting		
	Start by an Interrupt from a PC CPU		Others
Y21	Setting of the task No. to be started	Task No. 1	General-purpose output (tasks cannot be started.)
Y22		Task No. 2	
Y23		Task No. 3	
Y24		Task No. 4	
Y25		Task No. 5	
Y26		Task No. 6	
Y27		Task No. 7	
Y28		Task No. 8	

(c) Turn ON the device number (Y21 to Y28) which corresponds to the task which is started.

When task start signal (Y29) goes ON, start the designated task (task that was turned ON by Y21 to Y28).

However, if the task that is being executed is designated, (Y29) is ignored.

(d) Turn OFF the start task number designation signal that was turned ON after confirming that the designated task is executed.

Task starting can be confirmed by turning ON a general-purpose input (X) allocated to a reset (OFF) command by a BASIC program (as shown in the example on the following page).

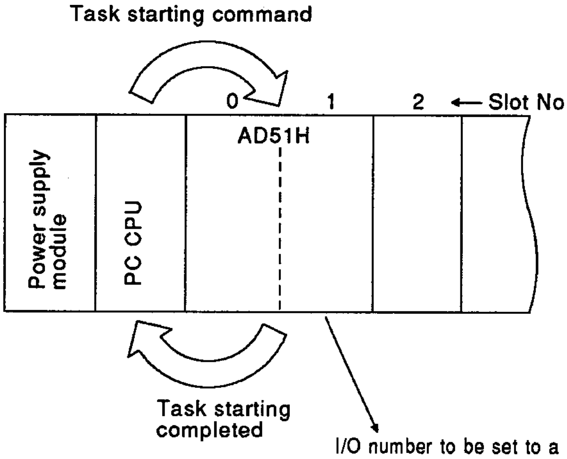
Example:

This example shows the procedure for (a) starting a task, and (b) turning OFF the start task number designation signal/task start signal.

[Conditions]

1) Start task number: 1

2) Start task No. designation signal/task start signal reset command: X21



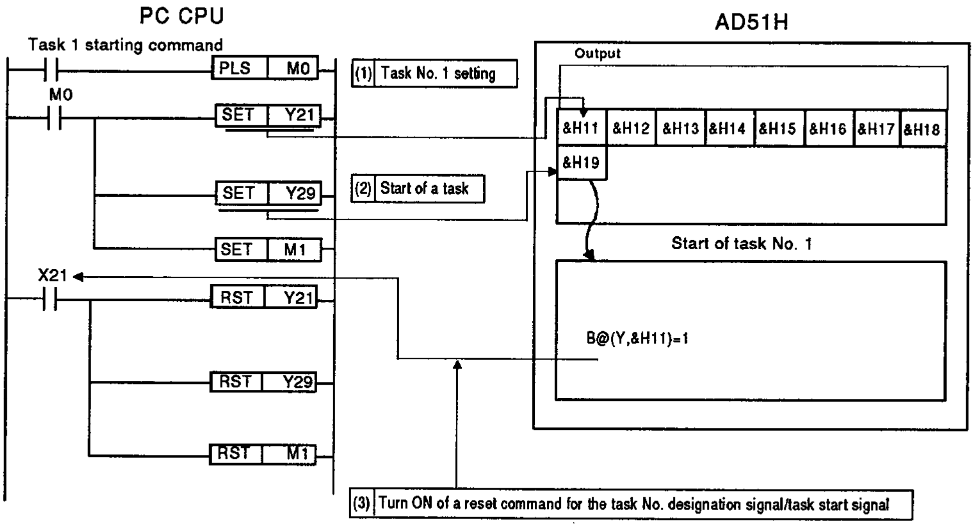
I/O number to be set to a PC CPU : X/Y10 to X/Y2F
Value to be designated by special variable B@ : X/Y&H00 to X/Y&H1F

[Procedure]

1) Task setting of number 1:
(Y21) is turned ON by a sequence program.

2) Starting the task:
(Y29) is turned ON by a sequence program.

3) The start task number designation signal/task start signal go OFF:
X&H11 are turned ON by the BASIC program.



(4) Task start signal (Y29)

(a) This signal starts the task designated by a start task number designation signal (Y21 to Y28).

Even if no task has been started by an interrupt from a PC CPU, this task start signal cannot be used as a general-purpose output.

(b) The task start signal is valid only when the multitask execution enable signal (X1B) is turned ON.

(c) When a task start signal goes ON, the (X1B) signal starts a designated task. *

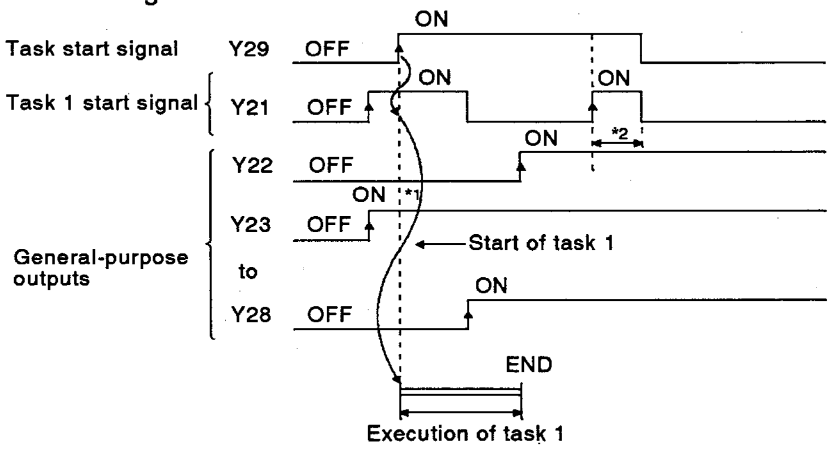
(When the task start signal is ON, if the task number designation signal is turned ON, the (X1B) signal is ignored.)

Example:

This example indicates a program to start task 1 from a PC CPU.

[Condition]

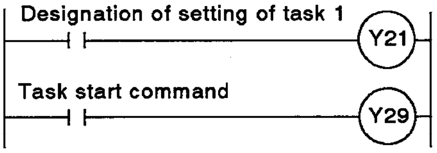
Set only task 1 to "IT (start by the interrupt from a PC CPU)" by multitask setting.



*1 : The task is not started because this is processed as a general-purpose output.

*2 : Task 1 is not started because a task start signal does not go ON.

[Sequence program]



(d) When a task designated by a start task number designation signal (Y21 to Y28) is being started, the start for any other task is ignored.

(The task currently being executed is continued.)

REMARK

* This is the task that corresponds to device number being turned ON (Y21 to Y28).

However, any device number that is not set to "IT (start by the interrupt from a PC CPU)" by multitask setting is ignored.

3.10 Relationship between the conditions for starting the AD51H and the execution program area of a memory card

When storing a BASIC program to a memory card by an MSAVE system command, BOOT is set as the condition for starting each task during multitasking. To set the starting condition to other than BOOT, use the SET system command.

Conditions for storing BASIC programs to a memory card vary according to the starting condition, as shown in Table 3.15.

Table 3.15 Relationship between the AD51H Starting Conditions and the Execution Program Areas of Memory Cards

Starting Condition		BASIC Program Storage Area	BASIC Program Read Timing	Reference Section
Multitask Setting	Symbol			
Started by multitask execution	START	• Execution program area of a memory card (the designated BASIC task number area) • Execution program area of the EP-ROM (the designated BASIC task number area)	• When power goes ON or is reset in the system mode • During multitasking in the execution or multitask debugging mode (RUN keyswitch : RUN)	3.13.2
Started by an interrupt from a PC CPU	IT	• Execution program area of a memory card (the designated BASIC task number area) • Execution program area of the EP-ROM (the designated BASIC task number area)		3.13.3
Started by other start requests from the task	BOOT	• Execution program area of a memory card (the designated BASIC task number area) • Execution program area of the EP-ROM (the designated BASIC task number area)		3.13.4
	ON	• File area in the memory card (The BASIC programs are stored as files.) • File area of the user's FD and the HD of a PC/AT (The BASIC programs are stored as files.)	• When executing by the ZSTART command (The program in the file name designated by the ZSTART command is read.)	3.13.4
Without multitask setting	OFF	—	• Reading of a BASIC program is not executed because there is no multitask setting	—

POINT

The BOOT and ON starting conditions are defined as follows.

- BOOT : The execution program is stored in the designated BASIC task number area of the memory card.
- ON : Only the memory capacity of the designated BASIC task number area of the memory card is set. The execution program is not stored in the BASIC task number area.

REMARK

This manual (online programming debug) gives details about MSAVE and SET system commands.

3.11 Memory Card Specifications, Memory Map, and Memory-Protect Range

This section explains the specifications of the memory cards that can be used with the AD51H, the memory map, and the memory-protect range.

3.11.1 Memory card specifications

Table 3.16 gives memory card specifications.

Table 3.16 Memory Card Specifications

Item	Memory Card Model Name		
	A6MEM-128KAW	A6MEM-256KAW	A6MEM-1024KAW
Memory capacity (bytes)	256K	512K	1024K
Access time (ns)	200		
Memory-protect	Enabled (by the WRITE PROTECT switch of the memory card)		
Backup function	By a built-in battery Battery: BR2325 *1		
Battery life (years)	5		
Outside dimensions	86.5 (3.40)(H) x 54 (2.12)(W) x 3 (0.12)(D)		

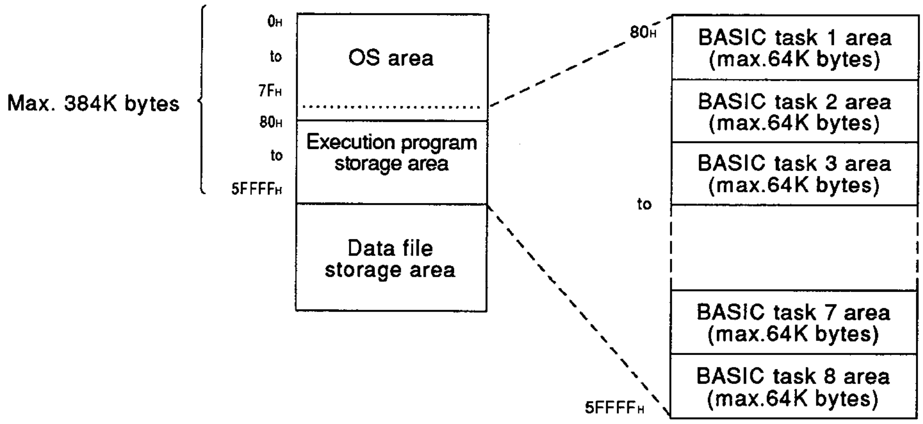
3.11.2 Memory map

(1) Memory map of a memory card installed in MEMORY CARD1

The execution program and file where a BASIC program is registered can be stored in the memory card installed in MEMORY CARD1.

(a) An execution program with (max.) eight tasks (384K bytes) can be stored in a memory card.

(b) Any area (except the execution program storage area) can be used for a BASIC program data file.


Max. 384K bytes
0H
to
7FH
80H
to
5FFFFH
OS area
Execution program storage area
Data file storage area
80H
to
5FFFFH
BASIC task 1 area (max.64K bytes)
BASIC task 2 area (max.64K bytes)
BASIC task 3 area (max.64K bytes)
BASIC task 7 area (max.64K bytes)
BASIC task 8 area (max.64K bytes)

REMARK

*1 When purchasing a new battery, consult your nearest Mitsubishi representative for details

3.11.3 Memory-protect range

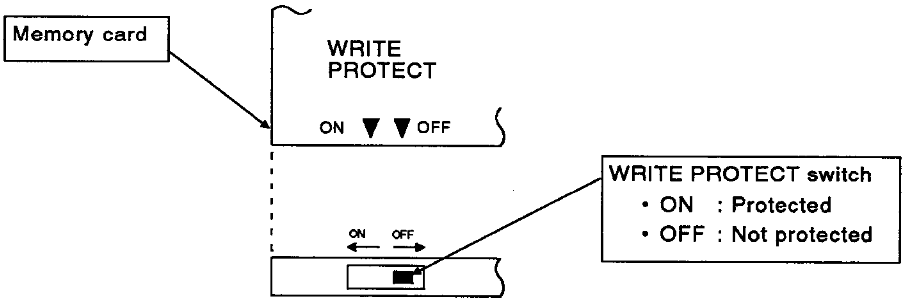
Set the memory-protect range by using either of the following methods:

- Batch protect by the memory card WRITE PROTECT switch
- Split protect by the AD51H memory-protect range setting switch

(1) Batch protect by the WRITE PROTECT switch of a memory card

The memory of a memory card is protected by the WRITE PROTECT switch of the memory card.

- If the WRITE PROTECT switch is set to ON, the memory in all areas is protected.
- If the WRITE PROTECT switch is set to OFF, the memory is not protected.

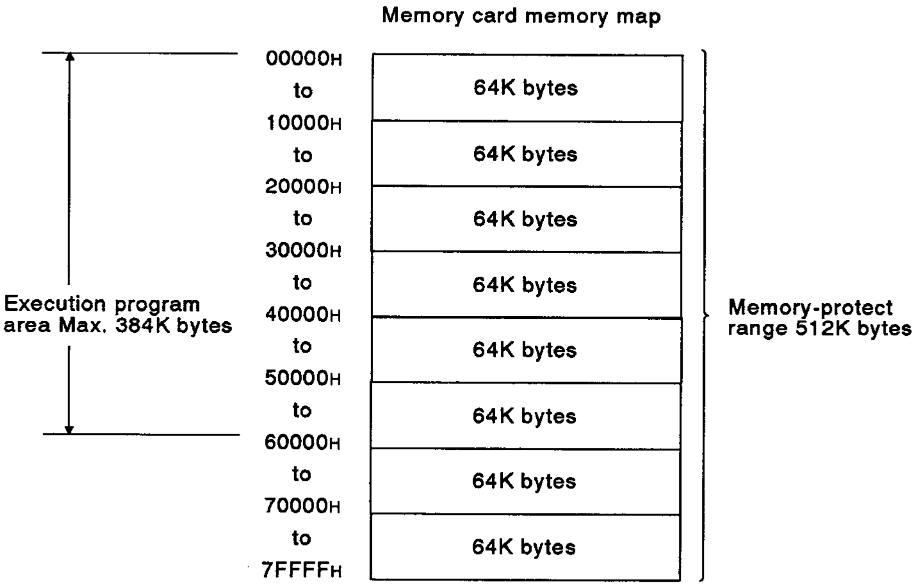


(2) Split protect by a memory protect range setting switch of the AD51H

There is a memory-protect range setting switch on a memory card installed in MEMORY CARD1 on the left side of the AD51H.

The memory-protect range has 512K bytes from address 0H to 7FFFH, and the range can be divided into 64K-byte areas.

The memory-protect range is as shown below.



3.12 AD51H Operations When the Power is Turned ON

When power to the AD51H goes ON, the operations change according to the following modes (set by mode setting switch 1).

(1) Programming mode

(2) Multitask debugging mode

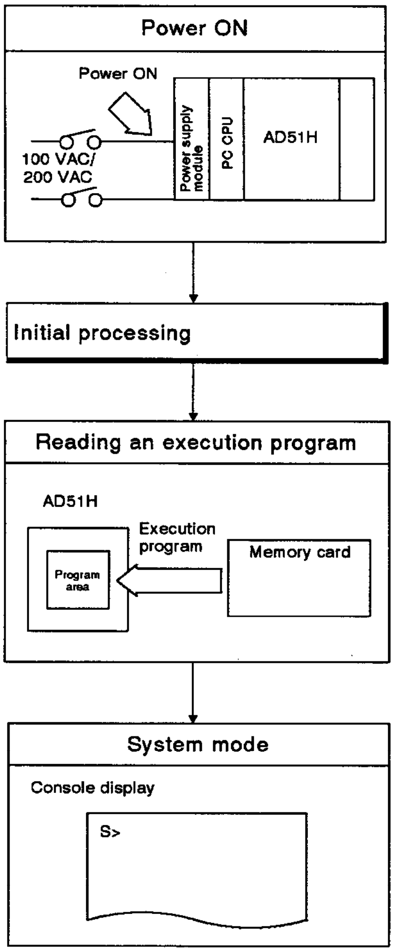
(3) Execution mode

This section explains the operations of the AD51H in the modes set by mode setting switch 1 when the power goes ON.

3.12.1 Programming mode

If the power supply is turned ON or the AD51H is reset in the programming mode, the mode is switched to the system mode after reading an execution program from the memory card to the program area of the AD51H.

(see Section 3.7.2.)



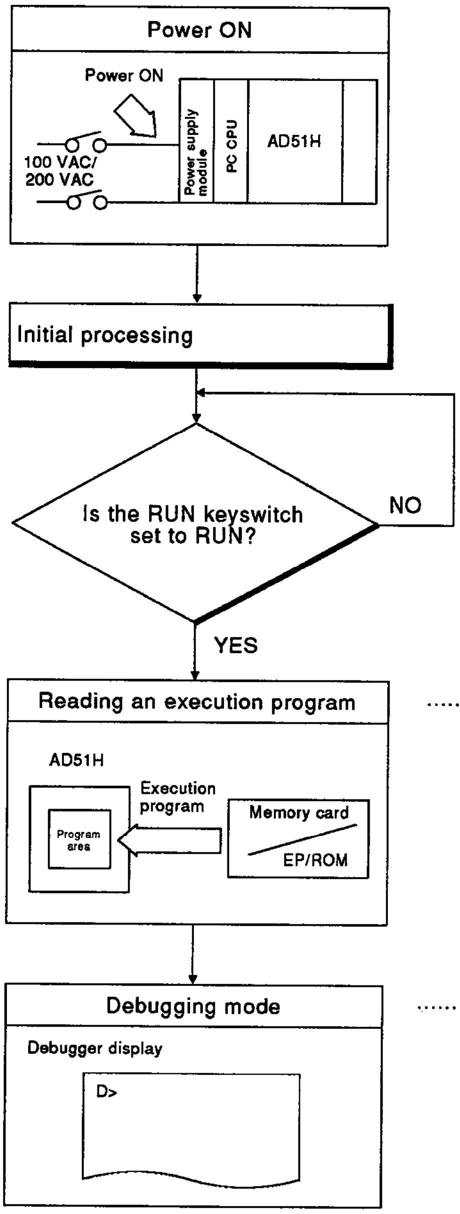
·········· The execution program of a memory card is read to the program area of the AD51H.

·········· Operation of a console is enabled when the system prompt is displayed.

3.12.2 Multitask debugging mode

After (a) turning ON the power supply or resetting the AD51H in the multitask debugging mode, and (b) when the RUN keyswitch is set to RUN, an execution program is read from a memory card or the EP-ROM to the program area of the AD51H, and the mode is switched to the debugging mode.

(see Section 3.7.2 .)



·········· The execution program is read to the execution program area of a memory card or the EP-ROM.

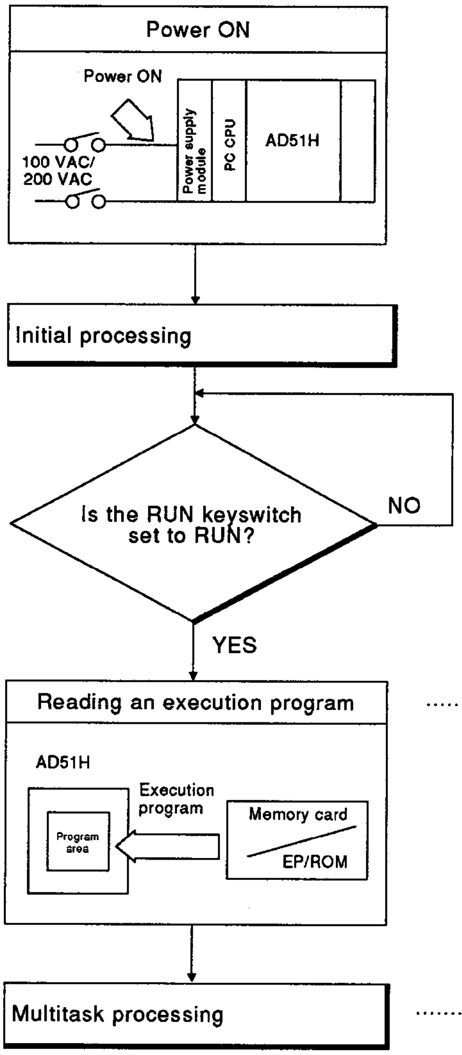
·········· The multitask operation by a debugger is enabled when a debug command is received.
A read execution program is executed according to the contents of multitask setting.

- Started by turning ON the power or resetting the AD51H
- Started by an interrupt from a PC CPU
- Started by other start requests from a BASIC program

3.12.3 Execution mode

After(a) turning ON the power supply or resetting the AD51H in the execution mode, and (b) when the RUN keyswitch is set to RUN, the execution program is read from the memory card or the EP-ROM to the program area of the AD51H, and the multitask processing is executed.

(see Section 3.7.2.)



Reads the execution program to the execution program area of a memory card or the EP-ROM.

A read execution program is executed according to the multitask setting contents.

- Started by turning ON the power or resetting the AD51H
- Started by an interrupt from a PC CPU
- Started by other start requests from a BASIC program

3.13 Executing Tasks (Created BASIC Programs)

This section explains the execution of tasks when the AD51H is in the execution mode or multitask debugging mode.

3.13.1 Conditions for starting and executing tasks

When all the following conditions are satisfied, the AD51H goes into the task execution enabled state.

(The actual start of a task depends on the starting condition when multitasking is set.)

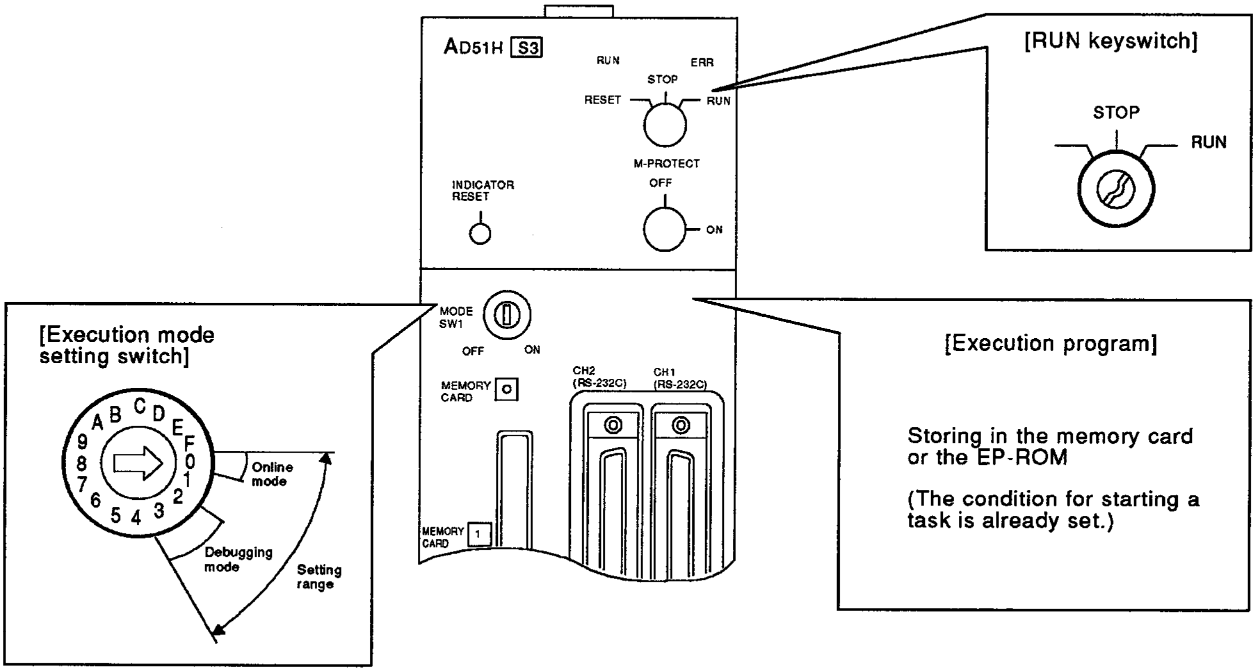
(1) Task execution conditions

The following gives the conditions for starting a task:

(a) A task starting condition is set with multitask setting.

(b) The mode setting switch 1 is set to "online mode (0, 1)" or "debugging mode (2, 3)".

(c) The RUN keyswitch is set to RU



Storing in the memory card or the EP-ROM

(The condition for starting a task is already set.)

(2) Condition for starting a task

Each task is executed according to the set starting condition.

The following starting conditions can be set with the AD51H.

(a) Starting by turning ON the power or resetting the AD51H (see Section 3.13.2)

(b) Starting by an interrupt from a PC CPU (see Section 3.13.3)

(c) Starting by other start requests from a BASIC program (see Section 3.13.4)

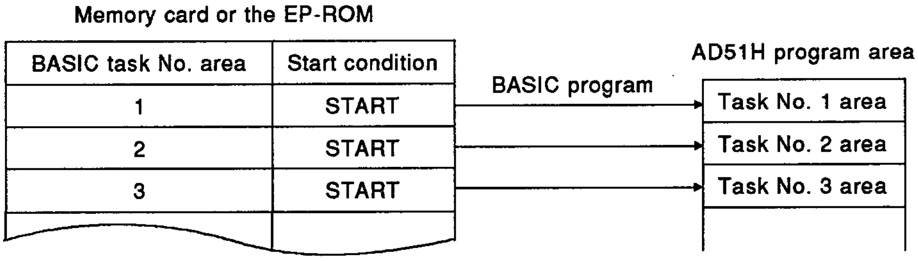
3.13.2 Starting by turning ON the power or resetting the AD51H(Start condition:START)

In this case the starting condition is set to START by the multitask setting.

When the execution conditions explained in Section 3.13.1 have been satisfied, the task is started.

(1) Processing when the power is turned ON or reset

When the power is turned ON or reset, the designated task is executed after reading an execution program stored in a memory card or the EP-ROM to the program area of the AD51H.



Memory card or the EP-ROM

BASIC task No. area	Start condition
1	START
2	START
3	START

(2) Execution of a task

A task started when the power is turned ON or reset can be executed by the created program as shown below.

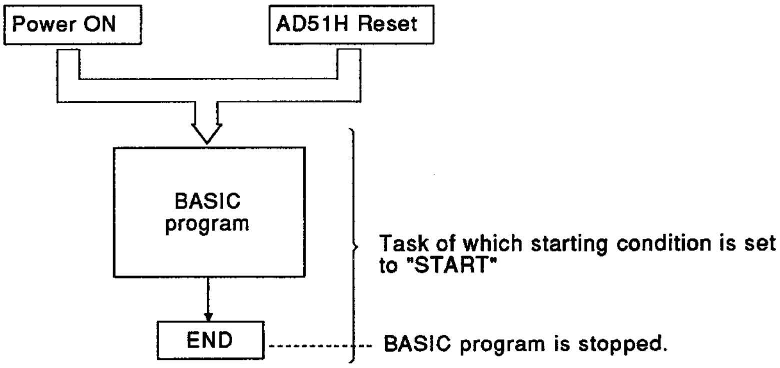
1) Executed only once

2) Always executed

(a) When the task is executed only once

If an END command is written at the end of a BASIC program, the task will only be executed once.

The task is started by turning ON the power or resetting the AD51H. When the END command has been executed, the operation stops.



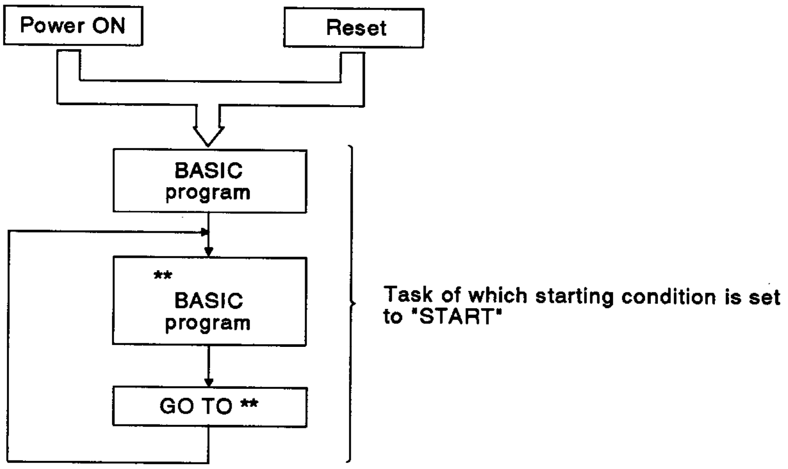
Power ON
AD51H Reset
BASIC program
END
Task of which starting condition is set to "START"
BASIC program is stopped.

(b) When the task is always executed

In this case, the program operation is jumped to a designated line to the end of a BASIC program by using the GOTO command (instead of the END command).

The task is started by turning ON the power or resetting the AD51H. Then, when the GOTO command has been executed, the operation jumps to the designated line.

Therefore, the operation can be executed over and over between the designated line and the GOTO command.



3.13.3 Starting by an interrupt from a PC CPU(Start condition:IT)

In this case, the starting condition is set to IT by the multitask setting.

When (a) the execution conditions explained in Section 3.13.1 have been satisfied, and (b) a start request is sent from a PC CPU, the task is started.

The start request from a PC CPU can be done by using either of the following:

1) Start task number designation signal(Y21 to Y28)

2) Task start signal(Y29)

(Section 3.9.3 gives details about these signals.)

(1) Processing when the power is turned ON or the AD51H is reset

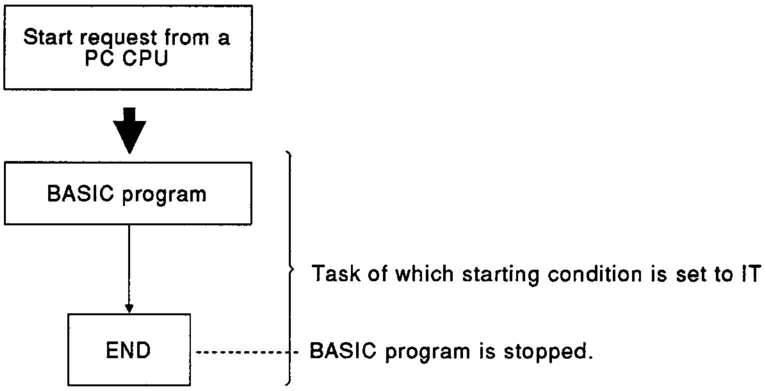
When the power is turned ON or the AD51H is reset, the execution program stored in a memory card or the EP-ROM is read to the AD51H internal memory.

The read BASIC program is executed by the start request from a PC CPU.

(2) Execution of a task

(a) When the power is turned ON or the AD51H is reset, the read BASIC program is started by sending the task start signal from a PC CPU and starting up the AD51H.

When the task start signal is sent, the task of which start task number target signal number is turned ON starts.


Start request from a PC CPU
BASIC program
END
Task of which starting condition is set to IT
BASIC program is stopped.

(b) Write the END command at the end of a BASIC program and execute it with the task started by an interrupt from a PC CPU.

When the END command has been executed, the interrupt from a PC CPU can be reset, and the following interrupt can be received.

3.13.4 Starting by other start requests from a BASIC program(Start condition:BOOT,ON)

In this case, the starting condition is set to BOOT or ON by the multitask setting.

When (a) the execution conditions explained in Section 3.13.1 have been satisfied, and (b) the start request is sent from another BASIC program by the ZSTART command, the task is started.

(The AD51H-BASIC Programming Manual gives details about the ZSTART command.)

(1) Processing when the power is turned ON or the AD51H is reset

(a) When the power is turned ON or the AD51H is reset, if the starting condition is set to BOOT, the execution program stored in a memory card or the EP-ROM is read to the AD51H internal memory.

The read BASIC program is executed by a start requests from another task.

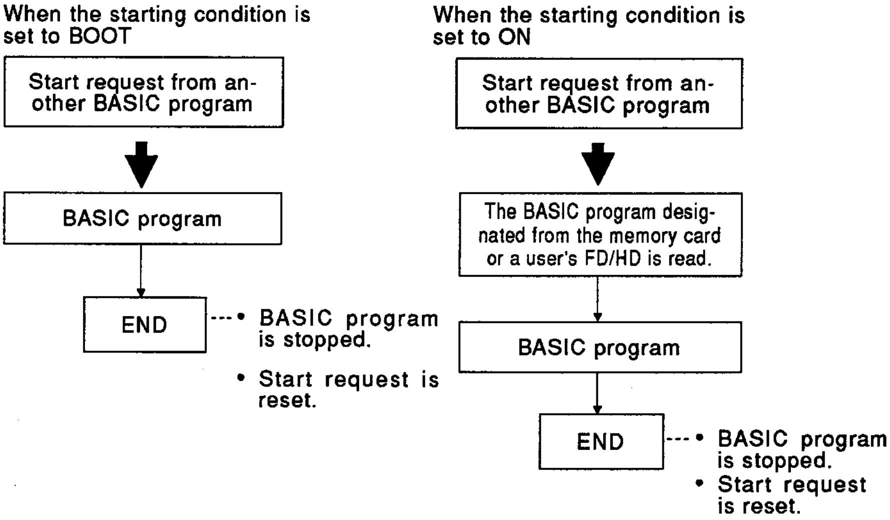
(b) When the power is turned ON or the AD51H is reset, if the starting condition is set to ON, the execution program stored in a memory card or the EP-ROM is not read to the AD51H internal memory.

If the start request is sent from another task, the designated BASIC program is read from the memory card or a user's FD/HD and is executed.

(2) Execution of a task

(a) When (a) the starting condition is set to BOOT, and (b) if the task has been designated by the ZSTART command from another BASIC program, the task starts.

If the starting condition is set to ON, the task is started after reading the BASIC program designated from the memory card or a user's FD/HD (whenever it is designated by the ZSTART command from another BASIC program).


When the starting condition is set to BOOT
Start request from another BASIC program
BASIC program
END
• BASIC program is stopped.
• Start request is reset.
When the starting condition is set to ON
Start request from another BASIC program
The BASIC program designated from the memory card or a user's FD/HD is read.
BASIC program
END
• BASIC program is stopped.
• Start request is reset.

(b) Execute the END command at the end of a BASIC program for the task to be started by the start request from another BASIC program.

If the END command is executed, the start request can be reset, and the following start reguest can be received.

3.14 Execution of a BASIC Program by Multitask Processing

This section tells how to execute BASIC programs of the AD51H.

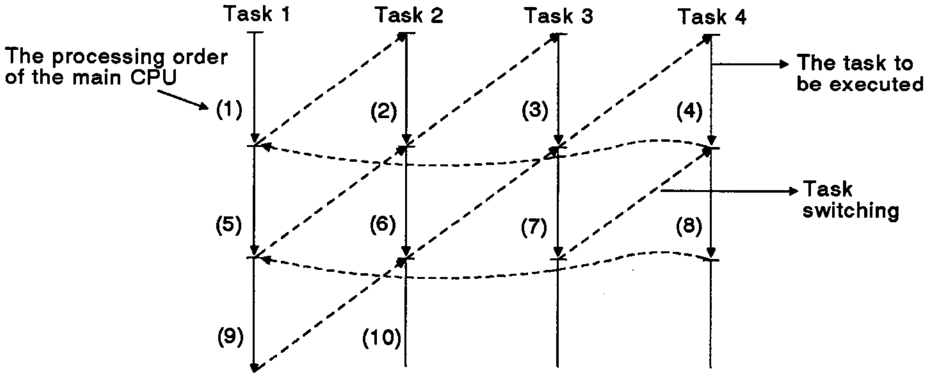
3.14.1 Multitask processing

The AD51H operates BASIC programs created by the user.

The AD51H can do parallel operations of BASIC programs by multitask processing. Multitask processing consists of switching the execution of several BASIC programs (max. eight tasks) one after another. This is done by the main CPU (central processing unit).

Example:

Processing (in this case, four tasks)



The execution of BASIC programs is switched by the multitask processing of the AD51H when the following state occurs:

- Set task schedule time (50 msec or 100 msec) has passed.
- An I/O (data communications) data command to a screen, keyboard, disk, peripheral or external device has been executed.
- The command that controls (interrupts, stops or completes) execution of a BASIC program has been executed.
- The command that controlled multitasking (changes the synchronism, start, and priority of execution) has been executed.
- The execution interruption state of a program was canceled. (Either the I/O was completed or the designated time was passed.)

REMARK

The AD51H-BASIC Programming Manual gives details about multitask processing and program execution switching commands.

3.15 BASIC program priorities

A created BASIC program can be given priority by the AD51H.

When BASIC program execution is started, all BASIC programs are set to the same priority.

Therefore, created BASIC programs are executed equally.

However, the priority can be changed by the ZURGENCY command. If the priority is changed, and if the program execution switching instruction is not executed, scheduling by the OS will not be executed and a BASIC program of the highest priority will be executed.

(Low-priority BASIC programs will not be executed.)

3.15.1 Synchronizing the executions of BASIC programs (event control)

Multitasking executes each BASIC program independently.

Therefore, when data transfer is executed between the BASIC programs, the receiver needs to read data after confirming that the sender transmitted that data.

Event control synchronizes execution of the BASIC programs of the sender and receiver.

Event control puts a BASIC program of the receiver signal in the wait state (WAIT), sends a signal (event) after the sender finishes transmitting data, and continues the BASIC program of the receiver.

Therefore, data transmitted by the sender can be read by a BASIC program of the receiver.

Event control is executed by the following commands:

• DEF-ZEVENT	: Definition of event
• ZEVENT ENABLE	: Event occurrence is valid.
• ZWAIT EVENT	: Event wait
• ZSIGNAL	: Event occurrence

REMARK

The AD51H-BASIC Programming Manual gives details about these event control commands.

3.15.2 Sharing of sources during multitasking

A source cannot be used simultaneously by several BASIC programs.

Exclusive control of a source prevents another BASIC program from using the same source.

Since exclusive control of this source is executed according to rules made by programmers, if these rules are not followed, exclusive control of a source cannot be achieved.

Exclusive control of a source is executed by the following commands:

- ZRESERVE : Requests exclusive control of a source.
- ZRELEASE : Completes exclusive control of a source.

REMARK

In this context, "source" means all the hardware and software that can be used by a BASIC program (the screen, printer, file, disk, memory, etc).

4. AD51H FUNCTIONS

This section explains the functions of the AD51H.

4.1 AD51H Functions

Table 4.1 gives the functions of the AD51H.

Table 4.1 AD51H Functions

Item		Description	Reference Section
BASIC Program Functions	Sub CPU function	1) A BASIC program can be used to calculate complex numerical values and functions that lengthen the scan time of a PC CPU. • Collection, analysis and compensation of measurement data • Functional operations such as sin, log, and square root.	The AD51H-BASIC Programming Manual gives details about commands and functions used by the AD51H.
	Monitoring display function	Displays the operating states such as production status, operating status, and trouble information on a console or terminal connected to the AD51H.	4.2
	Key operation function	Inputs the production schedule, the production quantity, operations, and set data from the keyboard of a console or terminal connected to the AD51H.	4.2
	Printer function	Prints the production plans, achievements, daily reports, trouble information, plan data, inspection results, and test results by using a printer connected to the AD51H.	4.3
	Data input function	Inputs data from a bar code reader or magnetic card reader connected to the AD51H. (By using a BASIC program, the AD51H can transmit and receive data in a free data format independently of the protocol of the communicating device.) • Inputs a production lot number, product name, quantity, etc. • Collects measurement values and test data	—
	External device connection function	Transmits and receives data from/to a BASIC program by connecting a computer to the RS-232C or RS-422 interface of the AD51H.	4.4
	File management function	Reads/writes data from/to a memory card installed in a memory card interface of the AD51H. • Management of set data and positioning data • Collection and storage of production data • Collection and storage of inspection data	—
	Clock function	Reads and writes clock data (year, month, day, hour, minute, second, and day of the week) from/to a PC CPU which has a clock function.	4.6
Offline programming function		Creates, corrects, and stores a BASIC program to a user's FD and prints it out independently by a PC/AT.	AD51H-BASIC Programming Manual (Programming, Edit, Debug)
Online programming function		Creates, executes, and corrects a BASIC program by using system commands when a console is connected to the AD51H. Stores/reads an execution program to/from a memory card using a system command.	
Multitask debugging function		Executes and debugs a BASIC program during multitasking by connecting a console and a debugger to the AD51H.	

4.2 Communications with a Console or Terminal

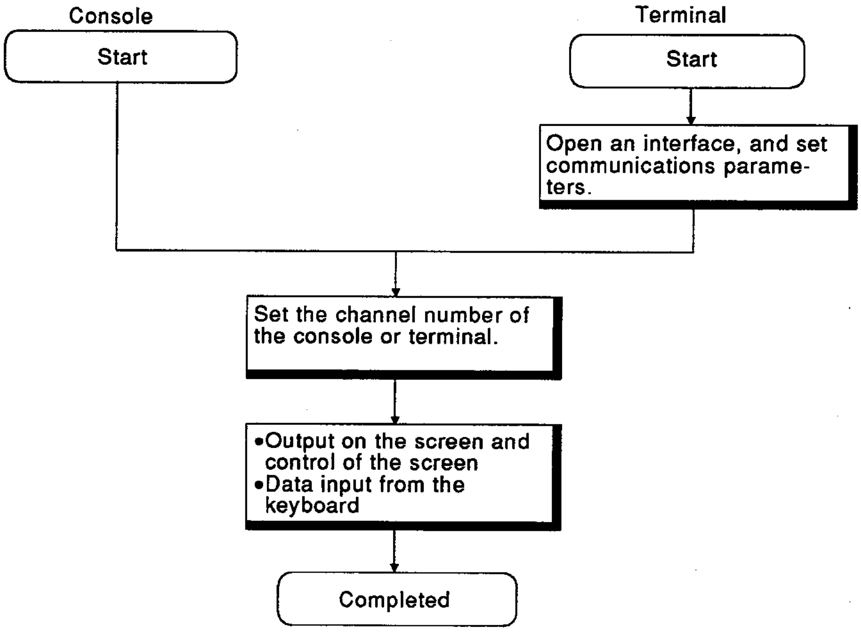
Data can be displayed on the screen of a PC/AT connected to the RS-422 interface (CH.3) of the AD51H or a VG-620, VT-382, or PC/AT connected to the RS-232C interface (CH.1 or 2) of the AD51H. Data can also be input from the keyboard.

A VG-620, VT-382, or PC/AT connected to CH.1 and a PC/AT connected to CH.3 can be used as a console and a terminal.

The device (PC/AT, VG-620, or VT-382) connected to a channel which is set for the console by mode setting switch 2 of the AD51H can be used as a console.

A device (PC/AT, VG-620, or VT-382) connected to a channel which is set for other than the console by the AD51H can be used as a terminal.

The procedures for (a) displaying a screen, and (b) inputting from the keyboard of a console or terminal are as follows:


Console
Start
Terminal
Start
Open an interface, and set communications parameters.
Set the channel number of the console or terminal.
•Output on the screen and control of the screen
•Data input from the keyboard
Completed

REMARKS

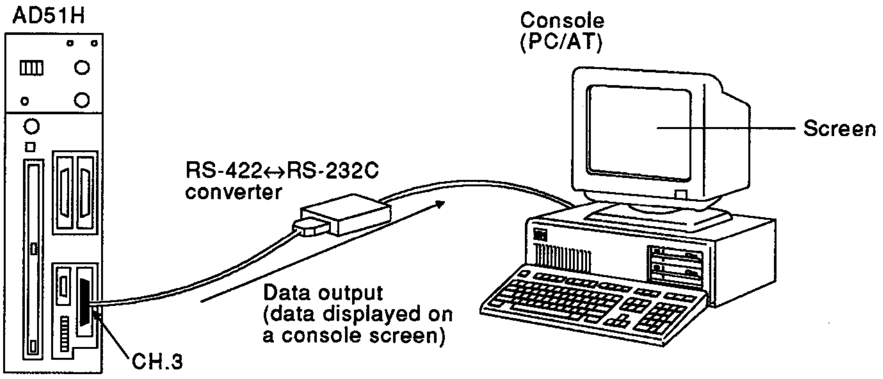
1) Chapter 2 gives details about consoles.

2) Section 5.2.1 explains how to set a console using mode setting switch 1.

4.2.1 Communications with a console

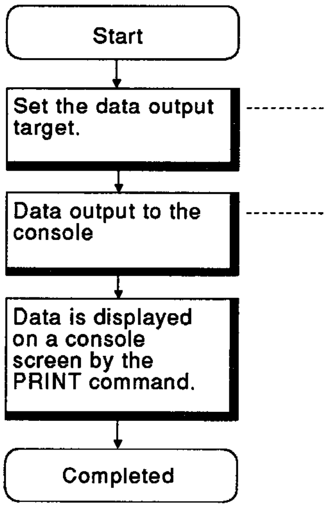
(1) Displaying data on a console screen

This section explains how data is displayed on a console screen.



[Procedure for displaying data on a console screen]

Use the following procedure to display data on a console screen:


Start
Set the data output target.
Data output to the console
Data is displayed on a console screen by the PRINT command.
Completed

Set the data output target to the console channel by using the ZODV command.

Output data to the console using the PRINT command.

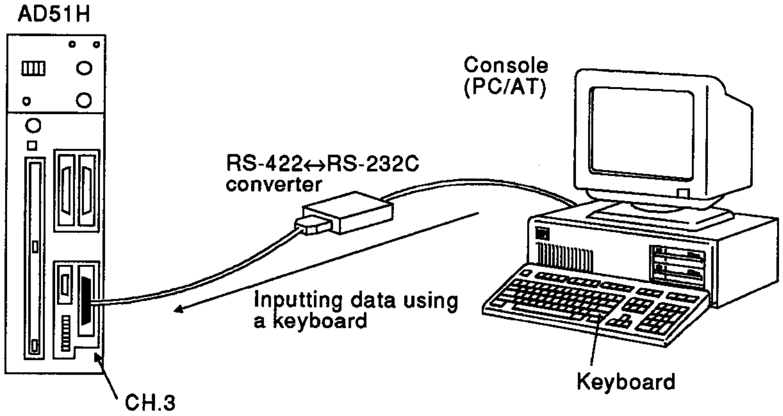
[Commands and functions]

The following list gives the commands and functions used for displaying data on a console screen.

Commands and Functions	Processing
ZODV	Switches a data output target.
PRINT	Displays data on a console screen.
PRINT USING	Displays data on a console screen. (Format can be designated.)
LOCATE	Designates the position where data is displayed on a console screen.
CLS	Clears a console screen.

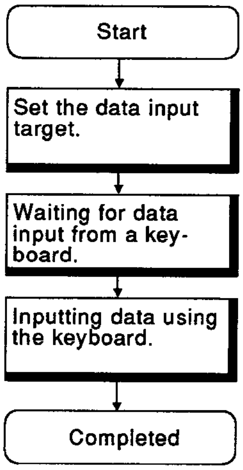
(2) Inputting data using a console keyboard

This section explains how to input data using a console keyboard.



[Procedure for inputting data using a console keyboard]

Use the following procedure to input data using a console keyboard:


Start
Set the data input target.
Waiting for data input from a key-board.
Inputting data using the keyboard.
Completed

Set the data input target to the console channel by using the ZIDV command.

Execution of the INPUT command

[Commands and functions]

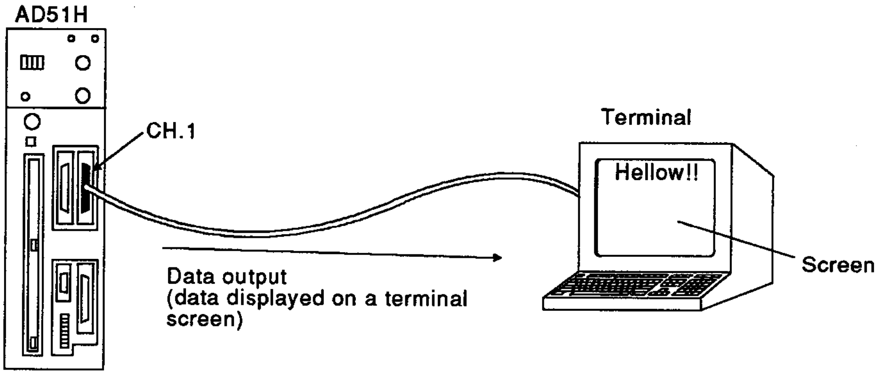
The following list gives the commands and functions used for inputting data using a console keyboard.

Commands and Functions	Processing
ZIDV	Switches a data input target.
INPUT	• Enables data input with the keyboard. • Cannot use the control codes 0H to 1FH, comma (2CH), and [DEL] (7FH) for character codes as data.
LINE INPUT	• Enables data input with the keyboard. • Cannot use the control codes 0H to 1FH and [DEL] (7FH) for character codes as data.
INKEY$ INPUT$	• Enables data inputting with a key. • Cannot use the control codes 0H, 03H, 13H, 80H, and FDH to FFH for character codes as data.

4.2.2 Communications with a terminal

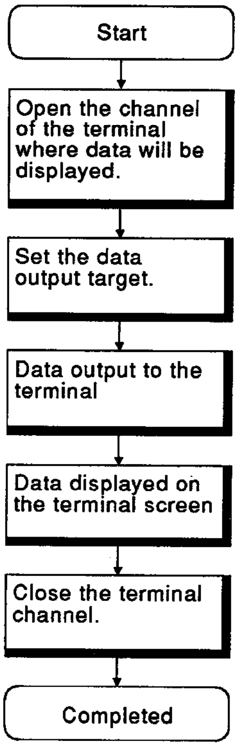
(1) Displaying data on a terminal screen

This section explains how data is displayed on a terminal screen.



[Procedure for displaying data on a terminal screen]

Use the following procedure to display data on a terminal screen:


Start
Open the channel of the terminal where data will be displayed.
Set the data output target.
Data output to the terminal
Data displayed on the terminal screen
Close the terminal channel.
Completed

Open the channel connected to the terminal by using the ZOPEN command.

Set the data output target to the terminal channel by using the ZODV command.

Data is output to the terminal by using the PRINT command.

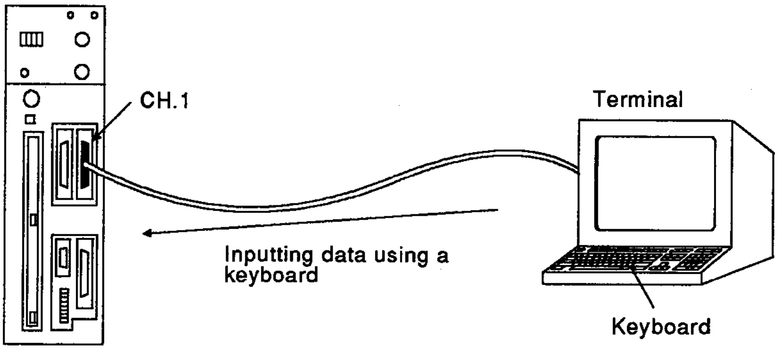
[Commands and functions]

The following list gives the commands and functions used for displaying data on a terminal screen.

Commands and Functions	Processing
ZOPEN	Opens an interface and prepares to communicate with an external device.
ZCLOSE	Close the interface.
ZCNTL	Sets communications parameters for an opened interface and reads the interface status.
ZODV	Switches a data output target.
PRINT	Displays data on a terminal screen.
PRINT USING	Displays data on a terminal screen. (Format can be designated.)
LOCATE	Designates the position where data is displayed on a terminal screen.
CLS	Clears a terminal screen.

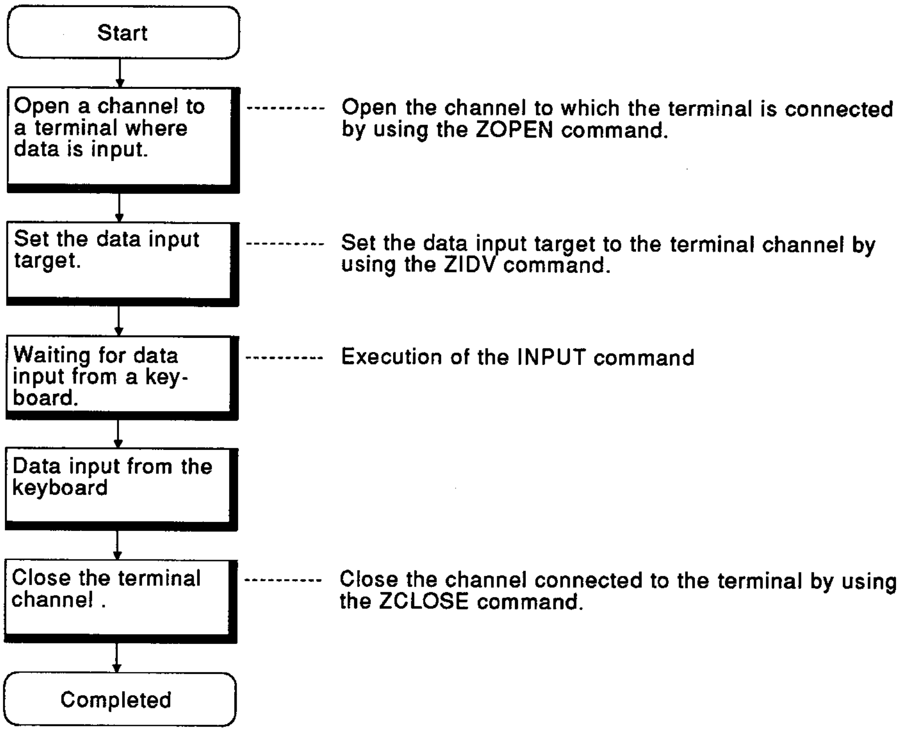
(2) Inputting data using a terminal keyboard

This section explains how to input data using a terminal keyboard.



[Procedure for inputting data using a terminal keyboard]

Use the following procedure to input data using a terminal keyboard:


Start
Open a channel to a terminal where data is input.
Set the data input target.
Waiting for data input from a key-board.
Data input from the keyboard
Close the terminal channel .
Completed

Open the channel to which the terminal is connected by using the ZOPEN command.

Set the data input target to the terminal channel by using the ZIDV command.

Execution of the INPUT command

Close the channel connected to the terminal by using the ZCLOSE command.

[Commands and functions]

The following list gives the commands and functions used for inputting data using a terminal keyboard.

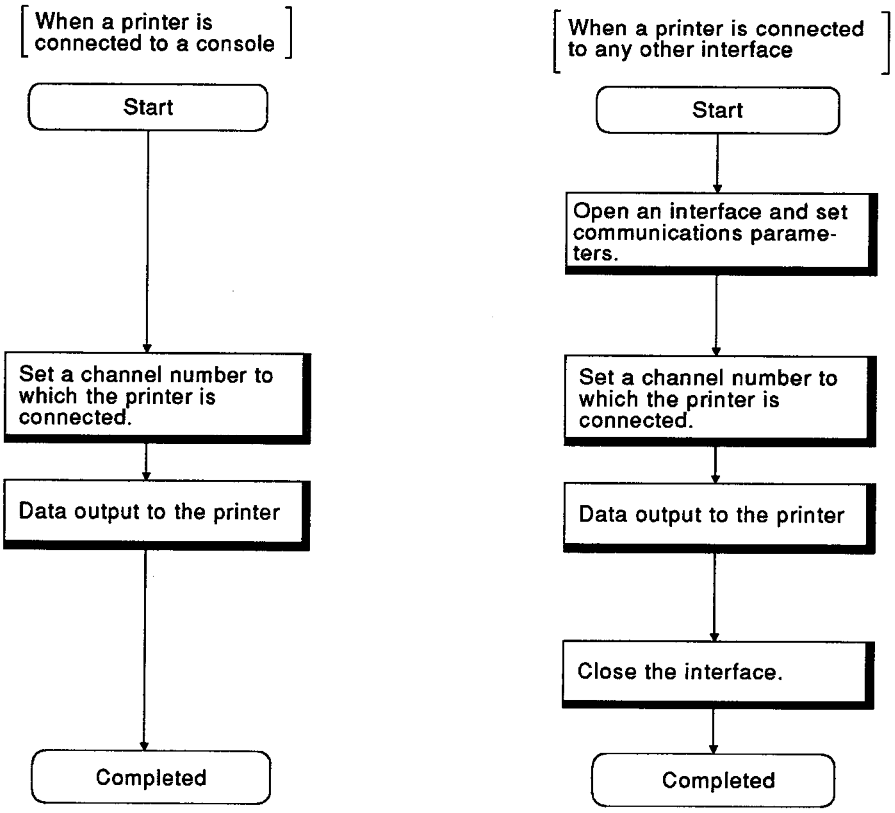
Commands and Functions	Processing
ZOPEN	Opens an interface and prepares to communicate with an external device.
ZCLOSE	Close the interface.
ZCNTL	Sets communications parameters for an opened interface and reads the interface state.
ZIDV	Switches a data output target.
INPUT	• Enables data inputting with a key • Cannot use the control codes 0H to 1FH, comma (2CH), and [DEL] (7FH) for character codes as data.
LINE INPUT	• Enables data inputting with a key. • Cannot use the control codes 0H to 1FH and [DEL] (7FH) for character codes as data.
INKEY\$ INPUT\$	• Enables data inputting with a key. • Cannot use the control codes 0H, 03H, 13H, 80H, and FDH to FFH) for character codes as data.

4.3 Printing out Data at Printer

Data can be printed by using a printer connected to the RS-232C interface, RS-422 interface, or parallel interface of the AD51H.

Data can also be printed by connecting a printer to a PC/AT used as a console.

The procedure using a printer connected to a console (PC/AT) is different from the procedure using a printer connected to any other interface.


When a printer is connected to a console
Start
Set a channel number to which the printer is connected.
Data output to the printer
Completed
When a printer is connected to any other interface
Start
Open an interface and set communications parameters.
Set a channel number to which the printer is connected.
Data output to the printer
Close the interface.
Completed

The printing procedure depends on:

(1) the printer connected to an interface other than the console interface; or

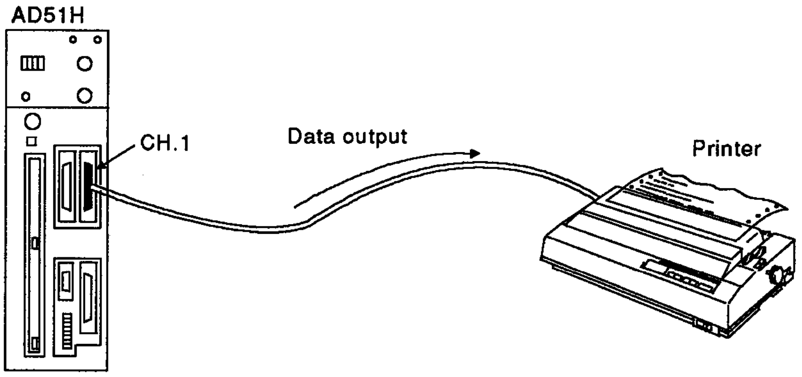
(2) the printer connected to the console (PC/AT).

POINT

Can be executed only for online programming using the SW1SRX/IVD/NX -AD51HP.

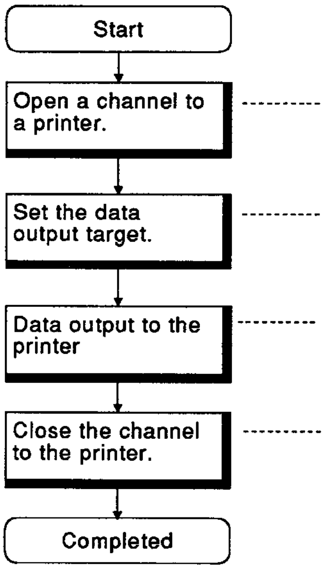
4.3.1 Printout using a printer connected to an interface which is not set for a console

This section explains how to print by connecting a printer to an interface which is not set for a console.



[Printout procedure]

Use the following procedure to print data by using a printer connected to an interface which is not set for a console:


Start
Open a channel to a printer.
Set the data output target.
Data output to the printer
Close the channel to the printer.
Completed

Open a channel to which a printer is connected by using the ZOPEN command.

Set the data output target to the printer by using the ZLDV command.

Output data to the printer by using the LPRINT command.

Close the channel to which the printer is connected by using the ZCLOSE command.

[Commands and functions]

The following list gives the commands and functions used for printing data by using a printer connected to an interface which is not set for a console.

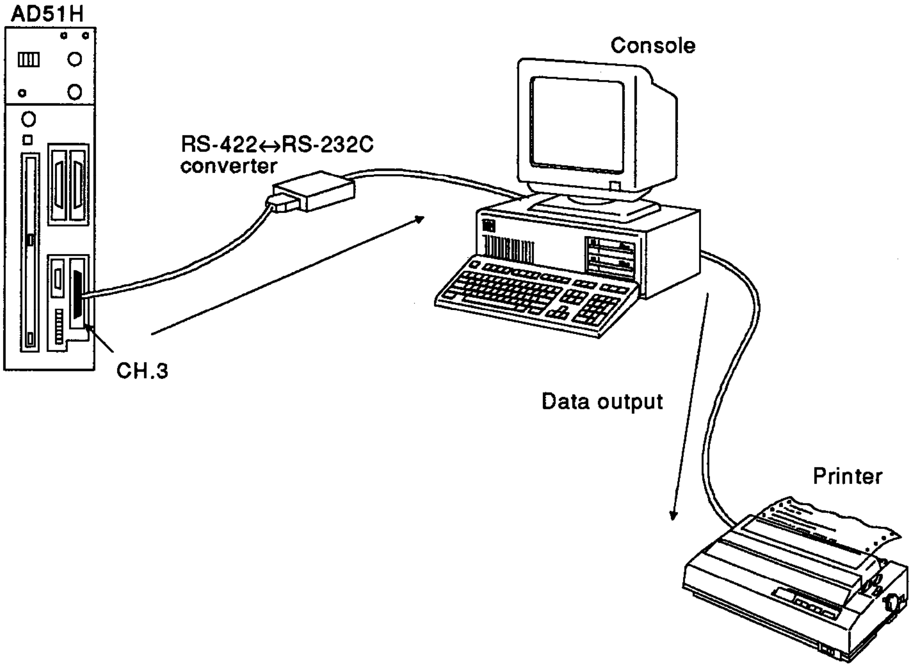
Commands and Functions	Processing
ZOPEN	Opens an interface and prepares for communications with an external device.
ZCLOSE	Closes the interface.
ZCNTL	Sets communications parameters for an opened interface and reads the interface state.
ZLDV	Switches a channel number to the channel number of the interface to which a printer is connected.
LPRINT	Outputs data to a designated printer.
LPRINT USING	Outputs data to a designated printer. (Format can be designated.)
LLIST	Outputs a program list to a designated printer.

4.3.2 Printout using a printer connected to a console (PC/AT)

The method of printout by connecting the printer to PC/AT with the interface set on the console is described.
Printer settings are required for printout with PC/AT.
Printer settings for PC/AT are made on the OS environment settings.

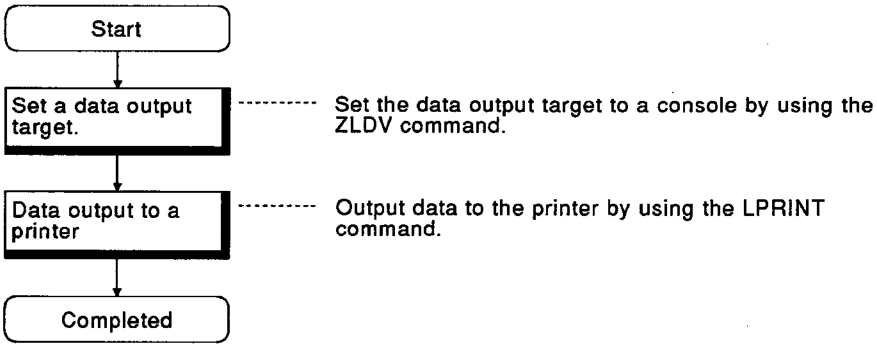
POINT

It is available only for the online programming using SW1IX-AD51HP.



[Printout procedure]

Use the following procedure to print data using a printer connected to a console:


Start
Set a data output target.
Set the data output target to a console by using the ZLDV command.
Data output to a printer
Output data to the printer by using the LPRINT command.
Completed

[Commands and functions]

The following list gives the commands and functions used for printing data by using a printer connected to a console.

Commands and Functions	Processing
ZLDV	Switches the output target to a printer connected to a console.
LPRINT	Outputs data to a designated printer.
LPRINT USING	Outputs data to a designated printer. (Format can be designated.)
LLIST	Outputs a program list to a designated printer.

4.4 Communications With a Device Other Than a Console, Terminal, or Printer

Communications is enabled by connecting an external device to the RS-232C interface or the RS-422 interface of the AD51H.

4.4.1 Sending data

This section explains how data is sent from the AD51H to a device other than a console, terminal, or printer.

Data send processes:

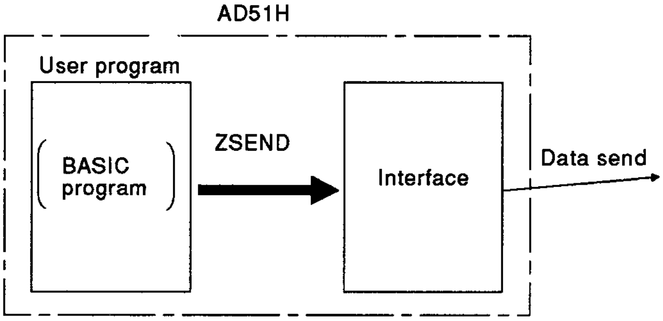
1) Opens an interface.

2) Prepares data to be sent.

3) Sets the number of bytes and the time-out duration of the sent data.

4) Transmits data by using the ZSEND command.

5) Closes the interface.



[Commands and functions]

The following list gives the commands and functions used for sending data from the AD51H to a device other than a console, terminal, or printer. *2

Commands and Functions	Processing
ZOPEN	Opens an interface and prepares for communications with an external device.
ZCNTL	Sets communications parameters for an open interface and reads the interface status.
ZSEND	Sends data from a designated interface.
ZCLOSE	Closes an interface.

REMARKS

*1 : Other than a console, terminal, or printer communications using consoles and terminals. Print operation.

*2 : The AD51H-BASIC Programming Manual gives details about commands and functions.

4.4.2 Receiving data

This section explains how the AD51H receives data sent from a device other than a console or terminal.

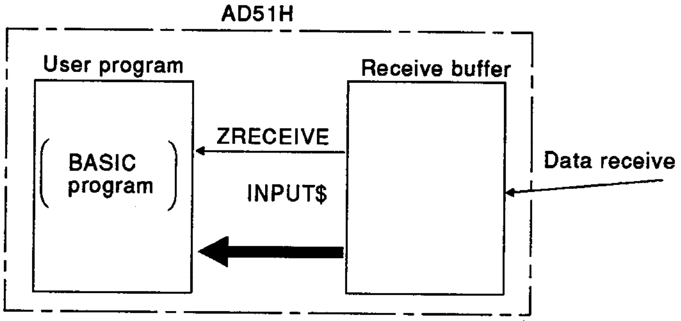
Data receive processes:

1) Opens an interface.

2) Receives data

3) Stores the received data in the variables area by using either the ZRECEIVE command or INPUT$ function.

4) Closes the interface.



[Commands and functions]

The following list gives the commands and functions used by the AD51H for receiving data sent from a device other than a console or terminal.

Commands and Functions	Processing
ZOPEN	Opens an interface and prepares for communications with an external device.
ZCLOSE	Close the interface.
ZCNTL	Sets communications parameters for an opened interface and reads the interface state.
INPUT$	• Stores data received from an interface in a designated variables area. • Cannot use the control codes 0H, 03H, 13H, 80H, and FDH to FFH of character codes as data.
ZRECEIVE	• Stores data received from an interface in a designated variables area.

REMARK

The AD51-BASIC Programming Manual gives details about commands and functions.

4.5 Communications With a PC CPU and a Special-Function Module

The AD51H can read/write data from/to a PC CPU by using a BASIC program.

The following shows data that can be read/written from/to a PC CPU:

No.	Data Name	Write	Read
1	Device memory	O	O
2	Extension file register	O	O
3	PC CPU model name	X	O
4	Parameter data	O	O
5	Sequence program	O	O
6	Microcomputer program	O	O
7	Comment data	O	O
8	Extension comment data	O	O
9	Remote RUN/STOP of a PC CPU	O	X

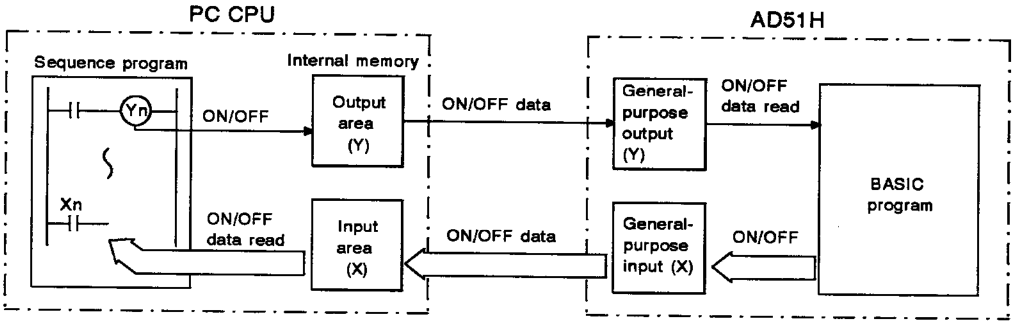
O: Enabled X: Disabled

Data can also be read/written from/to the buffer of a special-function module by using a BASIC program.

This section explains the read/write of (a) data (1 and 2 above) from/to a PC CPU, and (b) data from/to the buffer of a special-function module.

4.5.1 Communicating ON/OFF data using the general-purpose inputs/outputs (X/Y) between the AD51H and a PC CPU

ON/OFF data is sent and received between the AD51H and a PC CPU by using the general-purpose inputs/outputs (X/Y) of the AD51H.



(1) Send from a PC CPU to the AD51H

ON/OFF data is sent from a PC CPU to the AD51H by using general-purpose output (Y).

A general-purpose output turned ON or OFF by a PC CPU is read by using special variable B@ with a BASIC program of the AD51H.

(2) Send from the AD51H to a PC CPU

Data is sent from the AD51H to a PC CPU by using general-purpose input (X).

A general-purpose input turned ON or OFF by a BASIC program of the AD51H is read by a sequence program in the PC CPU.

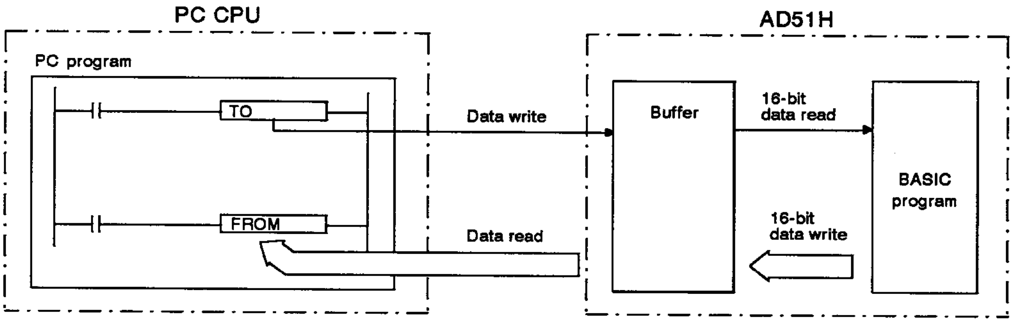
[Special variables]

The following list gives the special variables used for reading ON/OFF data of general-purpose input (X) and ON/OFF data of general-purpose output (Y).

Special Variables	Processing
B@ (X, expression)	Turns ON/OFF a general-purpose input (X) of the AD51H.
B@ (Y, expression)	Reads output data (Y) and general-purpose ON/OFF data of the AD51H.

4.5.2 16-bit data communications using the buffer

16-bit data is sent and received between the AD51H and a PC CPU by using the AD51H's buffer.



(1) Send from a PC CPU to the AD51H

16-bit data written to the AD51H buffer with a sequence program (TO command) of a PC CPU is read by using a BASIC program.

(2) Send from the AD51H to the PC CPU

16-bit data written to the AD51H buffer by using a BASIC program is read by using a sequence program (FROM command) of a PC CPU.

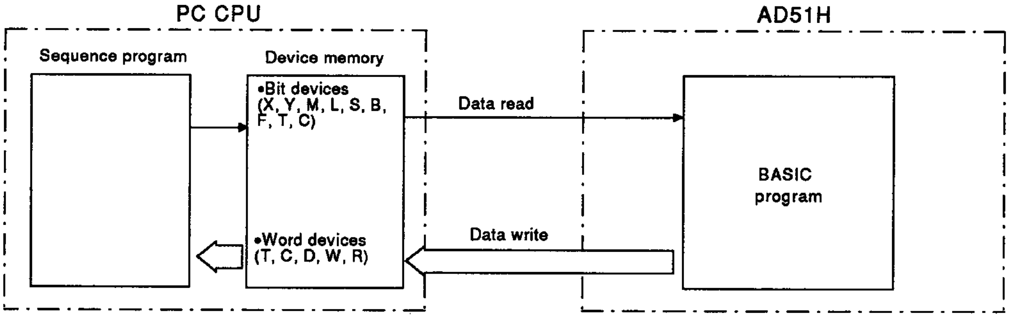
[Commands and functions]

The following list gives the commands and functions used for reading/writing 16-bit data from/to the AD51H buffer.

Commands and Functions	Processing
GETMEM	Reads the contents of the AD51H buffer.
PUTMEM	Writes 16-bit data to the AD51H buffer.

4.5.3 Reading/writing data from/to the device memory of a PC CPU

Data can be read/written from/to the device memory of a PC CPU by using a BASIC program of the AD51H.



The following devices can be read from and written to by using a BASIC program:

(a) Bit devices : X, Y, M, L, S, B, F, T, C

(b) Word devices : T(present value), C(present value), D, W, R

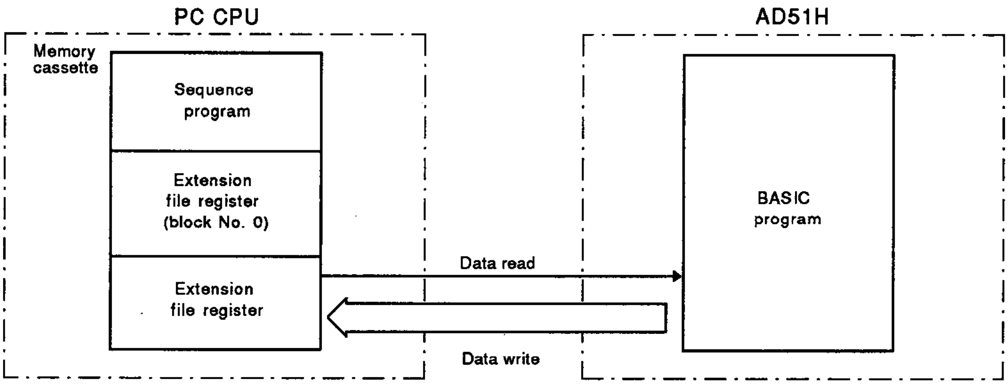
[Commands and functions]

The following list gives the commands and functions used for reading/writing data from/to the device memory of a PC CPU.

Commands and Functions	Processing
PCRD	Reads data from a PC CPU.
PCWT	Writes data to a PC CPU.

4.5.4 Reading/writing data from/to a PC CPU extension file register

Data can be read/written from/to the extension file register (extension file register of block No. 1 and later) of a PC CPU by using a BASIC program of the AD51H.



[Commands and functions]

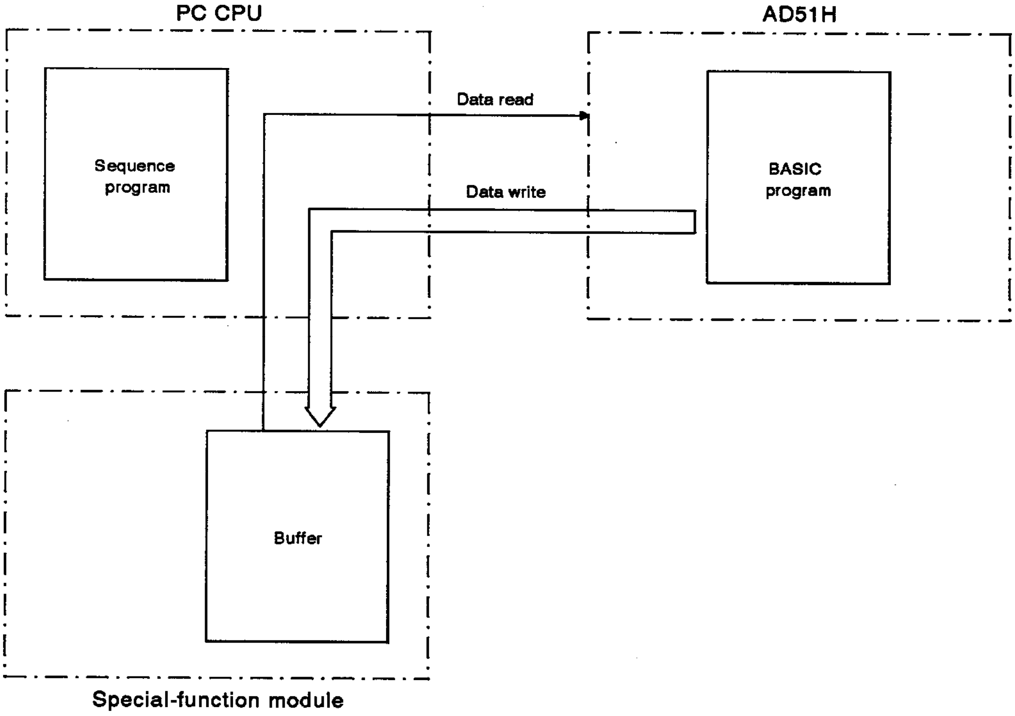
The following list gives the commands and functions used for reading/writing data from/to the extension file register of a PC CPU.

Commands and Functions	Processing
PCRD	Reads data from a PC CPU.
PCWT	Writes data to a PC CPU.

4.5.5 Reading/writing data from/to the buffer of a special-function module

Data can be read/written from/to the buffer of a special-function module by using a BASIC program of the AD51H.

Appendix 2 gives details about the special-function module's buffer addresses.



[Commands and functions]

The following list gives the commands and functions used for reading/writing data from/to the buffer of another special-function module.

Commands and Functions	Processing
PCRD	Reads data from a PC CPU.
PCWT	Writes data to a PC CPU.

REMARK

The AD51H-BASIC Programming Manual gives details about commands and functions.

4.6 Using Clock Data

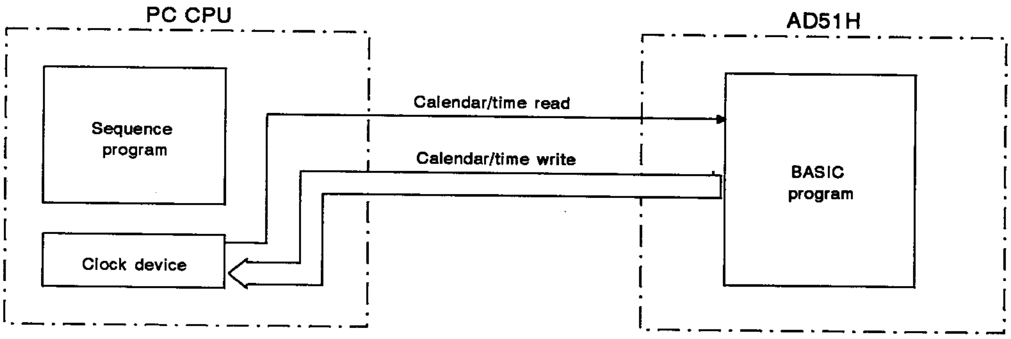
The AD51H does not have a clock function.

Therefore, when using the clock function on the AD51H, the calendar data (year, month, day, and day of the week) or time data (hour, minute, and second) must be read using a BASIC program from a PC CPU which has the clock function.

The following PC CPUs don't have a clock function:

- A0J2CPU
- A1CPU
- A2CPU(-S1)
- A3CPU
- A0J2HCPU
- A3HCPU
- A3MCPU

When using the clock function of a PC CPU, the calendar and time data must have been written to the clock device.



[Commands and functions]

The following list gives the commands and functions used for reading/writing the calendar and time from/to a PC CPU.

Commands and Functions	Processing
DATE$	Reads and writes calendar data from/to the clock device of a PC CPU.
TIME$	Writes data to the clock device of a PC CPU and reads the time from it.

4.7 Data Communications Between Tasks

Details about the variables and array of each task can be referred to the BASIC program for that task during multitasking.

The following is used for data communications between BASIC programs during multitasking:

1) Extension relay (EM) : ON/OFF data

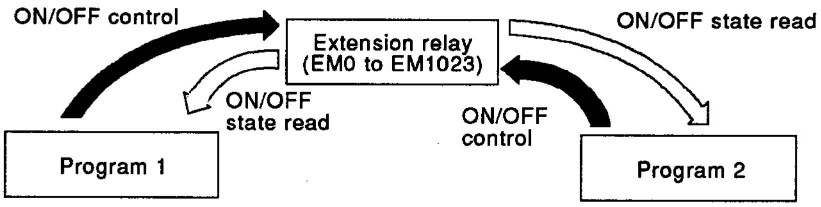
2) Extension register (ED) } : Data

3) Common memory

4.7.1 Communicating ON/OFF data using the extension relay (EM)

ON/OFF control of, and read of ON/OFF state data stored in, the extension relay (EM) of the AD51H can be executed from all tasks.

(Section 3.7.7 gives details about extension relays.)



The AD51H can use 1024 extension relay points from EM0 to EM1023.

[Commands and functions]

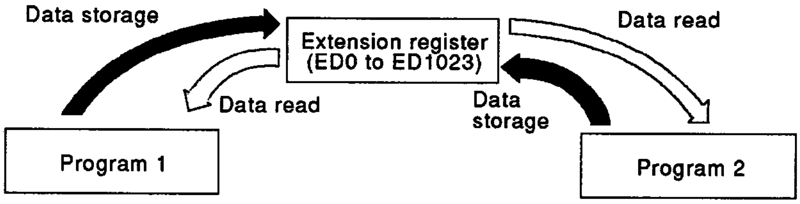
The following list gives the commands and functions used for receiving and sending ON/OFF data when using an extension relay.

Commands and Functions	Processing
B@	Reads ON/OFF control data and ON/OFF state data.

4.7.2 Data communications using the extension register (ED)

Data can be read/written from/to the extension register (ED) of the AD51H using any task.

(Section 3.7.5 gives details about extension registers.)



The AD51H can use 1024 extension register points from ED0 to ED1023.

[Commands and functions]

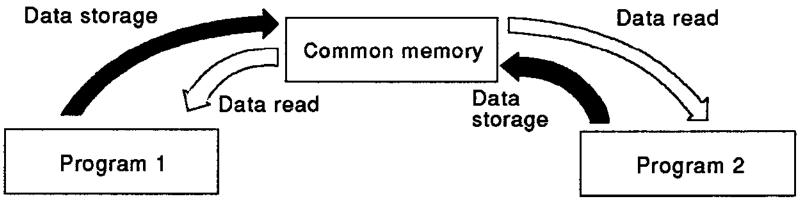
The following list gives the commands and functions used for receiving and sending data from/to an extension register.

Commands and Functions	Processing
W@	Reads and stores data from/in an extension register.
GETMEM	Reads data from an extension register.
PUTMEM	Stores data in the extension register.

4.7.3 Data communications using the common memory

The AD51H buffer is the common memory from/to which data can be read/written using any task.

(Section 3.7.4 gives details about the common memory.)



The AD51H can use 4K-word (8K bytes) common memory.

[Commands and functions]

The following list gives the commands and functions used for receiving and sending data when using the common memory.

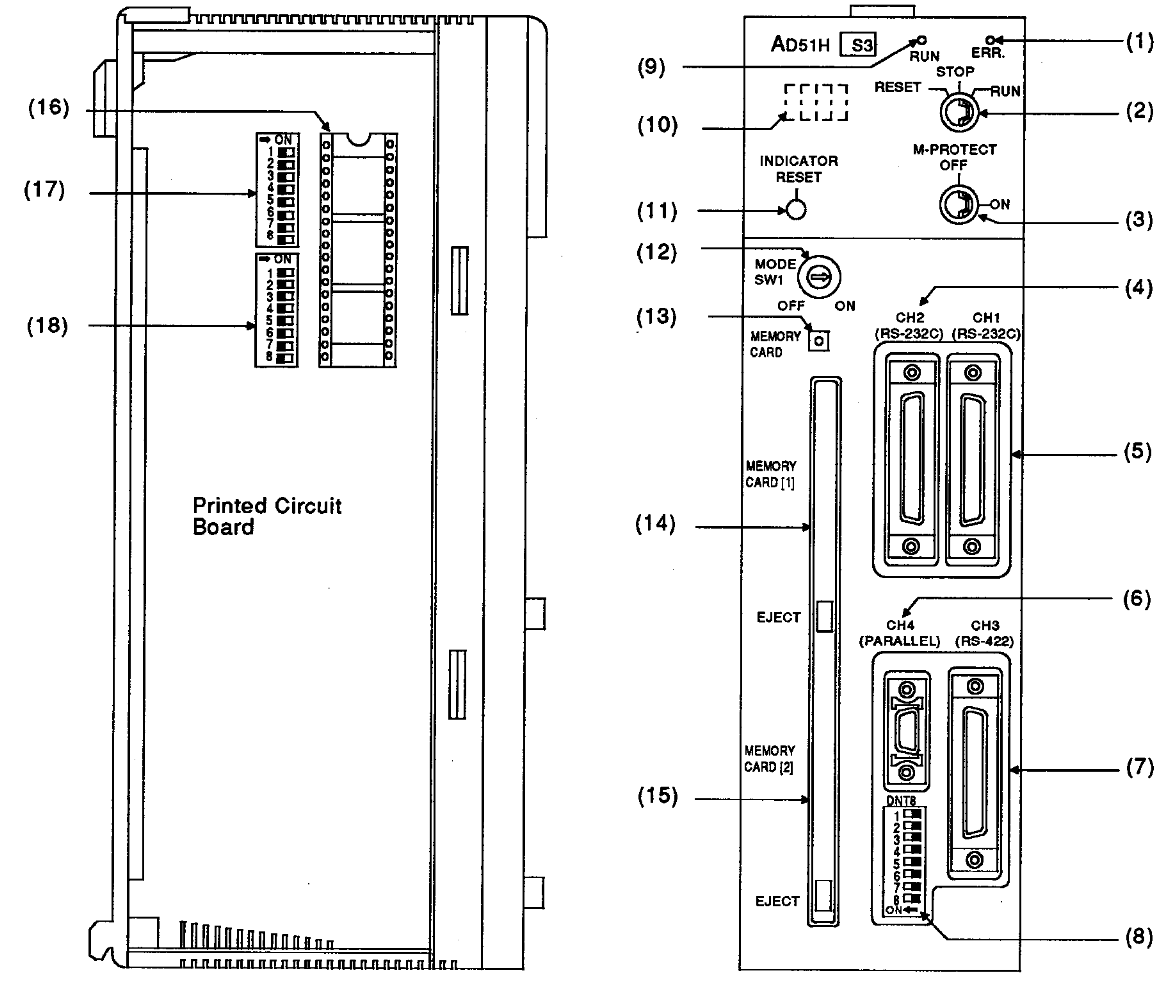
Commands and Functions	Processing
GETMEM	Reads data from the common memory.
PUTMEM	Stores data in the common memory.

5. NAMES, FUNCTIONS, AND SETTINGS OF PARTS

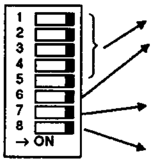
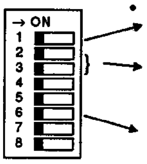
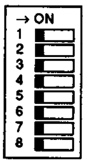
This section explains the names, functions, and settings of parts of the AD51H.

5.1 Names and Functions of Parts

The illustrations below show the names of parts of the AD51H.



No.	Name	Function	Reference Section
(1)	ERROR LED	• This displays the normal/abnormal (error) state of the AD51H. ON : When an error occurs OFF : Normal	—
(2)	RUN keyswitch	• Used for executing and stopping tasking and for hardware resetting. RUN : Tasking is executed (only valid in the execution or debugging mode). STOP : Execution of tasking is stopped (only valid in the execution or debugging mode). RESET : Hardware reset	—
(3)	Memory-protect keyswitch	• Used for the valid/invalid setting of the memory-protect range set by the memory-protect range setting switch. OFF : Invalid ON : Valid	5.2.9

No.	Name	Function	Reference Section
(4)	RS-232C interface	• Used to connect a debugger, general-purpose console, or printer.	3.2.4
(5)	RS-232C interface	• Used to connect a console, debugger, general-purpose console, or printer.	3.2.3
(6)	Parallel interface	• Used to connect a printer.	3.2.6
(7)	RS-422 interface	• Used to connect a console, debugger, general-purpose console, or printer.	3.2.5
(8)	Mode setting switch 2 (MODE SW2)	 • Used for the following settings of the AD51H: Console or debug port The program operation is suspended by inputting (a) the [BREAK] key, or (b) the [Ctrl] + [C] keys. Area to which an execution program is booted Scheduled time of task	5.2.2 to 5.2.5
(9)	RUN LED	• Displays the operating state of the AD51H. ON : Operating normally OFF : Operations interrupted	—
(10)	LED indicator	• Displays the operation state and the error code when an error occurs.	5.6
(11)	Indicator reset switch	• Used for resetting when an error is displayed on the LED indicator. (If the cause of the displayed error code has not been eliminated, the error code will be redisplayed.) • When multiple errors occur, error codes are displayed one after another every time an error code indicator is reset.	—
(12)	Mode setting switch 1	• Used for setting the operating mode of the AD51H.	5.2.1
(13)	Memory card access switch	• Used for installing and removing memory cards during the power supply is ON.	5.3.4
(14)	MEMORY CARD1	• A memory card is inserted here, and execution programs and data files are managed. • Read/write of execution programs and access to files are enabled.	3.3 and 5.3.4
(15)	MEMORY CARD2	• A memory card is inserted here, and data files are managed. • Access to files is enabled.	3.3 and 5.3.4
(16)	ROM socket	• Used to install the EP-ROM. • Used to read an execution program from the EP-ROM inserted in the socket.	3.4 and 5.3.3
(17)	Switch 2	 • Used for the following settings of the AD51H: Operation of the AD51H when resetting a PC CPU Possible access time of FROM/TO instructions from a PC CPU when resetting the AD51H EP-ROM type to be used • Set 4, and 5 to OFF.	5.2.5 to 5.2.8
(18)	Memory-protect range switch	 • Used for setting the memory-protect range of a memory card (used for BASIC programs) installed in MEMORY CARD1. ON : Protected OFF : Not protected	5.2.9

5.2 Settings

This section explains the required settings used with the AD51H.

5.2.1 When setting the operating modes (BASIC program execution, multitask debugging, and programming)

The execution mode must be set to use the AD51H.

(1) Operating modes

There are three kinds of operating modes.

(a) Execution mode

This mode is used to (a) read execution programs from a memory card or the EP-ROM to the program area of the AD51H, and (b) control operations. (See Section 6.4.)

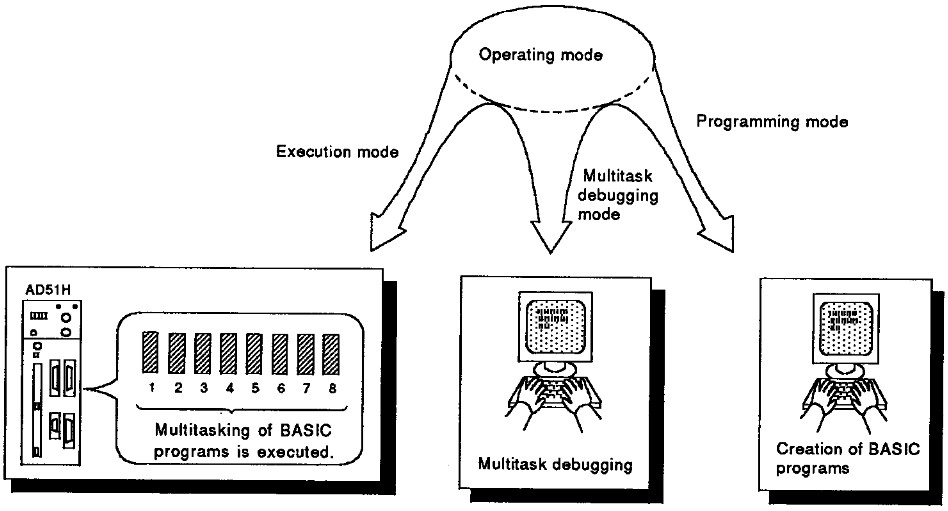
(b) Multitask debugging mode

This mode is used for debugging multiple BASIC programs created in the programming mode by using a debugger.

(See the online programming (debug) part.)

(c) Programming mode

This mode is used to (a)program one-task BASIC programs by using a console, (b) store BASIC programs to a memory card, and (c) do all multitask settings (except programming). (See online programming (debug) part.)

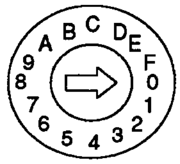


(2) Setting the operating mode

Use mode setting switches 0 to 4 to set the operating mode.

Table 5.1 shows the settings of mode setting switch 1.

Table 5.1 Settings of Mode Setting Switch 1

Mode Setting Switch 1	Operating Mode	Operating Mode Setting No.
	Execution mode	0 or 1
	Multitask debugging mode	2 or 3
	Programming	4
	Unusable	5 to F

5.2.2 Console and debugger settings

This section tells how to set an interface (RS-232C or RS-422) to which a console (used to create BASIC programs) or a debugger (used to do multitask debugging) is connected.

(1) Consoles and debuggers

Peripheral devices and general-purpose consoles which can be used as consoles and debuggers are explained below.

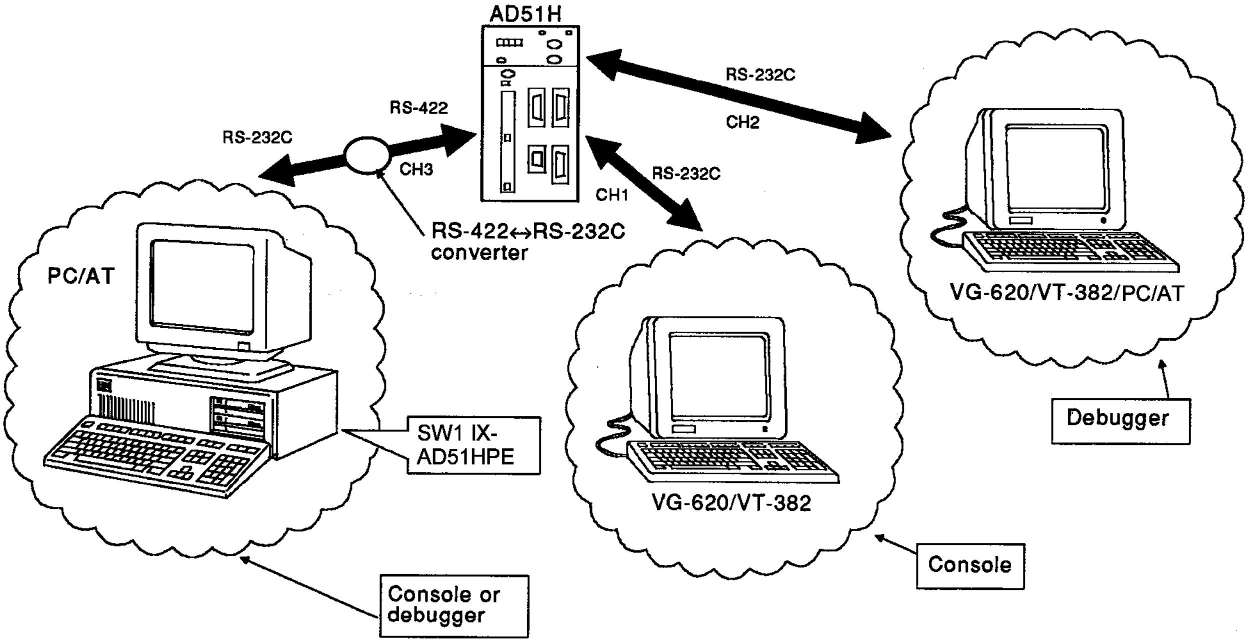
(a) Console

A console is operable in the programming mode (when mode setting switch 1 is set to "4").
Set the RUN keyswitch to either RUN or STOP.

1) Connect a PC/AT to the RS-422 interface via the RS-422 ↔ RS-232C converter.
2) Connect a VT-382, VG-620 or PC/AT to the RS-232C (CH.1) interface.

(b) Debugger

1) Connect a PC/AT to the RS-422 interface via the RS-422 ↔ RS-232C converter.
2) Connect a VT-382, VG-620 or PC/AT to the RS-232C (CH.1 or CH.2) interface.



POINT

A debugger can be operated when the RUN keyswitch is set to RUN in the debugging mode (when the execution mode setting switch is set to "2" or "3").
If the RUN keyswitch is set to STOP, a PC/AT, VT-382 or VG-620 set as the debugger cannot be operated.

(2) Console and debugger settings

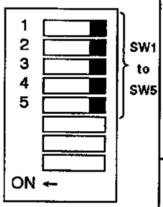
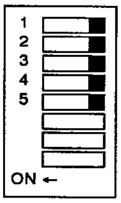
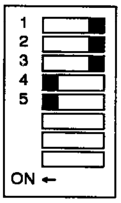
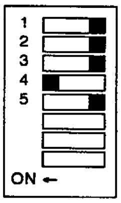
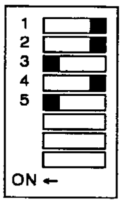
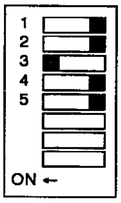
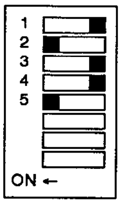
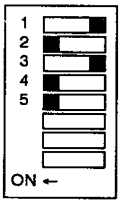
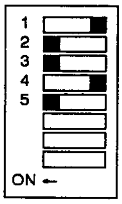
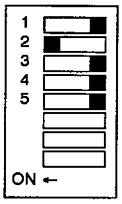
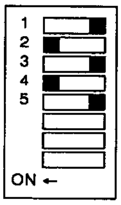
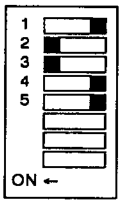
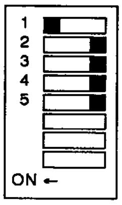
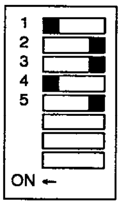
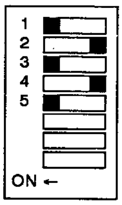
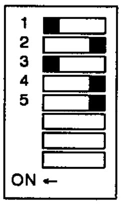
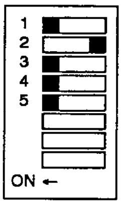
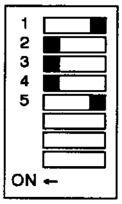
Use a combination of ON/OFF settings of SW1 to SW5 of mode setting switch 2 to set a device (PC/AT, VG-620 or VT-382) to be used as a console or a debugger.

If a console or debugger is connected to CH1 and CH2 of the RS-232C interface, only devices (VG-620, VT-382 or PC/AT) with the same model name can be used.

Therefore, a VG-620, VT-382 or PC/AT cannot be used simultaneously as a console and debugger.

Table 5.2 shows the settings of mode setting switch 2 for consoles and debuggers.

Table 5.2 Settings of Mode Setting Switch 2 for Consoles and Debuggers

Mode Setting Switch	Console	Debugger					
		None	RS-422 (CH3)	RS-232C (CH1)		RS-232C (CH2)	
			PC/AT	VT-382 or PC/AT	VG-620	VT-382 or PC/AT	VG-620
Mode setting switch 2  (Set each switch to the ■side.)	PC/AT (connected to CH3)		—				
	VT-382 or PC/AT (connected to CH1)			—	—		—
	VG-620 (connected to CH1)			—	—	—	
	None						

REMARK

Combinations indicated as — in Table 5.2 are not available.

5.2.3 Setting the BASIC program stop/continue operation instructed by the [Break], or [Ctrl] + [C] keys

This sets the inputting of the [Break] *2, or [Ctrl] + [C] keys on the console *1 to valid or invalid during multitasking in the execution mode.

(1) Operations when pressing the [Break], or [Ctrl] + [C] keys

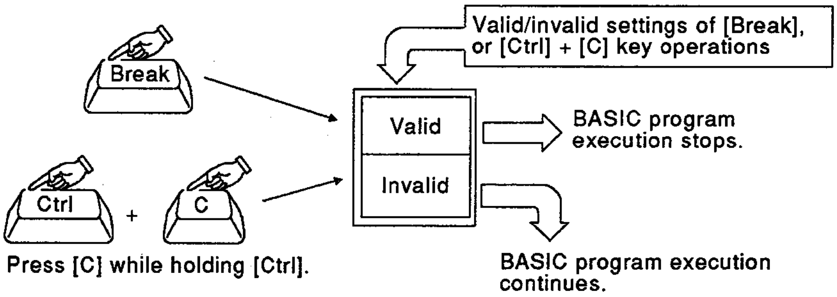
(a) When (a) the key operation is set to valid, and (b) if the [Break], or [Ctrl] + [C] keys (on the console) are pressed, multitasking stops and the operation mode switches to the system mode.

To forcibly stop multitasking from the console, set this to valid.

(b) When (a) this is set to invalid, and (b) if the [Break], or [Ctrl] + [C] keys (on the console) are pressed, multitasking continues.

Operation of the [Break], or [Ctrl] + [C] keys is received by the AD51H as "key code (03H)".

Set this to invalid to prevent accidental suspension of multitasking caused when the [Break], or [Ctrl] + [C] keys are pressed by mistake.

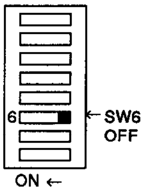
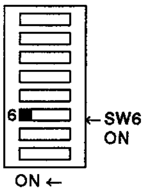

Valid/invalid settings of [Break], or [Ctrl] + [C] key operations
Break
Ctrl
+
C
Press [C] while holding [Ctrl].
Valid
Invalid
BASIC program execution stops.
BASIC program execution continues.

(2) Valid/invalid settings of [Break], or [Ctrl] + [C] key operations

Set the [Break], or [Ctrl] + [C] key operations to valid or invalid by the ON/OFF settings of SW6 of mode setting switch 2 (on the front of the AD51H).

Table 5.3 shows the setting descriptions and settings of mode setting switch 2.

Table 5.3 Valid/Invalid Settings of the [Break], or [Ctrl] + [C] Key Operations

Settings of the [Break], or [Ctrl] + [C] Key Operations	SW6 Setting
Invalid	
Valid	

REMARKS

1)*1 : A device set by SW1 to SW5 of mode setting switch 2 is set to a console.

2)*2 : Multitask can be stopped from a PC/AT only by pressing the [Break] or [Ctrl] + [C] key.

3) Setting the [Break], or [Ctrl] + [C] key operations to invalid prevents accidental suspension of multitasking caused when the [Break], or [Ctrl] + [C] keys are pressed by mistake.

5.2.4 Designation of the read target of execution programs

This section explains settings for reading execution programs stored in a memory card or the EP-ROM to the program area of the AD51H when (a) the power supply is turned ON, or (b) the AD51H is reset.

Designate the read target of execution programs to the memory card or the EP-ROM.

(1) Operations when designating the memory card or the EP-ROM

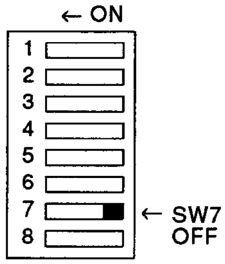
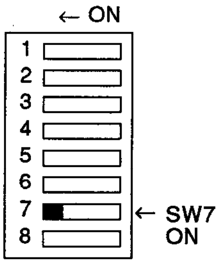
(a) If a memory card is designated, execution programs stored in the memory card installed in the MEMORY CARD1 interface are read to the internal memory of the AD51H.
If there is no memory card in MEMORY CARD1, execution programs will be read from the EP-ROM.

(b) If the EP-ROM is designated, execution programs stored in the EP-ROM (installed in the memory socket on the left side of the AD51H) are read to the program area of the AD51H.
If there is no EP-ROM in the memory socket, execution programs are read from a memory card.

(2) Designating the read target of the execution program

Designate the read target of the execution program by the ON/OFF settings of SW7 of switch 2 (on the front of the AD51H).
Table 5.4 shows settings of switch 2 and the read target priorities of the execution program.

Table 5.4 Read Target Priorities of the Execution Program

Read Target Priorities of the Execution Program	SW7 Setting
The memory card in MEMORY CARD1 has priority. (If there is no memory card in MEMORY CARD1, execution programs are read from the EP-ROM.)	
The EP-ROM has priority. (If there is no EP-ROM in the memory socket, the execution program is read from the memory card in MEMORY CARD1.)	

POINT

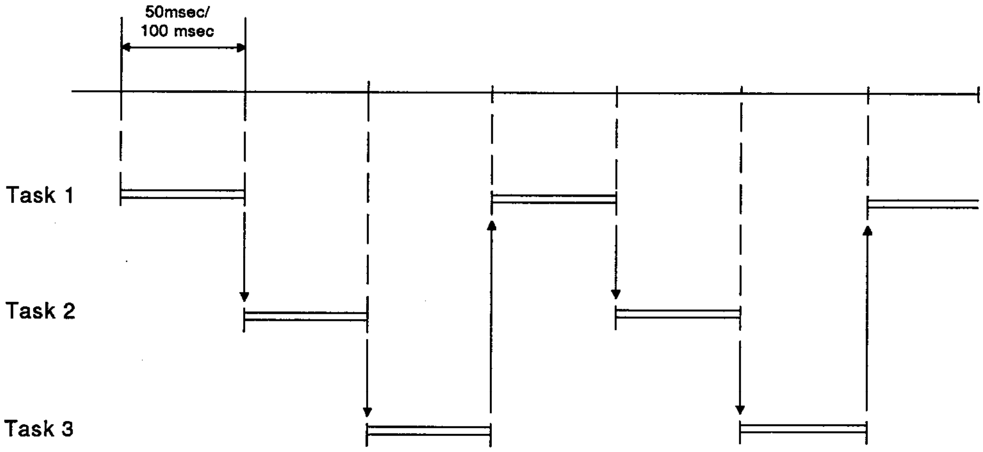
If no execution program is stored in a designated target memory card or EP-ROM, an INIE error occurs, and the execution mode is disabled.

5.2.5 Setting the task switching time

The task switching time is the time it takes to switch BASIC programs during multitask processing.

(1) BASIC program switching

BASIC programs are switched according to the scheduled times of set tasks.



(2) Setting the task switching time

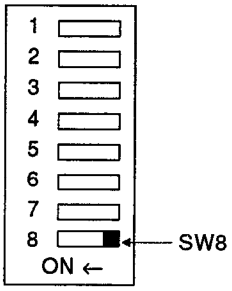
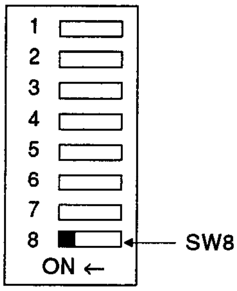
(a) The time is factory-set to 50 msec.

50 msec or 100 msec can be set.

(b) Set the task switching time by the ON/OFF setting of SW8 of switch 2.

Table 5.5 shows the setting conditions of switch 2 for the task switching time.

Table 5.5 Setting the Task Switching Time Using Switch 2

	Scheduled Time of a Task	
	50 msec	100 msec
Setting of switch 2		

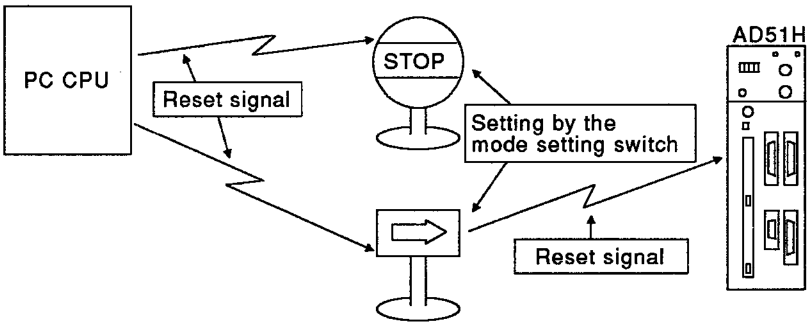
POINT

Section 3.14.1 and the AD51H-BASIC Programming Manual give details about switching of the BASIC program execution.

5.2.6 Setting AD51H's operations when a PC CPU is reset

When a PC CPU is reset, the I/O modules and special-function modules installed to the main base unit and extension base units are initialized and restarted up from the initial state. (This operation is the same as when the power supply is turned ON.)

AD51H operations continue or are restarted according to the valid/invalid setting of the PC CPU reset signal.
(The reset signal is factory-set to invalid.)



(1) Operation when resetting a PC CPU

(a) When the PC CPU reset signal is set to valid, the AD51H is initialized and restarts from the initial state.(The same operation as when the power supply is turned ON.)

(Section 3.12 gives the AD51H operations.)

(b) If the PC CPU reset signal is set to invalid, the AD51H continues operations.

The followings occur since the PC CPU has been reset.

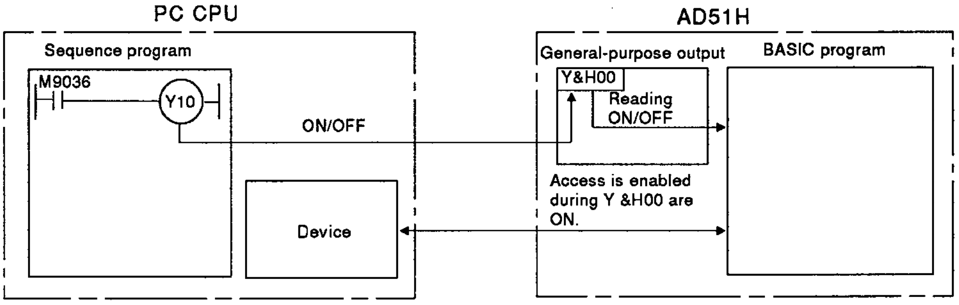
1) ON/OFF outputs controlled by the PC CPU turn OFF.

When the power supply to the PC CPU is turned ON, the task start signal goes ON, and any task to be started by an interruption signal from the PC CPU will be executed.

2) When PC CPU reset processing is executed (five seconds during a reset by the RUN keyswitch), the AD51H cannot access that PC CPU.

If the AD51H uses a BASIC program PCRD/PCWT commands to access the PC CPU, and if that PC CPU is reset, the PCRD/PCWT commands will cause a time excess error.

Use the output (Y) from the PC CPU as an interlocking signal so that the AD51H can access the PC CPU only when the interlock (output) signal is ON.



POINT

When a file in a memory card, FD or HD is being open, and if the AD51H executes reset processing, data in the opened file is destroyed.

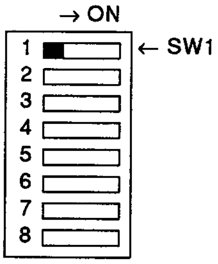
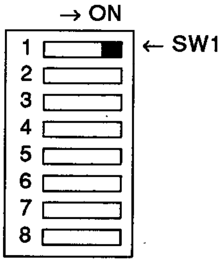
The PC CPU reset signal must be set to invalid when the AD51H is handling data in a file of a memory card, FD or HD.

(2) Valid/invalid setting of the PC CPU reset signal

Valid/invalid setting of the PC CPU reset signal is determined by the ON/OFF setting of SW1 of switch 2 on the left side of the AD51H.

Table 5.6 shows valid/invalid setting of the PC CPU reset signal using switch 2.

Table 5.6 Valid/Invalid Setting of the PC CPU Reset Signal

	Setting the Reset Signal to Invalid	Setting the Reset Signal to Valid
Setting of switch 2		

5.2.7 Setting the time to access a PC CPU after the AD51H is reset

(1) Operations when the AD51H is reset

(a) Communications cannot be executed with a PC CPU until the AD51H is initialized and started up from the initial state after using the RUN keyswitch to reset the AD51H.

If read/write from/to the AD51H's buffer is executed by the FROM/TO instruction from a PC CPU while the AD51H is executing reset processing, this causes a SP UNIT DOWN error and sequence program operations will stop.

(b) The multitask execution start flag (X1B) turns OFF while the AD51H executes reset processing.

The PC CPU access time needs to be set so that a PC CPU surely detects that the multitask execution start flag (X1B) is OFF after the AD51H is reset.

Set this time longer than the PC CPU scan time.

(2) Access time setting

(a) The access time is factory-set to 200 msec.

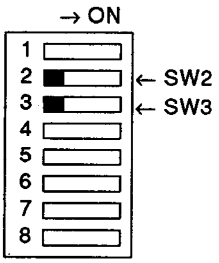
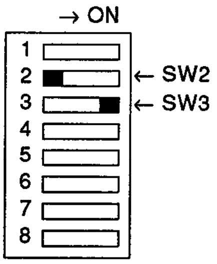
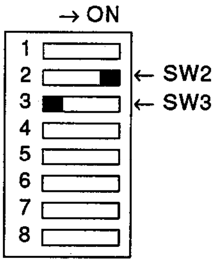
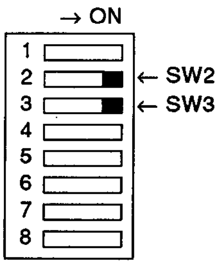
When the PC CPU scan time is 200 msec or more, set a value longer than the scan time as the PC CPU access time.

200 msec, 500 msec, 1000 msec, or 2000 msec can be set.

(b) Set the access time by the ON/OFF setting of SW2 to SW3 of switch 2 on the left side of the AD51H.

Table 5.7 shows the setting of the access time by the switch.

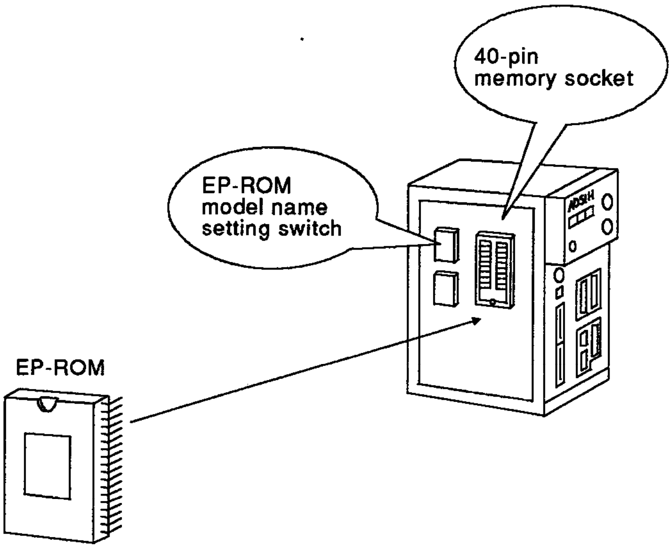
Table 5.7 Setting of the Access Time by the Switch 2

	Access Time			
	200 msec	500 msec	1000 msec	2000 msec
Setting of switch 2	 → ON 1 2 3 4 5 6 7 8 ← SW2 ← SW3	 → ON 1 2 3 4 5 6 7 8 ← SW2 ← SW3	 → ON 1 2 3 4 5 6 7 8 ← SW2 ← SW3	 → ON 1 2 3 4 5 6 7 8 ← SW2 ← SW3

5.2.8 Setting the EP-ROM model name

Either a 64K ROM, 128K ROM, or 256K ROM can be used with the AD51H.

The EP-ROM model name setting designates whether a 64K ROM, 128K ROM, or 256K ROM is installed in the ROM socket of the AD51H.

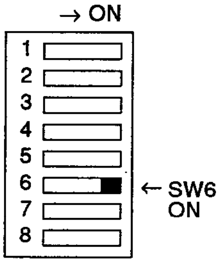
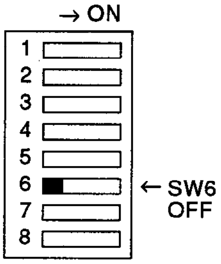


(1) How to set the EP-ROM model name

Set the EP-ROM model name by the ON/OFF setting of SW6 of the switch 2 on the left side of the AD51H.

Table 5.8 shows the setting of the EP-ROM model name by mode setting switch 2.

Table 5.8 Setting of the EP-ROM Model Name

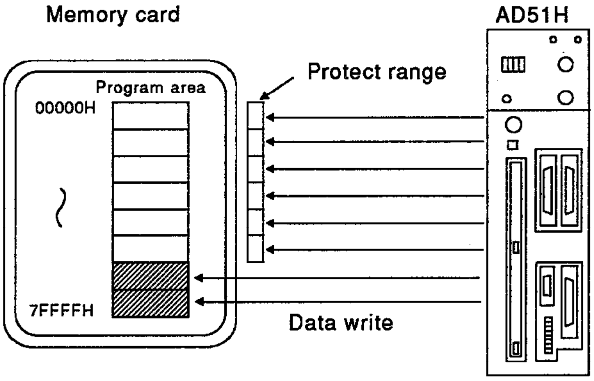
EP-ROM Model Name	Setting of SW6
64K ROM or 128K ROM	
256K ROM	

5.2.9 Setting the memory-protect range of a memory card

This section explains the setting of the protective range of the execution program area in a memory card installed in MEMORY CARD1.

Only execution programs stored in a memory card can be protected, and the rest of the area (except for execution programs) can be used for storing data files.

(If memory-protect is set by the memory card's protect switch, all areas of the memory card are set to memory-protect, and the area except the execution program area cannot be used for data storage.

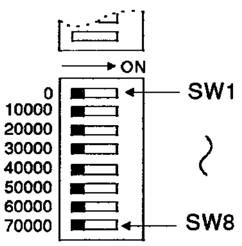


(1) Memory-protect is set by SW1 to SW8 of the memory-protect range setting switch.
If the memory-protect range setting switch is ON, memory-protect is set.

(2) Memory-protect can be set for the 512K-byte area beginning with the head of the memory area.
The memory-protect area can be divided into eight areas each of which has 64K bytes, and memory-protect can be set in units of 64K bytes. Set the memory-protect range from 0H continuously to the end address of the range.

(3) Table 5.9 shows the memory-protect range that can be set by the memory-protect range setting switch.

Table 5.9 Memory-Protect Range

Memory-Protect Range	Setting of the Memory-Protect Setting Switch	
0H to 0FFFFH	SW1	 ON 0 10000 20000 30000 40000 50000 60000 70000 SW1 SW8
10000H to 1FFFFH	SW2	
20000H to 2FFFFH	SW3	
30000H to 3FFFFH	SW4	
40000H to 4FFFFH	SW5	
50000H to 5FFFFH	SW6	
60000H to 6FFFFH	SW7	
70000H to 7FFFFH	SW8	

ON : Memory-protect is set.
OFF : Memory-protect is canceled.

POINT

The range set by the memory-protect setting switch is valid only when the memory-protect keyswitch is turned ON.

5.3 Handling and Installation

This section explains the handling instructions for, and the installation environment of the AD51H.

5.3.1 Handling instructions

This section gives the proper procedures for handling the AD51H.

(1) Since the case is made of plastic, do not drop the AD51H or subject it to mechanical shock.

(2) Do not remove the printed circuit board of any module from its case.

(3) When wiring, make sure wire offcuts do not get into the module.

(4) Tighten the module mounting screws (if necessary) as shown in the following table:

Screw	Tightening Torque
Module screws (optional) (M4 x 0.7)	78 to 118 N·cm

(5) To attach the AD51H to the base, press the AD51H against the base so that the latch locks securely. To detach the AD51H, push in the latch. Then, after the latch is disengaged from the base, pull the AD51H toward you.

(The User's Manual of each building block-type CPU gives details.)

5.3.2 Installation environment

Never install the system in the following environments:

(1) Locations where the ambient temperature is outside the range of 0 to 55°C.

(2) Locations where the ambient humidity is outside the range of 10 to 90%RH.

(3) Locations where dew condensation takes place due to sudden temperature changes.

(4) Locations with corrosive or combustible gas.

(5) Locations with a high level of conductive powder such as dust and iron filings, oil mist, salt, and organic solvent.

(6) Locations exposed to the direct sunlight.

(7) Locations where strong power and magnetic fields are generated.

(8) Locations where vibration and shock can be directly transmitted to the system modules.

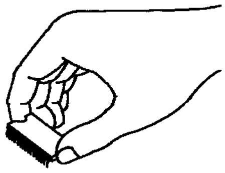
5.3.3 Inserting and removing the EP-ROM

The section gives the procedures and precautions when inserting or removing an EP-ROM in or out of the ROM socket.

(1) Inserting the EP-ROM

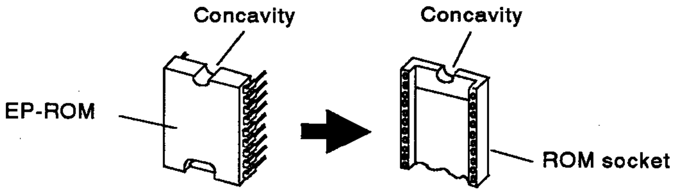
(a) Do not touch memory lead pins with the hand or fingers.

Static electricity may clear the memory, or a pin may be bent.



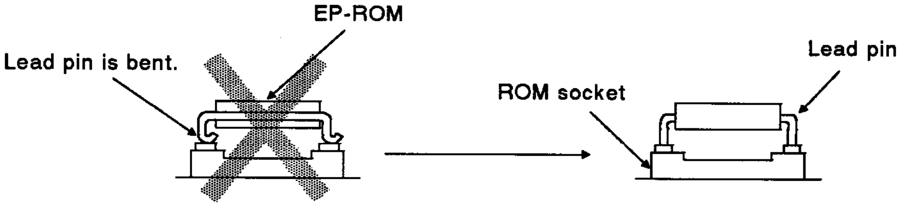
(b) When inserting the EP-ROM, make sure the concavity of the EP-ROM matches the concavity of the ROM socket.

If the EP-ROM is inserted incorrectly, the memory will be cleared when the power is turned ON.

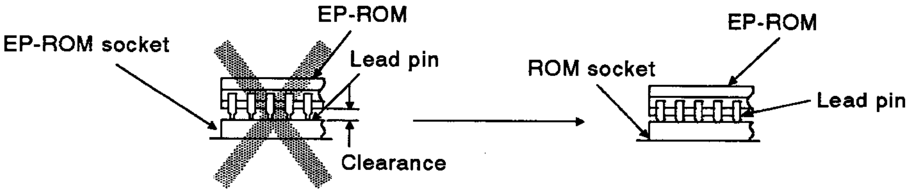


(c) Be very careful not to bend EP-ROM memory lead pins when inserting the EP-ROM.

If an EP-ROM pin is bent, when the power supply is turned ON, normal reading cannot be executed and the memory may even be cleared.



(d) After the EP-ROM is inserted, make sure it is set firmly in the socket without a clearance.



(2) Removing the EP-ROM

Use a special pulling tool (for 40-pin ICs) to remove the EP-ROM from the ROM socket.

5.3.4 Inserting and removing memory cards

(1) How to insert and remove memory cards when the power is OFF

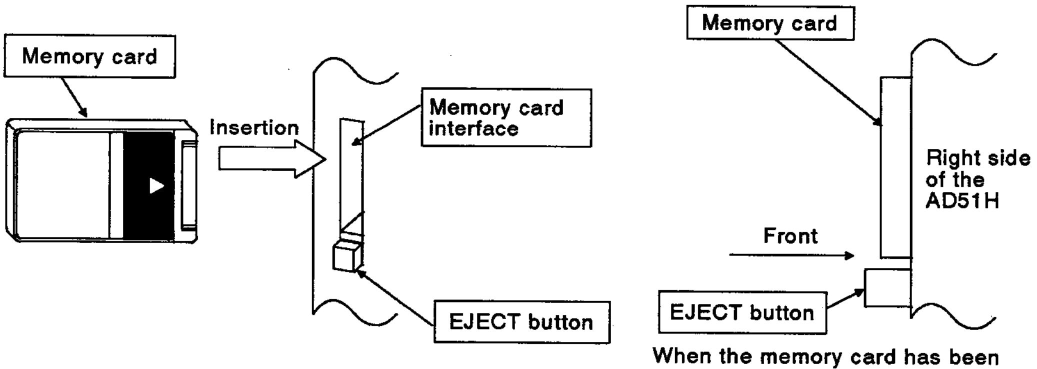
When inserting or removing a memory card while the power is ON, follow the procedure given in 5.3.4 (2).

(a) Inserting a memory card

When inserting a memory card, make sure the card is inserted with the arrow side down. Push the memory card in firmly until it is securely installed in the connector.

If the memory card is properly installed in the connector, the EJECT button will move out.

After inserting the memory card, make sure that the EJECT button has moved out, as shown below.



When the memory card has been properly inserted, the EJECT button has moved out from the unit.

(b) Removing a memory card

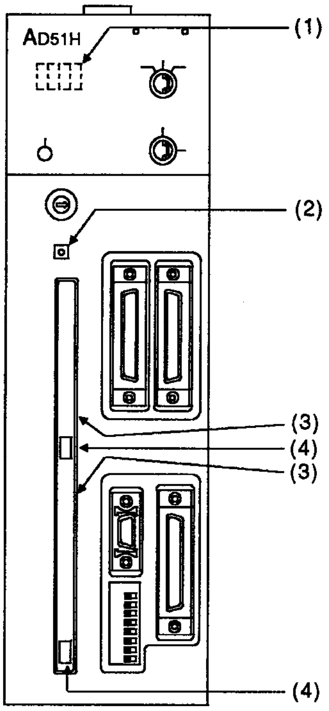
Press the EJECT button firmly to remove the memory card.

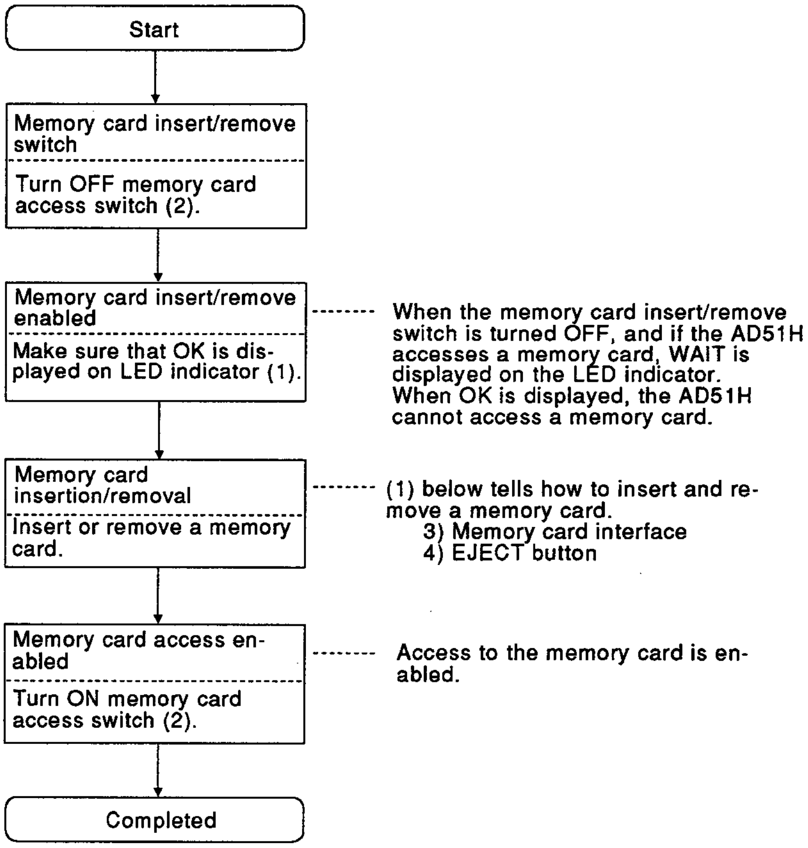
Pressing the EJECT button dislodges the memory card from the connector and slides it forward. The card can then be pulled out by hand.

(2) How to insert and remove memory cards when the power is ON

When the power is ON, the following procedure must be followed to insert or remove memory cards (after making sure the AD51H is not accessing a memory card).

Front side of AD51H





Start

Memory card insert/remove switch
Turn OFF memory card access switch (2).

Memory card insert/remove enabled
Make sure that OK is displayed on LED indicator (1).

When the memory card insert/remove switch is turned OFF, and if the AD51H accesses a memory card, WAIT is displayed on the LED indicator.
When OK is displayed, the AD51H cannot access a memory card.

Memory card insertion/removal
Insert or remove a memory card.

(1) below tells how to insert and remove a memory card.
3) Memory card interface
4) EJECT button

Memory card access enabled
Turn ON memory card access switch (2).

Access to the memory card is enabled.

Completed

POINTS

(1) When inserting or removing a memory card when the power is ON, follow the procedure shown in (2) above.

If a memory card is inserted or removed while the AD51H is accessing the memory card, the data in that memory card will be cleared.

(2) The memory card access switch is used for both MEMORY CARD1 and MEMORY CARD2.

If the memory card access switch is turned ON, the AD51H cannot access a memory card in MEMORY CARD1 or MEMORY CARD2.

(3) Memory cards can be accessed after accessibility to the memory card is confirmed by a special relay (EM9201) and a special register (ED9021).

(4) If the AD51H accesses a memory card in the file maintenance mode of an SW1IX-AD51HPE, do not turn OFF the memory card access switch.

If the memory card access switch is turned OFF, processing will be interrupted.

5.4 Precautions on Wiring

External wiring should be noise-resistant so that the functions of the AD51H can be fully performed with a high system reliability.

The following precautions must be taken when doing external wiring to the AD51H.

(1) Do not put external wiring near (closer than 100 mm) a main circuit cable, a high-tension line, or a load-carrying wire (other than the load-carrying wire of a PC).

Do not bundle the wires together.

If these precautions are not taken, the AD51H may be subject to noise, surges, and/or inductions.

(2) The FG terminal of a power supply module is not connected to the FG terminal of the RS-232C or RS-422 interface of the AD51H.

The shield of a shielded cable or wire should be connected to the ground of the external device by one-point grounding.

5.5 Memory card Battery Replacement

This section gives the period and method of replacement of the battery in a memory card.

5.5.1 When Should the Battery be Replaced?

(1) Indicates low battery power

The AD51H verifies whether battery power to a memory card inserted in MEMORY CARD1 or MEMORY CARD2 is low.

When the AD51H detects low battery power, an error message is displayed on the LED indicator.

Error Message	Meaning
BAT1	Indicates low battery power to the memory card in MEMORY CARD1
BAT2	Indicates low battery power to the memory card in MEMORY CARD2

(2) When should the battery be replaced?

The contents of a memory card can be retained about 168 hours (one week) after the low battery error message is displayed on the LED indicator.

However, when the low battery error message is displayed, the battery should be replaced as soon as possible.

(3) Battery life

The memory card is battery-backed. Mitsubishi recommends that the battery should be replaced at certain intervals as shown below.

Operating Condition \ Model		A6MEM-256KAW	A6MEM-512KAW
When the memory card is battery-backed	Minimum guaranteed lifetime	1900 Hr	900 Hr
	Real lifetime (typical)	11300 Hr	5900 Hr
Working time after the low battery error message is given (ambient temperature: 25°C)		79 Hr	39 Hr

Note: Mitsubishi cannot guarantee the backup performance if a new battery is used after five years of storage.

POINT

When the memory card is installed in MEMORY CARD1 or MEMORY CARD2 of the AD51H, the data in that memory card is backed up by the power of the AD51H.

5.5.2 Replacing the Battery

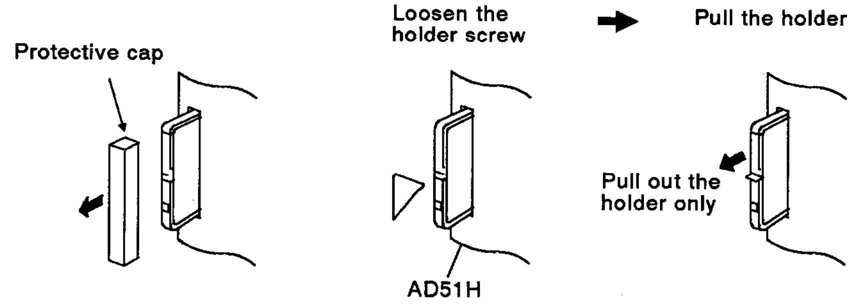
To replace the battery, install a memory card in MEMORY CARD1 or MEMORY CARD2, and make sure that power to the AD51H module is ON.

If this is not followed, all data in the memory card will be cleared because the memory card is not backed up by a capacitor.

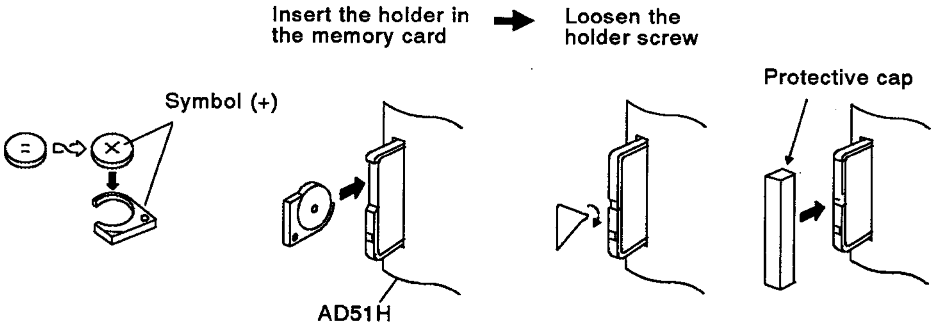
Be sure to turn OFF the power to the output modules of the PC system to which the AD51H is installed to prevent static electricity from damaging data in the memory card.

[Replacement procedure]

1) Prepare a lithium battery (model BR2325 or equivalent).
2) Insert a memory card into the memory card interface on the module.
   (Section 5.3.4 gives how to install a memory card in the AD51H.)
3) Set the RUN keyswitches of the AD51H and the PC CPU to the STOP position.
4) Make sure that power to the output modules is OFF. Then, turn ON the PC CPU power.
5) Remove the protective cap from the battery holder of the memory card.
6) Loosen the holder screw 4 to 5 mm, and pull the battery holder from the memory card.



7) Put the new battery in the battery holder, and install the battery holder to the memory card as described in 4. BATTERY INSTALLATION.



8) Attach the protective cap to the battery side of the memory card.

POINT

The instruction manual included in the memory card package gives details about memory card handling.

5.6 Messages displayed on the led indicator

Four types of messages are displayed on the LED indicator on the front of the AD51H:

(a) Operating state messages

(b) Warning messages

(c) BASIC program error messages

(d) System-down messages

(1) Priorities and indicator resetting

The LED indicator of the AD51H displays messages in order of priority. Messages of higher priority are displayed first.

Some of them can be cleared by the indicator reset switch on the front of the AD51H.

The display priority and clear enable/disable setting of the indicator reset switch are shown below:

	Display priority	Indicator reset
(a) Operating state messages	Low	Disabled
(b) Warning messages	↓	Enabled
(c) BASIC program error messages		Enabled
(d) System-down messages	High	Disabled

5.6.1 Operating State Messages

There are seven kinds of operating state messages as shown in Table 5.10.

Table 5.10 AD51H Operating State Messages

Message	Display Mode			Description
	Programming	Debugging	Execution	
BOOT	o	o	o	• Displayed when the power supply to the AD51H is turned ON or the AD51H is reset, and the AD51H system is booted.
CARD	o	o	o	• Displayed when the memory card install request switch is turned ON while a memory card is accessed. (OK is displayed when a memory card can be inserted or removed.)
INIT			o	• Displayed when an execution program is being loaded from a memory card or an EP-ROM to the AD51H.
OK			o	• Indicates that a memory card can be inserted or removed. • When the memory card install request switch is ON and if a file in the memory card is being accessed, this is displayed when the memory card is closed.
PROG	o			• Displayed when the mode selection rotary switch is set to programming mode (4).
RUN			o	• Multitasking is being executed.
STOP			o	• Displayed when multitasking is stopped because the RUN keyswitch is set to STOP.

5.6.2 Warning Messages

Warning messages are displayed when an error which does not lead to a system down of the AD51H occurs.

There are two kinds of warning messages as shown in Table 5.11.

Table 5.11 AD51H Warning Messages

Message	Error Code	Display Mode			Description	Corrective Action
		Programming	Debugging	Execution		
BAT1	0F 80	o	o	o	• Displayed when the battery of the memory card installed in MEMORY CARD1 is low.	Replace the battery (see Section 5.2).
BAT2	0F 81	o	o	o	• Displayed when the battery of the memory card installed in MEMORY CARD 2 is low.	Replace the battery (see Section 5.2).

* Error codes are stored in addresses ED9001 to ED9008.

5.6.3 System-Down Messages

System-down messages are displayed when an error which prevents the system from returning to normal occurs.

There are four kinds of system-down messages as shown in Table 5.12.

Table 5.12 AD51H System-down Messages

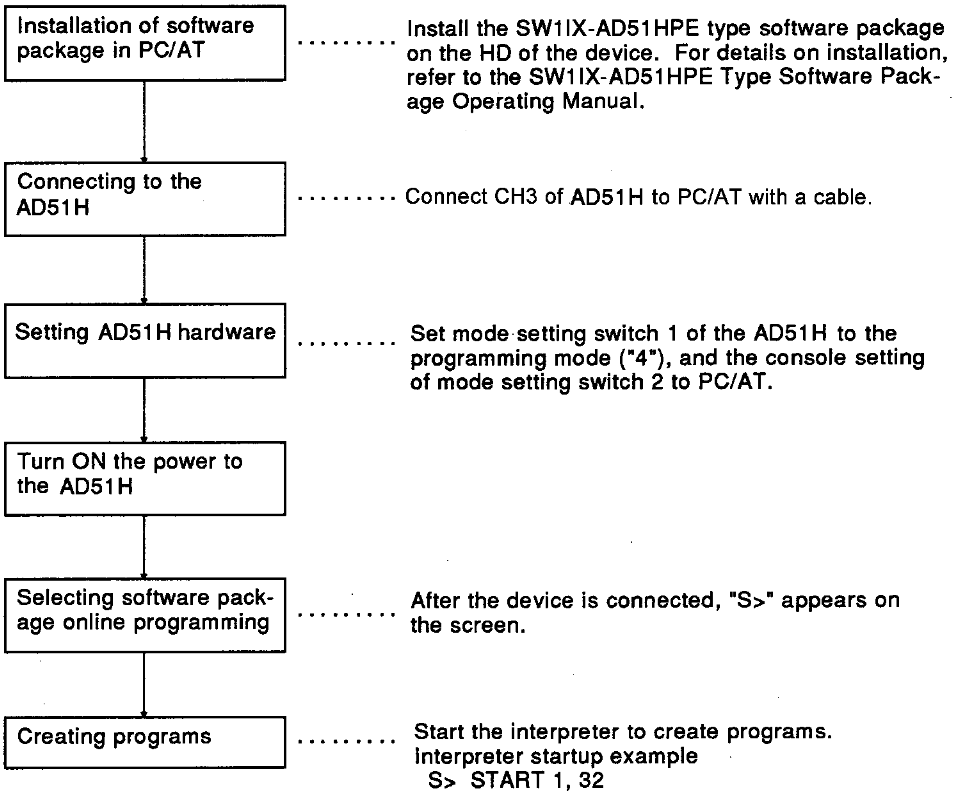
Message	Error Code	Display Mode			Description	Corrective Action
		Programming	Debugging	Execution		
INIE	0F 20			o	• Displayed when a memory card or EP-ROM in which an execution program is stored has not been installed.	• Install a memory card in which an execution program is stored. • Install an EP-ROM in which an execution program is stored.
KEYE	0F 10	o	o	o	• Displayed when the power supply to the AD51H is turned ON with the RUN keyswitch set to RESET.	• Set the RUN keyswitch to RUN or STOP, and turn ON the power supply again or reset the AD51H.
MTSE	0F 21			o	• Displayed when multitasking is incorrectly set.	• Reset multitasking correctly.
WDTE	0F 00	o	o	o	• Displayed when a WDT error occurs.	• Hardware fault Consult your nearest Mitsubishi representative.

6. AD51H STARTUP PROCEDURE

This chapter describes the steps from the connection of a console to the AD51H to starting up the AD51H.

6.1 Startup when a PC/AT is Used as the Console

The following flowchart shows the steps for starting a PC/AT when it is used as the console.


Installation of software package in PC/AT
Connecting to the AD51H
Setting AD51H hardware
Turn ON the power to the AD51H
Selecting software package online programming
Creating programs

Install the SW1IX-AD51HPE type software package on the HD of the device. For details on installation, refer to the SW1IX-AD51HPE Type Software Package Operating Manual.

Connect CH3 of AD51H to PC/AT with a cable.

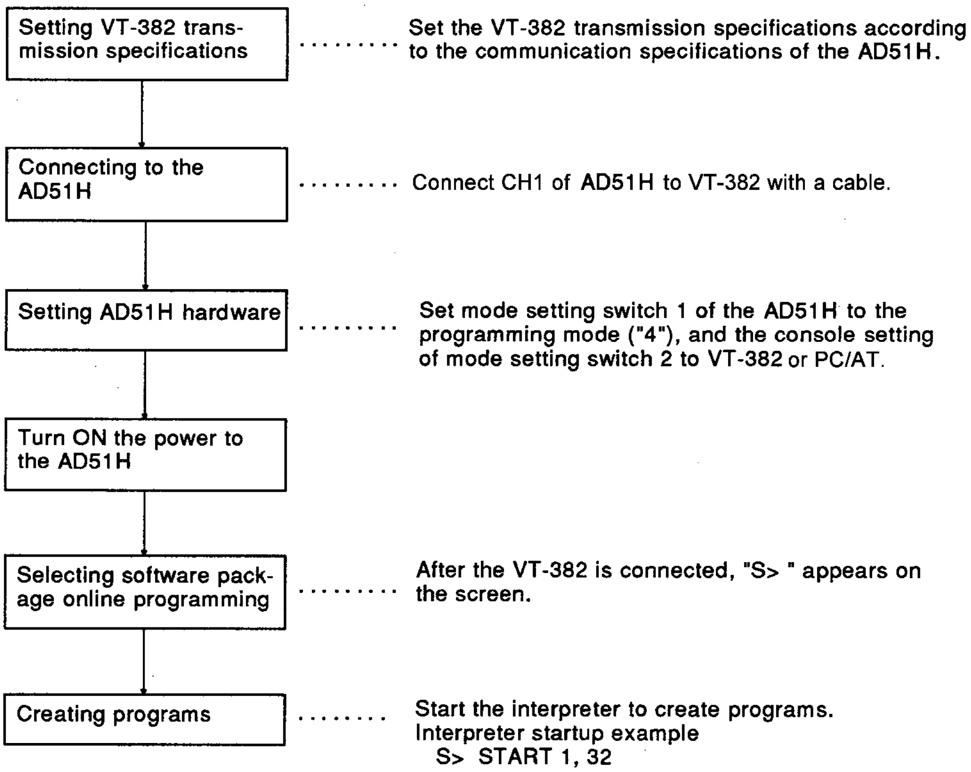
Set mode setting switch 1 of the AD51H to the programming mode ("4"), and the console setting of mode setting switch 2 to PC/AT.

After the device is connected, "S>" appears on the screen.

Start the interpreter to create programs.
Interpreter startup example
S> START 1, 32

6.2 Startup when a VT-382 is Used as the Console

The following flowchart shows the steps for starting a VT-382 when it is used as the console.



Setting VT-382 transmission specifications .... Set the VT-382 transmission specifications according to the communication specifications of the AD51H.

Connecting to the AD51H .... Connect CH1 of AD51H to VT-382 with a cable.

Setting AD51H hardware .... Set mode setting switch 1 of the AD51H to the programming mode ("4"), and the console setting of mode setting switch 2 to VT-382 or PC/AT.

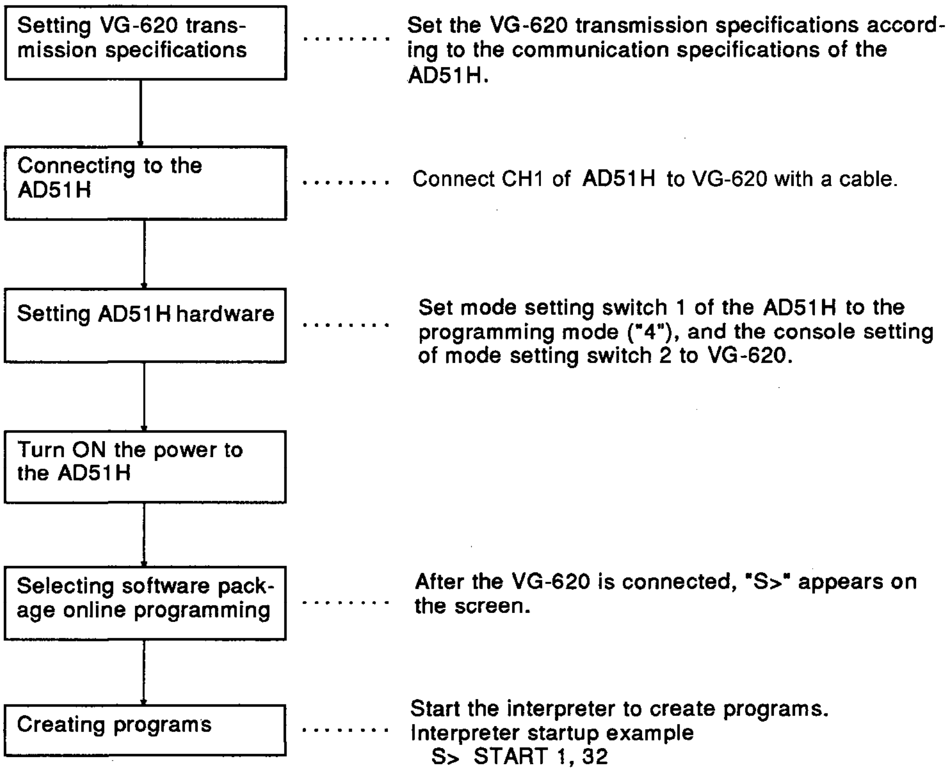
Turn ON the power to the AD51H

Selecting software package online programming .... After the VT-382 is connected, "S> " appears on the screen.

Creating programs .... Start the interpreter to create programs.
Interpreter startup example
S> START 1, 32

6.3 Startup when a VG-620 is used as the Console

The following flowchart shows the steps for starting a VG-620 when it is used as the console:



Setting VG-620 transmission specifications .... Set the VG-620 transmission specifications according to the communication specifications of the AD51H.

Connecting to the AD51H .... Connect CH1 of AD51H to VG-620 with a cable.

Setting AD51H hardware .... Set mode setting switch 1 of the AD51H to the programming mode ("4"), and the console setting of mode setting switch 2 to VG-620.

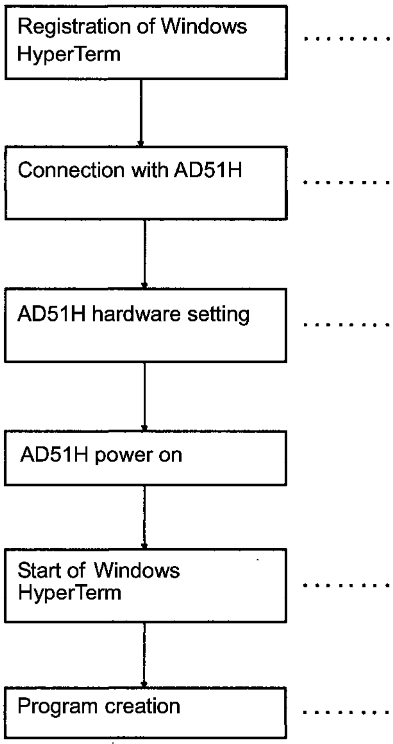
Turn ON the power to the AD51H

Selecting software package online programming .... After the VG-620 is connected, "S>" appears on the screen.

Creating programs .... Start the interpreter to create programs.
Interpreter startup example
S> START 1, 32

6.4 Startup when a PC/AT is used as the Console (When general-purpose editor is used)

The following provides a procedure to start up the a PC/AT using the general-purpose editor as a console.



Registration of Windows HyperTerm .... Register Windows HyperTerm to the HD. Refer to the Windows manual for registration.

Connection with AD51H .... Connect the AD51H CH1 and PC/AT by a cable.

AD51H hardware setting .... Set the mode setting switch 1 of the AD51H to the programming mode (set to "4") and the console setting of the mode setting switch 2 to VT382 or PC/AT.

AD51H power on

Start of Windows HyperTerm .... At a first startup, make the initial setting of Windows HyperTerm. Refer to the following manual for the Windows HyperTerm setting method.
AD51H-BASIC Programming Manual (Program Edit/Compile)
SH-080091

Program creation .... **Start the interpreter to create programs.**
Interpreter startup example

```
  S> START 1, 32
```

APPENDICES

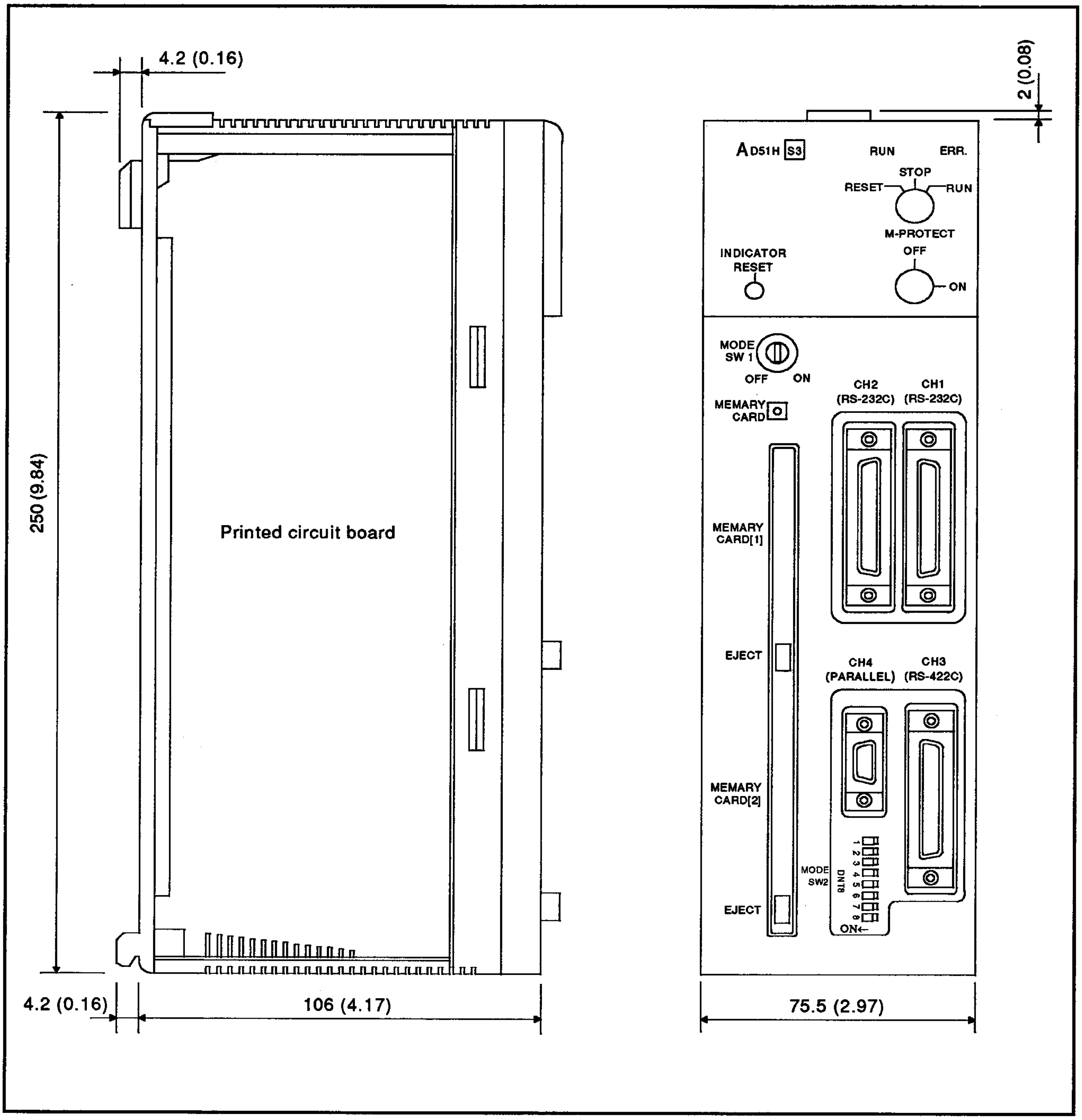
APPENDIX 1 FUNCTIONAL COMPARISON BETWEEN A1SD51S, AD51H-S3, AD51H AND AD51 (S3)

Table 1.1 compares functions between the A1SD51S, the AD51H-S3, the AD51H and the AD51 (S3).

Table 1.1 Functional Comparison between A1SD51S, AD51H-3, AD51H and AD51 (S3)

Item		A1SD51S	AD51H-S3	AD51H	AD51(S3)
CPU element		80C186(15MGz)			HD64180(8MHz)
Operation element		Not equipped			Equipped
Programming language		AD51H-BASIC (interpreter, compiler)		AD51H-BASIC (interpreter)	GPC-BASIC (interpreter)
Number of tasks		2	8		8
Task start condition		• When the power is turned ON. • When an interrupt is issued from the PC CPU(but impossible during compilation). • When another tasks issues a start request.			• When the power is turned ON. • When an interrupt is given from the PC CPU (for 1 task) • When a real time interrupt occurs.
Internal memory	Program	64 Kbytes (16/32 Kbytes selective for 1 task)	384kbytes (16/32/48/64 Kbytes selective for 1 task)		64 Kbytes + 48 Kbytes ↓ For storage to ROM
	Common memory	8 Kbytes			2 Kbytes
	Buffer memory	6 Kbytes			6 Kbytes
	EM	1024 points			——
	ED	1024 points			——
General-purpose I/O		Input: 27 points Output: 19 points (2 points: For task startup from PC CPU)	Input: 27 points Output: 26 points (9 points: For task startup from PC CPU)		Input: 26 points Output: 17 points
Memory protect		Available (EEP-ROM can be write-protect.)	Available (Memory card can be partly protected.)		Available
Interface	RS-232C	2 channel			2 channels
	RS-422	1 channel			2 channels
	Parallel	——	1 channel		——
	Memory card	——	2 channels		——
Clock function		Not equipped			Equipped
Power interruption backup		Equipped	Not equipped (Power interruption backup available for memory card)		Equipped

APPENDIX 2 DIMENSIONS



Unit: mm (in.)

WARRANTY

Please confirm the following product warranty details before starting use.

1. Gratis Warranty Term and Gratis Warranty Range

If any faults or defects (hereinafter "Failure") found to be the responsibility of Mitsubishi occurs during use of the product within the gratis warranty term, the product shall be repaired at no cost via the dealer or Mitsubishi Service Company. Note that if repairs are required at a site overseas, on a detached island or remote place, expenses to dispatch an engineer shall be charged for.

[Gratis Warranty Term]

The gratis warranty term of the product shall be for one year after the date of purchase or delivery to a designated place.
Note that after manufacture and shipment from Mitsubishi, the maximum distribution period shall be six (6) months, and the longest gratis warranty term after manufacturing shall be eighteen (18) months. The gratis warranty term of repair parts shall not exceed the gratis warranty term before repairs.

[Gratis Warranty Range]

(1) The range shall be limited to normal use within the usage state, usage methods and usage environment, etc., which follow the conditions and precautions, etc., given in the instruction manual, user's manual and caution labels on the product.

(2) Even within the gratis warranty term, repairs shall be charged for in the following cases.

1. Failure occurring from inappropriate storage or handling, carelessness or negligence by the user. Failure caused by the user's hardware or software design.
2. Failure caused by unapproved modifications, etc., to the product by the user.
3. When the Mitsubishi product is assembled into a user's device, Failure that could have been avoided if functions or structures, judged as necessary in the legal safety measures the user's device is subject to or as necessary by industry standards, had been provided.
4. Failure that could have been avoided if consumable parts (battery, backlight, fuse, etc.) designated in the instruction manual had been correctly serviced or replaced.
5. Failure caused by external irresistible forces such as fires or abnormal voltages, and Failure caused by force majeure such as earthquakes, lightning, wind and water damage.
6. Failure caused by reasons unpredictable by scientific technology standards at time of shipment from Mitsubishi.
7. Any other failure found not to be the responsibility of Mitsubishi or the user.

2. Onerous repair term after discontinuation of production

(1) Mitsubishi shall accept onerous product repairs for seven (7) years after production of the product is discontinued. Discontinuation of production shall be notified with Mitsubishi Technical Bulletins, etc.

(2) Product supply (including repair parts) is not possible after production is discontinued.

3. Overseas service

Overseas, repairs shall be accepted by Mitsubishi's local overseas FA Center. Note that the repair conditions at each FA Center may differ.

4. Exclusion of chance loss and secondary loss from warranty liability

Regardless of the gratis warranty term, Mitsubishi shall not be liable for compensation to damages caused by any cause found not to be the responsibility of Mitsubishi, chance losses, lost profits incurred to the user by Failures of Mitsubishi products, damages and secondary damages caused from special reasons regardless of Mitsubishi's expectations, compensation for accidents, and compensation for damages to products other than Mitsubishi products and other duties.

5. Changes in product specifications

The specifications given in the catalogs, manuals or technical documents are subject to change without prior notice.

6. Product application

(1) In using the Mitsubishi MELSEC programmable logic controller, the usage conditions shall be that the application will not lead to a major accident even if any problem or fault should occur in the programmable logic controller device, and that backup and fail-safe functions are systematically provided outside of the device for any problem or fault.

(2) The Mitsubishi general-purpose programmable logic controller has been designed and manufactured for applications in general industries, etc. Thus, applications in which the public could be affected such as in nuclear power plants and other power plants operated by respective power companies, and applications in which a special quality assurance system is required, such as for Railway companies or National Defense purposes shall be excluded from the programmable logic controller applications.
Note that even with these applications, if the user approves that the application is to be limited and a special quality is not required, application shall be possible.
When considering use in aircraft, medical applications, railways, incineration and fuel devices, manned transport devices, equipment for recreation and amusement, and safety devices, in which human life or assets could be greatly affected and for which a particularly high reliability is required in terms of safety and control system, please consult with Mitsubishi and discuss the required specifications.

Intelligent Communication Module Type AD51H-S3

User's Manual

MODEL	AD51H-S3-U-E
MODEL CODE	13JE16
IB(NA)-66401-B(0409)MEE	

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : 1-8-12, OFFICE TOWER Z 14F HARUMI CHUO-KU 104-6212,JAPAN
NAGOYA WORKS : 1-14 , YADA-MINAMI 5-CHOME , HIGASHI-KU, NAGOYA , JAPAN

When exported from Japan, this manual does not require application to the Ministry of Economy, Trade and Industry for service transaction permission.

Specifications subject to change without notice.