CONTENTS

COMMON

COMPUTER LINK FUNCTION

PRINTER FUNCTION

5.7 Device Memory Read/Write

This section describes how to designate the control procedure to read or write data from or to device memory, and designation examples.

5.7.1 Commands and device ranges

(1) The ACPU common commands and device ranges used for device memory read/write are described below.

(a) ACPU common commands

Item		Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
		Symbol	ASCII Code			During STOP	During RUN		
							Write Enabled	Write Disabled	
Batch read	Bit units	BR	42H, 52H	Reads bit devices (X, Y, M, etc.) in units of 1 point.	256 points	O	O	O	5.7.2
	Word units	WR	57H, 52H	Reads bit devices (X, Y, M, etc.) in units of 16 points.	32 words (512 points)				5.7.3
				Reads word devices (D, R, T, C, etc.) in units of 1 point.	64 points				
Batch write	Bit units	BW	42H, 57H	Writes data to bit devices (X, Y, M, etc.) in units of 1 point.	160 points	O	O	X	5.7.4
	Word units	WW	57H, 57H	Writes data to bit devices (X, Y, M, etc.) in units of 16 points.	10 words (160 points)				5.7.5
				Writes data to word devices (D, R, T, C, etc.) in units of 1 point.	64 points				
Test (Random write)	Bit units	BT	42H, 54H	Sets/resets bit devices (X, Y, M, etc.) in units of 1 point by designating the devices and device numbers at random.	20 points	O	O	X	5.7.6
	Word units	WT	57H, 54H	Sets/resets bit devices (X, Y, M, etc.) in units of 16 points by designating the devices and device numbers at random.	10 words (160 points)				5.7.7
				Writes data to word devices (D, R, T, C, etc.) in units of 1 point by designating the devices and device numbers at random.	10 points				
Monitor data registration	Bit units	BM	42H, 4DH	Sets the bit devices (X, Y, M, etc.) to be monitored in units of 1 point.	40 points*	O	O	O	5.7.8 (2)
	Word units	WM	57H, 4DH	Sets the bit devices (X, Y, M, etc.) to be monitored in units of 16 points.	20 words* (320 points)				
				Sets the word devices (D, R, T, C, etc.) to be monitored in units of 1 point.	20 points				
Monitor	Bit units	MB	4DH, 42H	Monitors the devices registered for monitoring.	—	O	O	O	5.7.8 (3)
	Word units	MN	4DH, 4EH						

Note : O.....Executable
X.....Not executable

For the numbers of processing points indicated by an asterisk (*), the number is one half of the values indicated in the table for input devices (X) when PC CPUs other than A3HCPU, AnACPU, AnUCPU or QnACPU are used. (Refer to *1 in Section 3.2.1 (1)).

POINT

When ACPU common commands are used to access the devices in an AnACPU, AnUCPU or QnACPU, the device number ranges described in (b) can be accessed.
Use the AnA/AnUCPU common commands described in (2) to access the extension devices.

(b) Device ranges when ACPU common commands are used

The devices and device number ranges that can be used for the device memory access operation are described below.

1) The device designation code consists of 5 characters.

Leading zeros in the device number (underlined zeros in X0070, for example) can be expressed with a blank code (20H).

Device + Device number = 5 characters

{1 character (2 characters for T/C)} {4 characters (3 characters for T/C)}

Bit Device			Word Device		
Device	**Device Number Ranges (Characters)**	**Decimal/ Hexadecimal Expression**	**Device**	**Device Number Range (Characters)**	**Decimal/ Hexadecimal Expression**
Input X	X0000 to X07FF	Hexadecimal	Timer (present value) T	TN000 to TN255	Decimal
Output Y	Y0000 to Y07FF		Counter (present value) C	CN000 to CN255	
Internal relay M	M0000 to M2047	Decimal	Data register D	D0000 to D1023	
Latch relay L	L0000 to L2047		Link register W	W0000 to W03FF	Hexadecimal
Step relay S	S0000 to S2047		File register R	R0000 to R8191	Decimal
Link relay B	B0000 to B03FF	Hexadecimal	Special register D	D9000 to D9255	
Annunciator F	F0000 to F0255	Decimal			
Special relay M	M9000 to M9255				
Timer (contact) T	TS000 to TS255				
Timer (coil) T	TC000 to TC255				
Counter (contact) C	CS000 to CS255				
Counter (coil) C	CC000 to CC255				

2) Designate the number of device points by converting the number of points for reading/writing to be executed by command (the number should be within the processing points executable in one cycle of communications shown in the table in (a)) into 2-digit ASCII codes (hexadecimal).

Set "00" for 256 points.

(Example: Numbers in parentheses are ASCII codes.)

5 points 05 (30H, 35H)
10 points 0A (30H, 41H)
20 points 16 (31H, 36H)
256 points 00 (30H, 30H)

POINTS

(1) Access the device number range which can be used with common ACPU commands and common AnA/AnU commands and that can be used with the PLC CPU to be accessed.

(2) To designate the bit device ranges in units of words, the bit device number must be a multiple of 16.

(3) Although the ranges are designated for M, L, and S, if the range for M is designated by L or S, the same processing occurs. This is also true for the ranges for L and S.

(4) The ranges of special relays (M9000 to M9255) and special registers (D9000 to D9255) are divided into the areas for read only, write only, and system use.
Trying to write data to the ranges outside the write-only area might cause the PC CPU to malfunction.
The ACPU programming manual gives details concerning special relays and special registers.

(5) When utilizing dedicated instructions for AnACPU and AnUCPU extension file registers with the following software packages, use the commands shown in Section 5.8 to read or write data from or to a file register (R):
- SW0GHP-UTLPC-FN1
- SW0SRX-FNUP

(2) The AnA/AnUCPU common commands and device ranges used for device memory read/write are described below.

(a) AnA/AnUCPU common commands

Item		Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
		Symbol	ASCII Code			During STOP	During RUN		
							Write Enabled	Write Disabled	
Batch read	Bit units	JR	4AH, 52H	Reads bit devices (X, Y, M, etc.) in units of 1 point.	256 points	O	O	O	5.7.2
	Word units	QR	51H, 52H	Reads bit devices (X, Y, M, etc.) in units of 16 points.	32 words (512 points)				5.7.3
				Reads word devices (D, R, T, C, etc.) in units of 1 point.	64 points				
Batch write	Bit units	JW	4AH, 57H	Writes data to bit devices (X, Y, M, etc.) in units of 1 point.	160 points	O	O	X	5.7.4
	Word units	QW	51H, 57H	Writes data to bit devices (X, Y, M, etc.) in units of 16 points.	10 words (160 points)				5.7.5
				Writes data to word devices (D, R, T, C, etc.) in units of 1 point.	64 points				
Test (Random write)	Bit units	JT	4AH, 54H	Sets/resets bit devices (X, Y, M, etc.) in units of 1 point by designating the devices and device numbers at random.	20 points	O	O	X	5.7.6
	Word units	QT	51H, 54H	Sets/resets bit devices (X, Y, M, etc.) in units of 16 points by designating the devices and device numbers at random.	10 words (160 points)				5.7.7
				Writes data to word devices (D, R, T, C, etc.) in units of 1 point by designating the devices and device numbers at random.	10 points				
Monitor data registration	Bit units	JM	4AH, 4DH	Sets the bit devices (X, Y, M, etc.) to be monitored in units of 1 point.	40 points	O	O	O	5.7.8 (2)
	Word units	QM	51H, 4DH	Sets the bit devices (X, Y, M, etc.) to be monitored in units of 16 points.	20 words (320 points)				
				Sets the word devices (D, R, T, C, etc.) to be monitored in units of 1 point.	20 points				
Monitor	Bit units	MJ	4DH, 4AH	Monitors the devices registered for monitoring.	—	O	O	O	5.7.8 (3)
	Word units	MQ	4DH, 51H						

Note : O.....Executable
X.....Not executable

(b) Device ranges when AnA/AnUCPU common commands are used

The devices and device number ranges that can be used for device memory access operation are described below.

1) The device designation code consists of 7 characters.

Leading zeros in the device number (underlined zeros in X000070, for example) can be expressed with a blank code (20H).

Device (1 character / 2 characters for T/C) + Device number (6 characters / 5 characters for T/C) = 7 characters

Bit Device			Word Device		
Device	**Device Number Ranges (Characters)**	**Decimal/ Hexadecimal Expression**	**Device**	**Device Number Range (Characters)**	**Decimal/ Hexadecimal Expression**
Input X	X000000 to X001FFF (X0007FF)	Hexadecimal	Timer (present value) T	TN00000 to TN02047	Decimal
Output Y	Y000000 to Y001FFF (Y0007FF)		Counter (present value) C	CN00000 to CN01023	
Internal relay M	M000000 to M008191	Decimal	Data register D	D000000 to D008191 (D006143)	
Latch relay L	L000000 to L008191		Link register W	W000000 to W001FFF (W000FFF)	Hexadecimal
Step relay S	S000000 to S008191		File register R	R000000 to R008191	Decimal
Link relay B	B000000 to B001FFF (B000FFF)	Hexadecimal	Special register D	D009000 to D009255	
Annunciator F	F000000 to F002047	Decimal			
Special relay M	M009000 to M009255				
Timer (contact) T	TS00000 to TS02047				
Timer (coil) T	TC00000 to TC02047				
Counter (contact) C	CS00000 to CS01023				
Counter (coil) C	CC00000 to CC01023				

* The device number in parentheses represents the maximum device number that can be designated with a computer link module other than the AJ71UC24, A1SJ71UC24-R2/R4/PRF.

2) Designate the number of device points by converting the number of points for reading/writing to be executed by command (the number should be within the processing points executable in one cycle of communications shown in the table in (a)) into 2-digit ASCII codes (hexadecimal).

Set "00" for 256 points.

(Example: Numbers in parentheses are ASCII codes.)

5 points 05 (30H, 35H)
10 points 0A (30H, 41H)
20 points 14 (31H, 34H)
256 points 00 (30H, 30H)

POINTS

(1) Access the device number range which can be used with common ACPU commands and common AnA/AnU commands and that can be used with the PLC CPU to be accessed. (For QnACPU this is the AnACPU range).

(2) To designate the bit device ranges in units of words, the bit device number must be a multiple of 16.
For special relays M, whose device number is M9000 or greater, designation is possible by using "9000 + multiples of 16".

(3) Although the ranges are designated for M, L, and S, if the range for M is designated by L or S, the same processing occurs. This is also true for the ranges for L and S.

(4) The ranges of special relays (M9000 to M9255) and special registers (D9000 to D9255) are divided into the areas for read only, write only, and system use.
Trying to write data to the ranges outside the write-only area might cause the PC CPU to malfunction.
The ACPU programming manual gives details concerning special relays and special registers.

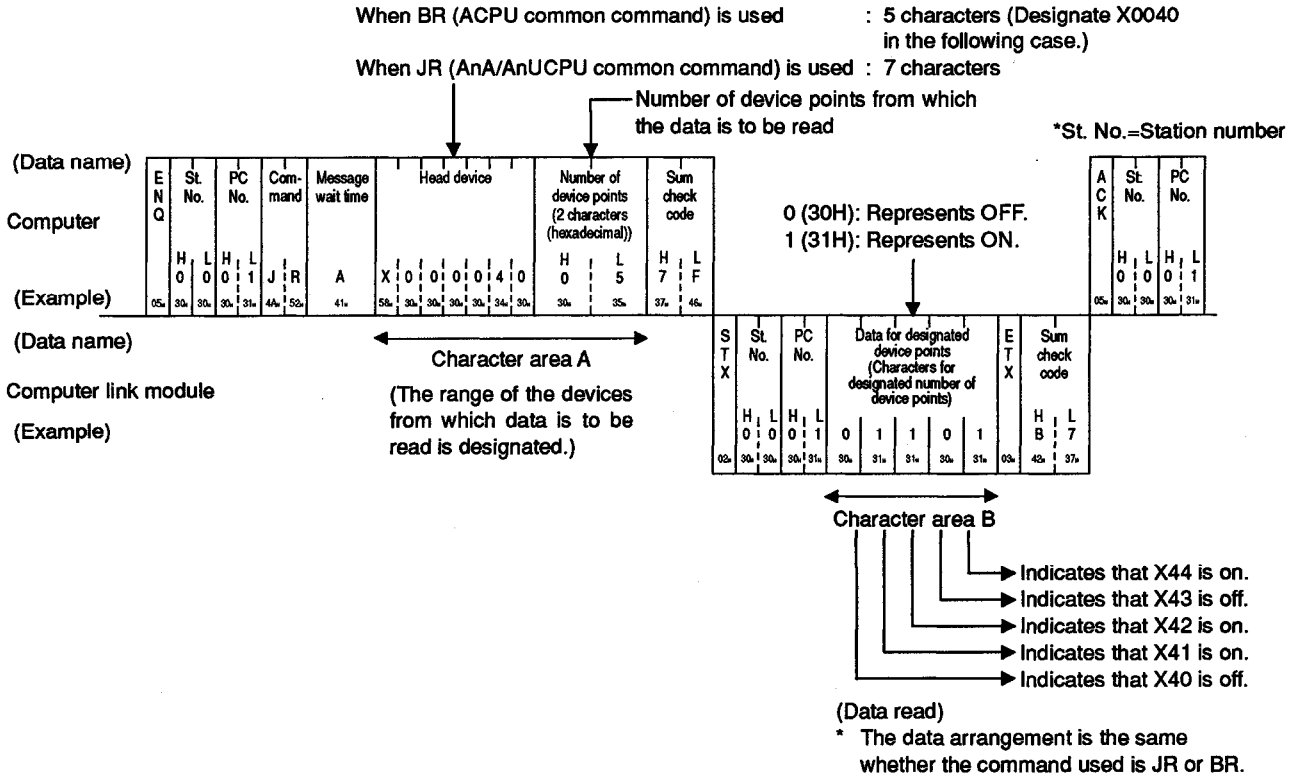
(5) When utilizing dedicated instructions for AnACPU and AnUCPU extension file registers, use the commands shown in Section 5.8 to read or write data from or to a file register (R).

5.7.2 Batch read in units of bits

In this section, an example is quoted to describe the control protocol for batch-reading data from devices in bit device memory using a BR or JR command.

[Control protocol] The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)
1) Message wait time is 100 ms
2) Data is to be read from five devices, X040 to X044.



POINTS

(1) To designate the device range, the following conditions must be met:
- 1 ≤ number of device points ≤ 256 (setting for 256 points is 00H)
- (Head device number) + [(number of device points) - 1] ≤ maximum device number

(2) The message wait time is designated in the range of 0 to 150 ms in units of 10 ms, using hexadecimal notation of 0 through FH. Therefore, 100 ms corresponds to "A".

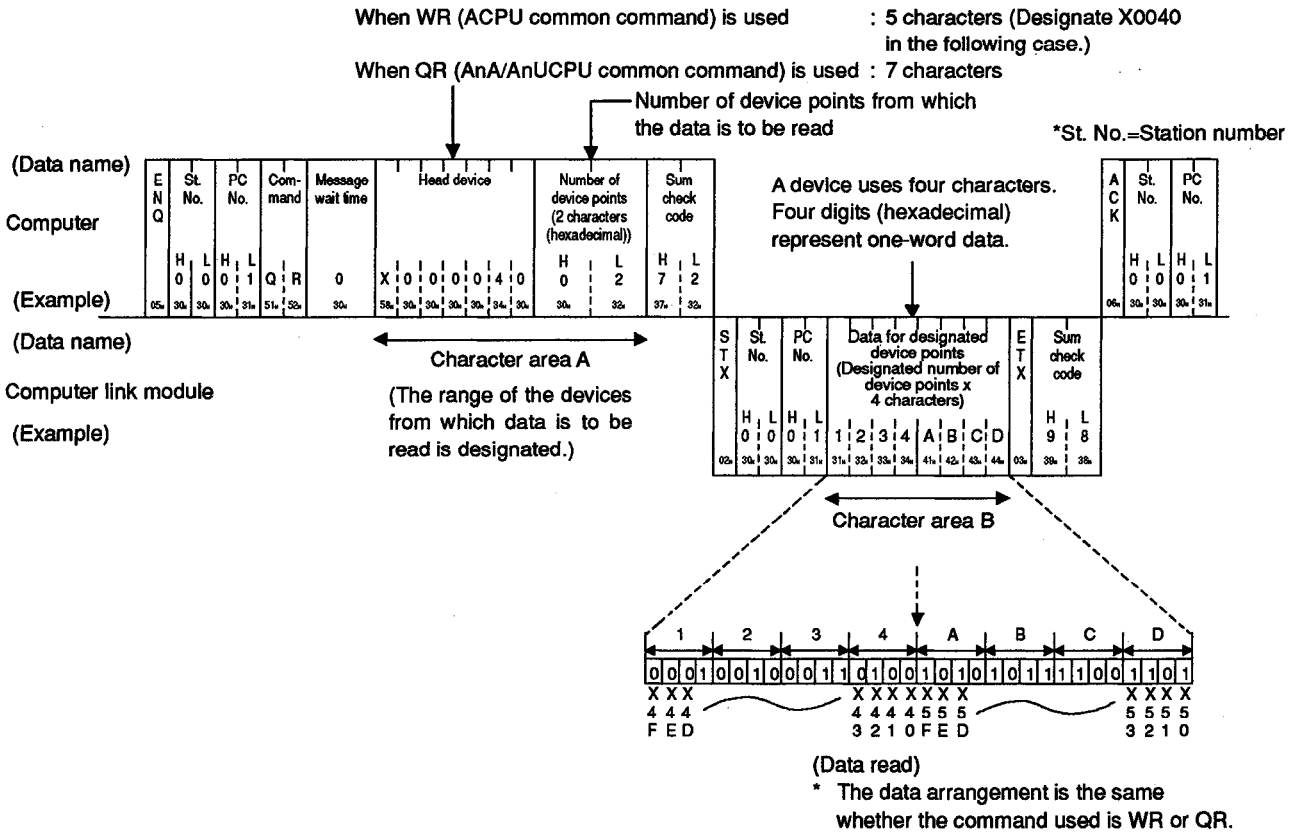
5.7.3 Batch read in units of words

In this section, examples are quoted to describe the control protocols for batch-reading data on devices from bit device memory (by 16 devices) and from word device memory using a WR or QR command.

[Control protocol] The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)
1) The message wait time is 0 ms.
2) Data is to be read from 32 devices, X040 to X05F, and the present values from two devices (both of them are two words), T123 and T124.

(1) Batch-reading data on devices from bit device memory

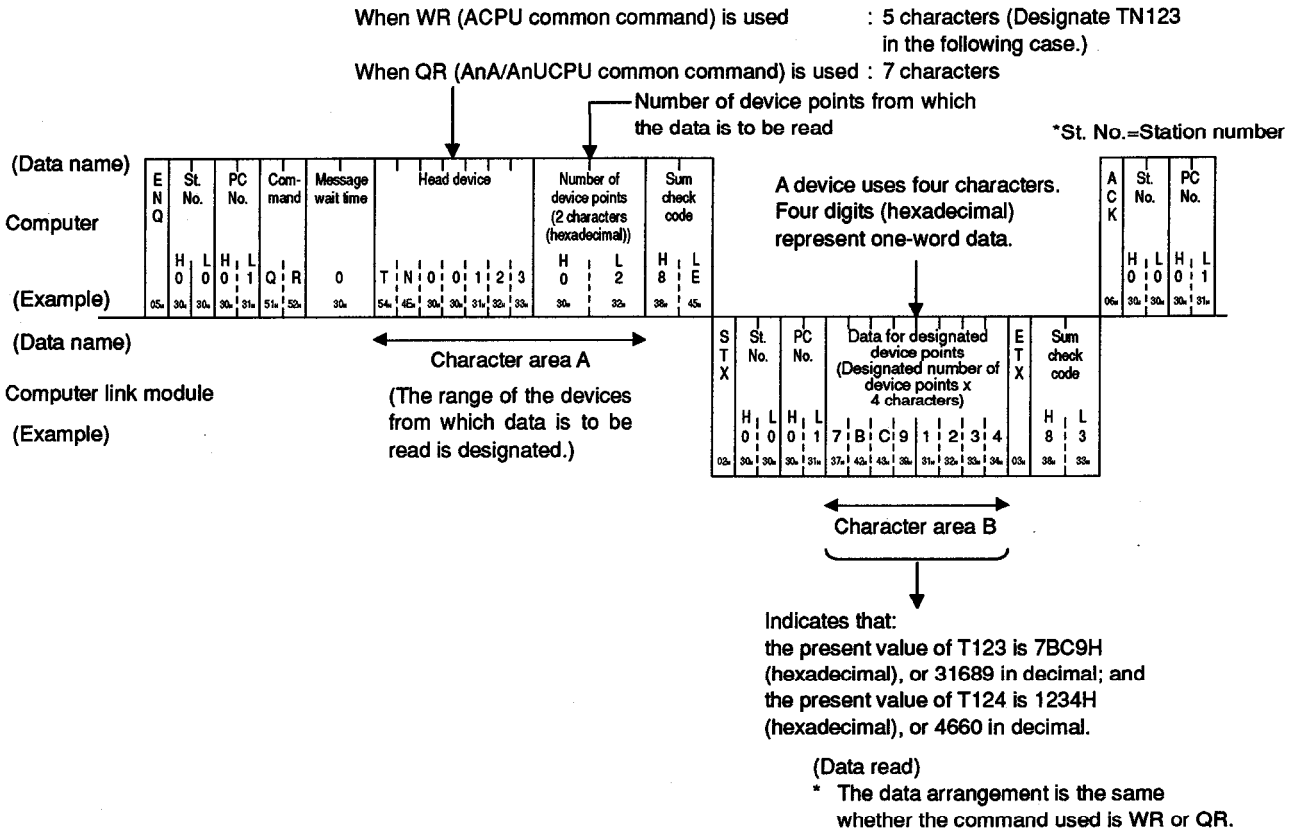


POINTS

(1) When batch-reading data from devices in bit device memory, the following conditions must be met to designate the device range:
- 1 ≤ number of device points ≤ 32
- (Head device number) + [(number of device points) x 16 - 1] ≤ maximum device number

(2) The WR and QR commands are used for word unit designation.
Designate "02" (16 points are designated as 1) for the number of device points in reading 32 points of devices from X40 to X5F.

(2) Batch-reading data from devices in word device memory



POINTS

(1) When batch-reading data from devices in word device memory, the following conditions must be met to designate the device range:
- 1 ≤ Number of device points ≤ 64
- (Head device number) + [(number of device points) - 1] ≤ maximum device number

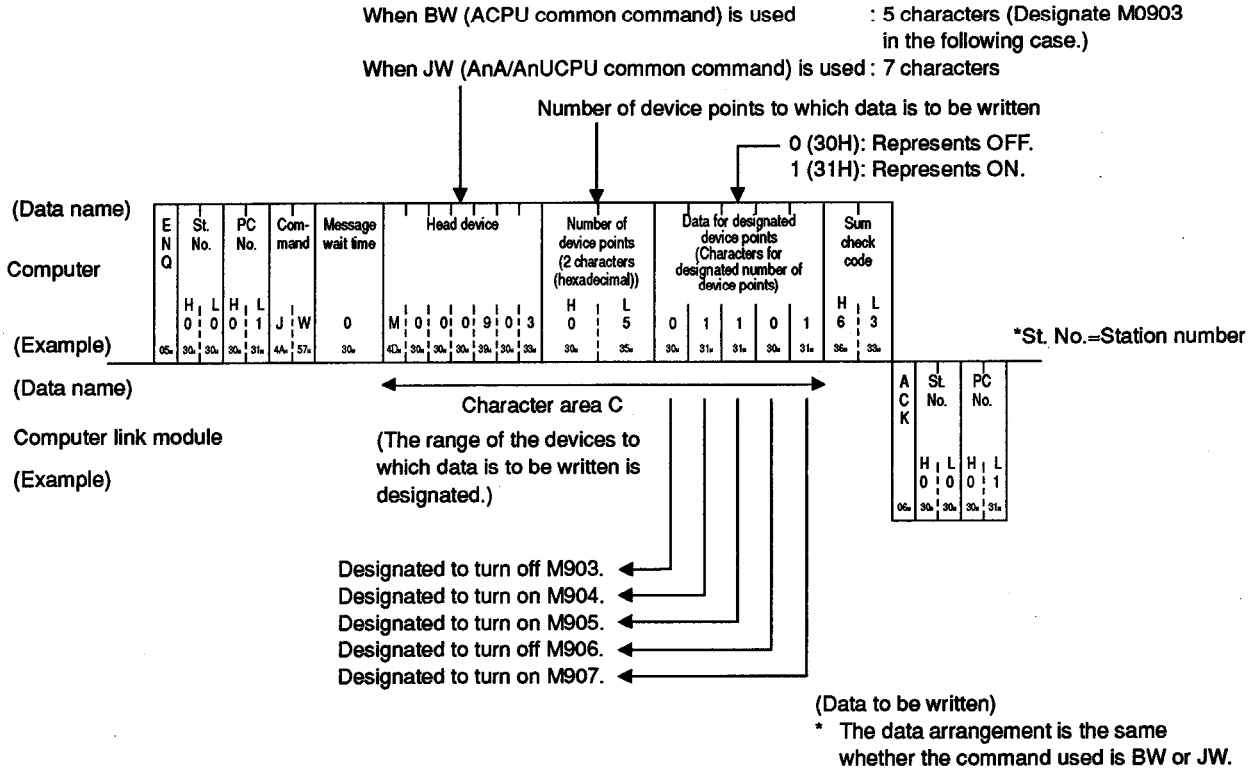
(2) The WR and QR commands are used for word unit designation. Designate "02" (a point is designated as 1) for the number of device points in reading the present values of T123 and T124.

5.7.4 Batch write in units of bits

In this section, an example is quoted to describe the control protocol for batch-writing data to devices in bit device memory using a BW or JW command.

[Control protocol] The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)
1) The message wait time is 0 ms.
2) Data is to be written to five devices, M903 to M907.



POINT

To designate the device range, the following conditions must be met:

- 1 ≤ number of device points ≤ 160
- (Head device number) + [(number of device points) - 1] ≤ maximum device number

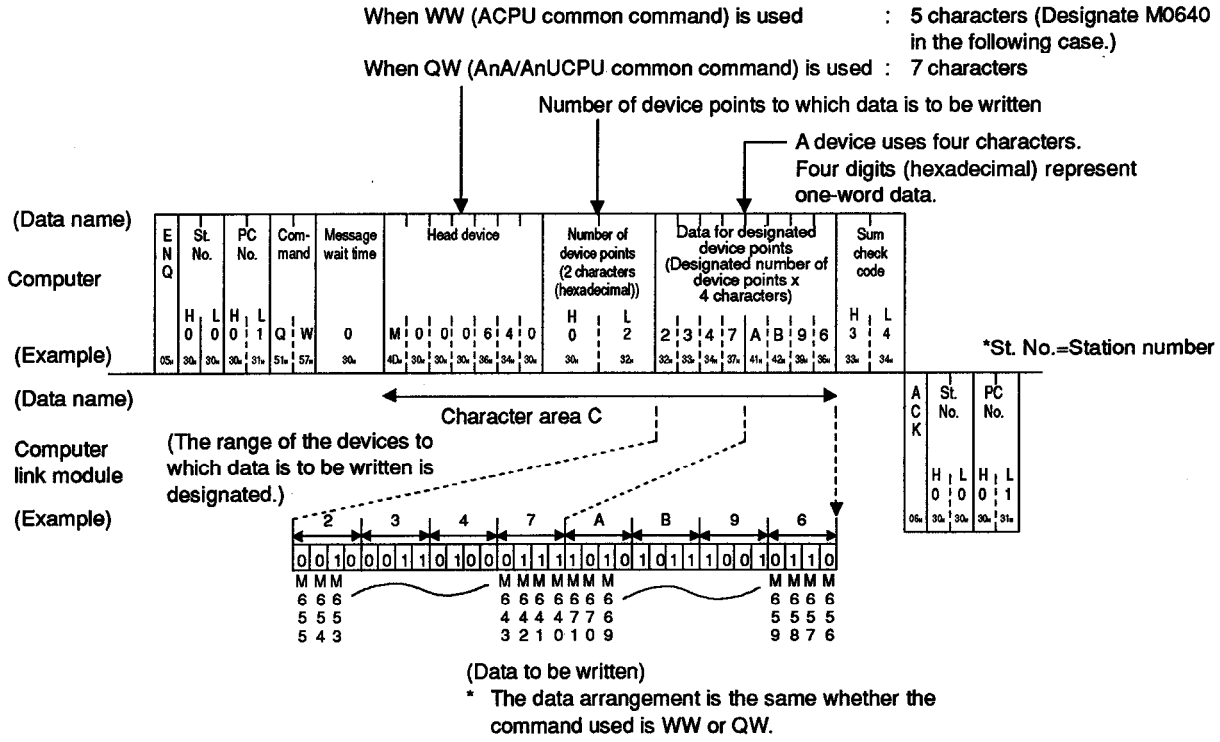
5.7.5 Batch write in units of words

In this section, examples are quoted to describe the control protocols for batch-writing data to devices in bit device memory (by 16 devices) and in word device memory using a WW or QW command.

[Control protocol] The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)
1) The message wait time is 0 ms.
2) Data is to be written to 32 devices, M640 to M671, and to two devices (two words), D0 and D1.

(1) Batch-writing data to devices in bit device memory



POINTS

(1) When batch-writing data to devices in bit device memory, the following conditions must be met to designate the device range:
- 1 ≤ number of device points ≤ 10
- (Head device number) + [(number of device points) x 16 - 1] ≤ maximum device number

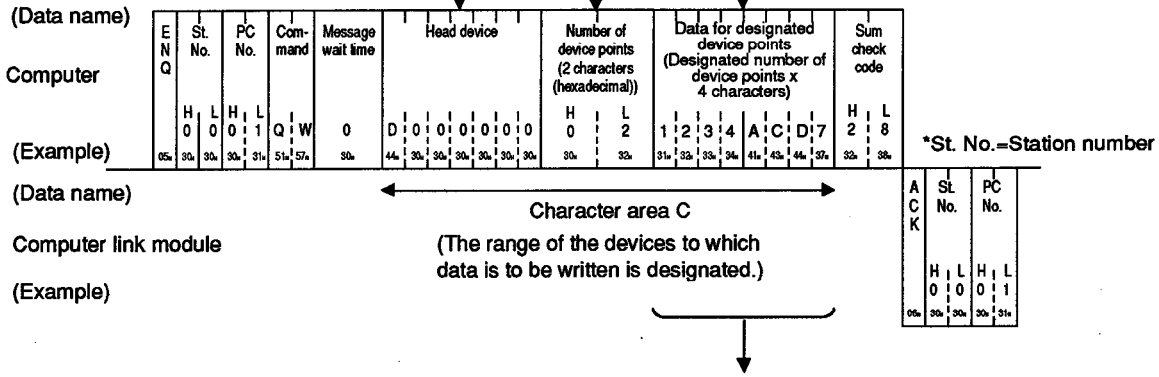
(2) The WW and QW commands are used for word unit designation.
Designate "02" (16 points are designated as 1) for the number of device points in writing 32 points of devices from M640 to M671.

(2) Batch-writing data to devices in word device memory

When WW (ACPU common command) is used : 5 characters (Designate D0000 in the following case.)
When QW (AnA/AnUCPU common command) is used : 7 characters

Number of device points to which data is to be written

A device uses four characters.
Four digits (hexadecimal) represent one-word data.



Character area C
(The range of the devices to which data is to be written is designated.)

Indicates that 1234H (hexadecimal) to D0, 4660 in decimal, is written.
Indicates that ACD7H (hexadecimal) to D1, -21289 in decimal, is written.

(Data to be written)

* The data arrangement is the same whether the command used is WW or QW.

POINTS

(1) When batch-writing data to devices in word device memory, the following conditions must be met to designate the device range:
- 1 ≤ Number of device points ≤ 64
- (Head device number) + [(number of device points) - 1] ≤ maximum device number

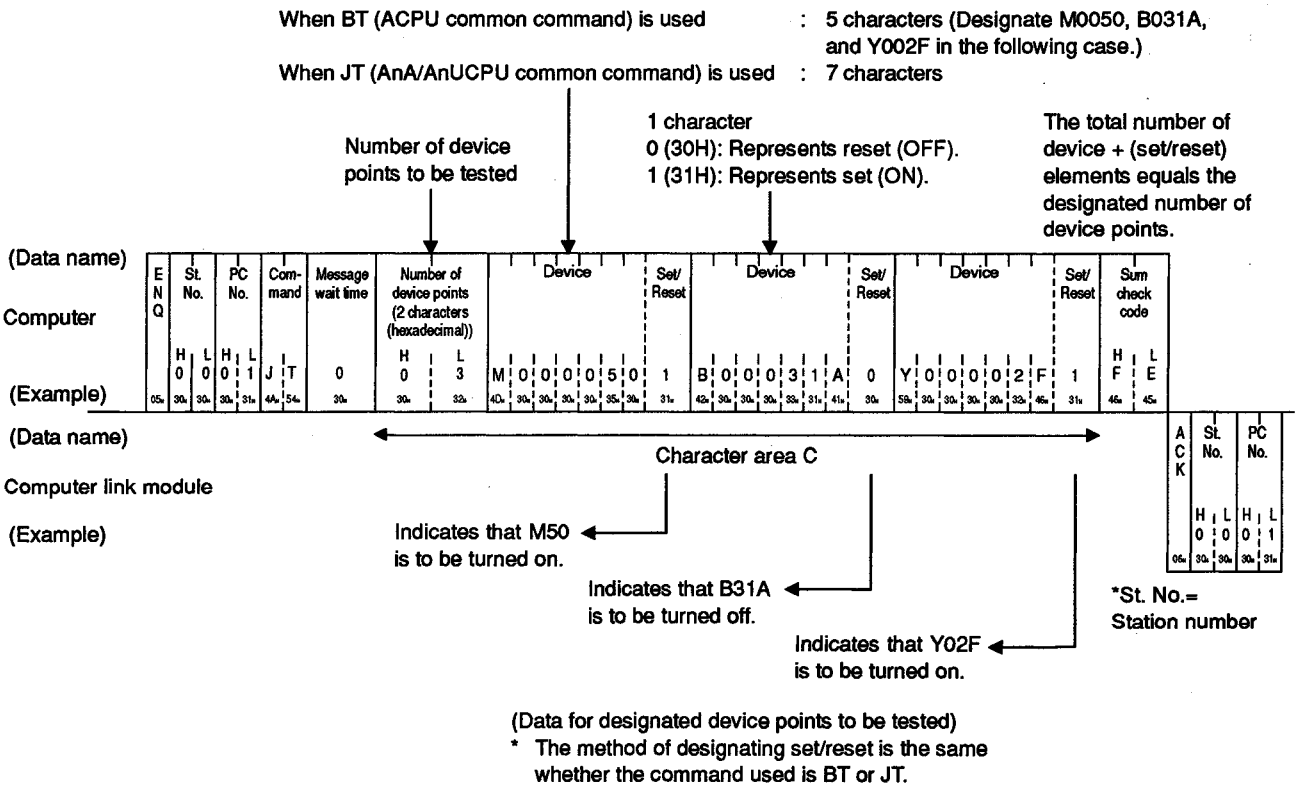
(2) The WW and QW commands are used for word unit designation. Designate "02" (a point is designated as 1) for the number of device points in writing to two devices, D0 and D1.

5.7.6 Testing device memory in units of bits (random write)

In this section, an example is quoted to describe the control protocol for writing data to randomly designated devices in bit device memory using a BT or JT command.

[Control protocol] The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)
1) The message wait time is 0 ms.
2) This test is conducted on three devices (three bits), M50, B31A, and Y02F, to turn on M50 and Y02F and to turn off B31A.



POINT

To designate the device range, the following condition must be met:
- 1 ≤ number of device points ≤ 20

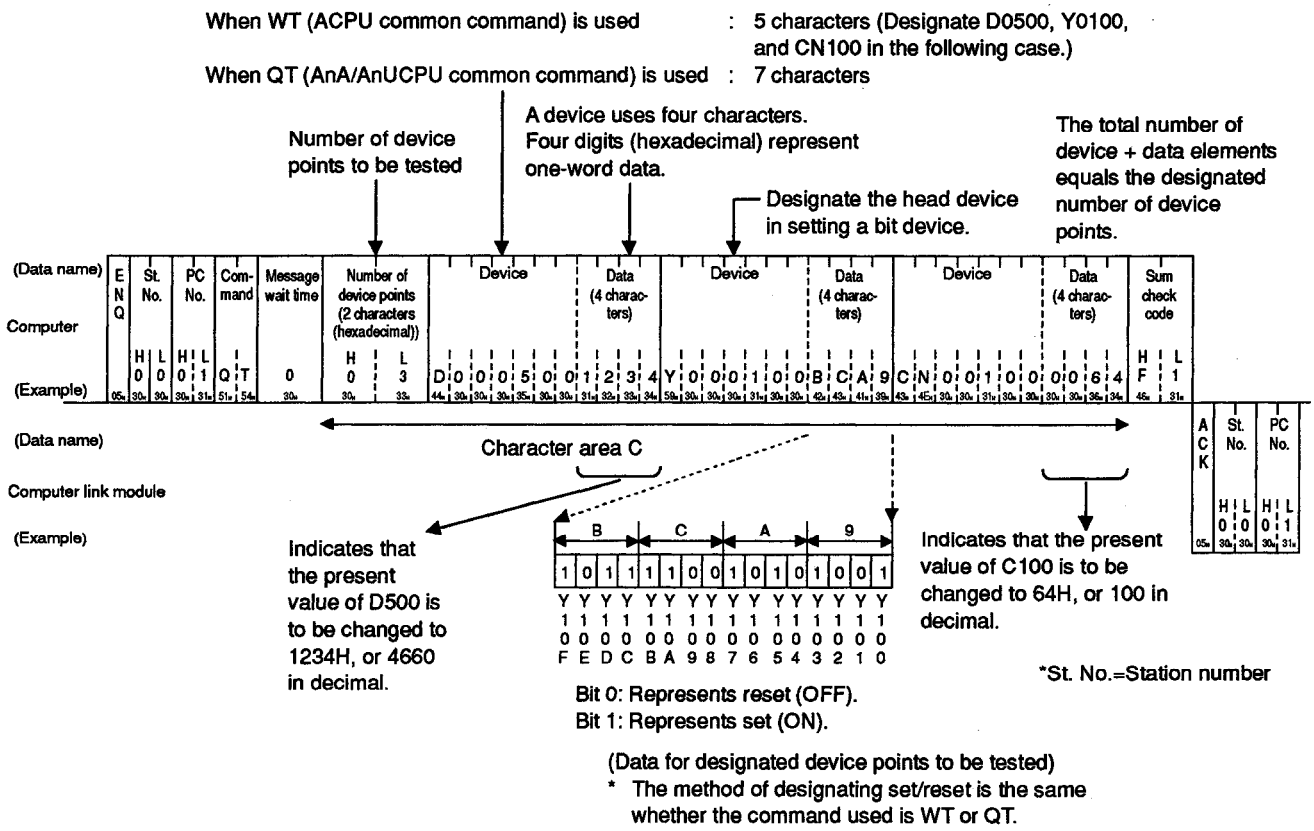
5.7.7 Testing device memory in units of words (random write)

In this section, an example is quoted to describe the control protocol for writing data to randomly designated devices in word device memory and in bit device memory (by 16 devices) using a WT or QT command.
Word devices and bit devices can be set together (by 16 devices).

[Control protocol] The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)

1) The message wait time is 0 ms.
2) This test is conducted to change the present values of three devices (three words), D500, Y100 (Y100 to Y10F), and C100, to 1234H, BCA9H, and 64H respectively.



POINT

To designate the device range, the following condition must be met:

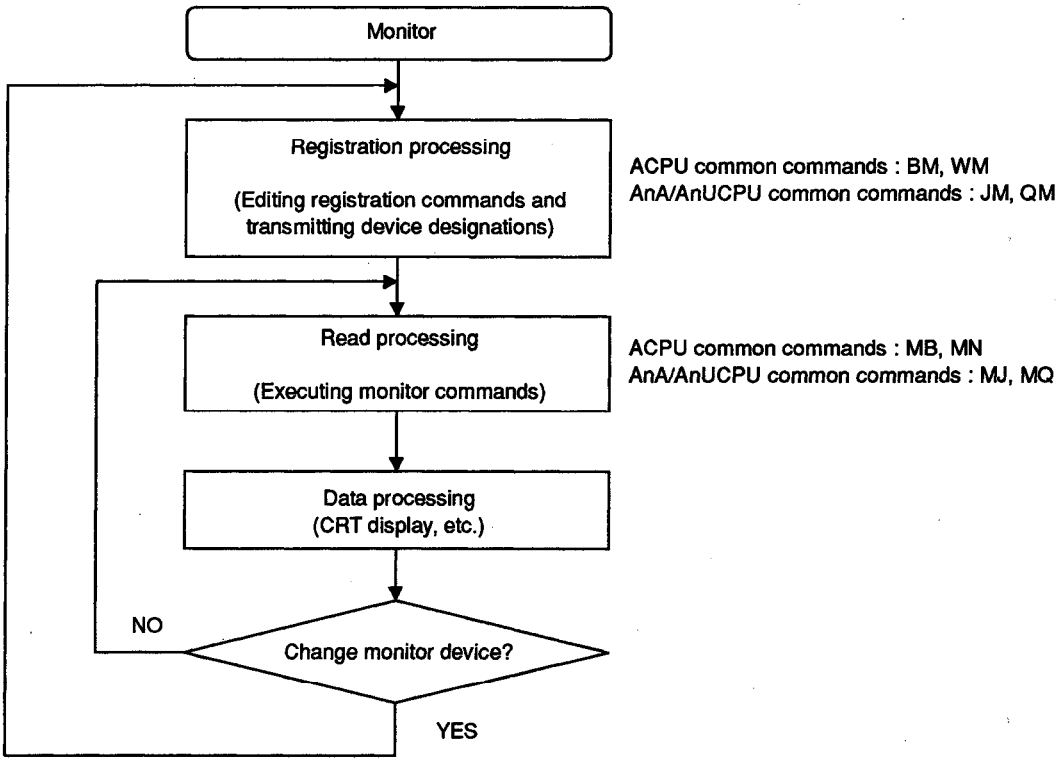
- 1 ≤ number of device points ≤ 10 (10 units for bit devices where 1 unit corresponds to 16 points)

5.7.8 Monitoring device memory

Monitor data registration is the function that registers the name and the number of the device to be monitored by the computer to the computer link module. The monitor is the function that (a) reads the data content of the device registered at the time the monitor read command is executed by the computer, and (b) executes the corresponding processing such as monitoring.

The device numbers must be consecutive when the device is read using the batch read (BR, WR/JR, QR) command. However, when this function is used, it is possible to read and monitor the devices by designating the device numbers at random.
In this section, examples are quoted to describe the control protocols for monitoring and registering devices to be monitored and their numbers in the computer link module.

(1) Control procedure for monitoring



POINTS

(1) As the flowchart shows, monitor data registration must be executed before monitoring. Attempting to execute monitoring without registering the monitor data will cause a protocol error.
(2) The contents of the monitor data registration area are cleared when the computer link module is reset.
(3) For monitor registration, five types of registration are possible. They are device memory in bit units (BM or JM), device memory in word units (WM or QM), and the extension file register (EM).
(4) For monitoring extension file registers, refer to Section 5.8.9.

(2) Registering devices for data monitoring in device memory

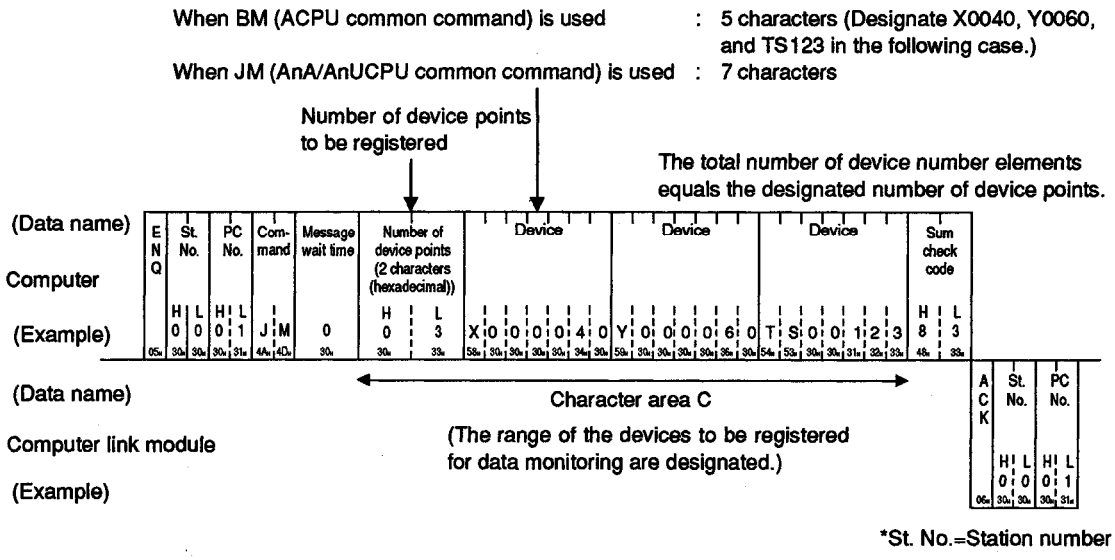
In this section, examples are quoted to describe the control protocols for registering devices to be monitored and their numbers in the computer link module using a BM, JM, WM or QM command.

(a) Registering devices for data monitoring in bit device memory in units of bits

[Control protocol]

The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Monitor data registration conditions)
1) The message wait time is 0 ms.
2) Three devices (three bits), X040, Y060, and T123 (contact), are to be registered for data monitoring.



*St. No.=Station number

POINTS

(1) To designate the device range, the following conditions must be met.
When the BM command is used and the PC CPU to be accessed is other than the A3HCPU, AnACPU or AnUCPU, a device X (input) is treated as two points.
- $1 \leq$ number of device points ≤ 40

(2) To monitor devices registered in device memory in units of bits, follow the procedure described in (3)(a) of this section.

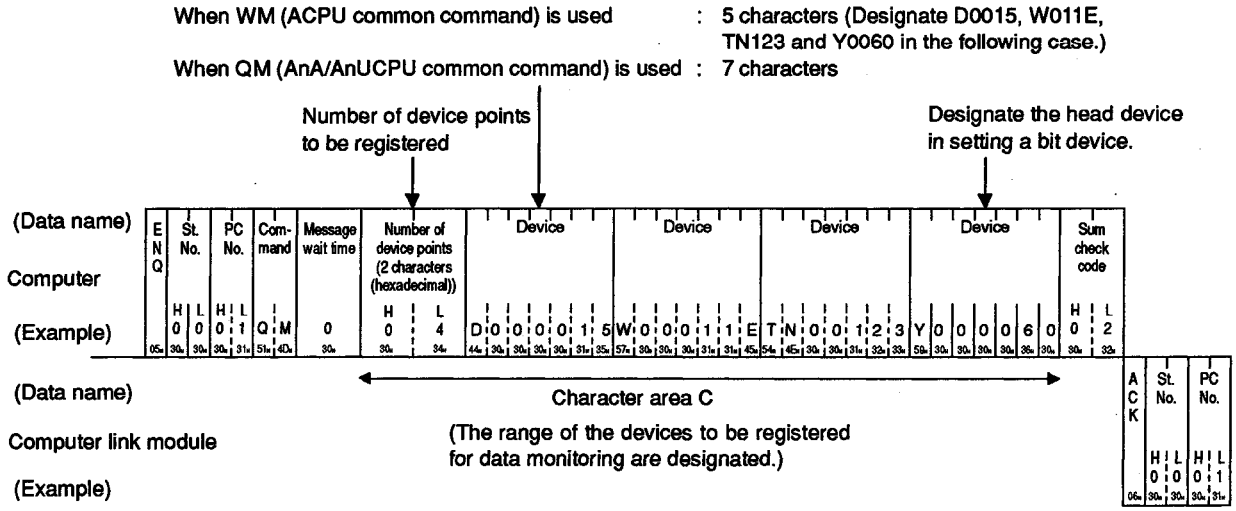
(b) Registering devices for data monitoring in bit device memory and word device memory in units of words

[Control protocol]

The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Monitoring data registration conditions)

1) The message wait time is 0 ms.
2) The present values of devices D15, W11E, and T123 and data on devices Y060 to Y06F (four words) are to be registered for monitoring.



*St. No.=Station number

POINTS

(1) To designate the device range, the following conditions must be met.
When the WM command is used and the PC CPU to be accessed is other than the A3HCPU, AnACPU or AnUCPU, a device X (input) is treated as two points.
- $1 \leq$ number of device points ≤ 20

(2) As shown in the above figure, word devices and bit devices can be set together (by 16 devices) in registering devices for data monitoring in units of words.

(3) To monitor devices registered in device memory in units of words, follow the procedure described in (3)(b) of this section.

(3) Monitoring devices registered in device memory for data monitoring

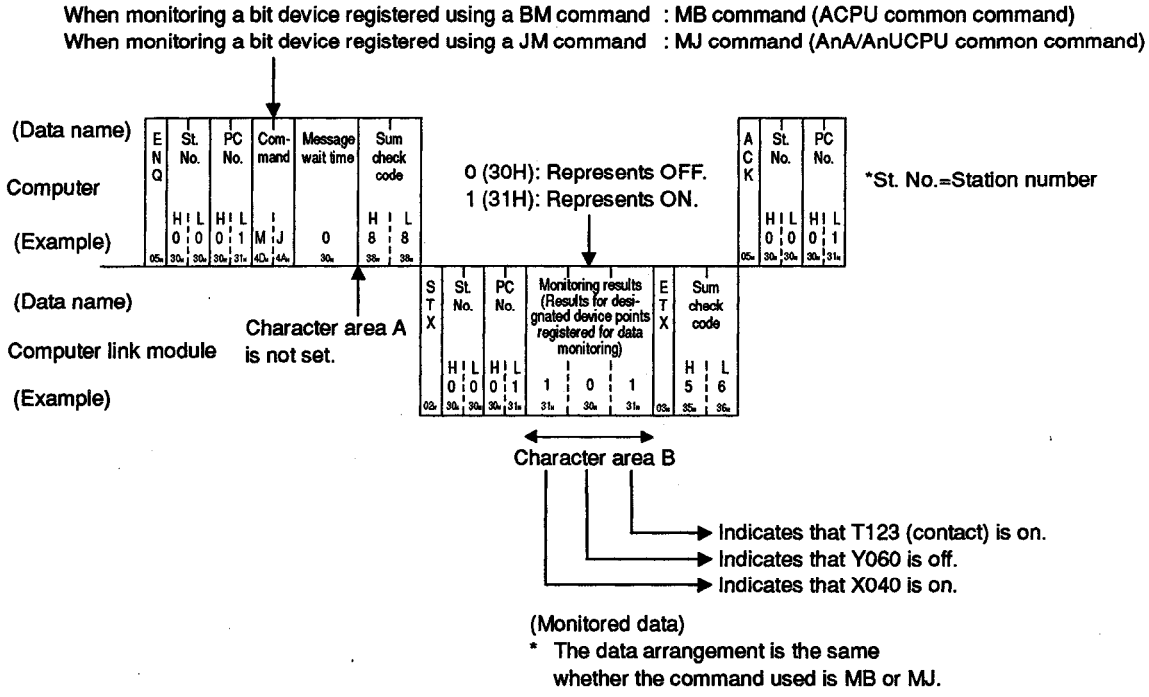
In this section, examples are quoted to describe the control protocols for monitoring devices registered with the computer link module, which are shown in (2), using an MB, MJ, MN or MQ command.

(a) Monitoring devices registered for data monitoring in bit device memory in units of bits

[Control protocol]

The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.
Devices registered in bit device memory using a BM command can be monitored with an MB command, while those registered using a JM command can be monitored with an MJ command.

(Monitoring conditions)
1) The message wait time is 0 ms.
2) Three devices (three bits) registered for data monitoring, X040, Y060, and T123 (contact), are to be monitored.



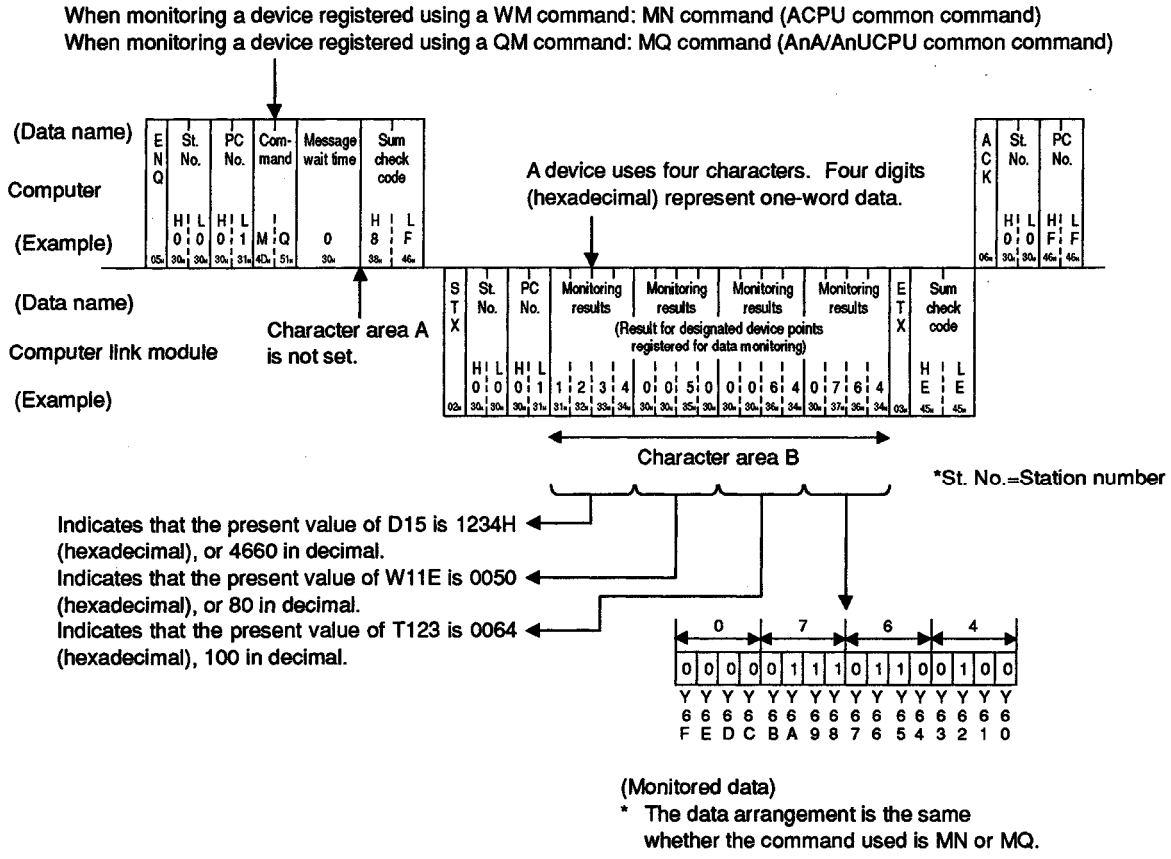
(b) Monitoring devices registered for data monitoring in bit device memory and word device memory in units of words

[Control protocol]

The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.
Devices registered in device memory using a WM command can be monitored with a MN command, while those registered using a QM command can be monitored with a MQ command.

(Monitoring conditions)

1) The message wait time is 0 ms.
2) The present values of D15, W11E, and T123 registered for data monitoring, and data on devices Y060 to Y06F (four words) are to be monitored.



5.8 Extension File Register Read and Write

A free area in the user memory area in the PC CPU and used as a file register is called extension file register. This memory area can store necessary data and the results of data processing and other operations executed by the "SW0GHP-UTLPC-FN1" (hereinafter referred to as the UTLP-FN1) or "SW0SRX-FNUP" (hereinafter referred to as the FNUP), both of which are software packages for extension file registers, or by dedicated instructions for extension file registers in the AnACPU and the AnUCPU.
In this section, examples are quoted to describe the control protocols for reading or writing data from or to extension file registers.

5.8.1 ACPU common commands and addresses

(1) ACPU common commands used for read/write of extension file registers

Item	Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
	Symbol	ASCII Code			During STOP	During RUN		
						Write Enabled	Write Disabled	
Batch read	ER	45H, 52H	Reads from extension file registers (R) in units of 1 point.	64 points	O	O	O	5.8.4
Batch write	EW	45H, 57H	Writes to extension file registers (R) in units of 1 point.	64 points	O	O	X	5.8.5
Test (random write)	ET	45H, 54H	Specifies the extension file registers (R) in units of 1 point using block or device number and makes a random write.	10 points	O	O	X	5.8.8
Monitor data registration	EM	45H, 4DH	Sets the device numbers to be monitored in units of 1 point.	20 points	O	O	O	5.8.9 (2)
Monitor	ME	4DH, 45H	Monitors the extension file registers after monitor data registration.	—	O	O	O	5.8.9 (3)

Note : O .....Executable
X.....Not executable

(2) Extension file register addresses

(a) The extension file register comprises blocks number 0 to "n", with "n" varying according to the memory cassette. Block number "0" contains the number of points designated by the PC CPU parameters and each block with numbers "1" to "n" has 8192 points of registers.

Read/write is possible in the range of parameters designated in block number 0.

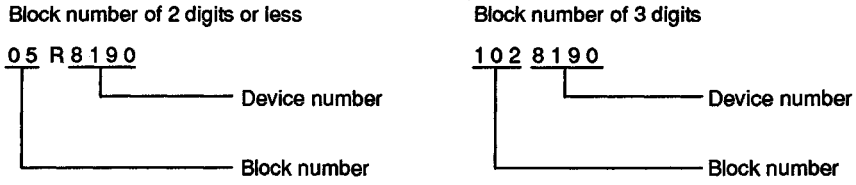
(b) The range of block numbers which can be designated varies according to the type of memory cassette and the PC CPU parameter setting.

The UTLP-FN1 or FNUP Operating Manual, or AnACPU or AnUCPU User's Manual give details.

(c) Each address is designated in 7 characters consisting of the block and device numbers.

- Block number of 2 digits or less:
  "Block number (2 digits)" + "R" + "Device number (4 digits)"
- Block number of 3 digits:
  "Block number (3 digits)" + "Device number (4 digits)"

Example:



5.8.2 AnA/AnUCPU common commands and device numbers

(1) The AnA/AnUCPU common commands used for direct read and direct write of extension file registers are described below.

These dedicated commands are used to access the extension file register of block numbers 1 to 256 by directly designating the address, which begins with address 0 in block number 1, as the device number.
(The address numbers used to access the extension file register go from 0 to "the usable number of blocks x 8192 points".)

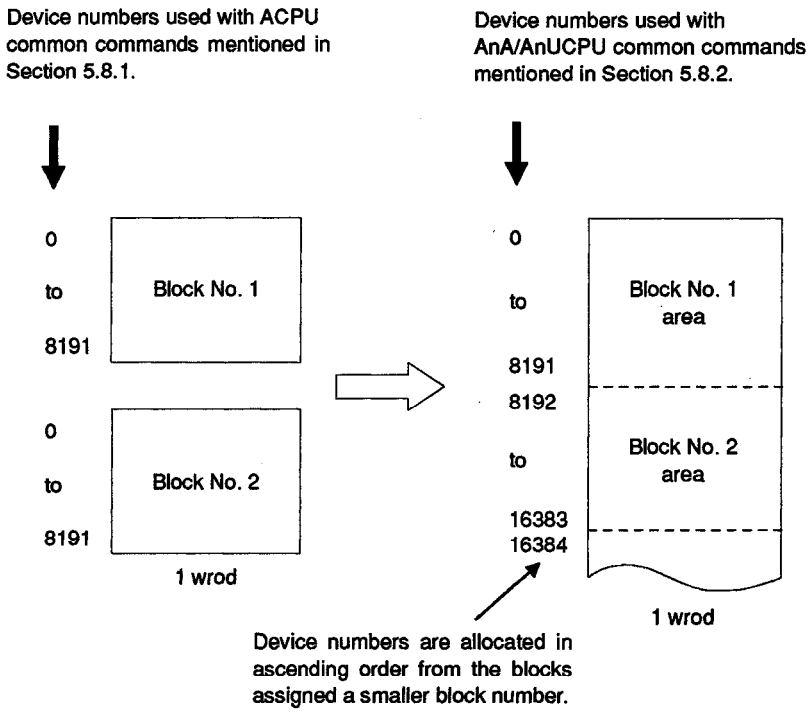
Item	Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
	Symbol	ASCII Code			During STOP	During RUN		
						Write Enabled	Write Disabled	
Direct read	NR	4EH, 52H	Reads data from the extension file register in units of 1 point (words) by designating the extension file register in successive numbers.	64 points	O	O	O	5.8.6
Direct write	NW	4EH, 57H	Writes data to the extension file register in units of 1 point (words) by designating the extension file register in successive numbers.	64 points	O	O	X	5.8.7

Note : O .....Executable
X.....Not executable

(2) Device numbers of extension file registers

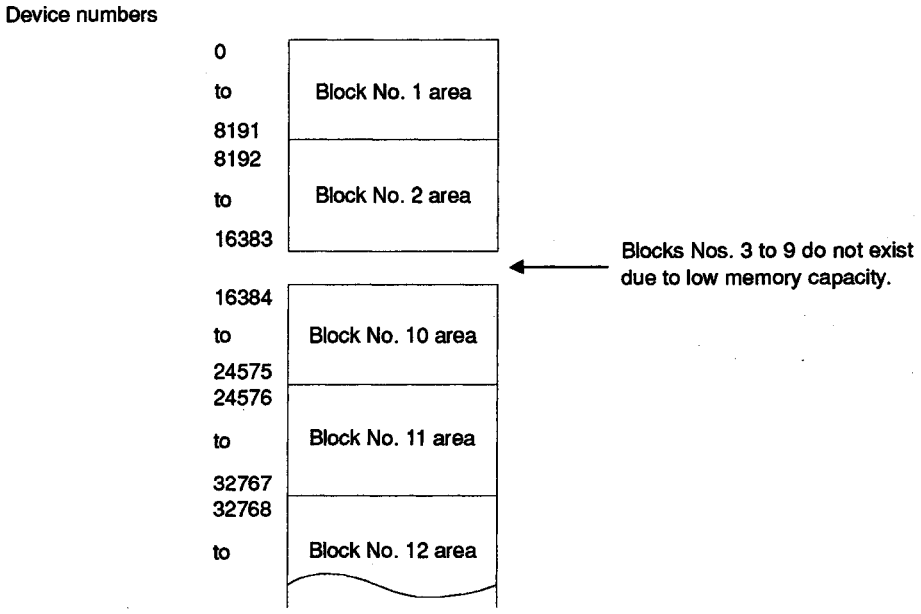
(a) Device number range

Range: 0 through [(the number of usable blocks x 8192) - 1]



The device numbers that can be designated vary according to the type of memory cassette and the PC CPU parameter setting. (The UTLP-FN1 or FNUP Operating Manual or the AnACPU or AnUCPU User's Manual give details.)

For block numbers that do not exist in the memory cassette, device numbers are not allocated. In this case, the device numbers are allocated as indicated below, skipping non-existent block numbers.



(b) A device number is designated in 7 characters.

Designation example 1:
To designate R10 in block number 1:

0000010 ← A blank code (20H) can be used to express leading zeros (the underlined 0s in 0008200).

Designation example 2:
To designate R8 in block number 2:

0008200

$(2 - 1) \times \underline{8192} + 8$

Block number — Block points

POINTS

(1) The AnA/AnUCPU common commands NR and NW can only be used for read/write operations at the extension file registers of block numbers 1 to 256.
They can be used regardless of the parameter's file register setting.

(2) Use the commands described in Section 5.8.1 to access the parameter set file registers (R) or to access a file register by designating a block number.

(3) The following equation is used to calculate the head device number to be designated with the AnA/AnUCPU common commands NR and NW. (To designate device number "m" (0 to 8191) in the "n"th block ($n \geq 1$))
Head device number = (n–1) x 8192 + m

REMARK

The range of device numbers (up to the 28th block) that can be designated with the NR or NW commands is shown below.

Device No.	Objective Block		Device No.	Objective Block	
0 to 8191	1st block	R0 to R8191	114688 to 122879	15th block	R0 to R8191
8192 to 16383	2nd block	R0 to R8191	122880 to 131071	16th block	R0 to R8191
16384 to 24575	3rd block	R0 to R8191	131072 to 139263	17th block	R0 to R8191
24576 to 32767	4th block	R0 to R8191	139264 to 147455	18th block	R0 to R8191
32768 to 40959	5th block	R0 to R8191	147456 to 155647	19th block	R0 to R8191
40960 to 49151	6th block	R0 to R8191	155648 to 163839	20th block	R0 to R8191
49152 to 57343	7th block	R0 to R8191	163840 to 172031	21st block	R0 to R8191
57344 to 65535	8th block	R0 to R8191	172032 to 180223	22nd block	R0 to R8191
65536 to 73727	9th block	R0 to R8191	180224 to 188415	23rd block	R0 to R8191
73728 to 81919	10th block	R0 to R8191	188416 to 196607	24th block	R0 to R8191
81920 to 90111	11th block	R0 to R8191	196608 to 204799	25th block	R0 to R8191
90112 to 98303	12th block	R0 to R8191	204800 to 212991	26th block	R0 to R8191
98304 to 106495	13th block	R0 to R8191	212992 to 221183	27th block	R0 to R8191
106496 to 114687	14th block	R0 to R8191	221184 to 229375	28th block	R0 to R8191

5.8.3 Precautions during extension file register read/write

Described below are precautions for reading or writing data from or to extension file registers using the commands shown in Section 5.8.4 through Section 5.8.9.

(1) The extension file register is not used by A1 and A1NCPU.

This function is not available during communications with A1 or A1NCPU.

(2) Some types of memory cassette loaded to the PC CPU are unable to detect an error (character area error 06H) if an attempt is made to read or write after specifying a block number which does not exist. In this case, data which is read may not be correct and writing such incorrect data may destroy the PC CPU user memory.

Always check the type of memory cassette and the parameter settings before using this function.

Type of Memory Cassette	Block Numbers Which do not Cause a Character Area Error (06H)		
	A0J2H, A2, A3CPU	A2N, A3NCPU	A3H, AnA, AnUCPU
A3NMCA-12	No. 10, No. 11		
A3NMCA-18	—	No. 10 to No. 28	
A3NMCA-24	—	No. 13 to No. 20	No. 13 to No. 28
A3NMCA-40	—		No. 21 to No. 28
A3AMCA-96	—		No. 21 to No. 48 (*1)

*1 The A3AMCA-96 can be used for the A3A, the A3U, and the A4UCPU.

(The UTLP-FN1 or FNUP Operating Manual or the AnACPU or AnUCPU User's Manual give details.)

(3) The A2USCPU(-S1) can deal with the following extension file register block numbers:

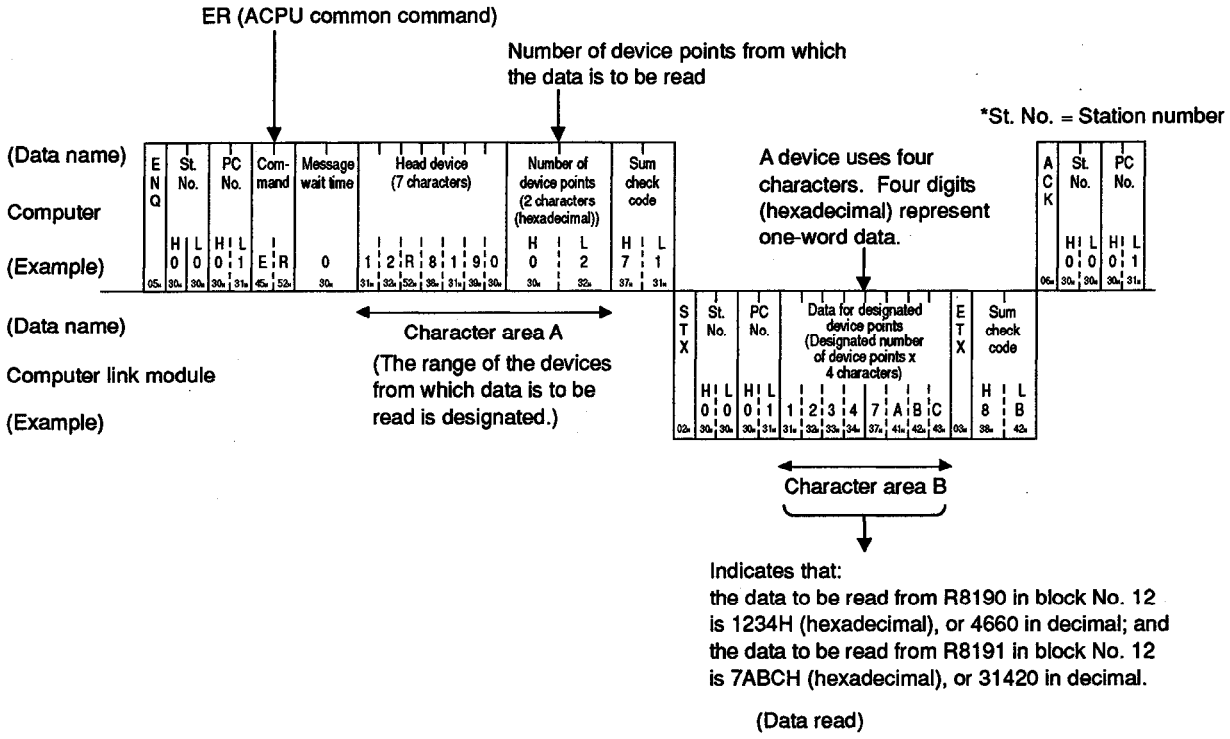
- A2USCPU ..... No. 1 to No. 3
- A2USCPU-S1 ..... No. 1 to No. 8, No. 10 to No. 16

5.8.4 Batch read of the extension file register

In this section, an example is quoted to describe the control protocol for batch-reading data from extension file registers using an ER command.

[Control protocol] The protocol shown below is in control format 1.
To access an extension file register in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)
1) The message wait time is 0 ms.
2) Data is to be read from two registers, R8190 and R8191, in block No. 12.



POINT

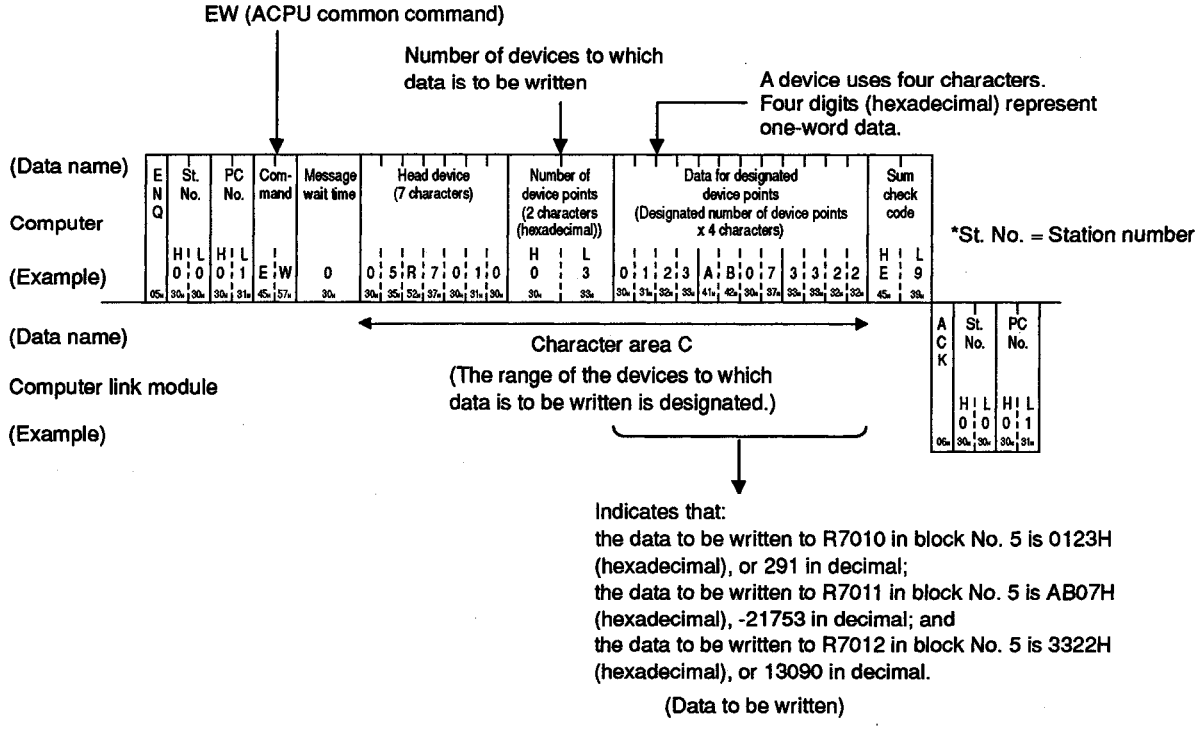
To designate the device range, the following conditions must be met:
- 1 ≤ number of device points ≤ 64
- (Head device number) + [(number of device points) - 1] ≤ maximum device number

5.8.5 Batch write of the extension file register

In this section, an example is quoted to describe the control protocol for batch-writing data to extension file registers using an EW command.

[Control protocol] The protocol shown below is in control format 1.
To access an extension file register in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)
1) The message wait time is 0 ms.
2) Data is to be written to three extension file registers, R7010 to R7012, in block No. 5.



POINT

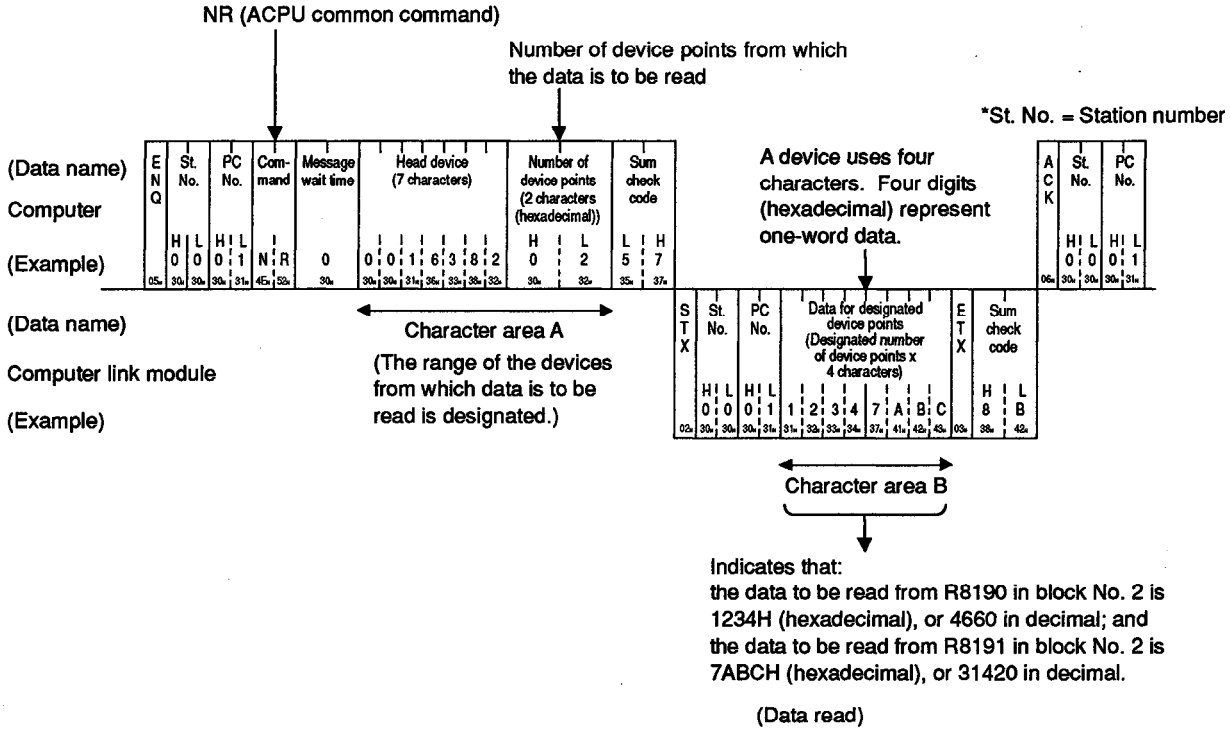
To designate the device range, the following conditions must be met:
- 1 ≤ number of device points ≤ 64
- (Head device number) + [(number of device points) - 1] ≤ maximum device number

5.8.6 Direct read of the extension file register

In this section, an example is quoted to describe the control protocol for reading data directly from extension file registers using an NR command.

[Control protocol] The protocol shown below is in control format 1.
To access an extension file register in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)
1) The message wait time is 0 ms.
2) Data is to be read from two extension file registers, R8190 and R8191, in block No. 2.



POINT

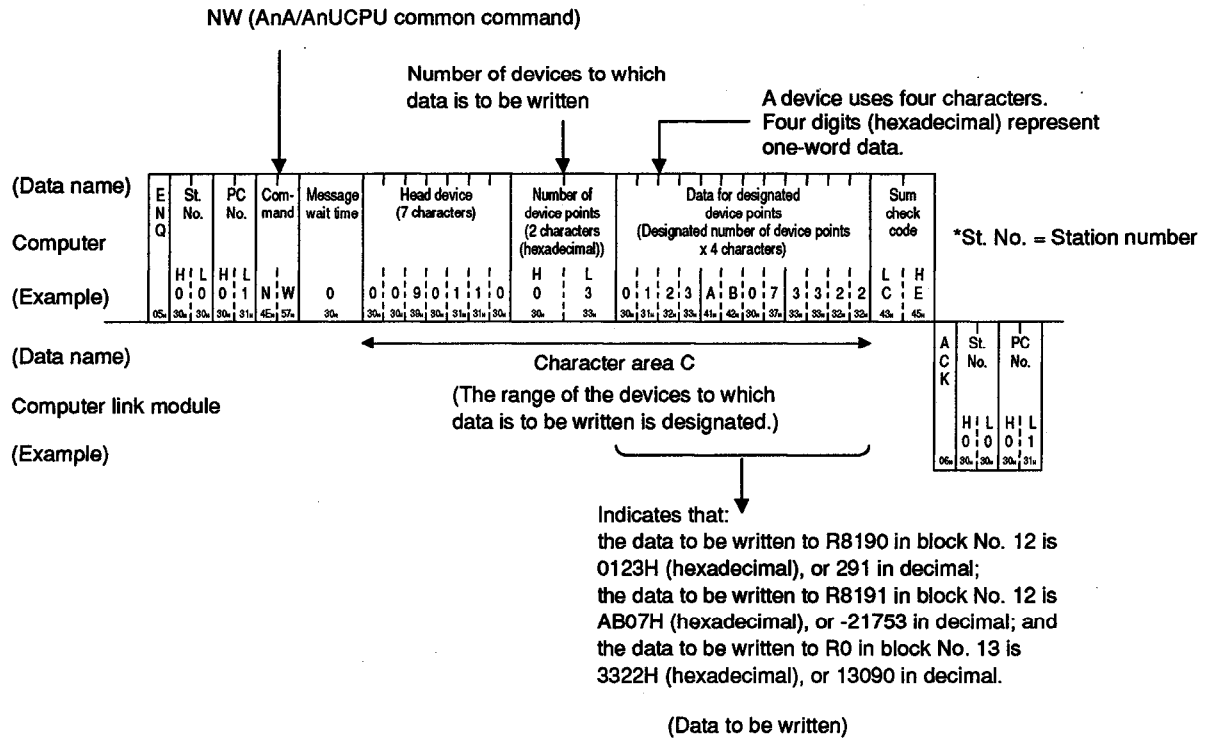
To designate the device range, the following conditions must be met:
- 1 ≤ number of device points ≤ 64
- (Head device number) + [(number of device points) - 1] ≤ maximum device number

5.8.7 Direct write of the extension file register

In this section, an example is quoted to describe the control protocol for writing data directly to extension file registers using an NW command.

[Control protocol] The protocol shown below is in control format 1.
To access extension file registers in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)
1) The message wait time is 0 ms.
2) Data is to be written to two extension file registers in block No. 12, R8190 and R8191, and to R0 in block No. 13 (when extension file registers are reserved for blocks No. 1 to No. 8 and for blocks No. 10 to No. 13).



POINT

To designate the device range, the following conditions must be met:
- 1 ≤ number of device points ≤ 64
- (Head device number) + [(number of device points) - 1] ≤ maximum device number

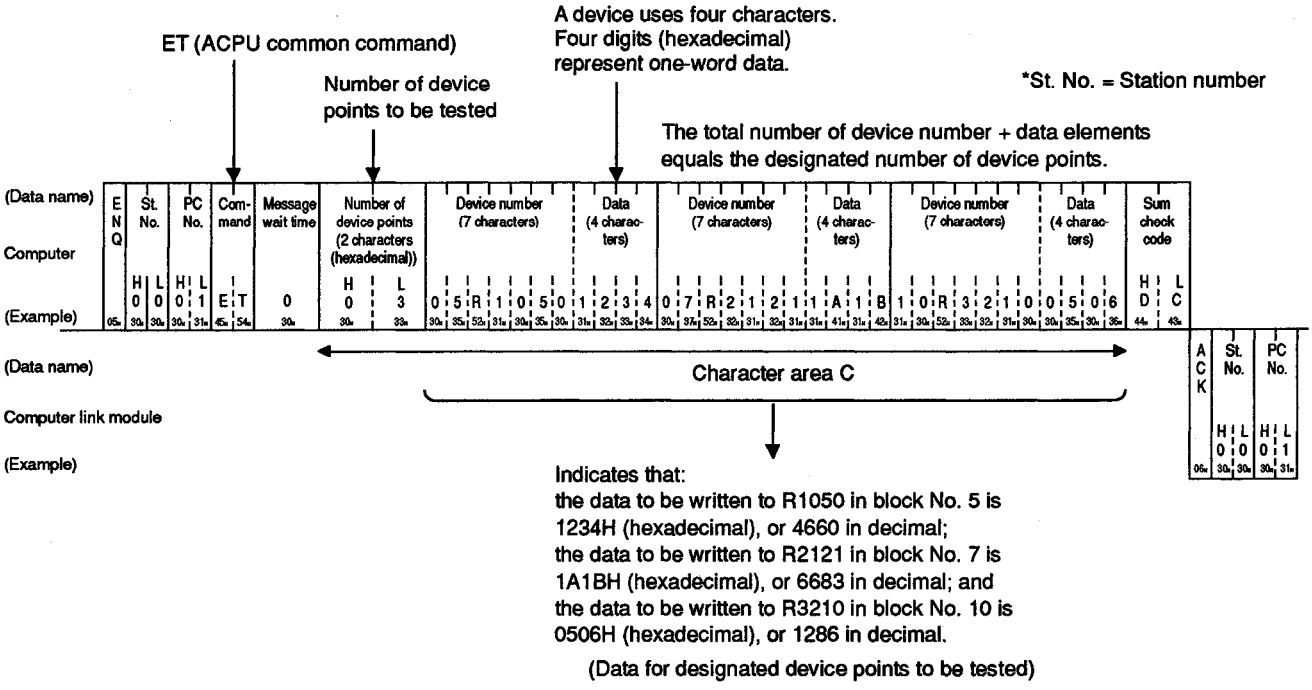
5.8.8 Testing (random write) the extension file register

In this section, an example is quoted to describe the control protocol for writing data to randomly designated device numbers in extension file registers using an ET command.

[Control protocol] The protocol shown below is in control format 1.
To access an extension file register in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)

1) The message wait time is 0 ms.
2) This test is conducted on three extension file registers (three words), R1050 in block No. 5, R2121 in block No. 7, and R3210 in block No. 10, to write 1234H to R1050, 1A1BH to R2121, and 506H to R3210 (when extension file registers are reserved for blocks No. 1 to No. 8 and for block No 10).



POINT

To designate the device range, the following condition must be met:

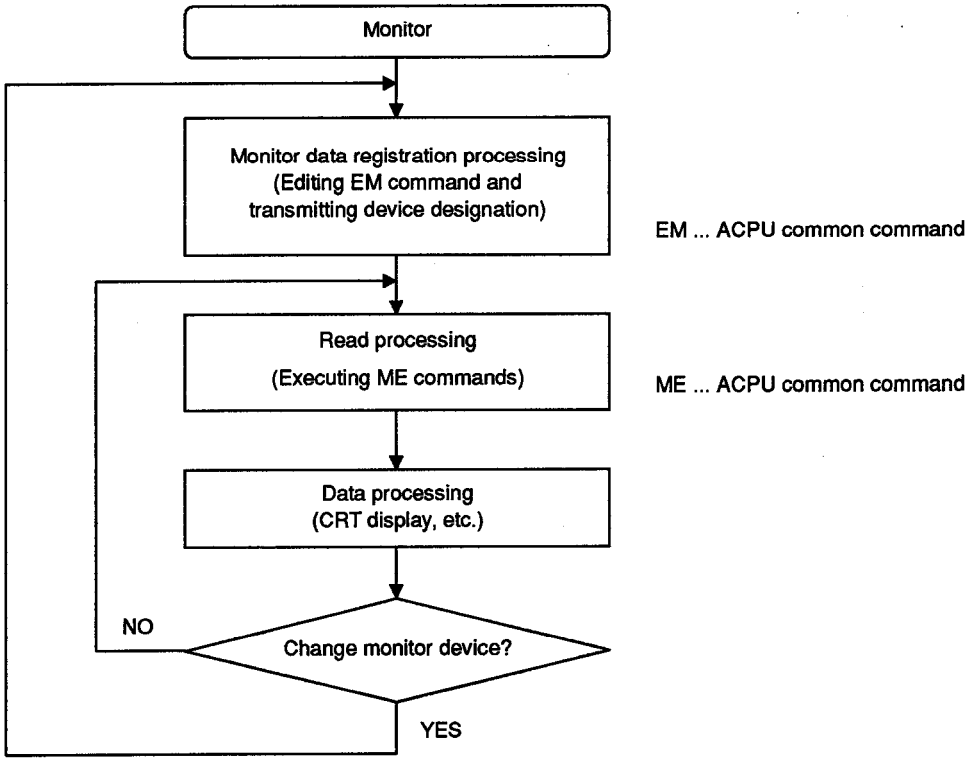
- 1 ≤ number of device points ≤ 10

5.8.9 Monitoring the extension file register

Monitor data registration is the function that registers the name and the number of the device to be monitored by the computer to the computer link module. The monitor is the function that (a) reads the data content of the device registered at the time the monitor read command is executed by the computer, and (b) executes the corresponding processing such as monitoring.

The device numbers must be consecutive when the device is read using the batch read (ER) or direct read (NR) command. However, when this function is used, it is possible to read and monitor the devices by designating the device numbers at random.
In this section, examples are quated to describe the control protocols for monitoring and registering devices to be monitored and their numbers in the computer link module.

(1) Control procedure for monitoring



POINTS

(1) As the flowchart shows, monitor data registration must be executed before monitoring. Attempting to execute monitoring without registering the monitor data will cause a protocol error.
(2) The contents of the monitor data registration area are cleared when the computer link module is reset.
(3) For monitor registration, five types of registration are possible. They are device memory in bit units (BM or JM), device memory in word units (WM or QM) and the extension file register (EM).
(4) For monitoring of device memory, refer to Section 5.7.8.

(2) Registering extension file registers for data monitoring

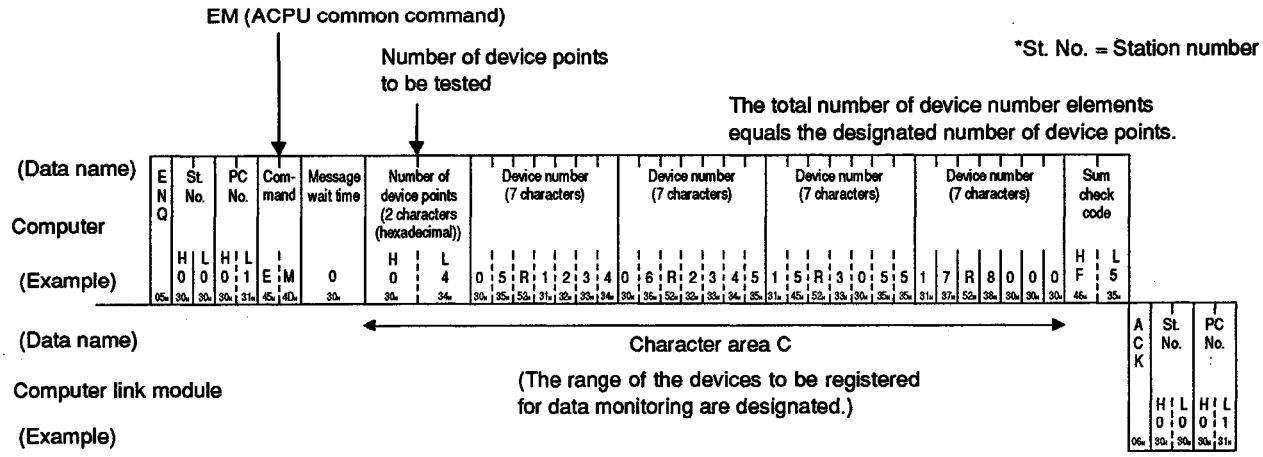
In this section, an example is quoted to describe the control protocol for registering device numbers in extension file registers to be monitored in the computer link module using an EM command.

[Control protocol]

The protocol shown below is in control format 1.
To access an extension file register in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Monitor data registration conditions)

1) The message wait time is 0 ms.
2) Four extension file registers (four words), R1234 in block No. 5, R2345 in block No. 6, R3055 in block No. 15, and R8000 in block No. 17 are to be registered for data monitoring (when extension file registers are reserved for blocks No. 1 to No. 8 and for blocks No. 10 to No. 17).



POINTS

(1) To designate the device range, the following condition must be met:
- 1 ≤ number of device points ≤ 20

(2) To monitor extension file registers registered using an EM command, follow the procedure described in (3) of this section.

(3) Monitoring extension file registers

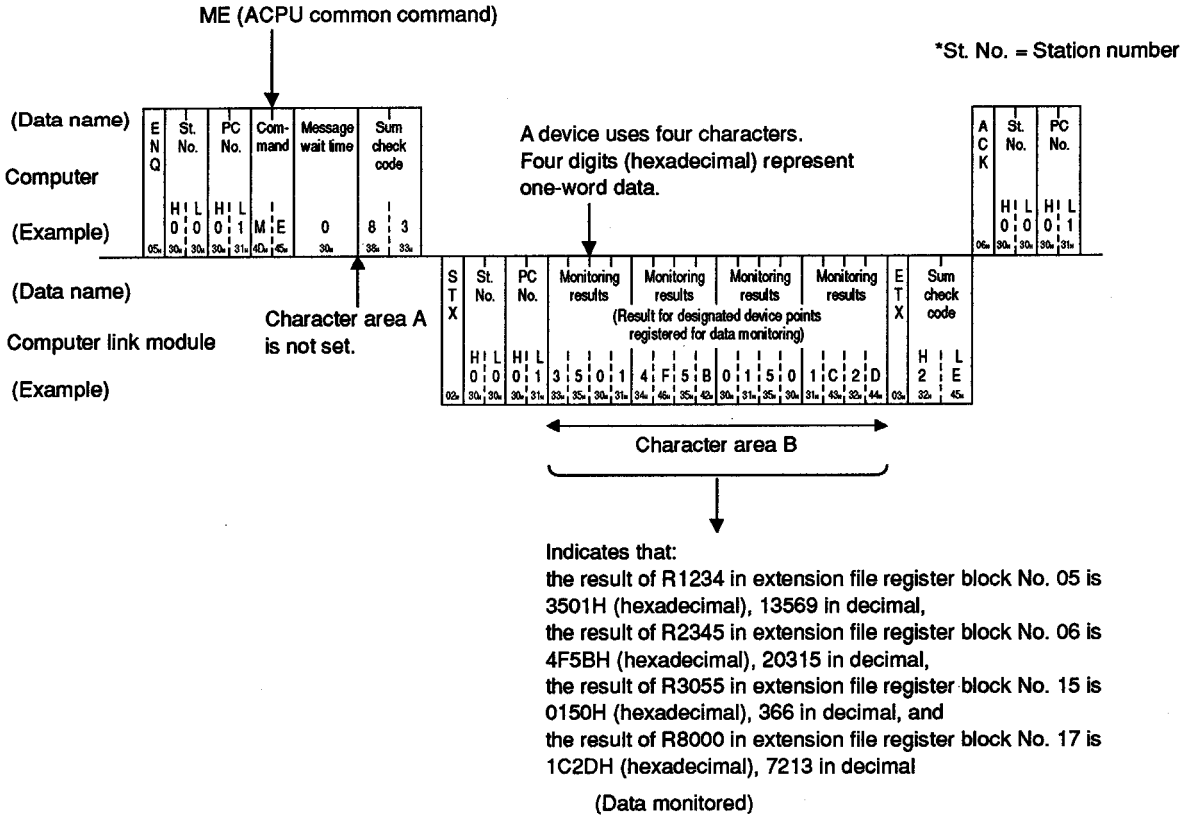
In this section, an example is quoted to describe the control protocol for monitoring extension file registers, which are registered with the computer link module using an EM command as shown in the preceding section (2), using an ME command.

[Control protocol]

The protocol shown below is in control format 1.
To access an extension file register in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Monitoring conditions)

1) The message wait time is 0 ms.
2) Four extension file registers (four words) registered for data monitoring, R1234 in block No. 5, R2345 in block No. 6, R3055 in block No. 15, and R8000 in block No. 17, are to be monitored (when extension file registers are reserved for blocks No. 1 to No. 8 and for No. 10 to No. 17).



5.9 Buffer Memory Read and Write

This function is used to read from and write to the computer link module buffer memory. When this function is used, communications between the computer and computer link module commences immediately when the computer sends a read or write request, without waiting for the PC CPU END processing. Therefore, the time T1, described in Section 5.5, is always equal to zero.
The PC CPU carries out buffer memory read and write using TO and FROM instructions.

The method for specifying the control protocol, meanings, and examples for carrying out this function are shown below.

5.9.1 Commands and buffer memory

This section describes commands to be used to read or write data from or to the buffer memory in the computer link module, and buffer memory addresses to be designated for the control protocol.

(1) Commands

Item	Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
	Symbol	ASCII Code			During STOP	During RUN		
						Write Enabled	Write Disabled	
Batch read	CR	43H, 52H	Reads from buffer memory.	64 words (128 bytes)	O	O	O	5.9.2
Batch write	CW	43H, 57H	Writes to buffer memory.					5.9.3

Note : O.....Executable

(2) Buffer memory and access unit

Buffer memory addresses are 0H to 7FFH (see Section 3.10).
One address consists of 1 word (16 bits).
Read and write are both executed in word units, regardless of the word/byte unit setting (set at buffer memory address 103H).

POINTS

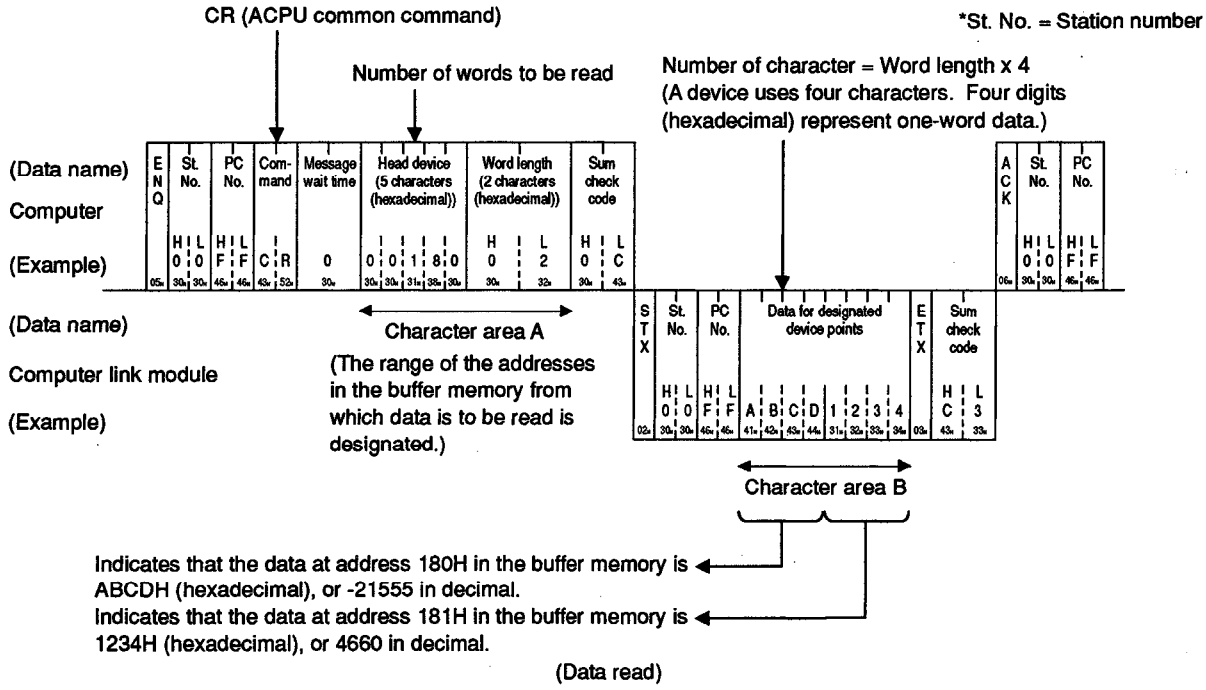
(1) Buffer addresses 100H to 11FH comprise the area. The computer link module will not operate correctly if any operations other than those described in the following sections are executed.
(2) When this function and the on-demand function in Section 5.14, the no-protocol mode in Section 6 or the bidirectional protocol in Section 7 are used together to access the user free area in the buffer memory simultaneously, do not designate the following areas in the buffer memory with the commands shown in Section 5.9.1 (1):
- No-protocol mode send area (or bidirectional mode send area)
- No-protocol mode receive area (or bidirectional mode receive area)
- On-demand area

5.9.2 Reading data from buffer memory

In this section, an example is quoted to describe the control protocol for reading data from the buffer memory in computer link module connected to the computer using a CR command.

[Control protocol] The protocol shown below is in control format 1.
To access the buffer memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)
1) The message wait time is 0 ms.
2) Data at two buffer memory addresses (two words), 180H and 181H, is to be read.



POINT

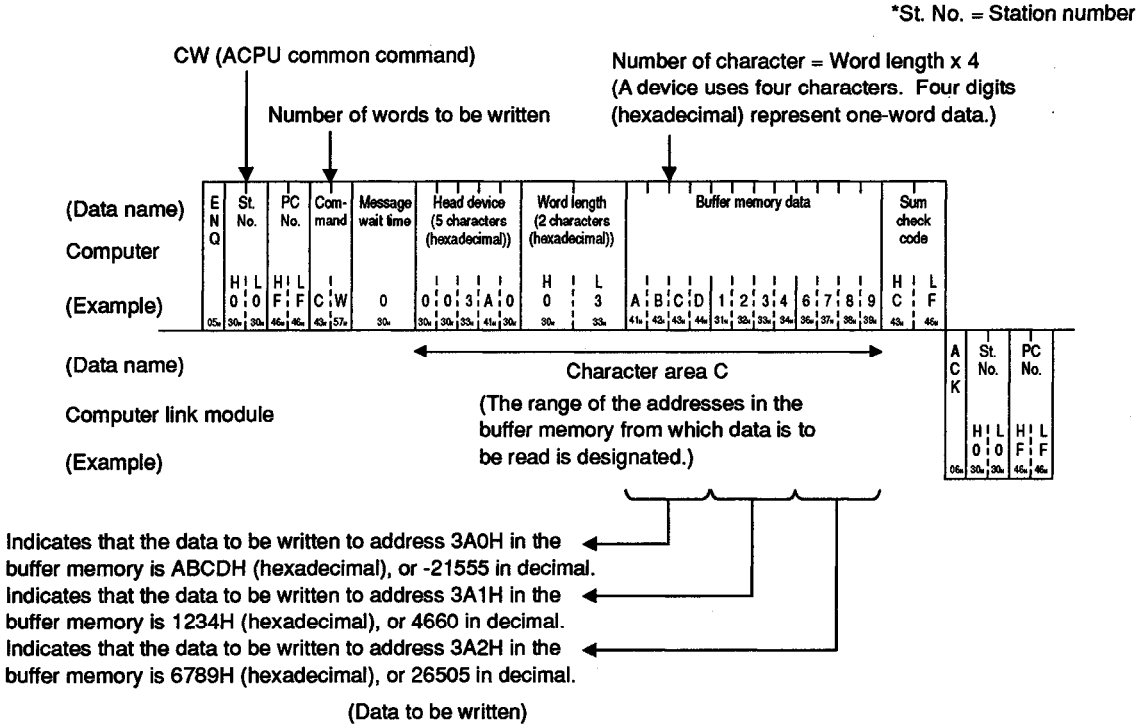
To designate the word length, the following conditions must be met:
- 1 ≤ word length ≤ 64
- (Head address) + (word length) – 1 ≤ maximum address number (DFFH)

5.9.3 Writing data to buffer memory

In this section, an example is quoted to describe the control protocol for writing data to the buffer memory in the computer link module connected to the computer using a CW command.

[Control protocol] The protocol shown below is in control format 1.
To access the buffer memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)
1) The message wait time is 0 ms.
2) Data (three words) is to be written to three buffer memory addresses, 3A0H to 3A2H.



POINT

To designate the word length, the following conditions must be met:
- 1 ≤ word length ≤ 64
- (Head address) + (word length) – 1 ≤ maximum address number (DFFH)

5.10 Special Function Module Buffer Memory Read and Write

In this section, examples are quoted to describe the control protocols for reading or writing data from or to the buffer memory in a special function module.
When a command described in this section is used, access to the buffer memory in a special function module is made in units of bytes.

5.10.1 Commands and designation

(1) ACPU common commands

Item	Command		Description	umber of Points Processed per Communication	PC CPU State			Reference Section
	Symbol	ASCII Code			During STOP	During RUN		
						Write Enabled	Write Disabled	
Batch read	TR	54H, 52H	Reads from special function module buffer memory.	128 bytes (64 words)	O	O	O	5.10.3
Batch write	TW	54H, 57H	Writes to special function module buffer memory.		O	O	X	5.10.4

Note : O .....Executable
X.....Not executable

(2) Accessible module types and buffer memory addresses

Refer to APPENDIX 3 for special function modules accessible with this function and the head address in the buffer memory to be designated.
When this function is used, data is read or written in units of bytes, irrespective of the word/byte units setting (specified at address 103H in the computer link module buffer memory).

(3) Special-function module buffer memory

The special-function module buffer memory is comprised of 16-bit (one word) addresses. Read and write of the special-function module buffer memory is executed by TO and FROM instructions transmitted between the PC CPU and special-function module.

When the computer reads from and writes to the special-function module buffer memory via the computer link module, it is done in byte units (1 address = 8 bits).

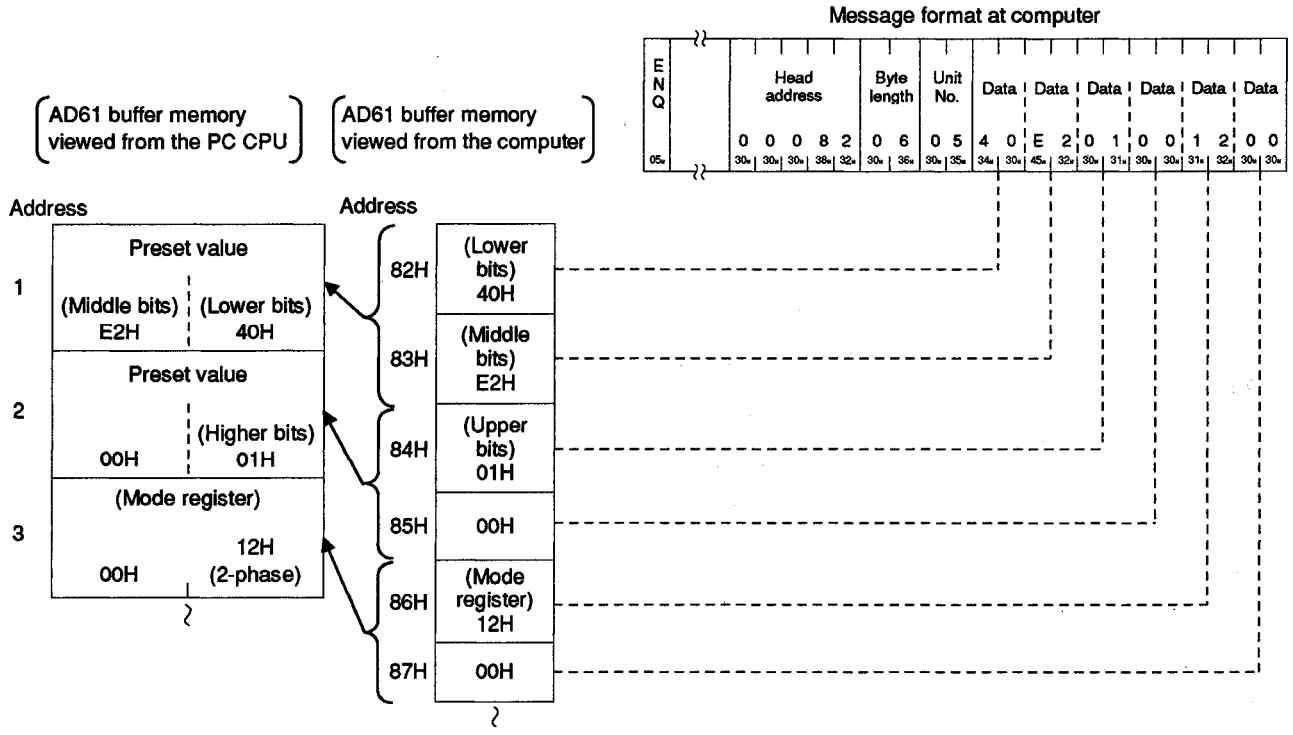
The addresses specified in the computer (hexadecimal) are converted from FROM/TO instruction addresses as shown below:

Designated address (hexadecimal) = [(FROM/TO instruction address × 2) converted into hexadecimal] + Module head

Example: To designate AD61 high-speed counter module FROM/TO instruction address 1 (CH.1 preset value).

Specified address		FROM/TO instruction address 1 × 2		Head address
82H	=	2H	+	80H

The data format when the computer makes a read or write to or from the special-function module buffer memory via the computer link module, is explained below using the AD61 module as an example.



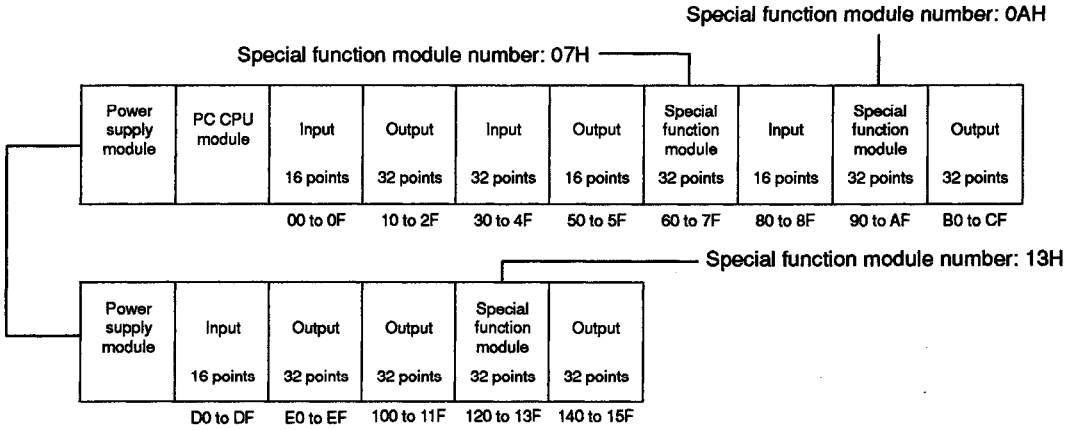
POINT

The buffer memory of each special-function module has its read and write area, read-only and write-only areas, and areas reserved for OS use, which are not available to the use. Refer to the manual for each module before using the buffer memory.
PC CPU or special-function module errors may occur if reading or writing is not done correctly.

5.10.2 Special function module numbers using control protocols

(1) Module numbers of special function modules occupying one slot

The special function module numbers designated by using control protocols are the upper 2 digits of the last special function module I/O address expressed in 3 digits.

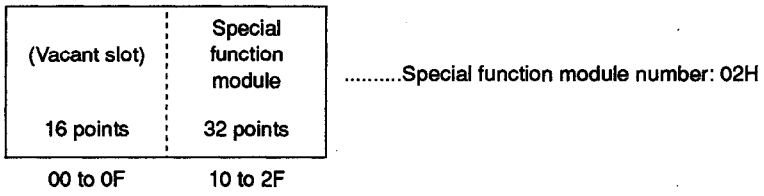


(2) Module number of special function modules occupying two slots

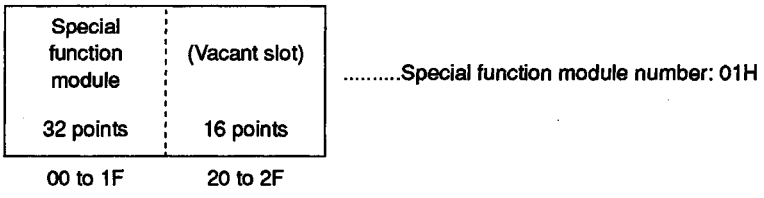
For special function modules occupying two slots, the number of points occupied by each slot is fixed for each module. The special function module number is the upper two digits of the three-digit number representing the last address of the slot allocated to the module.

The User's Manual for each special function module gives details about the allocation of slots to each module.

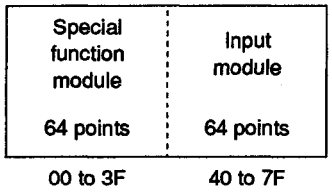
(a) Modules with the front slot allocated as the vacant slot (AD72, A84AD, etc.)



(b) Modules with the rear slot allocated as the empty slot (A61LS, etc.)



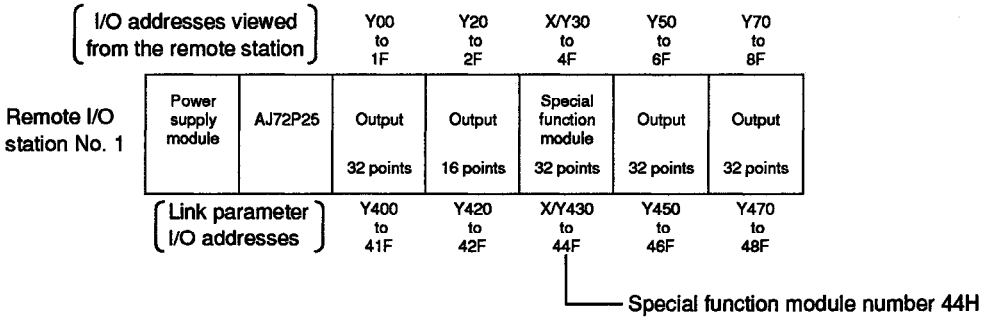
(c) Modules with the special function module allocation and I/O allocation mixed (A81CPU, etc.)



(3) Module numbers of special-function modules at MELSECNET remote I/O stations

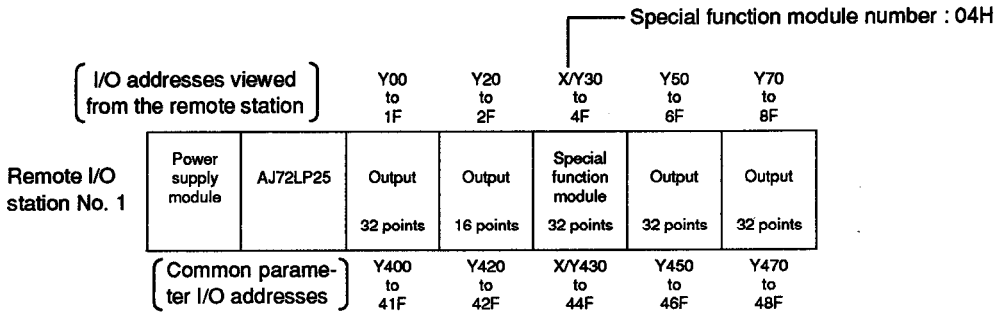
The module numbers of special function modules at MELSECNET remote I/O stations are determined by link parameters setting at the MELSENET master station.

L/R NO.	M ← L		M → R	M ← R	M → L/R		M ← L/R	
	B	W	W	W	Y	X/Y	X	Y/X
R1	----	----	29C–309	0F9–15E	400–48F	000–08F	430–44F	030–04F
R2	----	----	215–24F	080–0A3	510–67F	010–17F	500–65F	000–15F
R3	----	----	1B6–214	15F–1B5	270–32F	050–10F	220–28F	000–06F
	–	–	–	–	–	–	–	–
	–	–	–	–	–	–	–	–
	–	–	–	–	–	–	–	–
	–	–	–	–	–	–	–	–
	–	–	–	–	–	–	–	–



(4) Module numbers of special function modules at MELSECNET/10 remote I/O stations

The module numbers of special function modules at MELSECNET/10 remote I/O stations are expressed by the upper two digits of the three-digit numbers which represent the last addresses of the "I/O addresses viewed from the remote I/O station."
Designate special function module numbers according to the above rule, irrespective of the link parameters set on the master station on the MELSECNET/10 remote I/O network.



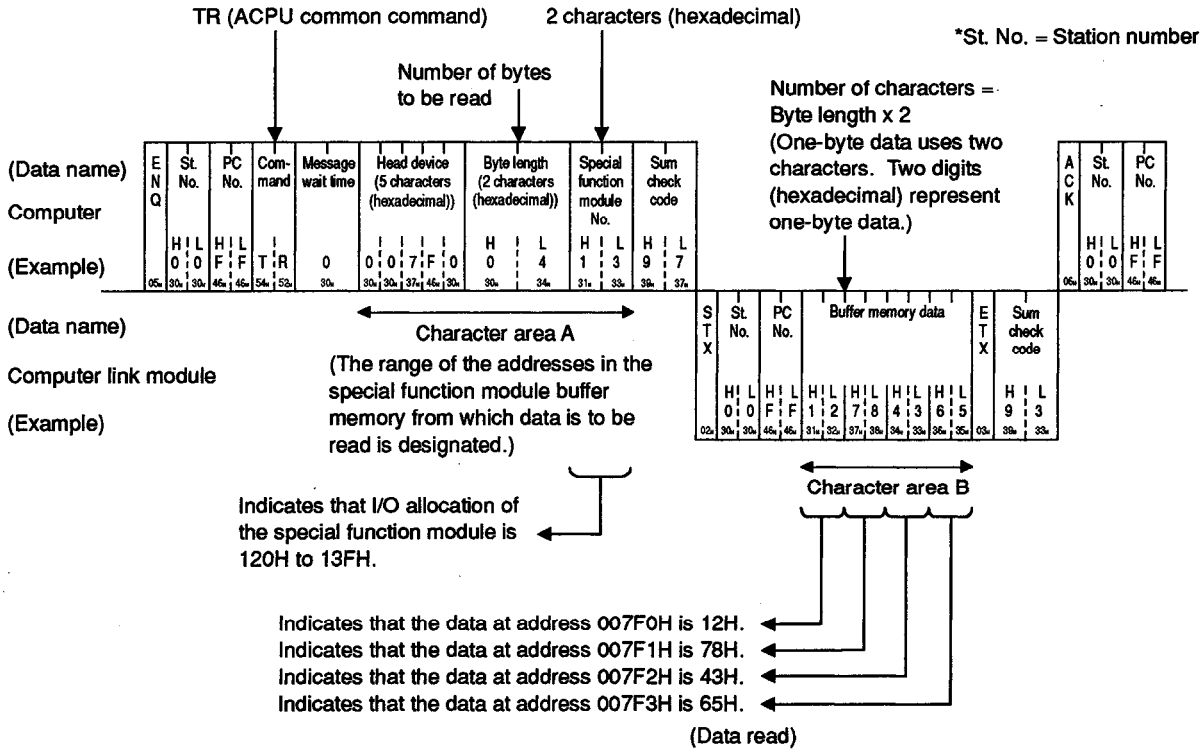
5.10.3 Reading data from the special function module buffer memory

In this section, an example is quoted to describe the control protocol for reading data from the buffer memory in a special function module using a TR command.

[Control protocol] The protocol shown below is in control format 1.
To access the buffer memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)

1) The message wait time is 0 ms.
2) Data (four bytes) is to be read from four addresses, 7F0H to 7F3H, in the buffer memory in the special function module with I/O addresses 120H to 13FH (module No. 13H).



POINTS

(1) To designate the byte length, the following condition must be met:
$1 \leq$ byte length ≤ 128

(2) With some special function modules, 2 or 3 bytes are used to express the data. Therefore, designate the byte length by referring to the manuals for each individual module.

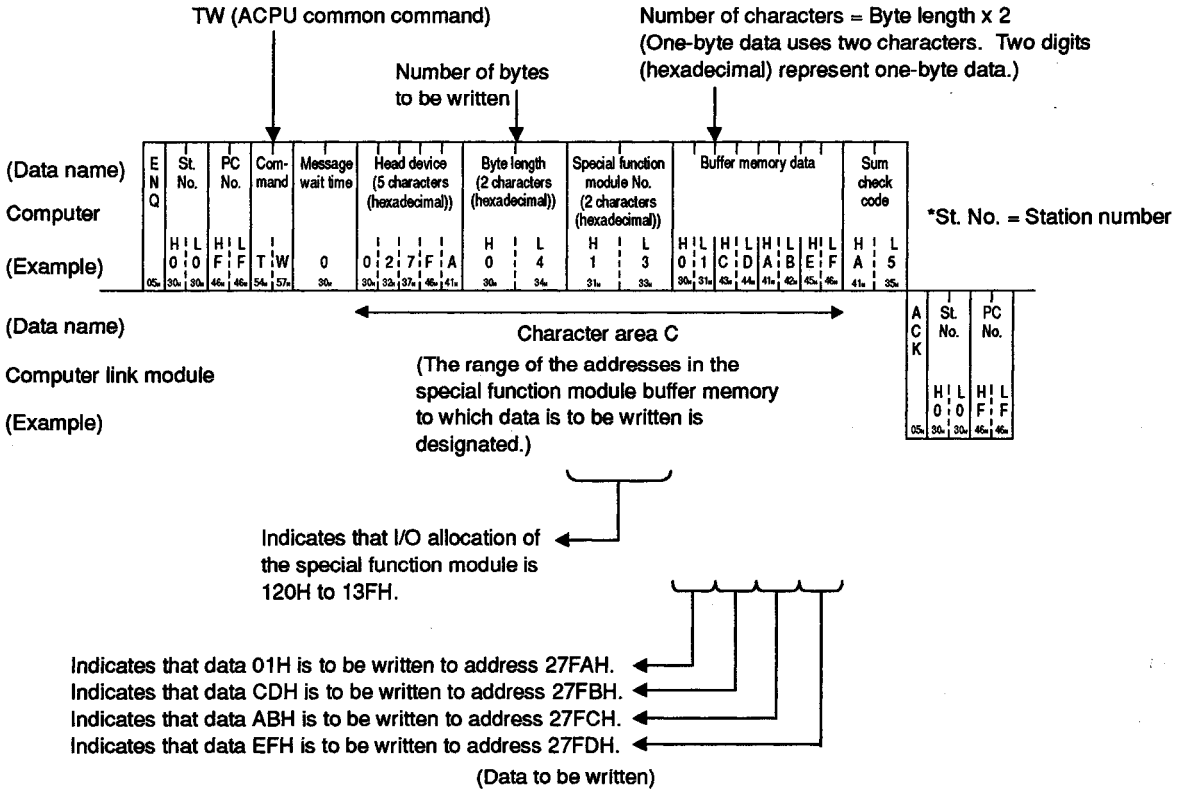
5.10.4 Writing data to the special function module buffer memory

In this section, an example is quoted to describe the control protocol for writing data to the buffer memory in a special function module using a TW command.

[Control protocol] The protocol shown below is in control format 1.
To access the buffer memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)

1) The message wait time is 0 ms.
2) Data (four bytes) is to be written to four addresses, 27FAH to 27FDH, in the buffer memory in the computer link module with I/O addresses 120H to 13FH (module No. 13H).



POINTS

(1) To designate the byte length, the following condition must be met:
$1 \leq$ byte length ≤ 128

(2) With some special function modules, 2 or 3 bytes are used to express the data. Therefore, designate the byte length by referring to the manuals for each individual module.

5.11 PC CPU Remote RUN/STOP and PC CPU Model Code/Name Read

These functions are used to remote-run or remote-stop the PC CPU from the computer, and to read what model of PC CPU is linked with the computer.

In this section, examples are quoted to describe the control protocols for these functions.

5.11.1 Commands and their functions

(1) The following table shows the ACPU common commands to be used to remote-run or remote-stop the PC CPU and to read the PC CPU model, Remote Run/Stop of PC CPU and Reading PC CPU Model Name

Item	Command		Description	PC CPU State			Reference Section
	Symbol	ASCII Code		During STOP	During RUN		
					Write Enabled	Write Disabled	
Remote RUN	RR	52H, 52H	Requests remote RUN of PC CPU.	O	O	O	5.11.2
Remote STOP	RS	52H, 53H	Requests remote STOP of PC CPU.	O	O	O	
PC CPU model code read	PC	50H, 43H	Reads the model code of the PC CPU.	O	O	O	5.11.3
PC CPU model code/name read	PU	50H, 55H	Reads the model code and name of the PC CPU.	O	O	O	

Note : O..... Executable

5.11.2 Remote RUN/STOP

(1) Remote RUN/STOP control

(a) RUN, STOP, PAUSE and STEP-RUN states are produced by the following combinations of PC CPU key switch positions and computer commands.

		PC CPU Key Switch Position			
		RUN	STOP	PAUSE	STEP-RUN
Command from computer	Remote RUN	RUN	STOP	PAUSE	STEP-RUN
	Remote STOP	STOP	STOP	STOP	STOP

REMARK

When a PC CPU is stopped by the remote STOP command given by an external computer, that PC CPU cannot be put into the RUN state by the computer connected to the PC CPU.

(b) The clearing of data memories on receiving a remote RUN instruction depends on the states of special relays M9016 and M9017 as shown below.

Special Relay		Data Memory State
M9016	M9017	
OFF	OFF	PC CPU enters the RUN state without clearing remote STOP data.
OFF	ON	Remote STOP data is cleared outside the latch range set in parameters. (In this case, Link X image is not cleared.)
ON	ON/OFF	PC CPU enters the RUN state after data memory is cleared.

REMARK

Always reset special relays M9016 and M9017 when data memory clearing is not required.

POINT

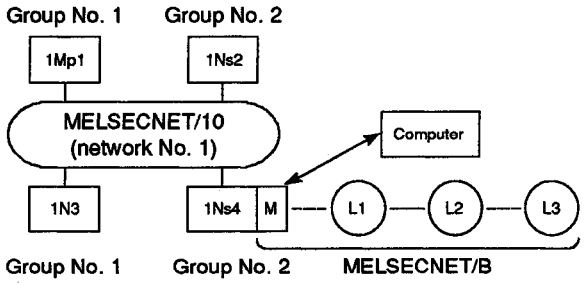
After operations remote RUN/STOP control from the computer are completed, the remote data will be lost if the power supply is turned ON or the PC CPU is reset.

(2) Remote RUN/STOP of stations on the MELSECNET/10 (Only when the stations connected to the computer are AnUCPUs)

When the station (self station) connected to the computer is an AnUCPU on the MELSECNET/10, all the stations on the network number registered with the self station (refer to Section 5.15.3) are remote-run or remote-stopped by designating either of the dedicated PC CPU numbers shown below.

Dedicated PC CPU No.	Station to be Remote-Run/Remote-Stopped
F0H	All AnUCPU stations on network registered
8[ ]H ([ ]: Group No. 1 to 9)	All AnUCPU stations in group No. [ ] on network registered

[Example] MELSECNET/10 and MELSECNET/B composite system



1Mp1 (AnUCPU) ..... Control station (network No. 1)
1Ns2 (AnUCPU) ..... Normal station (network No. 1)
1N3 (AnACPU) ..... Normal station (network No. 1)
1Ns4/M (AnUCPU) ... Normal station (network No. 1)
Master station
L1 (AnNCPU) ..... Local station
L2 (AnNCPU) ..... Local station
L3 (AnNCPU) ..... Local station

Data on Registered Network Including Stations Connected to Computer	PC CPU No. Designation Example	Station to be Remote-Run/Remote-Stopped						
		1Mp1	1Ns2	1N3	1Ns4/M	L1	L2	L3
MELSECNET/10 (network No. 1)	F0H	O	O	X	O	X	X	X
	81H	O	X	X	X	X	X	X
	82H	X	O	X	O	X	X	X
MELSECNET/B	F0H	X (The computer link module responds with an NAK message.)						
	81H							

O .....The PC CPU is remote-run or remote-stopped by designating a dedicated PC CPU number.

POINTS

(1) Only AnUCPUs can be remote-run or remote-stopped by designating a dedicated PC CPU number.

(2) Regarding the following systems, a dedicated PC CPU number can be designated in the message requesting remote run or remote stop sent from the computer:
   (a) The PC CPU loaded in the computer link module connected to the computer is an AnUCPU.
   (b) The PC CPU requested by the computer to be remote-run or remote-stopped is an AnUCPU which is connected on the MELSECNET/10.

(3) Remote run or remote stop is unavailable for systems and PC CPUs other than those described in (2) above even if a dedicated PC CPU number is designated.
   In the following cases, the computer link module sends an NAK message:
   (a) The PC CPU loaded in the computer link module connected to the computer is not an AnUCPU.
   (b) The PC CPU loaded in the computer link module connected to the computer is an AnUCPU, but the MELSECNET/10 is not registered as a network (refer to Section 5.15.3).

(4) After making a remote run or remote stop request from the computer with a dedicated PC CPU number designated, check the PC CPU status after the processing with the special register for MELSECNET/10 linking even if the computer link module sends an ACK signal.

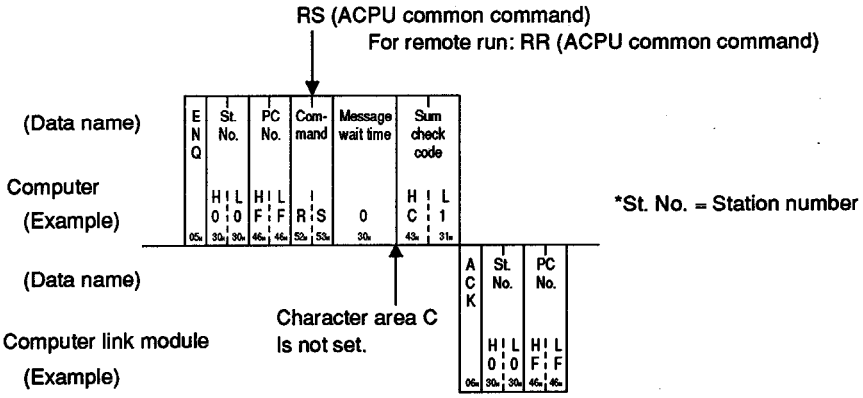
(3) Control protocols for remote RUN/STOP

In this section, examples are quoted to describe the control protocols for remote-running or remote-stopping the PC CPU using an RR or RS command.

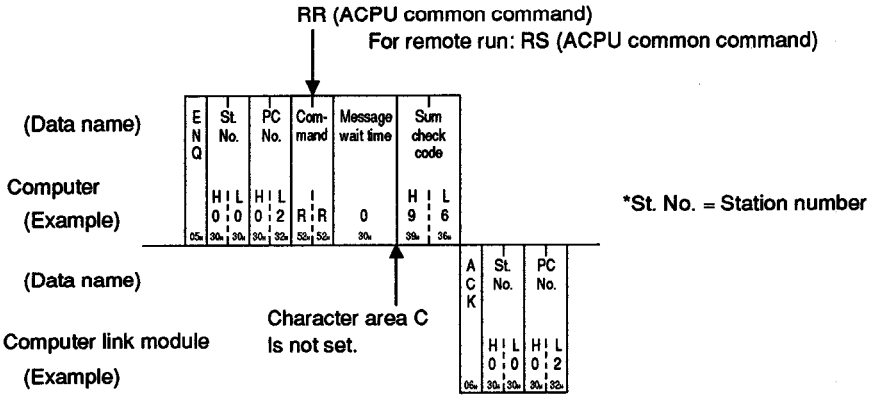
[Control protocol]

The protocol shown below is in control format 1.
To access the PC CPU in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Example 1: Remote control conditions)
1) The message wait time is 0 ms.
2) The PC CPU (self station) loaded with a computer link module (station No. 00) connected to the computer is to be remote-stopped.

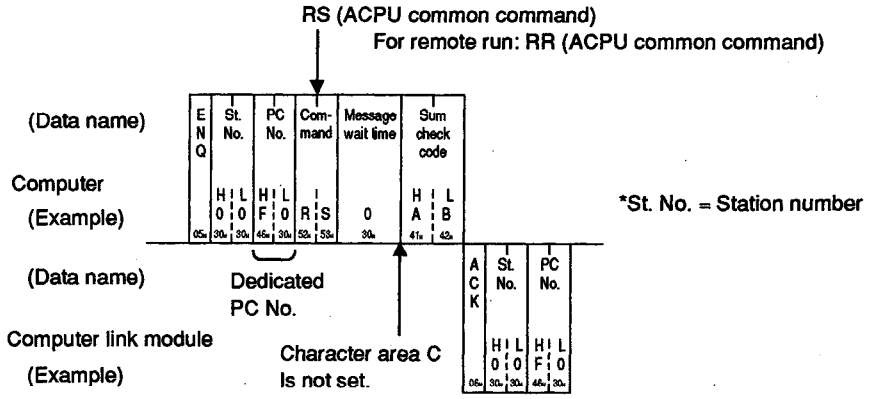


(Example 2: Remote control conditions)
1) The message wait time is 0 ms.
2) The PC CPU (station No. 02) on the data link system or network system is to be remote-run through the PC CPU loaded with a computer link module (station No. 00) connected to the computer.



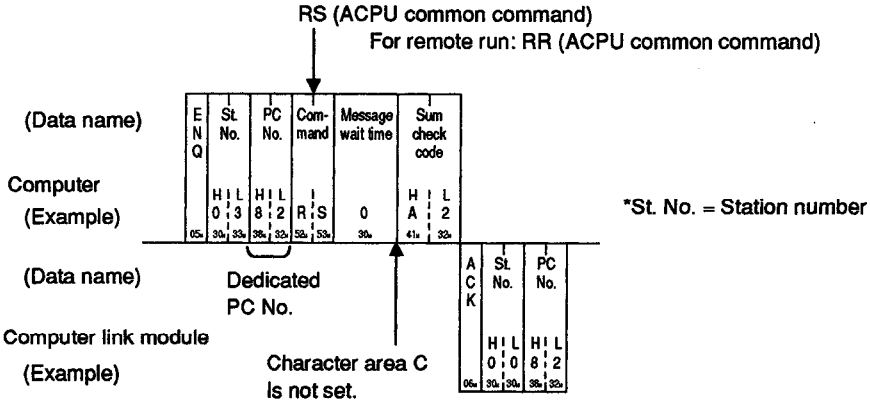
(Example 3: Remote control conditions)

1) The message wait time is 0 ms.
2) All AnUCPUs on the MELSECNET/10, which is registered with the AnUCPU (self station) loaded with a computer link module (station No. 00) connected to the computer, are to be remote-stopped.



(Example 4: Remote control conditions)

1) The message wait time is 0 ms.
2) All AnUCPUs in group No. 2 on the MELSECNET/10, which is registered with the AnUCPU (self station) loaded with a computer link module (station No. 03) connected to the computer, are to be remote-stopped.



5.11.3 PC CPU model code and name read

This function is used to read the model code and name of the PC CPU being linked with the computer.

(1) Readable PC CPU model codes and names

PC CPU Model	Data to be Read		PC CPU Model	Data to be Read	
	Model Code (Hexadecimal)	Model Name		Model Code (Hexadecimal)	Model Name
A0J2HCPU	98H	A0J2H			
A1CPU, A1NCPU	A1H	*1	A4UCPU	85H	A4U
A1SCPU(-S1), A1SJCPU	98H	A1S	A52GCPU	9AH	A52G
A1SHCPU	A3H	A1SH	A73CPU	A3H	*1
A1SJHCPU		A2SH	A7LMS-F	A3H	*1
A2CPU(-S1)	A2H	A2			
A2NCPU(-S1)		*1	AJ72P25/R25	ABH	*1
A2SCPU(-S1)		*1	AJ72LP25/BR15	8BH	L/B25
A2ACPU	92H	A2A			
A2ACPU-S1	93H	A2AS1	Q2ACPU(-S1)	93H	———
A2CCPU	9AH	A2C	Q3ACPU	94H	———
A2UCPU, A2ASCPU	82H	A2U	Q4ACPU		
A2UCPU-S1, A2ASCPU-S1	83H	A2US1	Q4ARCPU		
A2SHCPU(-S1)	A3H	A2SH1	AJ72QLP25	84H	———
A2USHCPU-S1	84H	A2USH	AJ72QBR15		
A3CPU	A3H	A3			
A3NCPU		*1	Q2ASCPU(-S1)		
A3ACPU	94H	A3A	Q2ASHCPU(-S1)	93H	———
A3HCPU	A4H	A3H			
A3MCPU		A3M			
A3UCPU	84H	A3U			

*1 represents blank (20H).

POINTS

(1) The model commands are all readable with either PC or PU command described in this section, however, only the PU command can help read the model names.

(2) The model codes are expressed in two characters, and the model names in five characters.
If a model name read is represented by less than five characters, the computer link module makes up the lacking character(s) with the equal number of blanks (20H), thereby returning a five-character name.

(2) Reading the PC CPU model code and name

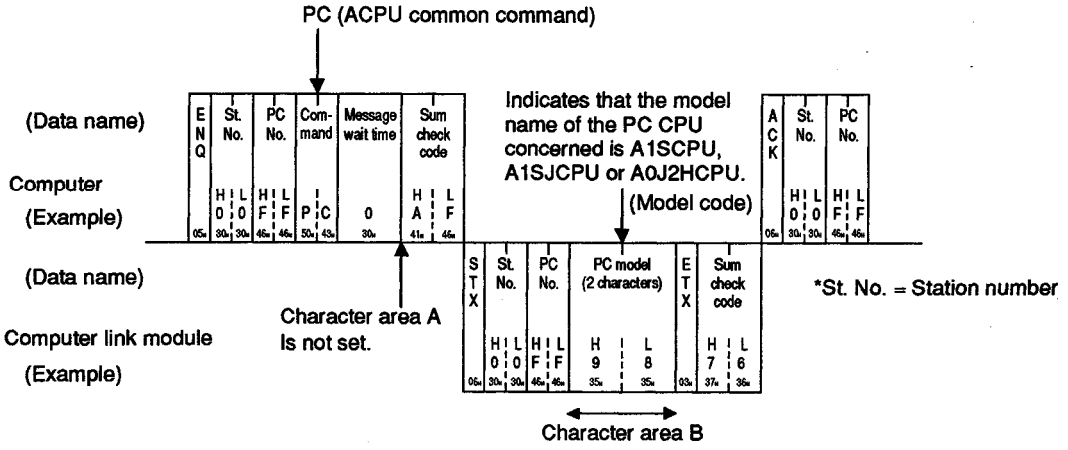
In this section, examples are quoted to describe the control protocols for reading the model code and name of the PC CPU linked with the computer using a PC or PU command.

(a) Reading the model code using a PC command

[Control protocol]

The protocol shown below is in control format 1.
To access the PC CPU in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)
1) The message wait time is 0 ms.
2) The model code of the PC CPU (self station) loaded with a computer link module is to be read.

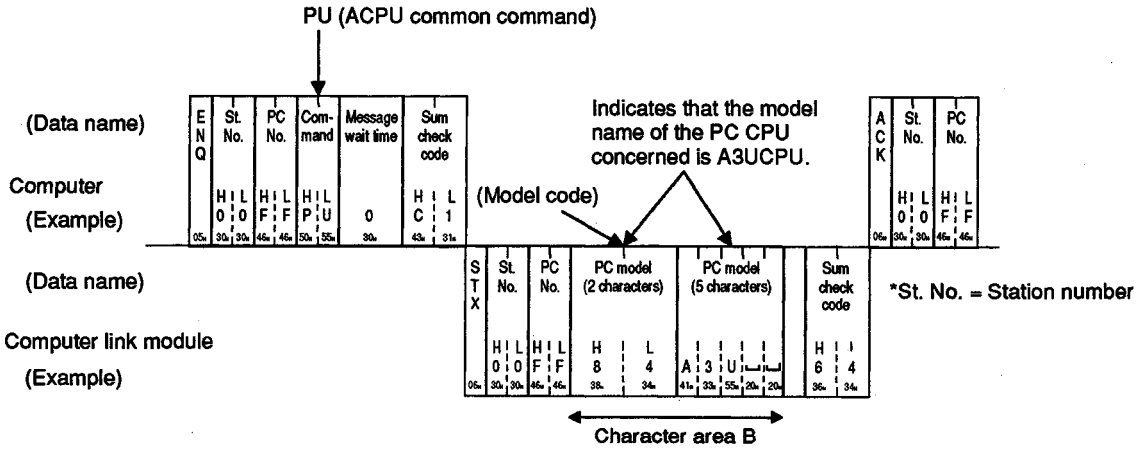


(b) Reading the model code and name using a PU command

[Control protocol]

The protocol shown below is in control format 1.
To access the PC CPU in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)
1) The message wait time is 0 ms.
2) The model code and name of the PC CPU (self station) loaded with a computer link module are to be read.



5.12 Program Read/Write

This function is used to transfer all types of programs (main and subsequence programs, microcomputer main and sub programs), parameters and comment data from the PC CPU and store them in the computer. The computer then carries out the appropriate controls by writing programs, parameters, and comment data to the PC CPU.

5.12.1 Precautions during program read/write

(1) When reading programs that have been written to the PC CPU, read all sequence programs, microcomputer programs, parameter data, and comment data from all areas.
When writing programs, write all stored data to the PC CPU. If all areas have not been written to, the PC CPU will not work correctly.

(2) Before writing programs, write parameter data and execute a parameter analysis request. Otherwise, the parameters in the PC CPU user memory will be changed but the parameters stored in the work area by the ACPU for operation will remain unchanged. Therefore, if a peripheral device is loaded and operated after the parameters are changed, processing will be carried out with the previous parameters, which are still stored in the work area.

(3) The number of points which can be processed per communications is fixed. When reading or writing data, divide the data into several groups to read or write the entire area. Parameter data should be divided into 3k bytes. Other data shoule be divided into units of data determined by parameter setting.

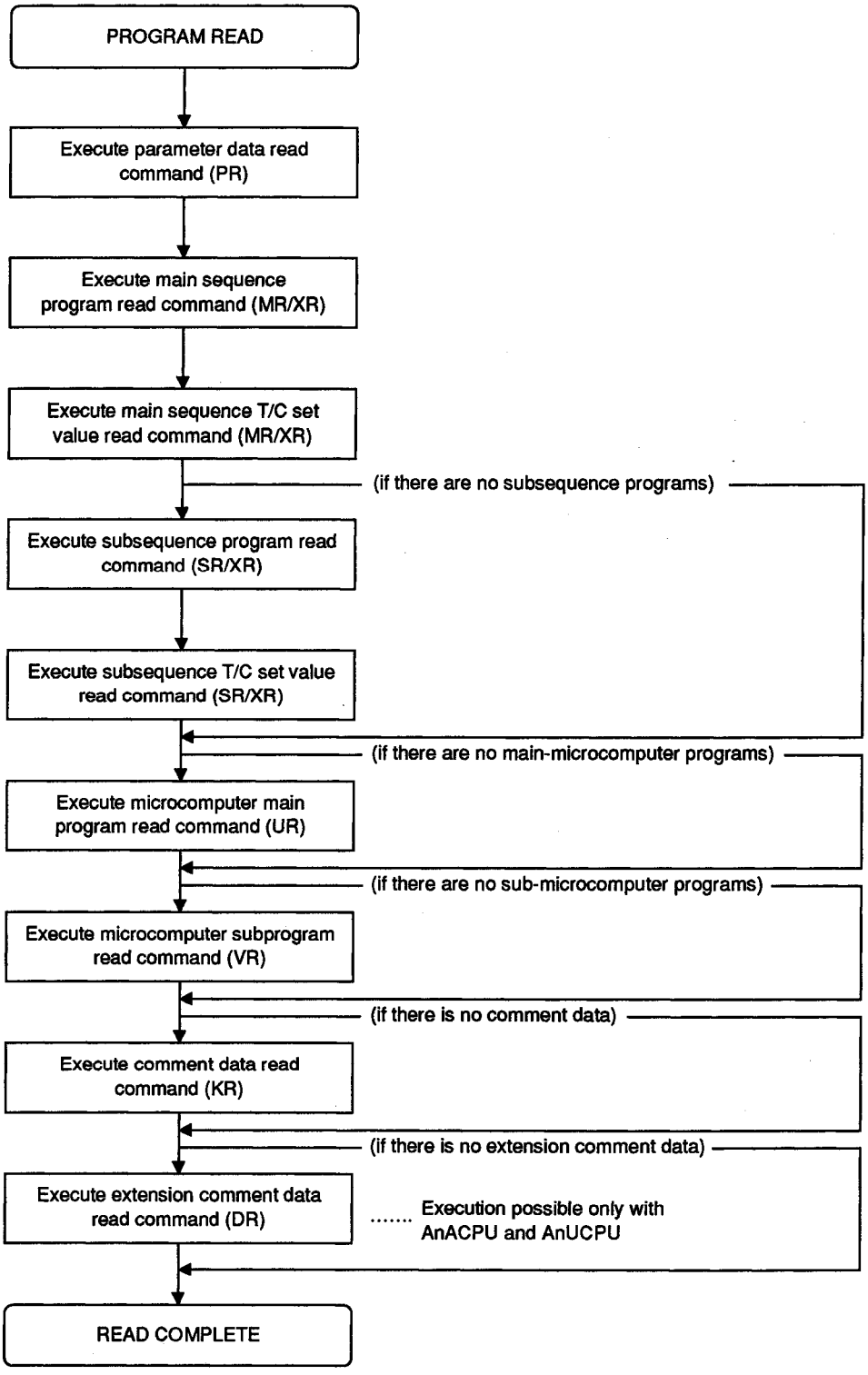
(4) When the PC CPU, which reads or writes the sequence program, is an AnUCPU, network parameters are read or written together with other parameters.
To read or write the network parameters, use the parameter memory read/write function described in Section 5.12.3.

(5) A program read by this function into the computer cannot be modified on the computer. Keep it as a backup copy.

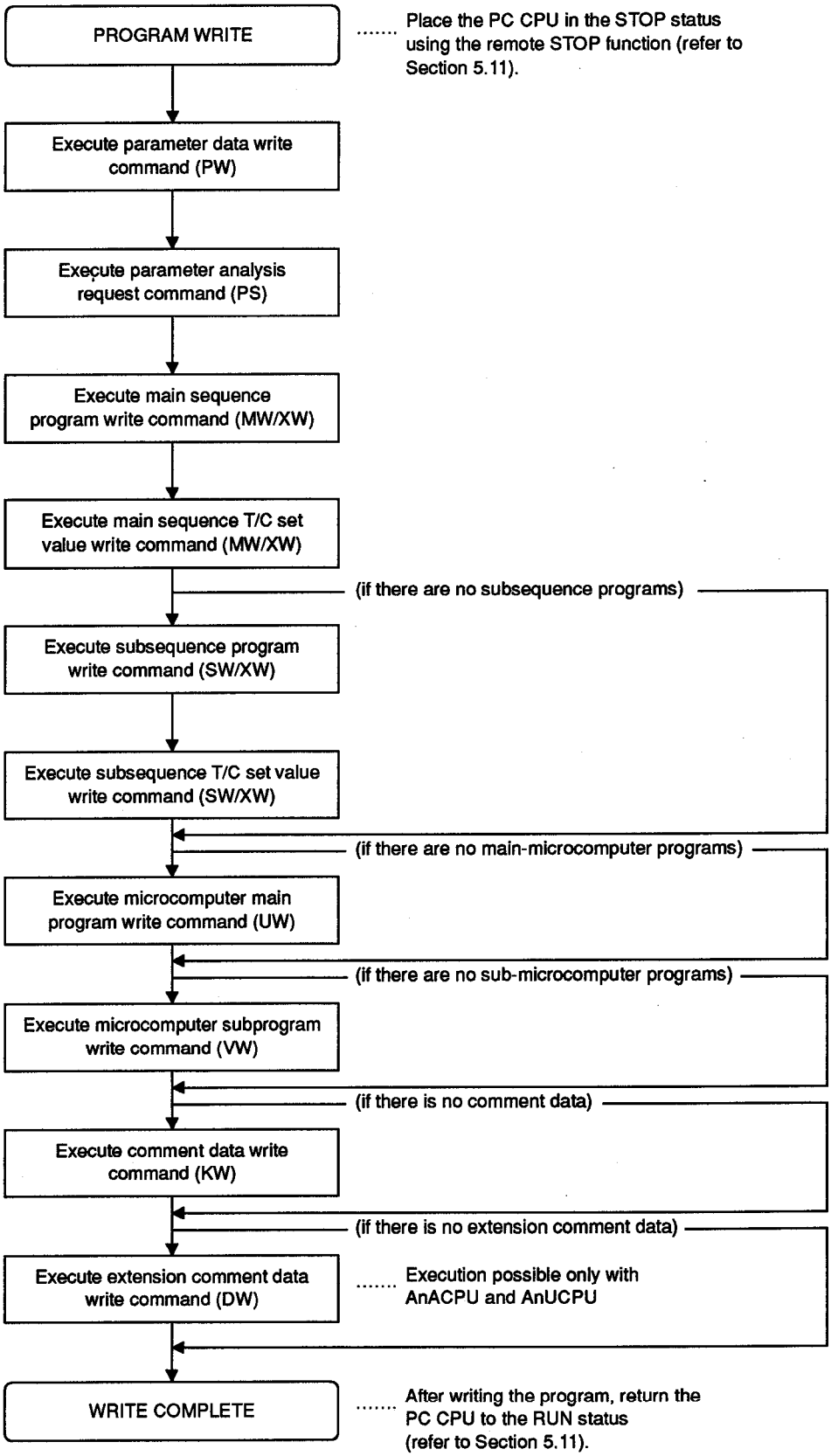
(6) When the PC CPU is an AnA or AnUCPU, the SFC program is read or written by the main microcomputer program read/write function.
To read or write the SFC program, use the microcomputer program read/write function described in Section 5.12.5.

5.12.2 Program read/write control procedures

(1) Reading



(2) Writing


PROGRAM WRITE
Place the PC CPU in the STOP status using the remote STOP function (refer to Section 5.11).
Execute parameter data write command (PW)
Execute parameter analysis request command (PS)
Execute main sequence program write command (MW/XW)
Execute main sequence T/C set value write command (MW/XW)
(if there are no subsequence programs)
Execute subsequence program write command (SW/XW)
Execute subsequence T/C set value write command (SW/XW)
(if there are no main-microcomputer programs)
Execute microcomputer main program write command (UW)
(if there are no sub-microcomputer programs)
Execute microcomputer subprogram write command (VW)
(if there is no comment data)
Execute comment data write command (KW)
(if there is no extension comment data)
Execute extension comment data write command (DW)
Execution possible only with AnACPU and AnUCPU
WRITE COMPLETE
After writing the program, return the PC CPU to the RUN status (refer to Section 5.11).

5.12.3 Parameter memory read/write

(1) Commands and addresses

(a) ACPU common commands

Item	Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
	Symbol	ASCII code			During STOP	During RUN		
						Write Enabled	Write Disabled	
Batch read	PR	50H, 52H	Reads parameters.	128 bytes	o	o	o	5.12.3 (2)
Batch write	PW	50H, 57H	Writes parameters.		o	x	x	5.12.3 (3)
Analysis request	PS	50H, 53H	Causes the PC CPU to acknowledge and check rewritten parameters.		o	x	x	5.12.3 (4)

Note : o..... Executable
x..... Not executable

(b) Parameter addresses

There are 3k bytes of parameter memory, addresses 00000H to 00BFFH.
For addresses, use 5-digit ASCII (hexadecimal).

POINTS

(1) This function is applicable to the parameters set in making the memory capacity and other settings for the GPP function, and to the network parameters for the MELSECNET/10.
When the PC CPU, which reads or writes the parameters, is an AnUCPU, the network parameters are read or written together with the parameters.
When reading or writing the network parameters, read or write all bytes of parameters, including the parameter memory (3 kbytes) and the network parameter memory.
The network parameter memory capacity is displayed on the network parameter setting screen for the GPP function.
The network parameters can be read or written with one of the following versions of computer link module software:

Applicable module	AJ71UC24	A1SJ71 UC24-R2/R4	A1SJ71 C24-R2/R4/PRF	A1SCPU C24-R2	A2CCPU C24	A2CCPU C24-PRF
Version	A	R	M	A	K	K

(2) After changing parameters, always call the parameter analysis request command (PS).
If this is not done, the parameters in PC CPU user memory will be changed but the parameters stored in the work area by the ACPU for operation will remain unchanged. Therefore, if a peripheral device is loaded and operated after the parameters are changed, processing will be executed with the previous parameters, which are still stored in the work area.

(2) Parameter memory batch read

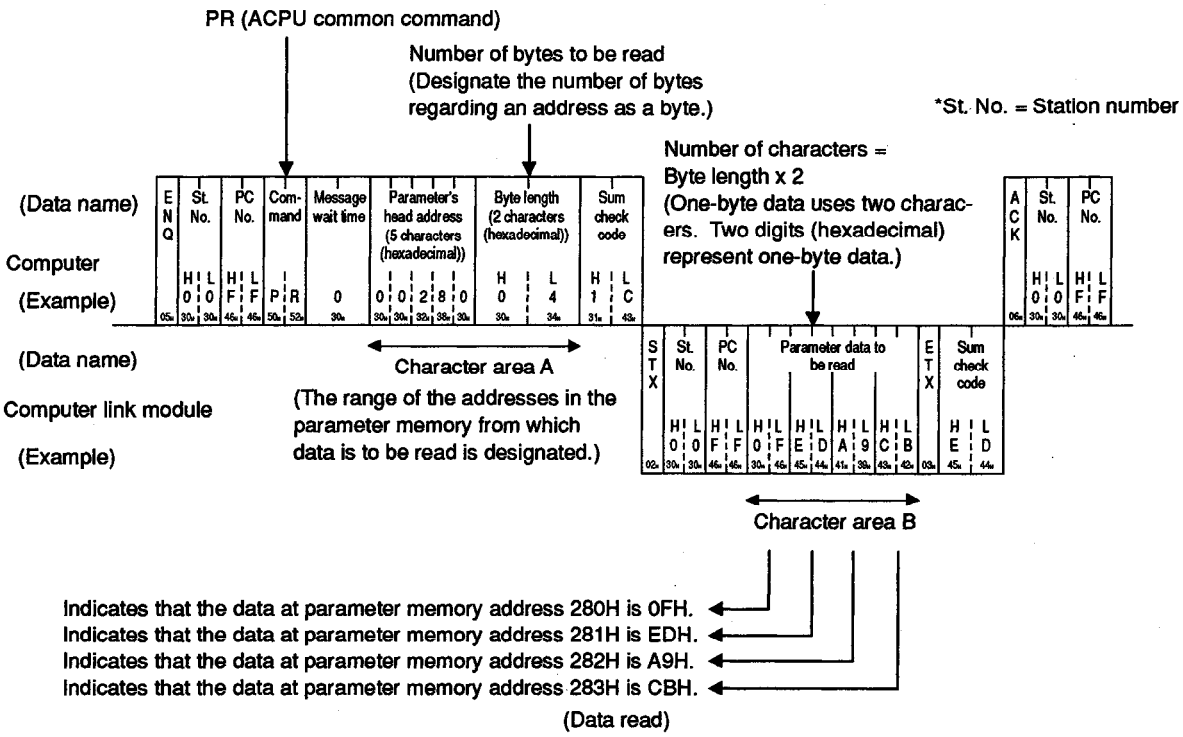
In this section, an example is quoted to describe the control protocol for batch-reading data from the PC CPU parameter memory using a PR command.

[Control protocol]

The protocol shown below is in control format 1.
To access the parameter memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)

1) The message wait time is 0 ms.
2) Data (four bytes) is to be read from four parameter memory addresses, 280H to 283H, in the PC CPU (self station) loaded with a computer link module.



POINT

To designate the byte length, the following condition must be met:

- 1 ≤ byte length ≤ 128

(3) Parameter memory batch write

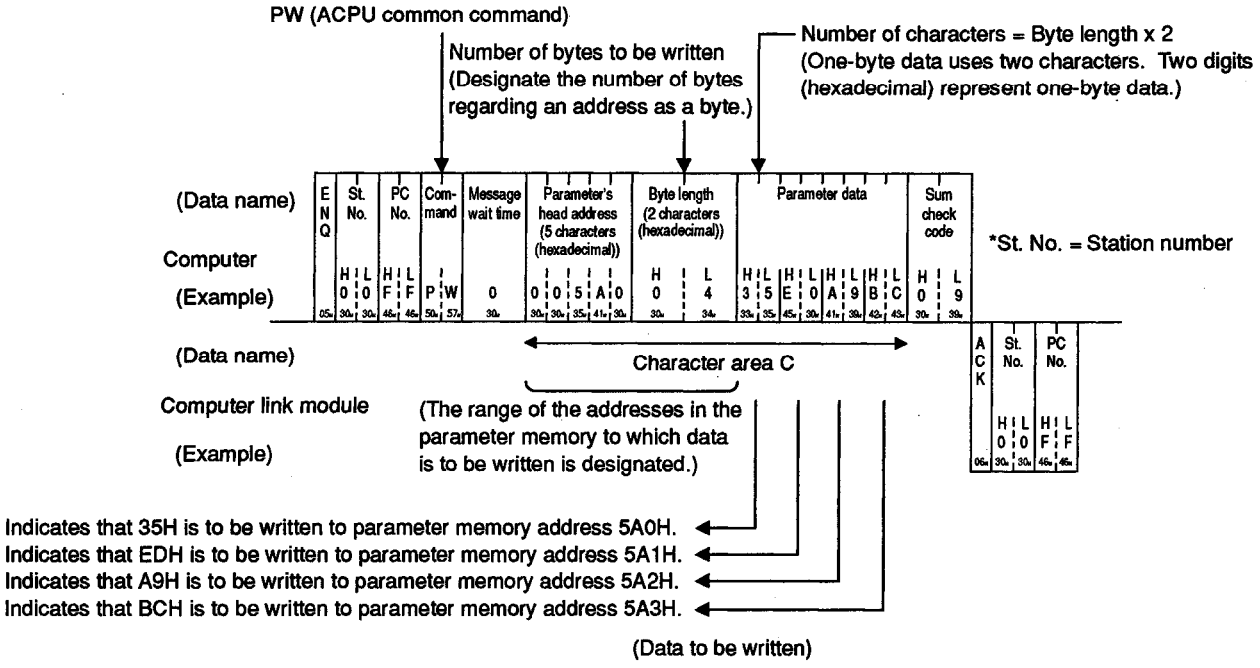
In this section, an example is quoted to describe the control protocol for batch-writing data to the parameter memory in the PC CPU using a PW command.

[Control protocol]

The protocol shown below is in control format 1.
To access the parameter memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)

1) The message wait time is 0 ms.
2) Parameter data (four bytes) is to be written to four parameter memory addresses, 5A0H to 5A3H, in the PC CPU (self station) loaded with a computer module.



POINT

To designate the byte length, the following condition must be met:

- 1 ≤ byte length ≤ 128

(4) Parameter memory analysis request

When data in the parameter memory in the PC CPU is changed, this function allows the PC CPU to identify the changed parameter and change the parameters in the work area in the PC CPU accordingly.

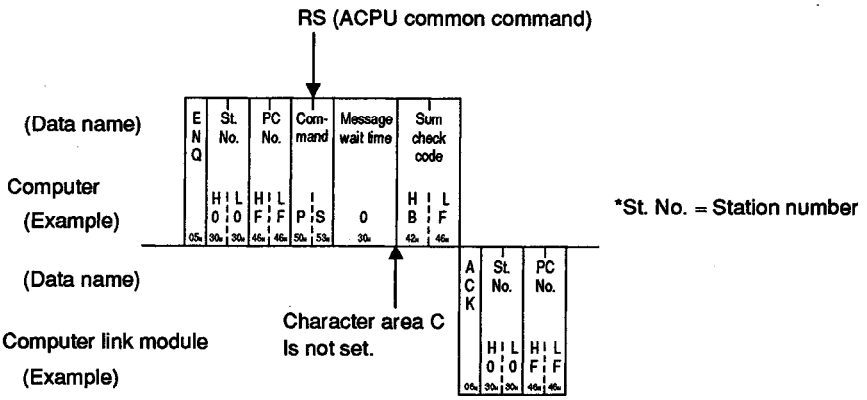
In this section, an example is quoted to describe the control protocol for executing an analysis request using a PS command.

[Control protocol]

The protocol shown below is in control format 1.
To access the parameter memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Analysis request conditions)

1) The message wait time is 0 ms.
2) An analysis of the parameters written to the parameter memory in the PC CPU (self station) loaded with a computer link module is to be requested.



5.12.4 Sequence program read/write

In this sections, examples are quoted to describe the control protocols for reading or writing the PC CPU sequence programs.

(1) Commands and step allocation

(a) ACPU common commands

Item			Command: Symbol	Command: ASCII Code	Description	Number of Points Processed per Communication	PC CPU State: During STOP	PC CPU State: During RUN: Writing Enabled	PC CPU State: During RUN: Writing Disabled	Reference Section
Batch read	Main	Except T/C set value	XR/MR	58H, 52H/ 4DH, 52H	Reads main sequence program.	64 steps	O	O	O	5.12.4 (2)
		T/C set value			Reads the T/C set value used in main sequence program.	64 points				
	Sub 1	Except T/C set value	XR/SR	58H, 52H/ 53H, 52H	Reads subsequence program 1.	64 steps	O	O	O	
		T/C set value			Reads the T/C set value used in subsequence program 1.	64 points				
	Sub 2 (A4U only)	Except T/C set value	XR	58H, 52H	Reads subsequence program 2.	64 steps	O	O	O	
		T/C set value			Reads the T/C set value used in subsequence program 2.	64 points	O	O	O	
	Sub 3 (A4U only)	Except T/C set value			Reads subsequence program 3.	64 steps	O	O	O	
		T/C set value			Reads the T/C set value used in subsequence program 3.	64 points	O	O	O	
Batch write	Main	Except T/C set value	XW/MW	58H, 57H/ 4DH, 57H	Writes main sequence program.	64 steps	O	O *	X	5.12.4 (3)
		T/C set value			Writes the T/C set value used in main sequence program.	64 points	O	O	X	
	Sub 1	Except T/C set value	XW/SW	58H, 57H/ 53H, 57H	Writes subsequence program 1.	64 steps	O	O *	X	
		T/C set value			Writes the T/C set value used in subsequence program 1.	64 points	O	O	X	
	Sub 2 (A4U only)	Except T/C set value	XW	58H, 57H	Writes subsequence program 2.	64 steps	O	O *	X	
		T/C set value			Writes the T/C set value used in subsequence program 2.	64 points	O	O	X	
	Sub 3 (A4U only)	Except T/C set value			Writes subsequence program 3.	64 steps	O	O *	X	
		T/C set value			Writes the T/C set value used in subsequence program 3.	64 points	O	O	X	

Note : O .....Executable
X.....Not executable

* Writing during a program run may executed out if all the following conditions are met:

1) The PC CPU is A3, A3N, A3H, A3M, A73, A3A, A3U, or A4U.
2) The program is not the currently running program (indicates a sub-program called by the main program, if the main program is being run).
3) The PC CPU special relay is in the following state:
   a) M9050 (signal flow conversion contact).....OFF (A3CPU only)
   b) M9051 (CHG instruction disable).....ON

POINTS

(1) When reading or writing the timer/counter setting values using the sequence program read/write command, range designations of T0 to T255 or C0 to C255 are possible.
(2) Extended ranges of T256 to T2047 and C256 to C1023 for AnA CPU should be used for storing the setting values; read or write the set values using the batch read/write command for devices (D, W, R) allocated by parameter setting.

(b) Designating the head address

The division between sequence programs and T/C set values, and their addresses in 4-digit ASCII are shown in the table below.

Example:

To read the set values T0 to T63

Head address = FE00H Command = XR or MR

Sequence Program	Designated Step for Protocol
T0 set value T1 set value to T255 set value	FE00H FE01H to FEFFH
C0 set value C1 set value to C255 set value	FF00H FF01H to FFFFH
Step 0 Step 1 to Step 30718 (30K)	0000H 0001H to 77FEH

Calculation of designated step

Head step number in designating the timer set value ..... FE00H + timer device No. (expressed in hexadecimal)

Head step number in designating the counter set value ..... FF00H + counter device No. (expressed in hexadecimal)

Head step number in designating a sequence program ..... 0000H + step No. (expressed in hexadecimal)

(c) Meaning of T/C set values

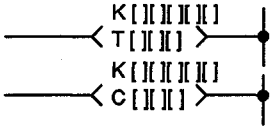
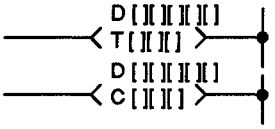
T/C set values are stored as hexadecimal values as shown in the table below.

When rewriting the PC CPU set values from the computer via the computer link module, designate the set value in 4-digit ASCII.

Example:

Data designated to change T10 setting value K10 to K20.....0014H

Data designated to change T11 setting value D30 to D10.....8014H

Ladder Example in Program	Setting in Program	Setting in Protocol
	K0 K1 to K9 K10 to K32767	0000H 0001H to 0009H 000AH to 7FFFH
	D0 D1 D2 to D1023	8000H 8002H 8004H to 87FEH

Calculation of protocol setting value

When the value is set in a constant (Kxxxxx) ..... 0000H + constant (expressed in hexadecimal)

When the value is set in a data register (Dyyyy) ..... 8000H + data register device No. x 2 (expressed in hexadecimal)

(2) Designating a program when an XR or XW command is used

To read or write a sequence program or T/C set values using an XR or XW command, designate the program with the corresponding selection number shown in the table below:

Program	Selection No.	PC CPU Concerned		
		PC CPU Incapable of Creating Subsequence Program *1	PC CPU Capable of Creating Subsequence Program (Except A4UCPU) *1	A4UCPU
Main sequence program	00H	O	O	O
Subsequence program (1)	01H	—	O	O
Subsequence program (2)	02H	—	X	O
Subsequence program (3)	03H	—	X	O

*1 For each PC CPU, refer to the PC CPU concerned shown in the "Functions available with the ACPU common commands" in Section 3.2.1 (1).

(3) Sequence program batch read

In this section, examples are quoted to describe the control protocols for batch-reading the data (machine language) or T/C set values from a sequence program using an XR, MR or SR command.

[Control protocol]

The control protocols for reading a sequence program (machine language) or T/C set values are all shown in control format 1.
To access a sequence program in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocols in this section.

POINTS

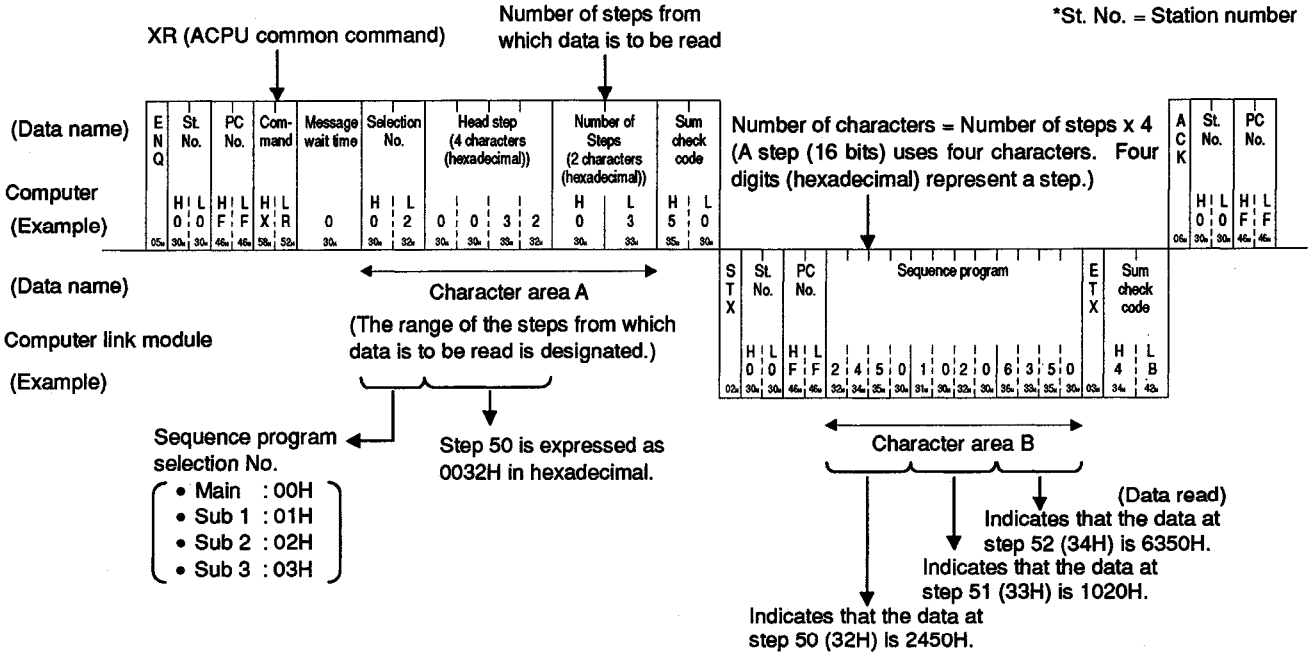
(1) The main program and its T/C set values can be read by the use of an XR or MR command.
(2) Subsequence program 1 and its T/C set values can be read by the use of an XR or SR command.
(3) The subprograms for the A4UCPU (sub 2, sub 3, sub 4) and their T/C set values can be read by the use of an XR command.
(4) An XR command can be used to read all sequence programs and their T/C set values.
(5) The number of steps must meet the following requirement:
- $1 \leq$ number of steps ≤ 64

(6) It is impossible to designate a T/C set value and the main sequence program together. Only either of them can be designated.

(a) Reading data from a sequence program using an XR command

(Reading conditions)

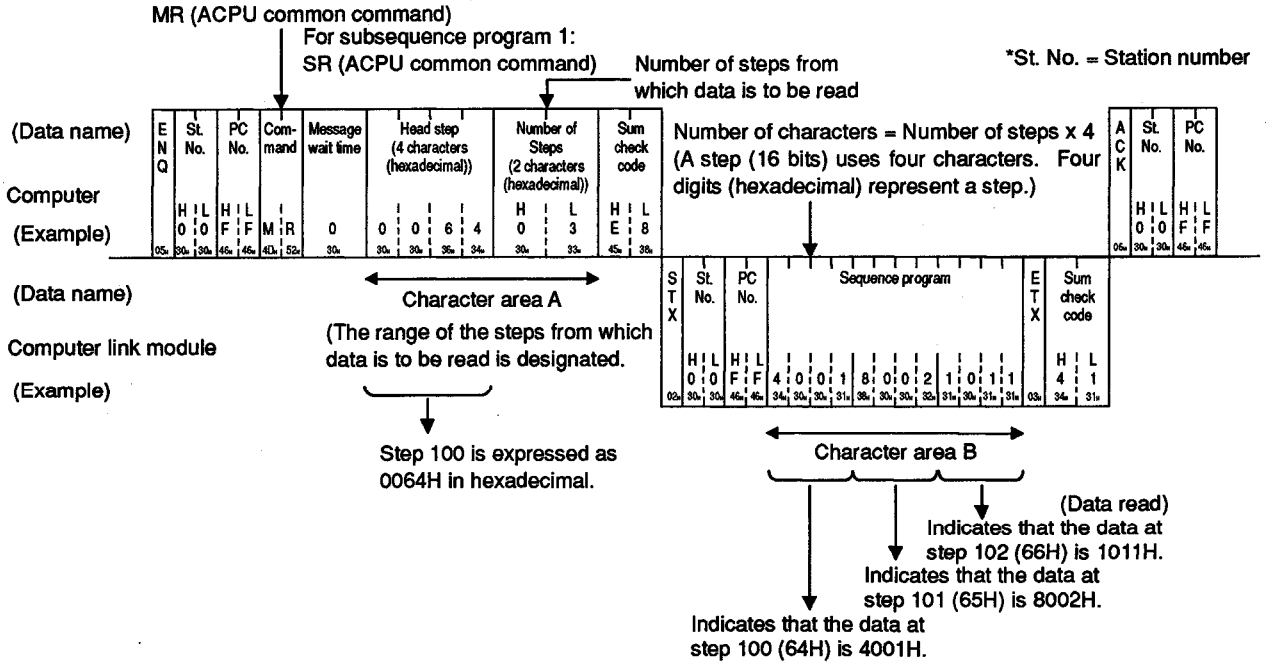
1) The message wait time is 0 ms.
2) Data at three steps, step 50 to step 52, in subsequence program 2 is to be read from the PC CPU (self station) loaded with a computer link module.



(b) Reading data from a sequence program using an MR or SR command

(Reading conditions)

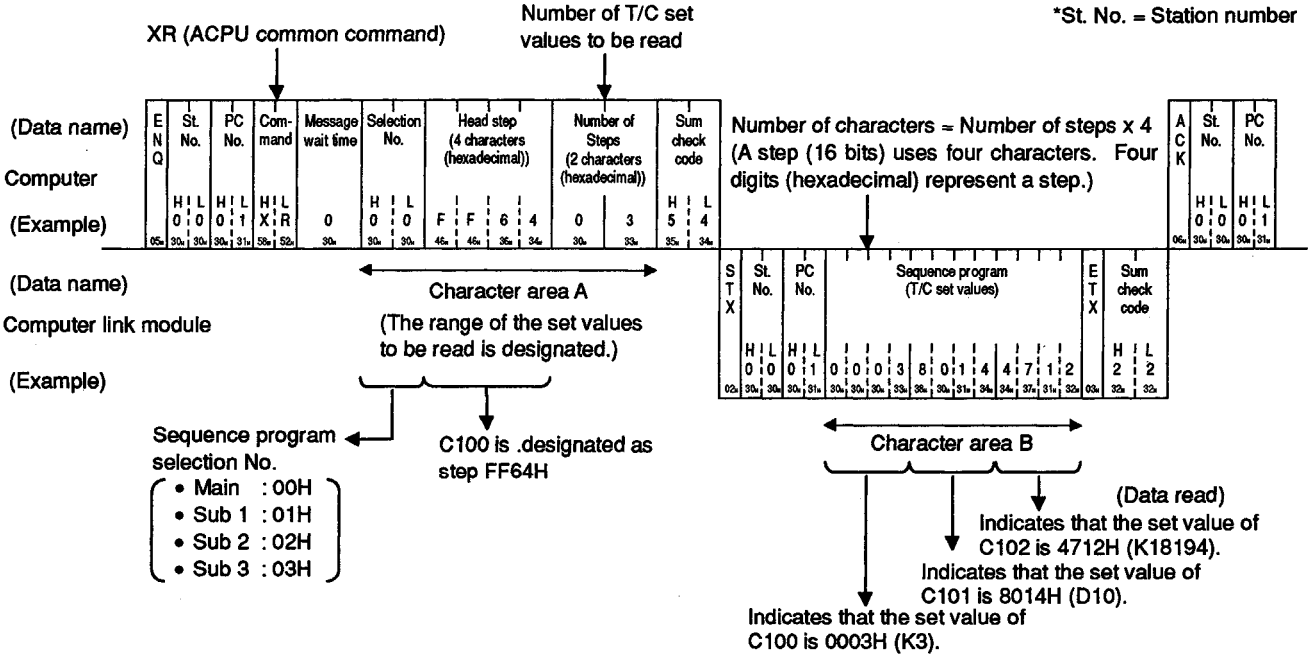
1) The message wait time is 0 ms.
2) Data at three steps, step 100 to step 102, in the main sequence program is to be read from the PC CPU (self station) loaded with a computer link module.



(c) Reading T/C set values using an XR command

(Reading conditions)

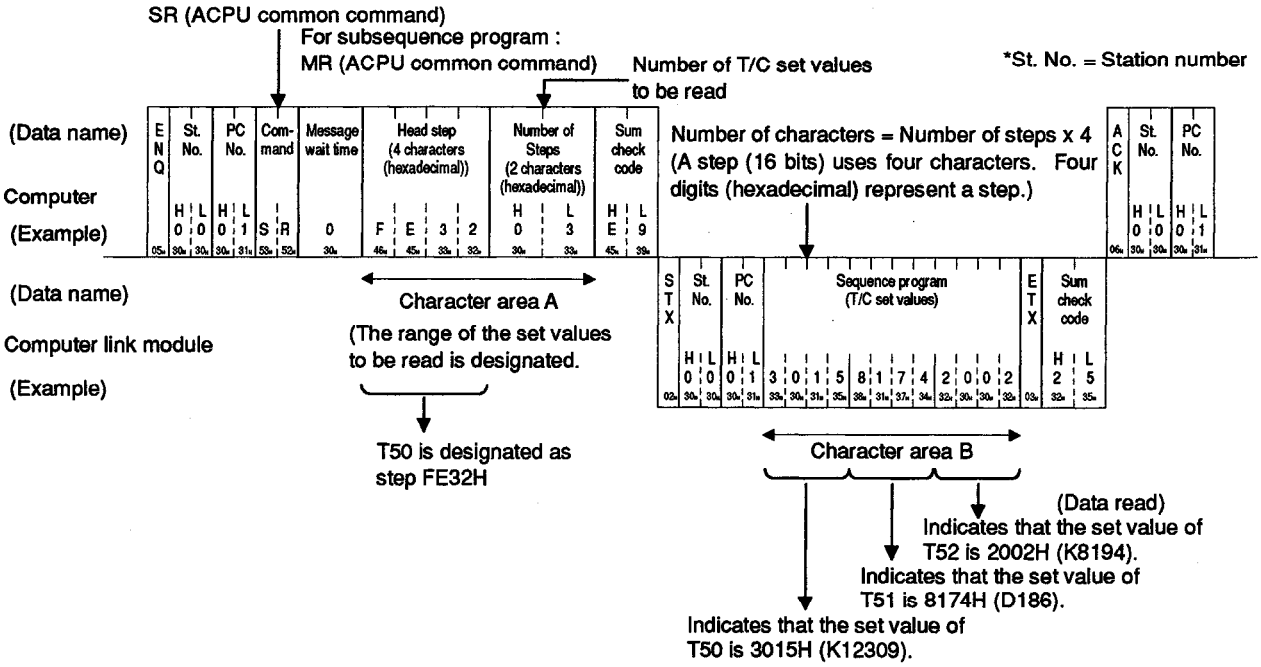
1) The message wait time is 0 ms.
2) The T/C set values of three counters, C100 to C102, for the main program are to be read from PC CPU No. 01 on the MELSECNET system.



(d) Reading T/C set values using an MR or SR command

(Reading conditions)

1) The message wait time is 0 ms.
2) The T/C set values of three timers, T50 to T52, for subsequence program 1 are to be read from PC CPU No. 01 on the MELSECNET system.



(4) Sequence program batch write

In this section, examples are quoted to describe the control protocols for batch-writing a sequence program or T/C set values using an XW, MW or SW command.

[Control protocol]

The control protocols for writing a sequence program (machine language) or T/C set values are all shown in control format 1.
To access a sequence program in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

POINTS

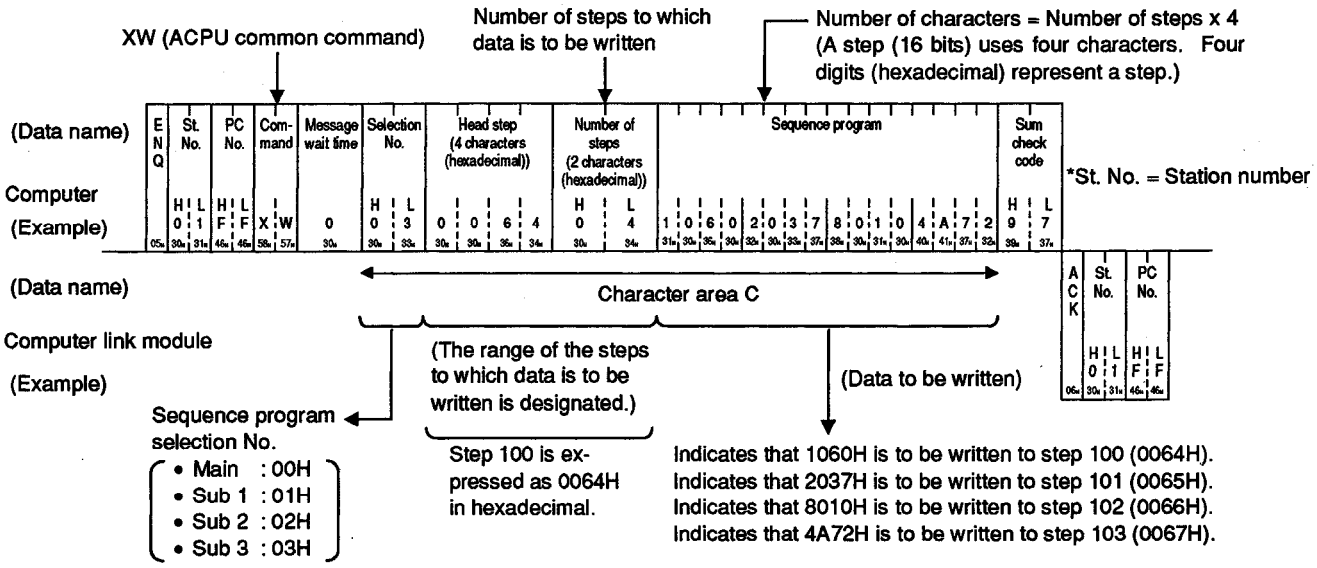
(1) The main program and its T/C set values can be written by the use of an XW or MW command.
(2) Subsequence program 1 and its T/C set values can be written by the use of an XW or SW command.
(3) The subsequence programs for the A4UCPU (sub 2, sub 3, sub 4) and their T/C set values can be written by the use of an XW command.
(4) An XW command can be used to write all sequence programs and T/C set values.
(5) The number of steps must meet the following requirement:
- $1 \leq$ number of steps ≤ 64

(6) It is impossible to designate a T/C set value and the main sequence program together. Only either of them can be designated.

(a) Writing data to a sequence program using an XW command

(Writing conditions)

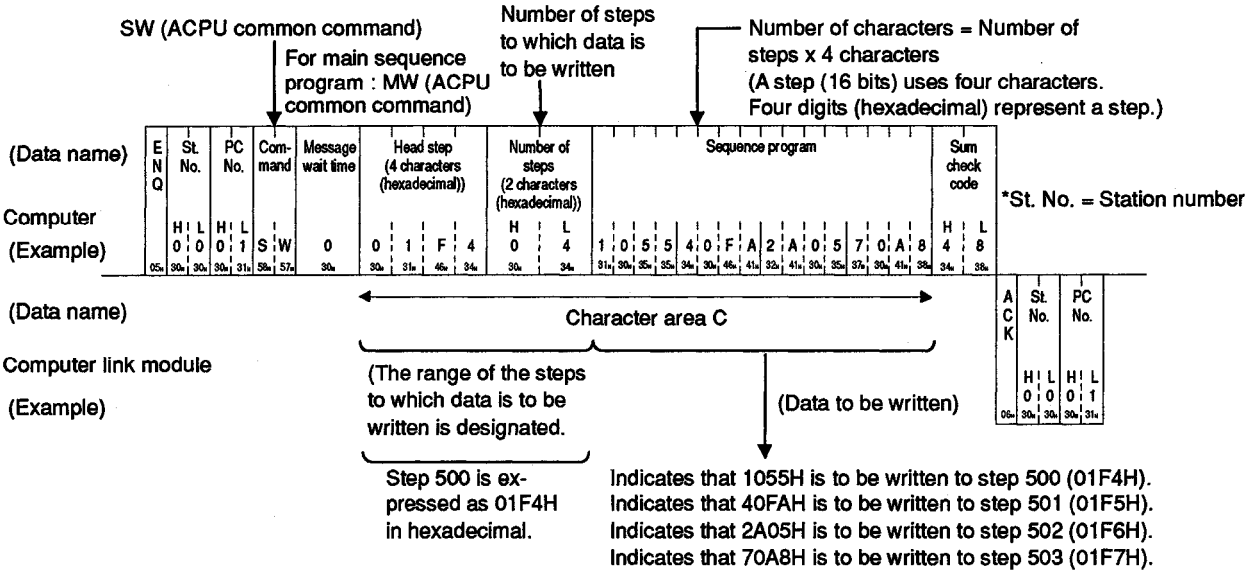
1) The message wait time is 0 ms.
2) Data is to be written to four steps, step 100 to step 103, in subsequence program 3 of PC CPU No. 01 on the MELSECNET system.



(b) Writing data to a sequence program using an MW or SW command

(Writing conditions)

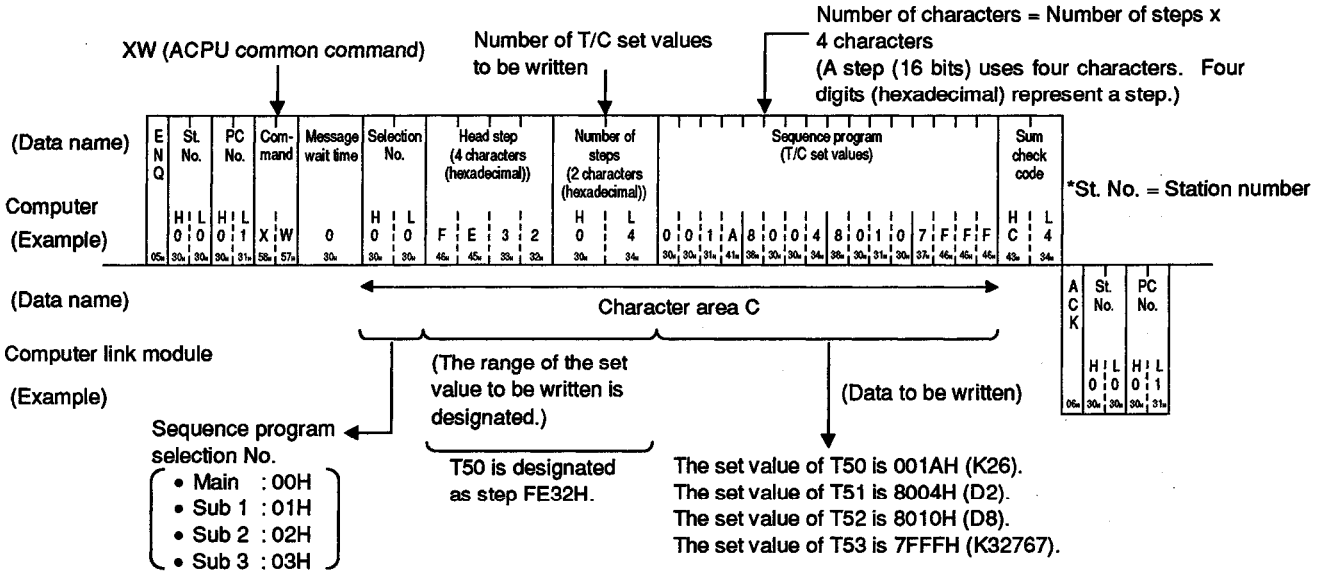
1) The message wait time is 0 ms.
2) Data is to be written to four steps, step 500 to step 503, in subsequence program 1 of PC CPU No. 01 on the MELSECNET system.



(c) Writing T/C set values using an XW command

(Writing conditions)

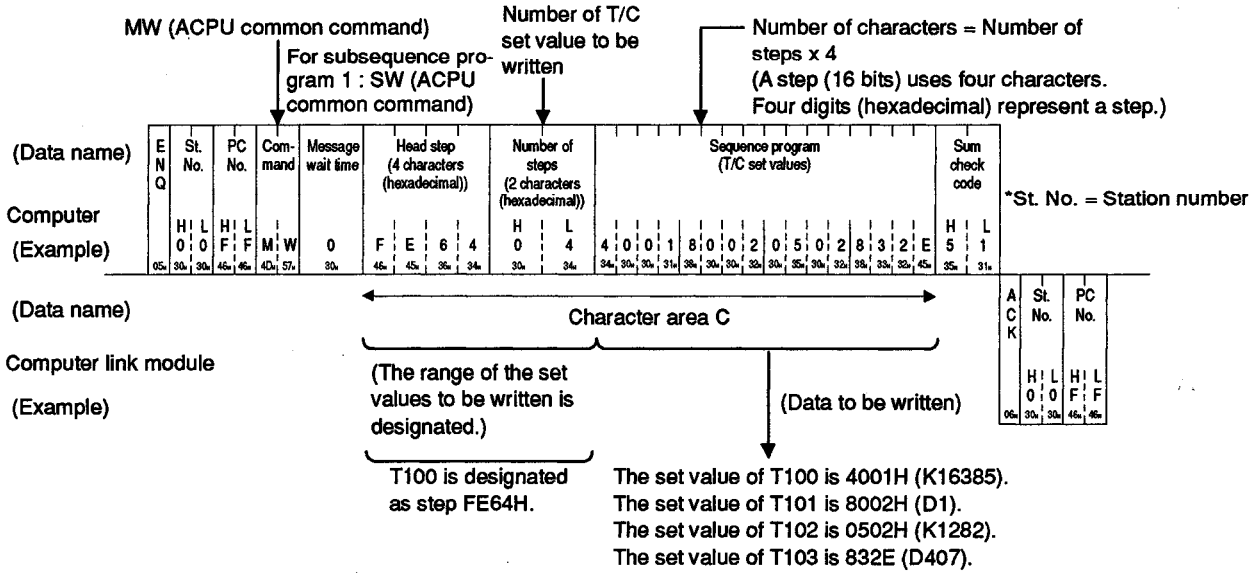
1) The message wait time is 0 ms.
2) The T/C set values of four timers, T50 to T53, for the main sequence program are to be written to PC CPU No. 01 on the MELSECNET system.



(d) Writing T/C set values using an MW or SW command

(Writing conditions)

1) The message wait time is 0 ms.
2) The T/C set values of four timers, T100 to T103, for the main sequence program are to be written to the PC CPU (self station) loaded with a computer link module.



5.12.5 Microcomputer program read/write

In this section, an example is quoted to describe the control protocol for reading or writing microcomputer programs for the PC CPU.

(1) Commands and addresses

Commands and program addresses to read and write microcomputer programs are explained below:

(a) ACPU common commands

Item		Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
		Symbol	ASCII Code			During STOP	During RUN		
							Write Enabled	Write Disabled	
Batch read	Main	UR	55H, 52H	Reads microcomputer main programs.	128 bytes	O	O	O	5.12.5 (2)
	Sub	VR	56H, 52H	Reads microcomputer subprograms.					
Batch write	Main	UW	55H, 57H	Writes microcomputer main programs.	128 bytes	O	O*	X	5.12.5 (3)
	Sub	VW	56H, 57H	Writes microcomputer subprograms.					

Note : O..... Executable
X..... Not executable

* Writing during a program run may be executed if all the following conditions are met:

1) The PC CPU is A3, A3N, A3H, A3M or A73.
2) The program is not currently running program (indicates a subprogram called by the main program, if the main program is being run).
3) The PC CPU special relay is in the following state:
   a) M9050 (signal flow conversion contact) ..... OFF (A3CPU only)
   b) M9051 (CHG instruction disable) ..... ON

POINT

When the PC CPU is an AnA or AnUCPU, the SFC program is read or written by the main microcomputer read/write function. (It cannot be written while the PC CPU is running.)
The SFC program must be read or written within the microcomputer program capacity and address ranges specified in (1) (b).
The microcomputer program capacity is the one set in making the memory capacity setting for the GPP function or on the SFC area capacity setting screen, an MELSAP-II function.

(b) Microcomputer program address

Microcomputer addresses are designated in the protocol as follows:

1) The range of addresses that can be set for each PC CPU is shown in the table on the next page.

CPU Model	Microcomputer Program Capacity	Microcomputer Program Addresses
A1SCPU(S1) A1SJCPU A1SHCPU A1SJHCPU A0J2HCPU A2CCPU A2CJCPU	Max. 14k bytes	0000H to 37FEH
A1CPU A1NCPU	Max. 10k bytes	0000H to 27FEH
A2SCPU(S1) A2ASCPU(S1) A2SHCPU(S1) A2CPU(S1) A2NCPU(S1) A2ACPU(S1) A2UCPU(S1)	Max. 26k bytes	0000H to 67FEH
A3CPU A3NCPU A3HCPU A3MCPU A3ACPU A3UCPU A4UCPU A73CPU	Main and sub Max. 58k bytes	0000H to E7FEH

2) Addresses are set by converting 4-digit hexadecimals into ASCII.

3) A character area error 06H occurs if the following condition is not met:

Head address + (number of bytes) − 1 ≤ microcomputer program capacity.

(2) Microcomputer program batch read

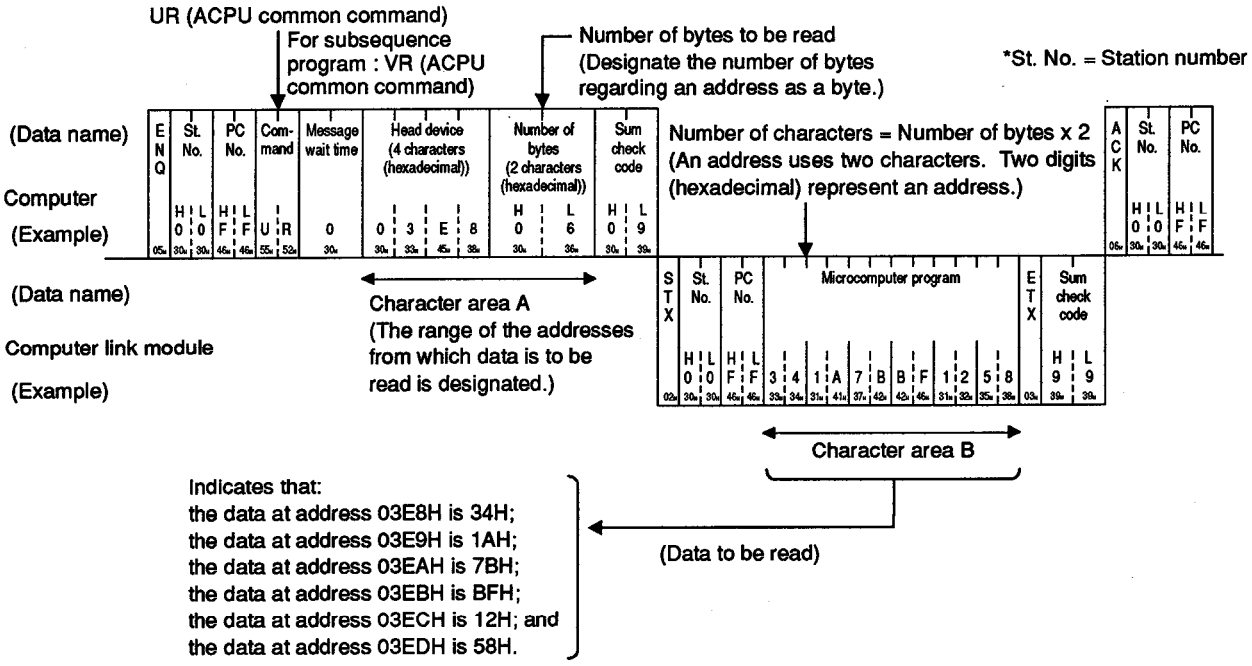
In this section, an example is quoted to describe the control protocol for batch-reading data from a microcomputer program using a UR or VR command.

[Control protocol]

The protocol shown below is in control format 1.
To access a microcomputer program in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)

1) The message wait time is 0 ms.
2) Data (six bytes) is to be read from six addresses, 03E8H to 03EDH, in the microcomputer program for the main sequence program of the PC CPU (selfs tation) loaded with a computer link module.



POINT

To designate the number of bytes, the following conditions must be met:

- 1 ≤ number of bytes ≤ 128
- Head address + number of bytes - 1 ≤ microcomputer program capacity

(3) Microcomputer program batch write

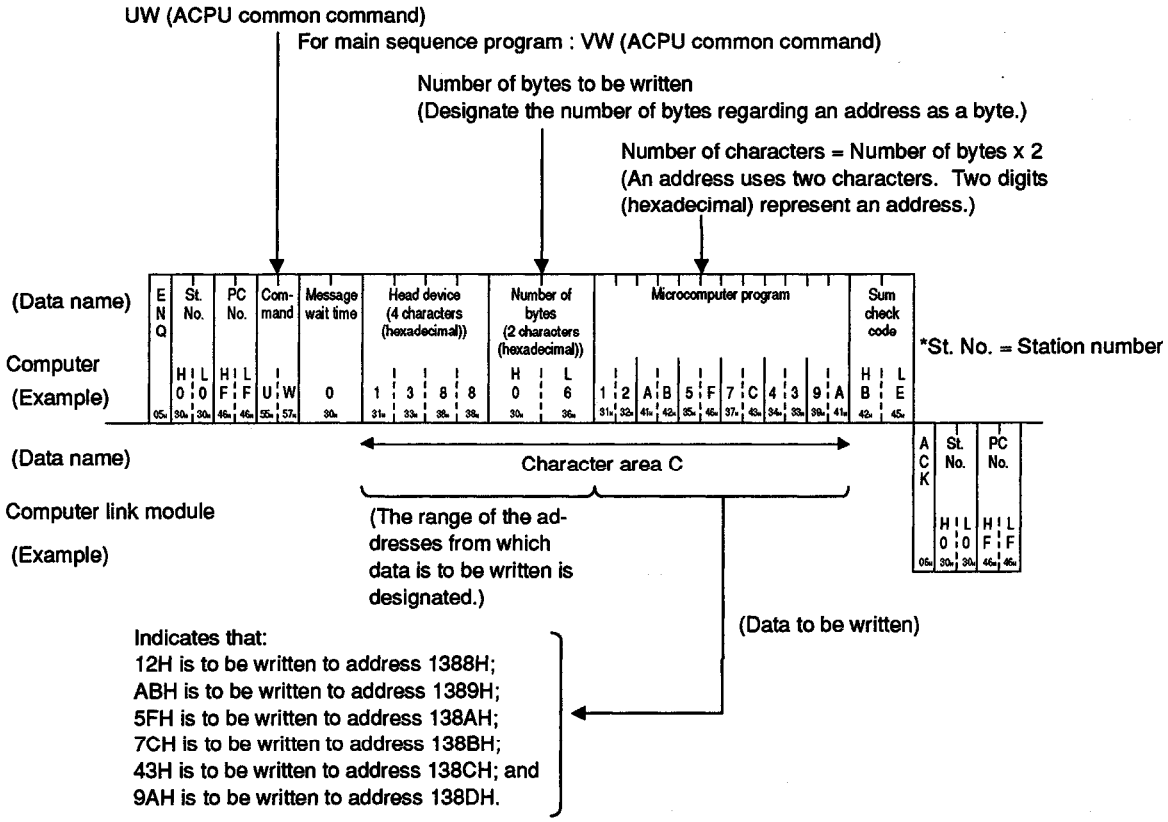
In this section, an example is quoted to describe the control protocol for batch-reading data from a microcomputer program using a UW or VW command.

[Control protocol]

The protocol shown below is in control format 1.
To access a microcomputer program in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)

1) The message wait time is 0 ms.
2) Data (six bytes) is to be read from six addresses, 1388H to 138DH, in the microcomputer program for the main sequence program of the PC CPU (self station) loaded with a computer link module.1. (The range of the addresses to which data is to be written is designated.)



POINT

To designate the number of bytes, the following conditions must be met:

- 1 ≤ number of bytes ≤ 128
- Head address + number of bytes - 1 ≤ microcomputer program capacity

5.12.6 Comment memory read/write

In this section, examples are quoted to describe the control protocols for reading and writing comment data from or to the PC CPU.

(1) Commands and addresses

Commands and comment data addresses to read and write comment data are explained below.

(a) ACPU common commands

Item	Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
	Symbol	ASCII Code			During STOP	During RUN		
						Write Enabled	Write Disabled	
Batch read	KR	4BH, 52H	Reads from comment memory.	128 bytes	O	O	O	5.12.6(2)
Batch write	KW	4BH, 57H	Writes to comment memory.	128 bytes	O	O	X	5.12.6(3)

Note : O .....Executable
X.....Not executable

(b) Comment memory addresses

The area to store comment data is managed using relative addresses from the head address 00H.

For example, for 2k bytes of parameter comments, the range in which the addresses may be specified for the head address is 00H to 7FFH.

1) Comment memory capacity is 64k bytes

The comment data address range is determined by the parameter setting.

2) Comment memory addresses are set by converting 4-digit hexadecimals into ASCII. (0000 to FFFF)

3) A character area error 06H occurs if the following condition is not met:

Head address + designated number of bytes ≤ comment memory capacity.

POINT

It is not possible to designate a particular device or device number when reading or writing comment data.
Always read or write all data from address 0H.

(2) Comment memory batch read

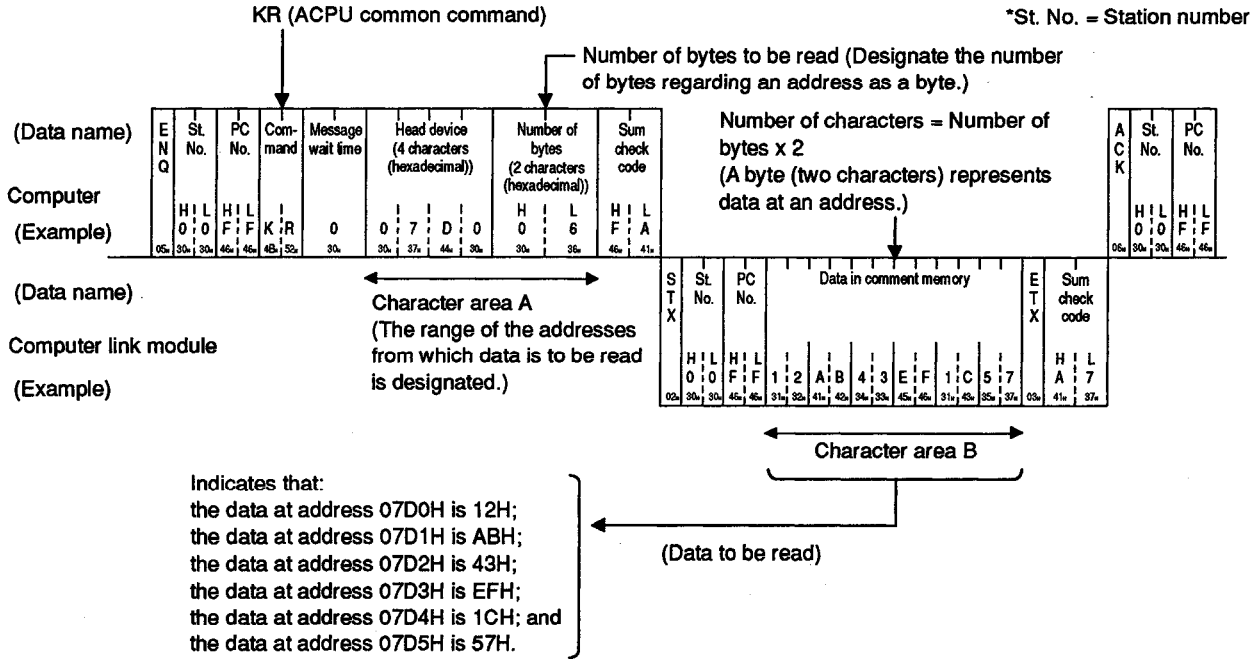
In this section, an example is quoted to describe the control protocol for batch-reading comment data from comment memory using a KR command.

[Control protocol]

The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)

1) The message wait time is 0 ms.
2) Comment data (six bytes) is to be read from six addresses, 07D0H to 07D5H, in the comment memory in the PC CPU (self station) loaded with a computer link module.



POINT

To designate the number of bytes, the following conditions must be met:

- 1 ≤ number of bytes ≤ 128
- (Head address) + [(number of bytes) – 1] ≤ comment memory capacity

(3) Comment memory batch write

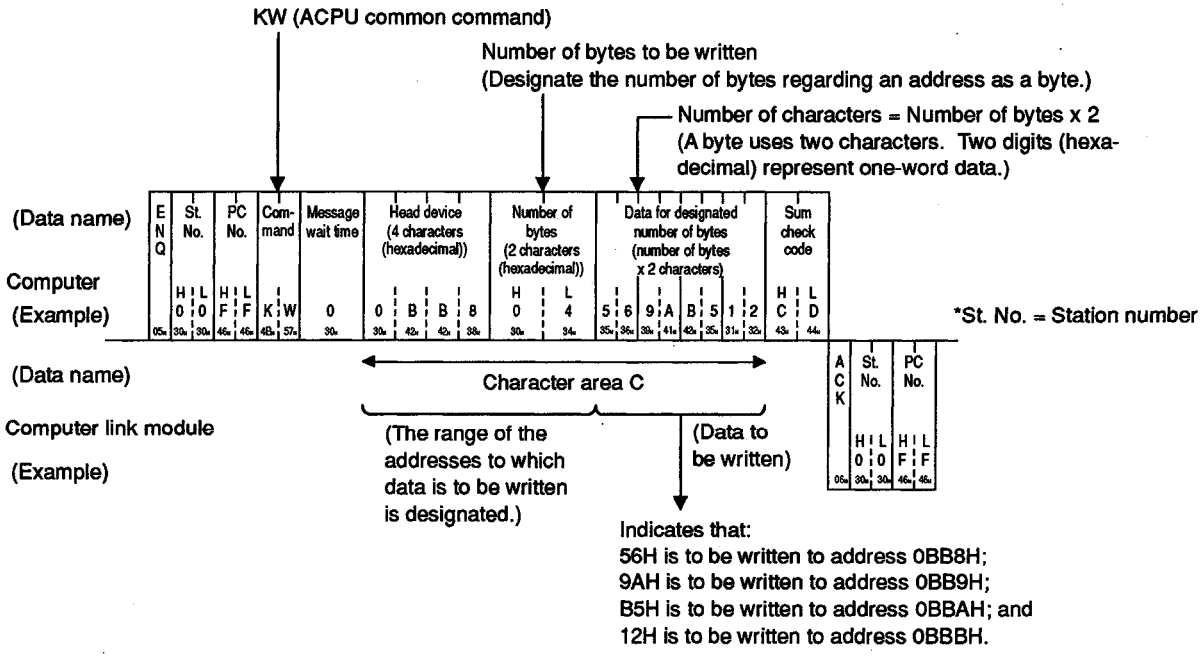
In this section, an example is used to describe the control protocol for batch-writing data to comment memory using a KW command.

[Control protocol]

The protocol shown below is in control format 1.
To access comment memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)

1) The message wait time is 0 ms.
2) Data (four bytes) is to be written to four addresses, 0BB8H to 0BBBH, in the comment memory in the PC CPU (self station) loaded with a computer link module.



POINT

To designate the number of bytes, the following conditions must be met:

- 1 ≤ number of bytes ≤ 128
- (Head address) + [(number of bytes) – 1] ≤ comment memory capacity

5.12.7 Extension comment memory read/write

In this section, examples are quoted to describe the control protocols for reading and writing data (extension comment 1) to and from the extension comment memory in the PC CPU.

(1) Commands and addresses

The following are the comments to be used to read and write extension comment data, and the extension comment data addresses.

(a) AnA/AnUCPU common commands

Item	Command		Description	Number of Points Processed per Communication	PC CPU State			Reference Section
	Symbol	ASCII Code			During STOP	During RUN		
						Write Enabled	Write Disabled	
Batch read	DR	44H, 52H	Reads from the extension comment memory.	128 bytes	O	O	O	5.12.7(2)
Batch write	DW	44H, 57H	Writes to the extension comment memory.	128 bytes	O	O	X	5.12.7(3)

Note : O .....Executable
X.....Not executable

(b) Extension comment memory addresses

The extension comment data storage area is managed in relative addresses with the head address 00H.

For example, the range that can be set to the head address for an extension comment memory of 3k bytes is 00H to BFFH.

1) The maximum extension comment memory area is 63k bytes.

The address range for the extension comment data is determined in accordance with the paraemter set capacity.

2) Designation of the extension comment memory address is made by converting 5-digit hexadecimal into ASCII code (00000 to 0FBFF).

3) A character error "06H" occurs if the extension comment memory capacity is not equal to or greater than [head address + (set number of bytes – 1)].

POINT

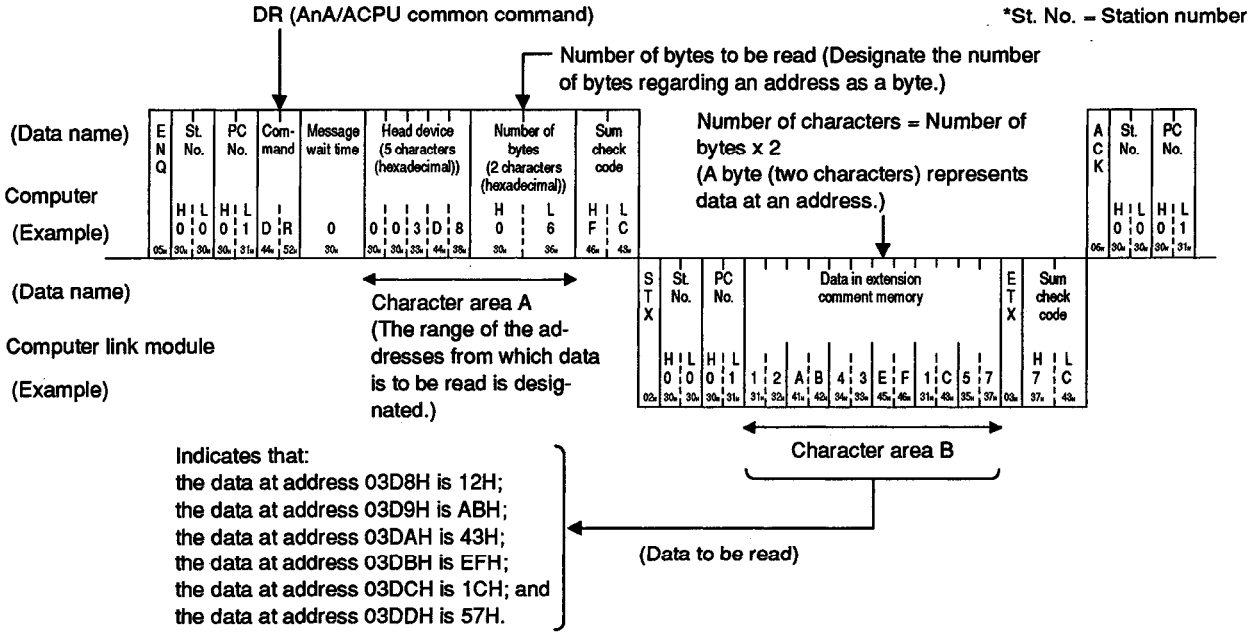
Reading or writing extension comment data by designating specific devices or device numbers is not possible.
Always read or write extension comment data beginning with address 0H.

(2) Extension comment memory batch read

In this section, an example is quoted to describe the control protocol for batch-reading extension comment data from extension comment memory using a DR command.

[Control protocol] The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Reading conditions)
1) The message wait time is 0 ms.
2) Extension comment data (six bytes) is to be read from six addresses, 03D8H to 03DDH, in the extension comment memory in PC CPU No. 1 on the MELSECNET system.



POINT

To designate the number of bytes, the following conditions must be met:
- 1 ≤ number of bytes ≤ 128
- (Head address) + [(number of bytes) – 1] ≤ extension coment memory capacity

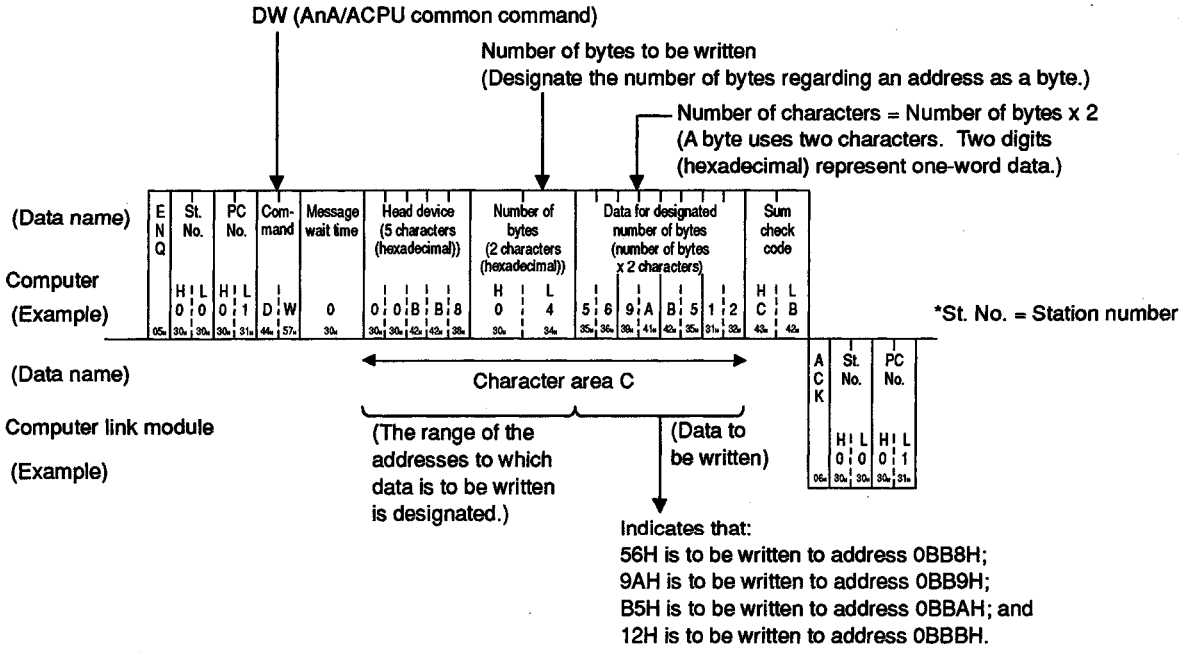
(3) Extension comment memory batch write

In this section, an example is quoted to describe the control protocol for batch-writing extension comment data to extension comment memory using a DW command.

[Control protocol] The protocol shown below is in control format 1.

To access extension comment memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Writing conditions)
1) The message wait time is 0 ms.
2) Extension comment data (four bytes) is to be written to four addresses, 0BB8H to 0BBBH, in the extension comment memory in PC CPU No. 1 on the MELSECNET system.



POINT

To set the number of bytes, the following conditions must be met:
- 1 ≤ number of bytes ≤ 128
- (Head address) + [(number of bytes) – 1] ≤ extension comment memory capacity

5.13 Global Function

The global function is used to switch the Xn2 input signal at each computer link module in all stations connected to the computer by the multidrop link (refer to Section 3.9).

This function is used for emergency instructions simultaneous start, etc., to the PC CPU.

In this section, examples are quoted to describe the control protocols when the global function is used.

5.13.1 Commands and control

(1) ACPU common commands

Item	Command		Description	PC CPU State			Reference Section
	Sym-bol	ASCII Code		During STOP	During RUN		
					Write Enabled	Write Disabled	
Global	GW	47H, 57H	Turns ON/OFF Xn2 of the computer link module loaded in each PC CPU system.	O	O	O	5.13.2

Note : O..... Executable

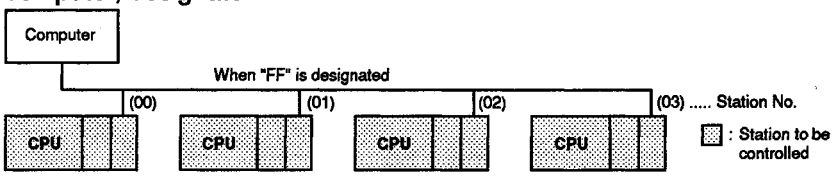
(2) Control

This function switches the Xn2 input signal at each computer link module in all stations linked to the computer.

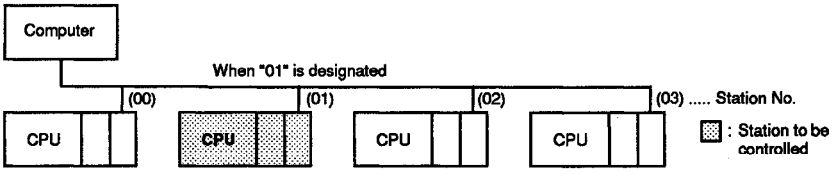
(a) The "n" in Xn2 is the number determined by the I/O signal at the computer link module loaded in the PC CPU.
(Example: When the I/O signal at the computer link module is between 90H and AFH, Xn2 is X92.)

(b) Designate "FF" or a number between "00" and "1F" for the station number for the control protocol.

1) To turn on the Xn2 at all computer link modules connected to the computer, designate "FF".



2) To turn on the Xn2 at a computer link module connected to the computer, designate the station number of the computer link module ("00" to "1F").



(c) The computer link modules do not respond to any instruction sent by this function from the computer.

(d) When the power supply to the PC CPU is turned off or when resetting or mode switching is made, the Xn2 will be turned off.

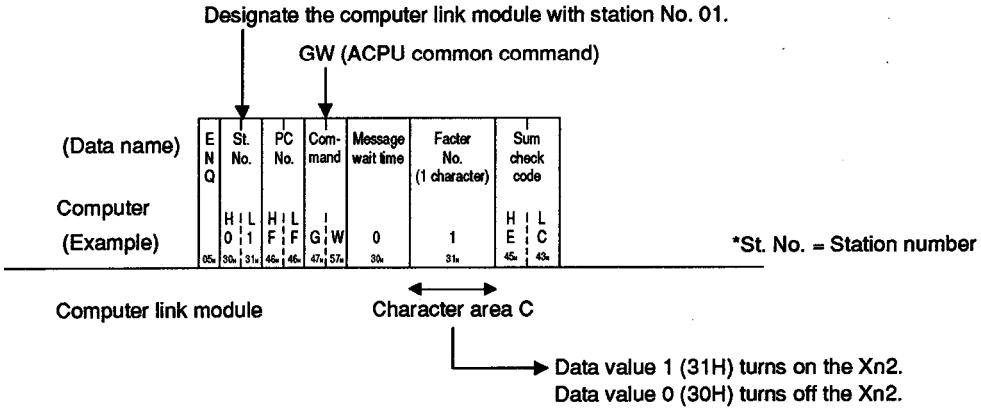
5.13.2 Control protocols for the global function

In this section, examples are quoted to describe the control protocols for turning on or off the Xn2 at the computer link module from the computer using a GW command.

[Control protocol] The protocol shown below is in control format 1.
To access device memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3 or 5.4.5 as well as by reference to the protocol in this section.

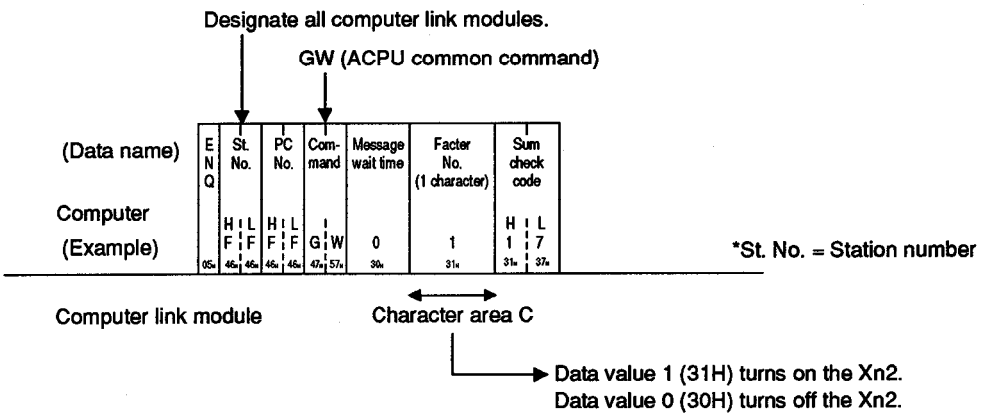
(1) To turn on the input signal at a station

(Input signal turning on conditions)
1) The message wait time is 0 ms.
2) The input signal (Xn2) at the PC CPU loaded with a computer link module (station No. 01) is to be turned on.



(2) To turn on the input signal at all multidrop-linked stations

(Input signal turning on conditions)
1) The message wait time is 0 ms.
2) The input signal (Xn2) at all multidrop-linked PC CPUs loaded with a computer link module is to be turned on.

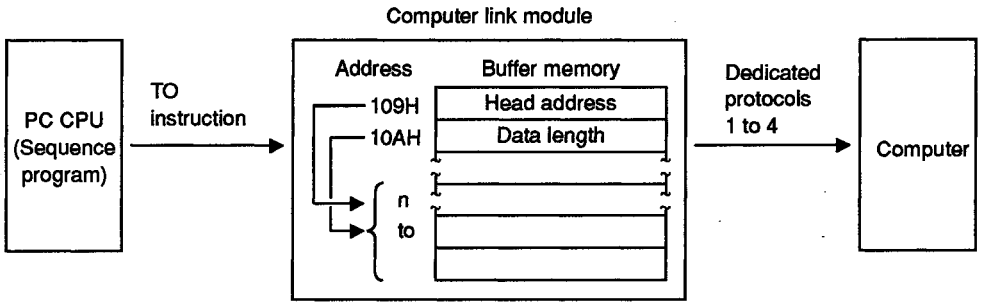


5.14 On-Demand Function

The on-demand function is used when the PC CPU has data to transmit to the computer. In this case, the PC CPU specifies the buffer memory area in which the data to be transmitted is stored and then starts transmission.

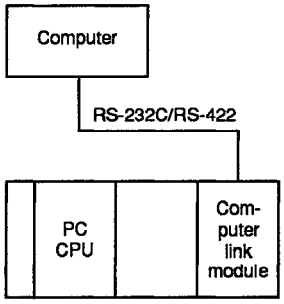
During data transmission between the computer and PC CPU using dedicated protocols 1 to 4, communications is normally initiated by the computer.

If the PC CPU has emergency data to transmit to the computer, the on-demand function is used.



POINT

This on-demand function is available when the computer-to-PC-CPU ratio is 1:1.



Do not use the on-demand function when the system configuration is not 1:1. If it is used when the ratio between the computer and the PC CPU, which are multidrop-linked with each other, is 1:n, 2:n or m:n, communications data in control formats 1 to 4 or data sent on demand is destroyed, or data cannot be sent normally.

5.14.1 On-demand handshake I/O signal and buffer memory

(1) On-demand handshake I/O signal

The on-demand I/O handshake signal turns ON when the PC CPU transmits a data send request to the computer to start transmission, and turns OFF when transmission of the data specified by the computer link module is completed. It acts as an interlock to prevent on-demand requests from being made simultaneously.

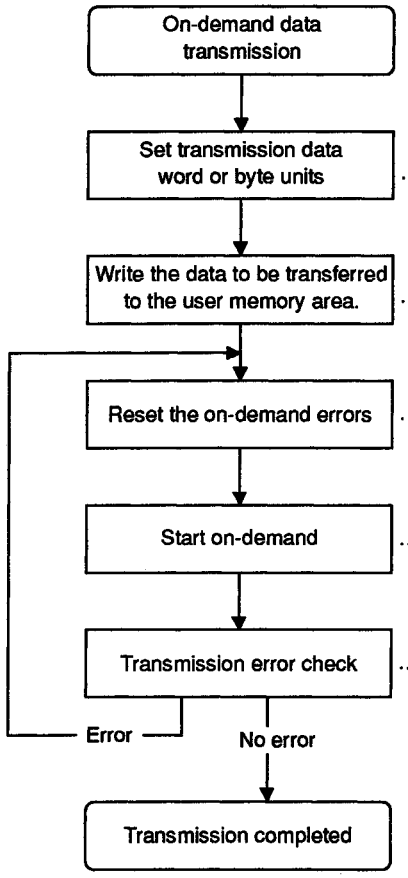
Handshake Signal	Description	Signal Turned ON/OFF by
Xn3	During execution of on-demand function ON : transmission underway OFF : transmission completed	Computer link module

(2) Buffer memory used by the on-demand function

Address	Name	Description
109H	Area to specify head address in on-demand buffer memory	The head address of the data stored in the buffer memory to be transmitted by the on-demand function is specified by the TO instruction of the Sequence program.
10AH	Area to specify data length	The length of the data to be transmitted by the on-demand function is specified by the PC CPU TO instruction of the sequence program.
10CH	On-demand error storage area	The computer link module writes a "1" to this address if a transmission error occurs during on-demand data transmission. 0 : No error 1 : Error

5.14.2 On-demand function execution procedure

(1) PC CPU execution procedure



Check (and if necessary change) the contents of buffer memory address 103H. Refer to Examples and 2 in Section 5.14.3 for transmission data formats when word or byte units are selected.

Write to buffer memory address 0H to FFH or 120H to DFFH (*1)
The special use areas, address 100H to address 1FFH, in the buffer memory are unusable.

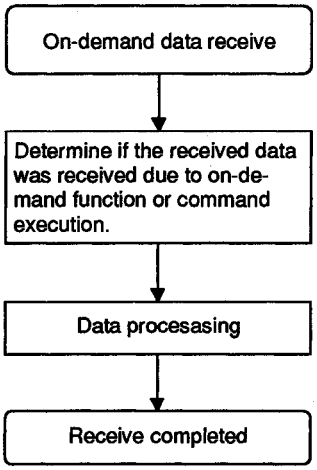
Clear the on-demand buffer memory address 10CH. (Start is disabled if the contents of this address are "1".)

The function starts by writing the data storage head address to buffer memory address 109H and the send data length to address 10AH (the Xn3 is turned on).

As soon as the Xn3 is turned off, data transmission is terminated.
If "1" is written to buffer memory address 10CH, an error occurs and the data is not transmitted

*1
When executing data communications simultaneously using the following functions, be careful that the area does not overlap with those for the functions:
- Reading or writing data from or to buffer memory using a dedicated protocol (Refer to Section 5.9.)
- Data communications in the no-protocol and bidirectional modes (Refer to Sections 6 and 7.)

(2) Computer execution procedure

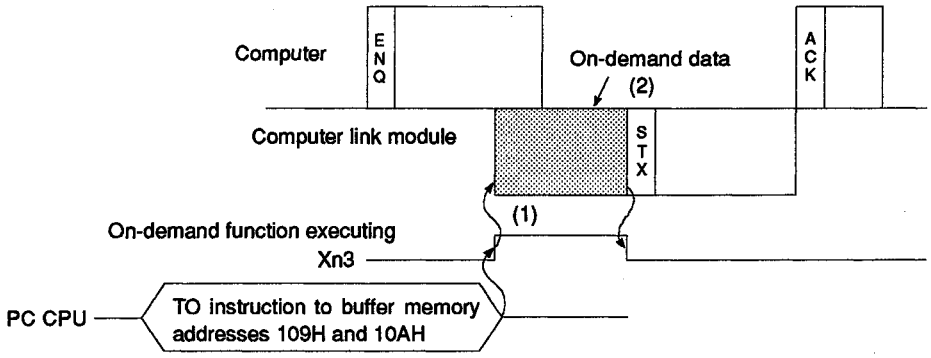


The computer link module appends "FE" as the PC CPU number.
Process data as on-demand data only if the PC CPU number of the received data is FE.

(3) On-demand request processing timing chart

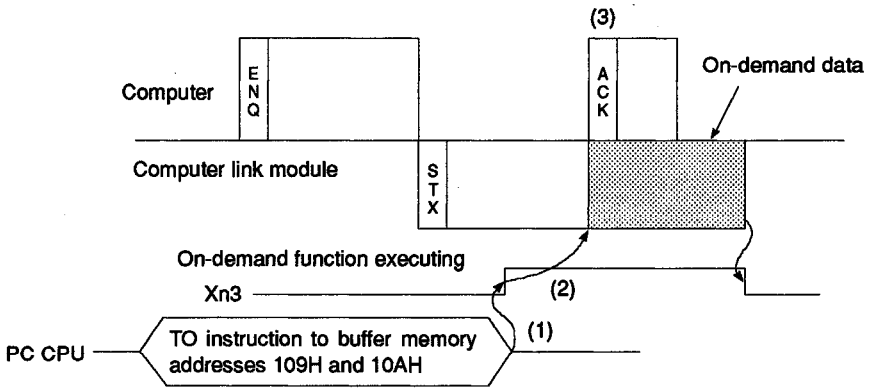
(a) Full-duplex communications

Computer is transmitting data



1) The on-demand function executing signal (Xn3) turns ON immediately and , the on-demand data is transmitted when the on-demand request is made.
2) Transmission of response data (beginning with STX) to the command data (beginning with ENQ) is suspended until the completion of on-demand data transmission.

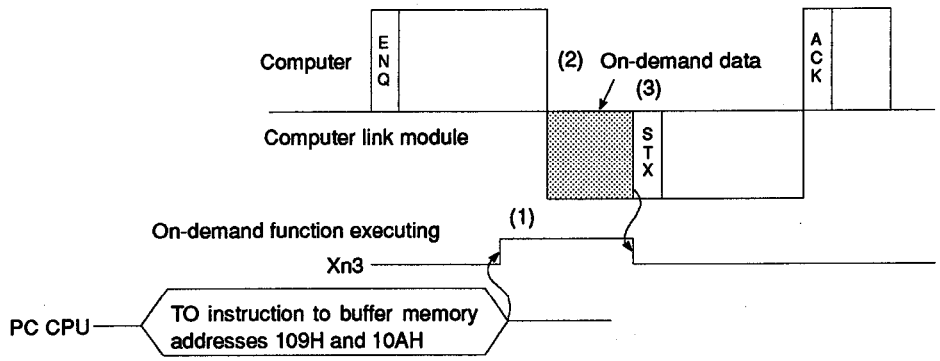
Computer is receiving data



1) The on-demand function executing signal (Xn3) turns ON immediately when the on-demand request is made.
2) Transmission of the on-demand data is suspended until the completion of the response data (beginning with STX) to the command data (beginning with ENQ).
3) Transmission of the response data (beginning with ACK) from the computer in response to the response data (beginning with STX) from the computer link module is possible while the on-demand data is received.

(b) Half-duplex communications ..... Refer to Section 10.

Computer is transmitting data

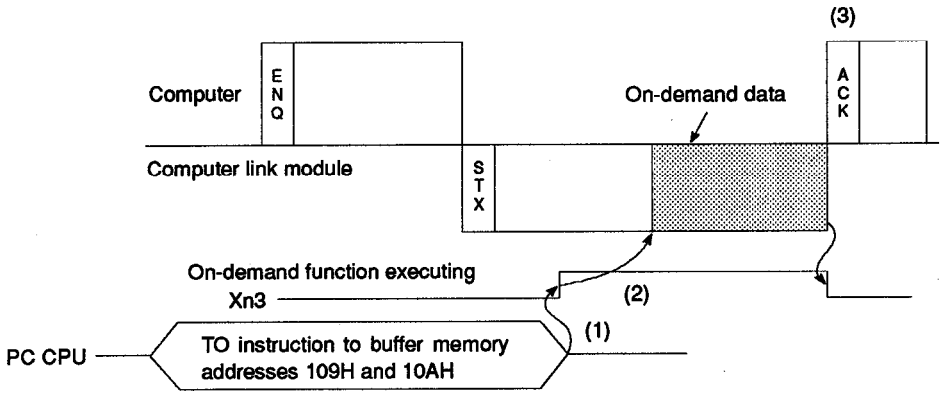


1) The on-demand function executing signal (Xn3) turns on immediately when the on-demand request is made.

2) Transmission of on-demand data is suspended until the completion of command data receive (beginning with ENQ) from the computer.

3) Transmission of response data (beginning with STX) to the command data (beginning with ENQ) is suspended until the completion of on-demand data transmission.

Computer is receiving data



1) The on-demand function executing signal (Xn3) turns ON immediately when the on-demand request is made.

2) Transmission of the on-demand data is suspended unil the completion of the response data (beginning with STX) to the command data (beginning with ENQ).

3) Transmission of the response data (beginning with ACK) from the computer in response to the response data (beginning with STX) from the computer link module should be made after the completion of on-demand data receive.

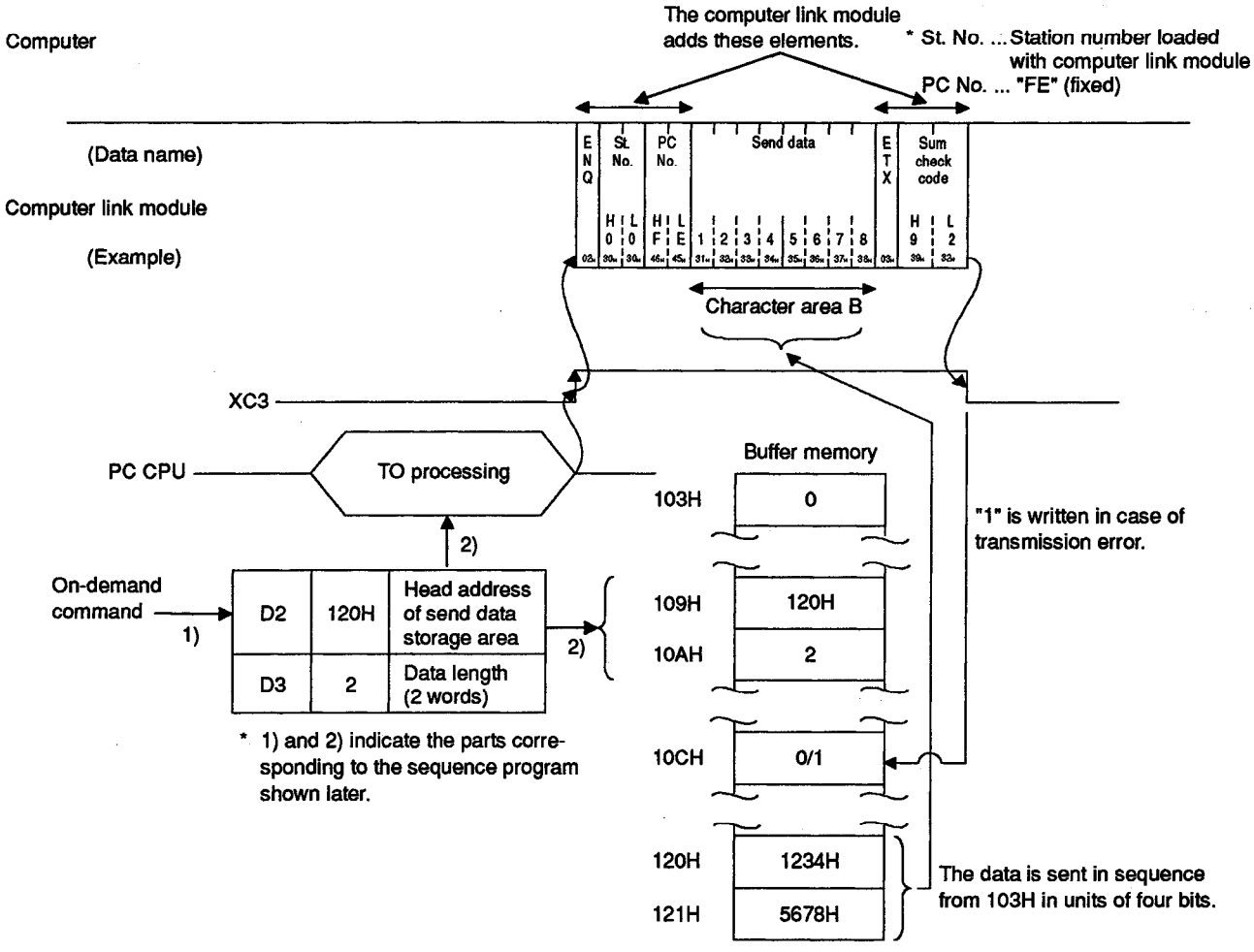
5.14.3 Control protocols for the on-demand function

In this section, examples are quoted to describe the control protocols for sending data from the PC CPU to the computer using a dedicated protocol.

[Control protocol] The protocols shown below are in control format 1.
To access buffer memory in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocols in this section.

(1) When "units of words" is set at computer link module buffer memory address 103H "area to specify word or byte units in no-protocol mode"

(Transmission conditions)
1) Data written at computer link module buffer memory addresses 120H and 121H is to be sent.
2) The I/O addresses in the computer link module are C0H to DFH.



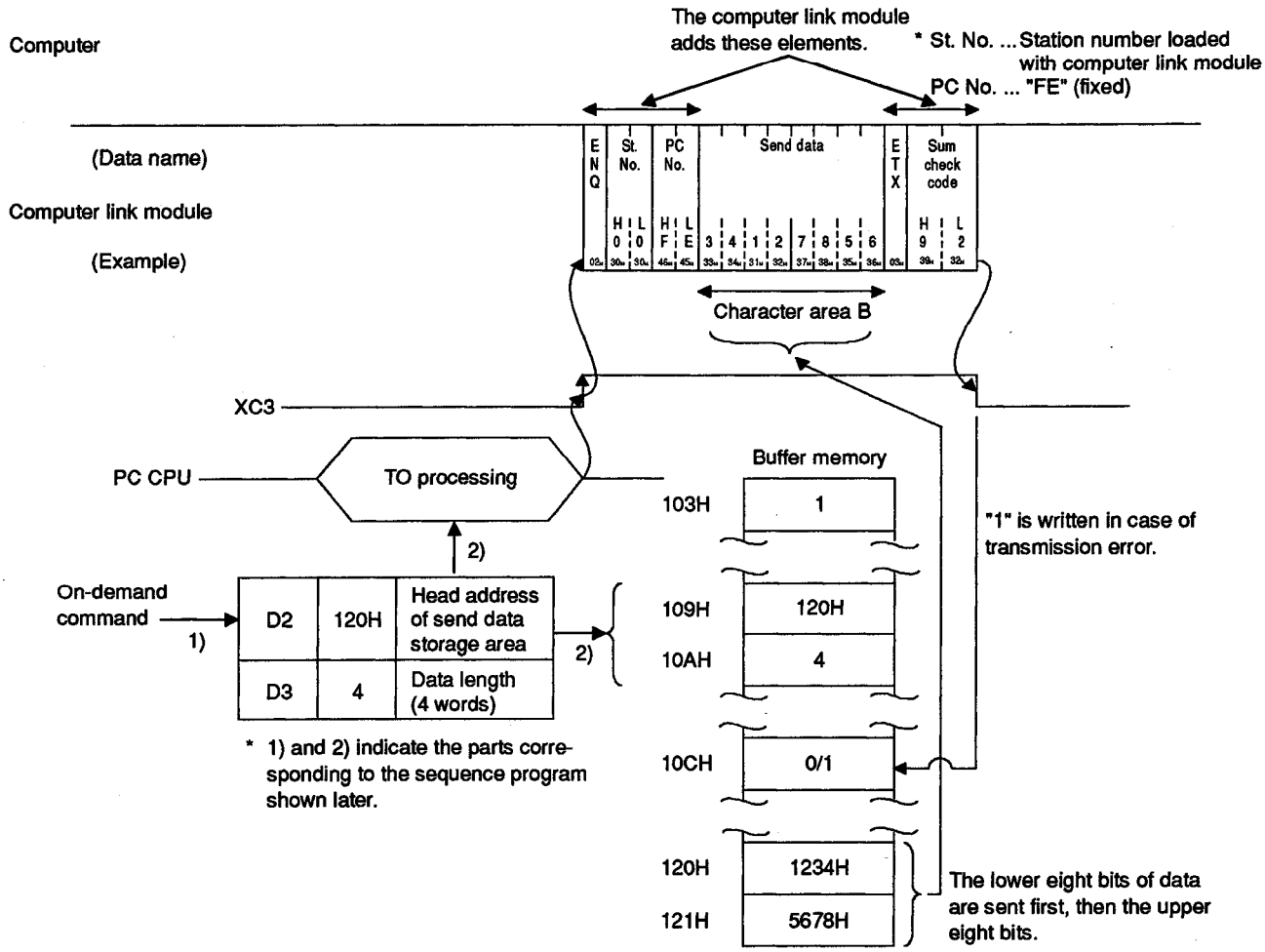
POINTS

(1) When the data is sent in control format 2, the block number is 00H.
(2) Number of sent data characters = data length x 4 characters
(One-word data uses four characters. Four digits (hexadecimal) represent one-word data.)

(2) When "units of bytes" is set at computer link module buffer memory address 103H "area to specify word or byte units in no-protocol mode"

(Transmission conditions)

1) Data written to computer link module buffer memory addresses 120H and 121H is to be sent.
2) The I/O addresses in the computer link module are C0H to DFH.

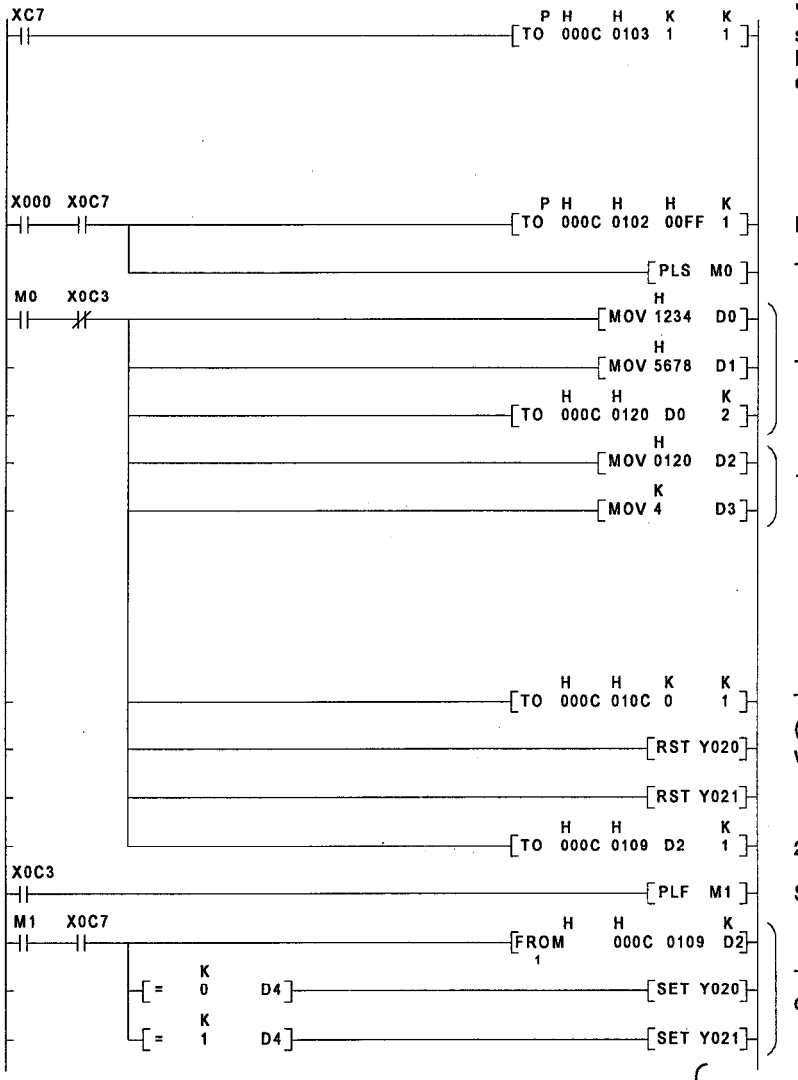


POINTS

(1) When the data is sent in control format 2, the block number is 00H.
(2) Number of sent data characters = data length x 2 characters
(One-byte data uses two characters. Two digits (hexadecimal) represent one-byte data.)
(3) When the value of data length is odd, the lower byte of data (bit 0 to bit 7) at the last designated address in the buffer memory is sent.

(3) Example of sequence program for the on-demand function

Shown below is an example of sequence program for data transmission described in (1) and (2) on the previous pages using the on-demand function.



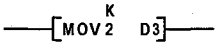
"Units of bytes" should be set for data transmission.
No setting is necessary when sending data in units of words.

Request to turn off the error indicator LED

The start-up command is converted to PLS.

The data to be sent is set.

1) The head address of the send data storage area and the data length are designated.
When the data is to be sent in units of words, set as follows:



The on-demand error is reset.
(The on-demand function does not start to operate when "1" is set at address 10CH.)

2) The on-demand error starts to operate.

Send completed flag

The data sent is read from the storage area to check whether it has been normally sent.

Address 10CH
- "0" : The data is normally sent.
- "1" : The data is not sent due to a transmission error.

POINTS

(1) Buffer memory addresses 100H to 11FH are specific use areas.
Do not use them as send data storage areas.
When reading or writing data from or to buffer memory using a dedicated protocol or when sending or receiving data in the no-protocol or bidirectional mode, the area for send data storage must not overlap with the areas for these operations.

(2) Designate the data length within the following address ranges:
- When buffer memory addresses 0H to FFH are used
  Head address + designated data length - 1 ≤ FFH
- When buffer memory addresses 120H to DFFH are used
  Head address + designated data length - 1 ≤ DFFH

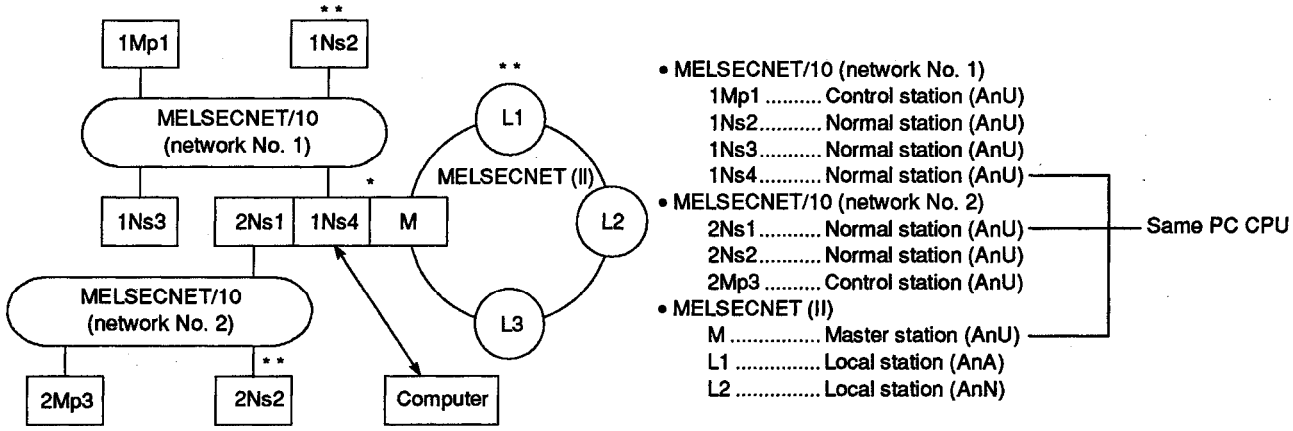
5.15 Switching between the Data Link System and the Network System to Access Another Station (for AnUCPU)

To establish a computer link between the computer and PC CPUs, which are not connected to the computer, in the following system configuration, the network must be registered beforehand, as described in this section.

(1) The PC CPU loaded with a computer link module is an AnUCPU.

(2) The PC CPUs of the stations to be accessed by the computer are connected through the MELSECNET (II), MELSECNET/B or MELSECNET/10.

(Example) MELSECNET (II)-MELSECNET/10 composite system



* : Station loaded with a computer link module

** : PC CPU to be accessed by the computer

This section describes the following processing operations necessary for computer-linking the computer and the PC CPUs in the above system configuration:

- Network registration
- Network reading
- Routing parameters reading

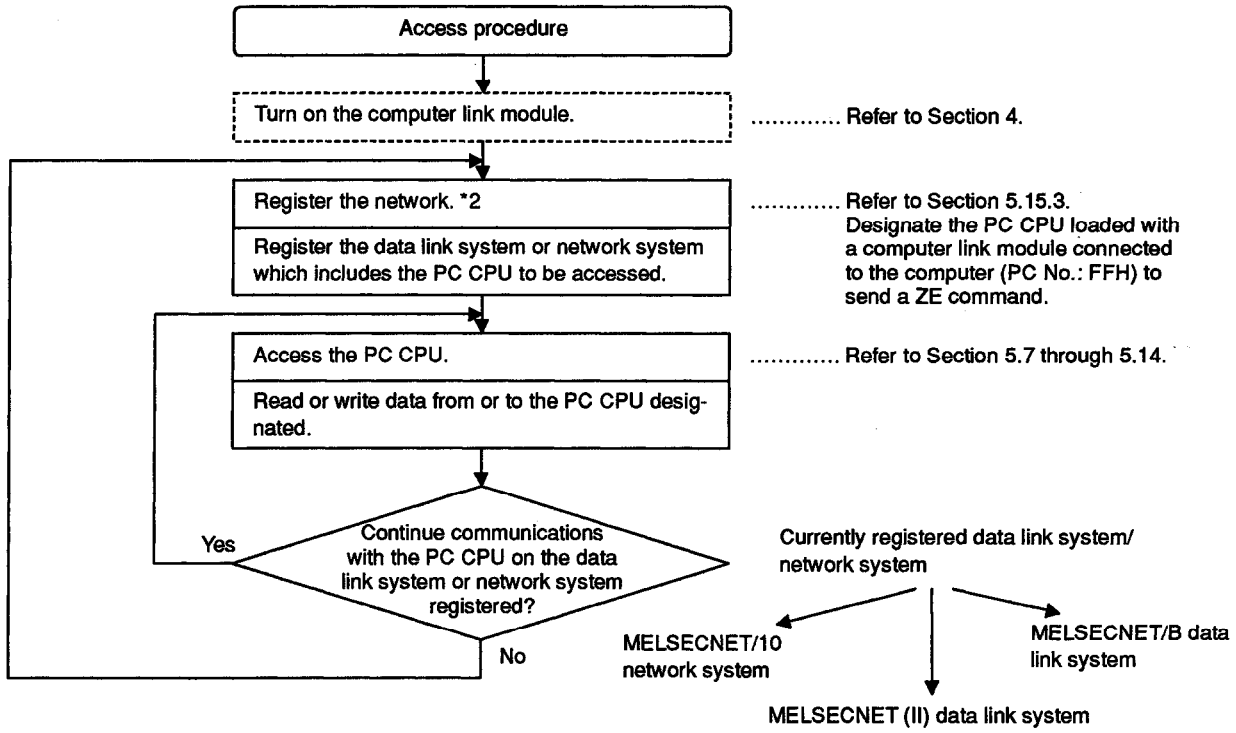
REMARK

When the PC CPU loaded (or to be loaded) with a computer link module is not an AnUCPU, the above-mentioned processing operations need not be executed on the computer. Skip this section.

5.15.1 Procedure for accessing another station

This section describes the procedure for making access from the computer to the PC CPU of another station when the PC CPU loaded with a computer link module is an AnUCPU.

Before turning on the computer link module, write parameters for the MELSECNET (II), MELSECNET/B and MELSECNET/10 to the PC CPUs concerned by the use of the GPP function of the MELSEC-A series.
For the parameters which should be written, refer to the Reference Manual for the data link system or network system adopted.*1

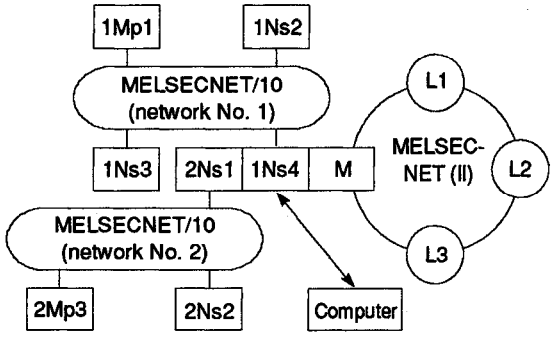


*1 Before registering the network, the following among the network parameters must be set with the PC CPU of the self station and of the relay stations:

- Number of modules set (All AnUCPUs must be set.)
- Routing parameters (They must be set with the self station and all relay stations.)

*2 Only one network can be registered.

(MELSECNET (II)-MELSECNET/10 composite system)

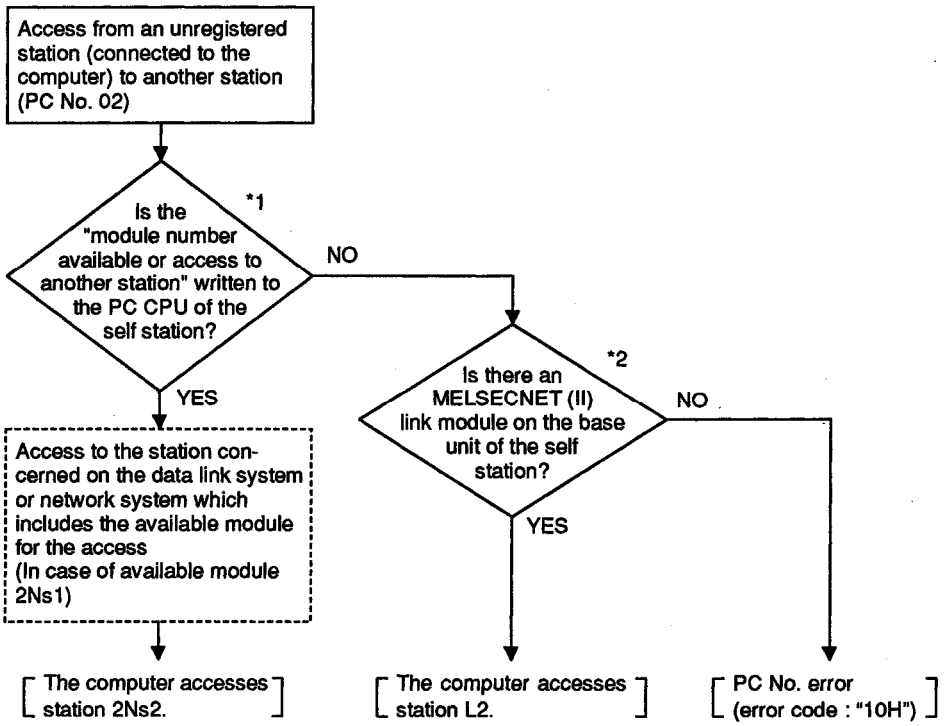


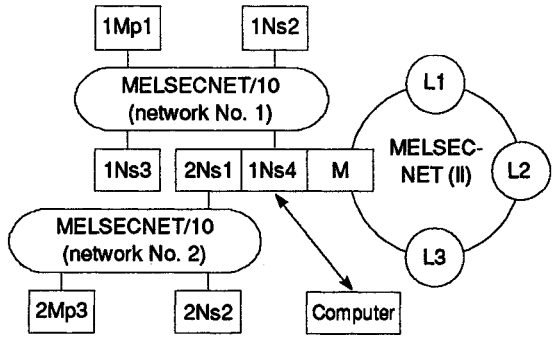
The following table shows the system codes and the network numbers which must be designated with regard to the left system in registering the network:

No.	PC CPU to be Accessed	System Code to be Designated	Network No. to be Designated
1	PC CPU on MELSECNET/10 (network No. 1)	01H	01H
2	PC CPU on MELSECNET/10 (network No. 2)	01H	02H
3	PC CPU on MELSECNET (II)	02H	00H

POINT

Before accessing another station, register the network with the self station.
If an attempt to access another station is made without registering the network, the following problems may occur:
(Example: MELSECNET/10-MELSECNET (II) composite system





*1: Set the "module number available for access to another station" in setting the number of modules by the use of the GPP function for the MELSECNET/10.
Refer to the GPP Function Operating Manual for the MELSECNET/10.

*2: When two MELSECNET (II) or MELSECNET/B link module are used, the computer accesses only the tier with the module loaded in the lower-numbered slot.

5.15.2 Commands and their functions

(1) AnUCPU dedicated commands

Item	Command		Description	Number of Points Processed per Communication	PC CPU State			Refernce Section
	Symbol	ASCII Code			During STOP	During RUN		
						Write Enabled	Write Disabled	
Network registration	ZE	5AH, 45H	Registers the name and network number of the system in which the station to be accessed is included.	——	O	O	O	5.15.3
Network reading	ZR	5AH, 52H	Reads the data on the network registered with the self station, an error code with the PC CPU or MELSECNET/10 or the name and network number of the system to be accessed.					5.15.4
Routing parameters reading	ZT	5AH, 54H	Reads the data on the network registered with the self station or the routing parameters set with the station to be accessed.					5.15.5

Note : O.....Executable

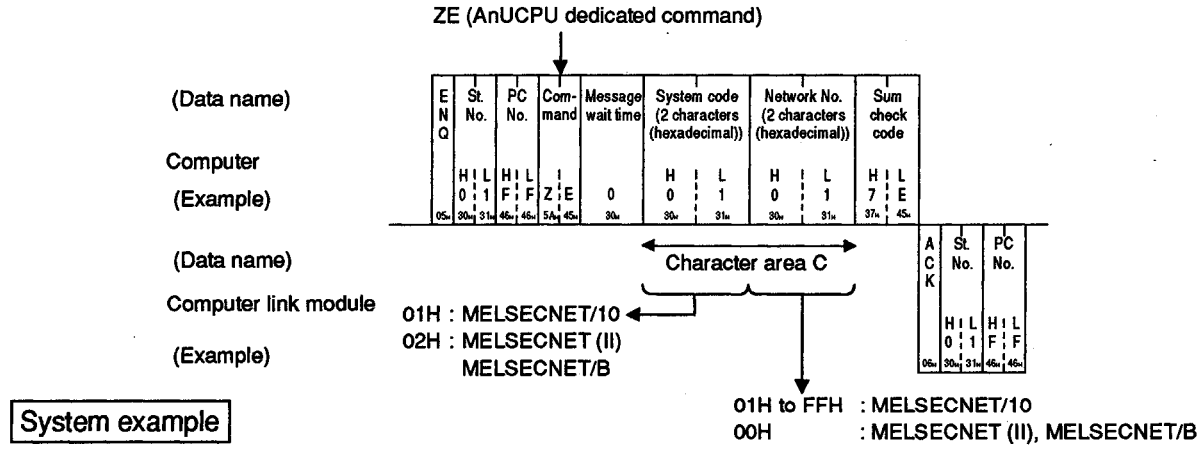
5.15.3 Network registration

In this section, examples are quoted to describe the control protocols for registering a network using a ZE command.

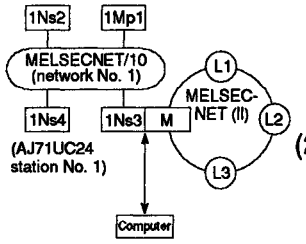
[Control protocols] The protocols shown below are in control format 1.
To access the system in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocols in this section.

(1) When registering the MELSECNET/10 (network No. 1)

(Network registration conditions)
1) The message wait time is 0 ms.
2) Register the MELSECNET/10 (network No. 1) in the left system.

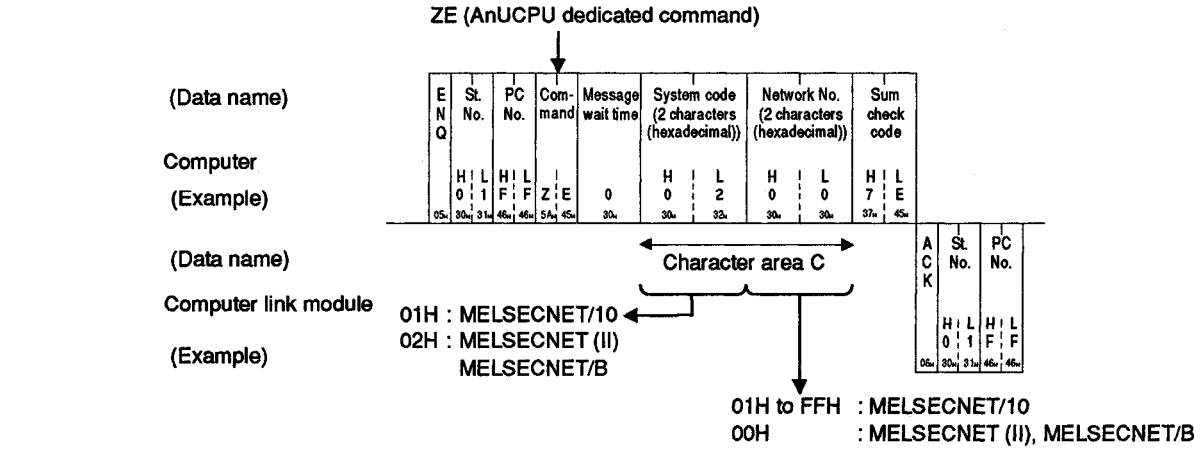


System example



(2) When registering the MELSECNET (II)

(Network registration conditions)
1) The message wait time is 0 ms.
2) Register the MELSECNET (II) in the left system.



POINT

(1) Since a network can be registered only with the self station, designate FFH for the PC No.

5.15.4 Network reading

The network reading function is used to help the computer check the following information:

(1) Error code

When the computer sends a command shown in Section 5.7 or the following sections, the computer link module returns an NAK message. When 40H or 41H is returned as an error code, the computer further identifies the error code.

(2) System code and network number

The computer checks the data on the currently registered network shown in Section 5.15.3.

(3) Data link module/Network module data

The computer checks the information about the data link module (MELSECNET (II), MELSECNET/B) or network module (MELSECNET/10) installed on the base unit of the PC CPU loaded with a computer link module.

In this section, an example is quoted to describe the control protocol for reading a network using a ZR command.

POINTS

(1) Only the stations on the network number to be registered or already registered with the self station are accessible.
(2) The information returned from the computer link module includes the data on the station (connected station) loaded with a computer link module connected to the computer and its station number, and the data on the station (accessed station) designated as the PC No.

Information Name	Station about Which Information is Returned		Description
	Connected Station	Accessed Station	
Error code	O	—	Information indicating the hexadecimal, ASCII code of an error existing in the MELSECNET/10 or the station (self station) loaded with a computer link module. (1) 1338H (4920) or higher error code: Indicates an error in the MELSECNET/10. Refer to the MELSECNET/10 Network System Reference Manual. (2) 1337H (4919) or lower error code: Indicates an error in the PC CPU. Refer to the User's Manual for the PC CPU concerned.
System code	O	—	Information about the currently registered network.
Network No.	O	—	
Data link module/Network module data			——
(1) System code	—	O	Information indicating the data link module (MELSECNET (II), MELSECNET/B) or network module (MELSECNET/10).
(2) Network No.	—	O	
(3) Data link system/ Network system station No.	—	O	Information indicating the hexadecimal, ASCII station number code of the PC CPU concerned on the data link system or network system.
(4) Group No.	—	O	Information indicating the hexadecimal, ASCII code of the group number set with the PC CPU concerned on the network system. The code is "00" when no group number is set with the PC CPU concerned or when the PC CPU concerned is a data link module.

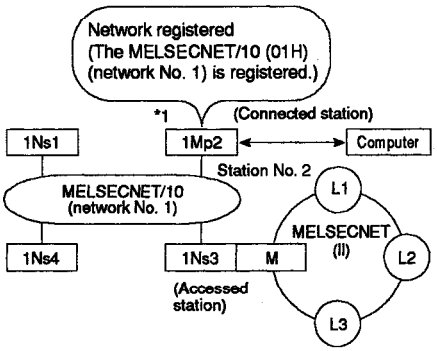
For the group number, refer to the MELSECNET/10 Network System Reference Manual.

[Control protocols] The protocol shown below is in control format 1.
To access the station concerned in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Network information reading conditions)

1) The message wait time is 0 ms.
2) The information about the network registered with the PC CPU (station 1Ns3/M) on the MELSECNET/10 (network No. 1) is to be read through the PC CPU (station 1Mp2) loaded with a computer link module (station No. 02) in the following system.

System example



*1
In the system example, the station number of the computer link module loaded in the station (1Mp2) connected to the computer is "2".
When network No. 1 is registered with station 1Mp2, the stations on the network are accessible.

*St. No. = Station number

The number of data link module/network module data equals the number of installed data link module/network module returned just before the data. A module uses eight characters.

ZR (AnUCPU dedicated command)

Data on first unit (1Ns3)

Data on second unit (M)

	ENQ	St. No.	PC No.	Command	Message wait time	Sum check code	ETK	St. No.	PC No.
(Data name)									
Computer (Example)		H L 02	H L 03	ZR	0	H L A1		H L 02	H L 03

	STX	St. No.	PC No.	Error code (4 characters (hexadecimal))	System code	Network No.	Number of installed data link module/network module	Data link module/Network link module data (1) (2) (3) (4)	Data link module/Network link module data (1) (2) (3) (4)	ETX	Sum check code
(Data name)											
Computer link module (Example)		H L 02	H L 03	0 0 0 0	H L 01	H L 01	H L 02	01 01 03 00	02 00 00 00		H L B3

Character area A is not set.

Character area B

Information about MELSECNET/10 or PC CPU of connected station
- 0000H : No error
- 0001H to 1337H : Error in PC CPU
- 1338H or higher : Error in MELSECNET/10

Information about network registered with connected station
- System code
  - 00H : No network registered
  - 01H : MELSECNET/10
  - 02H : MELSECNET (II), MELSECNET/B
- Network No.
  - 00H : No network No. registered, MELSECNET (II), MELSECNET/B
  - Other than 00H : Registered network No.

Information about accessed station
- (1) System code
  - 01H : MELSECNET/10
  - 02H : MELSECNET (II), MELSECNET/B
- (2) Network No.
  - 00H : MELSECNET (II), MELSECNET/B
  - Other than 00H : Set network No.
- (3) Data link system/Network system station No.
  - 00H to 40H : Station No. on data link system or network system concerned
- (4) Group No.
  - 00H : No group No. set, MELSECNET (II), MELSECNET/B
  - Other than 00H : Set group No.

Information about accessed station

5.15.5 Routing parameters reading

The routing parameters reading function is used to help the computer check the routing parameters set with the designated PC CPU.
In this section, an example is quoted to describe the control protocol for reading routing parameters using a ZT command.

POINTS

(1) Only the stations on the network number to be registered or already registered with the self station are accessible.
(2) When routing parameters are not set with the accessed station, the computer returns an NAK message and "32" as an error code.
(3) The information returned from the computer link module includes the data on the station (connected station) loaded with a computer link module connected to the computer and its station number, and the data on the station (accessed station) designated as the PC No.

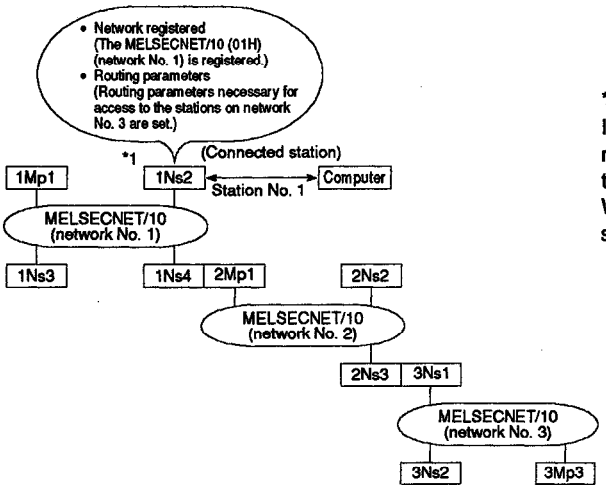
Information Name	Station about Which Information is Returned		Description
	Connected Station	Accessed Station	
System code	O	—	Information about the currently registered network.
Network No.	O	—	
Number of installed data link module/ network module	—	O	Information indicating the hexadecimal, ASCII code of the number of installed data link module or network module.
Number of set routing parameters	—	O	Information indicating the hexadecimal, ASCII code of the number of set routing parameters.
Routing parameter	—	O	Information indicating the hexadecimal, ASCII code of a set routing parameter.

For routing parameters, refer to the MELSECNET/10 Network System Reference Manual.

[Control protocol] The protocol shown below is in control format 1.
To access the station concerned in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Routing parameters reading conditions)
1) The message wait time is 0 ms.
2) Routing parameters (two parameters) set with the PC CPU (station 1Ns2, self station) loaded with a computer link module (station No. 01) in the following system are to be read.

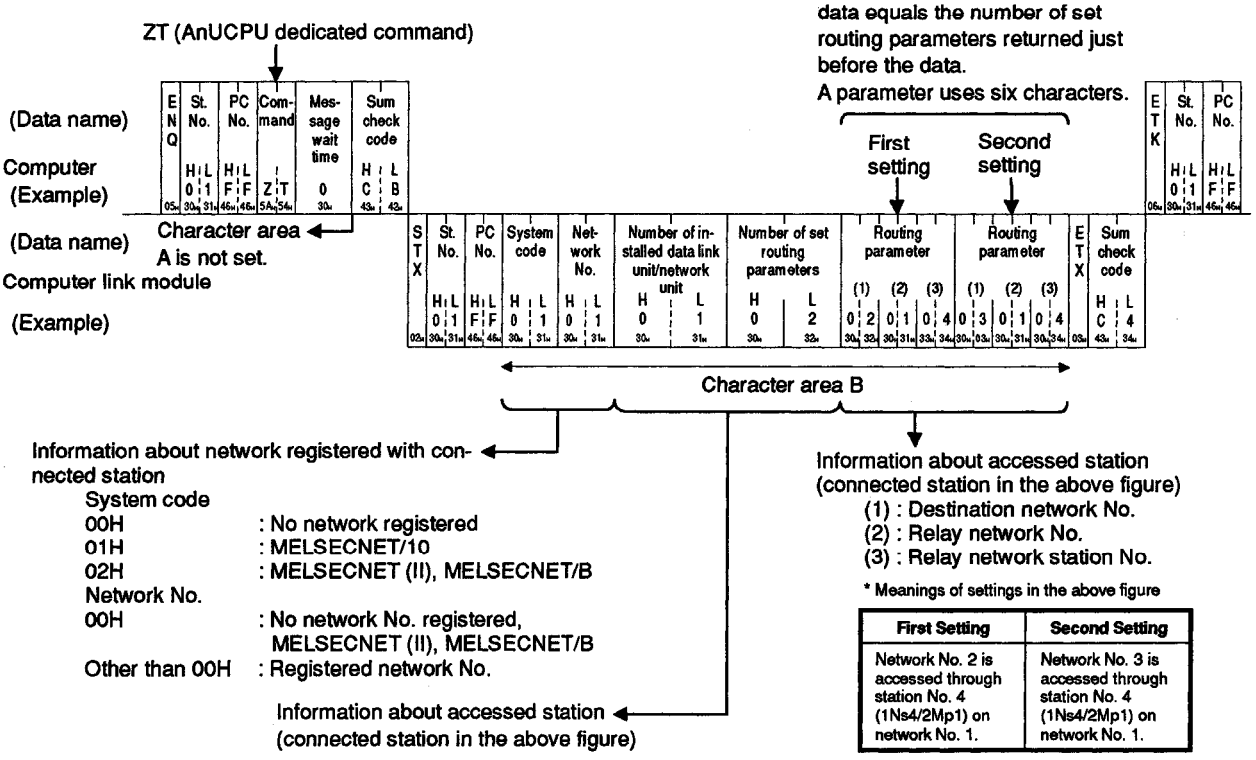
System example



*1
In the system, the station number of the computer link module loaded with the station (station 1Ns2) connected to the computer is "1".
When network No. 1 is registered with station 1Ns2, the stations on the network are accessible.

*St. No. = Station number

The number of routing parameter data equals the number of set routing parameters returned just before the data.
A parameter uses six characters.



Information about network registered with connected station
System code
00H : No network registered
01H : MELSECNET/10
02H : MELSECNET (II), MELSECNET/B
Network No.
00H : No network No. registered, MELSECNET (II), MELSECNET/B
Other than 00H : Registered network No.

Information about accessed station (connected station in the above figure)

Information about accessed station
(connected station in the above figure)
(1) : Destination network No.
(2) : Relay network No.
(3) : Relay network station No.

* Meanings of settings in the above figure

First Setting	Second Setting
Network No. 2 is accessed through station No. 4 (1Ns4/2Mp1) on network No. 1.	Network No. 3 is accessed through station No. 4 (1Ns4/2Mp1) on network No. 1.

5.16 Loopback Test

The loopback test function is used to check whether the computer and the computer link module communicate normally with each other. In this section, an example is quoted to describe the control protocol for using the function.

(1) ACPU common command and its functions

The following table shows the TT command to be used to conduct a loopback test, and its function.

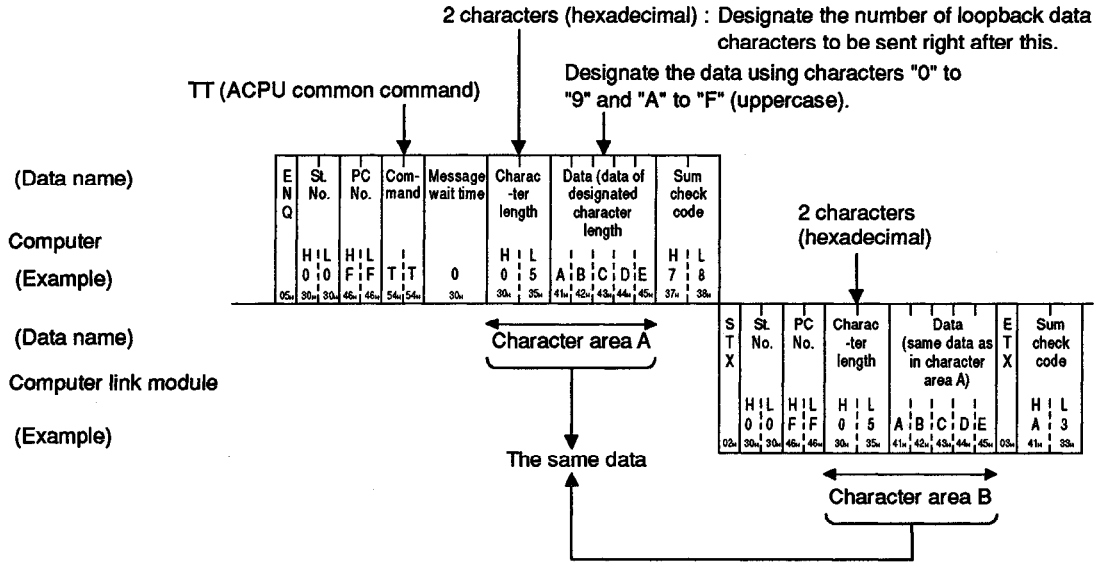
Item	Command		Description	Number of Points Processed per Communication	PC CPU State		
	Symbol	ASCII Code			During STOP	During RUN	
						Write Enabled	Write Disabled
Loopback test	TT	54H, 54H	Echoes back the characters to the computer as they are received	254 characters	O	O	O

Note : O.....Executable

(2) Loopback test control protocol

[Control protocol] The protocol shown below is in control format 1.
To access the station concerned in control format 2, 3 or 4, use the protocol described in Section 5.4.3, 5.4.4 or 5.4.5 as well as by reference to the protocol in this section.

(Loopback test conditions)
1) The message wait time is 0 ms.
2) Five characters "ABCDE" are to be sent and received as loopback data.



POINTS

(1) To designate the character length, the following condition must be met:
- 1 ≤ character length ≤ 254

(2) Designate "FF" for the PC No.

6. COMMUNICATIONS WITH EXTERNAL DEVICES IN THE NO-PROTOCOL MODE

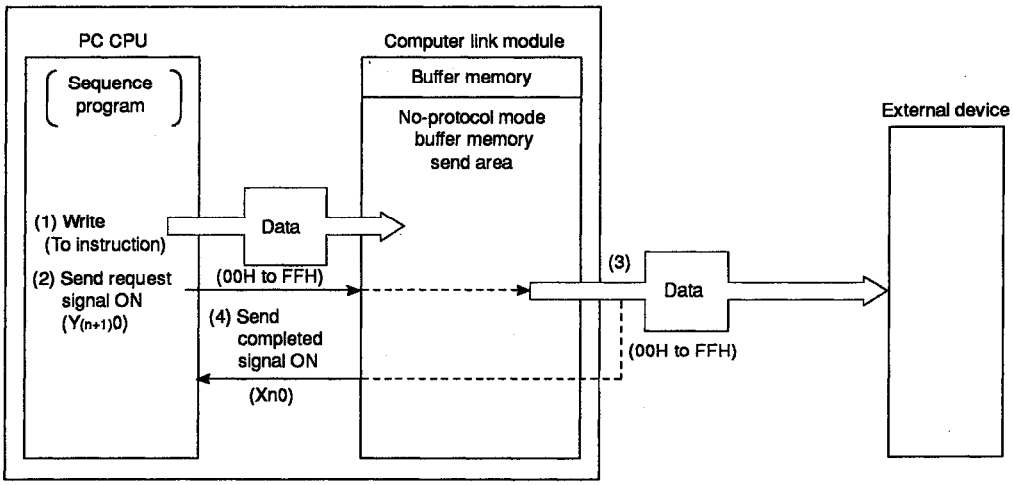
This section describes how to link the PC CPU with an external device (computers, printers, etc.) in the no-protocol mode.

> Read this section when using the RS-422 or RS-232C interface in the no-protocol mode by 1) setting the mode setting switch of the computer link module in a position of "1" to "9", or 2) switching the mode to 1 to 9.
> If these are used with the dedicated protocols and in the bidirectional mode, it is not necessary to read this section.

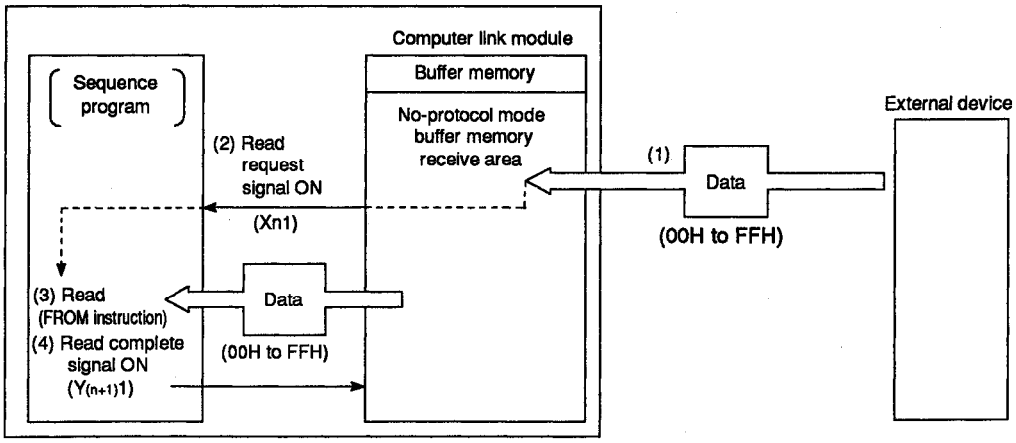
6.1 Data Flow in the No-Protocol Mode

The diagrams below are schematic representations of data communications between the PC CPU and the external device.
The computer link module transmits the data designated by the PC CPU to the external device in the same code.
The computer link module transmits the data received from an external device to the PC CPU also in the same code.
Thus, binary data (numeric data) can be received and sent in the no-protocol mode.

(1) When the PC CPU transmits data to the external device.



(2) When the PC CPU reads data received from an external device.



6.2 Programming Hints

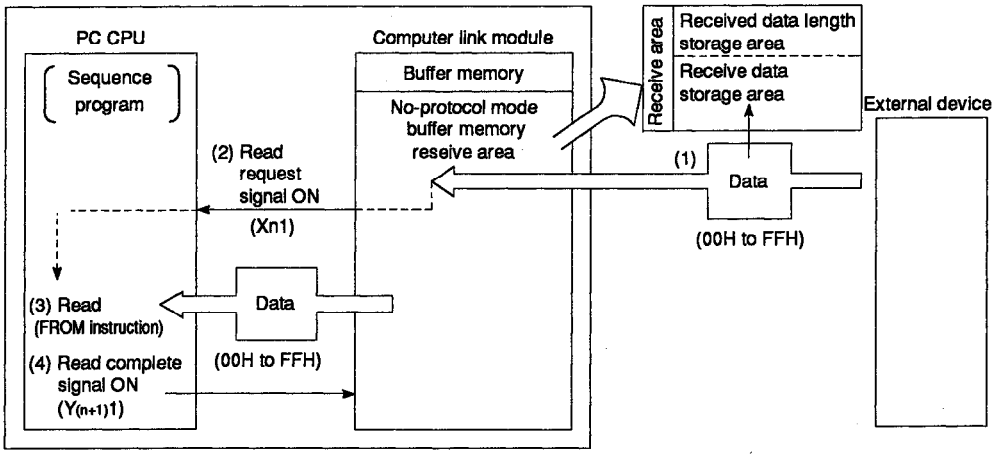
The following are hints concerning writing a program for linking a computer with an external device.

POINT

Functions in the no-protocol mode cannot be used in combination with functions in the bidirectional mode described in Section 7.
Select either mode by setting the mode setting switch of the computer link module (refer to Section 4.2.1) and set the buffer memory specific use area to the bidirectional mode designation area (refer to Sections 3.10 and 7.2.6).

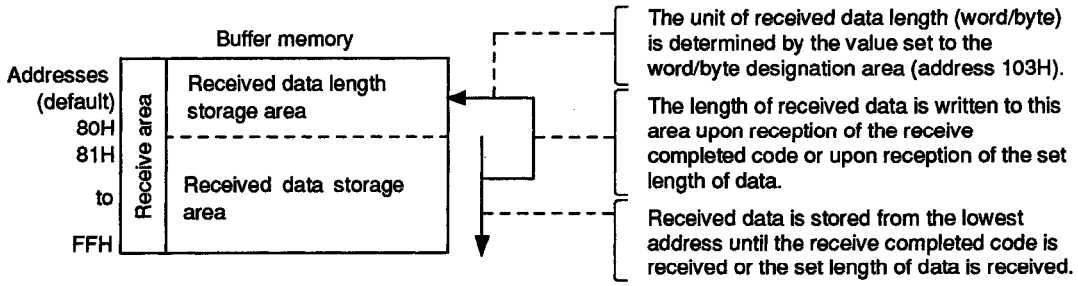
6.2.1 Receiving data from external device

This section describes the information about the procedure to operate the PC CPU to read data received from an external device.



(1) How to use the receive area

The receive area is provided in the buffer memory for storing data received from an external device and the length of the received data before the PC CPU reads the received data from this receive area.
The buffer memory addresses 80H to FFH are allocated as the receive area by default.
The receive area can be changed according to the purpose of data transmission, the specifications of the external device, and the received data length.
(Refer to 6.2.4(2)(d) for how to change the received data area.)



POINT

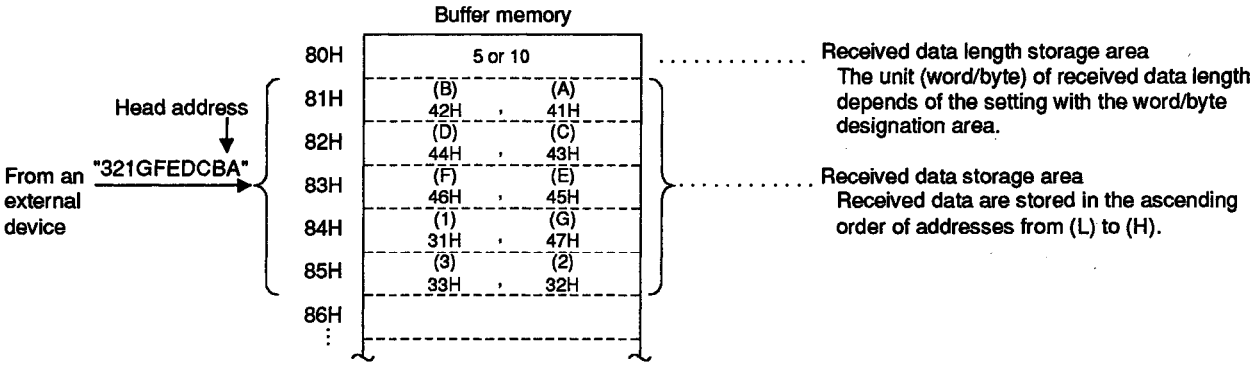
The amount of data sent from an external device to the computer link module in a single transmission operation should be kept below the size of the received data storage area.
(Received data storage area) >= (amount of data sent from an external device)
Change the receive area if it is necessary to send a greater amount of data than can be stored in the received data storage area.
For this purpose, the addresses and the size of the receive area can be changed.

(2) How to read received data

There are two methods available to enable the sequence program to read data received by the computer link module from an external device.
The user can choose either method.

In either method, the data received from an external device is read from the received data storage area.

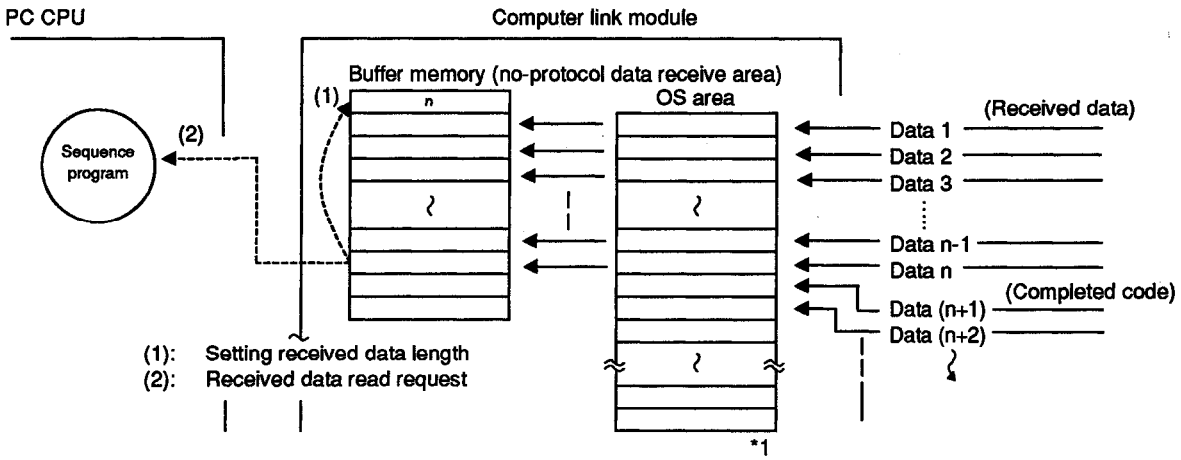
Example: To read received data ABCDEFG123. (The receive area in the example is default.)



The computer link module turns ON the received data read request signal (Xn1) to be sent to the sequence program when it receives 1) the receive completed data length set by the user or 2) the receive completed code.

(a) Reading received data upon reception of receive completed code (Data receive in variable length)

The computer link module makes a request to read the received data to the sequence program when it receives the receive completed code, set by the user to buffer memory address 100H.
Upon receiving the read request made by the computer link module, the sequence program reads the data received from an external device up to the receive completed code.
The receive completed code may be changed to any value which makes one word (one byte) in the range of 00H to FFH according to the specifications of the external device.
For the procedure to change the receive completed code, refer to Section 6.2.4 (2)(a).



Upon reception of the value set in the receive completed code designation area, the computer link module makes a read request to the PC CPU as follows.
(In the explanation below, the CR code is 0DH and the LF code 0AH.)

1) If the receive completed code is not changed (default: 0A0DH).
   a) When data is received within 60 ms after reception of CR.
      - The received data up to CR is stored in the receive area of the buffer memory and the read request signal is sent to the PC CPU.
      - The data received from this point on is stored in the OS area until it is read in the subsequent reading operations.
   b) When data other than LF is received within 60 ms after reception of CR.
      - The received data up to CR is stored in the receive area of the buffer memory and the read request signal is sent to the PC CPU.
      - The data received after CR is stored in the OS area to be read in the following read operation.
   c) When LF is received within 60 ms after reception of CR.
      - The received data up to LF(CR+LF) is stored in the receive area of the buffer memory and the read request signal is sent to the PC CPU.
      - The data that will be (or has been) received after LF is stored in the OS area to be read in the subsequent reading operations.
   d) When LF is received before CR.
      - The received data up to LF is stored in the receive area of the buffer memory and the read request signal is sent to the PC CPU.
      - The data received after LF is stored in the OS area to be read in the subsequent reading operations.
2) If the default receive completed code is changed to 00[][]H.
   a) When a user designated receive completed code is received.
      - The received data up to the user designated read completed code is stored in the receive area of the buffer memory and the read request signal is sent to the PC CPU.
      - The data that will be (or has been) received after the read completed code is stored in the OS area to be read in the subsequent reading operations.
3) If the read completed code is not set (FFFFH is set).

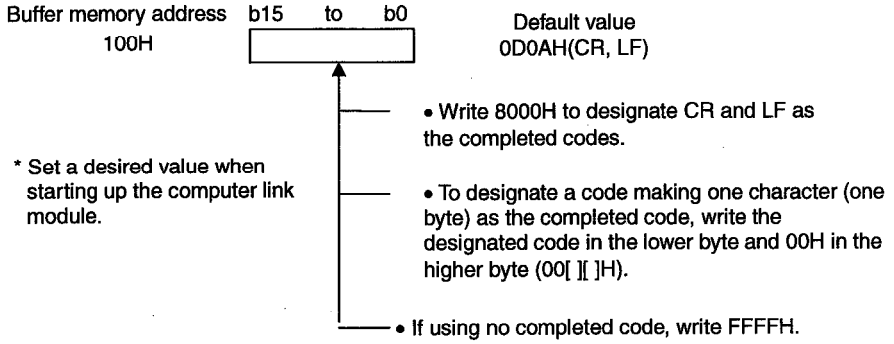
   This enables only the read of received data by fixed data length.
   See (b) of this section for details.

POINT

With the computer link modules with the software versions or later indicated in the table, it is possible to allow the read request signal to be turned ON only when CR and LF are received if the no-protocol receive completed codes are set as follows.

Applicable Module	AJ71UC24	A1SJ71UC24-R2/PRF/R4	A1SJ71C24-R2/PRF/R4	A1SCPUC24-R2	A2CCPUC24 /-PRF
Version	M	R	M	A	K

(No-protocol completed code designation area)

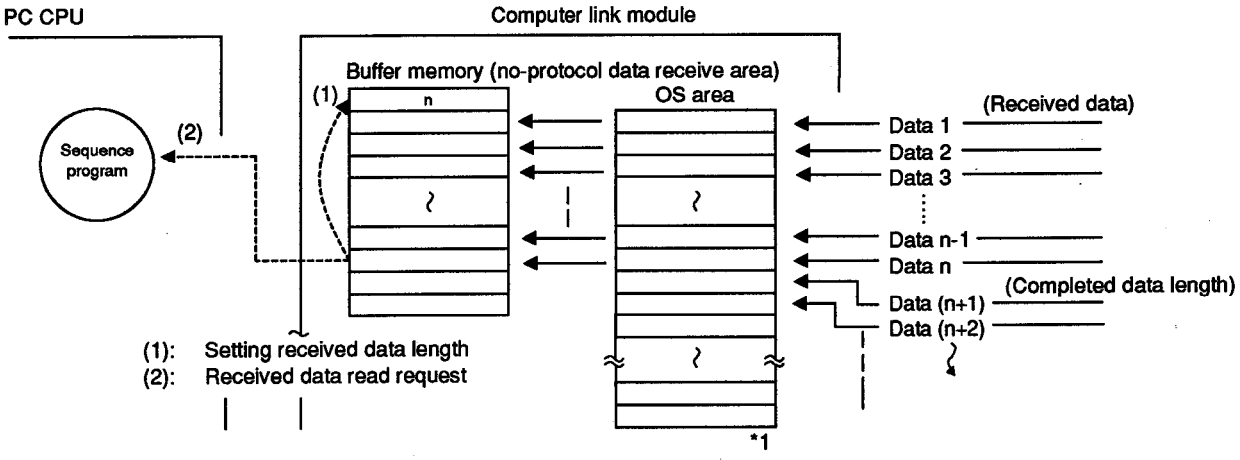


Upon reception of the code set in the receive completed code designation area, the computer link module makes the read request to the PC CPU as follows.

1) In the default setting of 0D0AH.
   • Refer to 1) a) to d) in the previous page.
2) When 8000H is set.
   • Only when CF and LF are sequentially received, the received data up to LF is stored in the receive area of the buffer memory and the read request signal is sent to the PC CPU.
   (No limit is set on the interval between the reception of CR and LF.)
   • The data received after LF is stored in the OS area to be read in the subsequently reading operations.
3) If the default receive completed code is changed to 00[ ][ ]H.
   • Refer to 2) in the previous page.
4) When FFFFH is set.
   Only data reading by the set data length is enabled.
   See (b) for details.

(b) Reading data by the set data length (data receive in fixed data length)

The computer link module makes a read request to the sequence program when it receives data which is equal to the number of data length set by the user to the buffer memory (address 108H) from the external device.
The sequence program can read the set length of data received from the external device when it has received the read request from the computer link module.
Default setting for the set data length is 127 words, but this value may be changed to any value not exceeding the size of the received data storage area.
(For the procedure to change the data length setting, refer to Section 6.2.4 (2)(b) and (e).)



*1 The OS area, as shown in the above diagram, is a memory provided in the computer link module for storing data received after the receive complete code or the set length of data is received.
The computer link module models covered by this manual are provided with OS areas with various capacities as indicated below.
The user cannot read or write these OS areas.

AJ71UC24 ..... 279 bytes
A1SJ71C24-R2, A1SJ71C24-PRF ..... 304 byte
A1SJ71C24-R4 ..... 304 bytes
A1SJ71UC24-R2/PRF/R4 ..... 304 bytes
A2CCPUC24, A2CCPUC24-PRF ..... 279 bytes

After the sequence program has been read the received data in response to the current read request, both the received data in the OS area and the data to be received subsequently is transferred to the data receive area in the buffer memory.
When the size of the vacant area in the OS area, where received data is stored, becomes smaller than 10 bytes, the following control operations are executed according to preset transmission control specifications. (For details, refer to Section 9. The RS signal is not turned OFF.)

- If the DTR control is enabled, the computer link module interrupts ongoing data communications with an external device by turning OFF the DTR signal.
- If the DC1/DC3 send control is enabled, the computer link module interrupts ongoing data communications with an external device by sending the DC3 code.

When there is no vacant area left in the OS area, so that no more received data can be stored, 2-SIO/4-SIO LEDs turn ON. From this point on, all the data subsequently received is abandoned until vacancy is made in the OS area.

POINT

When the receive completed code and the receive completed data length are set to the specific use area in the buffer memory, they are both effective.
In this case, if it receives the receive completed code before the data length to complete receive, the computer link module outputs the read request to the sequence program (Xn1 is ON).

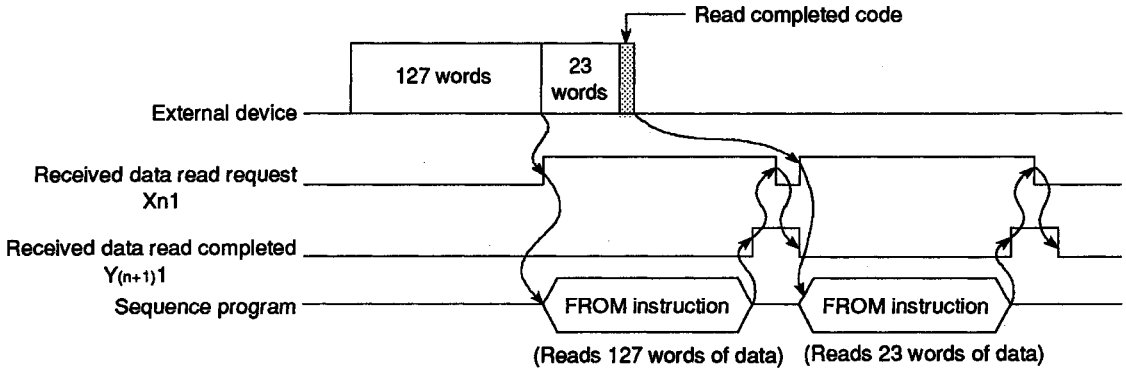
REMARKS

(1) If the received data length exceeds the capacity of the received data storage area, the data is processed as described below.

(a) When the receive completed code is used:

If the computer link module receives data that exceeds the received data storage area, it turns ON the received data read request signal Xn1 when data equivalent to the received data storage area has been received.
Reading the remaining data is enabled at the time the sequence program turns the receive data read completed signal Y(n+1)1 ON.
These steps are repeated until the receive completed code is received.
Set the receive area size so that "receive-completion data length" is less than "no-protocol mode receive buffer memory size".

Example: To receive 150 words of data while receive area is set at 80H to FFH (default).

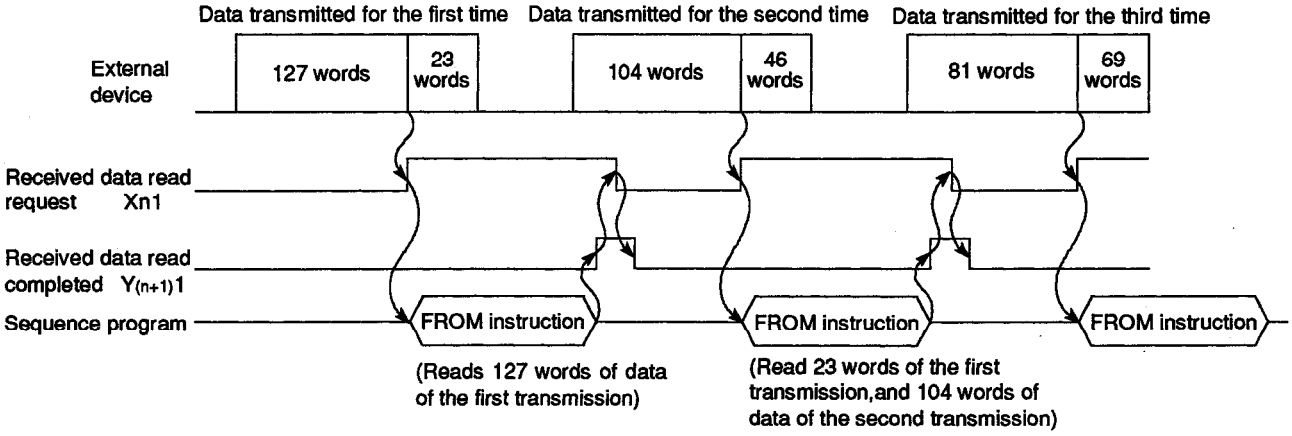


(b) When receive completion data length is used:

If the receive completion data length is set greater than the received data storage area, the no-protocol receive buffer memory size (default: 128 words) which is set at buffer memory address 107H is taken as the receive completion data length.

Set the receive area size so that "receive completion data length" is less than "no-protocol mode receive buffer memory size".

Example: To receive 150 words of data while receive area is set at 80H to FFH (default setting).



(2) Note that if the CD signal is turned OFF or the mode is switched during data communications with an external device the received data may be cleared as described below.

(a) If the CD signal is turned OFF:

- While data communication with an external device through the RS-232C line with the CD terminal check enabled and with no read request is being made to the PC CPU, received data is cleared if the CD signal is turned OFF for 4 ms or longer on the external device . (The receive area is not cleared.)
- If the CD signal goes OFF when a read request is being made to the PC CPU, the computer link module continues to process the read request to the PC CPU. The received data the PC CPU is request to read is not cleared.
- To resume data send from an external device, turn ON the CD signal.

(b) When the mode is switched:

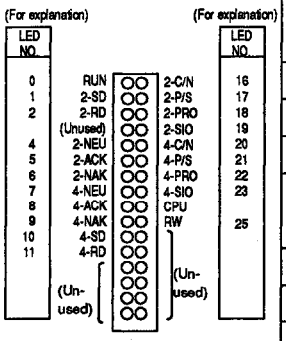
- If the mode of the module is forcibly switched as described in Section 12, the computer link module goes into the initialized state as immediately after power application, thus clearing the data. (The receive area is initialized.)
- To continue data communication, it is necessary to carry out initial settings of the specific use area in the buffer memory.
- Do not resume data communication from an external device until after the ready signal (Xn7) is turned ON and initial settings are completed to the specific use area in the buffer memory.

  (With some computer link systems, it is first necessary for the PC CPU to send a send resume enabled signal to an external device first before resuming data send from the external device.)

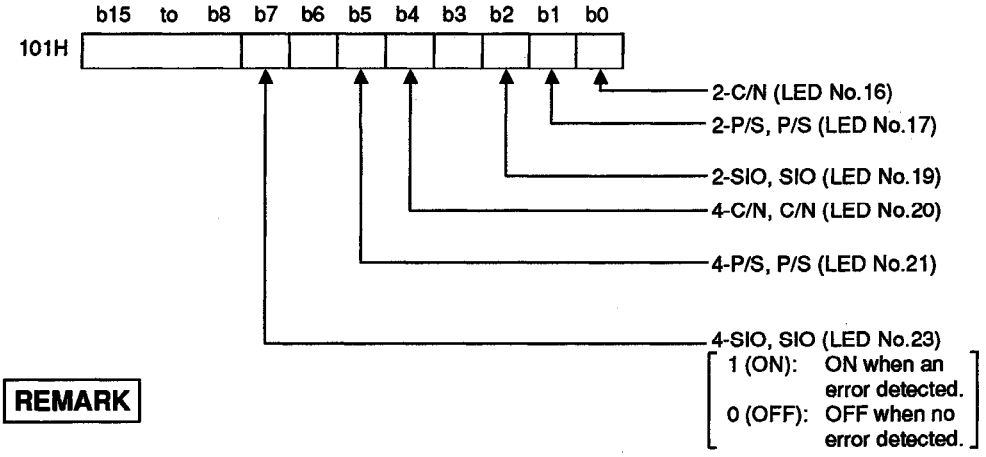
(3) Error detection (by reading the error LED display area or the buffer memory)

If the computer link module detects an error in the data received from an external device in the no-protocol mode, the computer link module turns ON the LEDs (2-C/N to 2-SIO or 4-C/N to 4-SIO) provided on its front panel and stores the error information in the buffer memory address 101H.
Check the occurrence of a data receive error by the LEDs located on the front side of the module on by reading the above-mentioned buffer memory by the sequence program.

(a) Applicable LEDs on the front panel

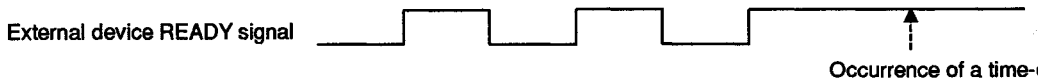
LED Layout (AJ71UC24)	LED No.	LED Name		Meaning of LED Display	LED ON	LED OFF	LED Initial Status
							
	16	2-C/N	C/N	Result of communications between PC CPU and RS-232C	Refer to Section 4.3 (4).	Normal	OFF
	17	2-P/S	P/S	Parity error on RS-232C	Parity error	Normal	OFF
	19	2-SIO	SIO	SIO error on RS-232C	Overrun, framing error	Normal	OFF
	20	4-C/N	C/N	Result of communications between PC CPU and RS-422	Refer to Section 4.3 (4).	Normal	OFF
	21	4-P/S	P/S	Parity error on RS-422	Parity error	Normal	OFF
	23	4-SIO	SIO	SIO error on RS-422	Overrun, framing error	Normal	OFF
	When an LED turns ON, it stays ON even when the cause of the error is eliminated. To turn OFF an LED, follow the procedure described in Section 8.1.						

(b) Buffer memory (See Section 8.1 for details.)



REMARK

Connect signals such as the READY signal from an external device to the PC CPU as an input signal and then run a time-out check on the ON/OFF intervals of the signal to determine the status of the external device.



The status of an external device may also be determined by running a time-out check on intervals of data reception.

(4) Clearing received data

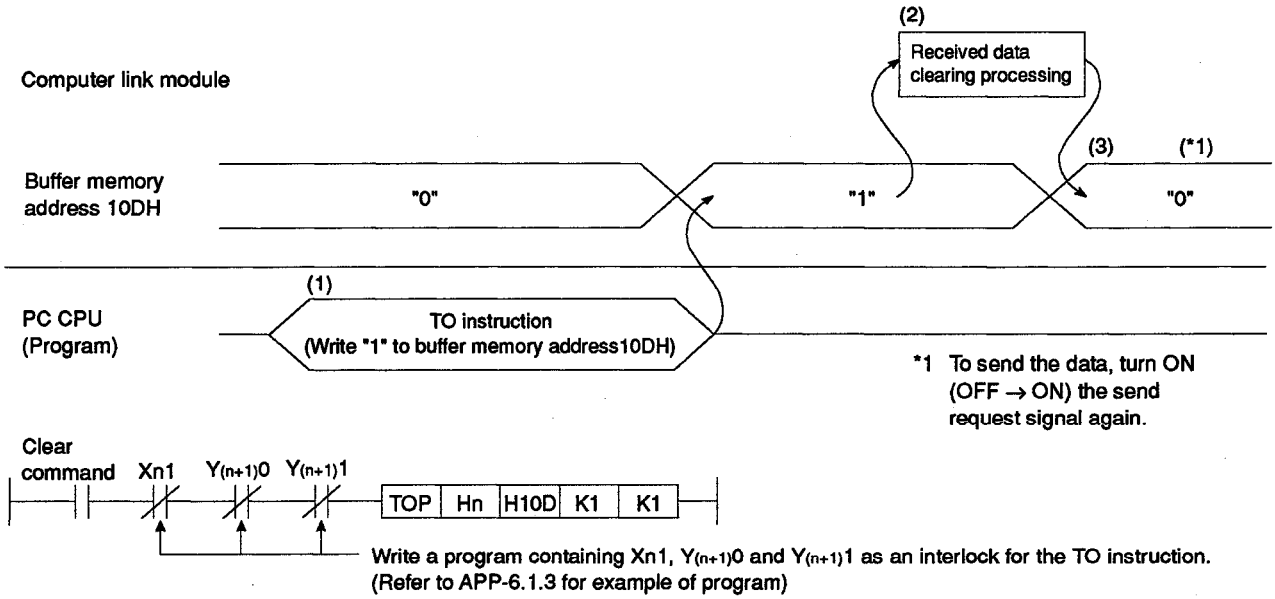
If an error occurs due to failure of an external device, for example, while receiving data from an external device in the no-protocol mode, the data received up to the error may be incorrect or interrupted.
To recover after an error has occurred it is possible to clear all received data and initialize the buffer memory.
To clear all received data, write and read to the buffer memory as described below.
When all received data is cleared, the computer link module nullifies the received data, so that there is no received data left.
(The previous data in the receive area of the buffer memory remains uncleared.)
Do not resume data send from an external device until after buffer memory address 10DH is changed to "0".
(With some computer link systems, it is first necessary for the PC CPU to send a send resume enabled signal to an external device before resuming data send from the external device.)

(How to clear received data)

(a) Write "1" to buffer memory address 10DH using the TO instruction of the sequence program while the received data read request signal (Xn1), the received data read completed signal ($Y_{(n+1)}1$) and the send request signal ($Y_{(n+1)}0$) are OFF.

(b) Now the computer link module clears the received data.

(c) After clearing the received data, the computer link module clears the "1" that was written to buffer memory address 10DH and write "0" in its place.

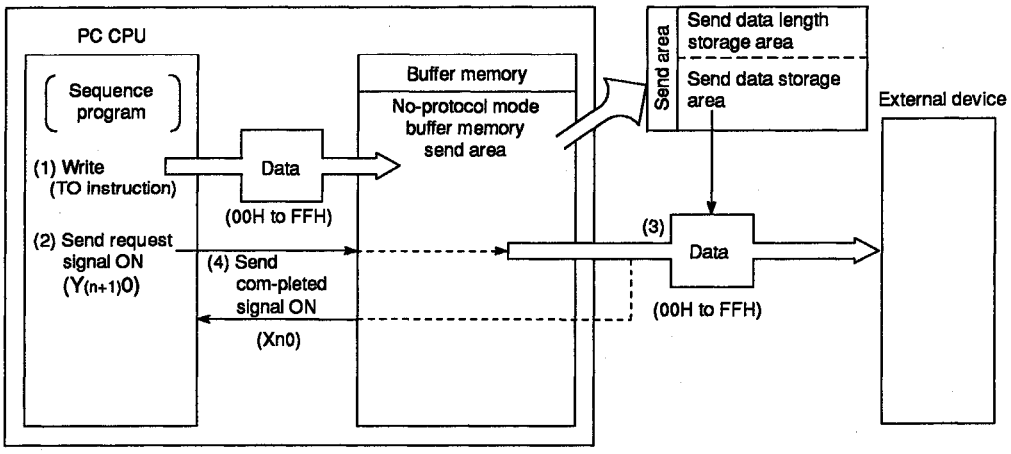


REMARK

Do not issue a receive data clear request as described above while the send request signal ($Y_{(n+1)}0$) is ON and data is being sent from a computer link module to an external device, otherwise the computer link module will stop sending the data and the send completed signal (Xn0) will come ON.

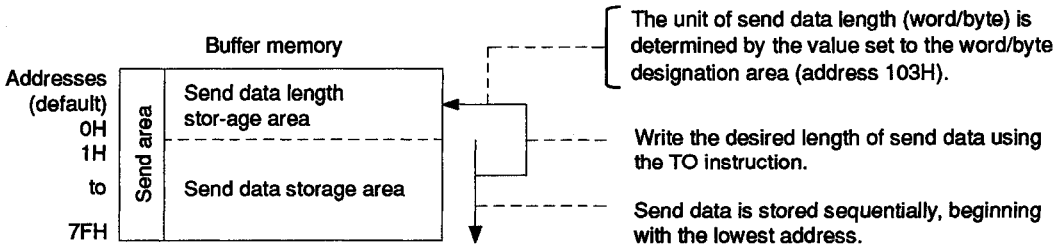
6.2.2 Sending data to external device

This section describes matters to be attended to for sending data to an external device.



(1) Data send area

The data send area is the buffer memory area provided in the computer link module where the PC CPU stores send data to be sent to the external device and the length of the send data.
The buffer memory addresses 0H to 7FH are allocated as the send area by default.
The send area can be changed according to the purpose of data send, the specifications of the external device, and the sent data length.
(See Section 6.2.4(2)(c) for how to change the received data area.)



POINT

There is a limit of the amount of data that can be sent from the PC CPU to an external device in a single transmission operation.
Such amount of data should be kept below the size of the send data storage area.
(Send data storage area) >= (amount of data sent from the PC CPU to an external device)
Change the receive area if it is necessary to send a greater amount of data than can be stored in the send data storage area in a single transmission operation.
For this purpose, the addresses and the size of the receive area can be changed.

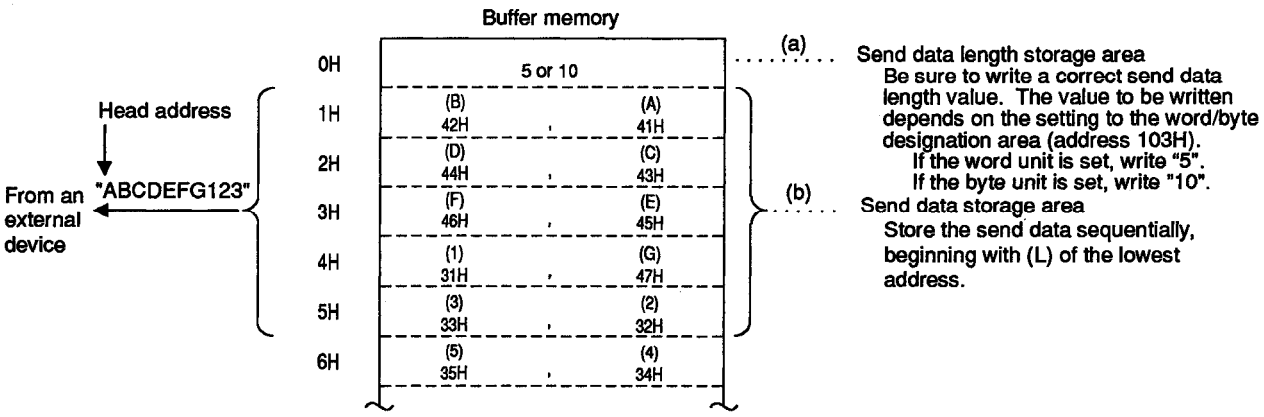
(2) How to write send data

The send data length and send data are written to the send data area as shown below.

(a) The length of send data to be written (or having been written) to the send data storage area is written to the no-protocol send data length storage area in either words or bytes.

(b) The data to be sent is written to the send data storage area.

Example: "ABCDEFG123" is sent. (The send area addresses are default.)

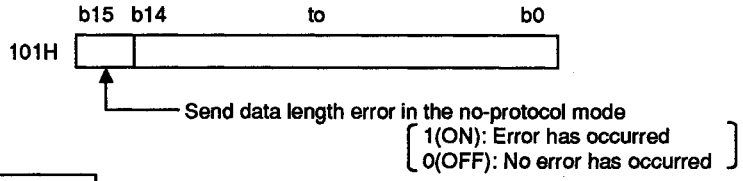


When the send request signal ($Y_{(n+1)}0$) is turned ON after (a) and (b) above have been executed, the computer link module transmits the set length of data from the send data storage area from the from the lowest address.

(3) How to detect a send error

The only error detection the computer link module can carry out in the data send to an external device in the no-protocol mode is whether a correct value is written to the send data length storage area in the buffer memory.
If the set length of send data is incorrect, the computer link module stores the error code at address 101H in the buffer memory.
To confirm occurrence of an error, check for the code at the buffer memory address using the sequence program.
Execute other error detecting procedures at an external device.

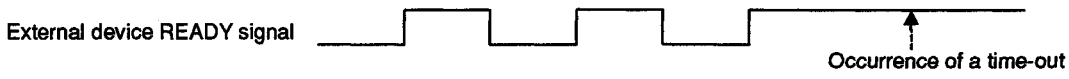
- Buffer memory (See Section 8.1 for details.)



REMARK

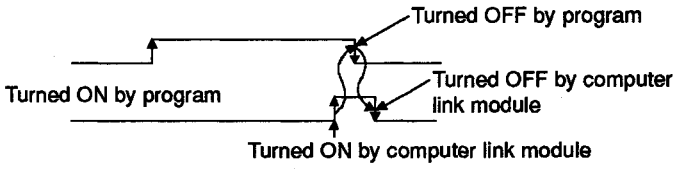
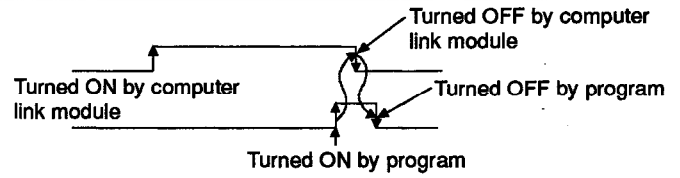
Determining the operating status of an external device

Connect signals such as the READY signal from an external device to the PC CPU as the input signal and then run a time-out check on the ON/OFF intervals of the signal to determine the status of the external device.



6.2.3 Handshake I/O Signals

Signals known as I/O handshake signals are required for no-protocol communications.
These signals (a) output data received from the sequence program to an external device, or (b) detect signals from an external device to enable the sequence program to read them.

	Signal		Timing
PC CPU ↓ External device	$Y_{(n+1)}0$	(Send request)	
	Xn0	(Send completed)	
External device ↓ PC CPU	Xn1	(Received data read request)	
	$Y_{(n+1)}1$	(Receive data read completed)	

The number "n" appended to X and Y is determined according to the position where the computer link module is loaded and the number of I/O modules loaded to this module.

REMARK

The I/O signals, other than those mentioned above, available in bidirectional mode are: Xn7 (computer link module READY signal) and XnD (computer link module watchdog timer error signal).
Refer to Section 3.9 for the I/O signals used with the PC CPU.

6.2.4 Reading and writing buffer memory

A sequence program for reading/writing buffer memory is necessary to execute the operations using the computer link module in the no-protocol mode.
(Refer to the buffer memory list in Section 3.10.)
Write a sequence program as required.
At a start-up of the computer link module, the default values are written to the specific use area in the buffer memory.

	Reading and Writing Operations that Need Sequence Programs	Reference Sections
(1)	To change the default value set to the buffer memory specific use area of the computer link module	Sections 6.2.4(2) and (3)
(2)	To receive data from an external device	Sections 6.2.1 and 6.4
(3)	To transmit data to an external device	Sections 6.2.2 and 6.5
(4)	To clear (nullify) all the data received up to the present from an external device	Section 6.2.1(4)
(5)	To read the LED ON/OFF statuses of the computer link module or turn OFF LEDs during computer linking.	Section 8
(6)	To read the module/signal statuses during computer linking.	Section 8
(7)	To execute transmission control with an external device using the DC code.	Section 9
(8)	To execute half-duplex communication on the RS-232C interface of the computer link module	Section 10

(1) Precautions on reading/writing the buffer memory specific use area

(a) The buffer memory is not backed up by a battery.

All data in buffer memory is reset to the default values when power is turned ON, when the PC CPU is reset, or when the mode is switched.
Data changed from the default values must be written whenever any of these events takes place.

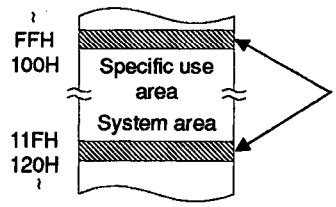
(b) Only TO instruction can be used to write data to the specific use area (100H to 11FH). Data written with any other instruction is invalid.
If data is written to the buffer memory using the command in a computer program, the computer module will not operate correctly.

(c) If the following functions are used together, make sure to allocate the user area in buffer memory so that the same area will not be used by different functions.
If the same area is allocated to different functions, the data in this area is rewritten and communication will not be correctly executed.

- No-protocol mode send
- No-protocol mode receive
- Buffer memory read/write (CR/CW commands) functions
- On-demand functions

The memory areas preceding and following the specific use area cannot be allocated as a single area.
The areas 0H to FFH and 120H to DFFH must be recognized as independent areas.

Example)



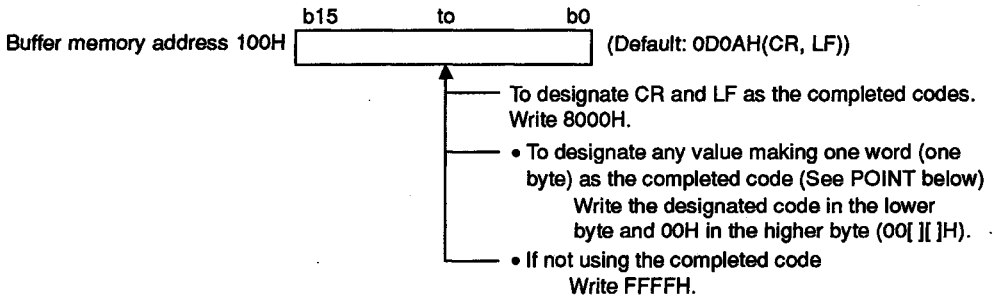
(2) Changing default values in the buffer memory specific use area

Write an applicable sequence program to change a default value in the specific use area in the buffer memory as follows.

(a) The no-protocol mode receive completed code designation area (address: 100H)

- By default the computer link module makes a read request to the PC CPU if the variable length data received from an external device contains either or both of the completed code CR and LF. (If both of the CR and LF codes are to be received, the LF code must be received within 60 ms from the reception of the RC code.)
- To change the completed codewrite a desired receive completion code at the start of the computer link module or at the timing explained in item (4).

SETTING METHOD



POINTS

(1) Refer to Section 6.2.1(2)(a) for the timing in which the computer link module makes a read request to the PC CPU after receiving the value set to the receive completed code designation area.

(2) When LF is received after CR with 8000H set as the completed code, the computer link module makes a read request to the sequence program (Xn1 is ON).
No limit is set on the interval between the reception of CR and LF.
It is possible to designate 8000H as the completed code with the computer link modules with the indicated software versions or later in the table.

Applicable Module	AJ71UC24	A1SJ71 UC24-R2/R4/PRF	A1SJ71 C24-R2/R4/PRF	A1SCPU C24-R2	A2CCPU C24	A2CCPU C24-PRF
Version	M	R	M	A	K	K

(3) The default completed code may be changed to any code making 1 byte in the range of 00H to FFH.

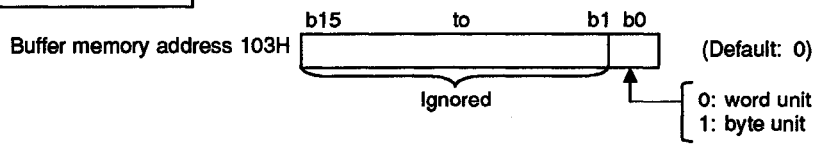
(4) If no completed code is used, write FFFFH at buffer memory address 100H.
This makes only the setting of data length effective, thus enabling only the read of received data by set data length.

(5) If the receive complete data length has also been set, a received data request is output (Xn1 comes ON) as soon as the receive completed code is received before receiving the data which is equal to the set complete data length.

(b) The no-protocol mode word/byte designation area (address: 103H)

- When the computer link module is started up, the default setting of the length of the data which is sent and received between the PC CPU and the computer link module is the word unit.
- To change the default setting to the byte unit, write "1" at address 103H after the computer link module has started up.
- These units are also used as the unit of send data length in the on-demand functions in the dedicated protocol.

SETTING METHOD

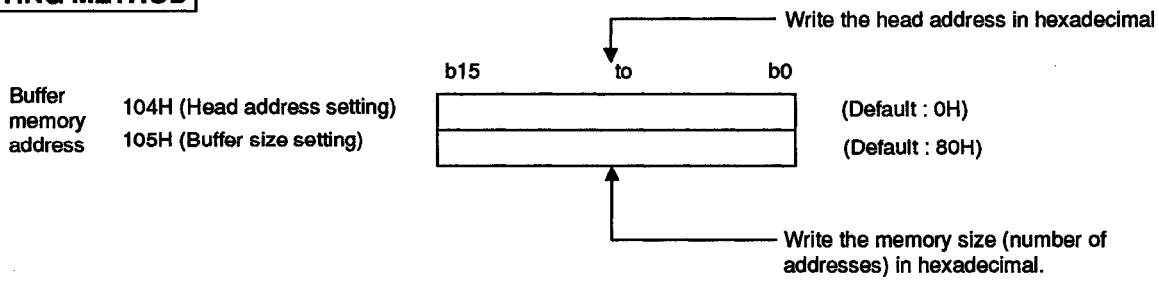


(c) The addresses for changing the buffer memory send area

The no-protocol send buffer memory area head address designation area (Address: 104H)
The no-protocol send buffer memory length designation area (Address: 105H)

- By default, buffer memory addresses of 0H to 7FH (size: 80H) are allocated to store send data from the PC CPU and also the length of the data.
- To change the default send data length and buffer memory area to store send data, write the head address of the new buffer memory area and the new buffer memory size when the computer link module is started up.
- Buffer memory areas 0H to FFH or 120H to DFFH can be allocated to the no-protocol mode send buffer memory (refer to (1)(c)).

SETTING METHOD



POINTS

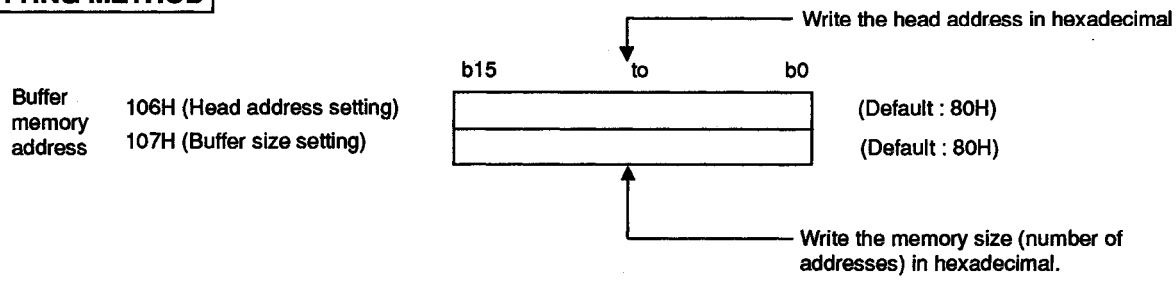
(1) Buffer memory addresses 100H to 11FH are for the specific use area and should not be set.
(2) Buffer memory address 105H should include the storage area of the no-protocol send data length.
(3) If any range except the user area is set, the computer link module will execute operations with defaults.

(d) The addresses for changing the buffer memory receive area

The no-protocol receive buffer memory area head address designation area (Address: 106H)
The no-protocol receive buffer memory length designation area (Address: 107H)

- By default, buffer memory addresses of 80H to FFH (size: 80H) are allocated to store data that has been sent from an external device and is to be received by the PC CPU and the length of such data.
- To change the default receive data length and buffer memory area to store received data, write the head address of the new buffer memory area and the new buffer memory size when the computer link module is started up.
- Buffer memory areas 0H to FHH or 120H to DFFH can be allocated to the no-protocol mode receive buffer memory. (Refer to (1)(c).)

SETTING METHOD



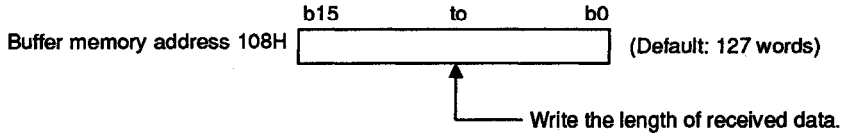
POINTS

(1) Buffer memory addresses 100H to 11FH are for the special application area and should not be set.
(2) Buffer memory address 107H should include the storage area of the no-protocol receive data length.
(3) If any range except the user area is set, the computer link module will execute operations with defaults.

(e) The no-protocol receive completion data length designation area (Address: 108H)

- In the default setting of the set data length reception, the computer link module makes a request to read data to the PC CPU when it has received 127 words of data.
  (The data unit of the receive completion data length is determined by the value set at memory buffer address 103H.)
- To change the completion data length, write the desired receive completion data length at the start of the computer link module or at the timing explained in item (4).

SETTING METHOD



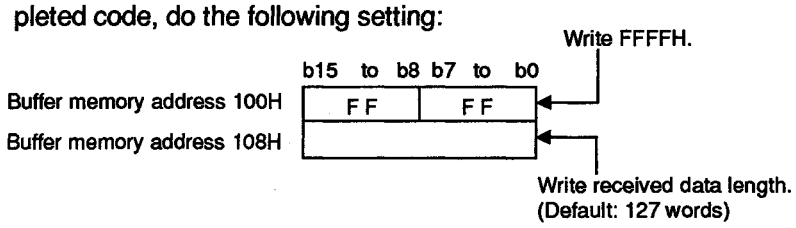
POINTS

(1) Set the length of data to complete the data receive in the following ranges:
- Length of data received ≤ no-protocol mode buffer size (when word units are set)
- Length of data received ≤ no-protocol mode buffer size x 2 (when byte units are set)
- If the received data length is larger than the no-protocol mode buffer memory size, then it is automatically set equal to the no-protocol mode buffer memory size.

(2) The unit (word/byte) for the data length to complete data receive is determined by the setting described in (2).

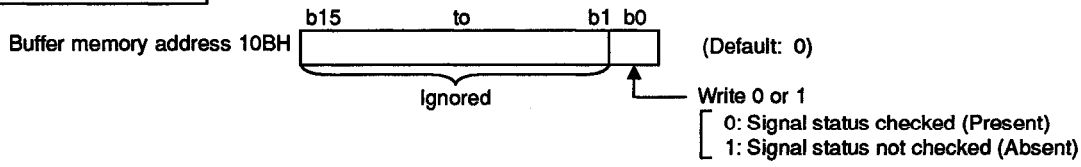
(3) If the receive completed code is set as well, the read request for the received data is sent when the receive completed code is received even if the set data length of data is not received. (Xn1 is ON.)

(4) To read the received data by fixed data length without setting the completed code, do the following setting:



(f) RS-232C CD terminal check designation area (Address: 10BH)
- When an external device is connected on the RS-232C of the computer link module, the setting described in Section 4.6.2 needs to be done.
- The default setting of the CD terminal check is ON.
- To turn OFF the CD terminal check, write "1" to buffer memory address 108H when the computer link module is started up.
- This setting is effective for any function in the dedicated protocol and no-protocol mode.

SETTING METHOD



REMARK

For the operation of the computer link module to the CD signal in an RS-232C CD terminal check, refer to Section 4.6.2.

(g) The no-protocol receive data clear request area (address: 10DH)

- All received data can be cleared by using this area if an error or some other trouble occurs while receiving data from an external device,
- Refer to Section 6.2.1 (4) for how to make the received data clear request.

(h) Areas for changing the communications mode of the RS-232C interface

RS-232C communications mode setting area (address: 10FH)
Simultaneous send priority/non-priority setting area (address: 110H)
Send method setting area when transmission is resumed (address: 111H)

- When an external device is connected to the RS-232C interface of the computer link module, the full-duplex transmission with the external device is possible when the computer link module is started up.
- To perform half-duplex transmission, change the default values when the computer link module is started up as described in Section 10.

(i) Areas for changing the transmission control mode with the external device

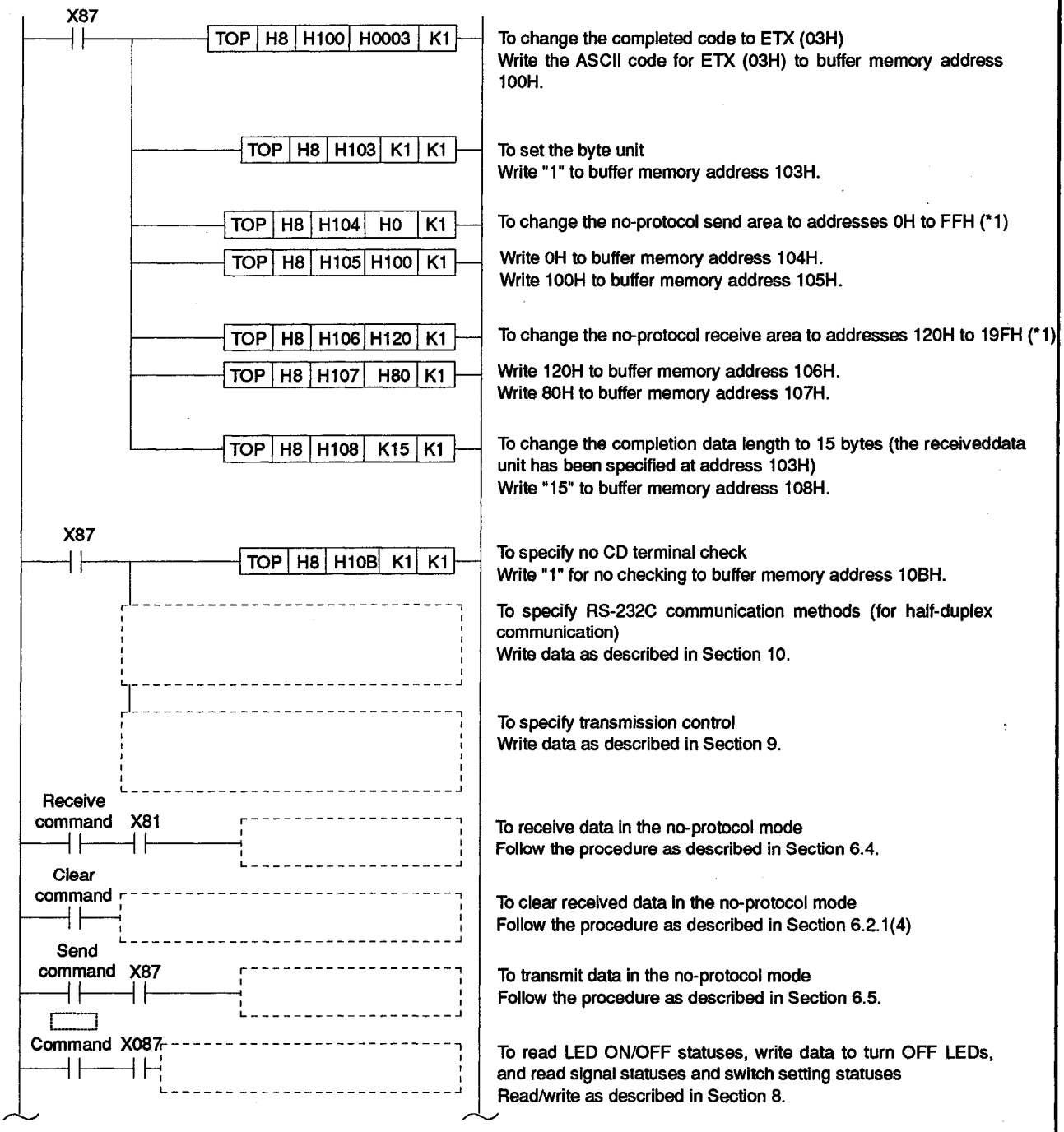
Transmission control specification area (address: 11AH)
DC1/DC3 control code specification area (address: 11BH)
DC2/DC4 control code specification area (address: 11CH)

- When the computer link module is started up, DTR/DSR control for the external device is executed only at the RS-232C interface.
- To execute DC code control at the RS-232C and RS-422 interfaces, change the default settings when the computer link module is started up, as described in Section 9.
- This setting is effective to all functions available with dedicated protocols and the no-protocol mode.

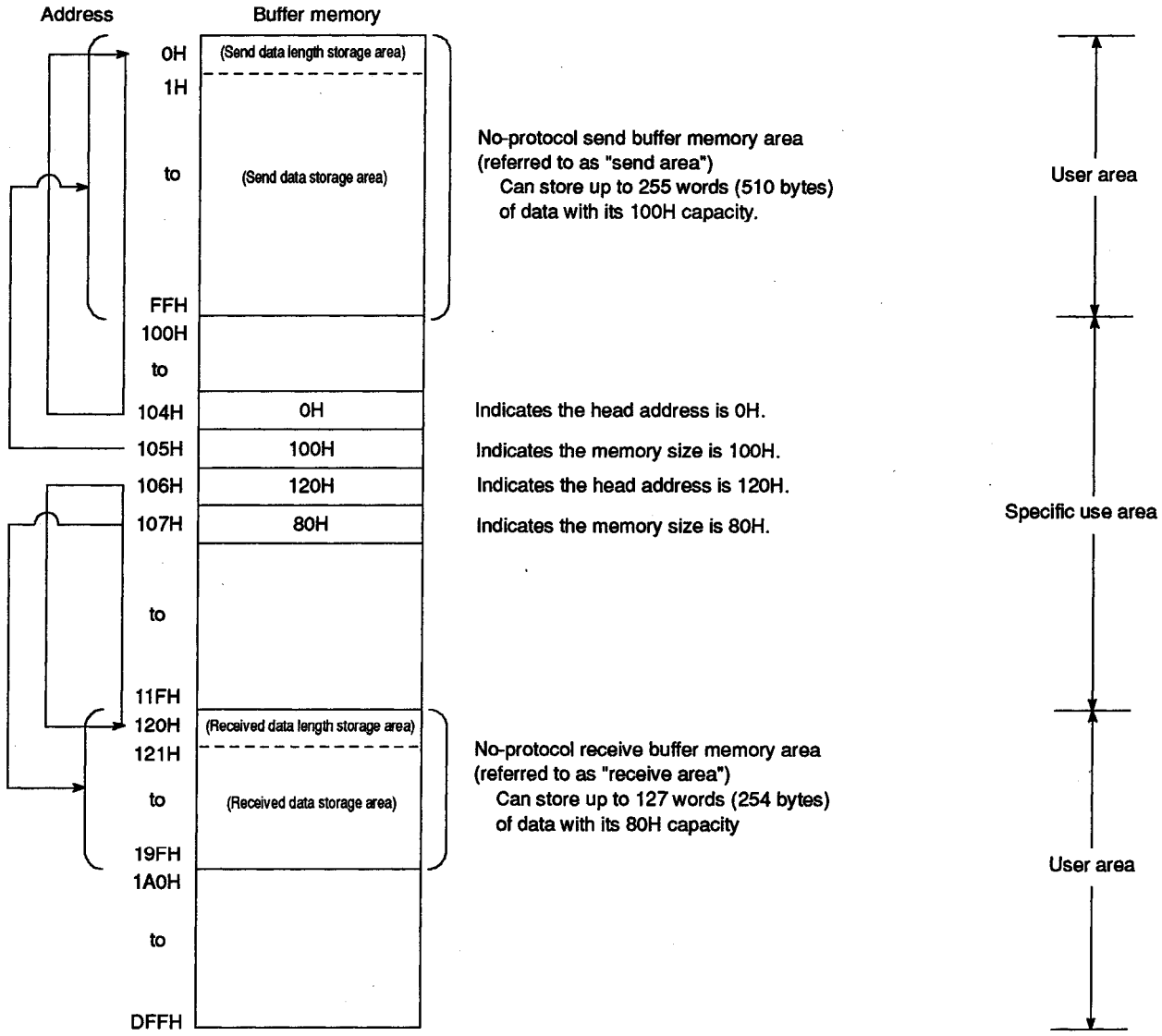
(3) Program examples to change the default values in the buffer memory specific use area

The following are program examples to change default values in the buffer memory specific use area.
Incorporate necessary programs only.
In the following examples, the I/O signals of the computer link module are 80H to 9FH which are handled by the PC CPU.

SETTING METHOD (Computer link module I/O signals: 80H to 9FH)



*1 The following diagram shows the locations of the send area and the receive area according to the above examples.



REMARK

Refer to Section 3.10 for details of the user area and the specific use area.

(4) The following explains the restrictions on modules, and the timing and procedure for changing the settings to continue data receive processing by changing the settings for receiving the data in the no-protocol mode after starting data communication.

(a) Type names of applicable computer link module

Data receive processing can be continued after changing the settings of the receive completion code and data length after starting data communication.
( ): Software version of the corresponding module.

- AJ71UC24 (Version M and later)
- A1SJ71UC24-R2, A1SJ71UC24-R4, A1SJ71UC24-PRF (Version M and later)
- A1SCPUC24-R2 (Version A and later)
- A2CCPUC24, A2CCPUC24-PRF (Version K and later)

(b) Settings that can be changed

Among the initial settings at the buffer memory of the computer link module, the following setting can be changed after starting data communication.

- No-protocol receive completion code (buffer memory address: 100H)
- No-protocol receive completion data length (buffer memory address: 108H)

* Conventionally, the setting could be changed only at the start up of the module (computer link module ready signal (Xn7) going from OFF to ON).
For other settings, change them at the start up of the computer link module as made previously.

(c) Timing and procedure for changing the setting

1) Timing

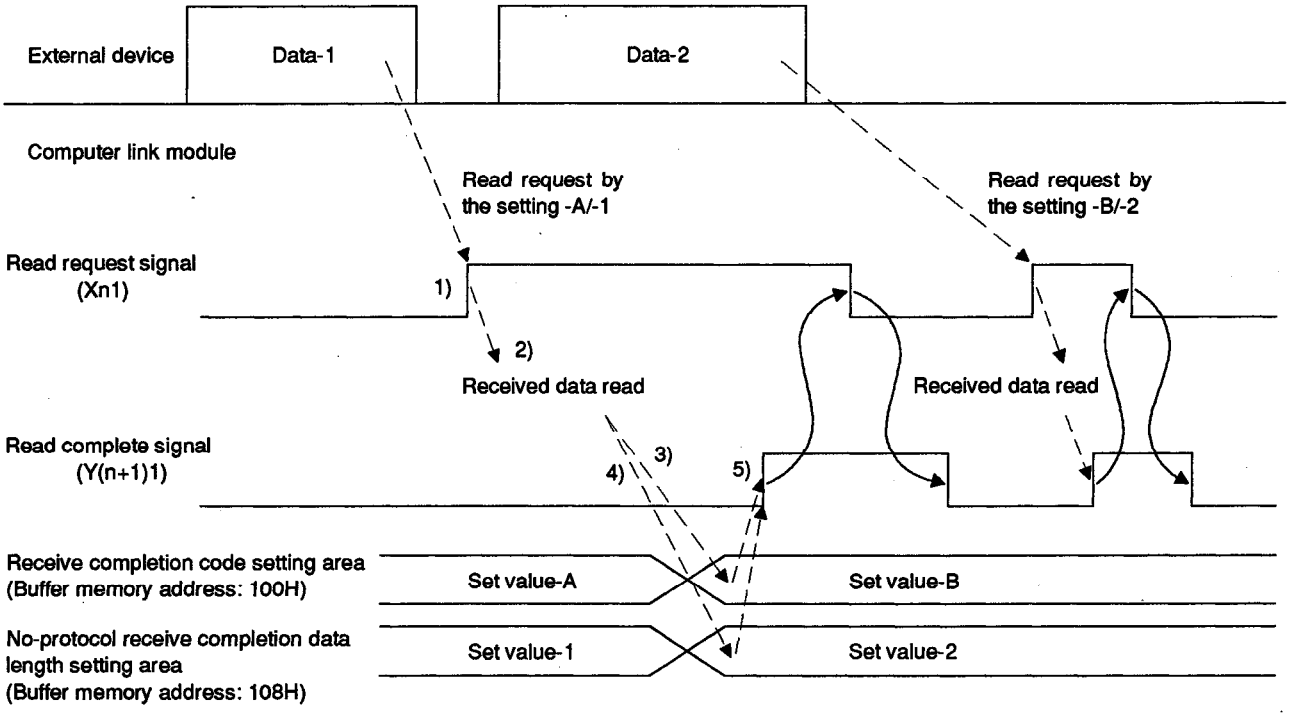
To change the receive completion code and/or the receive completion data length after starting data communication, do so when the I/O signals, related to reading of the receive data, between the PC CPU and the computer link module are in the state indicated below.

- Receive data read request signal (Xn1) : ON
- Receive data read completion signal (Y(n+1)1) : OFF

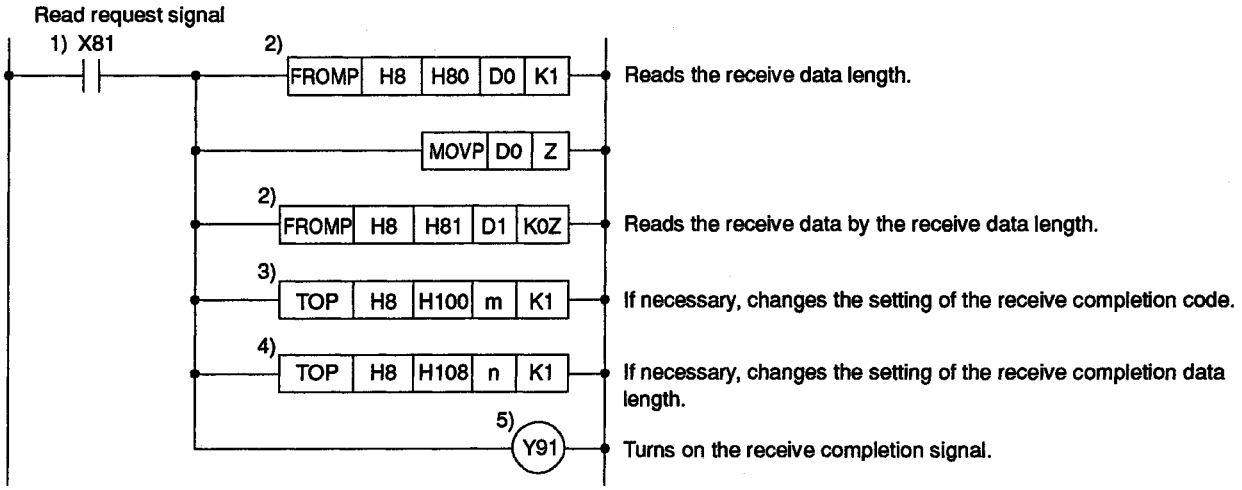
2) Procedure

Tun ON the receive data read completion signal after executing the following when the receive data read request signal is turned ON in response to the receiving of the data from an external device.

- Reading the received data.
- Changing the no-protocol receive completion code and the no-protocol receive completion data length.



(Program example .....Computer link module I/O signals: 80H to 9FH)



6.2.5 Precautions during data communications

(1) Conditions when the computer link module transmission sequence is initialized

The transmission sequence is initialized in the following cases:

- When the power is turned ON, the PC CPU is reset by the reset switch, or the mode is switched
- If the CD signal is turned OFF during send or receive processing when CD terminal check is enabled on the RS-232C in full duplex communications.

(2) Using a FROM/TO instruction to access the computer link module

A FROM/TO instruction to access the computer link module from a PC CPU should only be used when absolutely necessary.
This is because when data is transmitted from the computer link module to an external device at the same time a FROM/TO instruction from a PC CPU to the computer link module is executed, the FROM/TO instruction has priority and is processed first.
This delays the data send time from the computer link module by how long it takes to process the FROM/TO instruction.

(3) Data code conversion

In the no-protocol mode, data is not converted to ASCII code in the computer link module. If ASCII or JIS codes are required, the data must be processed into ASCII or JIS code in the PC CPU or an external device.

(4) Occurrence of framing errors on the external device

A framing error might occur at an external device while nothing is sent from the computer link module to the external device via the RS-422 interface. (Refer to Section 3.8.3.)
To prevent this, add an identification code to the head of sent data using a sequence program so that the head of the data processed by the computer link module can be recognized as such.
Before conducting data communications via the RS-422 interface, refer to the interface specifications of the computer link module shown in Section 3.8.3.

(5) Communications with the computer in multidrop link

In the 1:n multidrop link, the data sent from the computer is received by each computer link module.
The message must contain the objective PC CPU where the data is sent and, at the same time, it is necessary to write the sequence program that ignores the received data addressed to other stations.

(Message example)

STX	Space	Station number 0	Station number 2	Data length (Binary data)	Data	CR	LF
(02H)	(20H)	(30H)	(32H)			(0DH)	(0AH)

(6) Combined use with dedicated protocols

When data communications is executed in the no-protocol mode, if the computer link module mode setting switch (refer to Section 4.2.1) is placed in any position from 1 to 8, or any mode number from 1 to 8 is designated by the mode switching function, data communications with the other interface can be executed using the dedicated protocol.
Data communications cannot be executed by setting one interface to the bidirectional mode and the other to the no-protocol mode.

(7) Operations prohibited during data send

Do not attempt either or the following while data is being sent from a compute link module to an external device, otherwise the computer link module will stop sending the data and the send completed signal (Xn0) will come ON.

(Prohibited operations)

- When executing data communication via an RS232C interface, switching the CD signal OFF when "CD terminal check enabled" has been set in buffer memory address 10BH and the external device is controlling the CD signal of computer link module. (The computer link module will also fail to send data and will switch the send completed signal ON if the CD signal is OFF when sending is started.)
- Issuing a receive data clear request.

(8) Changing the receive completion code and the receive completion data length

To continue data receive processing by changing the setting for data receive processing (receive completion code and receive completion data length) after starting the data receive processing, change the setting at the timing and in the procedure described in 6.2.4, item (4).

6.3 Pre-Operation Check Points

Refer to Section 4 for the start-up procedure of the computer link module in the no-protocol mode.
Recheck the following items before starting operations to secure normal performance of the computer link module.

	Check Item	Reference Sections
(1)	Switch settings	Section 4.2
(2)	Operation of the computer link module	Sections 4.5 and 4.8.1
(3)	Connection with an external device	Section 4.6
(4)	Connection/setting of a terminal resistor	Section 4.7

POINT

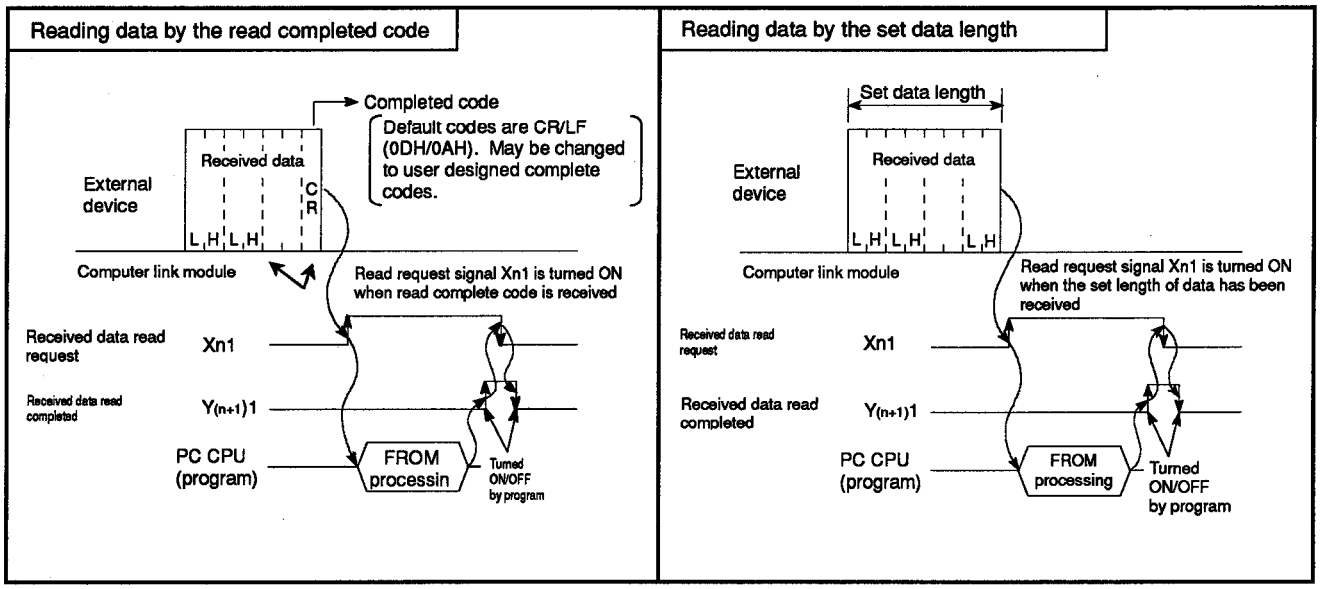
Unless these items are correctly set in the no-protocol mode, normal computer linking cannot be established.
Recheck the above items before starting operations.

6.4 Data Receiving Procedure in the No-Protocol Mode (External Device → Computer Link Module)

"Data receiving" is data processing where the computer link module stores data received from an external device in the buffer memory data receive area and then a sequence program reads the stored data using the FROM instruction.

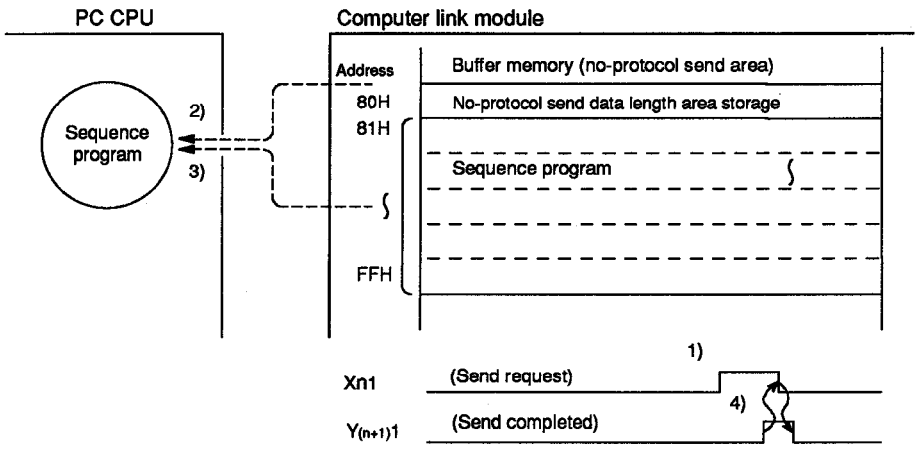
6.4.1 Data receive procedure

The following diagrams show the data receive process to read out data to the PC CPU in the no-protocol mode.



6.4.2 Data receive program

Shown below are basic sequence programs and programming examples to read data from the buffer memory (receive area) to the PC CPU.
Data is read from the buffer memory receive area (default: 80H to FFH).



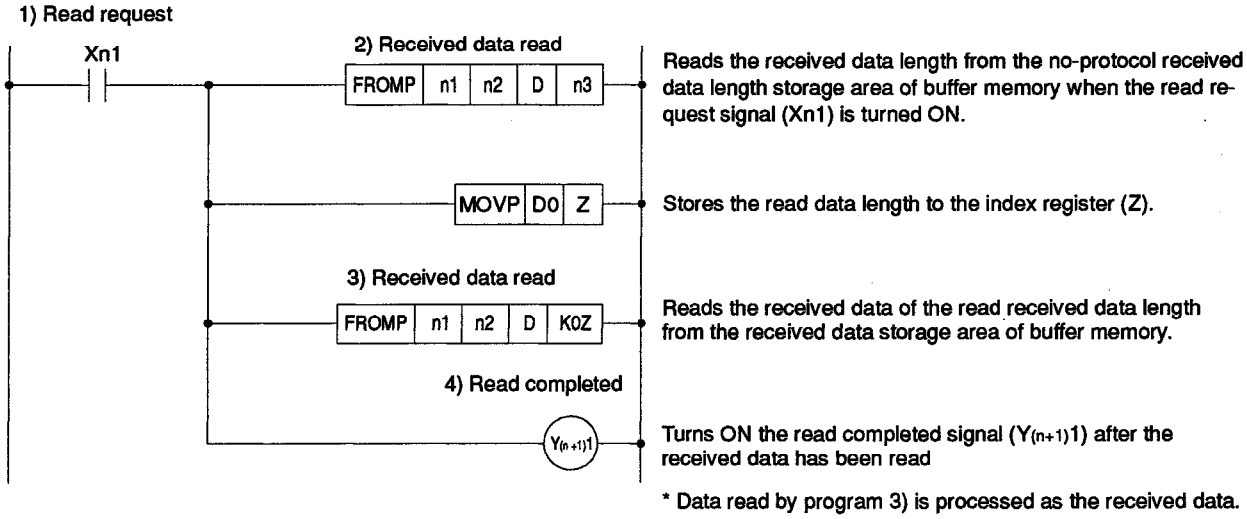
1) to 4) in the diagram corresponds to 1) to 4) in (1)(a) and (2)(a) of this section.

(1) Basic sequence program to use application instructions (FROM, FROMP, DFRO, DFROP instructions)

(a) Format to read from the receive area

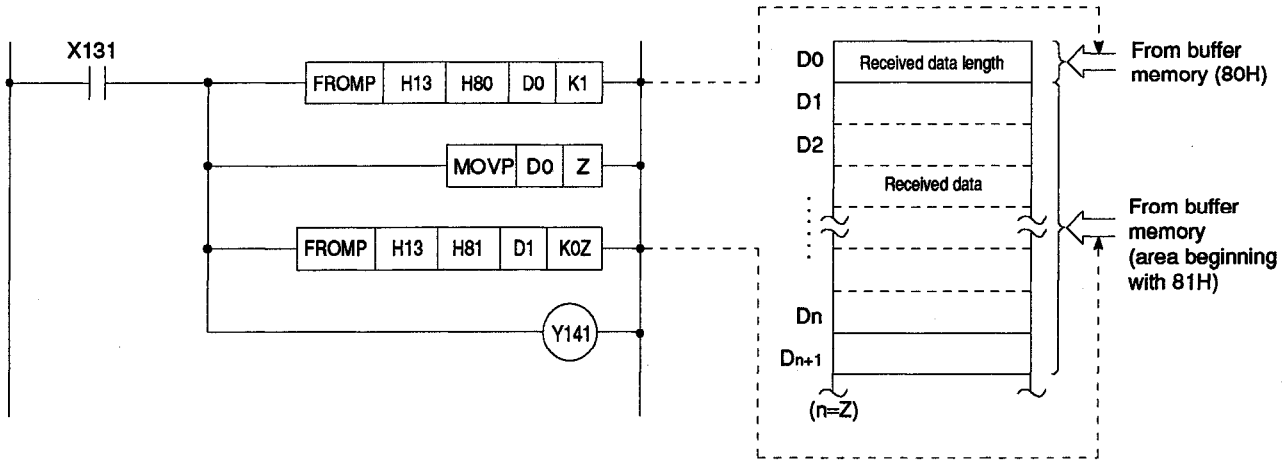
Refer to the Programming Manual (Common Instructions) for details.

FORMAT



(b) Programming example

To read the data of (n+1) words from the area, beginning with buffer memory address 80H, to the area beginning with D0 when the computer link module I/O numbers are allocated to 130H to 14FH.

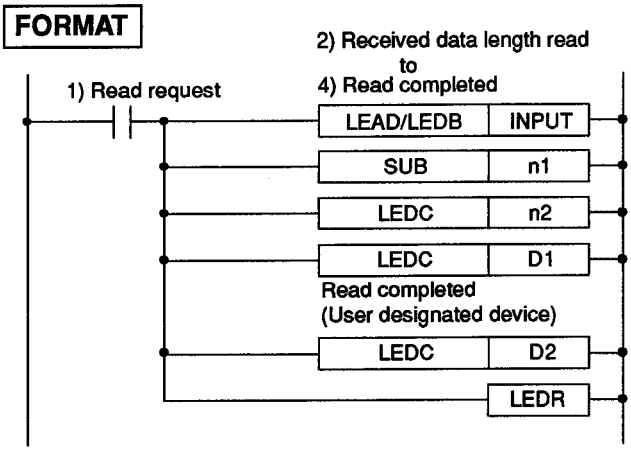


(2) Basic sequence program to use the dedicated instruction (INPUT in-struction)

(a) Format to read from the receive area

Refer to the Programming Manual (Dedicated Instructions) for details.

FORMAT



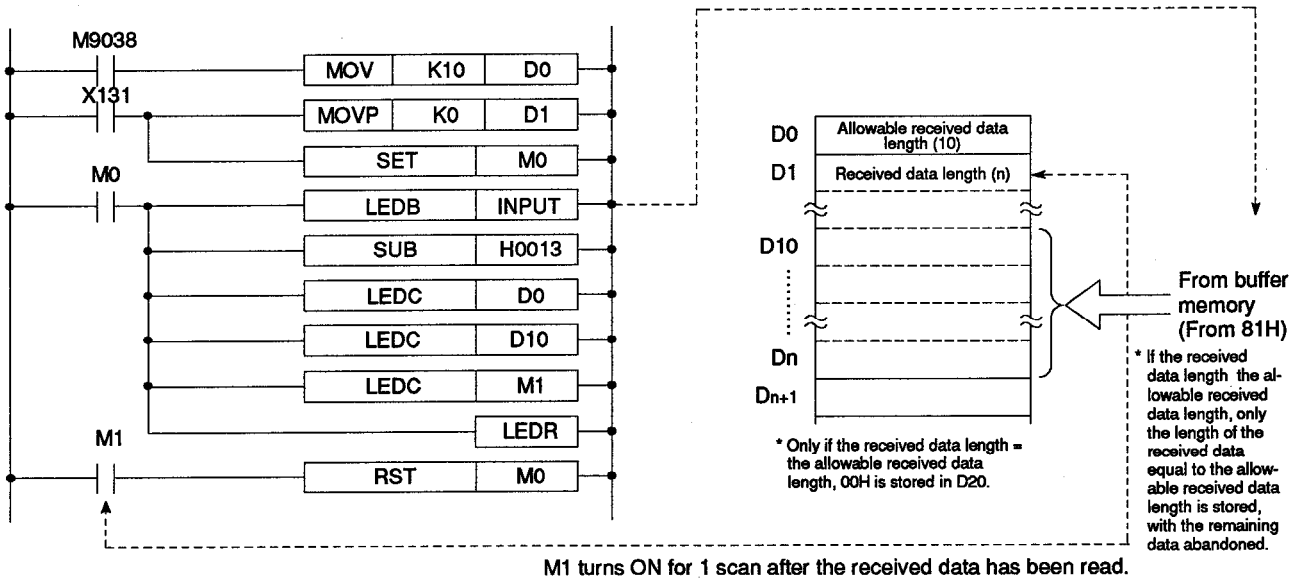
Reads the received data not exceeding the allowable received data length (user designated) from the received data storage area of buffer memory while the read request signal is ON or when it goes on.

The user designated read completed signal turns ON for 1 scan after the received data has been read.

* Data read by programs 2) to 4) is processed as the received data.

(b) Programming example

To transmit received data not exceeding 10 words to the memory area from D10 when the I/O signals of the computer link module are allocated to 130H to 14FH.



POINT

If the dedicated instructions are used, set the number of I/O points (F32 points) and the model name of the applicable module (as given in the table below) to the area for the slot to mount the computer link module in the I/O allocation parameters to be written to the PC CPU.
Refer to the Programming Manual (Dedicated Instructions) for details.

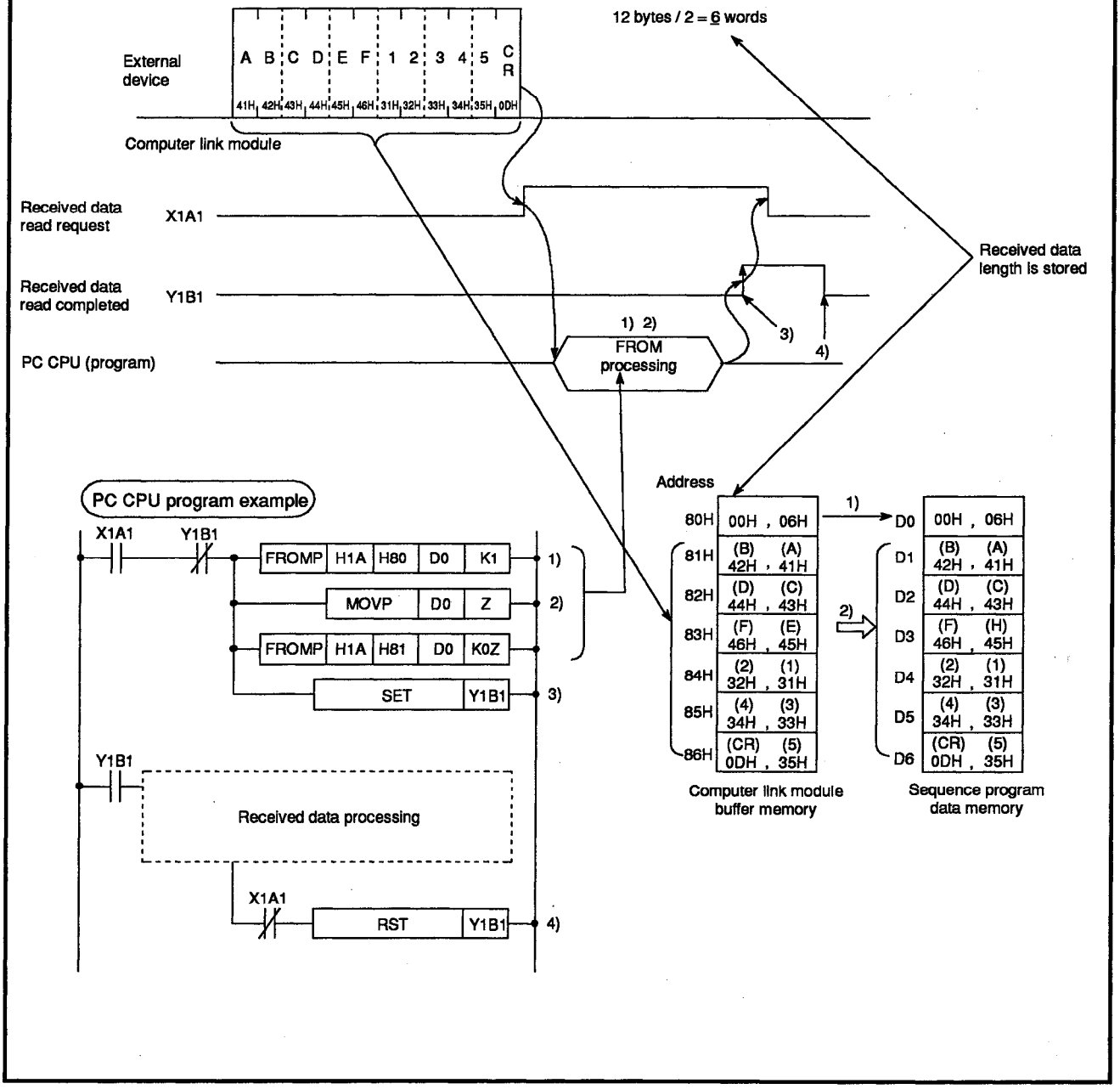
PC CPU Mounted with a Computer Link Module	Mounted Computer Link Module					
	AJ71UC24	A1SJ71UC24-R2/R4/PRF	A1SJ71C24-R2/R4/PRF	A1SCPU C24-R2	A2CCPU C24	A2CCPU C24-PRF
AnUCPU	AJ71UC24		AJ71C24S3	—		
AnACPU	AJ71C24S3			—		
Other than AnA/AnUCPU	No setting necessary; dedicated instructions unusable					

(3) Application example

The following describes examples to store received data in the read data register using application instructions.
Even if a dedicated instruction is used, the same result can be obtained, although different sequence programs are used.

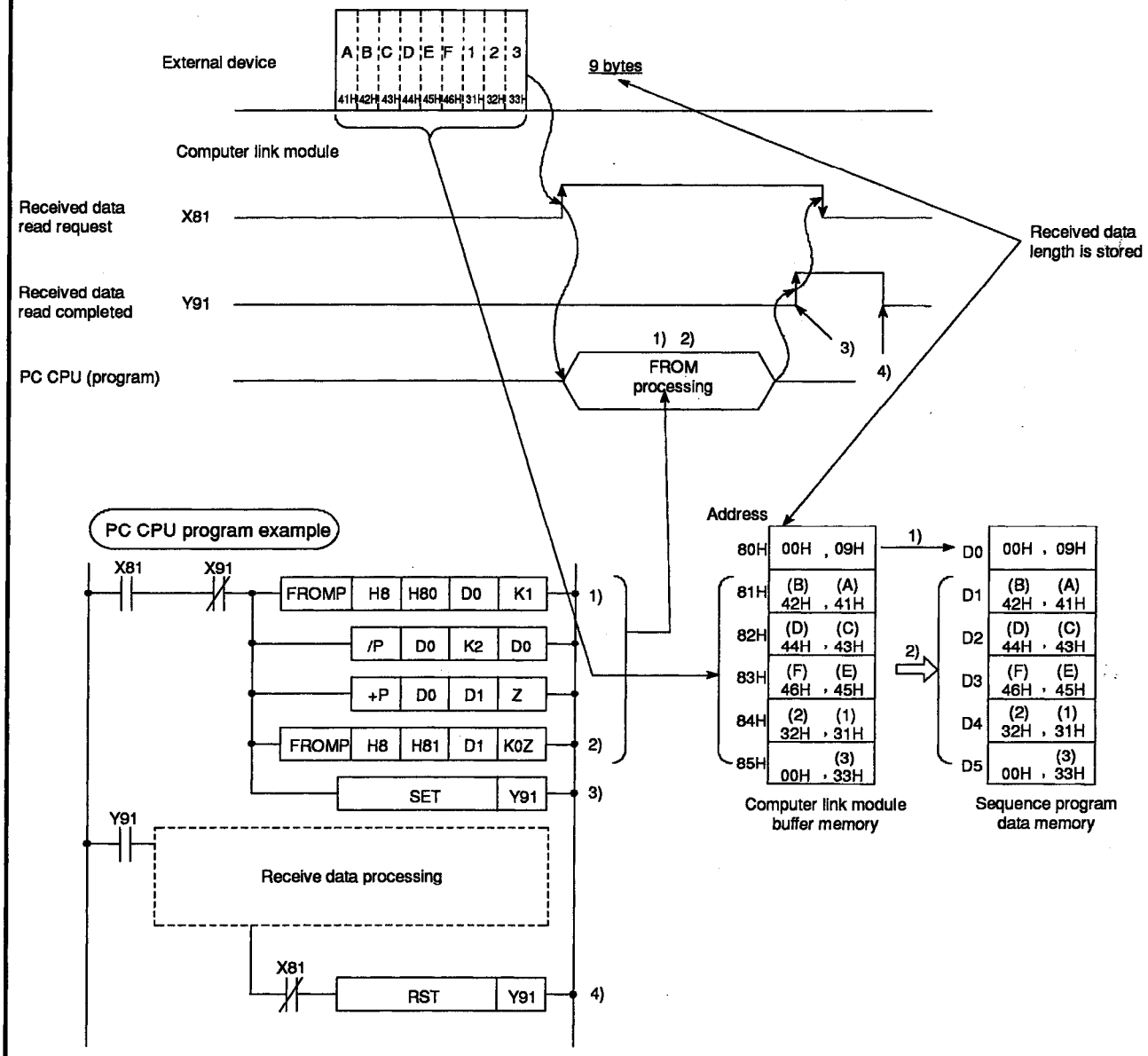
Example 1: By receiving completed code, in word units (buffer memory allocation: default)

To store "ABCDEF12345" CR, received from an external device, to D0 to D6 of the PC CPU. (Computer link module I/O addresses : 1A0H to 1BFH)



Example 2: By receiving the set length of data, in byte units (receive area allocation: default)

To receive "ABCDEF123" from an external device and store it to D0 to D5 of the PC CPU with the receive complete data length set as "9". (Computer link module I/O addresses: 80H to 9FH)



POINTS

(1) Even if transmission data units are set to byte units, the FROM instruction in a sequence program operates in word units. Therefore, the length of receive data must be converted to the number of buffer memory points (word units).

In the above example, 9 bytes of data must be converted into 5 words (9 ÷ 2 = 4.5...5).

(2) When an odd number of bytes of data is received, the higher 8 bits of the last address read by the FROM instruction are "00H".

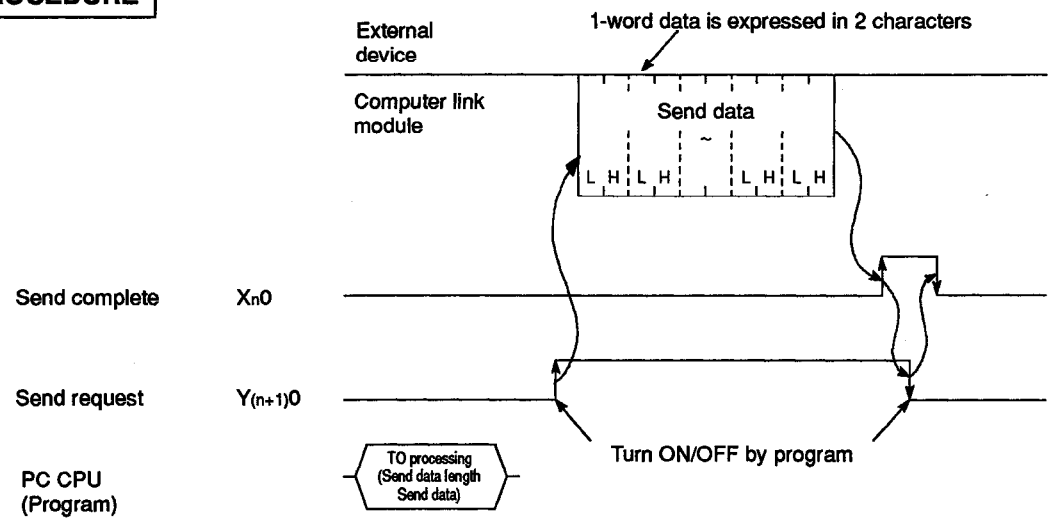
6.5 Data Sending Procedure in the No-Protocol Mode (Computer Link Module → External Device)

"Data sending" is data processing where, in response to the send request signal (Y(n+1)0) from the PC CPU, the computer link module transmits data written to the buffer memory send area by a sequence program using the TO instruction to an external device.

6.5.1 Data sending procedure

The following describes the procedure to send data written to the send area to an external device.

PROCEDURE



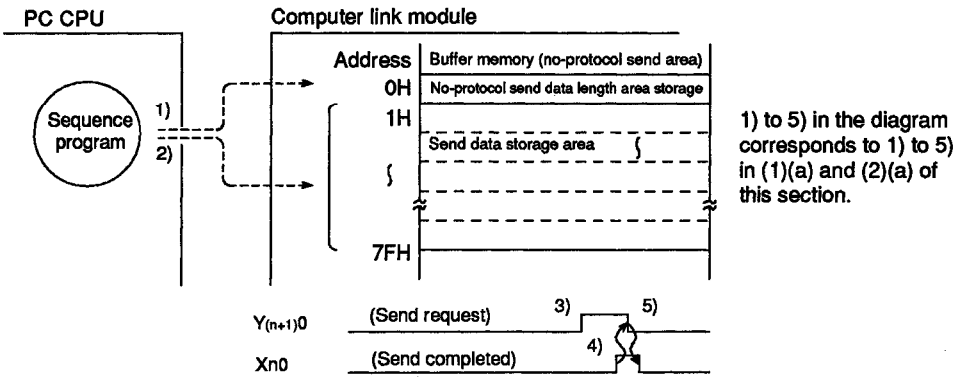
POINTS

(1) An error occurs if the send data length is greater than the send area, or equal to "0". In this case, "1" is written as transmission error information to the most significant bit (bit 15) of address 101H (error LED display area) in the specific use area. (Refer to Section 6.2.2 (3).)

(2) The send data length written to the head address of the send area is not transmitted.

6.5.2 Data sending program

Shown below are basic programs and programming examples to send data from the PC CPU to an external device.
Data is written to the buffer memory send area (default addresses: 0H to 7FH).



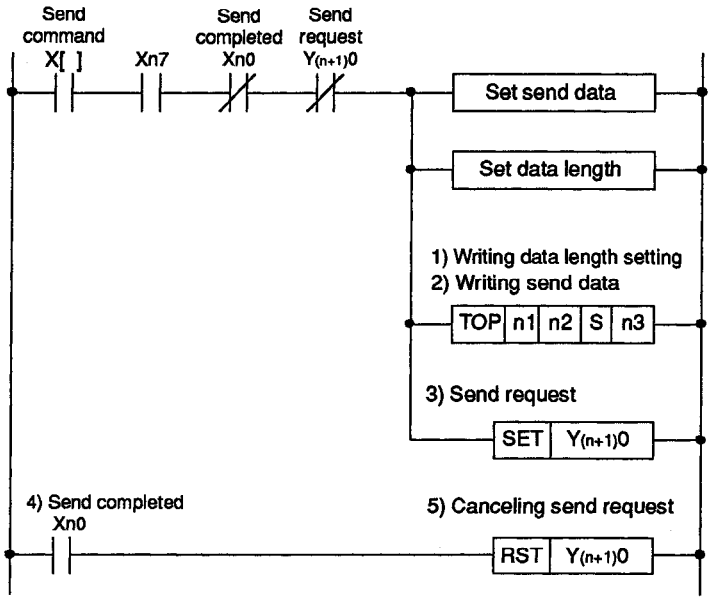
1) to 5) in the diagram corresponds to 1) to 5) in (1)(a) and (2)(a) of this section.

(1) Basic sequence program to use application instructions (TO, TOP, DTO, DTOP instructions)

(a) Format to write to the send area

Refer to the Programming Manual (Common Instructions) for details.

FORMAT



Sets the send data and send data length when the send command signal (X[ ]) is turned ON.

* Turn OFF the send command signal before the send completed signal (Xn0) is turned ON.

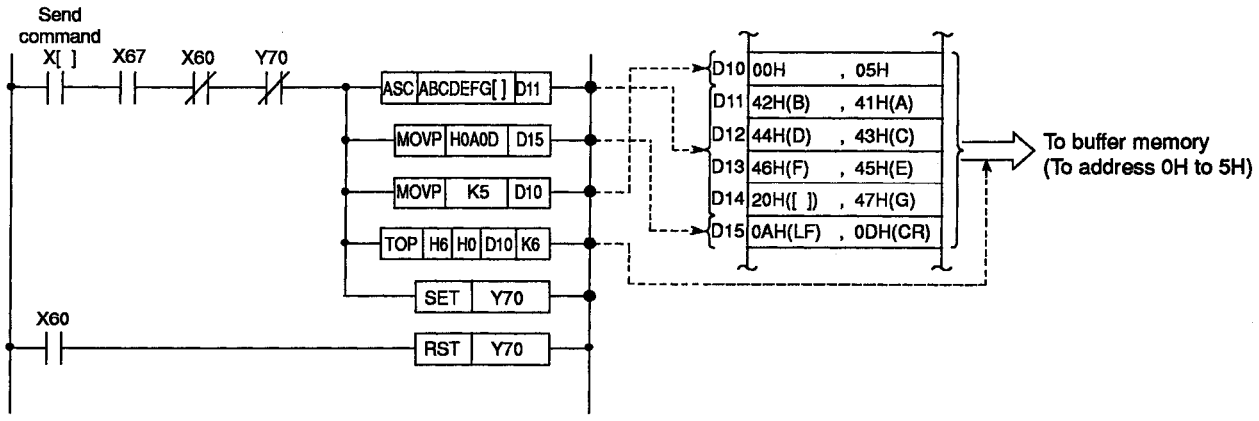
Writes the send data length and send data to the buffer memory.

Turns ON the send request signal (Y(n+1)0) after the data is written.

Turns OFF the send request signal when the data send completed signal (Xn0) is turned ON.

(b) Programming example

To transmit 5-word data after writing "ABCDEFG_CR.LF" to the buffer memory area from 1H when the computer link module I/O numbers are allocated to 60H to 7FH. (The designated unit of data is "word" in the example.)



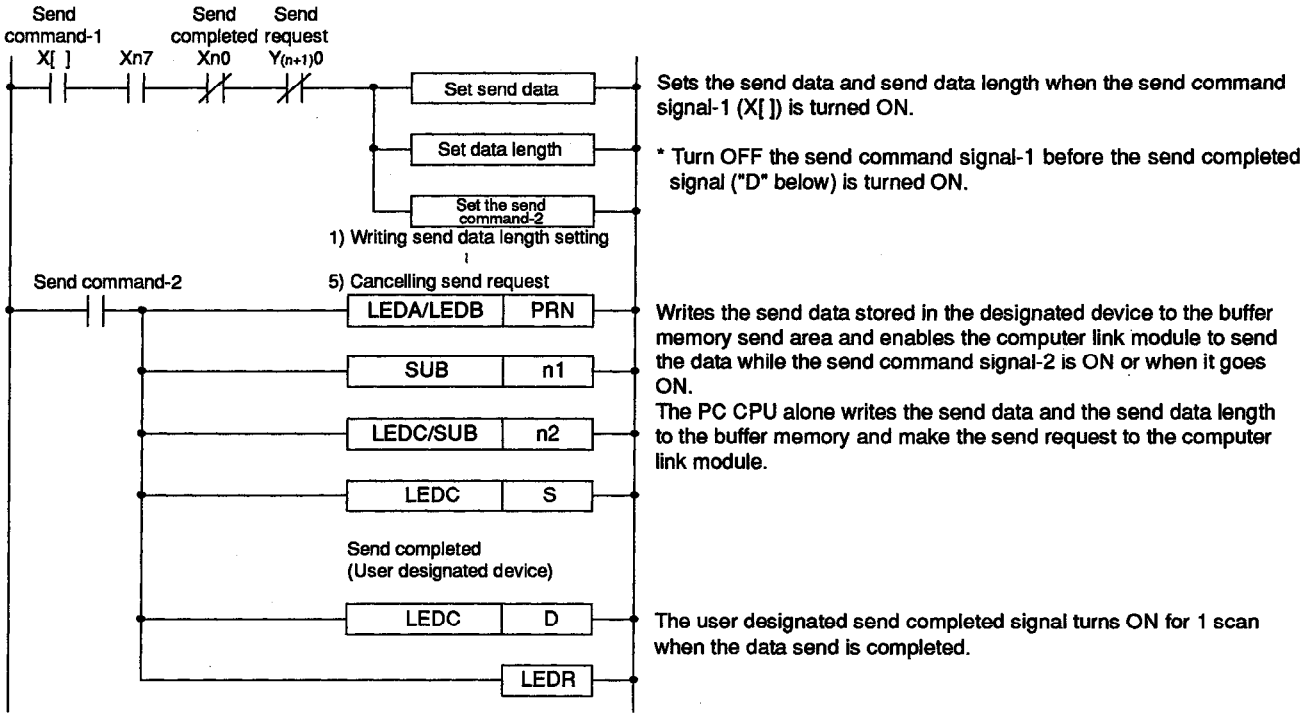
(2) Basic sequence program to use dedicated instructions (PRN, PR instructions)

[When to use the PRN instruction] .....To transmit set length of data in byte/word units

(a) Format to write to the send area

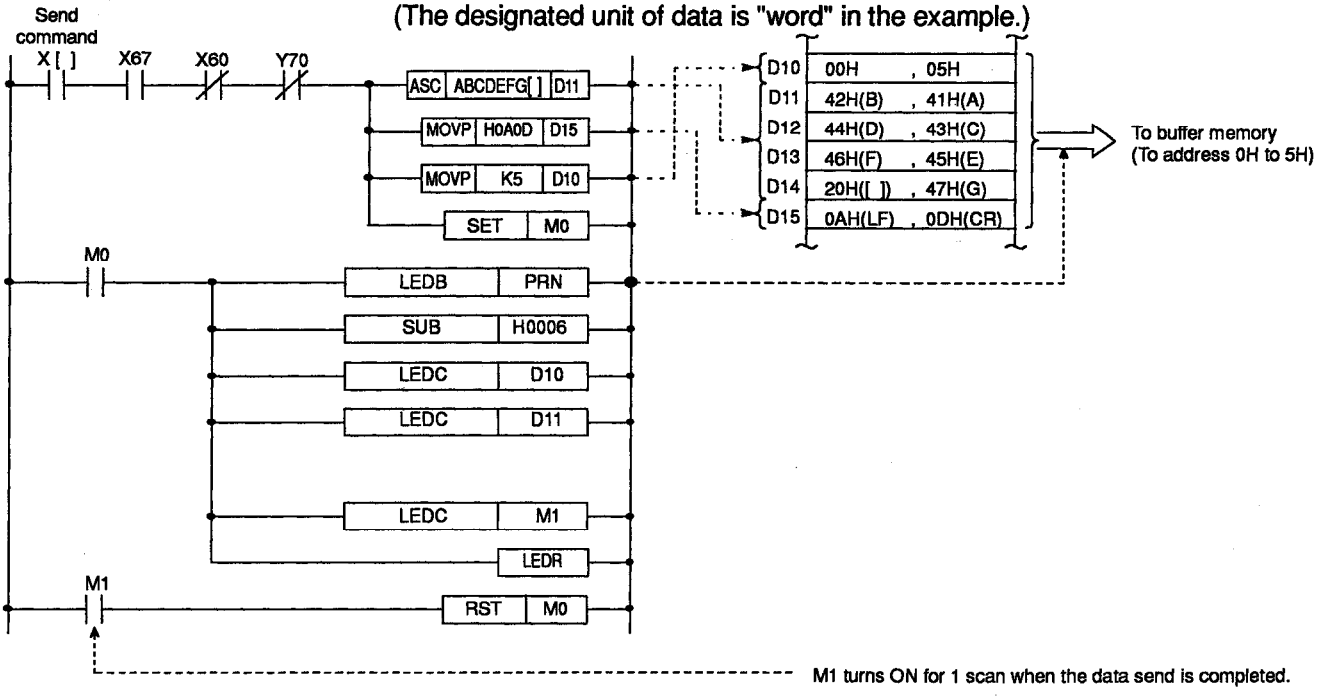
Refer to the Programming Manual (Dedicated Instructions) for details.

FORMAT



(b) Programming example

To transmit 5-word-long data after writing "ABCDEFG_CR.LF" to the buffer memory when the I/O numbers of the computer link module are allocated to 60H to 7FH.
(The designated unit of data is "word" in the example.)

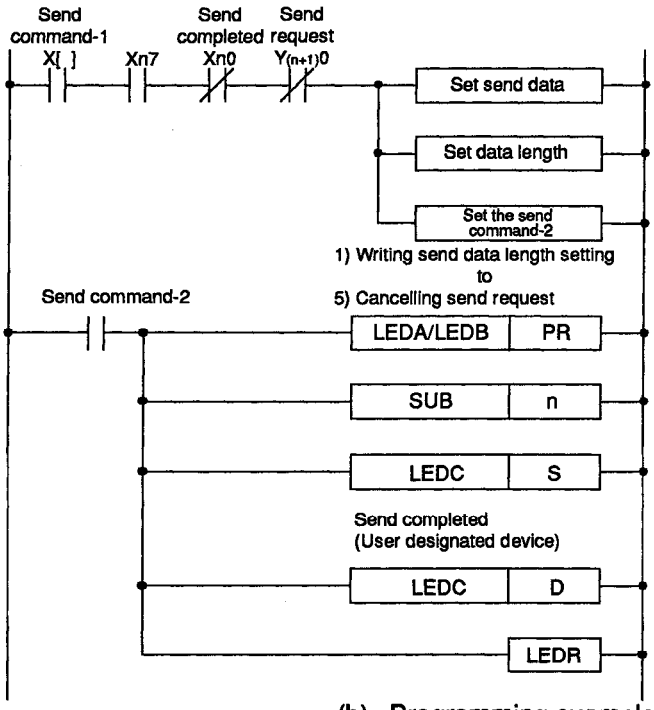


[To use the PR instruction] .....To transmit data up to the address immediately before the 00H code.

(a) Format to write to the send area

Refer to the Programming Manual (Dedicated Instructions) for details.

FORMAT



Sets the send data and the 00H code when the send command signal-1 (X[ ]) is turned ON.

(Being the send complete code, the 00H code cannot be included in the send data.)

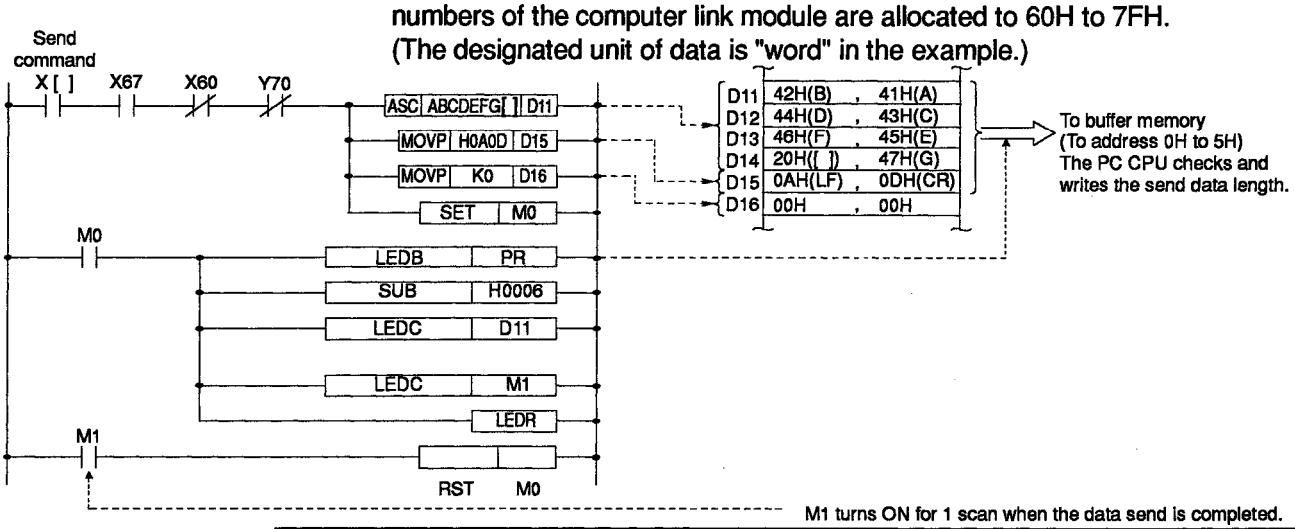
* Turn OFF the send command signal-1 before the send completed signal ("D" below) is turned ON.

Writes the send data stored in the designated device to the buffer memory send area and enables the computer link module to send the data while the send command signal-2 is ON or when it goes ON.
The PC CPU alone writes the send data and the send data length to the buffer memory and make the send request to the computer link module.

The user designated send completed signal turns ON for 1 scan when the data send is completed.

(b) Programming example

To transmit "ABCDEFG_CR.LF" to the buffer memory when the I/O numbers of the computer link module are allocated to 60H to 7FH.
(The designated unit of data is "word" in the example.)



POINT

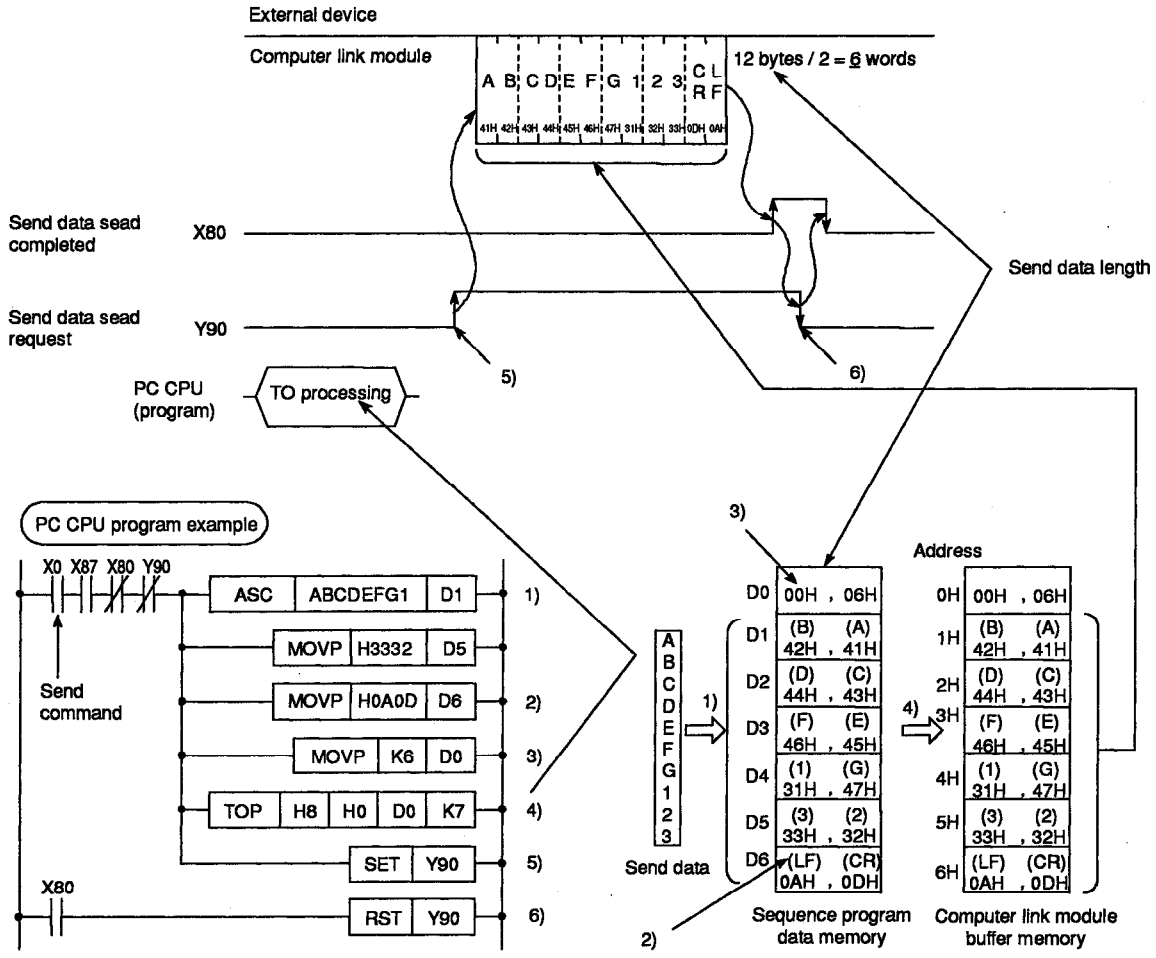
If the dedicated instructions are used, set the number of I/O points (F32 points) and the model name of the applicable module (as given in the table below) to the area for the slot to mount the computer link module in the I/O allocation parameters to be written to the PC CPU.
For the applicable model names, refer to POINT in Section 6.4.2(2).
Refer to the Programming Manual (Dedicated Instructions) for details.

(3) Examples of application instructions

The following describes examples to store received data in the read data register using application instructions.
Even if a dedicated instruction is used, the same result can be obtained, although different sequence programs are used.

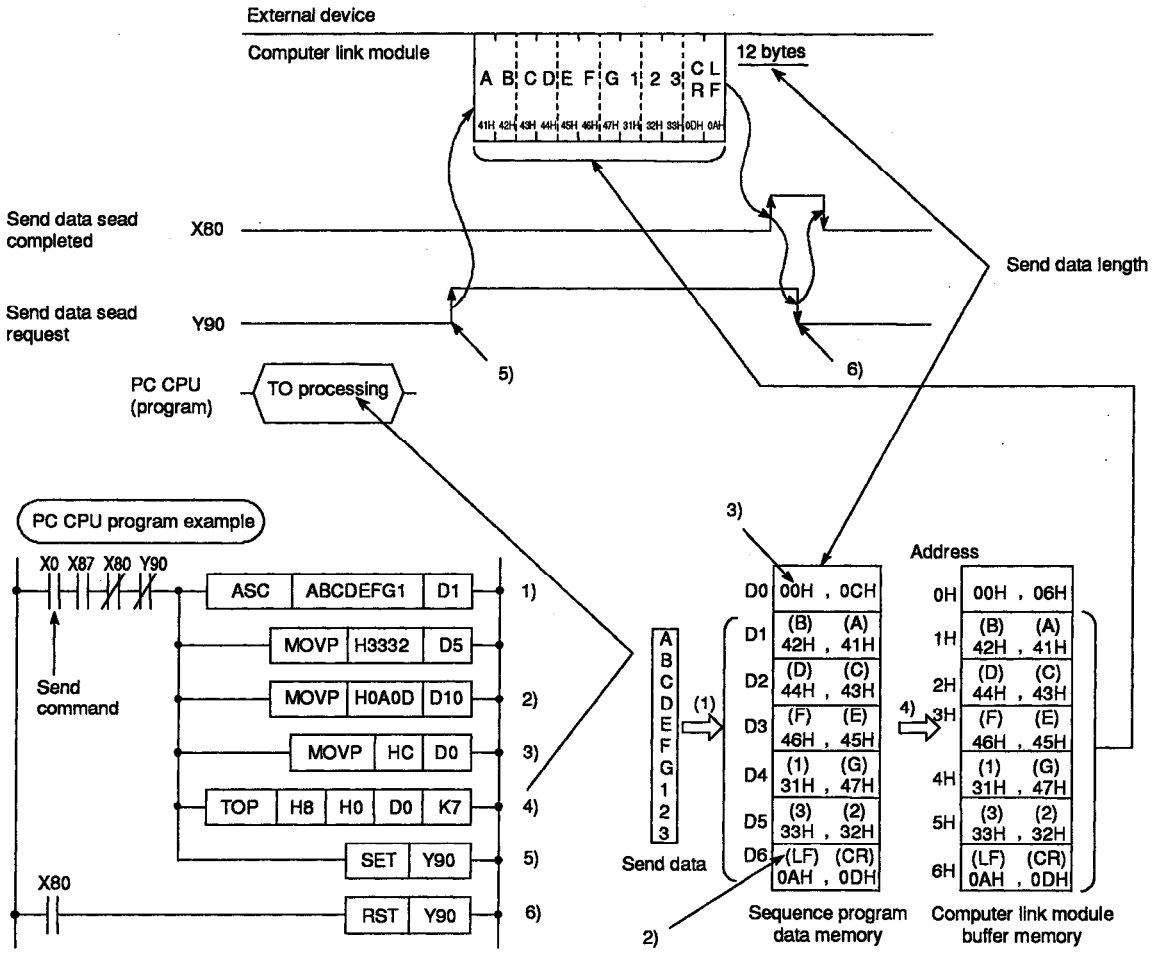
Example 1: Transmitting data in word units (send area allocation: default)

To write "ABCDEFG123" CR, LF by sequence program and transmit it to an external device (Computer link module I/O addresses: 80H to 9FH)



Example 2: Transmitting data in byte units (send area allocation: default)

To write "ABCDEFG123" CR, LF by sequence program and transmit it to an external device (Computer link module I/O addresses: 80H to 9FH)



POINT

Even if transmission data units are set byte units, the TO instruction in a sequence program operates in word units. Therefore, the length of send data differs from the data length set with the TO instruction.

MEMO

7. COMMUNICATIONS WITH A COMPUTER IN THE BIDIRECTIONAL MODE

This section describes the method applied to link a computer with a PC CPU for bidirectional transmissions (supported in bidirectional mode).
Bidirectional communications with a computer is possible only when a computer and a computer link module are linked in a 1 : 1 ratio.

Always read this section when the RS-422 and RS-232C interfaces are used in the bidirectional mode with the mode setting switch of the computer link module being set to any position from "1" to "8" or with the mode switching function being used to set the mode number to any of "1" to "8".
It is not necessary to read this section when the interfaces are used with a dedicated protocol and in the no-protocol mode.

POINT

Buffer memory used in the bidirectional mode
In sections other than this, buffer memory used in the bidirectional mode is described as the buffer memory used for the no-protocol mode. Because the application purposes are the same, simply think of the "no-protocol mode" as the "bidirectional mode".
Examples:

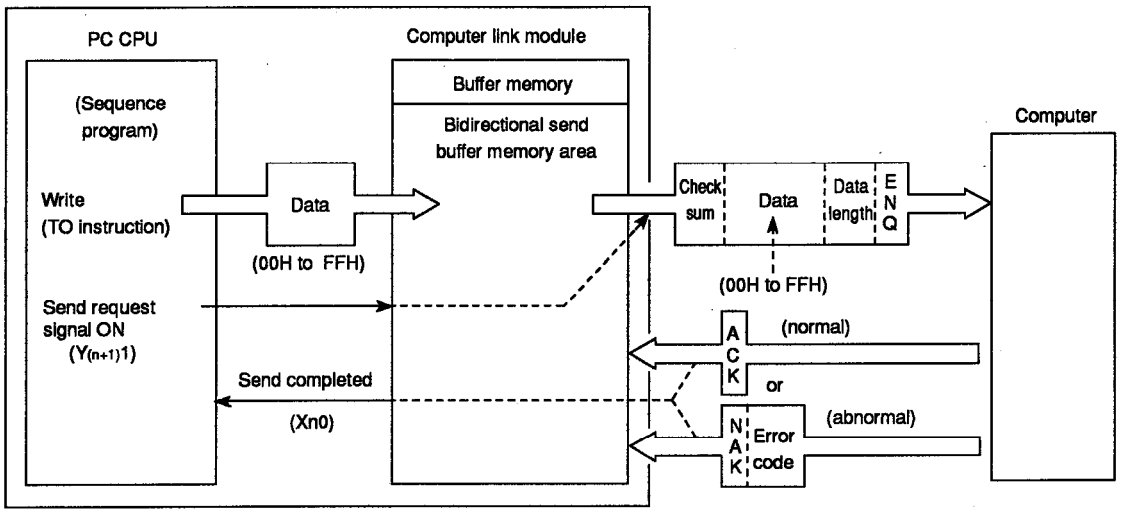
- No-protocol mode send area
  → Bidirectional mode send area
- No-protocol send buffer memory head address setting area
  → Bidirectional send buffer memory head address setting area

7.1 Data Flow in the Bidirectional Mode

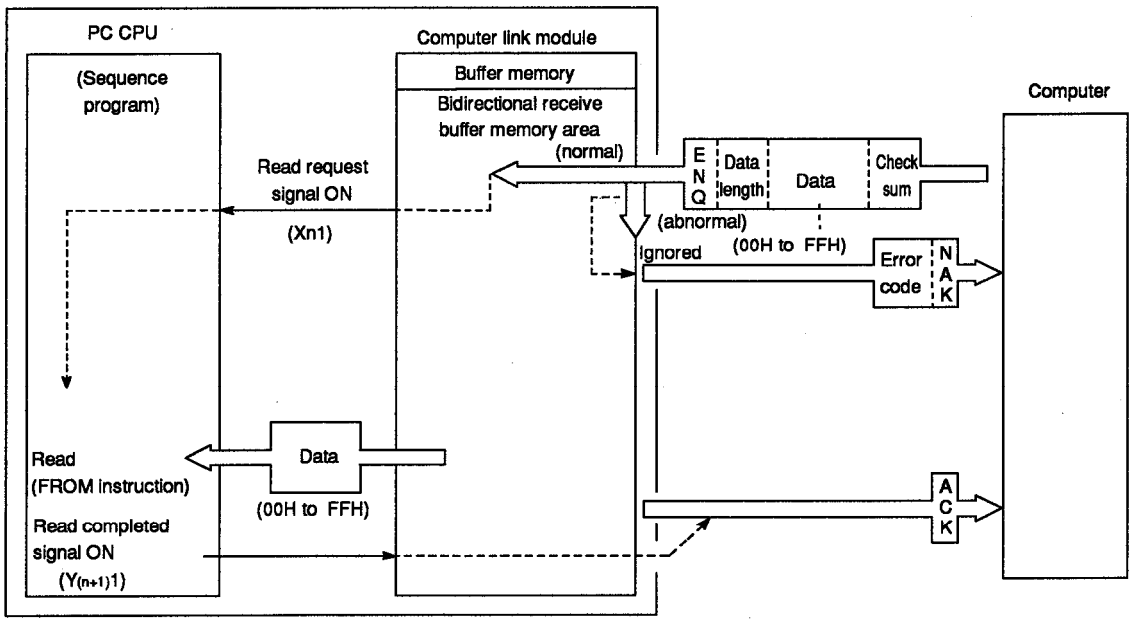
The following figures show schematic data transmission processes between a PC CPU and a computer.

When data is sent to a computer, the computer link module adds a control code etc. to the send data designated by the PC CPU and sends it to the computer. (The send data designated by the PC CPU is sent without changing its code.)
When data is received, the computer link module removes a control code etc. from the received message and transfers it to the PC CPU without changing its code.
Therefore, the BIN data (numeric data) can be handled as the transmission data.

(1) When a PC CPU sends data to a computer



(2) When a PC CPU reads data received from a computer



7.2 Programming Hints

Described below are precautions for preparing a program for a computer link in bidirectional mode.

POINT

It is impossible to use the functions available in bidirectional mode together with the functions available in no-protocol mode mentioned in Section 6.
The computer link in bidirectional mode is enabled by making settings with the mode setting switch on the computer link module (Section 4.2.1) and the bidirectional mode designation area in the buffer memory specific use area (Sections 3.10 and 7.2.6 (2) (f)).

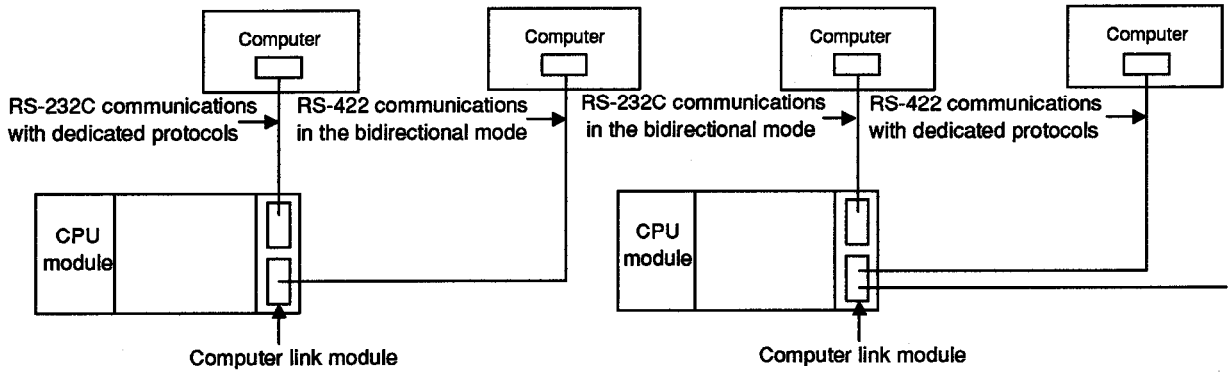
7.2.1 System configuration and communications mode for bidirectional mode communications

(1) System configuration and the computer link module mode setting

Data communications in the bidirectional mode is possible only in the system where a computer and the computer link module are linked in a 1 : 1 ratio. The mode setting switch in the computer link module should be set in any position of "1" to "8" (refer to Section 4.2.1), or the mode switching function can be used to set the mode number from "1" to "8" (refer to Sections 3.6 and Section 12).

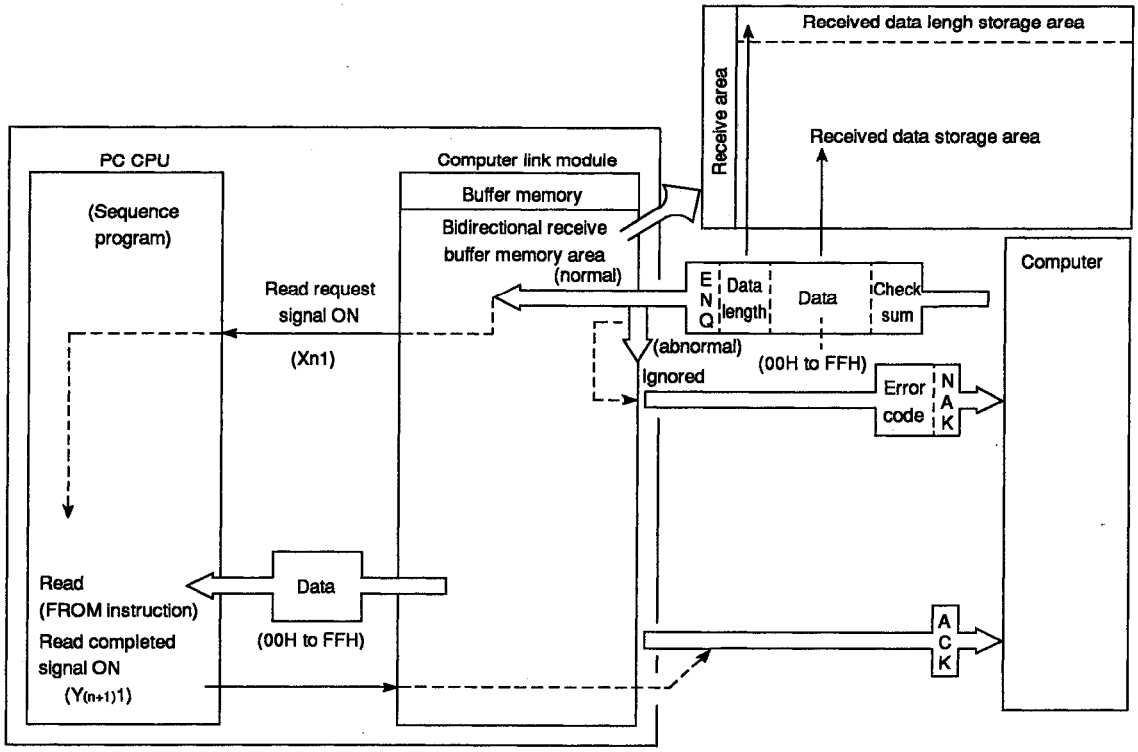
(2) Usable with dedicated protocols

When data communications is executed in the bidirectional mode, data communications using the dedicated protocol is possible with the other interface.



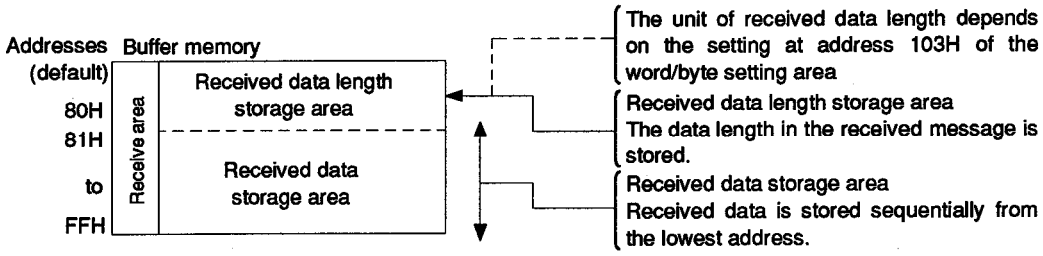
7.2.2 Receiving data from the computer

This section gives the items required to be understood to operate the PC CPU to read data received from the computer.



(1) Allocating the receive area

Receive area is a buffer memory area in the computer link module where the data and data length received from a computer are stored so that the PC CPU can read them.
Default settings of the receive area allocation are buffer addresses 80H to FFH.
The receive area allocation can be changed according to the purpose of data transmission, computer specifications, and receive data length.
(Refer to Section 7.2.6 (2) (c) for the changing procedure.)

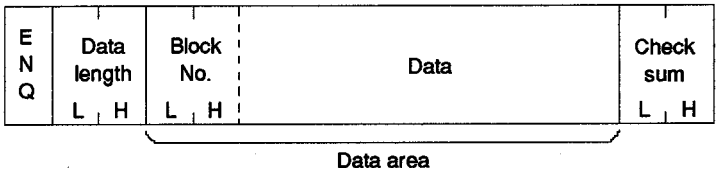


POINT

The length of data per one time of sending from a computer to the computer link module must be smaller than the size of the received data storage area in the computer link module.
(Received data storage area) >= (Data length received from a computer)
When it is necessary to send data whose length exceeds the size of the received data storage area, take either of the following methods.

(1) Changing the receive area
Change the address and size of the receive area.

(2) Dividing the send data
Divide the send data into parts so that each part is smaller than the receive area size. Add block numbers to such divided parts of data for identification.
(Message format example)



(2) Reading received data

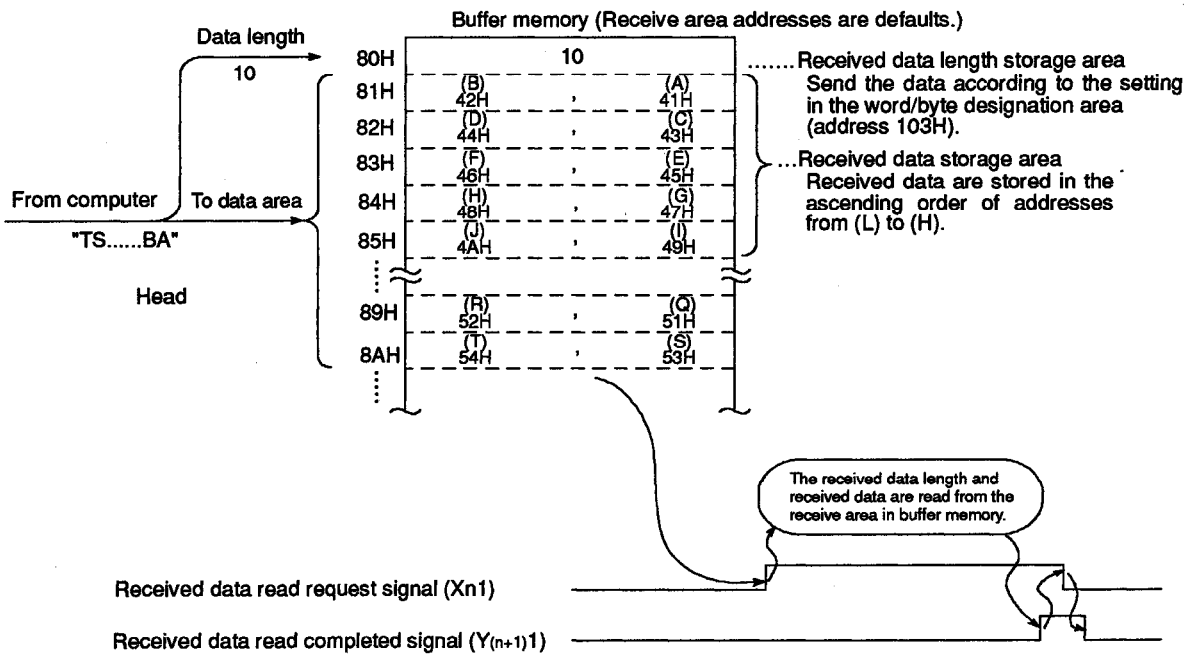
The computer link module issues a received data read request to the PC CPU by turning ON a received data read signal (Xn1) at the timing when the conditions are established as described below.

- When the data length in the received message is received and a data area which is equivalent to the word or byte length set with the bidirectional word/byte designation area (buffer address 103H) is received.
- When a check sum is handled, the condition is established when the above data area and a check sum are received.

(Example)
Word/byte designation area setting ..... Word unit
Data length in the received message..... 10

In this case, the computer link module issues a received data read request to the PC CPU when a data area of 10 words (or a data area of 10 words and a check sum of 1 word) has been received.

When a received data read request (Xn1) has been issued (ON), read the received data length and the data equivalent to the received data length by using a FROM instruction in a sequence program. And then, turn a received data read completed signal ($Y_{(n+1)}1$) from OFF to ON.



POINT

Set the data length in the sending and receiving message so that the same data length unit (word or byte) is used between the computer and the PC CPU.
The data length unit used with the PC CPU can be designated by setting it with the bidirectional word/byte designation area (address 103H) in buffer memory of the computer link module.

REMARK

If a CD signal is turned OFF or mode switching is performed during communications with a computer, received data will be cleared. The following gives precautions for the operation.

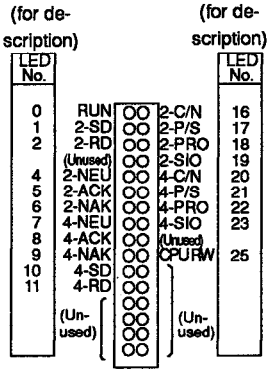
(1) If a CD signal is turned OFF:
- During data communications with a computer through an RS-232C channel when the CD terminal check function has been set, if a CD signal of the computer link module is turned OFF (4 ms or over) by the computer when a read request to the PC CPU has not been made, received data will be cleared. (The receive area will not be initialized.)
- If a CD signal is turned OFF when a read request to the PC CPU has been made, the computer link module continues to issue the read request to the PC CPU. (The received data for which the read request is made will not be cleared.)
  If the CD signal is (has been) turned ON after the received data read has been completed, the computer link module sends an ACK message to the computer.
- To restart data sending from the computer, turn ON the CD signal.

(2) If mode switching is performed:
- If the computer link module's operation mode is forcibly switched as described in Section 12, the computer link module turns into the initial start-up state and the received data will be cleared. (The receive area is initialized.)
- To perform subsequent data communications, setting with the specific use area in buffer memory is necessary.
- When restarting data sending from the computer, start sending after the computer link module READY signal (Xn7) has turned ON and settings with the specific use area have been completed.
  (Depending on the structure of the computer link system, it may be necessary to, before restarting data sending from the computer, notify the computer from the PC CPU of the state in which communications can be restarted.)

(3) Detecting a receive error (by the LEDs and buffer memory)

If the computer link module detects an error in the data received from the computer in bidirectional mode, the computer link module indicates it with the LEDs (2-C/N to 2-SIO or 4-C/N to 4-SIO) and stores the error information to buffer address 101H. Or, it stores corresponding error code to buffer address 117H.
Check the occurrence of a data receive error by the LEDs located on the front side of the module or by reading the above-mentioned buffer memory by the sequence program.

(a) Error indicating LEDs on the front side of the module

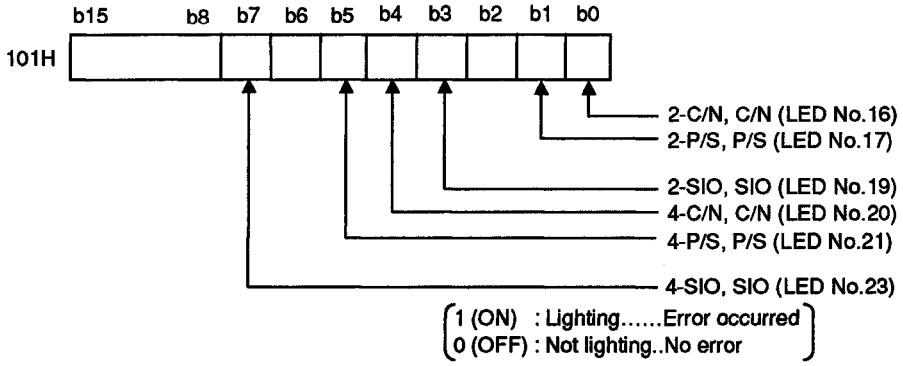


LED No.	LED Name		Meaning of LED Display	LED ON	LED OFF	LED Initial Status
16	2-C/N	C/N	Result of communications between PC CPU and RS-232C	Refer to Section 4.3 (4).	Normal	OFF
17	2-P/S	P/S	Parity error on RS-232C	Parity error	Normal	OFF
19	2-SIO	SIO	SIO error on RS-232C	Overrun, framing error	Normal	OFF
20	4-C/N	C/N	Result of communications between PC CPU and RS-422	Refer to Section 4.3 (4).	Normal	OFF
21	4-P/S	P/S	Parity error on RS-422	Parity error	Normal	OFF
23	4-SIO	SIO	SIO error on RS-422	Overrun, framing error	Normal	OFF

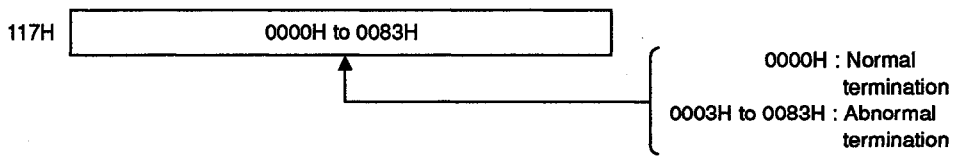
When an LED turns ON, it stays ON even when the cause of the error is eliminated.
To turn OFF an LED, follow the procedure described in Section 8.1.

(b) Buffer memory

- Buffer memory address 101H (See Section 8.1 for the details.)



- Buffer memory address 117H (Refer to Section 13.2 for the error codes and error contents.)

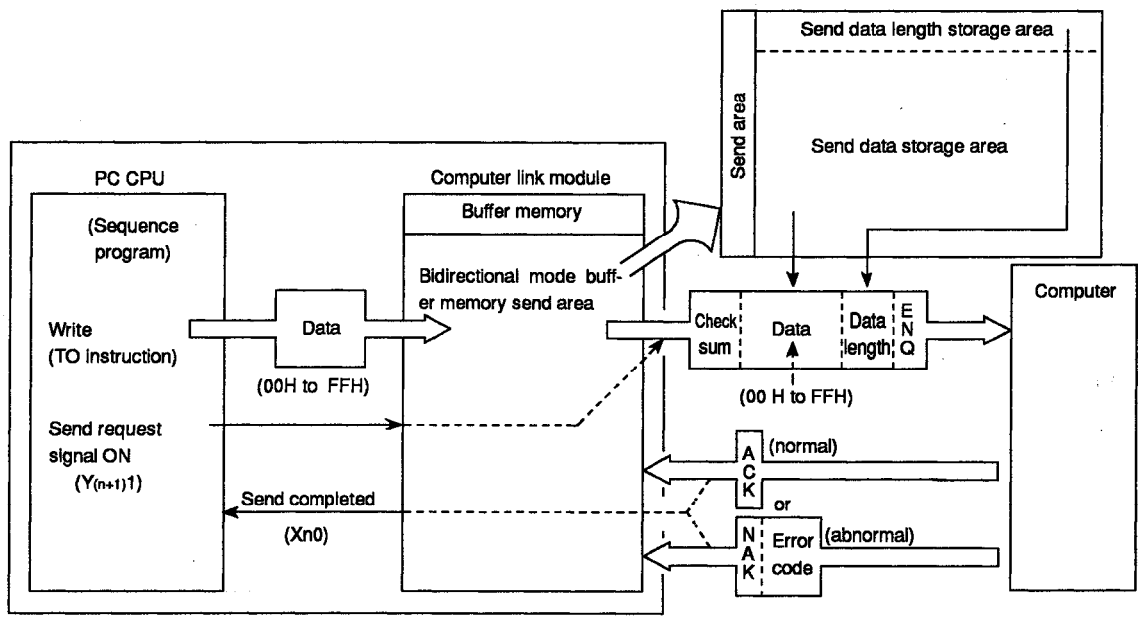


(4) Clearing the received data

Data transmission in bidirectional mode must be performed after receiving a response message to the previous data sending.
If the computer link module detects an error during data receiving, it sends a NAK message (response message) to the computer and ignores the data received when the error was detected.
Therefore, no processing is necessary to clear the received data.

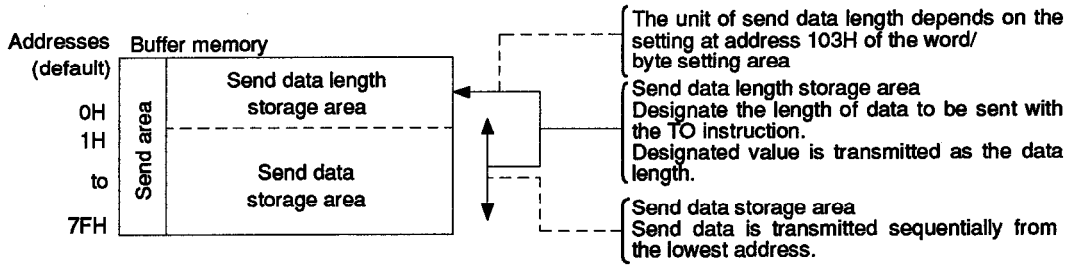
7.2.3 Sending data to the computer

This section gives the items required to be understood to send data to the computer.



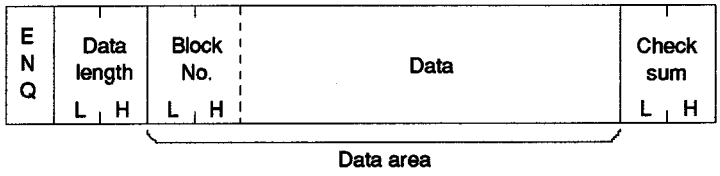
(1) Allocating the send area

Send area is a buffer memory area in the computer link module where the data and data length to be sent from the PC CPU to the computer are stored.
Default settings of the send area allocation are buffer addresses 0H to 7FH.
The send area allocation can be changed according to the purpose of data transmission, computer specifications, and send data length.
(Refer to Section 7.2 (2) (b) for the changing procedure.)



POINT

The length of data per one time of sending from the PC CPU to the computer must be smaller than the size of the send data storage area in the computer link module.
(Send data storage area) >= (Data length sent from PC CPU)
When it is necessary to send data whose length exceeds the size of the send data storage area, take either of the following methods.
(1) Changing the send area
Change the address and size of the send area.
(2) Dividing the send data
Divide the send data into parts so that each part is smaller than the send area size. Add block numbers to such divided parts of data for identification.
(Message format example)



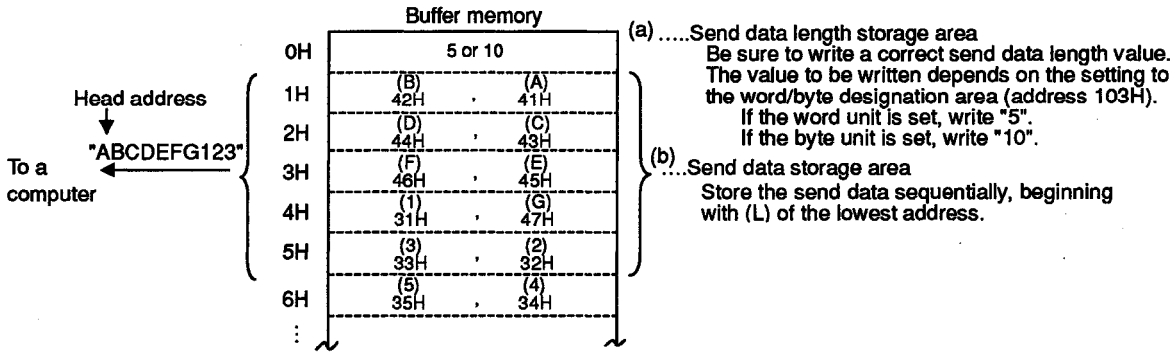
(2) Writing send data

The send data length and send data are written to the send area.

(a) The length of data to be written (having been written) to the bidirectional send data length storage area in either words or bytes.

(b) The data to be transmitted is written to the send data storage area.

(Example) To send "ABCDEFG123" (Send area addresses are defaults.)



After the operations (a) and (b), turn ON the send request signal ($Y_{(n+1)0}$). The computer link module starts sending the designated data of the designated data length beginning with the lowest address in the send data storage area.

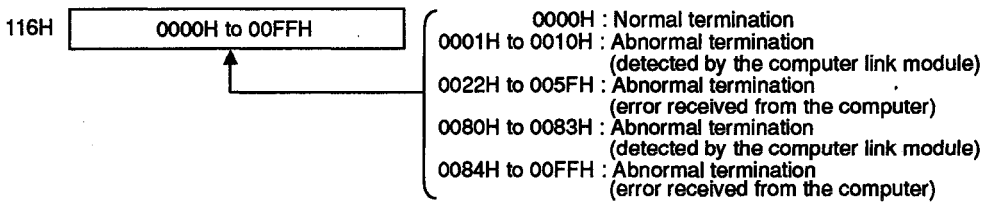
POINT

Set the data length in the sending and receiving message so that the same data length unit (word or byte) is used between the computer and the PC CPU.
The data length unit used with the PC CPU can be designated by setting it with the bidirectional word/byte designation area (address 103H) in buffer memory of the computer link module.

(3) Detecting a send error

If the computer link module detects an error in the data when sending to the computer received in bidirectional mode, the computer link module stores the error code to buffer address 116H.
Check the occurrence of a data send error by reading the above-mentioned buffer memory by the sequence program after the data send completed signal (Xn0) turned ON.

- Buffer memory (refer to Section 13.2 for the error codes and error contents)

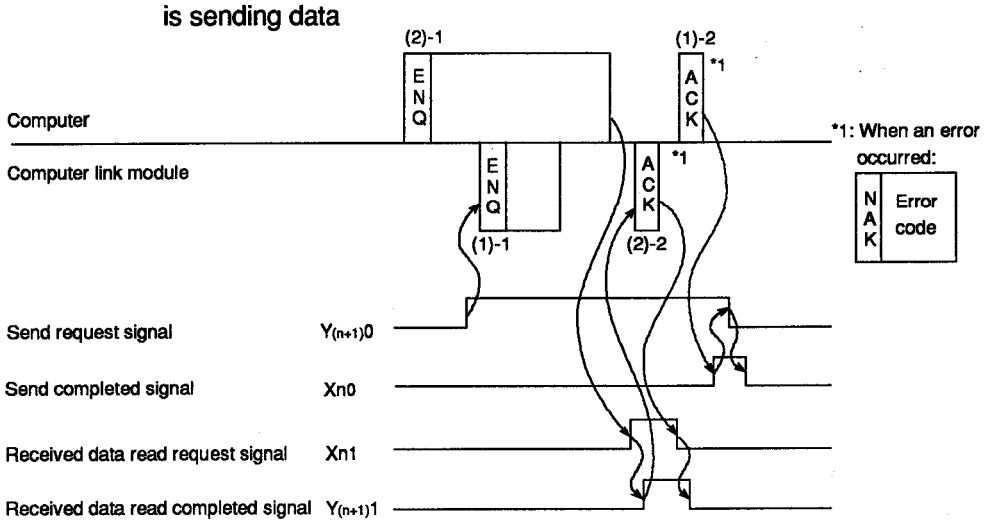


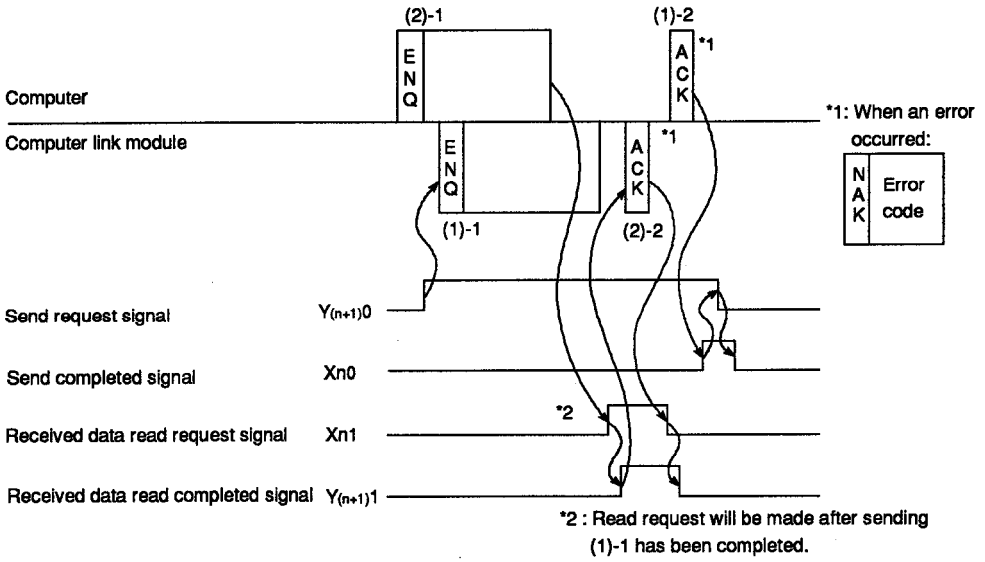
7.2.4 Processing a computer link module for simultaneous send in full-duplex mode

The following describes the processing in the computer link module when the computer and the computer link module simultaneously execute sending in bidirectional mode.
If the data communication by the half-duplex transmission is executed conforming to Section 10, sending from the computer and the computer link module will not be executed simultaneously. In that case, it is not necessary to read this section.

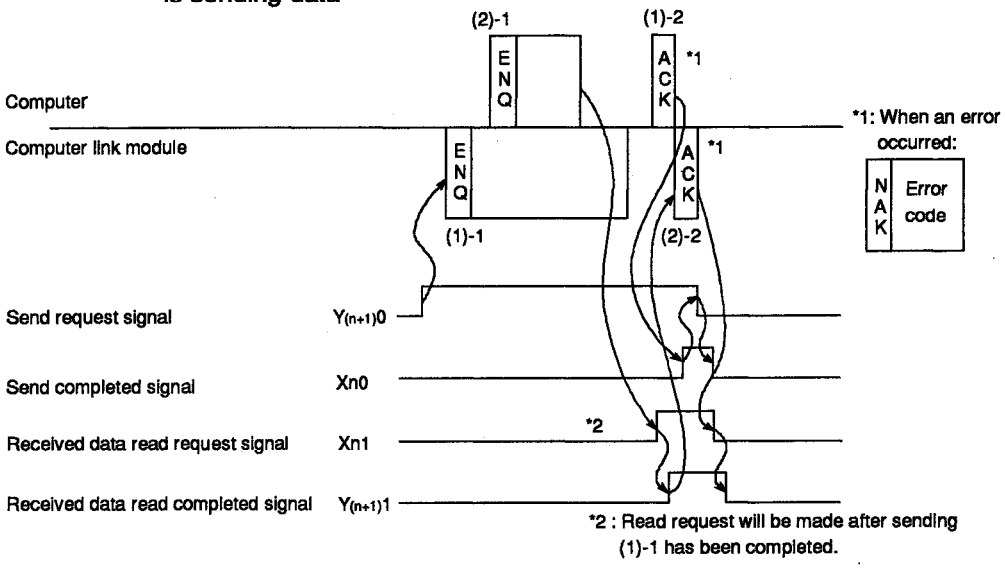
Processing by the computer link module varies depending on the setting (valid/invalid setting at simultaneous transmission) when the computer and the computer link module transmit data at the same time to each other.
The following shows the processing in the computer link module for respective settings.

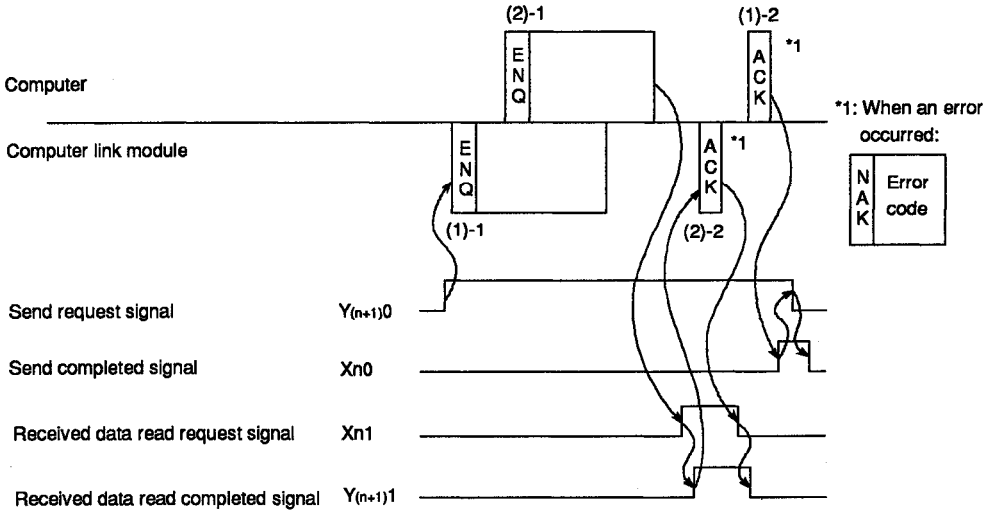
(Example 1) When the computer link module sends data while the computer is sending data





(Example 2) When the computer sends data while the computer link module is sending data





Buffer Memory Setting (Address 114H)	Setting	Processing Computer Link	
		Send Processing	Receive Processing
0000H	Send data : Valid Received data : Valid	After completing data send ((1)-1), the computer link module waits for response ((1)-2) while checking time-out error. Normal or abnormal send completion is confirmed by response and its status is transmitted to the sequence program via the buffer memory.	After completing data receive ((2)-1), the computer link module transmits the response ((2)-2). The received data and receive result are transmitted to the sequence program via the buffer memory.
0100H	Send data : Invalid Received data : Valid	After completing data send ((1)-1), the computer link module transmits the sequence program of a simultaneous transmission error (error code: 3) via the buffer memory. The comptuer link module does not wait for a response ((1)-2).	After completing data receive ((2)-1), the computer link module transmets the response ((2)-2). The received data and receive result are transmitted to the sequence program via the buffer memory.
0001H	Send data : Valid Received data : Invalid	After completing data send ((1)-1), the computer link module waits for response ((1)-2) while checking time-out error. Normal or abnormal send completion is confirmed by response and its status is transmitted to the sequence program via the buffer memory.	Data receive ((2)-1) is ignored and received data is discarded. The response ((2)-2) is not transmitted. Data receive is not transmitted to the sequence program.
0101H	Send data : Invalid Received data : Invalid	After completing data send ((1)-1), the computer link module transmits the sequence program of a simultaneous transmission error (error code: 3) via the buffer memory. The comptuer link module does not wait for a response ((1)-2).	Data receive ((2)-1) is ignored and received data is discarded. The response ((2)-2) is not transmitted. Data receive is not transmitted to the sequence program.

POINT

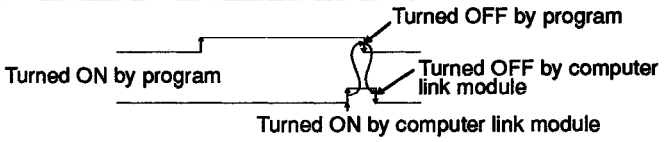
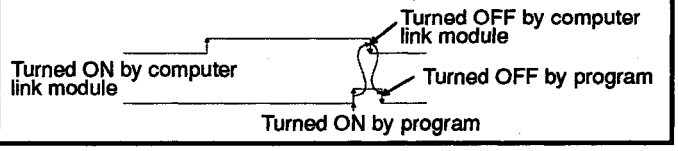
When the transmission control mentioned in Section 9 has been executed and when "send data valid" and "received data valid" are set with the simultaneous send data valid/invalid designation area (address 114H), the computer link module executes data send and receive processings as follows.

(1) Data send
- When it receives a send discontinue request (DC3 is received or DSR signal is OFF) from the computer during the data send, it will discontinue the data send.
- When it receives a receive enabled signal (DC1 is received or DSR signal is ON), it will restart sending.

(2) Data receive
- When sending a response message for the data receive to the computer and if it is impossible to send the response message due to a send discontinue request (DC3 is received or DSR signal is OFF) from the computer, it will send the response message after a send enabled state is established (DC1 is received or DSR signal is ON).

7.2.5 Handshake I/O signals

Signals known as I/O handshake signals are required for communications in the bidirectional mode.
These signals output data received from the sequence program to a computer or detect signals from an external device to enable the sequence program to read them.

	Signal		Timing
PC CPU ↓ Computer	$Y_{(n+1)}$ 0	(Send request)	
	X n 0	(Send completed)	
Computer ↓ PC CPU	X n 1	(Received data read request)	
	$Y_{(n+1)}$ 1	(Receive data read complete)	

The number "n" appended to X and Y is determined according to the position where the computer link module is loaded and the number of I/O modules loaded prior to this module.

REMARK

The I/O signals, other than those mentioned above, available in bidirectional mode are:
Xn7 (computer link module READY signal) and XnD (computer link module watchdog timer error signal).
Refer to Section 3.9 for the I/O signals used with the PC CPU.

7.2.6 Reading and writing buffer memory

A sequence program for reading/writing buffer memory is necessary to execute the operations using the computer link module in the bidirectional mode. (Refer to the buffer memory list in Section 3.10.)
Write a sequence program as required.
At a start-up of the computer link module, the default values are written to the specific use area in the buffer memory.

	Reading and Writing Operations that Need Sequence Programs	Reference Sections
(1)	To change the default value set to the buffer memory specific use area of the computer link module	Section 7.2.6 (2) and (3)
(2)	To receive data from the computer	Sections 7.2.2, 7.4, 7.5
(3)	To transmit data to the computer	Sections 7.2.3, 7.4, 7.6
(4)	To read the LED ON/OFF statuses of the computer link module or to turn OFF LEDs during computer linking	Section 8
(5)	To read the module/signal status during computer linking	Section 8
(6)	To execute the transmission control with the external device using the DC code	Section 9
(7)	To execute half-duplex communication on the RS-232C interface of the computer link module	Section 10

(1) Precautions on reading/writing to the buffer memory specific use area

(a) The buffer memory is not backed up by a battery

All data in buffer memory is reset to the default values when power is turned ON, when the PC CPU is reset, or when the mode is switched.
Data changed from the default values must be written whenever any of these events takes place.

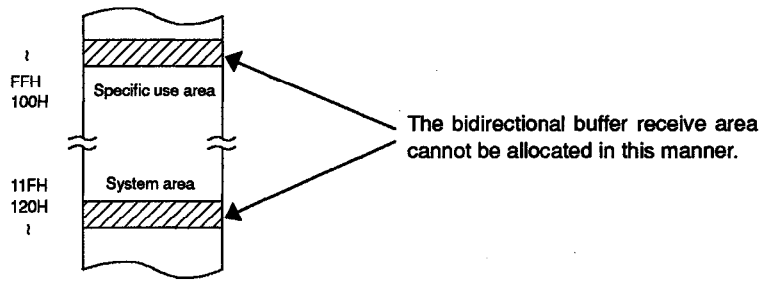
(b) Data can be written to the specific use area (100H to 11FH) excluding the mode switching area only by using a TO instruction in a sequence program.
If data is written to the buffer memory using the command in a computer program, the computer module will not operate correctly.

(c) If the following functions are used together, make sure to allocate the user area in buffer memory so that the same area will not be used by different functions.
If the same area is allocated to different functions, the data in this area is rewritten and communication will not be correctly executed.

- Bidirectional mode send
- Bidirectional mode receive
- Buffer memory read/write (CR/CW commands) functions
- On-demand functions

The memory areas preceding and following the specific use area cannot be allocated as a single area.
The areas 0H to FFH and 120H to DFFH must be recognized as independent areas.

Example:



(d) Do not write "1" (request to clear) to the no-protocol received data clear request area (address 10DH).

If "1" is written, all received data may be cleared, and data communications might not be correctly executed.

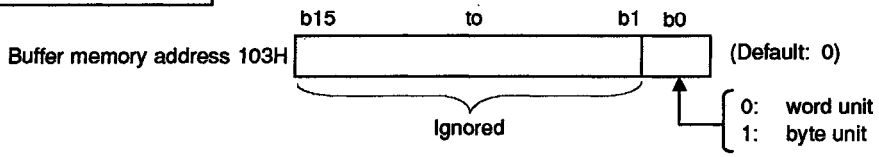
(2) Changing the default values in the buffer memory specific use area

Write an applicable sequence program to change a default value in the specific use area in the buffer memory as follows.

(a) Bidirectional word/byte designation area (address: 103H)

- When the computer link module is started up, the word unit is used for the send data length and received data length handled between the PC CPU and the computer link module and for the data size handled between the computer and the computer link module.
- To change the unit of the send data length, received data length, and data size to the byte unit, write "1" when the computer link module is started up.
- This unit is also used as the unit of send data length in the on-demand function in the dedicated protocol.

SETTING METHOD

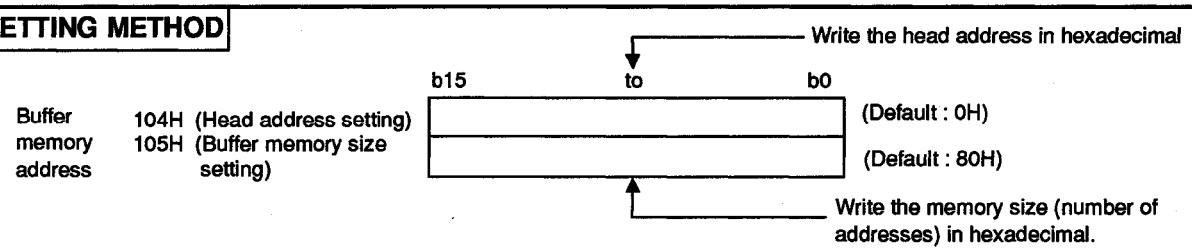


(b) The addresses for changing the buffer memory send area

The no-protocol send buffer memory area head address designation area (Address: 104H)
The no-protocol send buffer memory length designation area (Address: 105H)

- By default, buffer memory addresses of 0H to 7FH (size: 80H) are allocated to store send data from the PC CPU and also the length of the data.
- To change the default send data length and buffer memory area to store send data, write the head address of the new buffer memory area and the new buffer memory size when the computer link module is started up.
- Buffer memory areas 0H to FFH or 120H to DFFH can be allocated to the no-protocol mode send buffer memory (refer to (1)(c)).

SETTING METHOD



POINTS

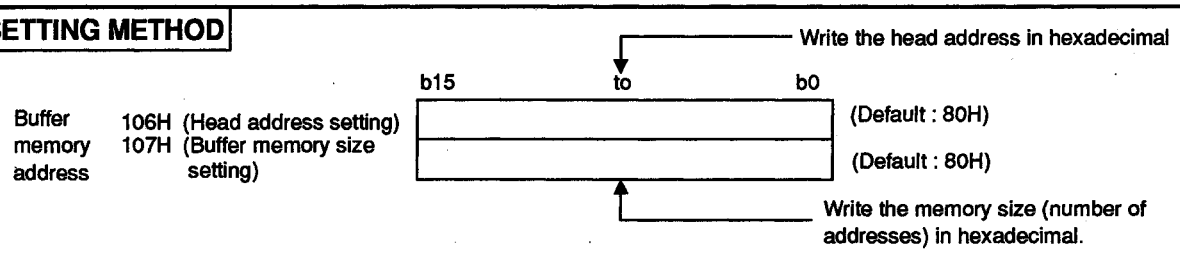
(1) Buffer memory addresses 100H to 11FH are for the specific use area and should set.
(2) Buffer memory address 105H should include the storage area of the send data length.
(3) If any range except the user area is set, the computer link module will execute operations with defaults.

(c) The addresses for changing the buffer memory receive area

The bidirectional receive buffer memory area head address designation area (Address: 106H)
The bidirectional receive buffer memory length designation area (Address: 107H)

- By default, buffer memory addresses of 80H to FFH (size: 80H) are allocated to store data that has been sent from an external device and is to be received by the PC CPU and the length of such data.
- To change the default receive data length and buffer memory area to store received data, write the head address of the new buffer memory area and the new buffer memory size when the computer link module is started up.
- Buffer memory areas 0H to FHH or 120H to DFFH can be allocated to the bidirectional mode receive buffer memory. (Refer to (1)(c).)

SETTING METHOD



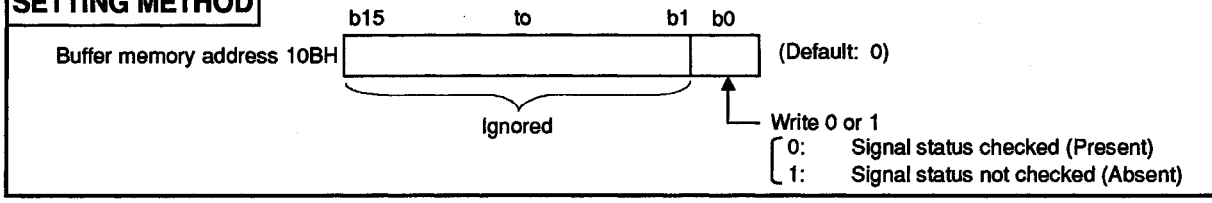
POINTS

(1) Buffer memory addresses 100H to 11FH are for the specific use area and should not be set.
(2) Buffer memory address 107H should include the storage area of the receive data length.
(3) If any range except the user area is set, the computer link module will execute operations with defaults.

(d) RS-232C CD terminal check designation area (Address: 10BH)

- When an external device is connected on the RS-232C of the computer link module, the setting described in Section 4.6.2 needs to be done.
- The default setting of the CD terminal check is ON.
- To turn OFF the CD terminal check, write "1" to buffer memory address 108H when the computer link module is started up.
- This setting is effective for any function in the dedicated protocol and bidirectional mode.

SETTING METHOD



REMARK

For the operation of the computer link module to the CD signal in an RS-232C CD terminal check, refer to Section 4.6.2.

(e) Areas for changing the communications mode of the RS-232C interface

RS-232C communications mode setting area (address: 10FH)
Simultaneous send priority/non-priority setting area (address: 110H)
Send method setting area when transmission is resumed (address: 111H)

- When the RS-232C interface of the computer link module is connected to an external device, the full-duplex transmissions with the external device is possible when the computer link module is started up.
- To perform half-duplex transmission, change the default values when the computer link module is started up as described in Section 10.

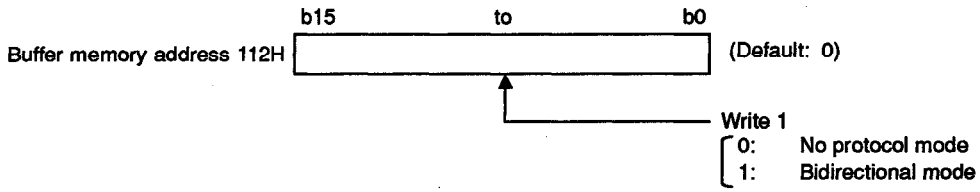
(f) Bidirectional mode designation area (address: 112H)

- The interfaces are set to the following modes according to the mode setting switch settings when the computer link module is started up:

Mode Setting Switch Settings	RS-232C Interface	RS-422 Interface	Remark
0	———		Setting impossible
1 to 4	Dedicated protocol	No-protocol mode	Bidirectional mode possible
5 to 8	No-protocol mode	Dedicated protocol	Bidirectional mode possible
9	No-protocol mode		Bidirectional mode impossible
A to D	Dedicated protocol		Bidirectional mode impossible
E	———		Setting impossible
F	———		For self-loopback test

- To set the no-protocol mode interface to bidirectional mode when the mode setting is 1 to 8, write "1" when the computer link module is started up.

SETTING METHOD



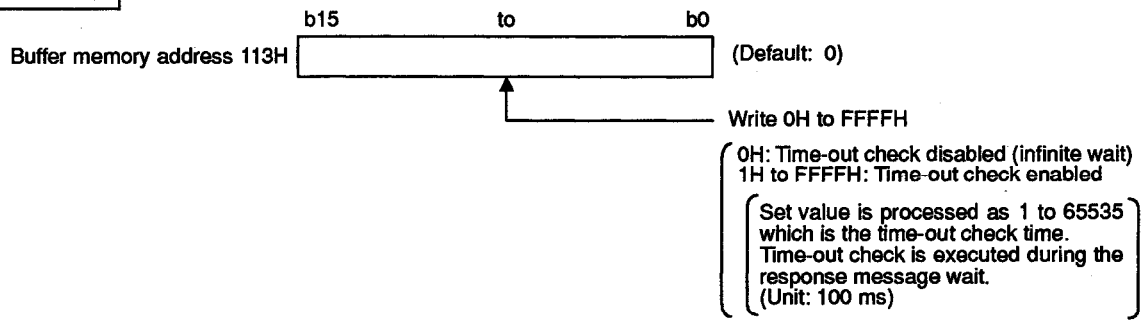
POINT

When the bidirectional mode is set, settings with buffer memory addresses 113H to 115H are valid.

(g) Time-out check time designation area (address: 113H)

- When data is sent from the PC CPU to the computer, the time-out check time setting for the reception of a response message is infinite when the computer link module is started up.
- To change the time-out check time setting, write the check time when the computer link module is started up.
- If a response message is not received within the designated time after data has been sent, the computer link module generates a time-out error. When the data send is completed, read buffer address 116H and check the result of data send.

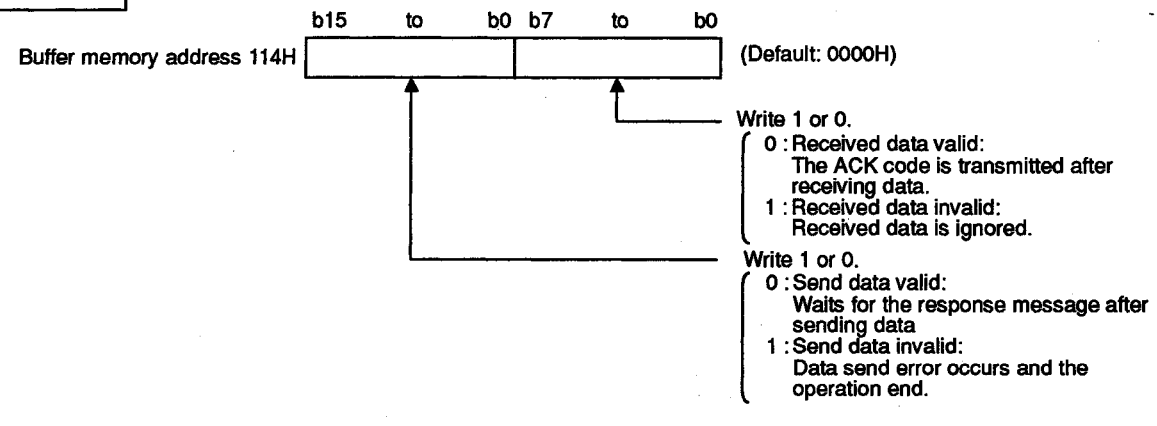
SETTING METHOD



(h) Simultaneous send data valid/invalid designation area (address: 114H)

- When the computer and the PC CPU simultaneously send data, both the data sent to the computer and the data received from the computer are valid when the computer link module is started up.
- To change the data valid/invalid setting in the computer link module at the simultaneous data send mentioned above, designate "send data invalid" or "received data invalid" when the computer link module is started up. (Refer to Section 7.2.4 for the operation contents)
- If "received data invalid" has been designated, the computer link module will ignore received data when it detects a simultaneous send.
- If "send data invalid" has been designated, the computer link module will generate a simultaneous send error when it detects a simultaneous send. When the data send is completed, read buffer address 116H and check the result of data send.

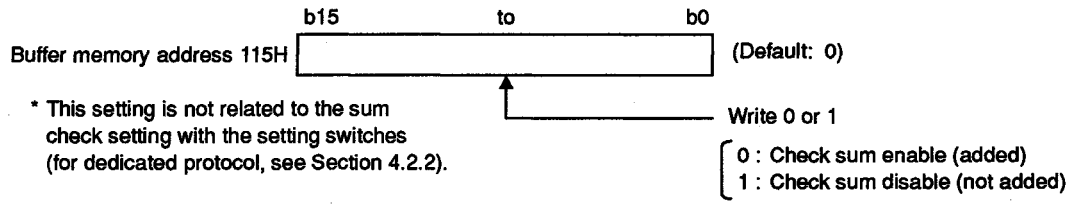
SETTING METHOD



(i) Check sum enable/disable designation area (address: 115H)

- "Check sum enabled" is set when the computer link module is started up.
- To set "check sum disabled", write "1" when the computer link module is started up.
- When "check sum disabled" has been set, the computer link module will not add check sum to the send message.
  And, messages received form the computer are regarded that check sum is not added.

SETTING METHOD



(j) Data send error storage area (address: 116H)

- When an error occurs during the data send to the computer due to a request by the PC CPU, the computer link module stores a corresponding error code to the send error storage area.
  (When a NAK message (response) to the data send is received, the received error code is stored to the send error storage area.)
- Before issuing a data send request, write "0" to buffer address 116H.
- When the data send is completed, read buffer address 116H and check the result of data send.
  The send error codes are listed in Section 13.2.

(k) Data receive error storage area (address: 117H)

- When an error occurs during the data receive from the computer, the computer link module stores a corresponding error code to the receive error storage area.
- When the data receive from the computer is faulty, read buffer address 117H and check the error content if necessary.
  The receive error codes are listed in Section 13.2.

(l) Area for changing the transmission control mode with the external device

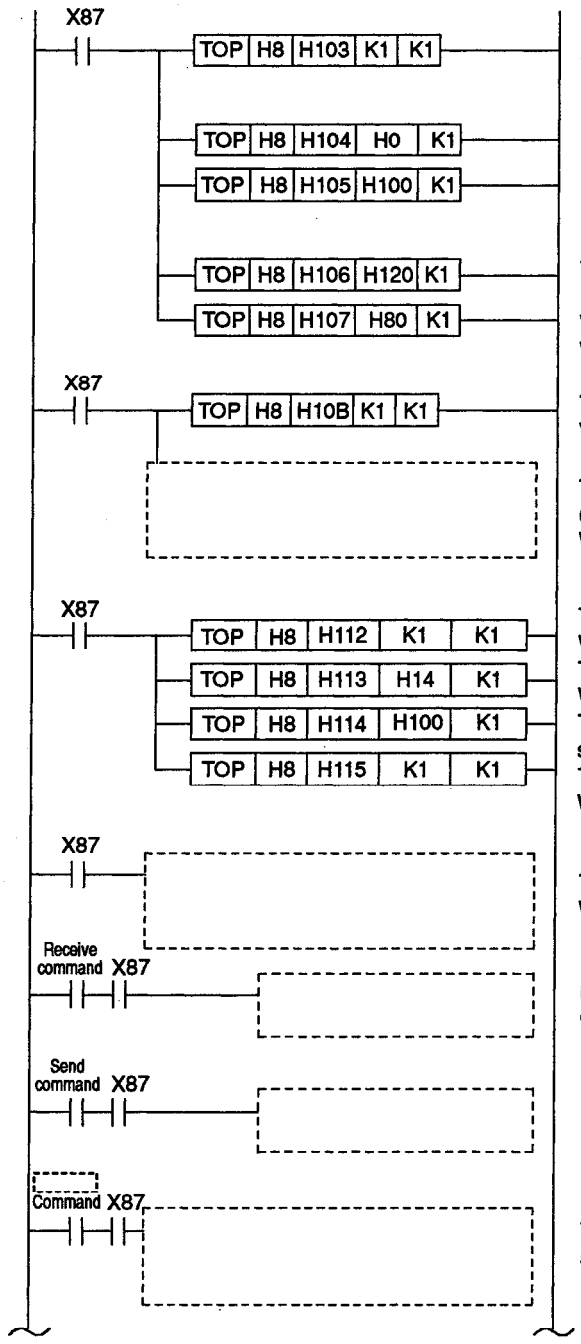
Transmission control designation area	(address: 11AH)
DC1/DC3 control code designation area	(address: 11BH)
DC2/DC4 control code designation area	(address: 11CH)

- When the computer link module is started up, the DTR/DSR control for the external device is executed only at the RS-232C interface.
- To execute the DC code control at the RS-232C and RS-422 interfaces, change the default settings when the computer link module is started up as described in Section 9.
- This setting is effective to all functions available with the dedicated protocol and bidirectional mode.

(3) Program examples change the default values in the buffer memory specific use area

The following are program examples change the default values in the buffer memory specific use area.
Incorporate necessary programs only.
In the following examples, the I/O signals of the computer link module are 80H to 9FH which are handled by the PC CPU.

SETTING METHODS (Computer link module I/O signals: 80H to 9FH)



To designate the byte unit for data handling
Write "1" (byte unit) to buffer address 103H.

To allocate the bidirectional send area to addresses 0H to FFH (*1)
Write "0H" to buffer address 104H.
Write "100H" to buffer address 105H.

To allocate the bidirectional receive area to addresses 120H to 19FH (*1)
Write "120H" to buffer address 106H.
Write "80H" buffer address 107H.

To set the CD terminal check "disabled"
Write "1" (check disabled) to buffer address 10BH.

To designate the communication mode of RS-232C (for half-duplex transmission)
Write the program according to Section 10.

To designate bidirectional mode
Write "1" to buffer address 112H.
To designate the time-out check time for 2 seconds (2000 ms)
Write "14H" to buffer address 113H.
To designate the data handling method (send data invalid) at simultaneous send Write "100H" to buffer address 114H.
To designate check sum "disabled"
Write "1" to buffer address 115H.

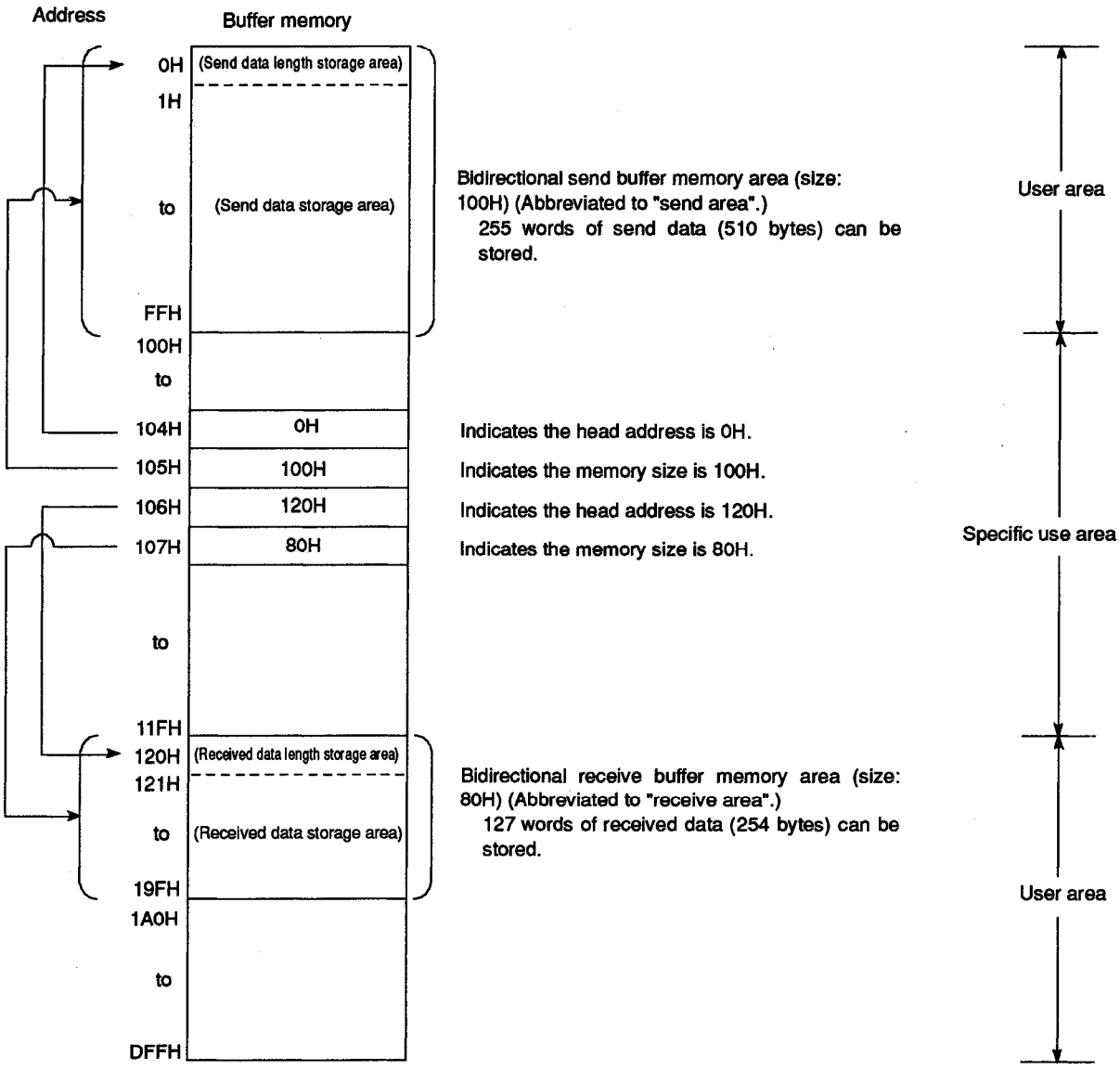
To designate the transmission control mode etc.
Write the program according to Section 9.

Data receive in bidirectional mode
Conform to Section 7.5.

Data send in bidirectional mode
Conform to Section 7.6.

To read the LED lighting status, write the LED off, and read the signal status and switch settings
Read/write according to Section 8.

*1 The following diagram shows the locations of the send and receive areas in bidirectional mode as designated in the example program given on the previous page.



REMARK

Refer to Section 3.10 for the user area and the specific use area.

7.2.7 Precautions during data communications

(1) Conditions when the computer link module transmission sequence is initialized:

- The power supply is turned ON, the PC CPU is reset with the reset switch, or the mode is switched.
- A response message (ACK, NAK) to the data send is received.
- The response message (ACK or NAK) is transmitted.
- When full-duplex transmission with the CD terminal check enabled is performed by the RS-232C (refer to Sections 4.6.2 and 7.2.6 (2) (d)), the CD signal is turned OFF.

(2) Send request signal made by the computer link module to PC CPU

To transmit data from a computer link module send area to a computer receive area, follow the steps described in Section 7.6.

Once the send request signal is turned OFF by turning ON the send completed signal, read the error code storage area (116H) for data transmission to check the send result before proceeding next transmission.

(3) When transmitting data from the computer or computer link module in bidirectional mode, follow the steps below.

- After completing the initial setting (No-protocol mode → Bidirectional mode, etc.) of the computer link module by turning on the PC, start the data communication.
- After completing the receiving/transmission of the message in response to the previous data transmission/receiving, start the next data communication.

(4) Send/receive data length and data area

Set the data length unit (word or byte) to be transmitted between the computer and the PC CPU so that the data is handled in the same unit.
For the PC CPU side, this setting can be made with the bidirectional word/byte designation area (address 103H) in the computer link module buffer memory.
The data area length in the message must be shorter than the size of the send data storage area and the received data storage area as mentioned in POINTS in Sections 7.2.2 (1) and 7.2.3 (1).

(5) NAK code

(a) Transmitting NAK from a computer link module to a computer

In full-duplex mode communication, a NAK is returned from the computer link module to the computer immediately on detection of a data length error, and after reception of the error message in the case of other errors. In half-duplex mode communication, the NAK is returned after the error detection message has been received.

[A computer link module ignores the designated length of received data if it detects an error while receiving data. if the data length is incorrect, the data received is ignored until ENQ code received.]

(b) Transmitting NAK from a computer to a computer link module

To transmit the NAK from a computer to a computer link module, transmit a 2-byte error code following the NAK code.
If the NAK code is received as the response, execute error processing according to the error code received directly after the NAK code.
The error codes related to the bidirectional mode communications are described in Section 13.2.
When the computer link module receives a NAK response when sending data to the computer, it reads the NAK after the data send is completed, writes the error code to buffer memory, and turns ON the send completed signal.

(6) Time-out check by a computer

If a time-out check is made for data transmitted from a computer send area to a computer link module receive area in the bidirectional mode, the time-out check time to be set must be longer than the value shown below.
(Maximum scan time of the PC CPU x 2) + 100 ms

(7) Data code

The computer link module in bidirectional mode does not convert data to the ASCII code. If it is necessary to handle data in either ASCII or JIS code, the data must be handled in ASCII or JIS code by the computer or the PC CPU.

(8) Framing error in the computer

A framing error happens to occur in the computer when there is no data being sent from the computer link module to the computer through the RS-422 channel.
In that case, the data received by the computer before ENQ, ACK, or NAK is sent from the computer link module must be ignored by the computer.
When using the RS-422 channel, make sure the interface specifications of the computer link module given in Section 3.8.3 before performing the data communications.

(9) Using a FROM/TO instruction to access the computer link module

A FROM/TO instruction to access the computer link module from a PC CPU should only be used when absolutely necessary.
This is because when data is transmitted from the computer link module to an external device at the same time as a FROM/TO instruction from a PC CPU is executed, the FROM/TO instruction has priority and is processed first.
This delays the data send time from the computer link module by how it takes to process the FROM/TO instruction.

(10) Data bit setting

If a check sum is comomunicated during data communications with a computer, set the data bit (set with a computer), set the data bit to eight bits.
Section 4.2.2 gives details about data bit setting.

7.3 Pre-Operation Check Points

Refer to Section 4 for the start-up procedure of the computer link module in bidirectional mode.
Recheck the following items before starting the operations to ensure normal performance of the computer link module.

	Confirm Item	Reference Sections
(1)	Switch settings	Section 4.2
(2)	Operation of the computer link module	Sections 4.5, 4.8.1
(3)	Connection with the external devices	Section 4.6
(4)	Connection/setting of a terminal resistor	Section 4.7

POINT

Unless these items are correctly set in the bidirectional mode, normal computer linking cannot be established.
Recheck the above items before starting operations.

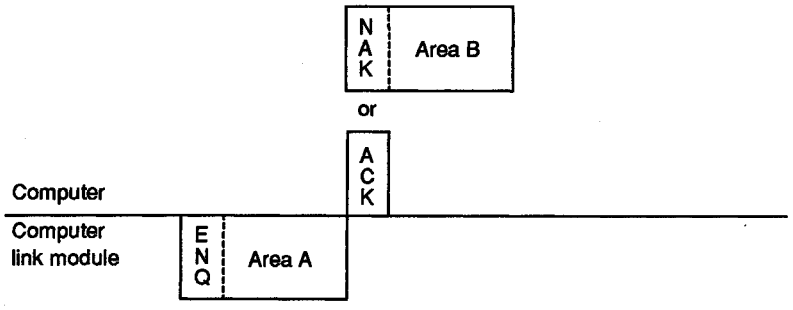
7.4 Bidirectional Control Procedure Basics

The following gives the control protocols and the contents of the items designated with the control protocols in bidirectional mode.

7.4.1 How to read the charts of bidirectional control protocols

This section explains how to read the charts of control protocols given in Section 7.4.2 and subsequent sections.

(1) Transmitting data from a computer link module to a computer



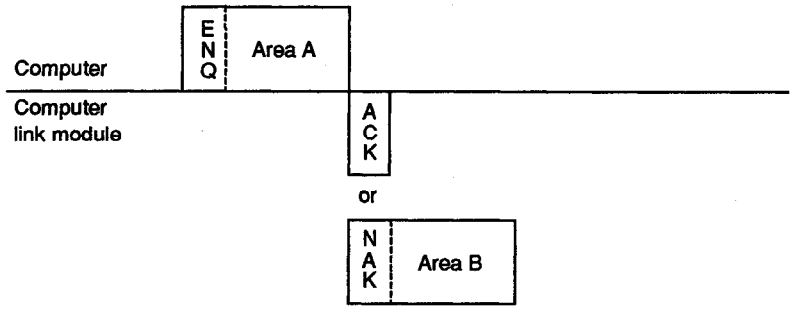
(a) Area A: Data send from a computer link module to a computer

(b) Area B: Data send from a computer to a computer link module

(c) Write a program so that data is transmitted from left to right.

(Example: For area A, data is transmitted ENQ to right)

(2) Transmitting data from a computer to a computer link module



(a) Area A: Data send from a computer to a computer link module

(b) Area B: Data send from a computer link module to a computer

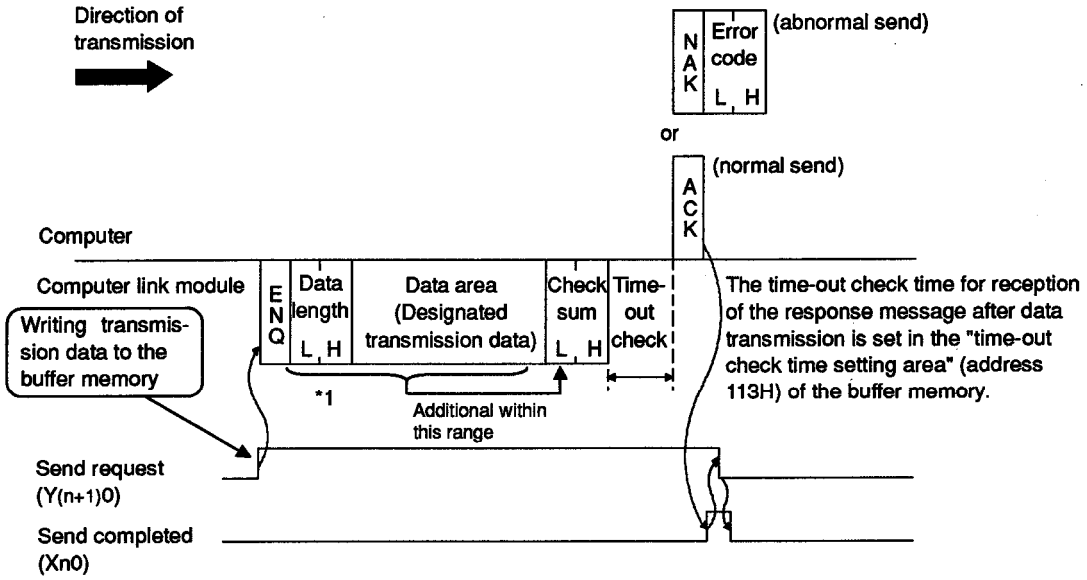
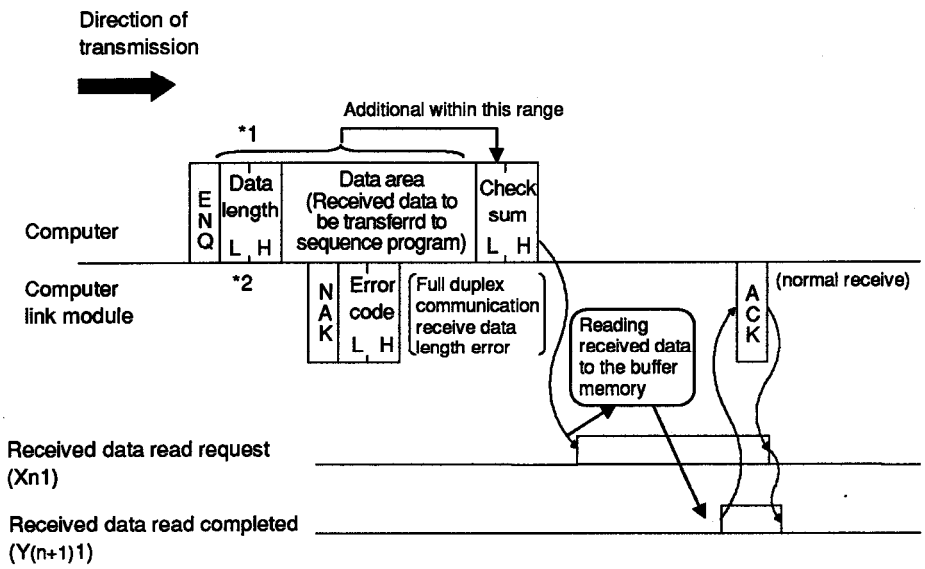
(c) Write a program so that data is transmitted from left to right.

(Example: For area A, data is transmitted ENQ to right)

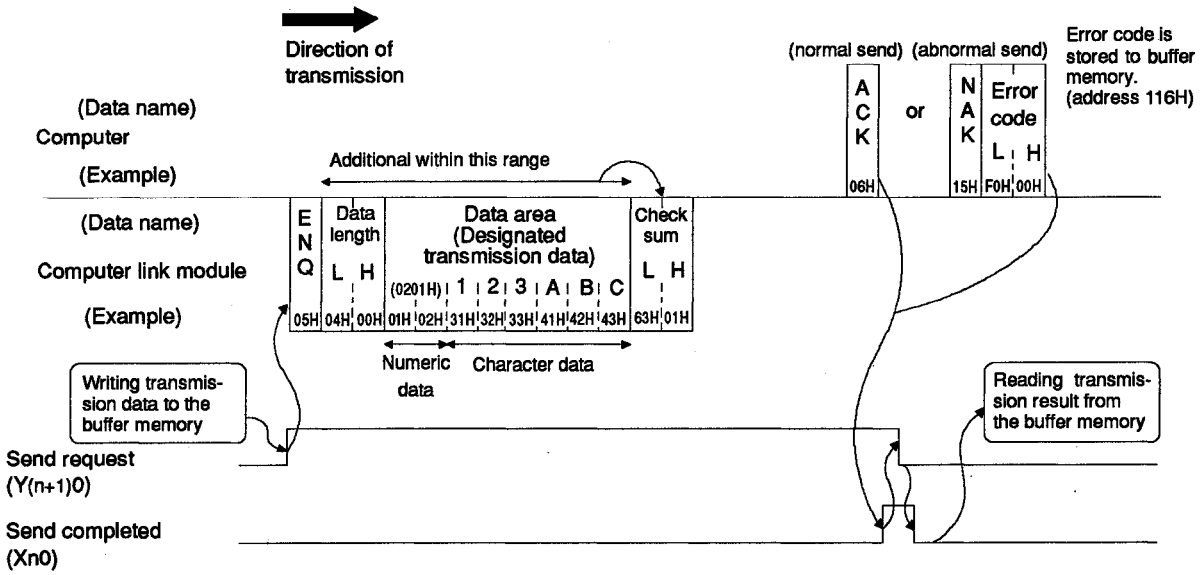
7.4.2 Control protocol in bidirectional mode

This section gives the basic format used for the data communications in bidirectional mode.

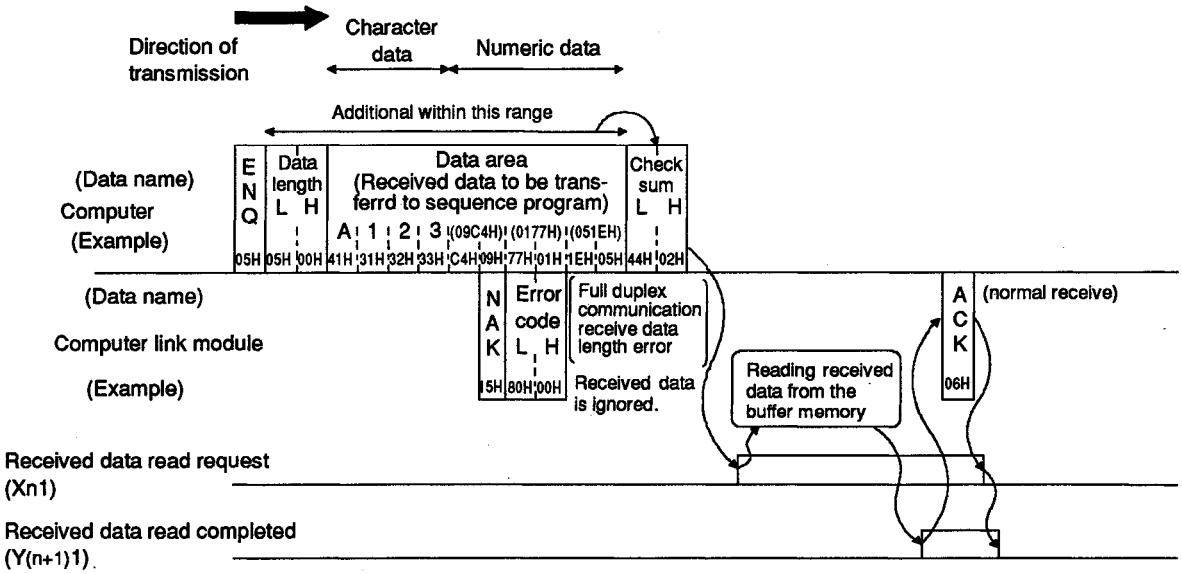
(Control protocol format)

<table>
<tr><th>Contents</th><th>Protocol</th></tr>
<tr><td>Transmission from a PC CPU to a computer</td><td> 
</td></tr>
<tr><td>the obove
:
A PC CPU receives data from a computer</td><td> 
</td></tr>
<tr><td>Remark</td><td>(1) Check sum exists only when "0" is set to address 115H of buffer memory.
(2) Check sum is calculated within the area indicated by *1.
(3) The computer link module returns a NAK at the following times on detection of an error during data reception.
1) In full-duplex mode communication
• On detection of a receive data length error : NAK is returned immediately on detection of the error ("*2" in the figure above).
• On detection of errors other than the above : NAK is returned after reception of the data
2) In half-duplex mode communication
NAK is returned after reception of the error detection message.
When the computer link module detects an error at data reception, the receive data is ignored (see Section 7.2.7(5)) so no receive data read request is issued at the PC CPU module side.</td></tr>
</table>

SETTING EXAMPLE Data length: Word unit



SETTING EXAMPLE Data length: Word unit



7.4.3 Contents of data designation items in bidirectional mode

This section gives the contents of data designated in the message sent and received between the computer and the computer link module.

(1) Control code

Signal Name	Code (Hexadecimal)	Meaning	Application
ENQ	05H	Enquiry	The code used to begin data send.
ACK	06H	Acknowledge	The code returned to the mating station when data has been received correctly.
NAK	15H	Nagative Acknowledge	The code return to the sending stations when data has not been receiving correctly. (immediately followed by an arror code)

(a) Data send from a computer link module to a computer

The computer link module appends the control code to be transmitted.

(b) Data send from a computer to a computer link module

The computer link module checks the control code received. It is not possible to read the control code from a sequence program.

(2) Data lengh

Data length expresses the number of bytes or words of data in the data area in 2-byte binary data. Data length units are determined according to the setting at address 103H of the buffer memory.

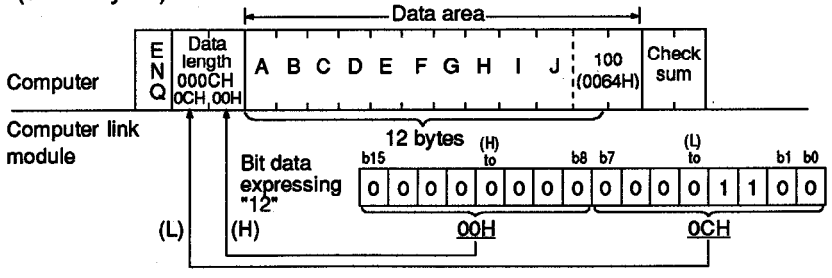
(a) Data send from a computer link module to a computer

The data length to be transmitted is the value written to the send data length storage area of the computer link module buffer memory by the TO instruction in a sequence program.
The computer link module transmits the written value as it is from the lower byte (L).

(b) Data send from a computer to a computer link module

The computer link module checks the received data length. When it is correct, the computer link module writes the first 1 byte to the lower byte position (L) of the received data length storage area of the computer link module buffer memory.

Example: Data length when transmitting "ABCDEFGHIJ" and value "100"
(units: bytes)



(3) Data area

The data of 00H to FFH code can be processed in a string of 1-byte data as the send data.

(a) Data send from a computer link module to a computer

The data area to be transmitted is the value written to the send data storage area of the computer link module buffer memory by the TO instruction in a sequence program.

The computer link module transmits the data according to the designated lenghth and byte/word units sequentially from the lower address in unchanged codes.

(b) Data send from a computer to a computer link module

The data area received is written to the received data storage area sequentially from the lower address in unchanged codes as they are received.

The data length to be written is determined by the data length in the received message (refer to (2)) and the designated word/byte units.

(4) Check sum

The check sum is the lower 2 bytes (16 bits) of the result obtained by adding the data length and the data area in the message as binary data.

If the setting at address 115H is "1", the check sum is not required.

(a) Data send from a computer link module to a computer

The computer link module calculates and adds the check sum.

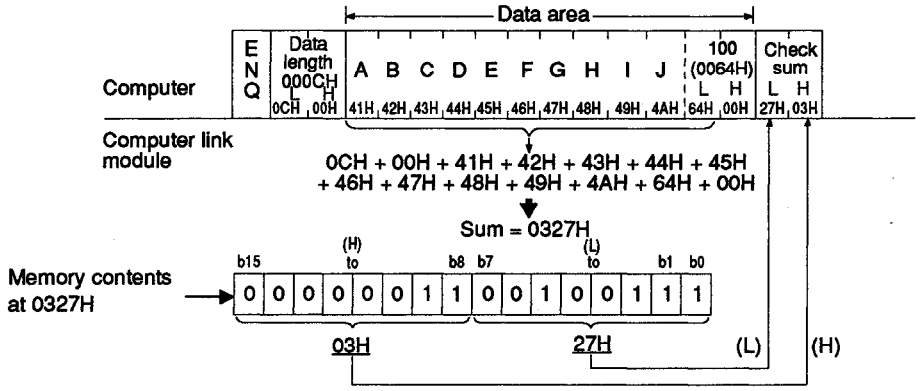
If the check sum is not processd, the check sum is not transmitted.

(b) Data send from a computer to a computer link module

The computer link module checks and processes the check sum received. It is not possible to read the check sum from a sequence program.

When the setting is "check sum is disabled", the received data following the data of the designated length is ignored up to the next control code (refer to (1)).

Example: Check sum when transmitting "ABCDEFGHIJ" and value "100" (units: bytes) is processed as follows.



(5) Error code

An error code indicates the error content when an NAK response is received. The code is transmitted and received in the range of 0001H to 00FFH. Section 13.2 gives error code detailes.

(a) Data send from a computer link module to a computer

The computer link module appends the error code.

When transmitting an error code the computer link module writes the same error code to its error code storage area in the received data buffer memory area.

(b) Data send from a computer to a computer link module

The computer link module writes the received error code to the error code storage area in its send data buffer memory area.

POINT

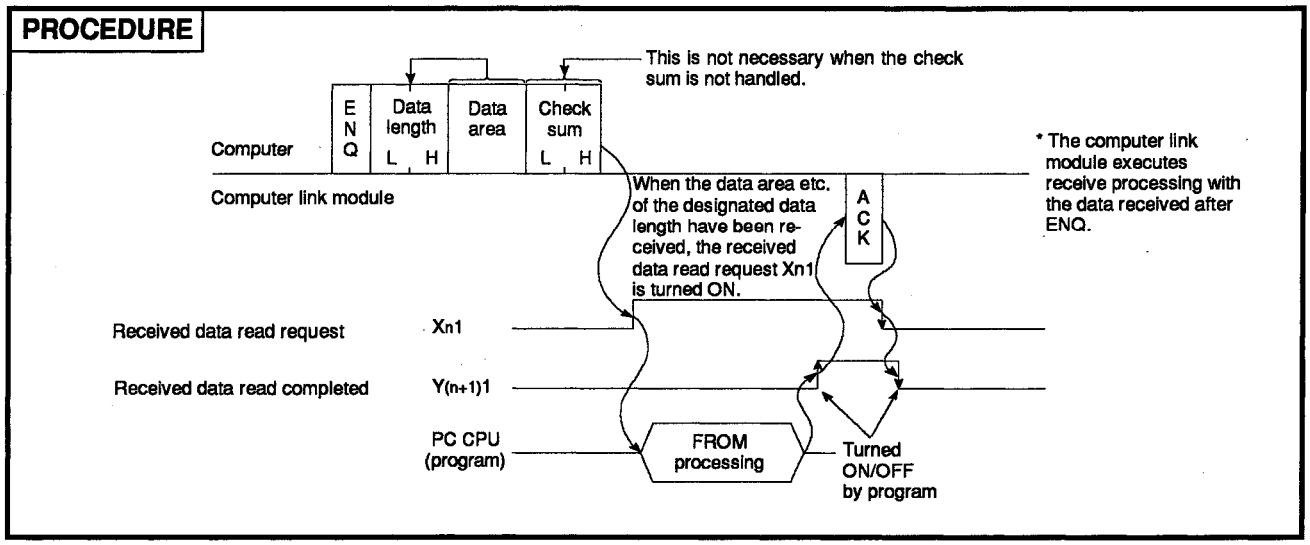
In bidirectional communications, check sum and error codes are all binary data. Note that in the dedicated protocol, they are handled in ASCII code.

7.5 Data Receiving Procedure in the Bidirectional Mode (Computer → Computer Link Module)

Receive processing stores the data received from the computer to the bidirectional receive buffer memory area (hereafter abbreviated to receive area) and reads the data by using a FROM instruction of the sequence program.

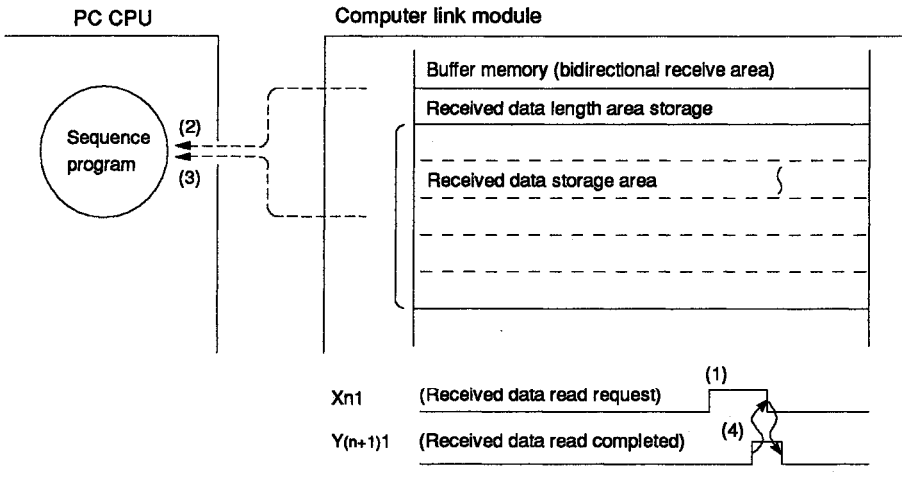
7.5.1 Data receive procedure

The following diagrams show the data receive process to read out data to the PC CPU in the bidirectional mode.



7.5.2 Data receive program

Shown below are basic sequence programs and programming examples when the PC CPU reads the data stored in the computer link module buffer memory (receive area) by using an application instruction.
The received data is read from the receive area (default: 80H to FFH).



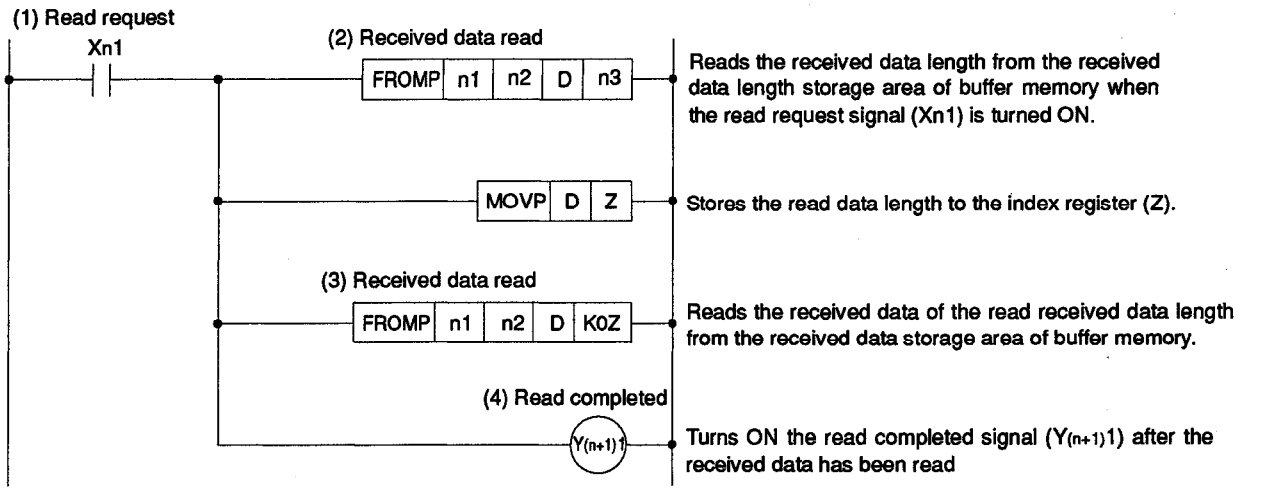
(1) to (4) in the diagram corresponds to (1) to (4) in (1)(a) and (2)(a) of this section.

(1) Basic sequence program (FROM, FROMP, DFRO, and DFROP instructions)

(a) Format to read from the receive area

Refer to the Programming Manual (Common Instructions) for the details.

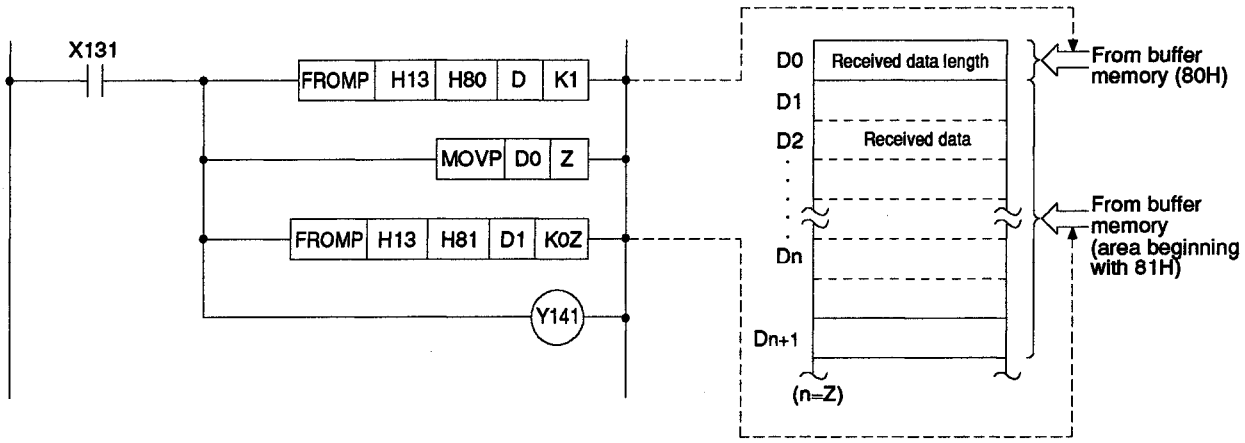
FORMAT



* Data read by program (3) is processed as the received data.

(b) Programming example

To read the data of n+1 words from buffer address 80H to D0 and after when the computer link module I/O signals are allocated to 130H to 14FH. (The designated unit of data is "word" in the example.)



REMARK

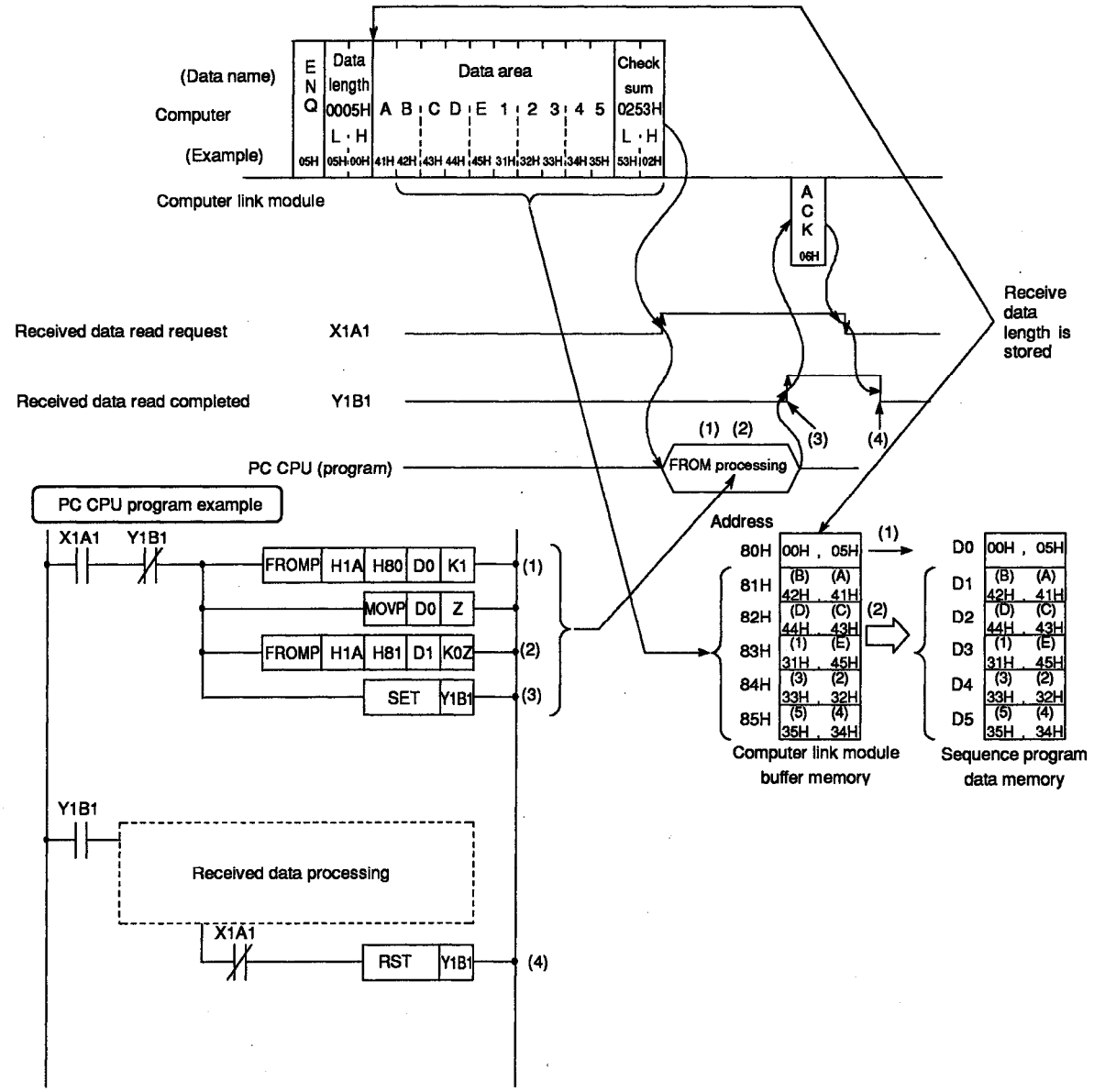
The AnA/AnUCPU dedicated instructions are not available for bidirectional mode.

(2) Application example

The following gives an application example to read received data and to store it to the data register.

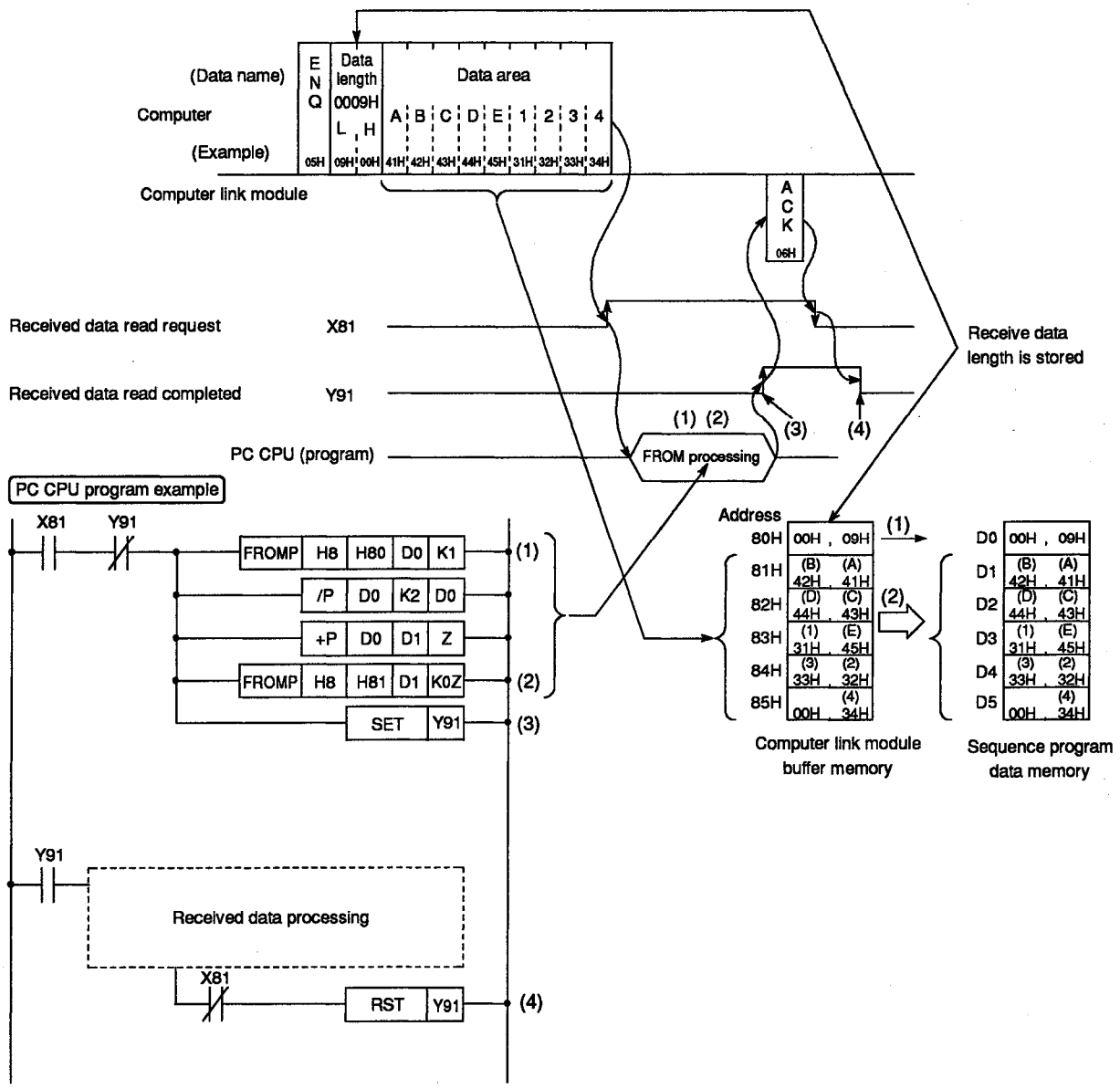
Example 1: Check sum enabled, in word units (receive memory area allocation: default)

To store "ABCDE12345", received from a computer, to D0 to D5 of the PC CPU. (Computer link module I/O signal: 1A0H to 1BFH)



Example 2: Check sum disabled, in word units (receive memory area allocation: default)

To store "ABCDE1234" , received from a computer, to D0 to D5 of the PC CPU. (Computer link module I/O signal: 80H to 9FH)



POINTS

- Even if send data units are set to byte units, the FROM instruction in a sequence program operates in word units. Therefore, the received data length must be converted to the number of buffer memory points (word units).
  In the above example, 9 bytes of data must be converted into 5 words (9 ÷ 2 = 4.5...5).
- When an odd number of bytes of data is received, the higher 8 bits of the last address read by the FROM instruction are "00H".