

MODEL
1866A
EPROM PROGRAMMER

INSTRUCTION MANUAL

MINATO

WARRANTY

MINATO ELECTRONICS Programmer is guaranteed against defect in materials and workmanship.

The warranty period for the Model 1866A is one (1) year.
The warranty period begins upon receipt of the programmer.

During one (1) year period, if the Model 1866A requires any repair due to defective materials or workmanship, contact the sales representative.

CONTENTS

1. GENERAL	3	6-15-3 Repeat	24	8-3-3 Example of connection with external equipment	44
2. SPECIFICATIONS	3	6-15-4 Power-on protection	24	8-4 Serial I/F Cable	44
3. CAUTION	4	6-15-5 Time-out function.....	24	8-4-1 Serial I/F Connector	44
4. EXTERNAL VIEW AND NAMES OF PARTS	4	7. REMOTE OPERATION	25	8-4-2 Table of serial I/F Connector pins	45
4-1 External View.....	4	7-1 Terminal equipment Connection	25	8-4-3 Cable	46
4-2 Display	5	7-2 Start of Remote Operation.....	25	8-5 Control Method and Transmission Protocol	47
4-3 Key Switches.....	5	7-3 Creating New Data.....	25	8-5-1 Control method	47
4-4 Rear Panel.....	6	- 7-3-1 How to Use Change Instruction	25	8-5-2 Transmission protocol	49
4-5 Bottom Switches	6	7-3-2 How to Use Insert Instruction	26	8-5-2-1 Non-control	49
5. OPERATION	7	7-4 Modifying Data.....	27	8-5-2-2 X-ON/X-OFF control	49
6. OPERATION MODES	7	7-5 Deleting Data	27	8-6 Instructions in Remote Mode	53
6-1 EDIT Mode	7	7-6 Inserting Data	27	8-6-1 Remote Instructions	53
6-1-1 PAE(Programming address entry mode).....	7	7-7 Transferring Data	28	8-6-2 Instructions description	53
6-1-2 CH(Change mode).....	9	7-8 Preparing List	28	8-6-2-1 Instructions for data transfer	53
6-1-3 INS(Insert mode).....	9	7-9 Punching Paper Tape	29	8-6-2-2 Instructions for P-ROM control	54
6-1-4 DEL>Delete mode).....	9	7-10 Reading Data from Paper Tape	29	8-6-2-3 Buffer operation instructions	54
6-1-5 INIT(INITIALIZE mode).....	10	7-11 Initializing Memory Contents	29	8-7 Instructions in Serial Mode	55
6-1-6 FORMAT	10	7-12 Programmer Control	29	8-7-1 Data format reception	56
6-1-7 COMP(COMPLEMENT mode)	10	7-12-1 Operation Address Range Specification	29	8-7-2 Data format output	57
6-1-8 SER(SEARCH mode)	10	7-12-2 Blank Check	30	8-7-3 List output	57
6-1-9 DEVICE	10	7-12-3 Program	30	8-7-4 Control code output	57
6-1-10 DEVICE Selection	11	7-12-4 Verify	30	8-7-5 Error	58
6-1-11 List of Device Codes	16	7-12-5 Copy	30	8-8 Data Format	58
6-1-12 List of Device Codes	17	7-12-6 Cont	30	8-8-1 Output format	59
6-2 SERI(SERIAL I/O mode)	20	7-12-7 Format Specification	30	8-8-2 Formats description	59
6-2-1 Entering Data	20	7-12-8 PTR Verify	30	1. TEKTRONIX HEX. (16 bits)	59
6-2-2 Outputting	20	7-12-9 Buffer's check sum	30	2. TEKTRONIX HEX. (8 bits)	61
6-2-3 Issuing List	20	7-12-10 PROM device select	30	3. INTEL HEX	61
6-2-4 Outputting Control Code	20	7-12-11 Address select	30	3-1 INTEL HEX. (8/16 bits)	61
6-2-5 Handshake	20	7-12-12 EEPROM's erase	30	3-2 INTEL HEX.	62
6-3 COPY	20	7-12-13 EEPROM's byte erase	31	4. MOTOROLA	63
6-4 BLANK	21	7-12-14 Remote Operation Release Command	31	4-1 Motorola Exerciser	63
6-5 PROG(PROGRAM) Mode	21	8. SERIAL INTERFACE	32	4-2 Motorola Exermax (16 bits)	63
6-6 VER(VERIFY) Mode	21	8-1 Introduction	32	5. TEXAS SDSMAC	64
6-7 CONT(CONTINUOUS) Mode	21	8-1-1 Serial I/F ratings	32	6. MINATO HEX	64
6-8 REM(REMOTE) Mode	21	8-1-2 Setting the serial I/F	33	7. ASCII-HEX	65
6-9 Address Mode Selection	21	8-1-3 Serial I/F selection	34	7-1 ASCII-HEX (SPACE), V1	65
6-10 AUTO/STEP Selection	22	8-2 RS-232C Level Interface	34	7-2 ASCII-HEX (SPACE), V2	65
6-11 Check Sum	22	8-2-1 Descriptions of interface signals	34	7-3 ASCII-HEX (SPACE), V3	65
6-12 Error Messages	22	8-2-2 Interface signal output	35	7-4 ASCII-HEX (SPACE), V4	65
6-12-1 Table of error codes	22	8-2-3 Connection to external equipment	36	8. BINARY	65
6-13 Block Operations in Copy, Program, Verity and Cont mode	23	8-2-3-1 Connection with data terminal equipment	37	9. HP-64000 ABSOLUTE	66
6-14 Setting Serial I/O Mode	23	8-2-3-2 Connection with data communication equipment	40	Contained in the pocket at back of the rear cover.	
6-15 Others	23	8-3 Current Loop Interface	42		
6-15-1 Key Lock	24	8-3-1 Current Loop Interface	42		
6-15-2 Test	24	8-3-2 Current Loop interface signal output	43		

1

GENERAL

The Model 1866A EPROM Programmer is a PROM programmer which incorporates 8 bit micro-processors and current state-of-the-art LSIs at its control section and avails itself of their functions fully.

The device to be programmed includes EPROMs/EEPROMs from 16 k to 512 k-bits.

The 1866A Programmer is provided with an interface RS232C

for outside service. If it is connected to a computer, programmed data may be transferred with ease from memories, disks, etc. of the computer thereinto.

It is provided with a 64 k-byte programming buffer memory, permitting data up to 512 k-bits to be processed.

The 1866A, owing to its small size and light weight, may be transported with ease, and may be used in wide applications.

2

SPECIFICATIONS

○ CPU	Z-80A (4.0MHZ)
○ Buffer RAM	64 k-bytes (512 k-bits)
○ Monitor RAM	2 k-bytes
○ Monitor ROM	8 k-bytes
○ Operation switch	Hexadecimal key switch
○ Display	LED
○ Interface	Serial: RS-232C, 20mA Current loop, switch selectable
○ Baud rate	110, 300, 600, 1200, 2400, 4800, 9600 switch selectable
○ Tape format	11 kinds of formats, selectable from keyboard
○ Tape parity	Odd, even, no parity switch selectable
○ Check function	Memory test, Power test, Reverse inversion test, programmer self test, Others
○ Temperature	+5 to +35°C
○ Power supply	AC 90V to 130V or 180V to 270V (±10%), 50/60Hz
○ Power consumption	30VA
○ Size	280(W) x 208(D) x 70(H) (mm)
○ Weight	1.5kg

3

CAUTION

When operating the Model 1866A, the following items must be observed:

○ Power supply

- (1) The power supply to be used must be capable of generating 90 to 110 percents (50 or 60Hz) of the rated voltage of your country.
- (2) The 1866A must be separated from equipment which is likely to produce noises (copier, cooler or eraser, etc.).
- (3) Do not turn on and off power with any PROM inserted in its socket; otherwise, the PROM may be damaged.

○ Storage and environmental conditions

- (1) The 1866A is provided with vents at the rear to prevent

any temperature rise; do not cover any vents and do not put the 1866A in any place with few ventilations.

- (2) Do not expose the 1866A to direct sunshine and heat.
- (3) Do not put the 1866A in any extremely wet or dusty place.
- (4) The 1866A is composed of electronic precision parts. Do not subject them to any shock.
- (5) Do not use the 1866A with any foreign materials (water, liquids, metals, etc.) contained; otherwise they may be damaged.

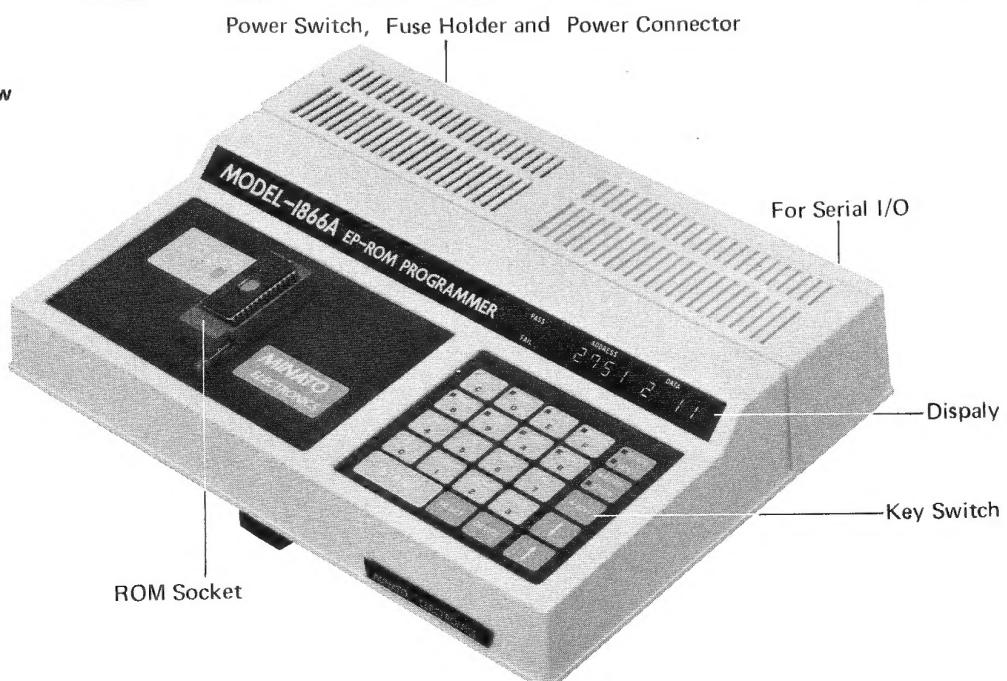
○ Troubles

If any troubles (e.g., abnormal odor or over-heat) are found, disconnect the power cord plug and contact the representative or MINATO ELECTRONICS INC.

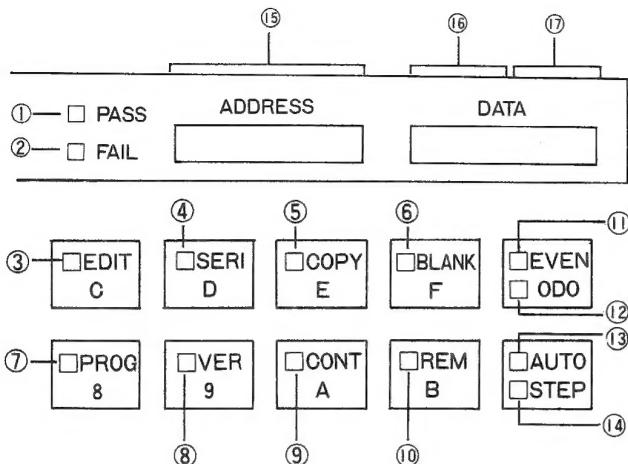
4

EXTERNAL VIEW AND NAMES OF PARTS

4-1 External View



4-2 Display



Indicated by the number ① is the Pass indicator. The indicator (PASS) lights if, in the set, all operation including programming terminates in a desirable manner.

Indicated by ② is the Fail indicator. The indicator (FAIL) lights if, in the set, an error occurs in operation including programming.

Indicated by ③ is the LED which indicates that the set is in the EDIT operation mode.

Indicated by ④ is the LED which indicates that the set is in the SERIAL I/O operation mode.

Indicated by ⑤ is the LED which indicates that the set is in the COPY mode.

Indicated by ⑥ is the LED which indicates that the set is in the BLANK check mode.

Indicated by ⑦ is the LED which indicates that the set is in the PROGRAM mode.

Indicated by ⑧ is the LED which indicates that the set is in the VERIFY mode.

Indicated by ⑨ is the LED which indicates that the set is in the **SCENE** (scene) mode.

Indicated by ⑩ is the LED which indicates that the set is

Indicated by ⑪ is the LED which indicates that the set is

Indicated by ⑫ is the LED which indicates that the set is

Indicated by ⑬ is the LED which indicates that the set is

in the AUTO mode. With the lamp lighting, the set continuously operates on all addresses.

Indicated by (14) is the LED which indicates that the set is in the STEP mode. With the lamp lighting, the set stops whenever execution is made on an address.

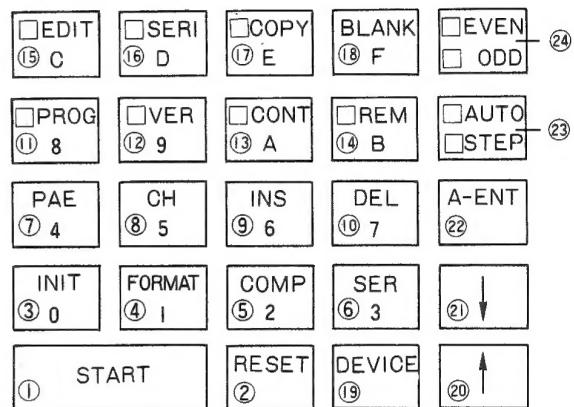
Indicated by ⑯ are LEDs which indicate an address being

operated in the EDIT mode and an address to be given to the P-ROM in the other modes. These LEDs are also used to indicate various kinds of messages.

Indicated by ⑯ are LEDs which indicate the contents of the buffer memory for data to be written.

Indicated by ⑯ are LEDs which indicates data read from the P-ROM.

4-3 Key Switches



The number ① signifies the START switch. Pressing the switch in COPY, BLANK, PROGRAM, VERIFY and CONTINUOUS modes causes operation concerned to be started.

The number ② signifies the RESET switch. Pressing the switch provides resetting and causes operation to be interrupted.

- Out of the switches described below, switches ⑧ through ⑩, ⑯, ⑰, ⑲, and ⑳ are valid only in the EDIT mode.

The number ⑧ signifies the memory INITIALIZE switch, which is also used to enter hexadecimal 0.

The number ④ signifies the switch used to enter hexadecimal 1, which is also used to data format.

The number ⑤ signifies the switch used to enter hexadecimal 2, which is also used to invert buffer memory data.

The number ⑥ signifies the switch used to enter hexadecinal 12, which is used to read data.

The number ⑦ signifies the switch used to specify an operation address range in each operation mode. The switch is shown in Figure 1.

The number ⑧ signifies the switch used to specify the CHANGE mode. The switch is also used to enter hexadecimal 4.

The number ⑧ signifies the switch used to specify the decimal 5.

INSERT mode. The switch is also used to enter hexadecimal 6.

The number ⑩ signifies the switch used to specify the DELETE mode. It is also used to enter hexadecimal 7.

The number ⑪ signifies the switch used to specify the PROGRAM mode. It is also used to enter hexadecimal 8.

The number ⑫ signifies the switch used to specify the VERIFY mode. It is also used to enter hexadecimal 9.

The number ⑬ signifies the switch used to specify the CONTINUOUS mode. It is also used to enter hexadecimal A.

The number ⑭ signifies the switch used to specify the REMOTE mode. It is also used to enter hexadecimal B.

The number ⑮ signifies the switch used to specify the EDIT mode. It is also used to enter hexadecimal C.

The number ⑯ signifies the switch used to specify the SERIAL I/O mode. It is also used to enter hexadecimal D.

The number ⑰ signifies the switch used to specify the COPY mode. It is also used to enter hexadecimal E.

The number ⑱ signifies the switch used to specify the BLANK check mode. It is also used to enter hexadecimal F.

The number ⑲ signifies the switch used to specify the DEVICE select mode.

The number ⑳ signifies the switch which increments an address.

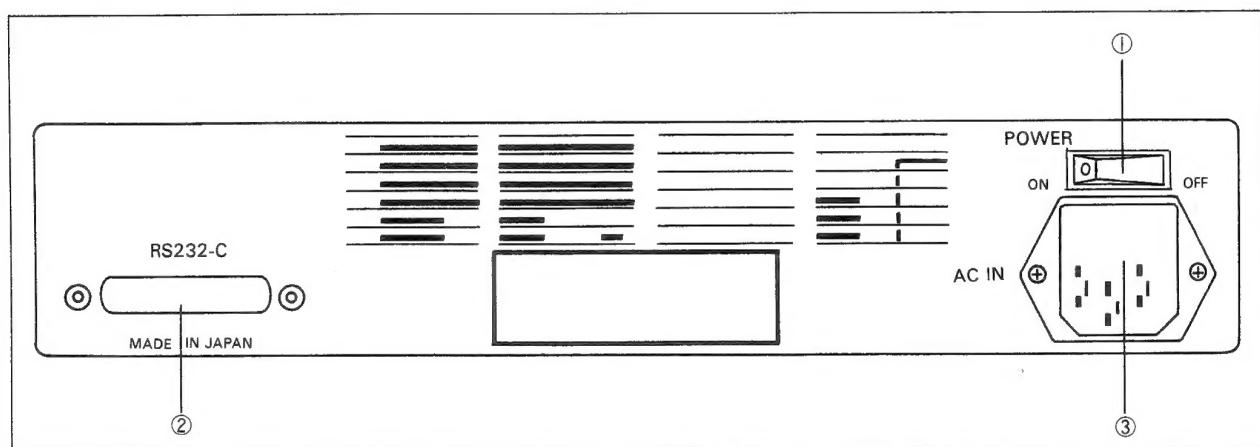
The number ㉑ signifies the switch which decrements an address.

The number ㉒ signifies the switch which directs an address to be entered.

The number ㉓ signifies the switch used to change over AUTO and STEP.

The number ㉔ signifies the switch used to change over EVEN and ODD address modes.

4-4 Rear Panel



The number ① signifies the POWER switch.

The number ② signifies the receptacle for serial I/O. The connector is used to RS-232C or 20mA current loop.

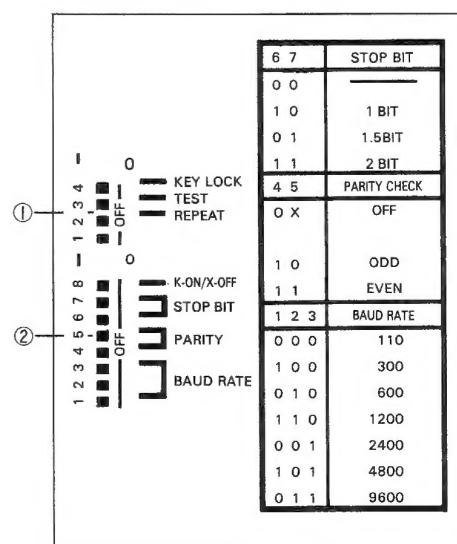
The number ③ signifies the receptacle for AC input.

The number ④ signifies the fuse holder which contains a fuse and a spare fuse.

4-5 Bottom Switches

The switch ① selects "KEY LOCK" "TEST" or "REPEAT" operation.

The switch ② selects I/O's baud rate, parity or protocol.



5

OPERATION

Turning on power

1. Check the power switch of the set to see if it is placed in the OFF position.
2. After checking that the receptacle is for AC volt, connect power.
3. Place the power switch in the ON position. If set operation is normal, turning on power causes the address to display the name of a programmable device. (Initially, 2716 is selected.) And the subsequent operation is waited for.

6

OPERATION MODES

The 1866A is provided with eight basic operation modes.

- EDIT mode

This mode edits data to be written and specifies an address to be programmed.

- SERIAL (SERI) mode

This is the SERIAL I/O mode which inputs/outputs data to be written using the serial I/O port.

- COPY mode

This mode enters data to be written from the master ROM.

- BLANK mode

This mode blank checks PROMs.

- PROGRAM (PROG) mode

This mode programs PROMs.

- VERIFY (VER) mode

This mode verifies programmed PROMs.

- CONTINUOUS (CONT) mode

This mode provides BLANK, PROGRAM and VERIFY operations successively.

- REMOTE (REM) mode

This mode remotely controls the Programmer from an external CPU or terminal using the SERIAL I/O port.

6-1 EDIT Mode

This mode has nine instructions.

6-1-1 PAE (Programming address entry mode)

This instruction specifies an operating range during data writing or reading. This instruction is capable of specifying a ROM's start address and stop address, and a RAM's start address.

In the Programmer, the operating address range is automatically determined by power on, but may be changed at need.

- Changeable contents

- PROM start address and PROM stop address

Specify the portion of a PROM to be programmed.

At power on, the start address and stop address are set at 0000_H and the end address of a PROM to be programmed.

- Memory (RAM) start address

When programming a PROM, specify which portion of the programme data is to be written from. This also applies to reading (Copy and Verify).

- Notes for change

The start address is equal to or less than the stop address. The stop address is equal to or less than

PROM's end address. Unless the above requirements are met, "E-4" is displayed, when "START" is pressed. Presented below is the procedure.

- (1) Press "EDIT".
- (2) Press "PAE". (PAE refers to Program Address Entry.)
- (3) Enter a programming start address (PROM start address) in a hexadecimal number consisting of four digits. If no change is required, press " \uparrow " to proceed to the step (5).
- (4) Press "A-ENT".
- (5) Enter a programming stop address (PROM stop address) in a hexadecimal number consisting of four digits. If no change is required, press " \uparrow " to proceed to the step (7).
- (6) Press "A-ENT".
- (7) Enter a memory's (RAM's) start address in a hexadecimal number consisting of four digits. If no change is required, press " \uparrow ". "END" is displayed and all operation terminates.
- (8) If "A-ENT" is pressed, "END" is displayed and all operation terminates.

Examples

1. Write the contents of the address 0600_H of the memory (RAM) into the portion of 2732 from the address 0400_H to the address $7FF_H$.

- (1) Set 2732 by the DEVICE select switch.
- (2) Press "EDIT".
- (3) Press "PAE". The ROM start address is displayed.



- (4) Enter the PROM start address (0400_H) in a hexadecimal number consisting of four digits.



- (5) Press "A-ENT". The ROM start address is set and the ROM stop address is displayed on the ADDRESS portion.



- (6) Enter the PROM stop address ($07FF_H$) in a hexadecimal number consisting of four digits.



- (7) Press "A-ENT". The ROM stop address is set and the RAM start address is displayed on the ADDRESS portion.



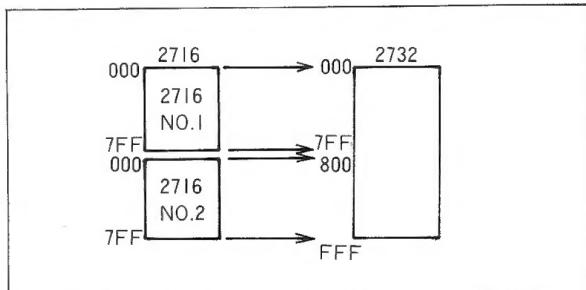
- (8) Enter the memory (RAM) start address (0600_H) in a hexadecimal number consisting of four digits.



- (9) Press "A-ENT". "END" is displayed.



2. Shift two 2716's data to 2732 (only by changing the RAM start address).



- (1) Set 2716 by the DEVICE select switch.
- (2) Insert the first 2716 into the socket, and press "COPY" and "START" in that order.
- (3) Press "EDIT" and "PAE" in that order.



- (4) Press " \uparrow ". (No ROM start address is changed.)



(5) Press "↑". (No ROM stop address is chagned.)

Ⓐ PASS	ADDRESS	DATA
Ⓑ FAIL	0000	

(6) Enter the memory (RAM) start address as 0800_H.

Ⓐ PASS	ADDRESS	DATA
Ⓑ FAIL	0000	0800

(7) Press "A-ENT".

Ⓐ PASS	ADDRESS	DATA
Ⓑ FAIL	End	

(8) Insert the second 2716 (No. 2) into the socket, and press "COPY" and "START" in that order.

(9) Set 2732 by the DEVICE select switch.

(10) Program 2732.

Through the above operations, two 2716's data can be shifted into 2732. By the same procedure, four 2716s and two 2732s (2532s) may be shifted into 2764 (2564).

6-1-2 CH (Change mode)

Described here is the operating method for altering data stored in the memory for programmed data.

1. Press the "EDIT" switch. (The EDIT lamp located on the display panel flashes.)

2. Press the "CH" switch.

Ⓐ PASS	ADDRESS	DATA
Ⓑ FAIL	-CH-	

3. Enter an address in a four-digit hexadecimal number.

Ⓐ PASS	ADDRESS	DATA
Ⓑ FAIL	-CH-	0123

In this example, 0, 1, 2 and 3 are entered.

4. Press "A-ENT". Pressing this switch causes the already entered hexadecimal number to be registered as the address.

Ⓐ PASS	ADDRESS	DATA
Ⓑ FAIL	0123	Rb

5. Enter data in a two-digit hexadecimal number. (In this example, 4 and 5 are entered.)

Ⓐ PASS	ADDRESS	DATA
Ⓑ FAIL	0123	Rb45

The new entered data is displayed here.

6. Press "↑" or "↓". Pressing these switches causes the already entered hexadecimal number to be registered as data.

Pressing "↑" causes the address to increment by one address, while pressing "↓" causes the address to decrement by one address.

If the address of data are subsequently entered, repeat step 5 and 6.

To enter data with a separate address, repeat from the step 3.

Ⓐ PASS	ADDRESS	DATA
Ⓑ FAIL	0124	CdCd

Pressing the "↑" advances one address. RAM data of address 124.

7. For getting out of this mode, press the "RESET" switch. The procedure also applies to the "INS" mode.

6-1-3 INS (Insert mode)

This mode inserts new data into memory data. In this mode, the data from inserted data on are moved by one address.

Operations in this mode are identical to those in the CH mode except for mode specification. However, the "↑" switch cannot be used.

Before data insertion After data insertion

Address

N-2	00	00
N-1	01	Data 10 is 01
N	02	inserted into 10
N+1	03	address N, 02
N+2	04	03

6-1-4 DEL (Delete mode)

This mode deletes memory data. This mode is broken down into one address deletion and block deletion.

Execution this operation causes the data from deleted data on to be left justified by one address.

Before data deletion After data deletion

Address

N-2	00	00
N-1	01	01
N	02	Address N is 03
N+1	03	deleted. 04
N+2	04	05

One address deletion

1. Press "EDIT".
2. Press "DEL".
3. Enter an address to be deleted in a four-digit hexadecimal number.
4. Press "A-ENT" to register the already entered hexadecimal number as an address.
5. Press "↑". Pressing "↑" causes the data of the already entered address to be deleted.

Block deletion

1. Press "EDIT".
2. Press "DEL".
3. Enter a deletion start address in a four-digit hexadecimal number.
4. Press "A-ENT" to register the entered hexadecimal number as an address.
5. Enter a deletion end address in a four-digit hexadecimal number.
6. Press "↑". Pressing "↑" causes the data between the entered start address and end address to be deleted.

6-1-5 INIT (INITIALIZE mode)

This mode initializes a memory to be programmed.

1. Press the "INIT" switch. Now, "-CL-" is displayed on the address display.
2. Enter a start address consisting of four digits, and press the "A-ENT" switch.
3. Enter an end address consisting of four digits, and press the "A-ENT" switch.
4. Enter initialize data consisting of two digits.
5. Press the "↑" switch. Now, the memory range specified through steps 2 and 3 is initialized by the data specified by the step 5.

Steps 2 and 3 may be omitted. In such a case, all memories are initialized. The step 4 may be omitted. In such a case, the selected ROM is initialized to an unprogrammed status.

6-1-6 FORMAT

This mode specifies a communication format to be used when Serial I/O is employed.

1. Press the "FORMAT" switch. Now, "-FO-" is displayed on the address display, and the number of a format currently selected format is displayed on the data display.
2. Specify a format to be selected using its number.
3. Press the "↑" switch.

Data format	Code number
Extended Tektronix Hex. (16 bits)	0
Standard Tektronix Hex. (8 bits)	1
Intel MCS-86 Hex. Object (16 bits)	2
Intel Intellic 8/MDS (8 bits)	2
Motorola Exormax (16 bits)	3
Motorola Exorciser (8 bits)	3
Texas Instruments SDSMAC (8/16 bits)	4
Minato Hex.	5
ASCII Hex. (Space)	6
Binary	7
Hewlett Packard 64000 Absolute	8

6-1-7 COMP (COMPLEMENT mode)

This mode inverts memory data.

1. Press the "COMP" switch. Now, "-CP-" is displayed on the address display.
2. Press the "↑" switch. On the termination of operation, "END" is displayed.

6-1-8 SER (SEARCH mode)

This mode searches for a data string in a memory. Data up to four bytes may be searched for.

1. Press the "SER" switch. Now, "-SH-" is displayed on the address display.
2. Enter a start address consisting of four digits, and press the "A-ENT" switch.
3. Enter an end address consisting of four digits, and press the "A-ENT" switch.
4. Enter data to be searched for using two digits, and press the "↑" switch. This step may be repeated up to four times.
5. Press the "START" switch. When specified data is found, its address is displayed.
6. If the "START" switch is pressed again, data begins to be searched from the subsequent address.
7. If the data is not found, "END" is displayed, and operation stops.

Steps 2 and 3 may be omitted. In such a case, the whole memory is subjected to this operation.

6-1-9 DEVICE

This mode selects a PROM to be programmed or read.

The PROM concerned may be selected by means of keys "0" through "1F" or the "↑" switch.

1. Press the "DEVICE" switch. Now, the name and number of a PROM currently selected are displayed on address

and data sections, respectively.

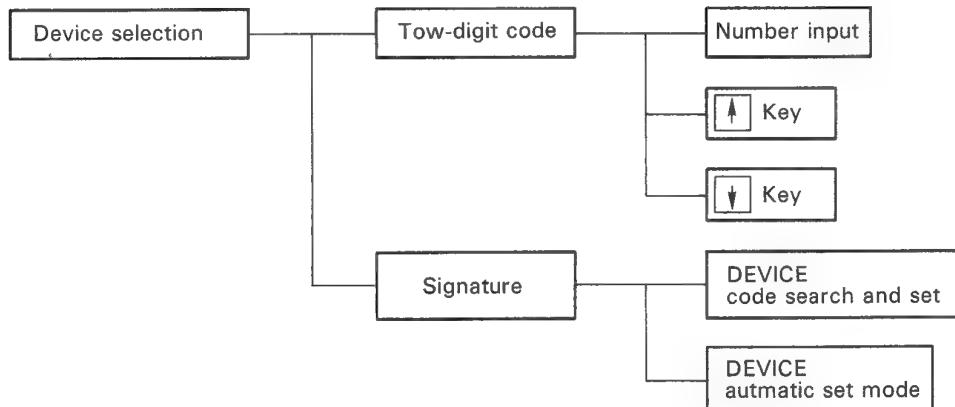
2. Select a PROM concerned using the "↑" or "↓" switch, or keys "00" through "1F".
- The switch "↑" permits selecting the PROM with the number following that of a PROM currently selected.
- The switch "↓" permits selecting the PROM with the number preceding that of a PROM currently

selected.

- Number "00" through "1F" permits directly selecting a PROM. The correspondence between number and PROM is shown in Figure.
- 3. Press the "↑" switch.
- 4. Press the "A-ENT" switch. Now, "END" is displayed and operation stop.

6-1-10 Device Selection

This mode selects PROM to be programmed and read. A PROM can be selected using a two-digit ROM code, \uparrow and \downarrow keys, and a silicon signature code.

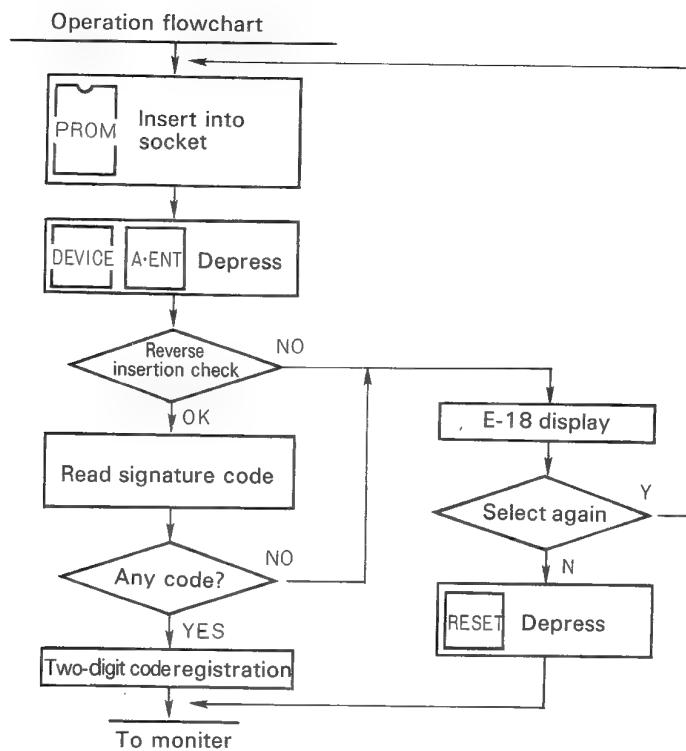


When a device is once selected, PROM operation range corresponds to its overall capacity. However, BUFF's operation start address remains the same.

PROM name display	Operation range	
	Capacity	Address
2716、48016、2815、2816 2816A、5213、27C16	2 KByt	0 0 0 0 ~ 0 F F F
2732、2732A、2532、2833 27C32、27C32A	4 KByt	0 0 0 0 ~ 0 F F F
2764、i 64H、i 64A H、F 64H、 2564、2564H、68764、2864A 58064、5764、64 b、A 64、27C64 C 64H、i C 64H、F C 64H	8 KByt	0 0 0 0 ~ 1 F F F
27I28、i I28、i I28H、i 286 A I28、C I28、C I28H、F C I28	16 KByt	0 0 0 0 ~ 3 F F F
i 256、256H、2566、A 256 C 256、F C 255	32 KByt	0 0 0 0 ~ 7 F F F
27512、27513、C 512Signature mode	64 kByt	0 0 0 0 ~ F F F F
		To be determined

• Device code search and set

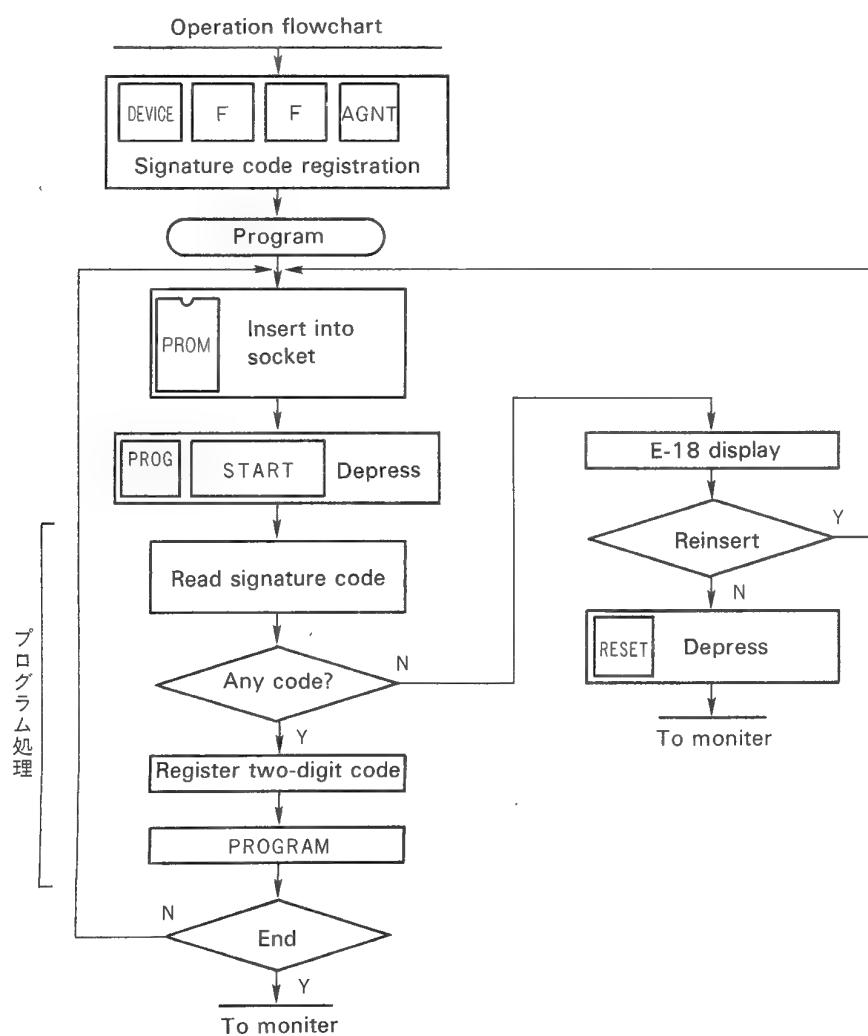
When a device is selected with a signatured PROM inserted into a socket, a corresponding two-digit code for the PROM is registered. When PROM has no signature, or when PROM's signature code is not registered to 1866A, error "E-18" is displayed. In this case, device selection state is as of before operation.



• Device automatic set mode

When signature code "FF" is registered at device selection, PROM's signature code is read, and then corresponding read/write method and capacity for the PROM are automatically set and operation starts every time [start] key is depressed. However, when PROM has no signature, or when PROM's signature code is not registered to 1866A, error "E-18" is displayed. In this case, select a device using a two-digit code.

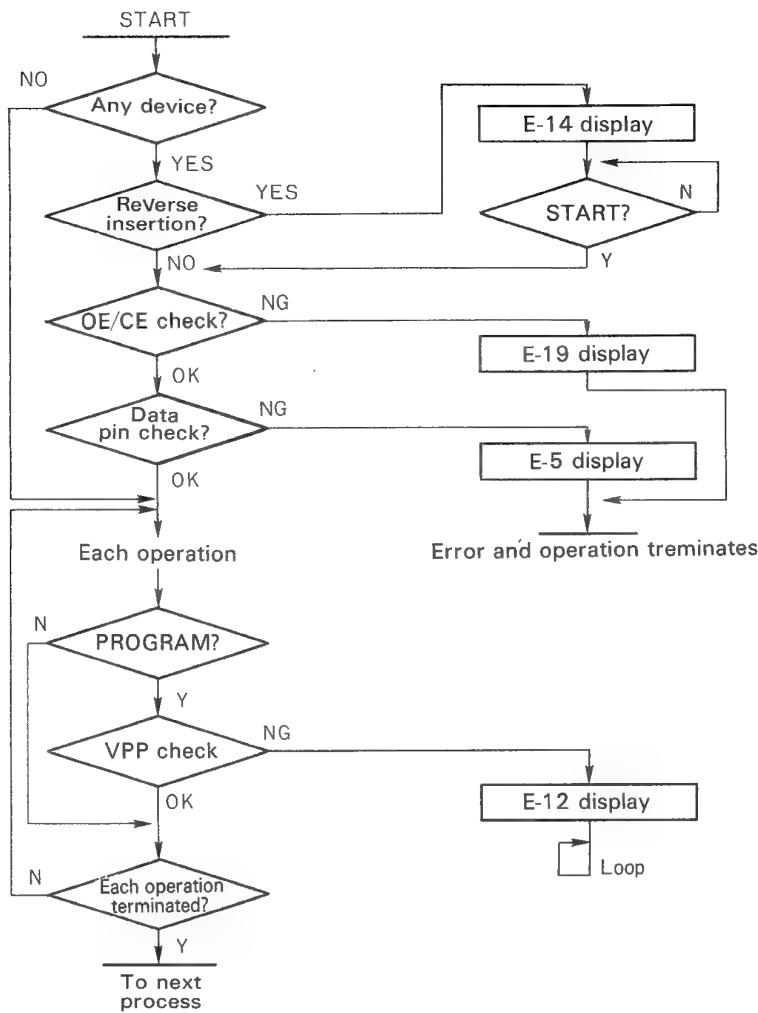
Note : In this mode, PROM to be programmed is not identified until operation starts. Also, setting of operation range and use of block mode can not be done. When the above setting is required, perform selection using a two-digit code, or code search and set mode.



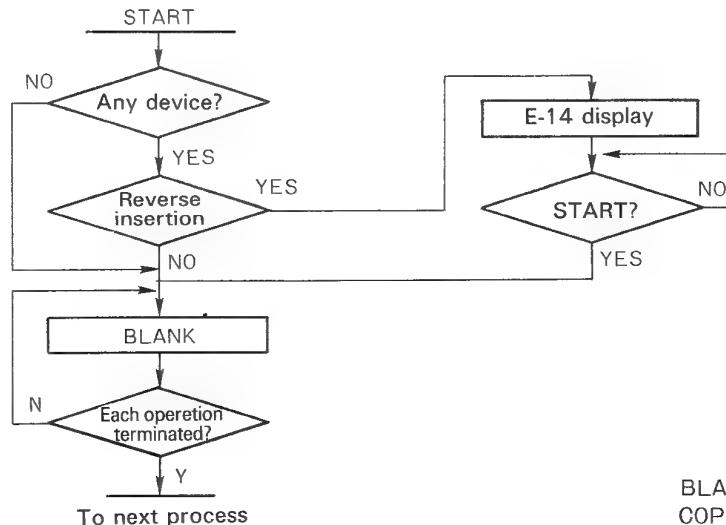
Operation flowchart

- When a device is registered:

- COPY, PROGRAM, VERI FY



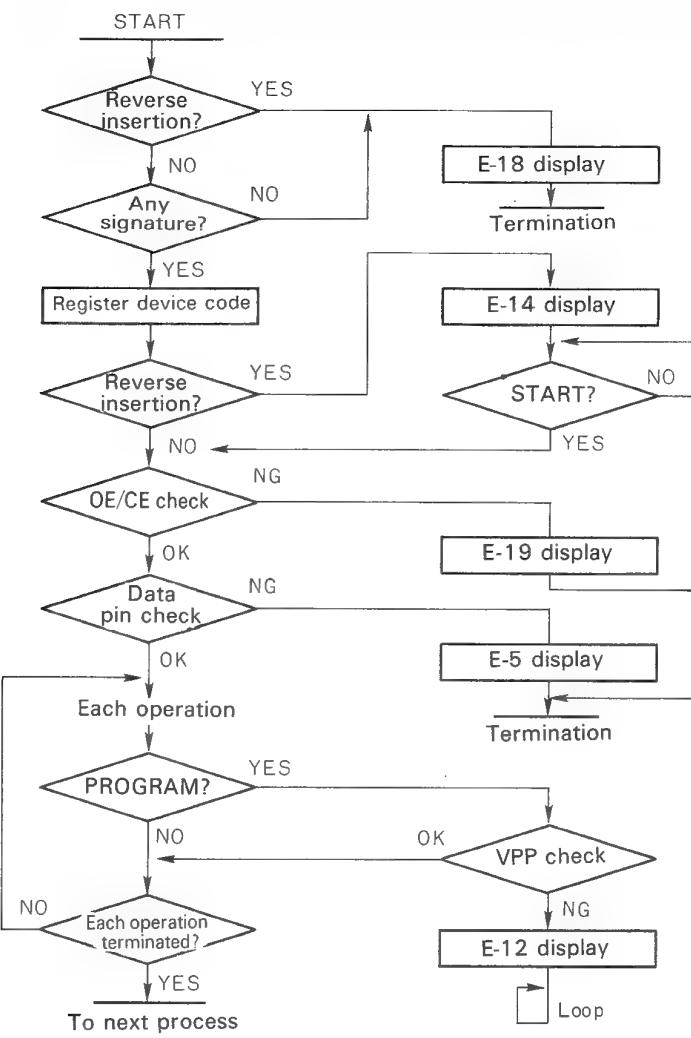
- BLANK



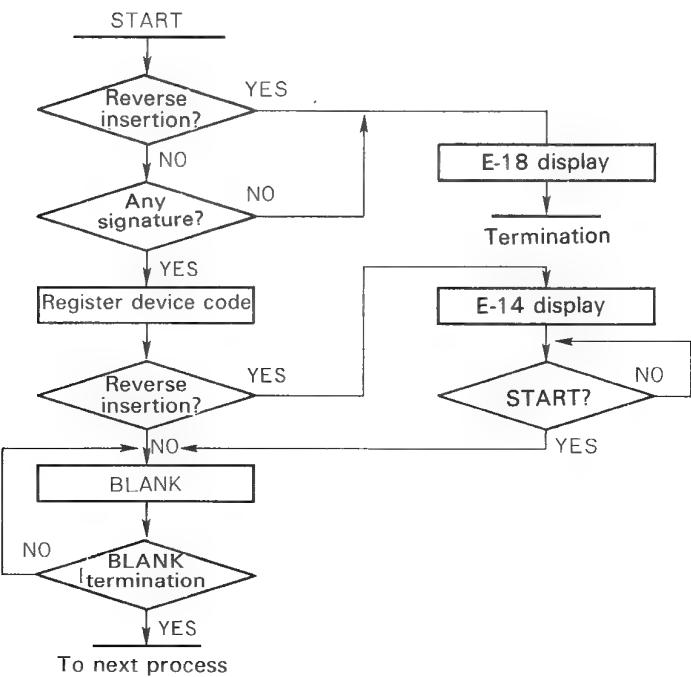
BLANK
COPY, PROGRAM, VERI FY

When a device has a signature code:

◦ COPY, PROGRAM, VERIFY



◦ BLANK



6-1-11 List of Device Codes
NMOS type list

Device code	Display	Covered PROM	Programming Algorithm	Programing voltage
00	2716, .00	i2716 or equivalent	Standard(50mS)	25
01	2732, .01	i2732 or equivalent	Standard(50mS)	25
02	2732A, .02	i2732A or equivalent	Standard(50mS)	21
03	2532, .03	TMS2532 or equivalent	Standard(50mS)	25
04	2764, .04	i2764 or equivalent	Standard(50mS)	21
05	.64H, .05	i2764 or equivalent	Intelligent- I	21
06	.64RH, .06	i2764A or equivalent	Intelligent-II	12.5
07	F64H, .07	MBM2764 or equivalent	Quick	21
08	2564, .08	TMS2564 or equivalent	Standard(50mS)	25
09	2564H, .09	TMS2564 or equivalent	Quick(12ms)	25
0A	68764, .0A	MCM68764 or equivalent	Loop	25
0B	27128, .0B	i27128 or equivalent	Standard(50mS)	21
0C	.128, .0C	i27128 or equivalent	Intelligent- I	21
0D	.128R, .0D	i2128 or equivalent	Intelligent-II	12.5
0E	F.128H, .0E	MBM27128 oz equivalent	Quick	21
0F	.256, .0F	i27256 or equivalent	Intelligent-II	12.5
10	256H, .10	TMM27256 or equivalent'	Quick(20ms)	21
11	27512, .11	i27512 or equivalent	Standard(60mS)	12.5
12	48016, .12	HN48016 or equivalent	Quick(10ms)	25
13	2815, .13	i2815 or equivalent	Quick(10ms)	21
14	2816, .14	i2816 or equivalent	Quick(10ms)	21
15	2816A, .15	X2816A or equivalent	Quick(10ms)	5
16	5213, .16	5213 or equivalent	Quick(10ms)	5
17	2864A, .17	X2864A or equivalent	Quick(10ms)	5
18	2833, .18	MCM2833 or equivalent	Quick(10ms)	5
19	58064, .19	HN58064 or equivalent	Intelligent-II	5
1A	5764, .1A	LH5764J or equivalent	Quick	12.5
1B	27513, .1B	i27513 or equivalent	Intelligent-II	12.5
Spare				
20	64H, .20	iP2764A or equivalent	Intel quick	12.75
21	128H, .21	iP27128A or equivalent	Intel quick	12.75
22	256H, .22	iP27256A or equivalent	Intel quick	12.75
23	512H, .23	iP27512A or equivalent	Intel quick	12.75
Spare				
28	A64, .28	AM2764A or equivalent	AMD quick	13.0
29	A128, .29	AM27128A or equivalent	AMD quick	13.0
2A	A256, .2A	AM27256A or equivalent	AMD quick	13.0
2B	A512, .2B	AM27512A or equivalent	AMD quick	13.0
Spare				
FF	FFFF, FF	Signature code		

* As of June'86(Above List is subject to change without notice)
* Programming methods

Intelligent : Intelligent programming algorithm(Intel methods)

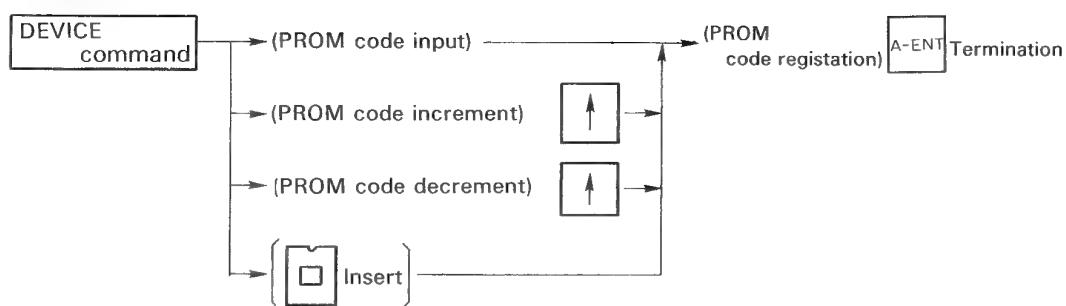
Intel Quick : Quick Pulse Programming algorithm

Quick : Quick programming method

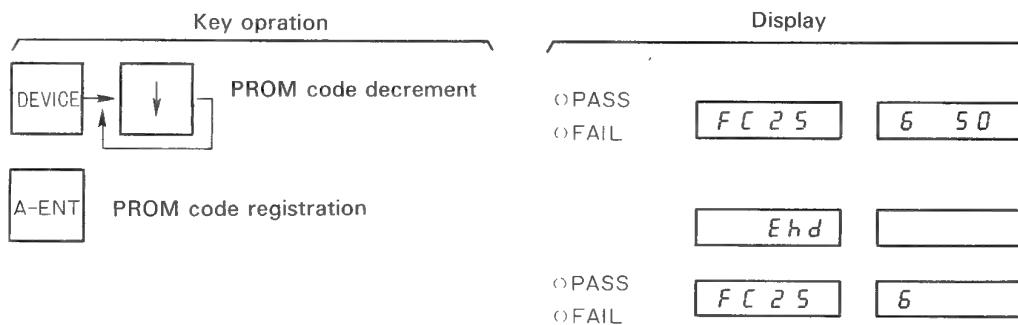
Standard : 50 mS Pulse

6-1-12 List of Device Codes CMOS type list

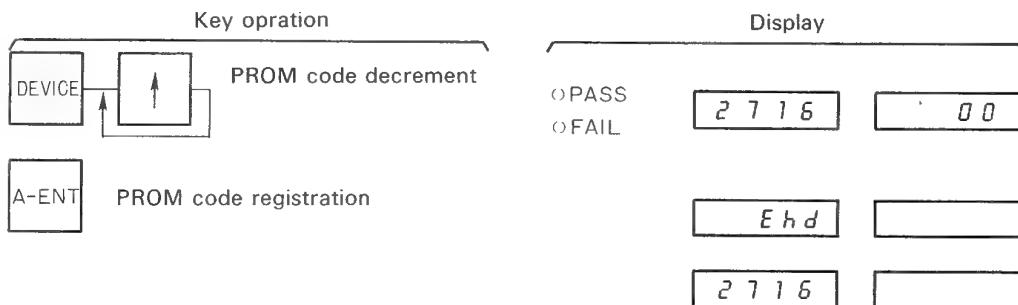
Operation method



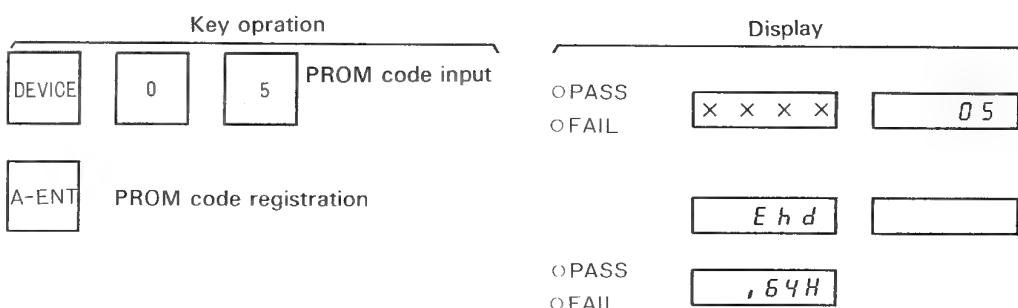
MBM27C256 selection method(using decrement)



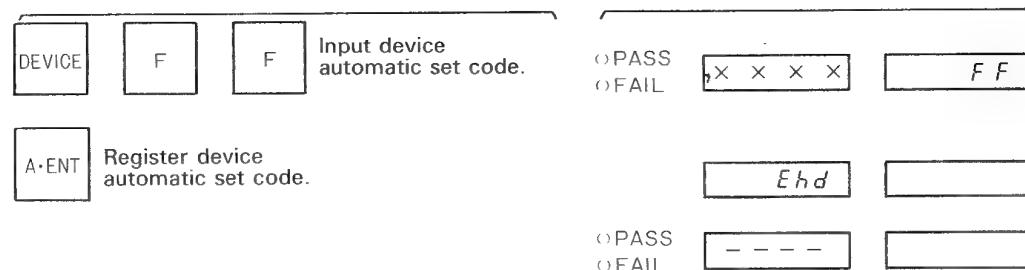
2716 selection method(using increment)



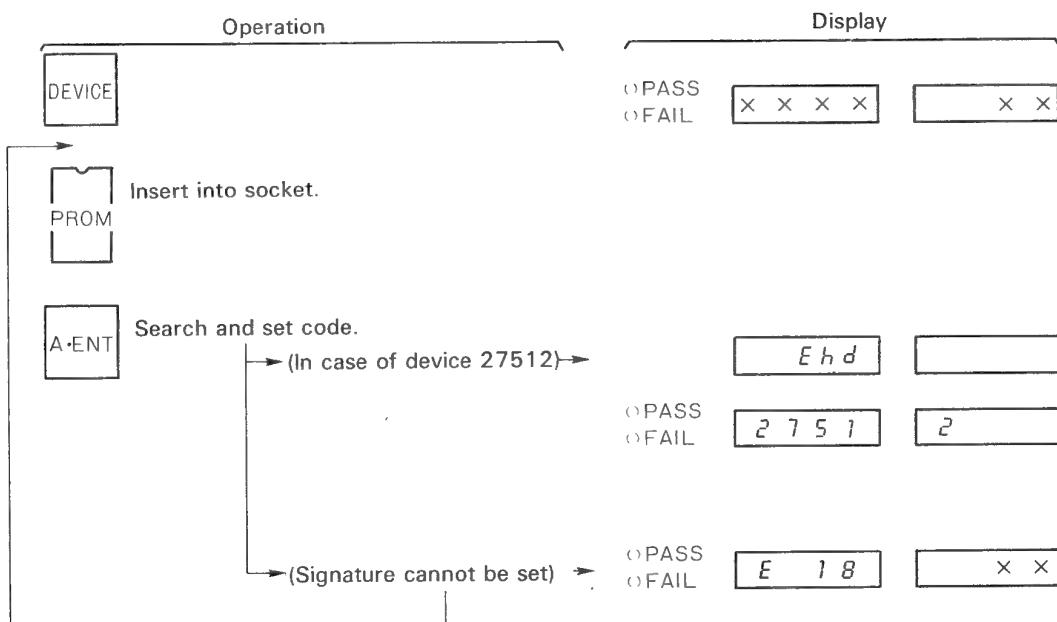
i27C64(Intelligent-I)selection method(using code input)



◦ Device automatic set mode



◦ Device code search and set



6-2 SERI (SERIAL I/O mode)

This mode enters data from the serial I/O port.

6-2-1 Entering Data

1. Press the "SERI" switch to set up the serial I/O mode.
2. Press "0" and "START". Now, data is entered in accordance with a selected format.

6-2-2 Outputting Data

1. Press the "SERI" switch to set up the serial I/O mode.
2. Press "1" and "START". Now, data is output in accordance with a selected format.

6-2-3 Issuing List

1. Press the "SERI" switch to set up the serial I/O mode.
2. Press "2" and "START". Now, the list is issued.

6-2-4 Outputting Control Code

1. Press the "SERI" switch to set up the serial I/O mode.
2. Press "3" and "START". Now, the code is output as follows:

Key Number	Control Code	Remarks
0	SOH (01H)	
1	STX (02H)	
2	ETX (03H)	
3	EOT (04H)	
4	ENQ (05H)	
5	DC1 (11H)	X-ON
6	DC2 (12H)	
7	DC3 (13H)	X-OFF
8	DC4 (14H)	
9	ETB (17H)	
A	EM (19H)	
B	SUB (1AH)	End Code of "MDS", End Code of "CP/M"
C	ESC (1BH)	
D	NUL (00H)	Spare
E	NUL (00H)	Spare
F	NUL (00H)	Spare

6-2-5 Handshake

This programmer adopts two methods for handshakes between devices to be applied when the serial I/O port is used.

1. RS232C's control wire

The first method utilizes the RS232C's control wire. This is the easiest method for connection to a device which uses the control wire.

2. X-ON/X-OFF signal

This method is utilized when the RS232C's control wire is not used, or for interfacing with a current loop.

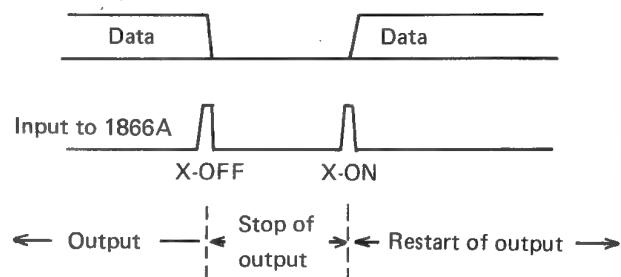
This method uses an X-ON signal to request data transmission, and an X-OFF signal to stop data transmission. X-ON and X-OFF signals are output to a line to transmit/receive data.

This Programmer uses a code of "11H" for the X-ON signal, and a code of "13H" for the X-OFF signal.

Example 1:

This example outputs data from the 1866A.

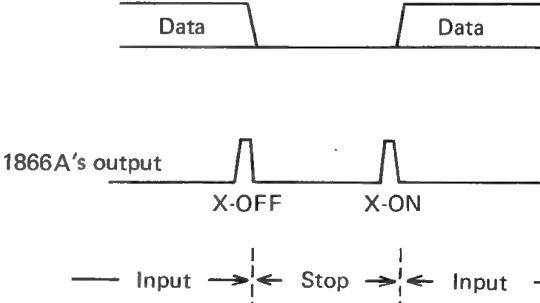
1866A output data



Example 2:

This example enters data from the 1866.

Input to 1866A



6-3 COPY

This mode enters data from a programmed PROM.

1. Press the "COPY" switch to set up the copy mode. When the copy mode is set, the COPY LED begins to flash.
2. Insert a master PROM into a socket.
3. Press the "START" switch. When the "START" switch is pressed, the Programmer begins to operate.

4. After all data have been read, the "PASS" LED flashes and the check sum value ^{*1} is displayed, and operation stops.
- The copy mode reads two data from a PROM and compares the data with each other to detect reading errors. When a reading error is detected, the "FAIL" LED flashes and an error message ^{*2} is displayed, and operation stops.
- The copy mode does not perform "STEP" operation. In the copy mode, the address specified in the "PAE" mode is invalid. However, the "RAM START" address is valid.
- In the copy mode, the address mode ^{*3} is valid.
- The order of executing steps 1 and 2 is left to the user's discretion.

^{*1}: See 6-11.

^{*2}: See 6-12.

^{*3}: See 6-9.

6-4 BLANK

This mode checks a PROM's unprogrammed or delete state.

1. Press the "BLANK" switch to set up the blank mode. When the blank mode is set up, the BLANK LED flashes.
2. Insert a PROM into a socket.
3. Press the "START" switch. Now, operation starts. When an error is found, its address data is displayed and operation stops. When the "↑" switch is pressed, operation restarts from the next address. If the "START" switch is pressed, operation is carried out from the start.
4. After the termination of checking all addresses, the "PASS" LED lights and operation stops.
- The blank check mode is capable of step operation.
- In the blank check mode, every PAE is valid.
- The order of executing steps 1 and 2 is left to the user's discretion.
- In the blank check mode, the address mode is valid.

6-5 PROG (PROGRAM) Mode

This mode programs a PROM.

1. Press the "PROG" switch to set up the program mode. When the program mode is set up, the "PROG" LED flashes.
2. Insert a PROM into a socket.
3. Press the "START" switch. Now, program operation starts.
4. After all addresses have finished being programmed, the "PASS" LED lights and the check sum is displayed, and operation stops. If an error is found, the same processing as the blank check is conducted.

- The program mode, on the termination of programming, carries out the verify check.
- For notes, see 6-4 "BLANK".

6-6 VER (VERIFY) Mode

This mode compares the contents of a programmed PROM with those of the RAM to check if it has been programmed normally.

1. Press the "VER" switch to set up the verify mode. When the verify mode is set up, the "VER" LED flashes.
2. Insert a PROM into a socket.
3. Press the "START" switch. Now, the verify check starts.
4. After the termination of programming all addresses, the "PASS" LED lights and the check sum is displayed, and operation stops. If an error is found, the same processing as the blank check is conducted.
- Verify operation is conducted four times, as conditions are varied.
- For the other notes, see 6-4 "BLANK".

6-7 CONT (CONTINUOUS) Mode

This mode provides three operations of Blank check, Program and Verify continuously.

For operations, see descriptions for the above three modes.

6-8 REM (REMOTE) Mode

This mode remotely controls the Programmer from the outside using the serial I/O port. For details of this function, see Chapter 7.

6-9 Address Mode Selection

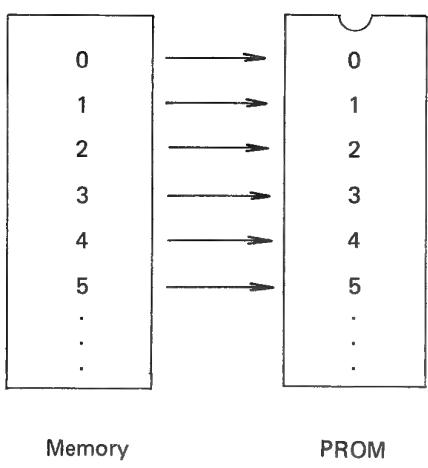
The Programmer, in copy, program and verify operations, is capable of selecting an address to be programmed.

If the two address mode indication LEDs go out, this signifies that the normal address mode has been set up. The successive memory addresses will be programmed.

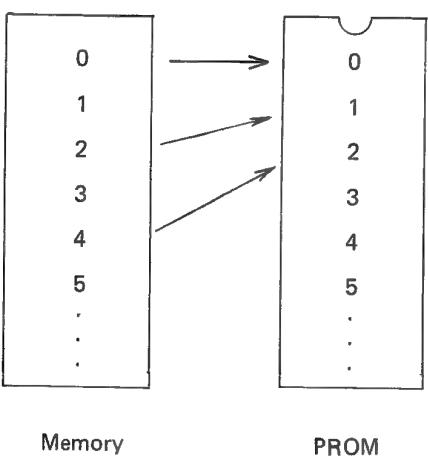
If the address mode display LED "EVEN" lights, this signifies that the even address mode has been set up. Only the even memory addresses will be programmed.

If the address mode display LED "ODD" lights, this signifies that the odd address mode has been set up. Only the odd memory addresses will be programmed.

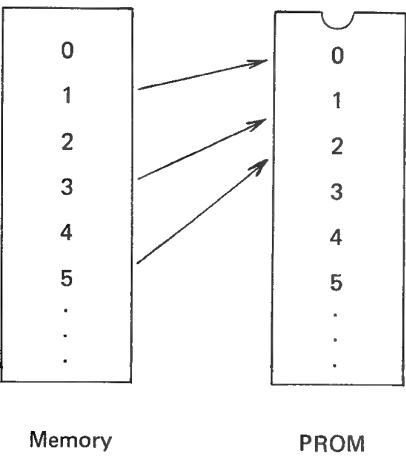
Normal address mode



Even address mode



Odd address mode



6-10 AUTO/STEP Selection

In Blank, Program, Verify and Cont operations, Auto and Step modes are valid.

Auto/Step is selected by pressing the "AUTO/STEP" switch. This switch is an alternate type.

In the step mode, operation is stopped by one address execution. Pressing the "START" switch causes the next address to be subjected to execution.

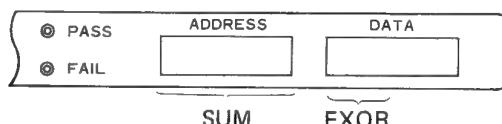
During the step mode, the auto mode may be set up.

The auto mode provides successive execution from the start. During the auto mode, the step mode cannot be set up.

6-11 Check Sum

This Programmer, at the end of the Copy, Program or Verify mode, generates and displays a PROM's check sum.

The check sum is obtained by adding and exclusively ORing data by address, and is displayed as shown below.



In addition, the check sum produced after RAM data in the programmer has been changed may be displayed.

1. Press the "BLANK" switch.
2. Press the "0" switch.
3. Press the "START" switch.

6-12 Error Messages

This Programmer, when power is turned on or during operation, carries out a number of checks. When an error is detected by these checks, the error message is displayed, and operation stops.

6-12-1 Table of error codes

E-1	Memory Error
E-2	Parity error (RS-232C)
E-3	Format error (Data format)
E-4 ADRS	Address entry error
E-5	PROM Socket data pin error
E-6	Uncataloged format
E-7	Load verify error
E-8	Instruction specify error
E-9 ERAS	Erase error
E-10	Buffer over error (RS-232C)
E-11	RS-232C DSR undetected error
E-12 m ^{*1}	Vpp error

E-13	n ^{*2}	Vcc error
E-14		Reverse insertion error
E-15		Other error
E-16		Power-on protection error
E-17		Time-out error

*1: m refers to numerics 1 through 9.

*2: n refers to alphabetics A through D.

The numerics and alphanumerics signify:

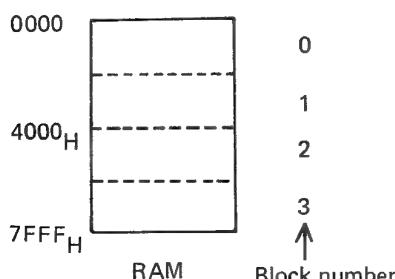
1	Vpp 12.5V	Pin 23
2	Ditto	Pin 21
3	Ditto	Pin 1
4	Vpp 21.0V	Pin 23
5	Ditto	Pin 22
6	Ditto	Pin 1
7	Vpp 25.0	Pin 23
8	Ditto	Pin 22
9	Ditto	Pin 1
A	Vcc 4.75V	
B	Vcc 5.00V	
C	Vcc 5.25V	
D	Vcc 6.00V	

6-13 Block Operations in Copy, Program, Verify and Cont modes

Usually, the RAM's addresses used in Copy, Program, Verify and Cont modes are specified by the PAE mode.

This mode, irrespective of specifications in PAE, splits the RAM into several blocks by selected ROM's capacity, numbers the blocks, and specifies RAM's locations to be used by these numbers.

In the following example, the 2764 (8 k-bytes) is selected.



The 1866A's RAM is split into 4 blocks composed of 8K-bytes.

Operations

1. Press the "↑" switch.
2. Enter a block number.
3. Press the "START" switch.

The subsequent operations may be carried out only by the "START" switch. The set block number may be altered and this mode may be released by pressing an operation mode switch such as "COPY" or the "RESET" switch.

6-14 Setting Serial I/O Mode

The serial I/O's baud rate, parity, stop bit and X-ON/X-OFF may be selected by means of the bottom switch.

1. Baud rate selection

The desired baud rate may be selected by switch's three bits.

67	STOP BIT
00	—
10	1BIT
01	1.5BIT
11	2BIT
45	PARITY CHECK
0X	OFF
10	ODD
11	EVEN
123	BAUD RATE
000	110
100	300
010	600
110	1200
001	2400
101	4800
011	9600

8	X-ON/X-OFF
7	—
6	STOP BIT
5	—
4	PARITY
3	—
2	BAUD RATE
1	—
0	OFF

This Programmer is capable of setting 110 bauds to 9600 bauds.

2. Parity selection

The desired parity may be selected by switch's bits 4 and 5. Any of no parity, even and odd may be selected.

3. Stop bit selection

The desired stop bit may be selected by switch's bits 6 and 7. Any of 1 bit, 1.5 bits and 2 bits may be selected.

4. X-ON/X-OFF selection

X-ON/X-OFF may be selected by the switch's bit 8.

6-15 Others

Three operations of Key Lock, Test and Repeat may be selected by means of the bottom switch.

6-15-1 Key Lock

Turning on this switch makes any switch operations other

than "START" invalid.

If this mode is used, any RAM data is not destroyed by erroneous operations, which contributes considerably to mass production.

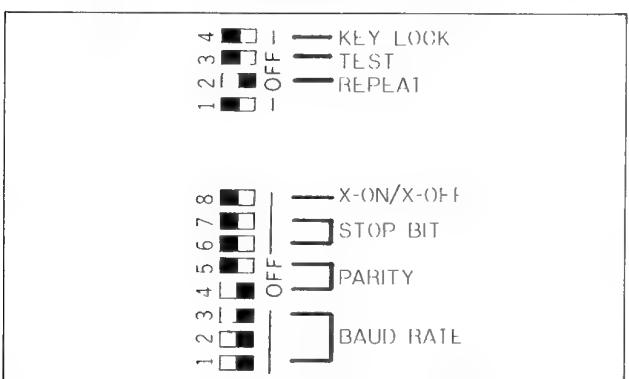
6-15-2 Test

Turning on this switch causes the 1866A to be put in the test mode. The test mode is capable of checking voltages and waveforms, etc. with ease.

Caution: In the test mode, any PROM must not be inserted into a socket. In the test mode, many of the protection functions the 1866A provides are not operated. Note that erroneous operations may destroy the 1866A and PROMs.

6-15-3 Repeat

Turning on this switch causes Copy, Blank, Program, Verify and Cont operations to be executed repeatedly.



6-15-4 Power-on protection

If power is on with PROM being inserted, "E-16" is indicated on the display to stop operation. (PROM could be damaged.)

6-15-5 Time-out function

Indication "E-17" is given in the display when time-out so as to put out "?" into the peripheral equipment, if the data input from peripheral is not made for about 120 seconds upon absence of data reception as Serial mode, or the command of "RL", "RLn", "RLV", or "RLVn" at Remote mode.

7

REMOTE OPERATION

The PROM programmer is provided with a Serial Interface as the standard. When it is connected to an external terminal equipment, the following functions are available by software:

No.	Command	Content
1	A	Address select.
2	B	Device blank check.
3	Bo	Checksum output.
4	C	Prepare and correct data.
5	D	Data deletion.
6	E	Release remote operation.
7	F	Initialize memory.
8	G	Execute within memory (not to be used)
9	I	Data insertion
10	L	Output list.
11	MD	Output operation address range.
12	MS	Assign device start address.
13	MP	Assign device end address.
14	MR	Assign memory start address.
15	N	Device select.
16	OP	Copy.
17	OPn	Copy specified block.
18	OT	Continue.
19	OTn	Continue specified block.
20	P	Output data using data format. (0 ~ 32K)
21	RL RL,xxxx	Input data format. (0 ~ 64K)
22	RL1	// (0 ~ 32K)
23	RL2	// (32 ~ 64K)
24	RLV RLV,xxxx	Compare data using data format. (0 ~ 64K)
25	RL1	// (0 ~ 32K)
26	RL2	// (32 ~ 64K)
27	S	Assign data format.
28	T	Move data.
29	V	Verify
30	Vn	Verify specified block.
31	W	Program.
32	Wn	Program specified block.
33	Z	EEPROM chip erase
34	ZB	EEPROM byte erase

The signal formats for terminal equipment connected are RS-232C and 20mA current loop. The tape parity and baud rate are vary depending on the terminal equipment connected;

for their setting, refer to Chapter 8.

Remember, in the description which follows, the hexadecimal notation (0-9 and A-F) is exclusively used and that any underlined character is an entry from terminal equipment.

7-1 Terminal equipment Connection

The PROM programmer can be remotely operated if it is connected to a terminal equipment. For such information as connector pin number, refer to 8 Serial Interface.

7-2 Start of Remote Operation

This program start its operation by selecting "REM" (REMTer) out of the function switches located on the panel of the programmer and displays "#", being put into instruction wait status.

7-3 Creating New Data

The change instruction or the insert instruction described later is used to create new data.

7-3-1 How to use Change Instruction

- Clear all memories by means of the "F" instruction.
This operation is not always required.
- By the use of the change instruction, specify an address where new data is to be created. Specifying an address causes the address specified on the terminal equipment and the contents stored in the address to be displayed, and provided the wait status of input from the terminal equipment. (See example 1.)
- If the wait status of input from the terminal equipment is provided, enter data. Entering data causes the data stored in the next address to be displayed and then provided wait status for the subsequent data.
- After data is created, enter "CR". Entering "CR" causes "#" to be displayed in the terminal equipment and then provides wait status for the subsequent instruction.
- Using the change instruction allows eight (8) data to be created in one line. The 9th data is created in the next line.
- Using an erroneous character (a character other than characters 0 through F) provides the shift to the start

of the next line, causes "?" to be displayed, provides the shift to the start of the next line, causes "#" to be displayed, and then provides wait status for the subsequent instruction. (This operation is common in all instruction.) (See example 2.)

(NOTE)

In remote operation, any address can consist of up to 4 digits in entry. Notice, however, 8 or more at the highest digit will be masked; for example, 8100 results in 0100. This applies to all the address entry which follows.

Example 1: Normal Case

#C . 0 CR

(1) (2) (3)

0000 00-11 00-22 00-33 00-44 00-55 00-66 00-77 00-88
(4) (5)(6) (6) (6) (6) (6) (6) (6)
0008 00-99 00-AA 00-BB 00-CC 00-DD 00-EE 00-FF 00-CR
(6) (6) (6) (6) (6) (6) (6) (7)

- (1) Enter "C" which signifies the change instruction.
- (2) Enter an address to be changed (or to be created) in not more than 4 digits.
- (3) Enter "CR" at the end of the instruction.
- (4) The specified address is displayed.
- (5) The contents of the specified address are displayed.
- (6) Enter new data.
- (7) After all data has been entered, enter "CR" at the end.

Example 2: Where Erroneous Character is Entered.

#C . 100 CR

0100 00-11 00-AA 00-BG

(1)

?
(2)

(3)

- (1) "G" is other than characters 0 through F and is an erroneous input.
- (2) The shift to the start of the next line is provided and "?" is displayed.
- (3) The shift to the start of the next line is provided again, "#" is displayed, and wait status for the subsequent instruction is provided.

(Note) In that case, the contents before an address to which "BG" has been entered are changed as specified, while the contents of the address to which "BG" has been entered are not changed.

7-3-2 How to Use Insert Instruction

- (a) Clear all memories. (This operation is not always required.)
- (b) Specifying an address where new data is to be created by the use of the insert instruction and entering "CR" provide the shift to the start of the next line and cause an stored data to be displayed. Enter new data to be created. After entering data, enter "SP" and proceed to the subsequent address. (See example 3.)
- (c) After all data has been entered, enter "CR" at the end, which provides the shift to the start of the next line, causes "#" to be displayed, and provides wait status for the subsequent instruction.
- (d) In entered data, two characters before "SP" or "CR" are valid and the other characters are ignored. (See example 4.)

Example: 3

#I . 0 CR

(1) (2) (3)

0000 00 11 22 33 44 55 66 77 88 99 AA BB CR
(4) (5)(6)(5) (7)

- (1) Enter "I" which signifies the insert instruction.
- (2) Enter an address to be inserted in alphanumeric consisting of not more than 4 digits.
- (3) Enter "CR" at the end of the instruction.
- (4) The specified address is displayed.
- (5) Enter new data.
- (6) Enter "SP" between data.
- (7) After all data has been entered, enter "CR".

The contents of the memory after this example has been executed are:

#L . 0 , B CR

0000 00 11 22 33 44 55 66 77 88 99 AA BB

Example: 4

#I . 5 CR

0005 1234 5678 CR

(1) (2)

- (1), (2) In both (1) and (2), data is entered in four digits. Valid data is two characters before "SP" and "CR". In (1), "3" and "4" are valid, while, in (2), "7" and "8" are valid. The contents of the memory after the instruction has been executed are:

#L . 0 , E CR

0000 00 00 00 00 00 34 78 00 00 00 00 00 00 00 00
Bank 1 0000-7FFF, Bank 2 Bank 8000-FFFF

When assignment of data elimination is through above 2 Banks, an error is generated.

When an inserted address exceeds 7FFF or FFFF, operation terminates.

7-4 Modifying Data

The change instruction "C" is used to modify data. The using method is identical to that described in the subsection "Creating New Data".

Example: 5

#L . 0 , F CR

0000 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 00 0E OF

Shown above are the contents of the memory before modification.

#C . D CR

(1) (2)

000D 00-0D 0E-CR

(3) (4) (5) (6)

(1) Enter the change instruction "C".

(2) Enter an address to be changed.

(3) The address to be changed is displayed.

(4) The contents of the address to be changed are displayed.

(5) Indicates new data.

(6) After change, enter "CR".

Shown below are the contents of the memory after modification. The characters enclosed by a rectangle are the changed portion.

#L . 0 , F CR

0000 00 01 02 03 04 05 06 07 08 09 0A 0B 0C [0D] 0E OF

7-5 Deleting Data

The delete instruction "D" is used to delete data. (See example 6.)

- Enter "D" which signifies the delete instruction.
- Enter a delete start address and a delete end address. And then enter "CR".
- An entered delete start address, if greater than a delete end address, is an erroneous input. (See example 7.)
- To delete only one address, enter only the delete start address and enter "CR". (See example 8.)

Example: 6

Shown below are the contents of the memory before the delete instruction is executed.

#L . 0 , 1F CR

0000 00 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF

0010 FF EE DD CC BB AA 99 88 77 66 55 44 33 22 11 00

Delete Instruction

#D . A , 15 CR

(1) (2) (3) (5)

(4)

- Enter "D" which signifies the delete instruction.

- Enter a delete start address in not more than four digits.
- Enter a delete end address in not more than four digits.
- Place, "," between addresses.
- Finally enter "CR". Entering "CR" causes operation to be executed.

Shown below are the contents of the memory after the delete instruction has been executed.

0000 00 11 22 33 44 55 66 77 88 99 99 88 77 66 55 44
0010 33 22 11 00 BB BB

Example of subsequent addresses data

Example: 7

#D . 150 , 100 CR

(1) (2) (3)

?

(4)

#

(5)

(1) Enter "D" which signifies the delete instruction.

(2) Indicate a delete start address.

(3) Indicates a delete end address. The delete and address, which is smaller than the delete start address (2), is an erroneous input.

(4) The shift to the start of the next line is provided and "?" is displayed.

(5) The shift to the start of the next line is provided, "#" is displayed, and wait status for the subsequent instruction is provided.

Example: 8

Shown below is an example for one-address deletion. The following are the contents of the memory before deletion. The characters enclosed by the rectangle are deleted.

#L . 0 , F CR

0000 00 01 02 03 04 [05] 06 07 08 09 0A 0B 0C 0D 0E OF
Enter an address to be deleted and then enter "CR".

#D . 5 CR

The following are the contents of the memory after deletion:

#L . 0 , F CR

0000 00 01 02 03 04 06 07 08 09 0A 0B 0C 0D 0E OF 10

Bank 1:0000-7FFF, Bank 2:8000-FFFF When assignment of data elimination is through above 2 Banks, an error is generated.

7-6 Inserting Data

The insert instruction "I" is used to insert data. The insert instruction allows new data to be inserted to an arbitrary address.

- Enter "I" which signifies the insert instruction.
- Enter an address to which data is to be inserted and

- (3) Place "," between addresses.
- (4) Enter the end address of a memory to be listed.
- (d) The end address of the list must be greater than its start address. Otherwise, the input is considered to be erroneous.

7-9 Punching Paper Tape

The punch instruction is used to write the contents of the RAM of a specified address to a paper tape by means of the TTY.

- (a) Enter "P" which signifies the punch instruction and the start and end addresses of a memory for punching. And finally enter "CR".

Example: 12

#P . 0 , 2F CR

- (1) (2)(4)(3)

#0000 40 C1 00 08 02 44 08 00 00 00 00 40 00 4C 0A CC
#0010 00 DC 00 44 01 C4 00 FC 00 F4 00CC 00 CC 00 CC
#0020 03 03 00 00 11 20 13 EB 33 A0 03 F0 00 B2 10 38

- (1) Enter "P" which signifies the punch instruction.
- (2) Indicates the start address for punching.
- (3) Indicates the end address for punching.
- (4) Place "," between addresses.

7-10 Reading Data from Paper Tape

The "R" instruction is used to read data on a paper tape from the TTY or the PTR.

Example: 13

#RL CR

Shown above is the read instruction from the reader of the TTY. Entering "CR" causes data to begin to be read. After data has been read, the shift to the start of the next line is provided, "#" is displayed, and wait status for the subsequent instruction is provided.

#RH CR

Shown above is the read instruction from the PTR.

7-11 Initializing Memory Contents

The "F" instruction is used to initialize an arbitrary address of the RAM located in the PROM Programmer for arbitrary data. The "F" instruction also allows a memory to be cleared. Enter "F" which signifies the initialize instruction, start and end addresses, and data, in that order. And finally enter "CR".

Example: 14

#F . 10 , 2F , AA CR

- (1) (2) (3) (4)

- (5) (5)

- (1) Enter "F" which signifies the initialize instruction.

- (2) Enter the start address.
- (3) Enter the end address.
- (4) Enter data.
- (5) Place "," between addresses and between an address and data.

Shown below is the result after execution of the instruction.

#L . 0 , 3F CR

0000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010 AA
0020 AA
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

7-12 Programmer Control

7-12-1 Operation Address Range Specification

The PROM programmer allows a terminal equipment to specify the range of operation in programs.

- 1) Display of operation range registered. An entry of "MD" allows the current registered operation range to be displayed.

#M D CR

xxxx 0000 △△△△

- ① ② ③

① ROM start address

② ROM stop address

③ RAM start address

- 2) ROM start address change

To change the ROM start address. Enter an "MS" which indicates the ROM start address is to be changed; the system displays the current registered address and waits for a new address to come. Enter a new address made up of 4 digits.

#M S CR

xxxx – 0000

- ① ②

① Current registered address

② Newly entered address

- 3) ROM stop address

To change the ROM address for stop operation. Enter an "MP" which indicates the ROM address for stop operation is to be changed; the system displays the current registered address and waits for a new address to come. Enter a new 4-digit address.

#M P CR

xxxx – 0000

- ① ②

- ① Current registered address
- ② Newly entered address

4) RAM start address

To change the RAM start address. Enter an "MR" which indicates the RAM start address is to be changed; the system displays the current registered address and wait for a new address to come.

#M R CR
xxxx - 0000

- ① ②

- ① Current registered address
- ② Newly entered address

7-12-2 Blank Check

Entering instructions through a terminal equipment permits the PROM programmer to perform blank checks. The operation is quite the same as the to all of 7-12-2 through 7-12-8.

#B CR

The "B" is a blank-check instruction. After the operation is completed, the system displays another "#" and waits for a new instruction to come. If the PROM is defective (or has been already programmed), the system displays a "?" and waits for a new instruction to come.

7-12-3 Program

Entering a program instruction "W" allows PROM program operation.

#W CR

If the program is defective, the system displays a "?".

7-12-4 Verify

Entering a verify instruction "V" allows a verify operation.

#V CR

If the verify operation fails, the system displays a "?".

7-12-5 Copy

Entering a copy instruction "OP" allows a copy operation.

#OP CR

If the copy operation fails, the system displays a "?".

7-12-6 Cont

Entering a CONT instruction "OT" allows a CONT operation.

#OT CR

#OTn CR (n = Block number at block mode)

If the blank, program or verify fails, the system displays a "?".

7-12-7 Format Specification

Entering an "S", format specification instruction, allows the format to be specified.

#S. O CR
 ↑
 O : MINATO format
 — Only the lowest digit is effective

7-12-8 PTR Verify

Entering a PTR verify instruction. "RLV", allows PTR verify operation to be carried out.

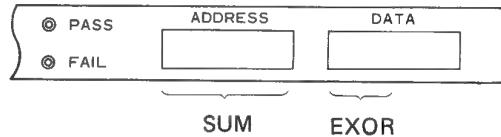
#RLV CR (Entry through TTY reader)

7-12-9 Buffer's check sum

Entering a buffer's (RAM) check sum instruction "BO" allows a buffer's check sum operation.

#BO CR

The check sum is obtained by adding and exclusively ORing data by address, and is displayed as shown below.



7-12-10 PROM device select

Entering a PROM device select instruction "N" allows PROM select instruction. (See a command number on page 23)

#N mm - nn CR

mm = The current device number is displayed.

nn = Entering a PROM select command number.

For the device number, two entered lower digits are valid. If only one digit has been entered, the second digits is assumed to be "0".

7-12-11 Address select

Entering a address select instruction "A" allows address select instruction.

#A m - n CR

m = The current address mode is displayed.

n = Entering address select command.

N: All address mode

m and n E: Even address mode

O: Odd address mode

7-12-12 EEPROM's erase

Entering a EEPROM chip erase instruction "Z" allows erase instruction.

#Z CR

7-12-13 EEPROM's byte erase

Entering a EEPROM chip's byte erase instruction "ZB"
allows byte erase instruction

#ZB CR

ADRS - xxx CR

↑ Address to be erased.

NOTE: Any remote operation has no data complement
instruction.

7-12-14 Remote Operation Release Command

The execution of this command releases the serial I/F
and returns to LOCAL mode.

Input method and output format.

E R L F * * △ S E R I A L △

E N D △ * * CR LF

8

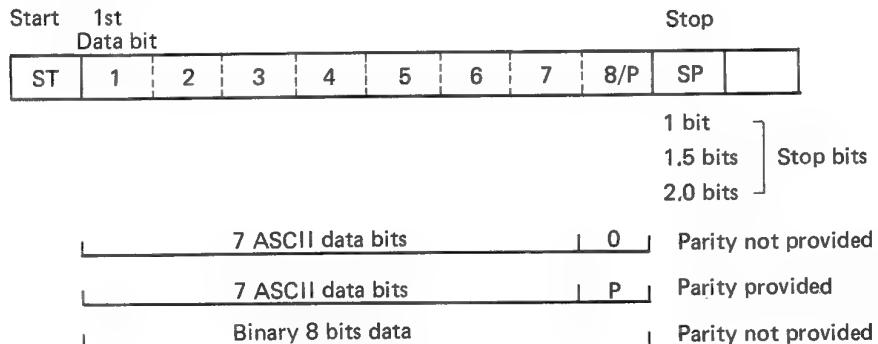
SERIAL INTERFACE

8-1 Introduction

The serial interface is broken down into an RS-232C level interface and a current loop interface. Presented below are descriptions of the meanings of the interface signals, and of the signal levels and transmission procedure.

8-1-1 Serial I/F ratings

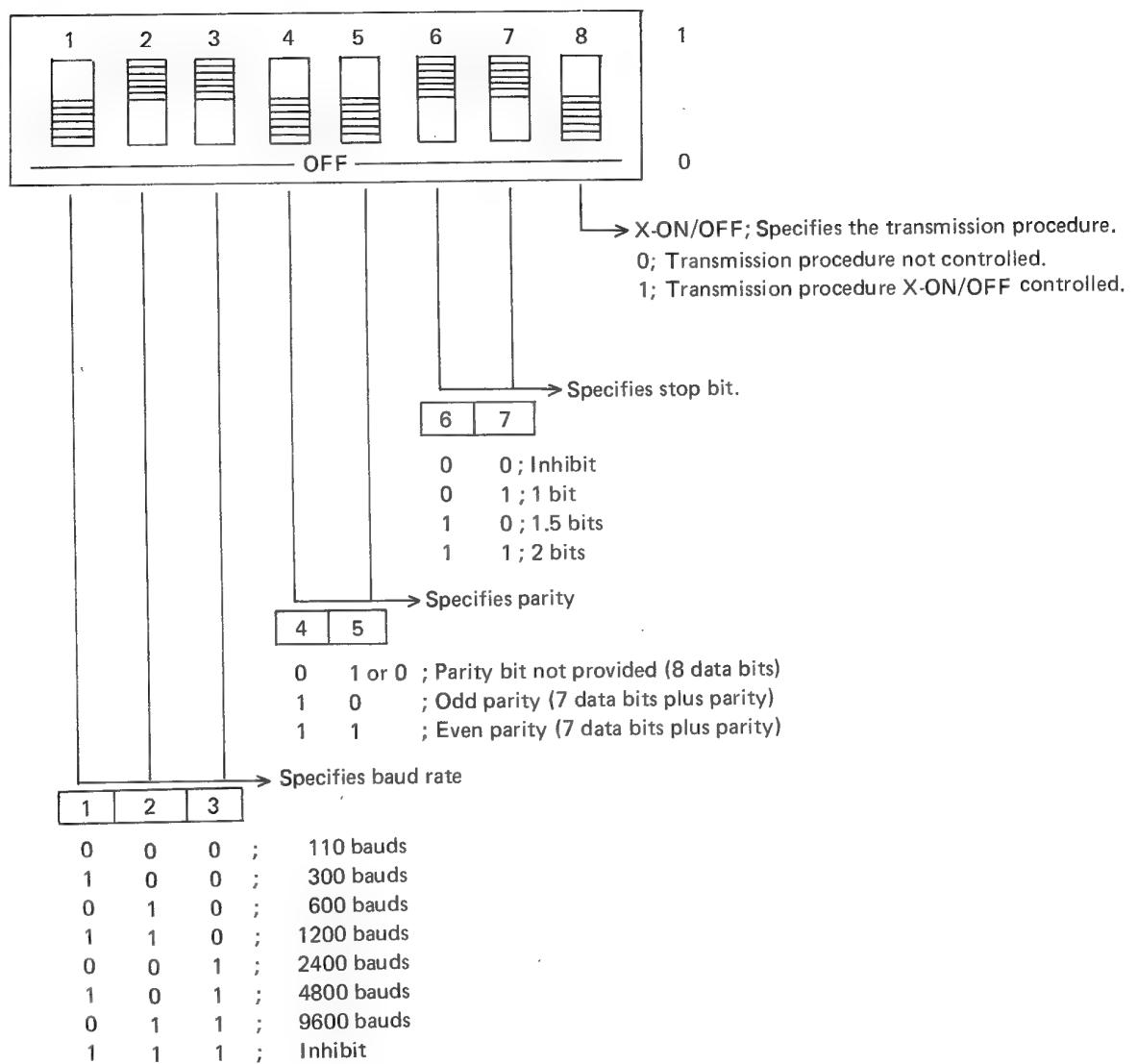
- (1) Baud rates: 110, 300, 600, 1200, 2400, 4800 and 9600
- (2) Start bit: 1 bit fixed
- (3) Data bit: Parity bit provided (7 data bits plus 1 parity bit = 8 bits)
Parity bit not provided (8 data bits = 8 bits)
- (4) Parity bit: Parity bit not provided.
Even parity: 1 bit
Odd parity: 1 bit
- (5) Stop bits: 1 bits, 1.5 bits, 2 bits
- (6) Sync system: Start-stop sync system



- (7) Communication system: Full duplex communication/
half duplex communication

8-1-2 Setting the serial I/F

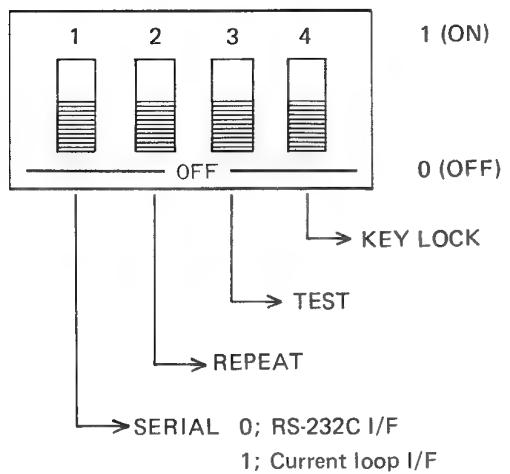
Serial interface selection is made by means of the bottom switch of the Programmer. The switches sets a desired baud rate, and specifies the presence or absence of parity check, a stop bit and X-ON/X-OFF control.



* The above sample setting is such that the transmission rate is 9600 bauds, no parity is provided, 2 stop bits are provided and the transmission procedure is not controlled.

8-1-3 Serial I/F selection

The RS-232C I/F or the current loop I/F may be selected using the bottom switch of the Programmer.



8-2 RS-232C Level Interface

The Programmer is provided with the interface equivalent to the EIA standard RS-232C, type D. The RS-232C is an interface for connection with a standard modem standardized in EIA. (Refer to CCITT recommendation, V. 24.)

8-2-1 Descriptions of interface signals

Name and abbreviation	Symbol	Pin No.	Signal direction	
			Programmer	External equipment
Protective ground (FG)	AA	1		
Transmitted data (TXD)	BA	2	←	
Received data (RXD)	BB	3	→	→
Request to send (RTS)	CA	4	←	
Clear to send (CTS)	CB	5	→	→
Data set ready (DSR)	CC	6	→	→
Signal ground (SG)	AB	7		
Data terminal ready (DTR)	CD	20	←	
Carrier detector (CD)	CE	8	→	→

1. AA; Protective ground
 2. BA; Received data. The Programmer receives data from an external equipment.
 3. BB; Transmitted data. The Programmer transmits data to an external equipment.
 4. CA; Request to send (RTS)

ON: Informs the Programmer that an external equipment is ready for reception.
 OFF: Informs the Programmer that an external equipment is not ready for reception.
 5. CB; Clear to send (CTS)

ON: Informs an external equipment that the Programmer is ready for reception.
 OFF: Informs an external equipment that the Programmer is not ready for reception.
 6. CC; Data set ready (DSR)

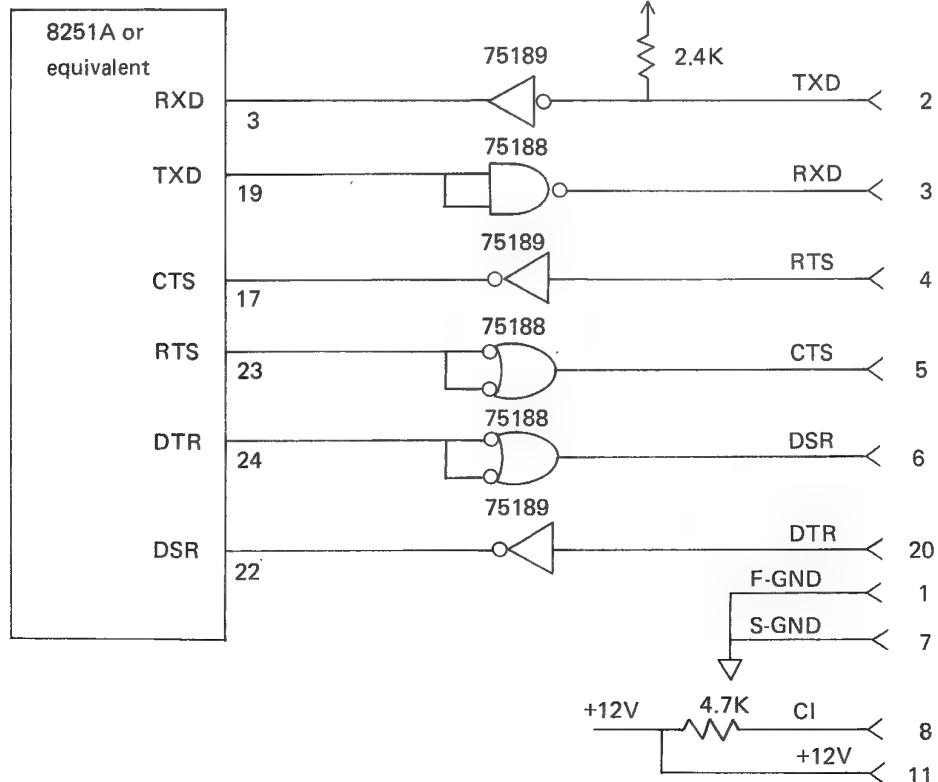
ON: Informs an external equipment that the Programmer is ready for send/receive.
 OFF: Informs an external equipment that the
- Programmer is not ready for send/receive.
7. AB; Signal ground
 20. CD; Data terminal ready (DTR)

ON: Informs the Programmer that an external equipment is ready for send/receive.
 OFF: Informs the Programmer that an external equipment is not ready for send/receive.
 8. CE; Carrier detector (CD)

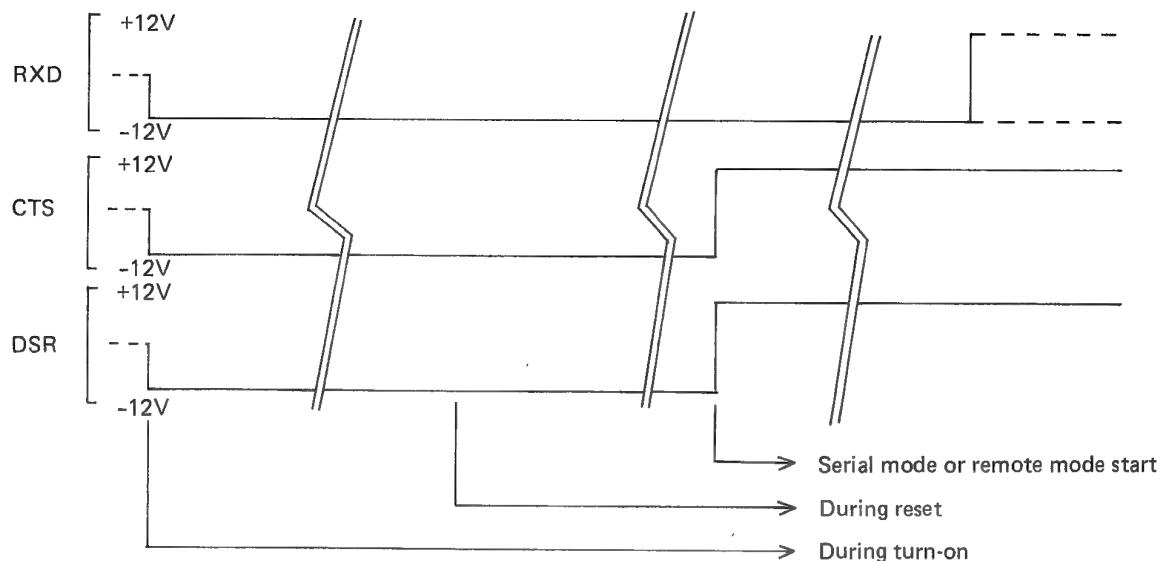
ON: Informs an external equipment that the Programmer has detected a carrier.
 OFF state not provided.
- Polarity of interface signals
- | Voltage | Data signal | Control signal |
|-------------|-------------|----------------|
| +3 to +12 V | 0 | ON |
| -3 to -12 V | 1 | OFF |

8-2-2 Interface signal output

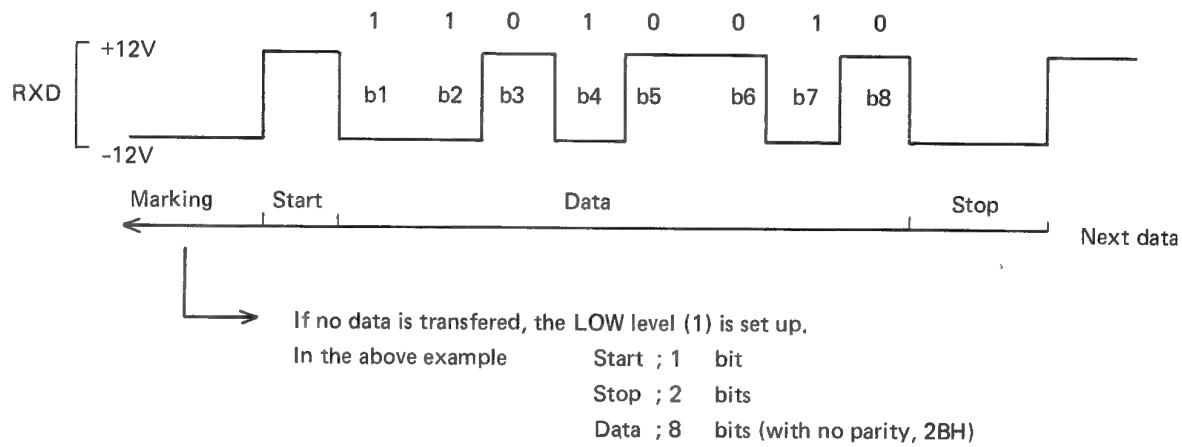
- Signal output



- Signal output level



- Sent data



- Received data

The external equipment, if it is not put in the LOW level (1) with no data transmitted, cannot be connected to the Programmer, (which is referred to marking).

8-2-3 Connection to external equipment

Presented below are descriptions on how to connect two separate devices having the RS-232C serial interface. In general, the devices having the serial interface are broken down into:

- 1) data terminal equipment (DTE), and
- 2) data communication equipment (DCE).

The Programmer is defined as the DCE. Terminals, PTRs and printers are defined as the DTE, while modems are as the DCE.

Paragraphs 8-2-3-1 and 8-2-3-2 show typical sample connections for the Programmer and DTE or DCE equipment. Allowing for a connector signals assignment and data send/receive control, the connection cable should be determined.

8-2-3-1 Connection with data terminal equipment

- Connection with DTE incorporating DTR and RTS

DTE		PROGRAMMER (DCE)	
ABBREVIATION	No.	No.	ABBREVIATION
FG	1	1	FG
TXD	2	2	TXD
RXD	3	3	RXD
RTS	4	4	RTS
CTS	5	5	CTS
DSR	6	6	DSR
SG	7	7	SG
DTR	20	20	DTR

The following are the devices adapted to such connection:

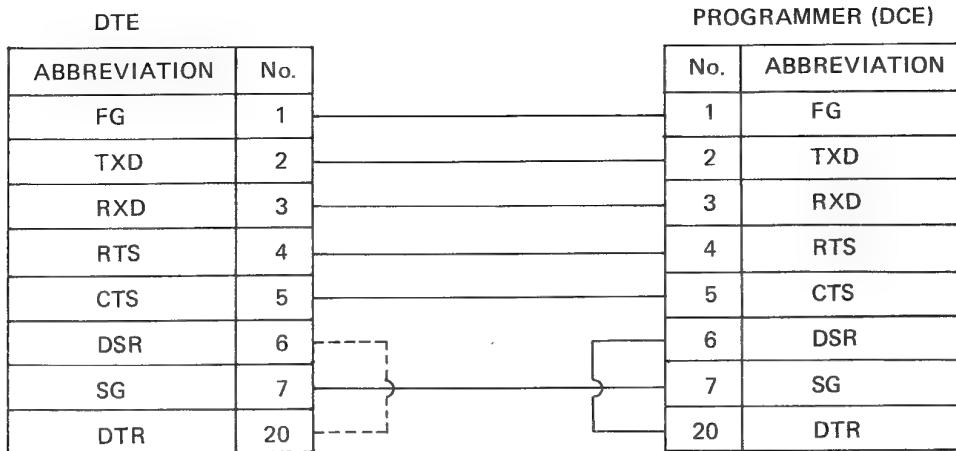
- HP64000 to modem connection
- ADM-3A (CRT terminal)
- MDS (Intel) ○ PC-8800 (NEC)

- Connection with DTE incorporating DTR

DTE		PROGRAMMER (DCE)	
ABBREVIATION	No.	No.	ABBREVIATION
FG	1	1	FG
TXD	2	2	TXD
RXD	3	3	RXD
RTS	4	4	RTS
CTS	5	5	CTS
DSR	6	6	DSR
SG	7	7	SG
DTR	20	20	DTR

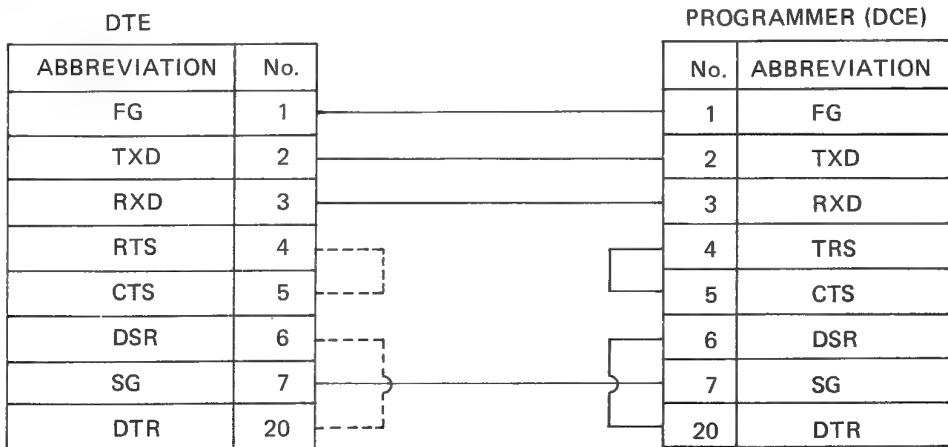
Note) When shorting Nos. 4 and 5 of the DTE, refer to
respective operations manual.

- Connection with DTE incorporating RTS



Note) When short-circuiting Nos. 6 and 20 of the DTE, refer to respective operations manuals.

- Connection with DTE using no control signal



Note) When shorting Nos. 4 and 5 or 6 and 20 of the DTE, refer to respective operations manuals.

- Connection with Casio typewriter (Model 750)

DTE		PROGRAMMER (DCE)	
ABBREVIATION	No.	No.	ABBREVIATION
FG	1	1	FG
TXD	2	2	TXD
RXD	3	3	RXD
RTS	4	4	RTS
CTS	5	5	CTS
DSR	6	6	DSR
SG	7	7	SG
DTR	20	20	DTR
CD	8	8	CD

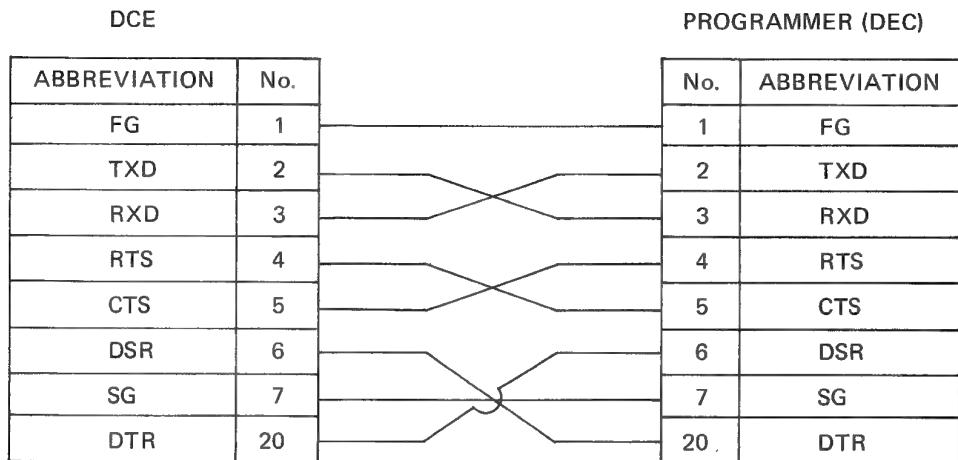
Note) With the PTP used, the transmission rate must not be more than 300 bauds. With the PTP not used, the transmission rate must not be more than 1200 bauds; otherwise, the typewriter must be modified. Refer to the operations manual for the typewriter.

- Connection with TOSHIBA personal computer (PASOPIA)

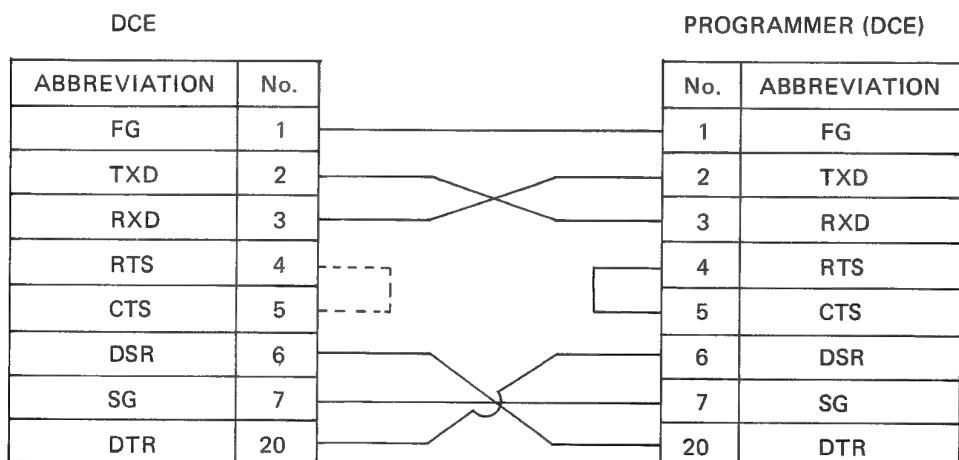
DTE		PROGRAMMER (DCE)	
ABBREVIATION	No.	No.	ABBREVIATION
FG(NC)	1	1	FG
TXD	2	2	TXD
RXD	3	3	RXD
RTS	4	4	RTS
CTS	5	5	CTS
DSR	6	6	DSR
SG(GND)	7	7	SG
DCD	8	20	DTR
NC	9		
RXC	10		
NC	11		
DTR	12		
NC	13		
CI	14		
TXC	15		
STI	16		

8-2-3-2 Connection with data communication equipment (DCE)

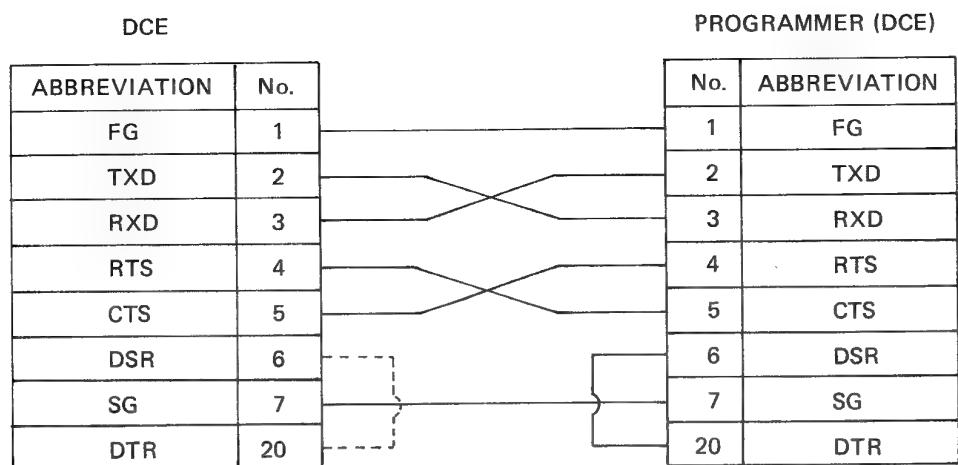
- Connection with DCE incorporating DTR and RTS



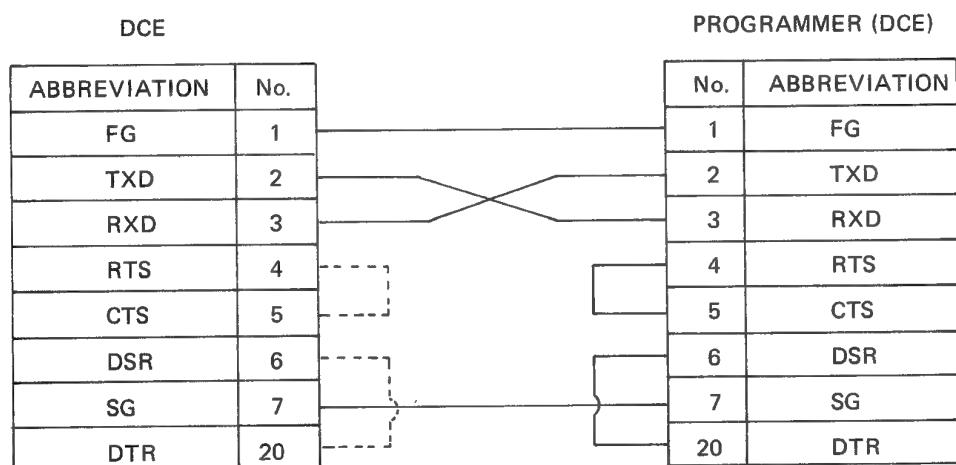
- Connection with DCE incorporating DTR



- Connection with DCE incorporating RTS



- Connection with equipment using no control signal



8-3 Current Loop Interface

The current loop used in the Programmer is of the 20mA full duplex, and the power supply is provided in it.

Signal name and abbreviation	Pin No.	Signal direction	
		Programmer	External equipment
Data in + (DIN+)	10	→	
Data in - (DIN-)	11	←	
Data out + (DOUT+)	12	→	
Data out - (DOUT-)	13	←	
Reader control + (READER+)	14	→	
Reader control - (READER-)	15	←	
Ground	1, 7		

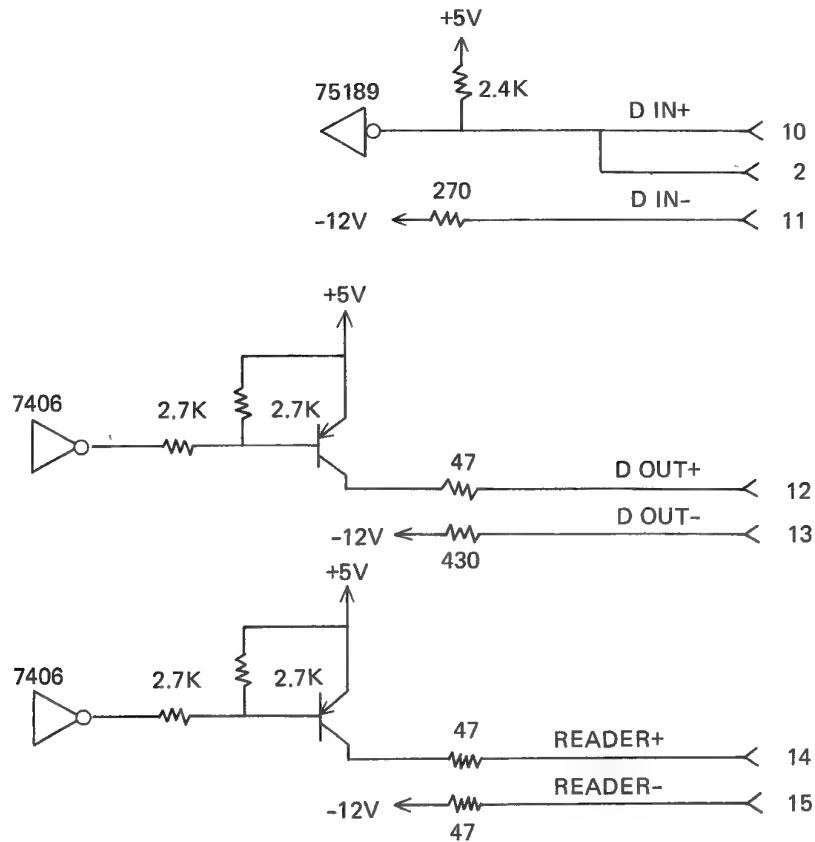
- 10. DIN+;
- 11. DIN-; The Programmer receives data from an external equipment.
- 12. DOUT+;
DOUT-; The Programmer sends data to an external equipment.
- 14. READER+;
- 15. READER-; ON: PTR drives an external equipment.
OFF: PTR stops an external equipment.

- Polarity of interface signals

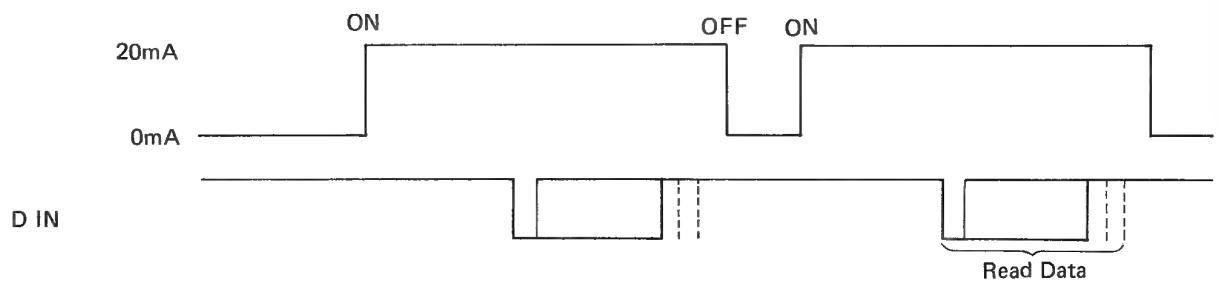
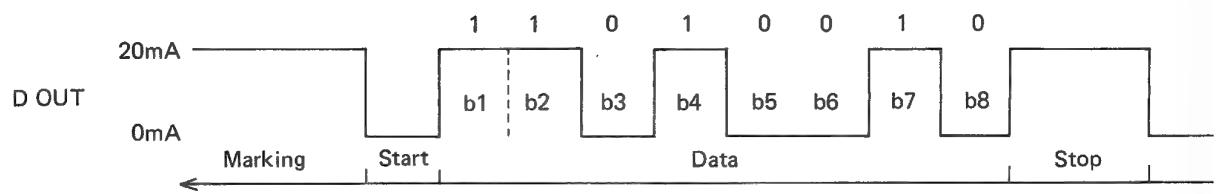
Drive circuit	Data signla	Control signal
ON	1	ON
OFF	0	OFF

8-3-2 Current loop interface signal output

- Signal output



- Output signal levels



8-3-3 Example of connection with external equipment

- Connection with CASIO typewriter (Model 750)

Typewriter		Programmer (Currents)	
Signal Name	No.	No.	Signal Name
Data Out +	1	10	D IN +
Return -	2	11	D IN -
Data In +	3	12	D OUT +
Return -	4	13	D OUT -
Reader Start +	5	14	READER +
Return -	6	15	READER -
Remote Select +	9	4	RTS
Return -	10	5	CTS
Data Bus +	13	6	DSR
Return -	14	20	DTR

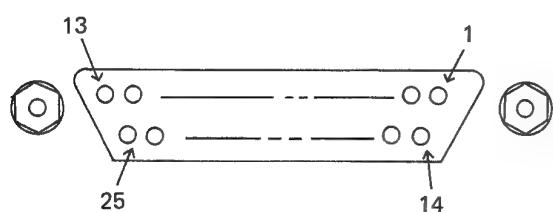
RS-232C
Control
Signal

Note) The transmission rate must not be more than 1200 bauds.

8-4 Serial I/F Cable

8-4-1 Serial I/F connector

This is a connector covering 25 DSUB pins, and is used along with the RS-232C and the current loop.



8-4-2 Table of serial I/F connector pins

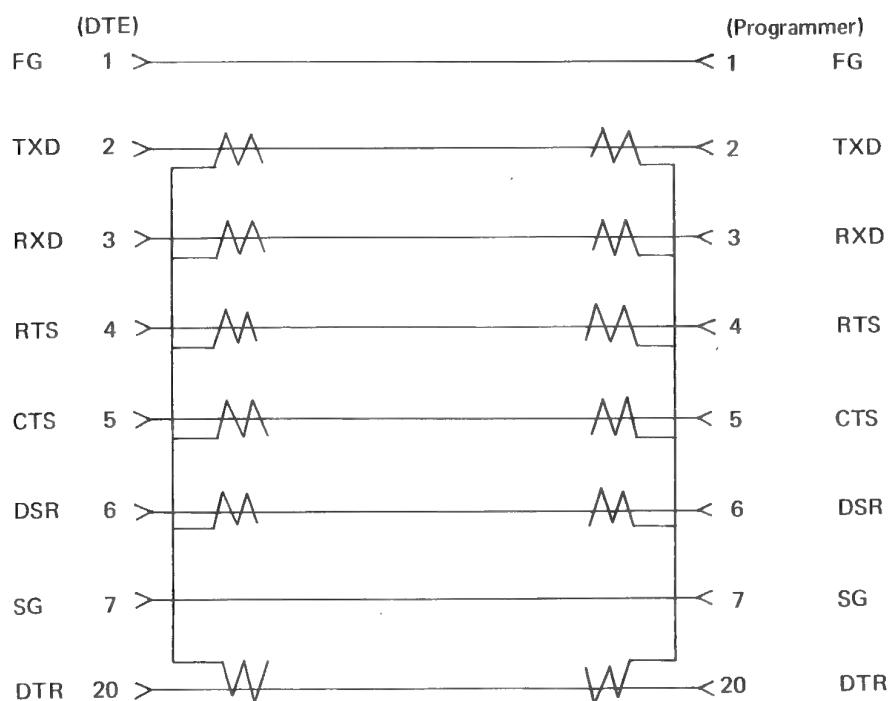
Pin No.	Current loop I/F	RS-232C I/F
1	Ground	Protective ground
2		Transmitted data
3		Received data
4		Request to send
5		Clear to send
6		Data set ready
7	Ground	Signal ground
8		Carrier detector
9		
10	Data in +	
11	Data in -	
12	Data out +	
13	Data out -	
14	Reader control +	
15	Reader control -	
16		
17		
18		
19		
20		Data terminal ready
21		
22		
23		
24		
25	+12V	

8-4-3 Cable

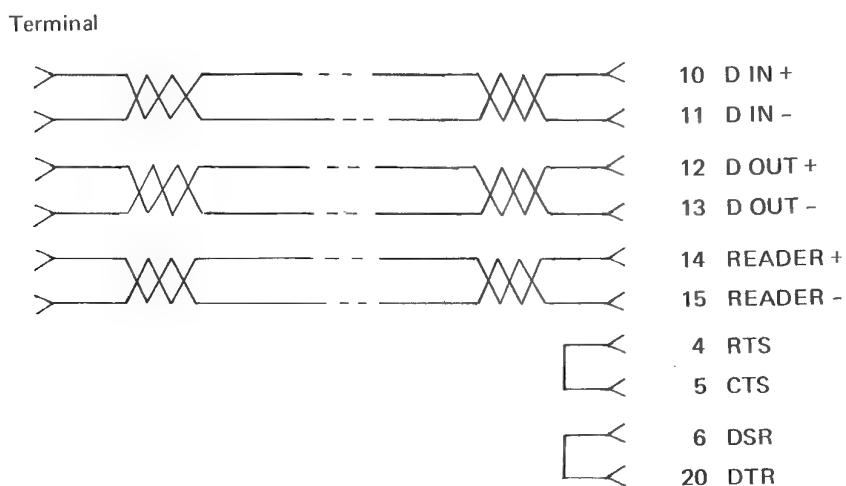
It is recommended that the RS-232C IF or current loop I/F

cable be used in the following connection:

① RS-232C I/F



② Current loop I/F

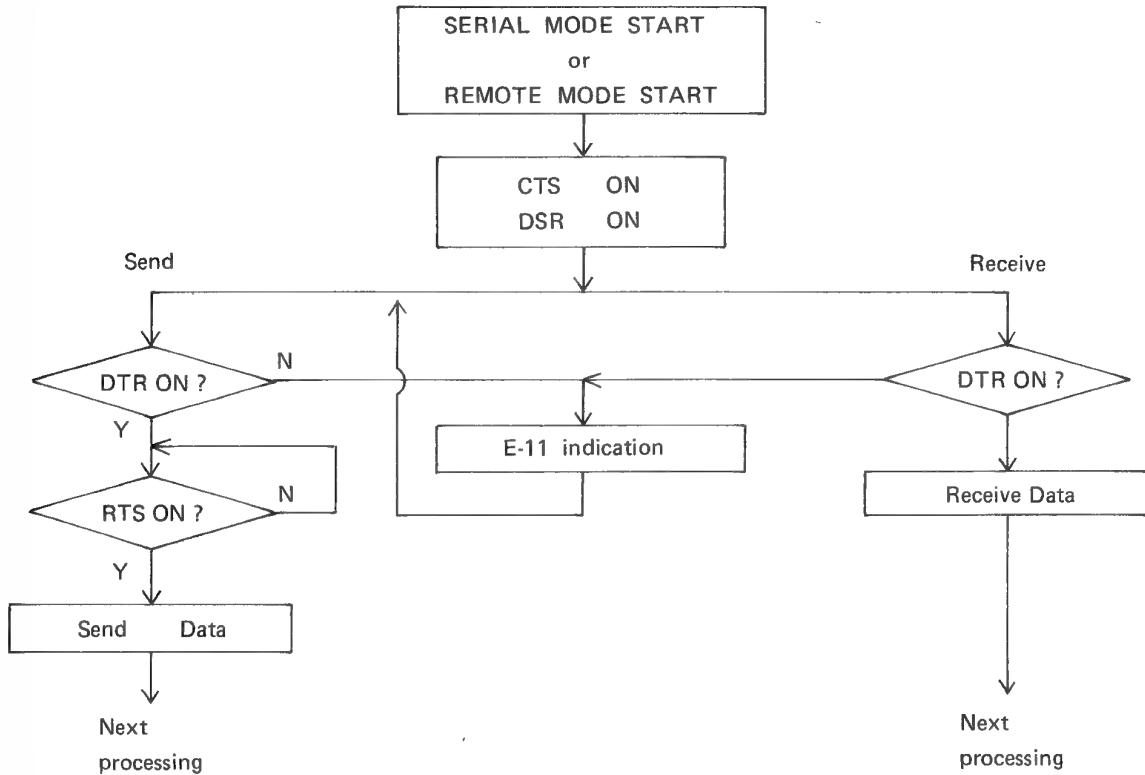


8-5 Control Method and Transmission Protocol

In the RS-232C level interface, the data send/receive control method and the transmission protocol are not standardized. Various control methods and transmission protocols are

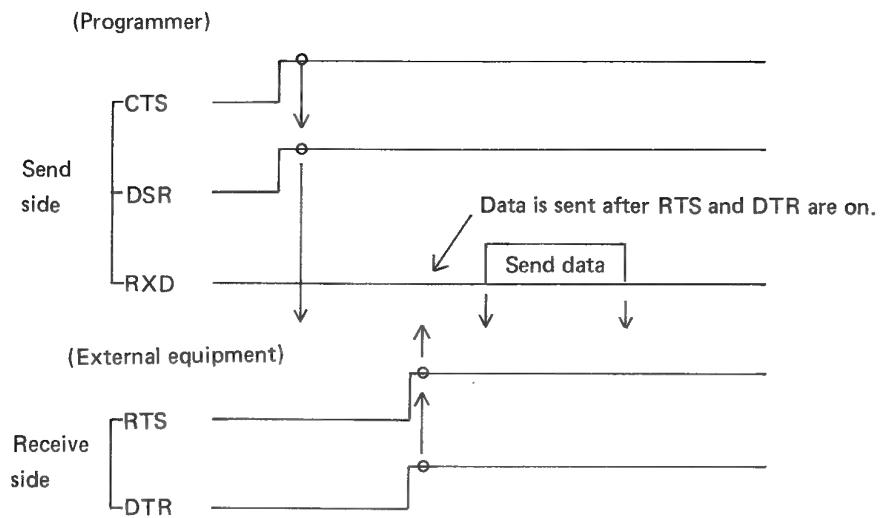
used depending on equipment. Presented below are the control method and the transmission protocol used in the Programmer.

8-5-1 Control method

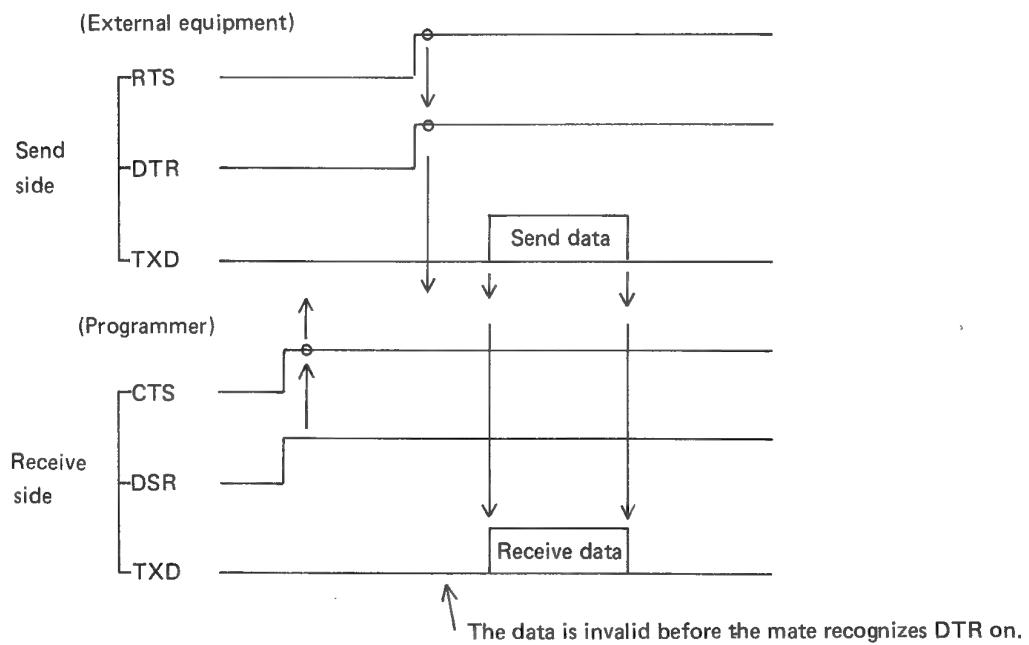


Notes: The RS-232C I/F for the Programmer uses communication interface ICs equivalent to 8251A. Data may be sent only with the RTS signal on; (otherwise, the 8251A provides hardware inhibition.) Both send and receive are enabled by the turned on DTR signal. If this signal is off, "E-11" is displayed on the LED panel, and the Programmer waits for the DTR to be turned on.

- Send (sample signals)



- Receive (sample signals)



8-5-2 Transmission protocol

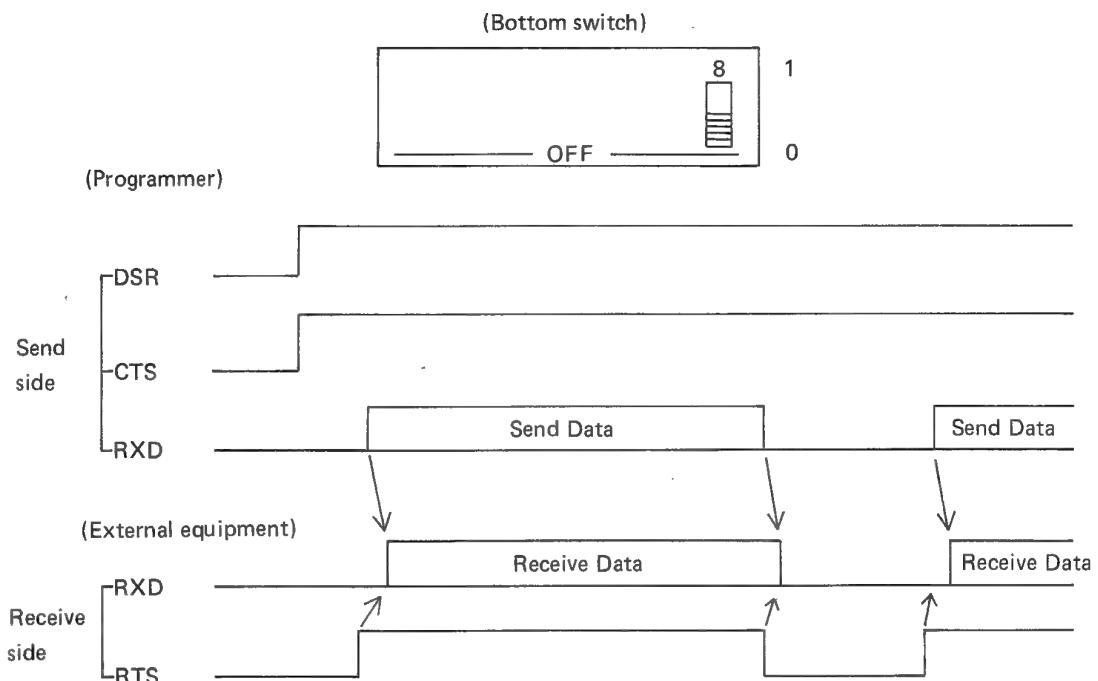
The Programmer offers the following two transmission protocols:

- 1) Non-control
- 2) X-ON/X-OFF control

256 bytes of the receive buffer are provided.

8-5-2-1 Non-control

The transmission protocol gets free from control by placing X-ON/X-OFF (bottom switch) in the "0" (OFF) position. However, control by DTR and RTS signals described in 8-5-1 is available.

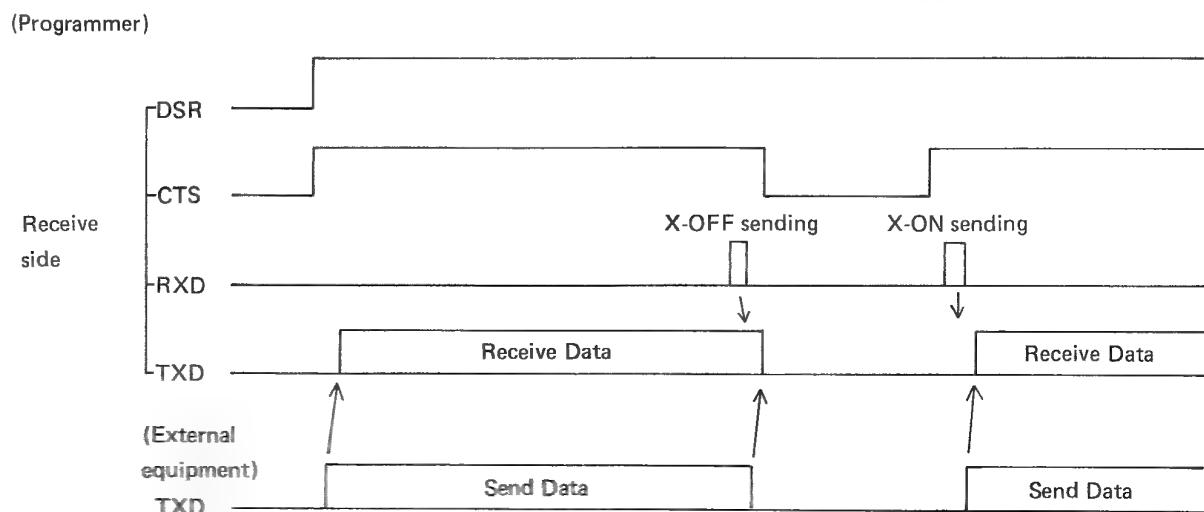


8-5-2-2 X-ON/X-OFF control

a) Remote mode, receive

While the Programmer is receiving data from an external equipment if data arrives at the receive buffer which exceeds 2/3 of its capacity, the code "X-OFF (13H)"

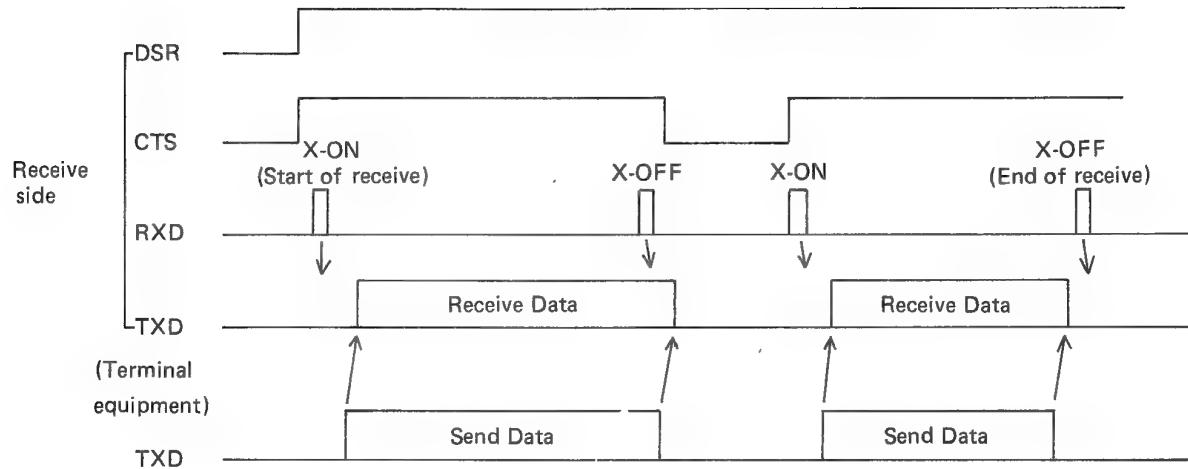
is sent to an external equipment, and then, the CTS signal is turned off. When the data becomes less than 1/3 of the buffer's data capacity, the Programmer turns on the CTS signal, and then sends the code X-ON (11H) to an external equipment. (Full duplex mode)



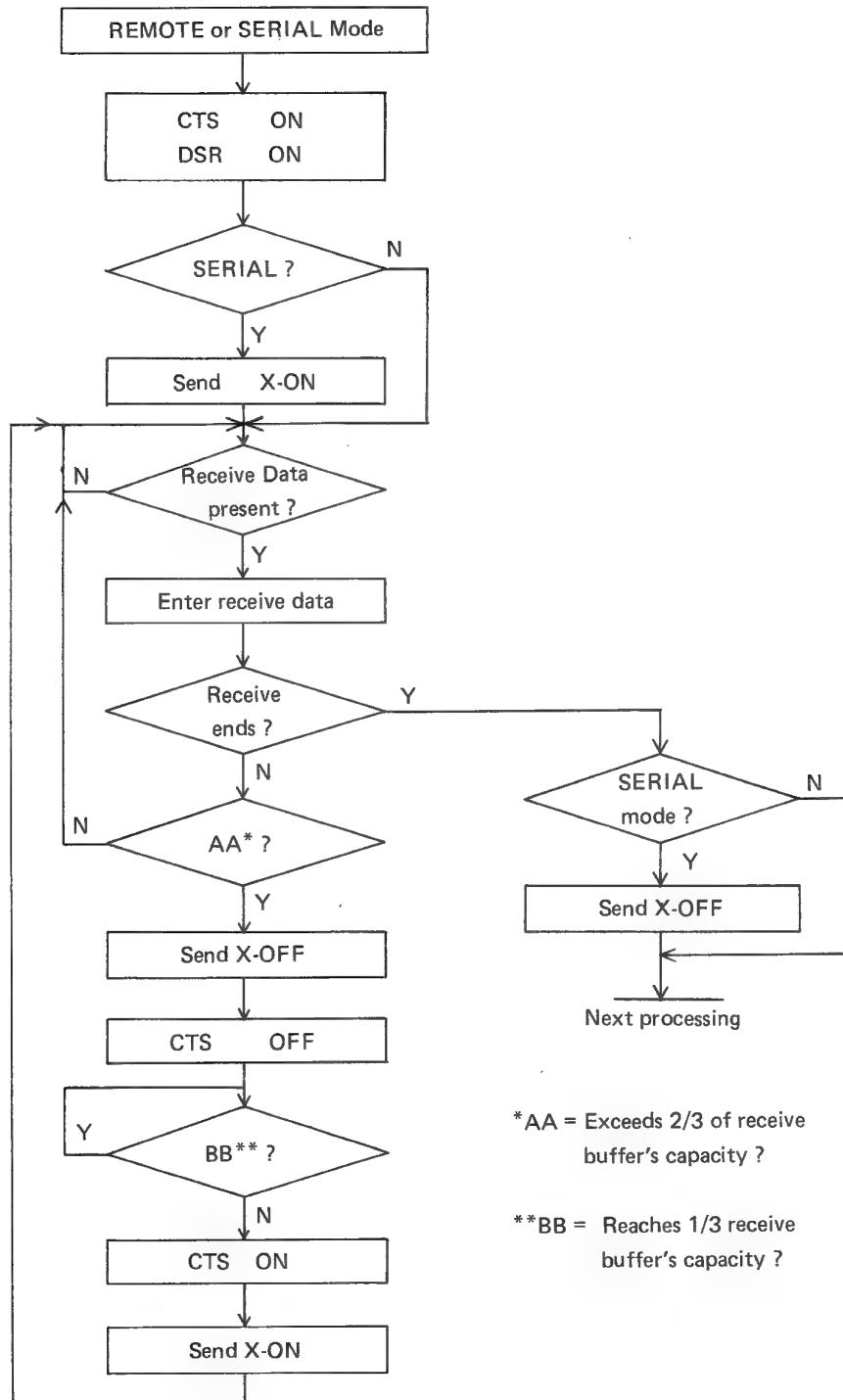
b) Serial mode, receive

Appended to the remote mode (receive) function. At the start of the serial mode, the "X-ON" code is sent, and the "X-OFF" code is sent on the termination of receive of all data.

(Programmer)

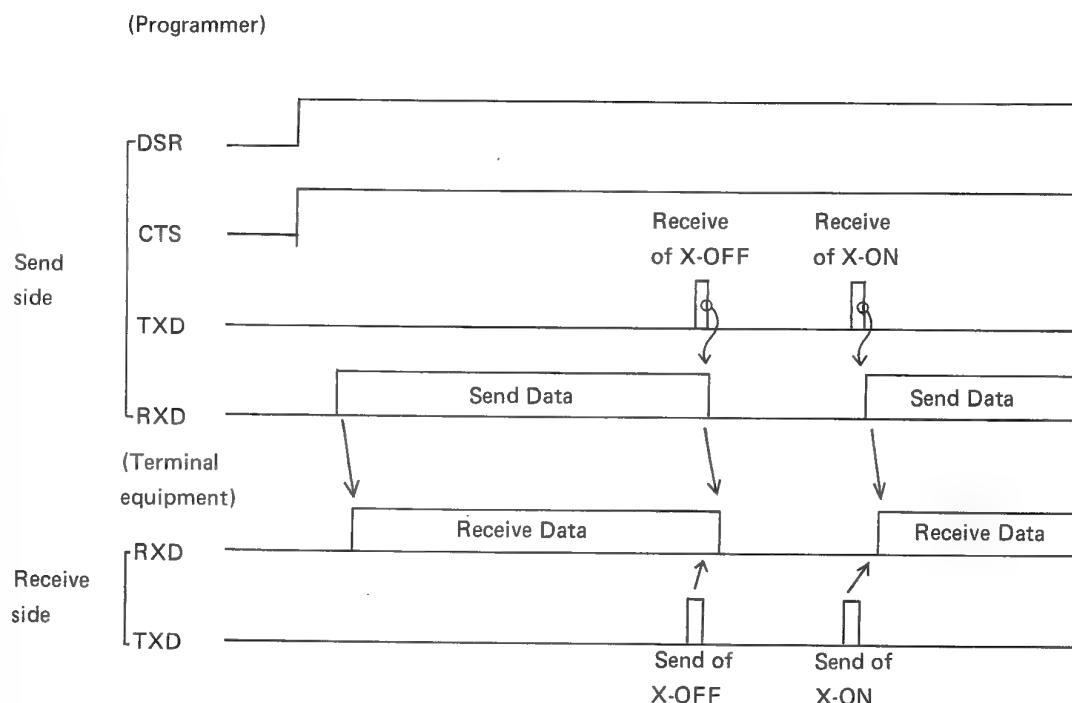


c) X-ON/X-OFF control receive flow chart

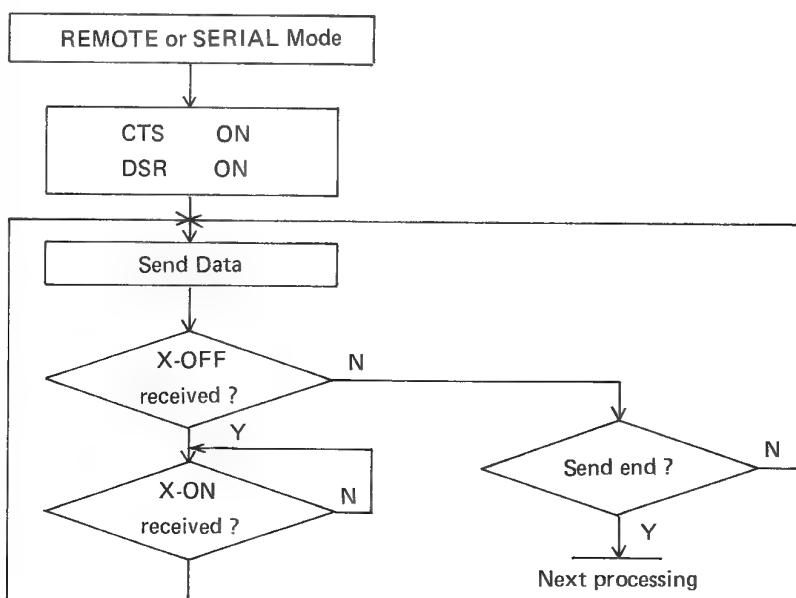


d) Remote or serial mode, receive

While the Programmer sends data to an external equipment if it receives the "X-OFF" from the external equipment, data send is temporarily stopped. When the "X-ON" code is received, data send is restarted. For control from an external equipment using the RTS signal, see 5-2-1. (Full duplex mode)



e) X-ON/X-OFF control flow chart



○ Data insert instruction "I"

#I X X X X CR LF

5 2 3 Up to 16 bytes
X X X X 0 0 1 1 FF CR LF

XX Y X CR CR LF #
4

1. Insert start address

2. Data to be inserted

3. Data separate mark

4. End code

5. Insert address

○ CONT instructions "OT" and "OTn"

#OT CR Normal CR LF #
Abnormal CR LF ? CR LF #

"OTn" block CONT instruction

#OT n CR Same as the OT instruction.

○ Buffer check sum instruction "BO"

Sample output of a check sum for a range specified using PAE

#B O CR

882C 4E

Exclusive OR
Addition

#B O CR LF 882C 4E CR LF #

○ PROM device select instruction "N"

Selected by entering the number assigned to a P-ROM.
(See a command number.)

#N 00 - CR

If only CR has been entered, the device is not changed.
The current device number is displayed.

#N 00 - C CR

Sample 27128 high-speed programming specification.
The current device number is displayed.

For the device number, two entered lower digits are valid.

If only one digit has been entered, the second digit is assumed to be "0".

#N 0 0 - nn CR Normal CR LF #
Abnormal CR LF ? CR LF #

○ Address select instruction "A"

This selects the all address (N) mode, the even (E) mode or the odd (O) mode as the operation address mode.

#A O CR

If only CR has been entered, the address mode is not changed.

The current address mode is displayed:

N: All address mode

E: Even address mode

O: Odd address mode

#A N E CR

Specifies the even address mode.

At present, the all address mode is established.

#A N E CR CR LF #

○ In the address mode, only "N", "O" and "E" may be entered as input data. The other data is considered to be an error.

#A N A CR LF ? CR LF #

○ EEPROM chip erase instruction "Z"

Chip-erases an EEPROM. For the EPROM, the instruction is not accepted.

#Z CR

#Z CR Normal CR LF #
Abnormal CR LF ? CR LF #

○ EEPROM byte erase instruction "ZB"

Byte-erases an EEPROM. For any device which cannot be byte-erased, the instruction is not accepted.

#Z B CR

AD RS - 10 CR

Address to be erased.

#Z B CR LF AD RS - 10 CR Normal CR LF #

Abnormal CR LF ?

CR LF #

8-7 Instructions in Serial Mode

The serial instruction for the Programmer permits exchanging data with an external equipment through key operations. Data exchange is activated by a specified data format. (Note 1)

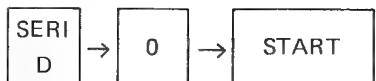
The serial instruction offers four functions.

- Receives a data format sent from an external equipment.
- Outputs a data format from the Programmer to an external equipment.
- Outputs a list from the Programmer to an external equipment.
- Outputs a control code from the Programmer to an external equipment.

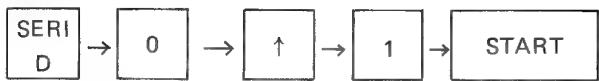
Note 1: See 9-8, Data Format. The format is specified by "EDIT", "FORMAT" and "FORMAT No." keys.

8-7-1 Data format reception

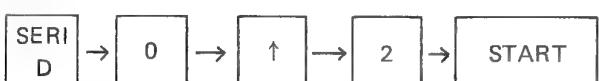
- All data reception



- Reception in the range of 0 to 32 k-bytes

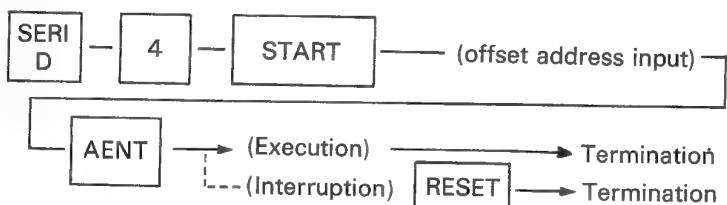


Reception in the range of 32 k to 64 k-bytes

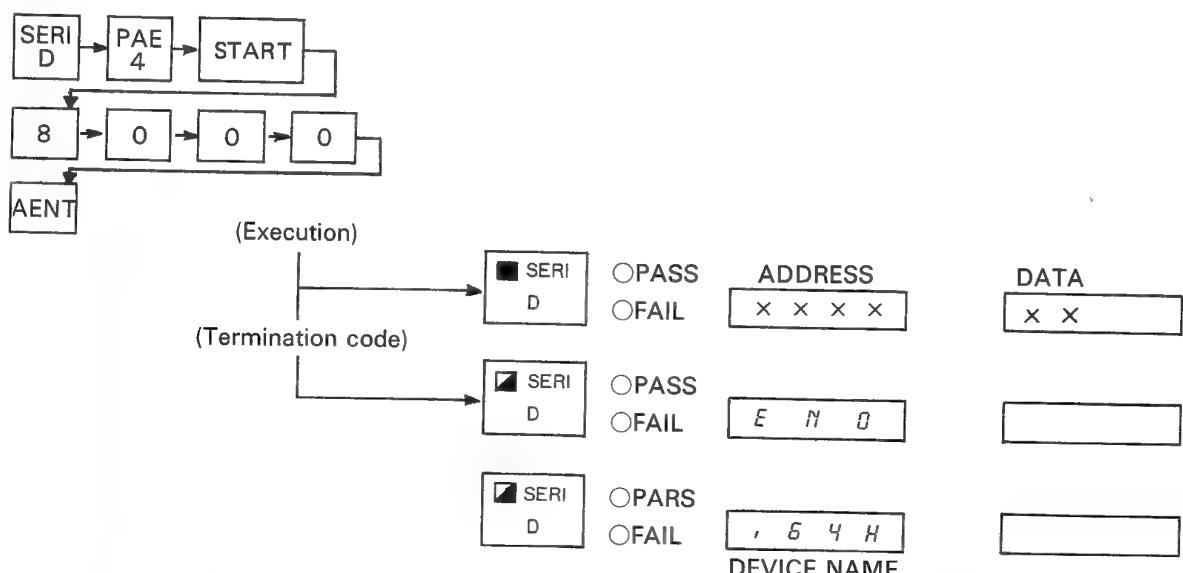


Note 1: If no data sent using this instruction after the Programmer waits for approx. 120 seconds, "E-17" is displayed on the display address section, resulting in time out. "?" is output to an external equipment, and the Programmer waits for a command in the same mode.

- Reception with Offset command



- Reception with Offset command (from 8000)



* Transfer command in CP/M machines where RS-232C is defined as "TTY" (machines with echoback)

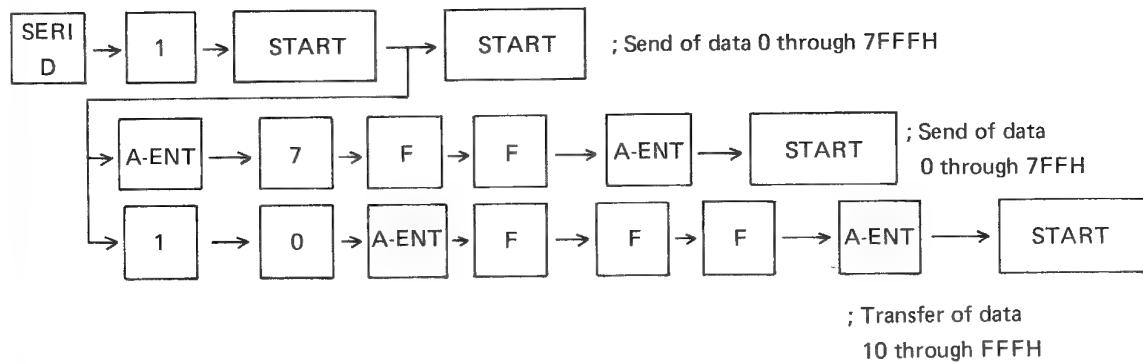
Format output (up to 9600 baud)



List output (up to 9600 baud)



8-7-2 Data format output



8-7-3 List output

Output in the same format as the REMOTE "L" instruction.



8-7-4 Control code output



Key No.	Control code	Remarks
0	SOH (01H)	
1	STX (02H)	
2	ETX (03H)	
3	EOT (04H)	
4	ENQ (05H)	
5	DC1 (11H)	X-ON
6	DC2 (12H)	
7	DC3 (13H)	X-OFF
8	DC4 (14H)	
9	ETB (17H)	
A	EM (19H)	
B	SUB (1AH)	INTEL MDS terminate code, CP/M terminate code
C	ESC (1BH)	

8-7-5 Error

The following are the errors in use of the serial interface:

E-2 — Parity error

With even or odd parity set, any detected parity error is displayed. The serial mode is set again.

E-3 — Format error

Data format errors have been detected.

E-7 — Load verify error

This error occurs in the RLV instruction. The buffer differs from the data format.

E-8 — Instruction specify error

E-10 — RS-232C buffer over error

E-11 — RS-232C RTS undetected error

Note: This is not possible to send data from 1866A to external equipment in remote-mode.

The data as follows:

- A) DC1 (Hex. 11) and DC3 (Hex. 13) in RL instruction.
- B) DC2 (Hex. 12) and DC4 (Hex. 14) in P instruction.

Presented below is the procedure.

- (1) Press the "REM" switch.
- (2) Press the "0" switch.
- (3) Press the "START" switch.

8-8- Data Format

The following 14 formats are provided as the standards.

Key number	Format
0	Tektronix HEX (16 bits)
1	Tektronix HEX (8 bits)
2	Intel HEX (16 bits), only for receive. Intel HEX (8 bits)
3	Motorola exermax (16 bits), only for receive. Motorola exerciser (8 bits)
4	Texas SDSMAC
5	Minato HEX
6	ASCII-HEX (SPACE). V1 ASCII-HEX (SPACE). V2, only for receive. ASCII-HEX (SPACE). V3, only for receive. ASCII-HEX (SPACE). V4, only for receive.
7	Binary
8	Hewlett Packard 64000 absolute

Note: Data is read into the addresses from the address field for a format on. If there is no address in the format, data is read into the addresses from the address 0 on.

8-8-1 Output format

Presented below are sample data formats output from the Programmer. The data is "11H" bytes from "0000H" to "0010H". "0" corresponds to the address "0", "1" to the address "1", and "10" to the address "10", and so on.

1. TEKTRONIX HEX. (16 bits)

```
NUL 100 byte (Note 1)
%2A68F400000102030405060708090A0B0C0D0E0F10
%0C6194001011
%0A81640000
NUL 100 byte (Note 1)
```

2. TEKTRONIX HEX. (8 bits)

```
NUL 100 byte
/000010010102030405060708090A0B0C0D0E0F1079
/001001021102
/00000000
NUL 100 byte
```

3. INTEL HEX. (8 bits)

```
NUL 100 byte
:100000000102030405060708090A0B0C0D0E0F1068
:0100100011DE
:00000001FF
NUL 100 byte
```

4. MOTOROLA EXERCISER

```
NUL 100 byte
S00900004D494E41544F2E
S11300000102030405060708090A0B0C0D0E0F1064
S104001011DA
S9030000FC
NUL 100 byte
```

5. TEXAS INSTRUMENT SDSMAC

```
NUL 100 byte
00000MINATO 7FCDF
90000B0102B0304B0506B0708B090AB0B0CB0D0EB
0F107F61DF
90010*117FE43F
:
NUL 100 byte
```

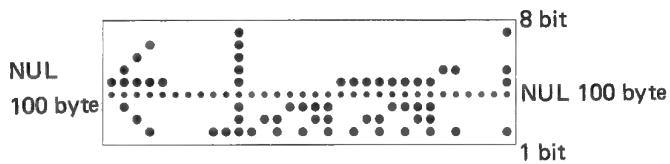
6. MINATO HEX.

```
NUL 100 byte
[#0000 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
10 #0010 11]
NUL 100 byte
```

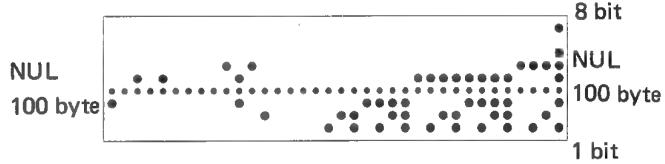
7. ASCII HEX. V1

```
NUL 100 byte
$A0000,
01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10
$A0010,
11
$S0099,
NUL 100 byte
```

8. BINARY



9. HP-64000 ABSOLUTE



Note 1: "NUL 100 byte" before and after the data format are output only in the remote mode.

8-8-2 Formats description

1. TEKTRONIX HEX. (16 bits)

a) Data record

%	LL	6	SS	4AAAAA	DDDDDDDD	----	DD
	1	2	3	4	5		6

(1) Record mark ASCII "%"

Indicates the start of a record.

(2) Record digit count ASCII

Indicates in HEX the number of ASCII codes from record digit counts "2" to "6" output in a two-digit ASCII code.

(3) Data record format ASCII "6"

Indicates a data record.

(4) Check sum ASCII

Indicates in HEX the ASCII codes for (2) record digit count, (3) record format, (5) address and (6) data, and in a two-digit ASCII code 8 bits of their addition.

(5) Address, ASCII

The first ACSII code indicates an address output digit count, and the address is output in the ASCII code by a desired digit count.

(6) Data, ASCII

One byte of data is indicated in a two-digit ASCII code and output.

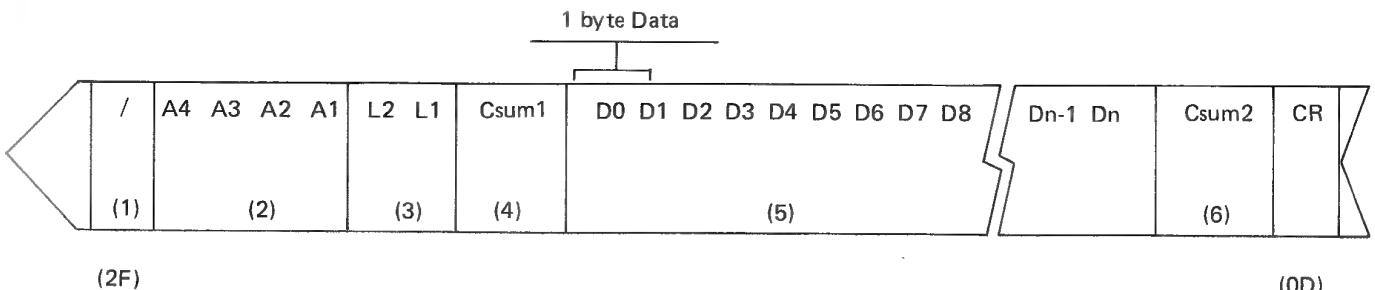
b) End record

% LL 8 SS 4AAAAA
1 2 7 5 6

(7) End record format ASCII "8"

Indicates an end record.

2. TEKTRONIX HEX. (8 bits)



(1) Record mark ASCII "/" HEX "2F"

Indicates the start of a record.

(2) Start address ASCII

Indicates the start address of data to be entered. The address is expressed in 16 bits and is output in a four-digit ASCII code. A4 makes up the most significant digit.

A4	A3	A2	A1
15	12	11	8 7 4 3 0

(3) Record digit count, ASCII

Indicates the data digit count (5). The digit count is expressed in 8 bits and output is a two-digit ASCII code. L2 represents the most significant digit. "00" through "FF" may be specified. "00" signifies the end of data.

Record digit count "00"	End of data
"01"	Data digit count 1
"FF"	Data digit count 255

(4) Header check sum, ASCII

The ASCII data from A4 (2) to L1 (3) is expressed in HEX and 8 bits of addition are expressed in a two-digit ASCII code for output.

(5) Data, ASCII

Indicates one byte of data in a two-digit ASCII code. A data amount specified in data digit count (3) is output. The first data represents the most significant digit.

(6) Data check sum, ASCII

This is a check sum for D0 thru Dn of (5). Operated in the same manner as the header check sum (4).

Codes used: ASCII, 7 bits and 8 bits

Numeric characters: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Alphabetic characters: A, B, C, D, E, F

Symbol: /

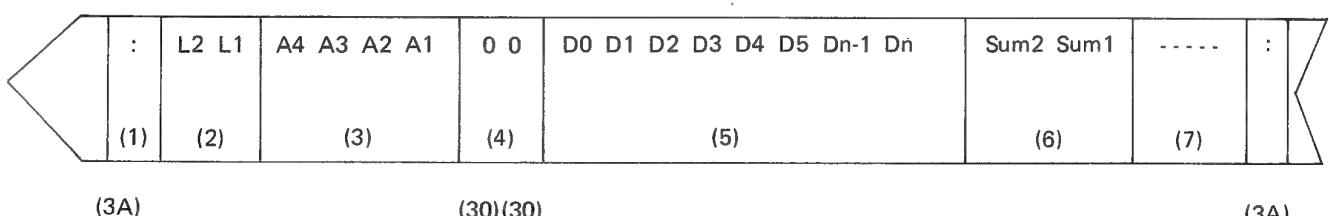
The other codes such as DEL, NUL, CR and LF are ignored.

Note: Other than "DEL" and "NUL" must not be entered between (1) and (6).

3. INTEL HEX

3-1 INTEL HEX (8/16 bits)

a) Data record



(1) Record mark

Indicates the start of a record. ":" in the ASCII code and "3A" in the hexadecimal number.

(2) Record digit count

Indicates a data byte count in a two-digit ASCII code. The first data indicates the most significant digit. "0" thru "255" may be specified as the capacity. "00" thru

"FF" in the ASCII code, and "3030" thru "4646" in the hexadecimal number.

(3) Start address

Indicates the start address of data to be entered. The data is expressed in a four-digit ASCII code. The first data signifies the most significant digit.

(4) Record type

Indicates the types of records. The data record is "00" in ASCII and "3030" in the hexadecimal indication.

(5) Data

Indicates 1 bytes (8 bits) of data in two bytes of the ASCII codes. The first data indicates the most significant digit.

(6) Check sum

This indicates a check sum for the "record digit count" (2) to the end byte of "data" (5). The calculation method is such that data expressed in two bytes of the ASCII code is converted in 1 byte of the hexadecimal number and byte data is added thereto. And the resulting complement is given in two bytes of the ASCII code. The first data is the most significant digit.

Example:

:	0 8	0 1	0 0	0 0	1 2	3 4	A	B	C	D	C 8	--	
(3A)	(30) (38)	(30) (31)	(30) (30)	(30) (30)	(31) (32)	(33) (34)	(41) (42)	(43) (44)	(43) (44)	(43) (44)	(38)		

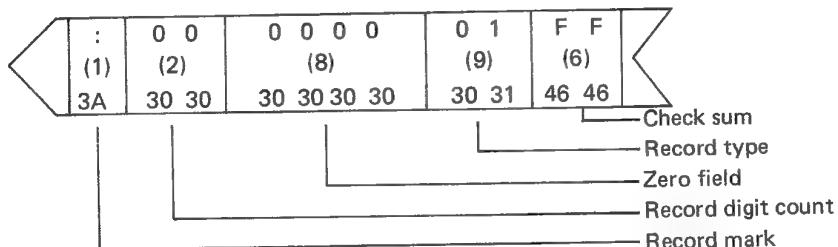
Address: 0100 0101 0102 0103

Data : 12 34 AB CD

(Tape for 4 bytes of data from the address "0100")

(7) The data preceding the record mark is ignored. (CR, LF, etc.)

b) End record



(8) Zero field

The contents of the zero field are "0000" in the ASCII code and "30303030" in the hexadecimal number.

(9) Record type

The end record is "01" in the ASCII code and "3031" in the hexadecimal number.

Data used

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

NUL (00 in the hexadecimal number) and DEL (7F) are ignored.

Absolute address = [extended record address] + [data record address]

: 02 0000 02 AAAA SS
1 2 3 4 5 6

(1) Record mark: ASCII ":"

Indicates the start of a record.

(2) Record digit count: ASCII "02"

The record digit count is fixed at "02".

(3) Start address: ASCII "0000"

Fixed at "0000".

(4) Extended record format: ASCII "02"

Indicates an extended record.

(5) Base address: ASCII

Used along with a data record address. Indicates an absolute address.

(6) Check sum: ASCII

This indicates a check sum from the record digit count (2) to the address (5). Data expressed in two bytes of the ASCII code is converted to 1 byte of the hexadecimal number for addition. The resulting 2's complement is given in two digits of the ASCII code.

3-2 INTEL HEX

a) Data record

Same as the 8 bit format.

b) End record

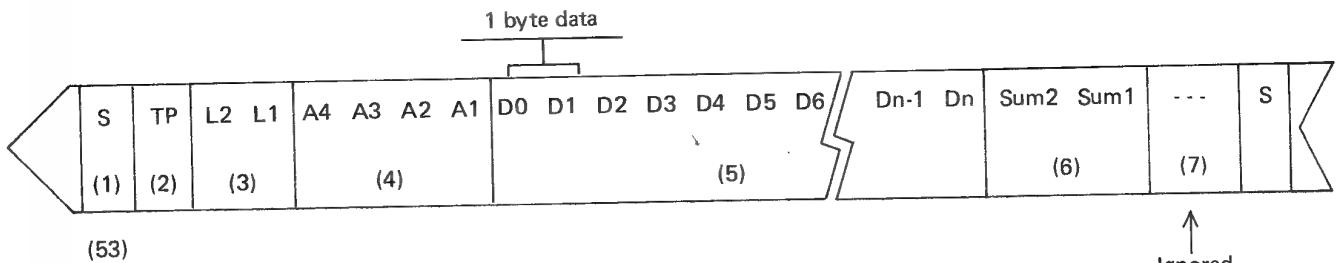
Same as the 8 bit format.

c) Extended record

The data record address shall be regarded as a relative address, while its addition with the base address given in an extended record shall be regarded as an absolute address.

4. Motorola

4-1 Motorola Exerciser



(1) Record mark: ASCII "S", HEX "53"

Indicates the start of a record.

(2) Record type: ASCII "0", "1", "9", HEX "30", "31", "39" Indicates a record type.

0: Header record

1: Data record

9: End record

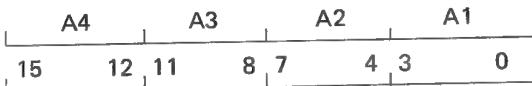
(3) Record digit count: ASCII

Indicates a byte data count from "A4" (4) to "SM1"

(6). Data is expressed in 8 bits, and is output in two digits of the ASCII code.

(4) Start address: ASCII

Indicates the start address of data to be entered. The address is expressed in 16 bits, and is output in four digits of the ASCII code. A4 indicates the most significant digit.



(5) Data: ASCII

Indicates one byte of data in two digits of the ASCII code. The first data indicates the most significant digit.

For a data count, see "record digit count" (3).

(6) Check sum: ASCII

This indicates a check sum from "L2" (3) to "Dn" (5).

4-2 Motorola Exermax (16 bits)

a) Header record

Same as the exerciser (8 bits).

b) Data record, 16 bit address range

Same as the exerciser.

c) End record, 16 bit address range

Same as the exerciser.

d) Data record, 24 bit address range



(1) Record mark: ASCII "S"

Indicates the start of a record.

(2) Record type: ASCII "2"

Indicates the address range of 24 bits of a data record.

(3) Record digit count: ASCII

Indicates a byte data count from the address (4) to the check sum (6).

(4) Start address: ASCII

Indicates the start address of data to be entered.

Gives 24 bits of address data in six digits of the ASCII code.

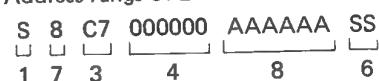
(5) Data

Indicates one byte of data in two digits of the ASCII code.

(6) Check sum

Indicates a check sum from the record digit count (3) to data (5). Data expressed in two byte of ASCII code is converted to one byte of the hexadecimal number for addition. The resulting complement is given in two digits of the ASCII code.

e) Address range of 24 bits of end record



(7) Record mark: ASCII "7"

Indicates the address range of 24 bits of an end record.

(8) File start address: ASCII

Indicates the start address of an absolute file, in six digits of the ASCII code.

5. Texas SDSMAC

The following tags are appended at the start, followed by data.

a) Header

O or K tag

This is located at the start of an object, followed by two fields.

1st field : 4 bytes. Total number of relative codes.

2nd field: 8 bytes. Program name

b) Data

B tag

The subsequent 4-digit ASCII code indicates two bytes of data.

* tag

The subsequent two-digit ASCII code indicates one byte of data.

c) Check sum

Tag 7

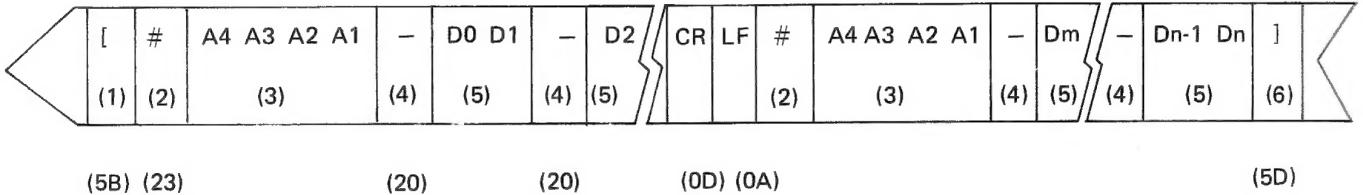
The subsequent four-digit ASCII code indicates a check sum (two byte data).

F tag

Indicates the end mark of a check sum.

The check sum is calculated by totaling characters (hexadecimal addition of ASCII codes) for the start of the record to the tag 7 and by taking the 2's complement of the sum.

6. Minato HEX



(1) Start mark: ASCII "[" , HEX "SB".

Indicates the start of a tape.

(2) Address mark: ASCII "#", HEX "23"

Indicates the start of an address.

(3) Start address: ASCII

Indicates the start address of data to be entered. The address is given in 16 bits, and is output in a four-digit ASCII code. A4 indicates the most significant digit.

A4	A3	A2	A1
15	12 11	8 7	4 3 0

(3) (-) (E)

4 bits 3 2 2 0 4 5
 0 3 0 E

(6) Stop mark: ASCII "[", HEX "50"

Indicates the end of a tape.

Codes used: ASCII, 7 bits, 8 bits

Numeric characters: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Alphabetic characters: A, B, C, D, E, F

Symbols: [, ~, -,]

Note: CR and LF must be placed right after an address.

DEL and NUL are skipped.

(4) Data mark: ASCII "-", HEX "20"

Indicates the start of data.

(5) Data: ASCII

Indicates one byte of data in a two-digit ASCII code.

The first data indicates the most significant digit. Four-bit data is expressed in a one-digit ASCII code, and the most significant digit is made to be "0".

Example

(3) (E) (-) (D) (3)

8 bits 3 3 4 5 2 0 4 4 3 3
 3 E D 3

7. ASCII-HEX

7-1 ASCII-HEX (SPACE), V1

STX \$A NNNN CR LF
 (1) (2) (3) (4)

DD DD DD --- DD CR LF
 (5) (6)

SS CCCC EXT CR LF
 (7) (8) (4) (9)

(1) Start mark: ASCII "STX", HEX "02H"

Indicates the start of data.

(2) Address mark: ASCII "\$A"

Regards the portion before the separator (4) as an address.

(3) Address: ASCII

The address is expressed in 16 bits and output is a four-digit ASCII code. If the address consists of less than four digits, zeros are placed. If the address consists of more than four digits, the last four digits are valid.

(4) Separator: ASCII ","

This is a separator code for an address and a check sum.

(5) Data: ASCII

Indicates one byte of data in a two-digit ASCII code.

(6) Data mark: ASCII "-"

The two-digit ASCII code prior to the mark is regarded as data.

(7) Check sum mark: ASCII "\$S"

The portion before the separator (4) is regarded as a check sum.

(8) Check sum: ASCII

Adds the data portion (5) as one byte of hexadecimal data, and creates 16-bit data. The result is expressed in a four-digit ASCII code, and is appended after the last data.

(9) End mark: ASCII "EXT", HEX "03H"

Indicates the end of data.

7-2 ASCII-HEX (SPACE), V2

STX DD — DD — DD DD CR LF
 (1) (5) (6)

DD — DD ETX
 (9)

(1), (5), (6), (9): Same as ASCII-HEX (SPACE).

7-3 ASCII-HEX (SPACE), V3

STX \$H AAAA — DD — DD — DD — ETX
 (1) (10) (3) (4) (5) (6) (9)

Data includes no CR and LF.

(1), (3), (4), (5), (6), (9): Same as ASCII-HEX (SPACE).

(10): Address mark: ASCII "\$H".

The portion before the separator (4) is regarded as an address.

7-4 ASCII-HEX (SPACE), V4

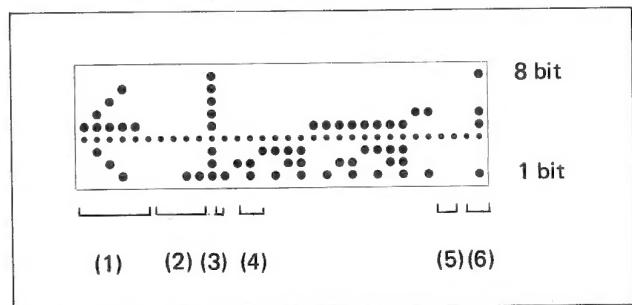
STX AAAA * DD — DD — DD — DD — CR LF
 (1) (3) (11) (5) (6)
AAAA ? DD — ETX
 (11) (9)

(1), (3), (5), (6), (9): Same as ASCII-HEX (SPACE).

(11): Address separator: ASCII

Any symbol other than the — code may be used as an address separator.

8. Binary



(1) Header mark

Indicates the start of a binary format. It consists of six bytes.

(2) Byte digit count

Indicates a data byte count. It is expressed in a hexadecimal number, ranging from 0000H (00000000H) to FFFFH (0F0F0F0FH).

(3) Data mark: HEX "FFH"

Indicates the start of data.

(4) Data

Indicates binary data.

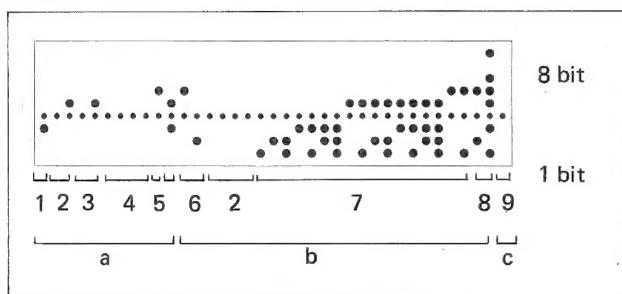
(5) Check sum mark: HEX 0000H

The subsequent two bytes are regarded as a check sum.

(6) Check sum

The check sum is created as 16 bit data by adding data (4).

9. HP-64000 Absolute



a) Header record

- (1) Indicates the number of words in a record.
 - (2) Indicates data word length.
 - (3) Indicates an address mode base.
 - (4) Indicates a transfer address.
 - (5) Indicates check sum data. $(2) + (3) + (4) = \text{sum data}$
- b) Data record
- (6) Indicates a data byte count.
 - (7) Indicates binary data.
 - (8) Indicates check sum data. $(6) + (2) + (7) = \text{sum data}$
- c) End record
- (9) Indicates the end of data. HEX = 00



MINATO ELECTRONICS INC.

4105 MINAMI YAMADA-CHO KOHOKU-KU
YOKOHAMA 223 JAPAN

Phone: 045-591-5611
Cable: "MINATOELEC" YOKOHAMA
Telex: 3822-244 MINATO J
Fax: 045-591-5618

Printed in JAPAN 86-08-1000MA