Chapter 11 7508 CPU

This chapter describes functions and use of the 7508 4-bit sub-CPU.

11.1 7508 CPU Functions

The 7508 CPU performs the following functions:

(1) Serving keyboard functions such as keyboard scan and auto repeat.

(2) Controlling the POWER switch.

(3) Controlling the RESET switch.

(4) Serving the one-second interval timer function.

(5) Measuring the battery voltage.

(6) Serving the alarm function.

(7) Turning on and off the main CPU switch.

(8) Reading temperature data.

(9) Serving the calendar and clock functions.

(10) Reading data from an AD converter.

(11) Controlling the DIP switches.

(12) Transferring serial data to and from main CPU.

(13) Controlling the DRAM refresh mode.

In addition to generating interrupts, the 7508 CPU transfers commands and data to and from the Z80 CPU via

a serial data line using a handshake technique.

The processing results for functions (1) through (6) on the previous page are returned to the Z80 in the form of interrupts.

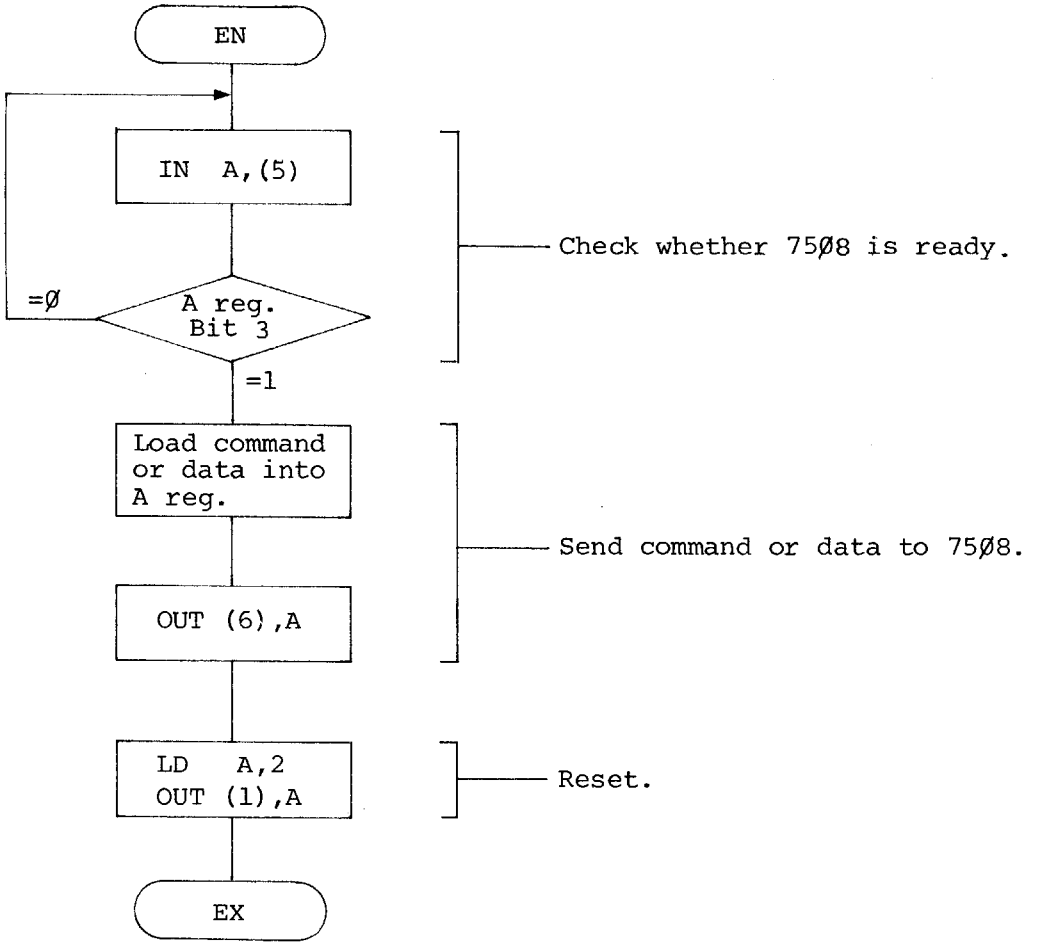
11.2 Interface to Z8Ø

The Z80 CPU uses the following ports when interfacing with the 7508 CPU:

Port	Read/Write	Meaning
06H	Read	Data from the 7508.
	Write	Data to the 7508.
05H	Read	Bit 3 carries the control signal for the serial bus to the 7508. 1: Accessible. 0: Inaccessible.
01H	Write	Used to reset the above control signal. 1: Resets. 0: Does nothing.

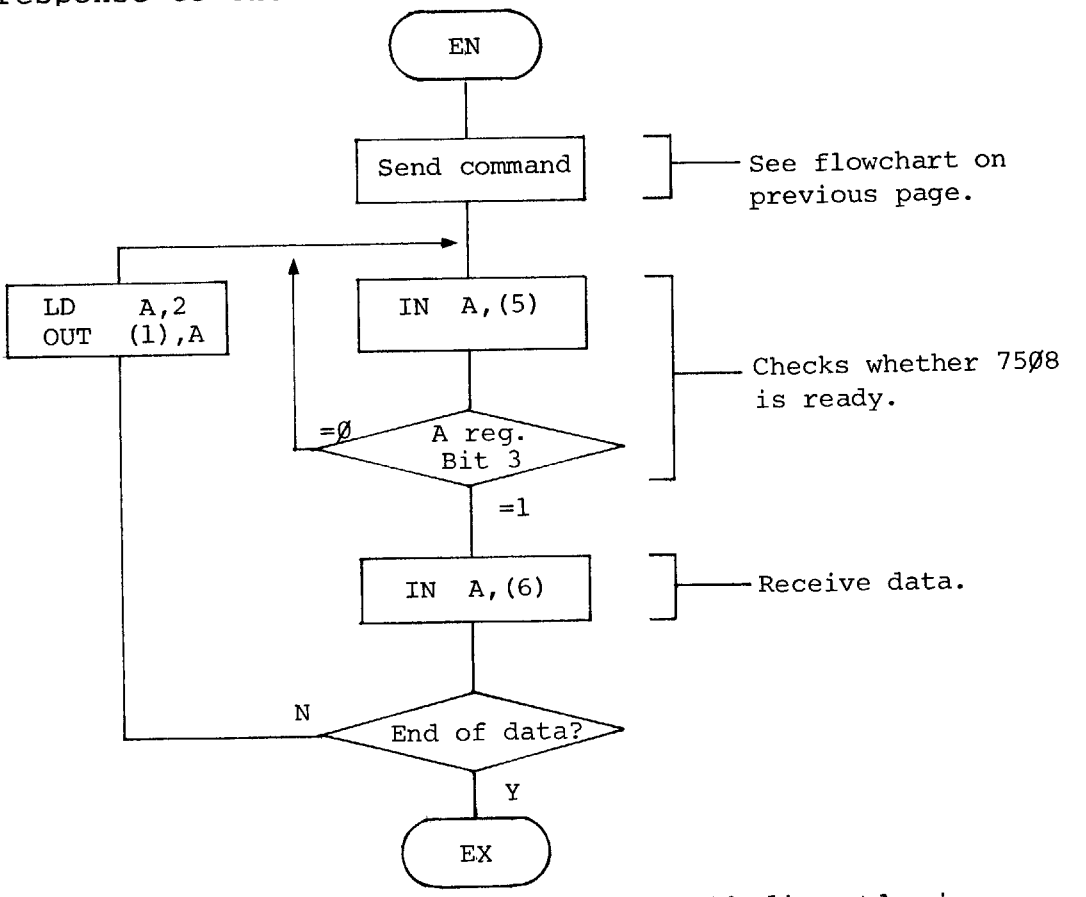
The flowchart on the next page illustrates the procedure for transferring commands or data to and from the 7508 CPU using the above I/O ports.

When sending a command or data to the 75Ø8:



When one or more parameters are to be sent following the command, the above procedure is repeated the number of times equal to the number of command and parameter bytes.

When sending a command and receiving data as the response to the command:



Points to be noted when using the 7508 directly in an application program

1) Disable 7508 interrupts while transferring a command or data to or from the 7508 CPU. (Use the DI instruction or the BIOS MASKI call.)
If a 7508 interrupt occurs while the application program is communicating with the 7508 CPU, the Z80 CPU may not receive correct return information or, at the worst case, it may hang up because the Z80 CPU will call for a new 7508 service from its interrupt handling routine and consequently the original command to the 7508 CPU will be ignored.

2) Complete the send or receive sequence for a command before proceeding with the next command. Normal processing cannot be guaranteed unless the application program sends or receives the required number of data bytes; otherwise, a system hangup would result in the worst case.

11.3 7508 Commands

The table below lists the commands that the 7508 CPU receives from the Z80 CPU.

Command Chart

Command function	Code	Command function	Code
Power off Z8Ø	ØlH	Read time.	Ø7H
Read 75Ø8 status.	Ø2H	Set alarm.	19H
Reset keyboard.	Ø3H	Read alarm.	Ø9H
Set keyboard repeat start time	Ø4H	Disable alarm.	29H
Set keyboard repeat interval.	14H	Enable alarm.	39H
Read keyboard repeat start time	24H	Read battery voltage.	ØCH
Read keyboard repeat interval.	34H	Read temperature.	1CH
Disable keyboard auto repeat.	Ø5H	Read analog input 1.	2CH
Enable keyboard auto repeat.	15H	Read analog input 2.	3CH
Disable key-in interrupt	Ø6H	75Ø8 power-on reset	ØFH
Enable key-in interrupt.	16H	Read DIP-SW	ØAH
Disable one-second interrupt.	ØDH	Set power failure detect voltage.	ØBH
Enable one-second interrupt.	1DH	Set full charge voltage.	1BH
Set time.	17H	Read power or trigger switch.	Ø8H

(1) Power off Z80

Code: 01H

Send data: None.

Receive data: None.

Function: Turns off power to the Z80.

Note: This command is not used in application programs. It is used by the POWEROFF BIOS function.

(2) Read 7508 status

Code: 02H

Send data: None

Receive data: 1 byte (7508 status)

Function: Reads the 7508 status. It is used to read the 7508 status when an interrupt occurs to identify the interrupt source. The meanings of the status byte are as follows:

- 0BEH and below: Interrupts from the keyboard.
- 0C0H and above: Interrupts from sources other than the keyboard.
- 0BFH: End of status.

1) Interrupts from the keyboard

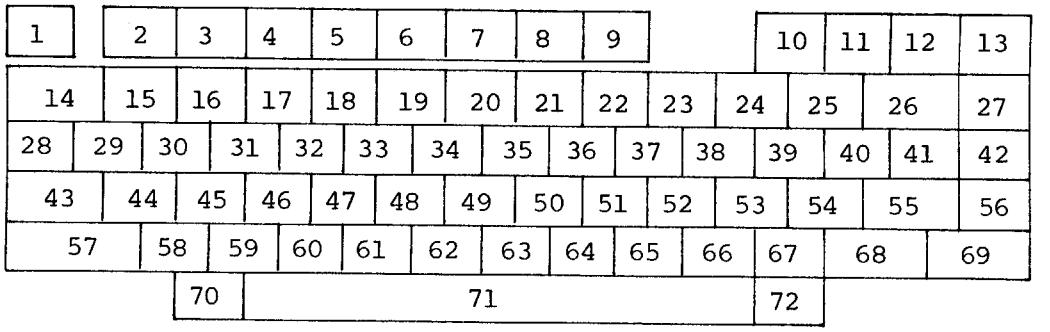
The status byte 0BEH and below indicate interrupts from the keyboard. The correspondence between the keys and status values are shown on the next page. For example, status code 73H is returned when the space key (No. 71 on the keyboard) is pressed.

The 7508 returns only one status code when an ordinary key is pressed and released. For a key No. 43, 57, 70, 72, 68, or 69, however, the 7508 returns a status code (0B2H - 0B7H) when the key is pressed and returns another code (0A2H - 0A7H) when it is released.

2) Interrupts from sources other than the keyboard

The status byte ØCØH and above indicate interrupts from sources other than the keyboard. Each bit of the status byte has the meaning listed below. When two or more interrupts occur simultaneously, the corresponding bits are set to 1.

Correspondence between the key numbers and key codes



Higher / Lower	0	1	2	3	4	5	6	7	8	9	A	B
0	2	1	29	46	62	21	37	54	12			
1	3	14	30	47	63	22	38	55	13			
2	4	15	31	48	64	23	39	56			OFF 43	ON 43
3	5	16	32	49	65	24	40	71			OFF 57	ON 57
4	6	17	33	50	66	25	41	58			OFF 70	ON 70
5	7	18	34	51	67	26	42	59			OFF 72	ON 72
6	8	19	35	52	10	27	44	60			OFF 68	ON 68
7	9	20	36	53	11	28	45	61			OFF 69	ON 69
8												
9												

Status

Bit 7: Always set to 1.

Bit 6: Always set to 1.

Bit 5: Set to 1 when a one-second interrupt occurs.

Bit 4: Set to 1 when the RESET switch on the left-side panel of the MAPLE main unit is pressed.

Bit 3: Set to 1 when the INITIALIZE RESET switch on the rear panel of the main unit is pressed.

Bit 2: Set to 1 when a power fail interrupt occurs.

Bit 1: Set to 1 when an alarm interrupt occurs.

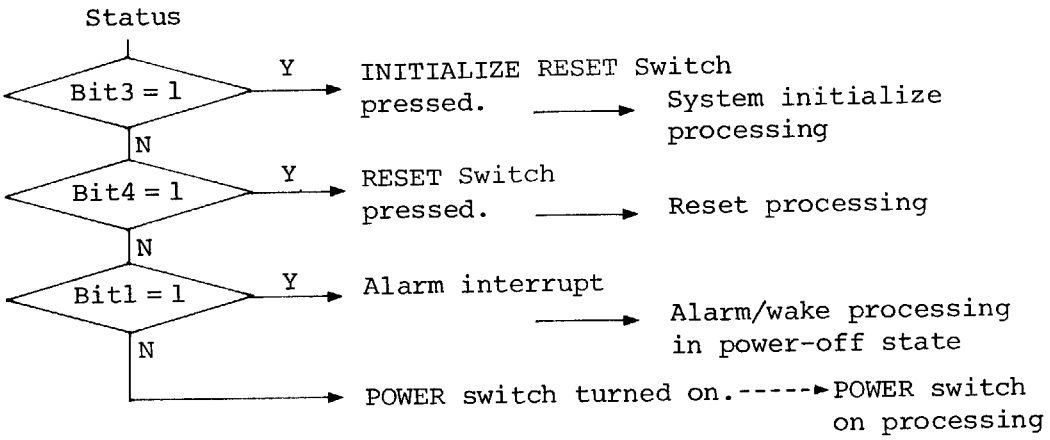
Bit Ø: Indicates the POWER switch state.

1: Power turned on.

Ø: Power turned off.

This status information is used to distinguish between address Ø start interrupts and power-on interrupts.

i) The source of an address Ø start interrupts (Z80 CPU starting at address 0) can be identified by examining the status bits in the sequence shown below.



ii) 16 status values may be returned by power-on interrupts. Since status byte bit Ø always indicates the state of the POWER switch, the correct interrupt source cannot be determined unless the POWER switch state immediately before the interrupt is known. The table below is used to identify the interrupt source for status values of ØCØH through ØC7H. The interrupt sources for status values of ØEØH through ØE7H correspond to ØCØH through ØC7H on a one-to-one basis and their meanings are identical except that they also indicate the occurrence of a 1-second interrupt.

Status	POWER switch state before interrupt	
	OFF	ON
CØH		POWER switch turned off.
C1H	POWER switch turned on.	
C2H	Alarm interrupt	POWER switch off and alarm interrupts occurred simultaneously.
C3H	POWER switch turned on.	Alarm interrupt.
C4H	Power fail interrupt	POWER switch turned off.
C5H	Power fail interrupt	Power fail interrupt
C6H	Power fail interrupt	POWER switch turned off.
C7H	Power fail interrupt	Power fail interrupt

*1: Since power is already turned on, the interrupt handling routine need nothing but to set the flag.

*2: This state cannot occur.

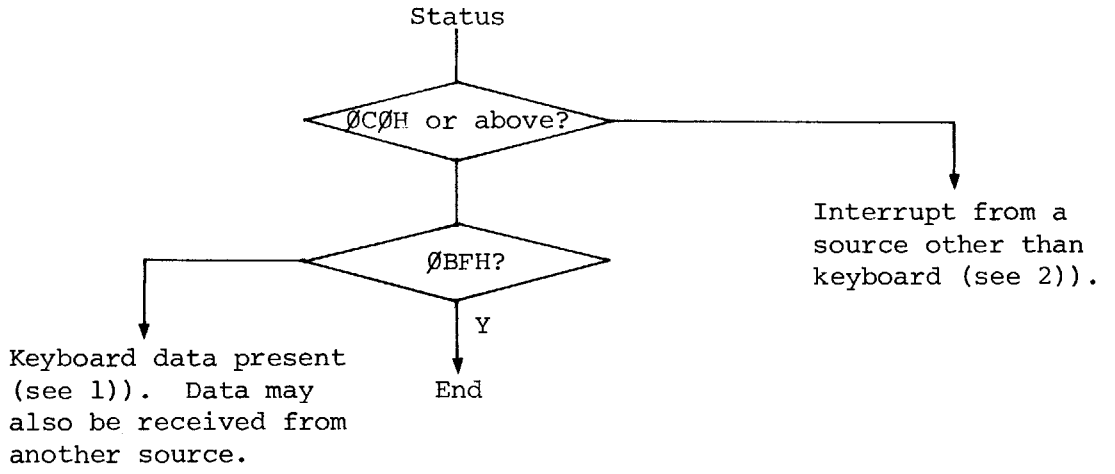
Interrupts are handled as follows when more than one status bit is 1:

- The power fail interrupt has the highest priority.
- Processing of the alarm interrupt may be deferred since a total of 10 alarm interrupts are generated.

For interrupt status values 0E0H to 0E7H, the interrupt handling routines need only perform one-second interrupt processing in addiction to the interrupt processing associated with status values 0C0H to 0C7H.

3) When the status value is 0BFH

The 7508 sub-CPU has a 7-byte buffer for storing keyed in data. It returns status code 0BFH when its key buffer holds no keyboard data. To read all data in the keyboard buffer, the application program need only execute this command repeatedly until a 0BFH code is received.



(3) Reset keyboard

Code: Ø3H

Send data: None.

Receive data: None.

Function:

1) Initializes the keyboard as follows:

- Sets the keyboard repeat start time to 656 ms.
- Sets the keyboard repeat interval to 7Ø ms.
- Clears the buffer.
- Enables interrupts from the keyboard.

2) Scans the keyboard and places the information concerning the currently pressed key.

(4) Set keyboard repeat start time

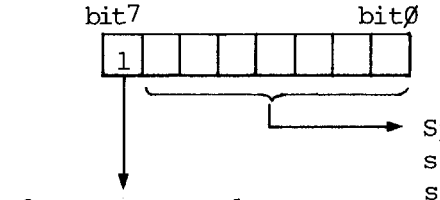
Code: Ø4H

Send data: 1 byte

Receive data: None.

Function: This command defines the interval between the time when a key is first pressed (one key code is loaded into the buffer) and the time when the auto repeat function is to be started. This function causes a key code to be read repeatedly as long as the corresponding key is held pressed.

The send data is made up of one byte and has the following format:



Specifies the repeat start time in 1/64 second increments. The maximum repeat start time is approximately two seconds ($1/64 \times 2^7$).

Always set to 1.

(5) Set keyboard repeat interval

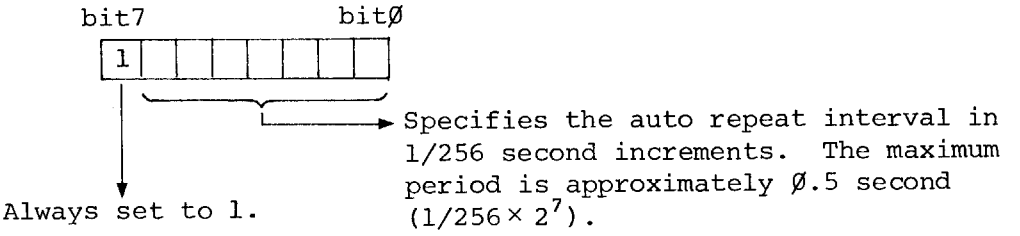
Code: 14H

Send data: 1 byte

Receive data: None.

Function: Defines the interval at which the key code of a key being held pressed is to be entered repeatedly.

The send data consists of one byte and has the following format:



(6) Read keyboard repeat start time

Code: 24H

Send data: None.

Receive data: 1 byte

Function: Returns the currently set keyboard repeat start time. The send data specifies the keyboard repeat start time in 1/64 second increments. Bit 7 is always set to Ø.

(7) Read keyboard repeat period

Code: 34H

Send data: None.

Receive data: 1 byte

Function: Returns the currently set keyboard repeat interval. As with command (5), send data specifies the repeat interval in 1/256 second increments. Bit 7 is always set to Ø.

(8) Disable keyboard auto repeat

Code: Ø5H

Send data: None.

Receive data: None.

Function: Disables the keyboard auto repeat function.

(9) Enable keyboard auto repeat

Code: 15H

Send data: None.

Receive data: None.

Function: Enables the keyboard auto repeat function.

(1Ø) Disable key-in interrupts

Code: Ø6H

Send data: None

Receive data: None

Function: Disables key-in interrupts to the Z80 CPU. When a key is pressed after this command is executed, only the key code is placed in the 7508 buffer and no interrupt request is sent to the Z80 CPU. When a command (11) is subsequently executed, a key-in interrupt is generated at this moment to the Z-80 unless the buffer is empty.

(11) Enable key-in interrupts

Code: 16H

Send data: None.

Receive data: None.

Function: Enables key-in interrupts to the Z80 CPU.

(12) Disable one-second interrupts

Code: ØDH

Send data: None.

Receive data: None.

Function: Disables one-second interrupts.

(13) Enable one-second interrupts

Code: 1DH

Send data: None.

Receive data: None.

Function: Enables one-second interrupts.

(14) Set time

Code: 17H

Send data: 8 bytes

Receive data: None.

Function: Specifies the year, month, day, hour, minute, second, and day of the week for the calendar/clock controlled by the 7508 CPU. The send data has the following format:

	bit 7	4	3 bit Ø
①	1	Ø Ø Ø	Tens digit of year
②	1	Ø Ø Ø	Units digit of year
③	1	Tens digit of month	Units digit of month
④	1	Tens digit of day	Units digit of day
⑤	1	Tens digit of hour	Units digit of hour
⑥	1	Tens digit of minute	Units digit of minute
⑦	1	Tens digit of second	Units digit of second
⑧	1	Ø Ø Ø	Day of the week

All items are defined in BCD notation. The calendar/clock is updated when the last parameter byte is received. Any item whose bits are set to all 1s is not updated (this allows partial

update). The day of the week is automatically updated within the range Ø through 6.

Since the 7508 CPU makes no check on the set data, the contents of the calendar/clock will not be guaranteed if logically invalid data is given.

Bit 7 of send data bytes is always set to 1. The time is represented in 24-hour system.

(15) Read time

Code: Ø7H

Send data: None.

Receive data: 8 bytes

Function: Reads the contents of the 7508 calendar/clock. The format of the send data is shown below. All items are specified in BCD.

	bit 7 4	3 bit Ø
①	Ø Ø Ø Ø	Tens digit of year
②	Ø Ø Ø Ø	Units digit of year
③	Tens digit of month	Units digit of month
④	Tens digit of day	Units digit of day
⑤	Tens digit of hour	Units digit of hour
⑥	Tens digit of minute	Units digit of minute
⑦	Tens digit of second	Units digit of second
⑧	Ø Ø Ø Ø	Day of the week

(16) Set alarm

Code: 19H

Send data: 6 bytes

Receive data: None.

Function: Sets the month, day, hour, minute, second, and day of the week for the alarm. The format of send data is as follows:

	bit 7	4	3 bit Ø
①	1	Tens digit of month	Units digit of month
②	1	Tens digit of day	Units digit of day
③	1	Tens digit of hour	Units digit of hour
④	1	Tens digit of minute	Units digit of minute
⑤	1	Ø Ø Ø	Units digit of second
⑥	1	Ø Ø Ø	Day of the week

All items are specified in BCD. Items whose bits are all 1s are "don't care." (Setting the minute field to all 1s causes alarm interrupts to be generated every minute.) The second must be set in ten second increments.

The time is represented in 24-hour system. Bit 7 of send data bytes is always set to 1. Since the 7508 makes no check on the set data, the contents of the alarm will not be guaranteed if logically invalid data is sent to the 7508 CPU.

Command (19) must be executed after this command to enable the alarm function.

(17) Read alarm

Code: 09H

Send data: None.

Receive data: 6 bytes

Function: Reads the currently set alarm time (month, day, hour, minute, second, and day of the week).

The send data must be specified in the following format:

	bit 7 — 4	3 — bit Ø
①	Tens digit of month	Units digit of month
②	Tens digit of day	Units digit of day
③	Tens digit of hour	Units digit of hour
④	Tens digit of minute	Units digit of minute
⑤	Ø Ø Ø Ø	Units digit of second
⑥	Ø Ø Ø Ø	Day of the week

All items are specified in BCD notation.

(18) Disable alarm

Code: 29H

Send data: None.

Receive data: None.

Function: Disables alarm interrupts to the Z80 CPU.

(19) Enable alarm

Code: 39H

Send data: None.

Receive data: None.

Function: Enables alarm interrupts to the Z80 CPU. It must be executed at least once after the alarm is set by command (16). (This command may be executed before setting the alarm time.)

(20) Read battery voltage

Code: 0CH

Send data: None.

Receive data: 1 byte

Function: Reads the main battery voltage in digital form. The relationship between the voltage and receive data is shown in the figure on the next page.

(21) Read temperature

Code: 1CH

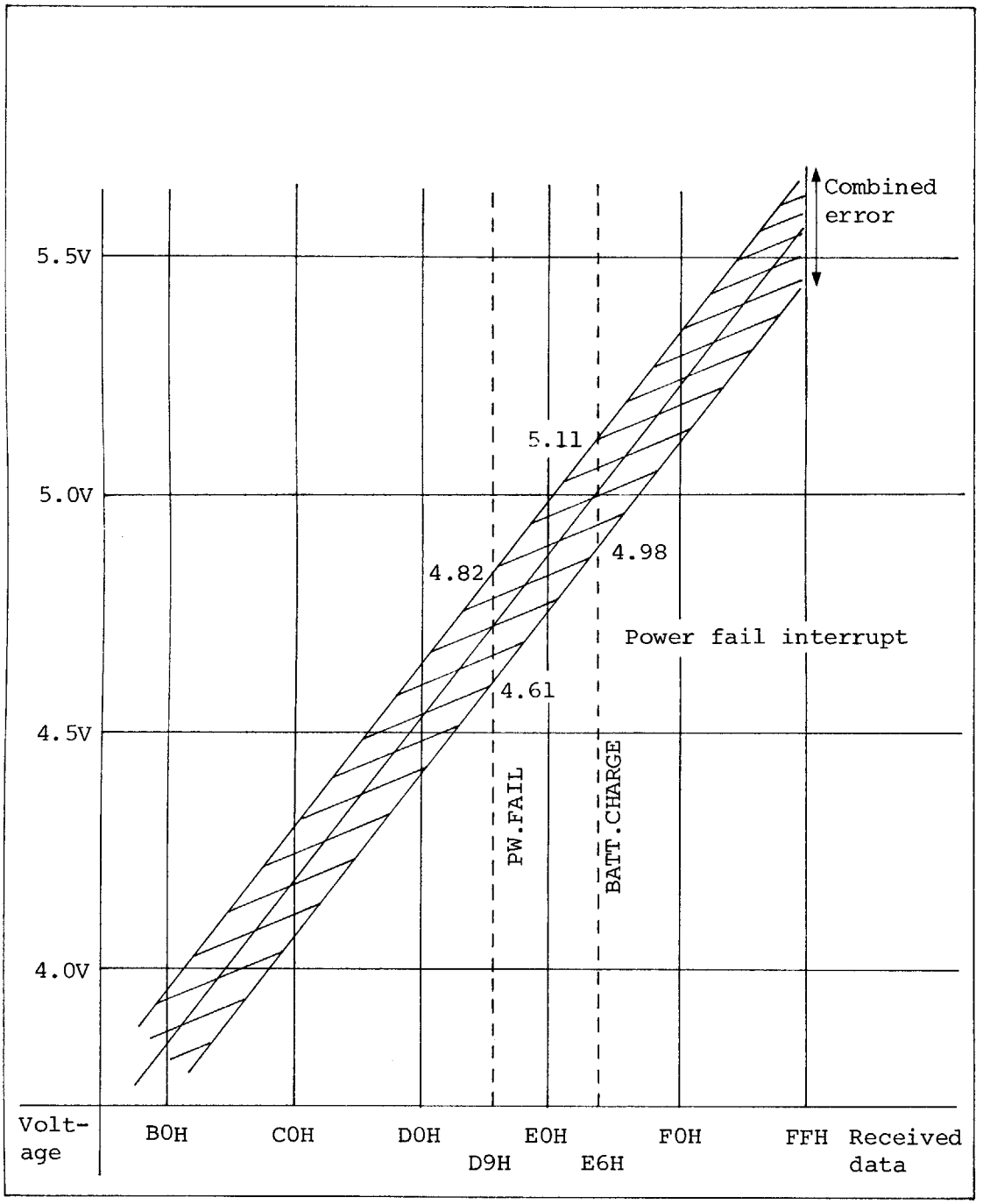
Send data: None.

Receive data: 1 byte

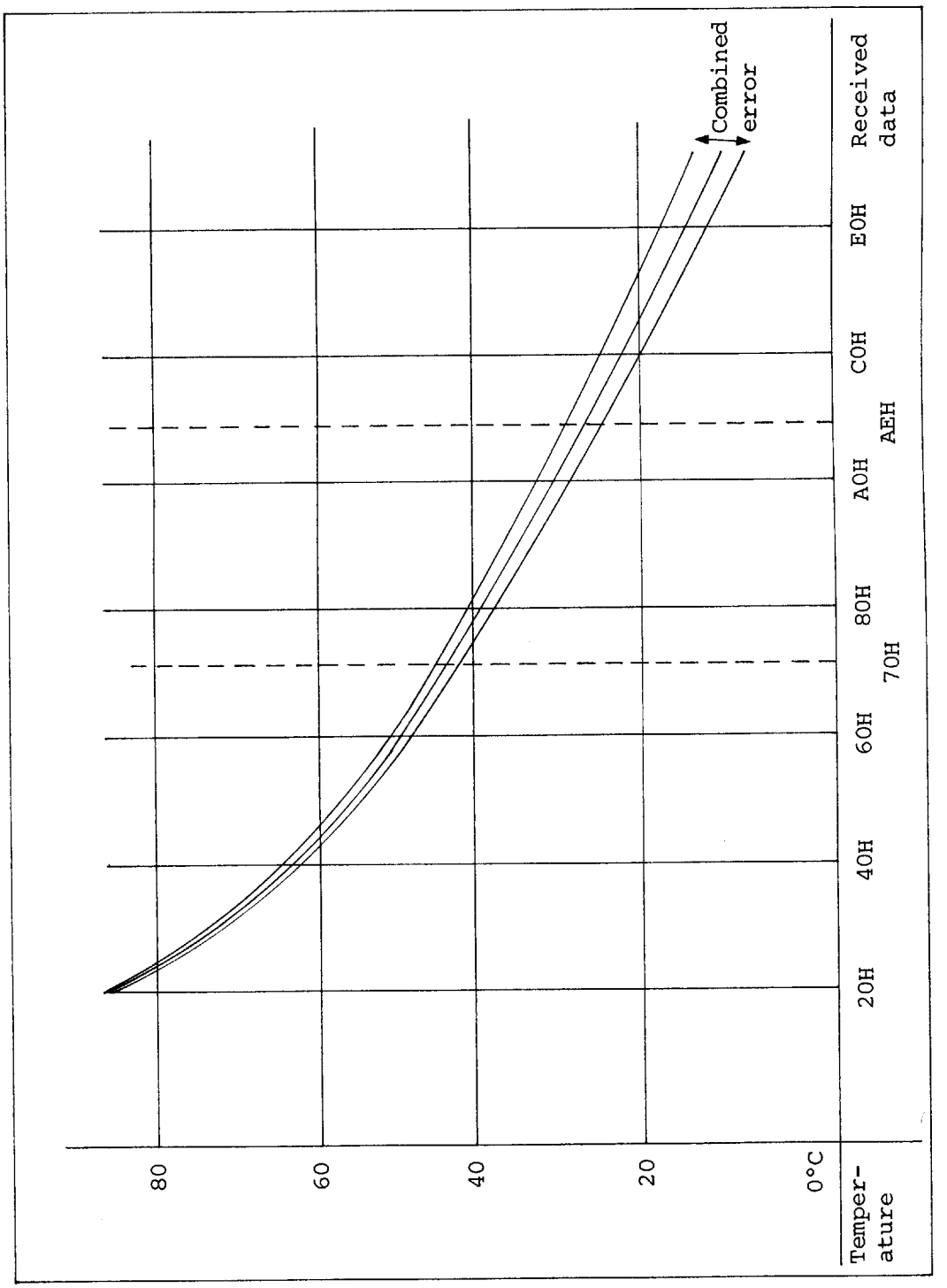
Function: Reads the current temperature in digital form. The relationship between the temperature and receive data is shown in the figure on the next page.

Battery voltage and receive data (7508)

- The receive data is linearly proportional to the battery voltage.
- The combined errors including the scatters of resistance and standard voltage are shown below.



The graph below shows the correspondence between the temperature and the received data with combined errors.



(22) Read analog jack 1

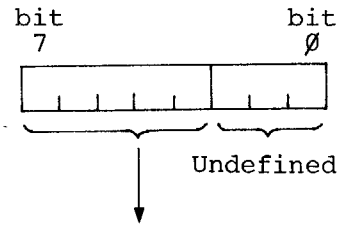
Code: 2CH

Send data: None.

Receive data: 1 byte

Function: Converts analog data from the analog data input jack to digital data.

The received data has the following format:



The highest order 6 bits represent analog data voltages Ø to +2 V. Each bit represent an increment of $2V \div 2^6 \cong 32$ mV (resolution). These bits are set to all 1s when a voltage higher than +2V is input. They are set to Ø when a negative voltage is input.

(23) Read analog jack 2

Code: 3CH

Send data: None.

Receive data: 1 byte

Function: Converts analog data from the bar code reader jack to digital data. The format of the received data is the same as that for the Read Analog jack 1 command (22) .

(24) 7508 power-on reset

Code: ØFH

Send data: None.

Receive data: None.

Function: Resets (initializes) the 7508 sub-CPU.

Note: This command is not used by application programs.

(25) Read DIP-SW

Code: ØAH

Send data: None.

Receive data: See Chapter 15.

Function: Reads the settings of the DIP switches on the main unit rear panel. See Chapter 15 for the functions of the individual DIP switches.

(26) Set power failure detect voltage

Code: ØBH

Send data: 1 byte

Receive data: None.

Function: Defines the voltage at which power fail interrupts are to be generated to the Z-8Ø CPU. A power fail interrupt is generated when battery voltage falls below this voltage. The relationship between the send data and the set voltage is the same as that shown in the figure on page 11-24.

(27) Set full charge voltage

Code: 1BH

Send data: 1

Receive data: None

Function: Defines the voltage at which full charging for the back-up battery is to be started. The relationship between the send data and the set voltage is the same as that shown on page 11-24.

There are tow ways to charge batteries: full charging (a battery is fully charged in eight hours) and trickle charging (a battery is fully charged in 30 hours). When the AC adapter is connected, full charging is performed for the first eight hours and then switched to trickle charging. Battery voltage drops gradually if the MAPLE is kept in operation during a tickle charge. This command is used to set the voltage at which full charging is to be started.

(28) Read POWER or TRIGGER switch

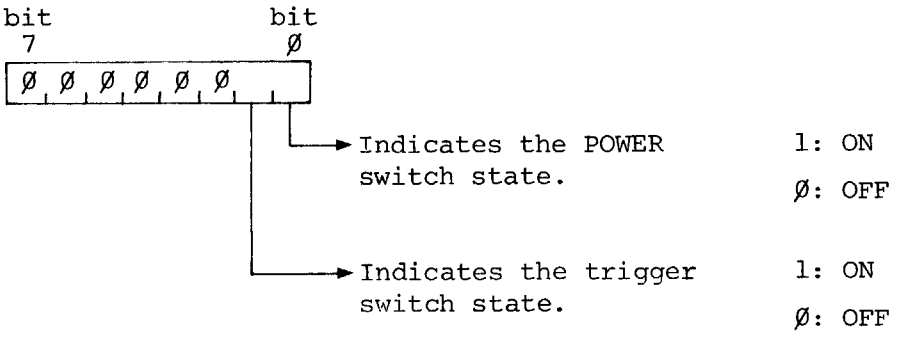
Code: ØBH

Send data: None.

Receive data: 1 byte

Function: Reads the state of the POWER switch (slide switch) on the main unit right-side panel or the state of the analog data input connector trigger.

The receive data has the following format:



Chapter 12 Using 8251A Programmable Serial Controller

The MAPLE uses a CMOS type RS-232C controller equivalent to the Intel 8251A Programmable Serial Controller. Refer to an 8251A manual for details on the functions and specifications for the 8251A. This chapter explains how to interface the Z80 CPU with the 8251A and how to control the transmitter/receiver clocks that determine the bit rate of the RS-232C interface.

12.1 Interface between the Z80 and the 8251A

The Z80 CPU can exchange commands and data with the 8251A through the I/O port addresses 0CH and 0D.

Port	Read Mode	Write Mode
0CH	8251A receive data	8251A send data
0DH	8251A status	Command to 8251A

The Z80 CPU uses no special sequence when accessing I/O port addresses 0CH and 0D. Refer to an 8251A manual for the meanings of the 8251A status and commands.

12.2 Controlling the 8251A Transmitter/Receiver Clocks

The 8251A needs a clock with a frequency 1x, x16, or x64 times higher than the bit rate when it is to be used in asynchronous mode. The MAPLE controls the clock generator for that clock by outputting a command data into bits 4-7 of port 00H. The bit rate factor (X1, X16, or X64) can be specified by outputting a mode command into the 8251A.

To output data into port 0, use the following sequence:

```
LD   A, (CTLR1)
AND  0FH
OR   [    ]        ;Bits 7-4: Select one of the
                     clocks listed in the table on the
                     next page.
                     Bits 3-0: Set to all zeros.
LD   (CTLR1), A
OUT  (0), A
```

CTLR1: Overseas version = 0F0B0H

Japanese-language version = 0ED90H

CTLR1 is the data output to port 0.

Port Ø				Clock		RS-232C Baud Rate			
						X 16		X64	
bit 7	bit 6	bit 5	bit 4	Transmit (TxC)	Receive (RxC)	Transmit (Tx)	Receive (Rx)	Transmit (Tx)	Receive (Rx)
Ø	Ø	Ø	Ø	1.74545 KHz	←	110	←		
Ø	Ø	Ø	1	2.4K	←	150	←		
Ø	Ø	1	Ø	4.8K	←	300	←		
Ø	Ø	1	1	9.6K	←	600	←	150	←
Ø	1	Ø	Ø	19.2K	←	1200	←	300	←
Ø	1	Ø	1	38.4K	←	2400	←	600	←
Ø	1	1	Ø	76.8K	←	4800	←	1200	←
Ø	1	1	1	153.6K	←	9600	←	2400	←
1	Ø	Ø	Ø	19.2K	1.2K	1200	75		
1	Ø	Ø	1	1.2K	19.2K	75	1200		
1	Ø	1	Ø	307.2K	←	19200	←	4800	←
1	1	Ø	Ø	3.2K	←	200	←		

Note: Some MAPLE overseas versions do not support a bit rate of 2ØØ (32 KHz clock).