Entry Name	TIMDAT (1)	Entry Address	WBOOT + 4BH
Function	Reads the time.		
Entry parameter	C = ØØH DE = Time descriptor starting address.		
Return parameter	DE = Time Descriptor starting address.		
Explanation			

TIMDAT (1) loads the time descriptor fields (1) to (7) with the year, month, day, hour, minute, second, and day of the week to set the clock.

Entry Name	TIMDAT (2)	Entry Address	WBOOT + 4BH
Function	Sets the time.		
Entry parameter	C = ØFFH DE = Time descriptor starting address.		
Return parameter	DE = Time descriptor starting address.		
Explanation			

TIMDAT (2) loads the time descriptor fields (1) to (7) with the year, month, day, hour, minute, second, and day of the week that are read from the clock. The BCD digits which are loaded with ØFH codes retain the previous time settings.

Since TIMDAT (2) makes no check, the validity of the subsequent information supplied by the clock is not guaranteed if logically invalid data is specified in this function.

Entry Name	TIMDAT (3)	Entry Address	WBOOT + 4BH
Function	Enables an alarm/wake function.		
Entry parameter	C = 80H		
Return parameter	None.		
Explanation			

No alarm/wake interrupt will be generated even when an alarm/wake time is specified until the alarm/wake function is enabled by TIMDAT (3).

Entry Name	TIMDAT (4)	Entry Address	WBOOT + 4BH
Function	Disables an alarm/wake function.		
Entry parameter	C = 81H		
Return parameter	None.		
Explanation			

No alarm/wake interrupt occurs once TIMDAT (4) is executed.

To use the alarm/wake function again, it is necessary to redefine alarm/wake time using the following steps:

1) Specify the alarm/wake time.

2) Enable the alarm/wake time.

Entry Name	TIMDAT (5)	Entry Address	WBOOT + 4BH
Function	Specifies the alarm/wake time.		
Entry parameter	C = 82H DE = Time descriptor starting address.		
Return parameter	DE = Time descriptor starting address.		
Explanation			

Call TIMDAT (5) after filling the month to address fields (entries (2) - (9)) in the time descriptor .

The year cannot be specified for the alarm/wake function. The value in the unit place in the second field (the lowest 4 bits of (6)) is also ignored because TIMDAT (5) monitors only the value in the ten's place.

Any BCD digits which are set to ØFH (four bits are all set to 1) in the entries from the month to the day of the week are regarded as matching any time value. For example, alarm/wake will be invoked at the specified time every day if the month and day are set to ØFFH.

Since TIMDAT (5) makes no entry data check, normal clock operation cannot be guaranteed if invalid data is specified. No alarm/wake interrupt will be generated even when an alarm/wake time is specified until the alarm/wake function is enabled by TIMDAT (5).

Entry Name	TIMDAT (6)	Entry Address	WBOOT + 4BH
Function	Reads the alarm/wake time.		
Entry parameter	C = 84H DE = Time descriptor starting address.		
Return parameter	DE = Time descriptor starting address.		
Explanation			

The current alarm/wake settings are loaded into the year to status fields of the time descriptor ((1) - (1Ø)) after TIMDAT (6) is executed. The year field and the first digit of the second field are always set to ØFFH and ØFH, respectively. This is because they are never set by TIMDAT (5).

The validity of the data loaded into the time descriptor is not guaranteed if TIMDAT (6) is executed with no alarm/wake information specified.

Entry Name	MEMORY	Entry Address	WBOOT + 4EH
Function	Does nothing.		
Entry parameter	None.		
Return parameter	None.		
Explanation			

Entry Name	RSIOX	Entry Address	WBOOT + 51H
Function	Performs various RS-232C functions.		
Entry parameter	Described below.		
Return parameter	Described below.		
Explanation			

RSIOX provides the following ten functions which are identified by the contents of the B reg.:

1. Opens RS-232C. (B = 1ØH)
2. Closes RS-232C. (B = 2ØH)
3. Informs whether RS-232C has received data. (B = 3ØH)
4. Checks whether RS-232C is enabled for transmission. (B = 4ØH)
5. Receives one character from RS-232C. (B = 5ØH)
6. Sends one character from RS-232C. (B = 6ØH)
7. Checks the control line status. (B = 7ØH)
8. Sets the control line. (B = 8Ø)
9. Checks the error status. (B = 9ØH)
10. Checks whether RS-232C is open. (B = ØFØH)

Entry Name	RSIOX (OPEN)	Entry Address	WBOOT + 51H
Function	Opens the RS-232C interface.		
Entry parameter	B = 1ØH HL = Parameter block starting address.		
Return parameter	A = ØØ: Normal termination. (Z flag = 1) = Ø2H: Already open. (Z flag = Ø) = Ø3H: Invalid parameter. (Z flag = Ø) HL = Parameter block starting address. (The parameter block is loaded with status data.)		
Explanation			

RSIOX (OPEN) initializes the RS-232C interface based on the conditions set in the specified parameter block, turns RS-232C power on, enables the RS-232C controller (8251) for receive interrupts) to ready the interface for communication.

RSIOX (OPEN) has the same functions as RSOPEN (WBOOT + 39H) except that it allows the user to initialize the RS-232C interface.

The calling program must always call RSIOX before performing I/O operations to or from the RS-232C interface.

Parameter block structure

(HL) ------→

1	Receive Buffer Starting Address	2 bytes
2	Receive Buffer Length	2 bytes
3	Baud Rate	1 byte
4	Bits/Char	1 byte
5	Parity	1 byte
6	Stop Bits	1 byte
7	Special Parameter	1 byte

(1) Receive Buffer Starting Address

Specifies the starting address of the receive buffer. The buffer may be located anywhere in the CP/M TPA.

(2) Receive Buffer Length

Specifies the length of the receive buffer.

(3) Bit Rate

Specifies the bit rate. The table below lists the codes that correspond to the available bit rates.

Code	Bit Rate (BPS)	
ØFH	192ØØ	
ØEH	96ØØ	
ØDH	48ØØ	
ØCH	24ØØ	
ØAH	12ØØ	
Ø8H	6ØØ	
Ø6H	3ØØ	
Ø5H	2ØØ	←Not supported in the overseas versions.
Ø4H	15Ø	
Ø2H	11Ø	
81H	75/12ØØ (Tx/Rx)	
8ØH	12ØØ/75 (Tx/Rx)	

→Tx and Rx represent the transmit and receive bit rates, respectively. Tx and Rx may be different.

(4) Bit/Char

Specifies the character length in bits.

Ø2H --- 7 bits/character

Ø3H --- 8 bits/character

(5) Parity

Specifies parity check type.

ØØH --- No parity

Ø1H --- Odd

Ø3H --- Even

(6) Stop Bits

Specifies the number of stop bits.

Ø1H --- 1 bit

Ø3H --- 2 bits

(7) Special Parameter

Specifies the RS-232C operating modes and status on a bit basis.

Bit	Description
Ø	Controls the DTR line. Ø: OFF (-8V) 1: ON (+8V)
1	Controls the RTS line. Ø: OFF (-8V) 1: ON (+8V)
2	Specifies whether SI/SO is to be controlled. Ø: Controlled. (Valid only for 7 bits/char. data width) 1: Not controlled.
3	Not used.
4	Specifies whether XON/XOFF control is to be used. Ø: Controlled. 1: Not controlled.
5 - 7	Not used.

This byte must be set to ØFFH when not used.

Parameter block contents on return

On return, the HL reg. retains the starting address of the parameter block that was specified on entry. The contents of the parameter block are changed as follows:

(HL)--->		
1	Status	1 byte
2	GET Point	2 bytes
3	PUT Point	2 bytes
4	Receive Buffer Starting Address	2 bytes
5	Receive Buffer Length	2 bytes

(1) Status

Indicates the RS-232C status.

Bit	Description
Ø	Indicates whether RS-232C is open. Ø: Open. 1: Not open.
1	Indicates whether the receive buffer is full. Ø: Not full. 1: Full.
2	Indicates whether a receive buffer overflow occurred. Ø: No overflow occurred. 1: Overflow occurred. Some data must have been discarded.
3	Indicates the CD line status (inverted). Ø: CD line is high. (+3 ~ +15V) 1: CD line is low. (-3 ~ -15V)
4	Indicates whether a parity error occurred. Ø: No parity error occurred. 1: Parity error occurred.
5	Indicates whether an overrun error occurred in 8251 during data reception. Ø: No overrun error occurred. 1: Overrun error occurred.

	Overrun errors are likely to occur when data transfer is too fast.
6	Indicates whether a framing error occurred during data reception. Ø: No framing error occurred. 1: Framing error occurred. Framing errors occur when the parameters of the RS-232C (bit rate, bits/char, parity, stop bits) do not match those of the counterpart terminal.
7	Indicates the DSR line status. Ø: DSR line is high. (+3 ~ +15V) 1: DSR line is low. (-3 ~ -15V)

Bits Ø, 1, 3, and 7 always indicate the current status. Bits 2, 4, 5, and 6, on the other hand, retains the error status until the RSIOX error check function is executed once an error occurred.

(2) GET Point

The address of the next data to be taken from the receive buffer.

(3) PUT point

The receive buffer address into which the next data received by 8251 is to be placed.

(4) Receive Buffer Starting Address

The address specified on entry.

(5) Receive Buffer Length

The length specified on entry.

Entry Name	RSIOX (CLOSE)	Entry Address	WBOOT + 51H
Function	Closes the RS-232C interface.		
Entry parameter	B = 20H		
Return parameter	None.		
Explanation			

RSIOX (CLOSE) turns RS-232C power off and disables RS-232C receive interrupts. The functions of RSIOX (CLOSE) is identical to those of RSCLOSE (WBOOT + 3CH).

Entry Name	RSIOX (INSTS)	Entry Address	WBOOT + 51H
Function	Indicates whether there is any data in the receive buffer.		
Entry parameter	B = 30H HL = Starting address of the field for storing 9-byte return information.		
Return parameter	Described below.		
Explanation			

The status information that RSIOX (INSTS) returns on termination is as follows:

(1) Z flag = 1: Normal termination.

A = 0FFH: Data has been received.

= 00H: No data in the receive buffer.

BC = Number of bytes of received data in the buffer.

HL = Address specified on entry. The nine bytes starting at this address contains the return information described earlier (see RSIOX (OPEN)).

(2) Z flag = 0: Abnormal termination.

A = 03H: RS-232C is not open.

HL retains the previous value.

Entry Name	RSIOX (OUTST)	Entry Address	WBOOT + 51H
Function	Checks whether RS-232C is enabled for transmission.		
Entry parameter	B = 40H HL = Starting address of the field for storing 9-byte return information.		
Return parameter	Described below.		
Explanation			

The status information RSIOX (OUTST) returns on termination is as follows:

(1) Z flag = 1: Normal termination.

A = 00H: Transmission disabled.

= 0FFH: Transmission enabled.

HL = The address specified on entry. The nine bytes starting at this address contains the return information described earlier (see RSIOX (OPEN)).

The RS-232C interface is enabled for transmission when the following two conditions are satisfied:

1) 8251 TxRDY = 1

(For Overseas Version 1.0, TxEMPTY must also be set to 1.)

2) No XOFF is received when XON/XOFF control is specified.

(2) Z flag = 0: Abnormal termination.

A = 03H: RS-232C is not open.

HL retains the previous value.

Entry Name	RSIOX (GET)	Entry Address	WBOOT + 51H
Function	Receives one character from the RS-232C interface.		
Entry parameter	B = 50H HL = Starting address of the field for storing 9-byte return information.		
Return parameter	Described below.		
Explanation			

RSIOX (GET) returns the following status on termination:

(1) Z flag = 1: Normal termination.

A = Received data.

HL = The address specified on entry. The nine bytes starting at this address contains the return information described earlier (see RSIOX (OPEN)).

(2) Z flag = 0: Abnormal termination.

A = 03H: RS-232C is not open.

= 04H: CTRL/STOP key is pressed.

HL retains the previous value.

The actual function of RSIOX (GET) is identical to that of RSIN (WBOOT + 45H).

Entry Name	RSIOX (PUT)	Entry Address	WBOOT + 51H
Function	Transfers one character to the RS-232C interface.		
Entry parameter	B = 60H C = Send data HL = Starting address of the field for storing 9-byte return information.		
Return parameter	Described below.		
Explanation			

RSIOX (PUT) returns the following status on termination:

(1) Z flag = 1: Normal termination.

HL = The address specified on entry. The nine bytes starting at this address contains the return information described earlier (see RSIOX (OPEN)).

(2) Z flag = 0: Abnormal termination.

A = 03H: RS-232C is not open.

= 04H: CTRL/STOP key is pressed.

HL retains the previous value.

The actual functions of RSIOX (PUT) is identical to those of RSOUT (WBOOT + 48H).

Entry Name	RSIOX (CTLIN)	Entry Address	WBOOT + 51H
Function	Reads the control line status.		
Entry parameter	B = 70H		
Return parameter	Described below.		
Explanation			

RXIOX (CTLIN) returns the DSR and CD status when the RS-232C is open.

(1) Z flag = 1: Normal termination

A reg. Bit 7 = DSR status.

0: +3V to +8V

1: Lower than +3V

Bit 3 = CD status.

0: Lower than +3V

1: +3V to +8V

All bits other than bits 7 and 3 are set to 0.

(2) Z flag = 0: Abnormal termination

A = 03H: RS-232C is not open.

Entry Name	RSIOX (SETCTL)	Entry Address	WBOOT + 51H
Function	Sets control lines.		
Entry parameter	B = 80H C = Data set (see below).		
Return parameter	Z flag = 1: Normal termination. Z flag = 0: Abnormal termination. A = 03H: RS-232C is not open.		
Explanation			

RSIOX (SETCTL) sets the DTR and/or RTS line states according to the contents of the C reg.

C reg. Bit 0: Sets the DTR state.

= 0: DTR set to - 8V (Low)

= 1: DTR set to + 8V (High)

Bit 1: Sets the RTS state.

= 0: RTS set to - 8V (Low)

= 1: RTS set to + 8V (High)

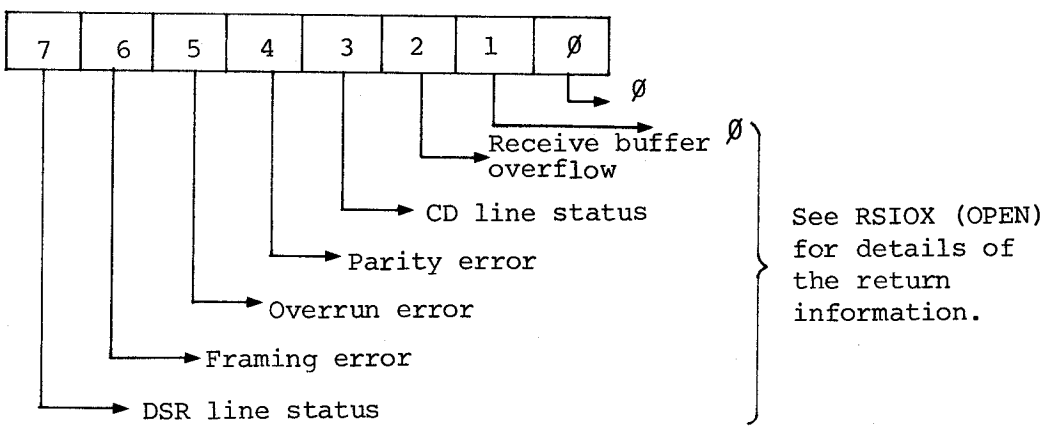
Bits 2 - 7: Not used.

Entry Name	RSIOX (ERSTS)	Entry Address	WBOOT + 51H
Function	Checks the RS-232C error status.		
Entry parameter	B = 90H		
Return parameter	Described below.		
Explanation			

RSIOX (ERSTS) returns the error status of the RS-232C interface when it is open. All errors are cleared on termination of RSIOX (ERSTS).

(1) Z flag = 1: Normal termination.

A = Error status



(2) Z flag = 0: Abnormal termination.

A = 03H: RS-232C is not open.

Entry Name	RSIOX (SENS)	Entry Address	WBOOT + 51H
Function	Checks whether the RS-232C interface is open.		
Entry parameter	B = ØFØH		
Return parameter	Z flag = 1: RS-232C is not open. A = ØØH Z flag = Ø: RS-232C is open. A = Ø2H		
Explanation			

Entry Name	LIGHTPEN	Entry Address	WBOOT + 54H
Function	Does nothing.		
Entry parameter	None.		
Return parameter	None.		
Explanation			

Entry Name	MASKI	Entry Address	WBOOT + 57H
Function	Sets or resets interrupt mask.		
Entry parameter	Described below.		
Return parameter	Described below.		
Explanation			

MASKI enables or disables the six interrupts supported by MAPLE.

(1) Entry parameters

B = Function

= 0: Inhibits interrupts from the devices whose corresponding bit in the C reg. is 1.

= 1: Enables interrupts from the devices whose corresponding bit in the C reg. is 1.

$\geq$ 2: Checks the current enabled or disabled status.

C = Specifies which type of interrupts are to be processed according to the contents in the B reg.

Bit 0: 7508 interrupts

Bit 1: RS-232C (8251) receive interrupts

Bit 2: RS-232C Carrier Detect interrupts

Bit 3: FRC overflow interrupts

Bit 4: Bar code reader interrupts

Bit 5: External interrupts

Bit 6: Not used.

Bit 7: Not used.

The Interrupts for which the corresponding bits are set to 1 are processed according to the specification in the B reg. The interrupts for which the corresponding bit is set to Ø retain their previous state. Bits 6 and 7 must be set to Ø.

(2) Return parameter

A = Loaded with return information indicating whether the individual interrupts are enabled after this function is executed. The correspondence between the bits and interrupt types is the same as that shown above. Interrupts are enabled if the corresponding bit is set to 1 and disabled if it is set to Ø.

See Chapter 1Ø for details on individual interrupts.

Entry Name	LOADX	Entry Address	WBOOT + 5AH
Function	Reads one byte of data from the specified bank.		
Entry parameter	C = Bank from which data is to be read. ØØH = User bank ØFFH = System bank HL = Address of the data to be read.		
Return parameter	A = Data Other registers retain the previous values.		
Explanation			

LOADX is used in application programs to read the contents of OS ROM. The user bank is selected when a value other than ØØH and ØFFH is specified in the C reg.

Entry Name	STORX	Entry Address	WBOOT + 5DH
Function	Writes one byte of data to the specified bank.		
Entry parameter	A = Data to be written. C = Bank to which data is to be written. ØØH = User bank ØFFH = System bank HL = Address at which data is to be written.		
Return parameter	All registers retain the previous values.		
Explanation			

STORX is not used in application programs. Nothing will happen if it is used to write data into the system bank ROM.

The user bank is selected when a value other than ØØH and ØFFH is specified in the C reg.

Entry Name	LDIRX	Entry Address	WBOOT + 60H
Function	Transfers the data on the specified bank onto another bank.		
Entry parameter	A = 00H: Transfers data from the system to user bank. = 0FFH: Transfers data from the user to system bank. HL = Starting address of the data to be transferred. DE = Starting address of the destination to which data is to be transferred. BC = Number of bytes of data to be transferred.		
Return parameter	A = 00H BC = 0000H DE = DE + BC HL = HL + BC } Register contents on termination.		
Explanation			

LDIRX is used in application programs to transfer the contents of OS ROM to RAM. Specifying a value other than 00H and 0FFH in the A reg. causes the same effect as specifying 00H.

Entry Name	JUMPX	Entry Address	WBOOT + 63H
Function	Jumps to the specified bank address.		
Entry parameter	(DISBNK) = ØØH: Jumps to the specified address on the user bank. = ØFFH: Jumps to the specified address on the system bank. IX = Destination of jump.		
Return parameter	None.		
Explanation			

JUMPX causes program execution to jump to an address in OS ROM. JUMPX is rarely used in application programs.

This BIOS call is also terminated when a RET statement is encountered in the routine at the jump address. Since control branches with the stack in the BIOS, an error may occur if the stack level goes too deep during the execution of the called routine.

The DISBNK address is:

0F539H --- for Overseas Version OS

0F2B6H --- for Japanese Version OS

Specifying a value other than 00H and 0FFH in DISBNK has the same effect as specifying 00H.

Entry Name	CALLX	Entry Address	WBOOT + 66H
Function	Calls the specified bank address.		
Entry parameter	(DISBNK) = ØØH: Calls the specified address on the user bank. = ØFFH: Calls the specified address on the system bank. IX = Called routine address		
Return parameter	None.		
Explanation			

CALLX is used by application programs to directly call a routine in OS ROM.

Since the routine is called with the stack for BIOS still, unexpected results may occur if the called subroutine uses too large an amount of stack area.

The DISBNK address is:

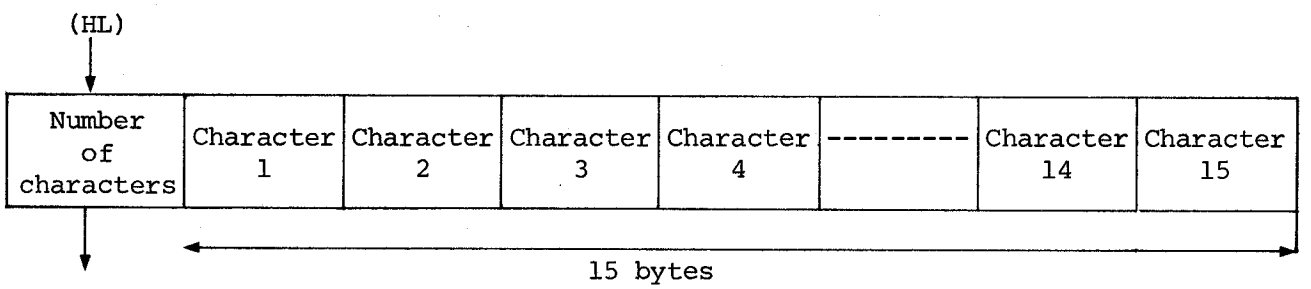
0F539H --- For Overseas Version OS

0F2B6H --- For Japanese Version OS

Specifying a code other than ØØH and ØFFH in DISBNK has the same effect as specifying ØØH.

Entry Name	GETPFK	Entry Address	WBOOT + 69H
Function	Reads in PF key data.		
Entry parameter	C = PF key number - 1 PF1 = 00H --- PF10 = 09H HL = Starting address of the character string to be read.		
Return parameter	HL = Retains the previous value.		
Explanation			

GETPFK gets a character string defined for a PF key in 16-byte format as shown below. GETPFK does nothing when a value other than 00H to 09H is specified in the C reg.



ØØH - ØFH:
Indicates the number of characters in the string. ØØH indicates that no string is defined for this PF key.

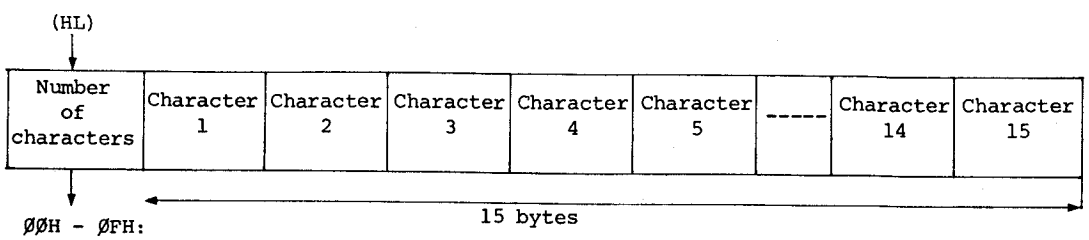
Example

Ø3H	"P"	"I"	"P"				----------		

The contents of the subsequent bytes are not guaranteed.

Entry Name	PUTPFK	Entry Address	WBOOT + 6CH
Function	Defines a PF key.		
Entry parameter	C = PF key number - 1 PF1 = ØØH --- PF1Ø = Ø9H HL = Starting address of the character string to be assigned.		
Return parameter	HL = Retains the previous value.		
Explanation			

PUTPFK assigns a character string to a PF key in the 16-byte format. The maximum string length is 15 characters. PUTPFK does nothing when a value other than ØØH to Ø9H is specified in the C reg.



Specifies the number of defined characters in binary. ØØH indicates that no string is defined for the specified PF key.

If old PF key definitions are displayed on the screen, they are also updated as they are redefined by PUTPFK.

Entry Name	ADCVRT	Entry Address	WBOOT + 6FH
Function	Performs an analog data input operation.		
Entry parameter	C = Analog data to be selected.		
Return parameter	A = AD conversion results		
Explanation			

ADCVRT converts analog data selected by the parameter in the C reg. to digital data and returns the results to the A reg.

C = 00H: A/D channel 1 --- Data from the analog jack.

C = 01H: A/D channel 2 --- Data from the bar code reader connector.

C = 02H: DIP SW settings.

C = 03H: Battery voltage.

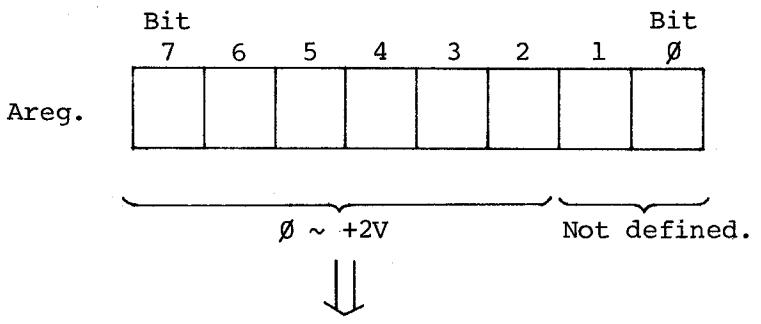
C = 04H: Main switch (for Power ON/OFF) and analog input connector trigger terminal settings.

ADCVRT does nothing when the C reg is loaded with a value other than 00H to 04H.

The pages that follow describe what data is returned to the A reg. according to the value specified in the C reg.

(1) When the C reg. = 00H or 01H

A voltage 0 to +2V applied to the A/D jack is converted to a digital quantity and placed into the highest 6 bits of the A reg. (resolution of 6 bits).



Each bit corresponds to $2V \div 2^6 \fallingdotseq 32mV$. These bits are all set to 1 when a voltage higher than +2V is input. They are set to Ø when a negative voltage is input.

(2) When the C reg. = 02H

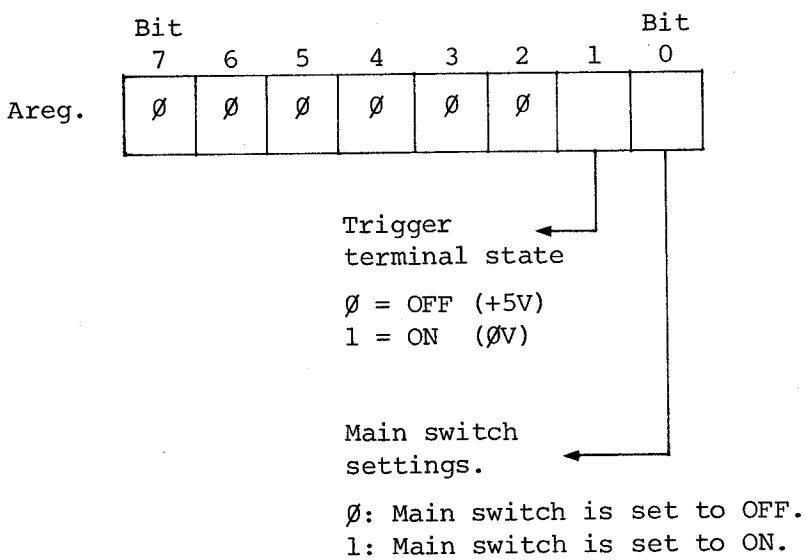
The settings for the DIP switches on the main unit back panel are placed into the A reg. in the following format:

(3) When the C reg. = 03H

The data about the battery voltage is placed in the A reg. See Chapter 11 for the correspondence between the battery voltages and the A reg. values.

(4) When the C reg. = 04H

The main switch settings and the analog input connector trigger terminal state are placed into the A reg.



MAPLE may be started even when the main switch is in the off position (by the wake function).

Entry Name	SLAVE	Entry Address	WBOOT + 72H
Function	Controls the communication with the SLAVE CPU.		
Entry parameter	DE = Communication packet starting address.		
Return parameter	A = ØØH: Normal termination. ≠ ØØH: Abnormal termination. DE = Retains the previous value.		
Explanation			

SLAVE is used by the application program to control the SLAVE CPU directly. See Chapter 13 for details on the functions that SLAVE can perform and the command and data used by SLAVE.

The SLVFLG field in the work area must be set as follows before this BIOS function is called:

The SLVFLG address is:

0F358H --- For Overseas Version OS

0F080H --- For Japanese Version OS

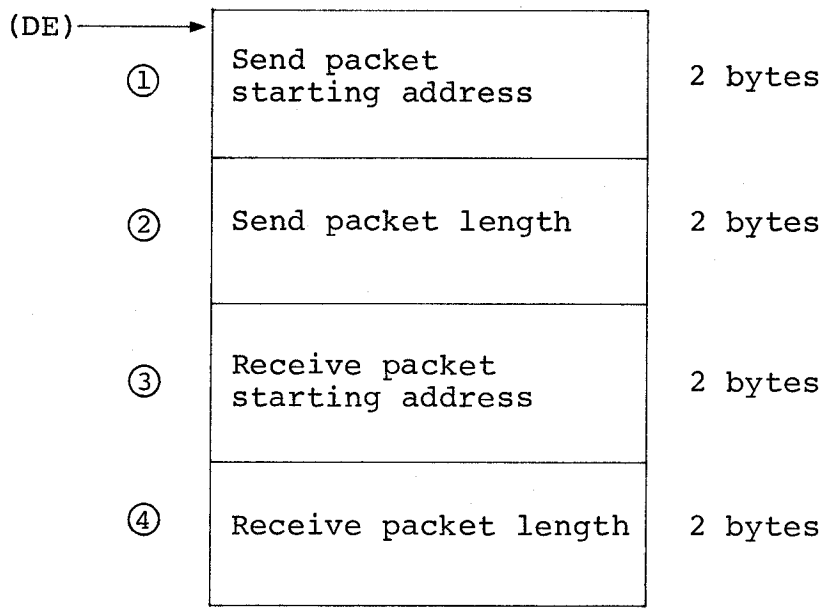
Bit 7: Always set to ON.

Bit 6: Set to ON when accessing SLAVE memory (executing command ØØH, Ø1H, or Ø2H). Otherwise, this bit is set OFF.

Bit 5: Set to ON when writing data into the SLAVE CPU privileged memory (addresses 80H - 0ADH). Otherwise, this bit is set to OFF.

SLAVE immediately terminates abnormally if the SLVFLG field is found to be set improperly. The calling program must clears the SLVFLG field to 00H after returning from this BIOS subroutine.

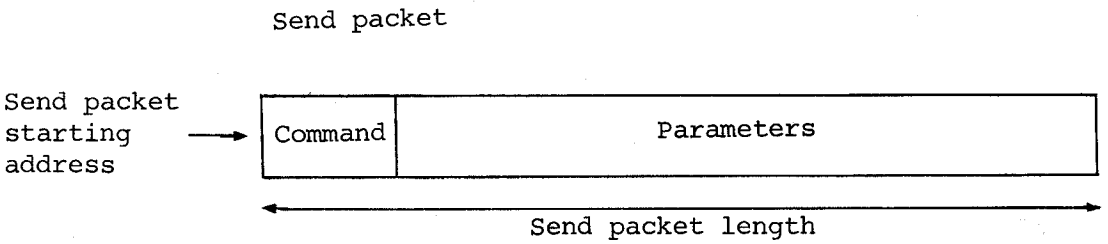
The communication packet has the following format:



(1) Send packet starting address

(2) Send packet length

A send packet refers to a buffer area which contains a command or a command plus parameters to be passed to the slave CPU.

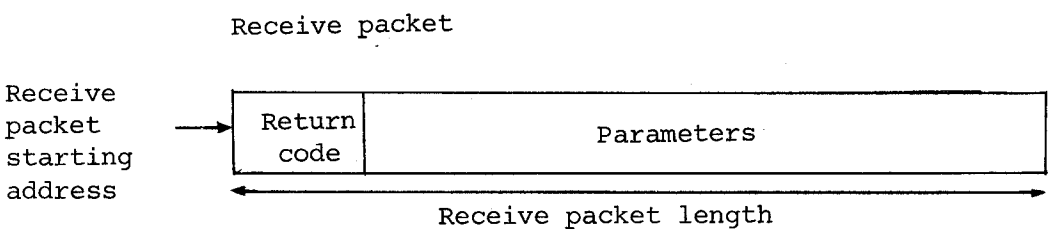


A send packet always begins with a 1-byte command, so the length of a send packet is normally longer 1 byte. When the length is Ø, SLAVE does nothing for send requests and performs only receive processing.

(3) Receive packet starting address

(4) Receive packet length

A receive packet is an area for storing the return code and parameters, if any, which the slave CPU returns after processing the command and the parameters passed from the SLAVE CPU in the above format.

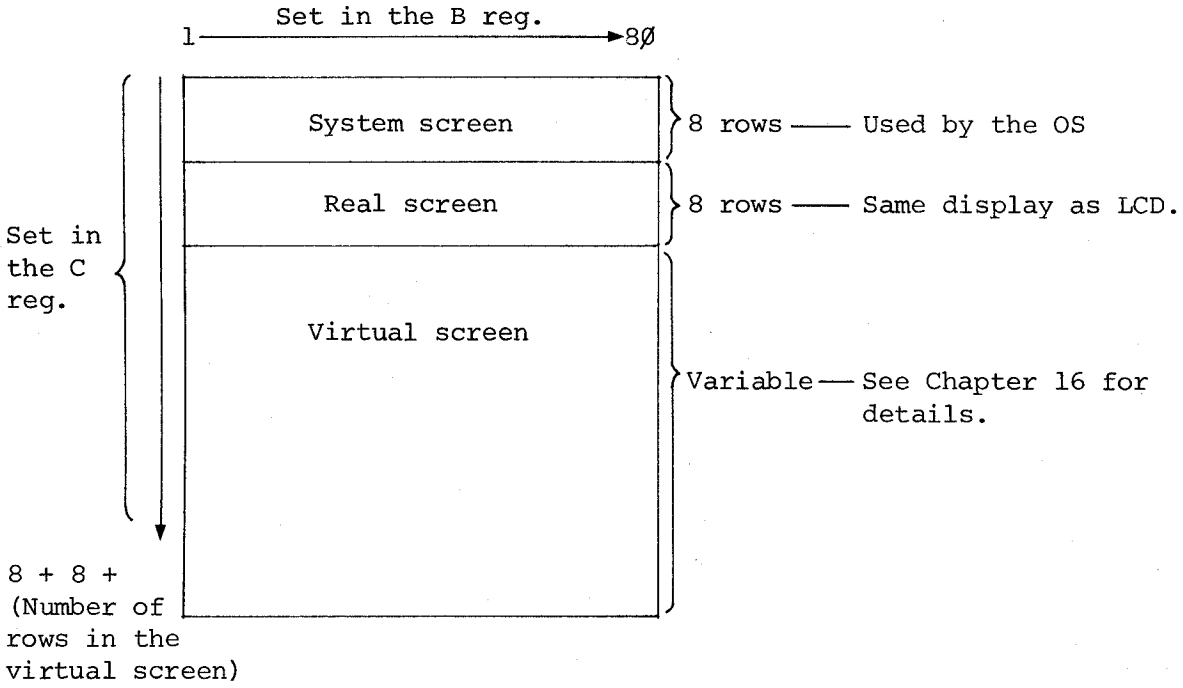


The return code and the contents of the A reg. are the same when SLAVE terminates normally. Since a return code is always returned on normal termination, the receive packet length should longer than 1 byte. When the receive packet length is Ø, no data is received from the slave CPU.

The main program can do its own tasks while the slave CPU is processing a command from the main program. The calling program can receive the return code and parameters that the slave CPU returns in response to the previous command by first sending a command or parameter with a receive packet length of Ø specified, then, after performing its main task, issuing a command with a send packet length of Ø. During this operation, however, the main program cannot perform any operation which involves slave CPU processing (e.g., screen or MCT processing). (Attempting to do so would result in a SLAVE hang-up.)

Entry Name	RDVRAM	Entry Address	WBOOT + 75H
Function	Reads the contents of VRAM.		
Entry parameter	B = Starting column number in which read is to begin. (1 -80) C = Starting row number in which read is to begin. (1 - Bottom of screen) DE = Number of characters to be read. HL = Address of the area for storing the read data.		
Return parameter	A = ØØH: Normal termination. = Ø1H: Display extends beyond the screen during a read. = FFH: Screen is in graphics mode. Or the starting position specified by B and C is outside the virtual screen. HL = Retains the previous value.		
Explanation			

RDVRAM reads the data on the character mode screen. The screen has the following structure:



RDVRAM reads the number of characters specified by DE starting at the position designated by B and C and stores them sequentially into the area starting at the address designated by DE. Characters are read from left to right in a row. After the 80th character is read, the leftmost character in the next row is read.

When the number specified in DE is too large and display extends beyond the screen, 00H codes are returned as extra characters until the number of the returned characters equals the value specified in DE. In this case, the A reg. is loaded with a return code of 01H.

Entry Name	MCMTX	Entry Address	WBOOT + 78H
Function	Processes MIOS communication.		
Entry parameter	B = MIOS function code (ØØH - 15H)		
Return parameter	Described below.		
Explanation			

MCMTX is used to communicate with MIOS (entering commands or receiving data) to control MCT directly.

See Chapter 14 for details of MIOS functions.

Entry Name	POWEROFF	Entry Address	WBOOT + 7BH
Function	Turns main power off.		
Entry parameter	C = ØØH: Main power turned off in continue mode. = Ø1H: Main power turned off in restart mode.		
Return parameter	None.		
Explanation			

POWEROFF is used in application programs to turn MAPLE main power off.

If power has been set off in continue mode, execution continues with the command following this BIOS call when power is turned on. The I/O settings established before the power-off is restored at the same time. This BIOS call must be followed by an EI instruction when power is turned off in continue mode.

If power has been set off in restart mode, execution will start at WBOOT when power is turned on.

See Chapter 9 for details of power-on/off.

Entry Name	USERBIOS	Entry Address	WBOOT + 7EH
Function	Provides the entry to USERBIOS.		
Entry parameter	None.		
Return parameter	None.		
Explanation			

USERBIOS provides an entry point through which the application program makes BIOS calls after loading its own BIOS routine in the RAM USERBIOS area. Presently, USERBIOS serves no purpose.

The following procedure must be observed when using a user-provided BIOS routine through the entry point at USERBIOS:

1) Load the BIOS routine into the RAM USERBIOS area.

2) Replace the contents of addresses (WBOOT + 7EH) + 1 and (WBOOT + 7EH) + 2 with the entry address bytes of the user routine in the USERBIOS area.

3) Call this BIOS in the application program.

See "USERBIOS Usage" for details.