Chapter 4   BIOS Subroutines

MAPLE BIOS is greatly extended for support of a number

of I/O functions. In fact, it contains as many as 44

subroutines. This chapter gives a detailed description

of these subroutines. The BIOS CONOUT routine has many

options, and therefore, a whole chapter is reserved for

it for full description of the function (see Chapter 5).

* Programming Notes on the use of BIOS calls

1) The entry to each function is indicated by the offset

from WBOOT. Find its effective address by adding this

offset to the entry address to the WBOOT located in 01H

and 02H.

2) Save the contents of registers if necessary because

the contents of the registers except those for receiving

the return parameter are not guaranteed.

The entry addresses and functions of BIOS Subroutines

Offset from WBOOT	ENTRY NAME	Function
-Ø3H	BOOT	Performs a cold BOOT.
±ØØH	WBOOT	Performs a warm BOOT.
+Ø3H	CONST	Returns the console input status.
+Ø6H	CONIN	Inputs one character from the console.
+Ø9H	CONOUT	Outputs one character to the console.
+ØCH	LIST	Outputs one character to the LIST device.
+ØFH	PUNCH	Outputs one character to the PUNCH device.
+12H	READER	Inputs one character from the READER device.
+15H	HOME	Positions the disk head to track ØØ.
+18H	SELDSK	Specifies the device.
+1BH	SETTRK	Specifies the track for read or write.
+1EH	SETSEC	Specifies the sector for read or write.
+21H	SETDMA	Specifies the DMA starting address for read or write.
+24H	READ	Reads the specified sector.
+27H	WRITE	Writes data to the specified sector.
+2AH	LISTST	Returns the status of the list device.
+2DH	SECTRN	Translates a logical sector to a physical sector.
+3ØH	PSET	Converts graphics screen data for display.
+33H	SCRNDUMP	Takes a hard copy of the displayed data.
+36H	BEEP	Sounds the speaker.
+39H	RSOPEN	Opens the RS-232C interface.

Offset from WBOOT	ENTRY NAME	Function
+3CH	RSCLOSE	Closes the RS-232C interface.
+3FH	RSINST	Informs whether the RS-232C interface has received data.
+42H	RSOUTST	Checks whether the RS-232C interface is ready for transmission
+45H	RSIN	Receives one character from the RS-232C interface.
+48H	RSOUT	Transfers one character to the RS-232C interface.
+4BH	TIMDAT	Performs clock or alarm functions.
+4EH	(MEMORY)	Does nothing.
+51H	RSIOX	Performs RS-232C functions.
+54H	(LIGHTPEN)	Does nothing.
+57H	MASKI	Sets or resets the interrupt mask.
+5AH	LOADX	Reads the data in the specified bank.
+5DH	STORX	Writes data into the specified bank.
+60H	LDIRX	Transfers data between banks.
+63H	JUMPX	Jumps to the specified bank address
+66H	CALLX	Calls the subroutine at the specified bank address.
+69H	GETPFK	Gets a PF key.
+6CH	PUTPFK	Defines a PF key.
+6FH	ADCVRT	Performs analog data input operations.
+72H	SLAVE	Processes communication with the SLAVE CPU 6301.
+75H	RDVRAM	Reads the contents of VRAM.
+78H	MCMTX	Processes communication with MIOS.

Offset from WBOOT	ENTRY NAME	Function
+7BH	POWEROFF	Turns main power off.
+7EH	USERBIOS	Entry point to the User BIOS.

Entry Name	BOOT	Entry Address	WBOOT - 03H
Function	Performs a CP/M cold boot.		
Entry parameter	None.		
Return parameter	None.		
Explanation			

BOOT is entered by a 7508 or system initialize reset (SIFT/GRPH/RESET), or the depression of the RESET key. This routine is used not by application programs but by the operating system.

BOOT performs the following:

1. Sets the current drive to A:.

2. Sets the I/O byte to 10101001B.

    LST: = LPT: (RS-232C)

    PUN: = UP1: (RS-232C)

    RDR: = UR1: (RS-232C)

    CON: = CRT: (Output: LCD, Input: Keyboard)

3. Displays the CP/M sign-on message.

4. Reads informations of the DIP switches and saves
   their settings in a work area to identify the
   keyboard (nationality) and the character set to be
   used.

5. Loads the CTRL/HELP entry in the keyboard
   subroutine table with the system display address
   and the CTRL/PF5 entry with the hardcopy address.

6. Sets the pointer to the PF key table to the system
   table.

7. Initializes the cursor movement key (arrowed key)
   codes.

8. Jumps to the routine shared with WBOOT.

Entry Name	WBOOT	Entry Address	WBOOT $\pm$0H
Function	Performs a CP/M warm boot.		
Entry parameter	None.		
Return parameter	None.		
Explanation			

WBOOT is entered when power is turned on in restart mode or a JUMP 0 is executed.

WBOOT performs the following:

1. Writes the write data left in the FDD buffer into the floppy disk.

2. Initializes the MCT parameters.

3. Restores the cursor into the state defined by CONFIG.

4. Sets the pointer to the PF key table to the system table.

5. Displays the PF key definitions on line 8 when PF key display mode is specified.

The following processing is common to WBOOT and BOOT:

6. Sets SP to the value for BIOS.

7. Turns the RS-232C interface power off.

   Turns the ROM capsule power off.

   Stops the microcassette.

8. Loads addresses 0 to 2 with the object code of JP WBOOT.

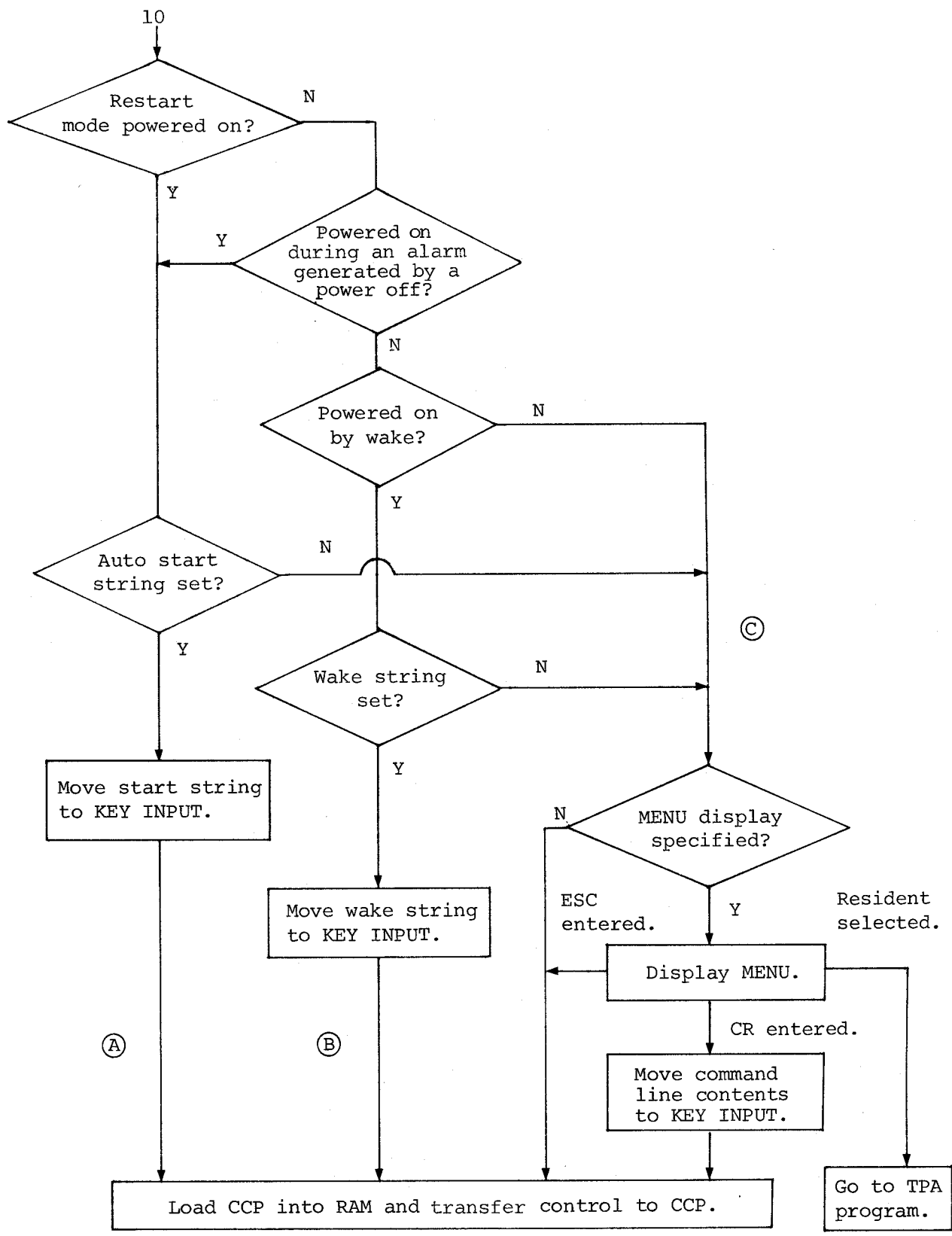   Loads addresses 5 to 7 with the object code of JP BDOSE.

                                        RAM BDOS starting address + 6

9. Loads BDOS into RAM.

The subsequent actions of WBOOT depends on the system conditions under which it has executed so far. The actions are shown in the flowchart on next page.

```
                              10
                               |
                               v
                          /         \
                         /  Restart   \        N
                         \ mode powered on? /---------+
                          \         /                 |
                            \     /                    |
                              | Y                       v
                              |                    /         \
                              |           Y       / Powered on  \
                              |      +-----------/ during an alarm \
                              |      |           \ generated by a  /
                              |      |            \ power off?    /
                              |      |              \         /
                              |      |                  | N
                              |      |                  v
                              |      |             /         \        N
                              |      |            / Powered on  \-----------------+
                              |      |            \  by wake?   /                 |
                              |      |             \         /                    |
                              |      |                 | Y                         |
                              |      |                 |                           |
                         /         \  N              __                           |
                        / Auto start \ ------------/  \-------------------------->|
                        \ string set? /             (break)                       |
                         \         /                   |                          |
                           \     /                     v                          |  (C)
                             | Y                   /         \      N             |
                             |                    / Wake string \---------------->|
                             |                    \   set?      /                 |
                             |                     \         /                    |
                             |                        | Y                         |
                             |                        |                           v
                     +----------------+               |                     /         \
                     | Move start string|             |              N     / MENU display \
                     | to KEY INPUT.    |             |          +--------/   specified?   \
                     +----------------+               |          |         \             /
                             |                        |          |          \         /
                             |                        |       ESC |            | Y       Resident
                             |                        |     entered.           |         selected.
                             |                +----------------+   |           v
                             |                | Move wake string|  |     +--------------+
                             |                | to KEY INPUT.   |  |     | Display MENU.|----+
                             |                +----------------+   |     +--------------+    |
                             |                        |          |           |               |
                        (A)  |                   (B)  |          |     CR entered.            |
                             |                        |          |           v                |
                             |                        |          |     +----------------+     |
                             |                        |          |     | Move command   |     |
                             |                        |          |     | line contents  |     |
                             |                        |          |     | to KEY INPUT.  |     |
                             |                        |          |     +----------------+     |
                             |                        |          |           |                |
                             v                        v          v           v           +----------+
    +--------------------------------------------------------------------------+          | Go to TPA|
    |     Load CCP into RAM and transfer control to CCP.                        |          | program. |
    +--------------------------------------------------------------------------+          +----------+
```

A: When an auto start string is specified and

- The power switch is turned on.

- The power switch is turned on while an alarm generated in the power off state is being displayed.

B: When power is turned on by wake with a wake string specified.

C: - After BOOT is executed.

- After WBOOT is executed.

- When power switch is turned on by wake with no wake string specified.

Entry Name	CONST		Entry address	WBOOT + 03H
Function	Returns the status of the console.			
Entry parameter	None.			
Return parameter	A = ØØH: Console input buffer is empty. A = ØFFH: Data is present in console input buffer.			
Explanation				

CONST checks the CON: field (bits Ø and 1) of the I/O byte (at address 3) to determine whether the console input device is the keyboard or RS-232C interface and returns the status of the console.

```
CON: Bit 1    Bit Ø
       Ø        Ø    ⎫  Indicates whether the keyboard buffer
       Ø        1    ⎬  is empty.
       1        Ø    ⎫  Indicates whether the RS-232C receive
       1        1    ⎬  buffer is empty.
```

Entry Name	CONIN	Entry address	WBOOT + 06H
Function	Returns one character read from the console.		
Entry parameter	None.		
Return parameter	When YPFCMFLG ≠ 0FFH A = ASCII code When YPFCMFLG = 0FFH C = 00H --> A = ASCII code C = 0FFH --> A reg. contains one of 0E0H through 0E9H which correspond to PF1 through PF9.		
Explanation			

CONIN checks the CON: field of the I/O byte like CONST, and

receives one character from the keyboard or RS-232C interface.

This routine waits until a character is received.

(1) When the keyboard is assigned to the console (I/O byte,

    bits 1 and 0 are 00 or 01)

CONIN operates in different ways depending on the state of

YPFCMFLG (at 0F108H) which controls the handling of the PF keys.

1)   When YPFCMFLG ≠ FFH

    When a PF key is pressed, CONIN returns the string defined

for that PF key. Consequently, CONIN cannot determine what
PF key is pressed. When a key other than PF keys is
pressed, CONIN returns the corresponding ASCII code.

2)   When YPFCMFLG = ØFFH

      CONIN returns via the C reg. the information as to whether a
      PF key is pressed.

      - When C = ØØH

      Indicates that a key other than PF keys is pressed and the
      corresponding ASCII code is placed in the A reg.

      - When C = ØFFH

      Indicates that a PF key is pressed. The A reg. contains
      either one of EØH through E9H which correspond to PF1
      through PF9.

YPFCMFLG is set by directly rewriting the work area or by writing
ESC + ØBØH or ESC + ØB1H through the CONOUT routine.
CONIN waits until input data is received. When the auto power
off time expires, however, power is automatically turned off
during the CONIN routine in the continue mode. When power is
turned on again, execution resumes at the CONIN wait state.

(2) When the RS-232C interface is assigned to the console
    (I/O byte, bits 1 and 0 are 10 or 11)

CONIN places the data received from the RS-232C interface into
the A reg.   When no data is present at the RS-232C interface,
CONIN waits until data is received.   The operation of this
routine is identical to that of RSIN.

Entry Name	CONOUT	Entry Address	WBOOT + 09H
Function	Outputs one character to the console.		
Entry parameter	C = output data		
Return parameter			
Explanation			

See Chapter 6 for details.

Entry Name	LIST	Entry Address	WBOOT + 0CH
Function	Outputs one character to the list device.		
Entry parameter	C = output data		
Return parameter	None		
Explanation			

LIST checks the LST: field (bits 7 and 6) of the I/O byte and sends one character to the corresponding device.

I/O byte

Bit 7	Bit 6		
0	0	(TTY):	Outputs to the serial port.
0	1	(CRT):	Outputs to the LCD (LIST operates in the same way as CONOUT).
* 1	0	(LPT):	Outputs to the RS-232C interface (LIST operates in the same way as RSOUT). LIST waits until DSR and TxRDY are set to 1 indicating that the counterpart receiver is ready for reception.

1   1   (UL1): Does nothing.

             *: Default setting.

When the I/O byte is set to serial or RS-232C interface and

LIST is used for the first time after WBOOT, LIST outputs

the command ESC + "R" + x to select the character set

corresponding to the country currently set before sending

the output data.

Entry Name	PUNCH		Entry Address	WBOOT + ØFH
Function	Outputs one character to the punching device.			
Entry parameter	C = output data			
Return parameter	None			
Explanation				

PUNCH checks the PUN: field (bits 5 and 4) of the I/O byte and sends one character to the corresponding device.

I/O byte

Bit 5	Bit 4	
Ø	Ø	(TTY): Does nothing.
Ø	1	(PTP): Outputs to the LCD (operates in the same way as CONOUT).
* 1	Ø	(UP1): Outputs to the RS-232C interface (operates in the same way as RSOUT). PUNCH waits until DSR and TxRDY are set to 1 indicating that the counterpart receiver is ready for reception.

1   1   (UP2): Does nothing.

        *: Default setting.

Entry Name	READER	Entry Address	WBOOT + 12H
Function	Inputs one character from the reader device.		
Entry parameter	None.		
Return parameter	A = input data.		
Explanation			

READER checks the RDR: field (bits 3 and 2) of the I/O byte and reads one character from the corresponding device. When no input data is present, READER waits until data is received.

I/O byte

Bit 3	Bit 2		
0	0	(TTY):	Reads from the keyboard (operates in the same way as CONIN).
0	1	(PTP):	Does nothing.
1	0	(UP1):	Reads from the RS-232C interface (operates in the same way as RSIN).
1	1	(UP2):	Does nothing.

In OS ASCII versions B and later, READER always returns 1AH (EOF) when the PTR or UR2 is selected.

Entry Name	HOME	Entry Address	WBOOT + 15H
Function	Positions the disk head to track 00.		
Entry parameter	None.		
Return parameter	None.		
Explanation			

HOME writes the write data left in the FDD buffer into the floppy disk and moves the disk head to track 00.

Entry Name	SELDSK	Entry Address	WBOOT + 18H
Function	Specifies the drive.		

Entry parameter	C = logical drive No. 00H = A: --> 08H = I:
	 Bit 0 of the E reg. indicates whether the drive
	 is to be accessed for the first time after WBOOT.
	    Bit 0 = 0: The first access after WBOOT.
	    Bit 0 = 1: Not the first access after WBOOT.		

Return parameter	HL = 0000H: Parameter error.
	 HL ⧸ 0000H: Normal termination.
	    HL contains the DPE (disk parameter header)
	    address of the physical drive corresponding
	    to the logical drive.		

Explanation			

Entry parameters 00H through 08H correspond to the logical drives
A: through I:, respectively. Since the correspondence between
the logical drives A: through G: and the actual physical drives is
not fixed, SELDSK specifies the drive after translating the
logical drive into the physical drive. (See "Changing Drives"
for details about logical and physical drives.)

SELDSK sets or resets bit Ø of the E reg. to indicate whether the drive is to be accessed for the first time. When bit Ø = Ø, SELDKS takes the following actions according to the selected physical drive:

1. RAM DISK   (Default logical drive is A:.)

   Does nothing.

2. ROM capsule   (Default logical drives are B: and C:.)

1) Turns the ROM capsule power on.

2) Checks whether the 2 bytes of the ROM header contains ØE5H and 37H to determine whether ROM is actually installed and whether the ROM is for ROM capsules. A parameter error is signaled if an error occurs.

3. FDD   (Default logical drives are D:, E:, F:, and G:.)

1) Opens the serial port for communication and turns the drive power on.

2) If the write buffer has been already loaded with write data, SELDSK writes the data onto the FD.

   Example: ON the TF-2Ø which contains two drives, if drive E: is specified when the preceding write data for drive D: is only placed in the buffer but not actually written on the FD, SELDSK flushes out the buffer before designating drive E:.

3) Otherwise, SELDKS issues the RESET command to the FDD.

   Once the FDD buffer is cleared through operation 2) or 3), the

FDD can be used with the newly specified drive designation.
A parameter error will be reported if an error occurs during
the above processing; e.g., the serial port cannot be opened
or the RESET command is terminated abnormally (no FDD is
installed or no floppy disk is inserted).

4. Microcassette drive   (Default logical drive is H:.)

Does nothing.

5. ROM capsule in the extended unit   (Default logical drive is
   I:.)

1) Checks whether the extended unit is installed.

2) Checks whether ROM is installed in the ROM capsule in the
   extended unit and whether the 2 bytes of the header are ØE5H
   and 37H which identify the ROM for ROM capsules.
   A parameter error will be signaled if an error occurs during
   operation 1) or 2).

Entry Name	SETTRK	Entry Address	WBOOT + 1BH
Function	Specifies the track for read or write.		
Entry parameter	BC = track No.		
Return parameter	None.		
Explanation			

The following track numbers can be specified depending on the drive type:

Physical drive	Logical drive	Track No.
RAM DISK	A:	0 - 2: Internal RAM disk 0 - 7: 60K RAM disk unit 0 - 7: 64K RAM disk unit 0 - 14: 120K RAM disk unit 0 - 15: 128K RAM disk unit
ROM capsule	B: C:	0 - 7
FDD	D: E:	0 - 39

	F:	
	G:	
MCT	H:	0 - 4
ROM capsule in extended unit	I:	0 - 15

Since SETTRK makes no entry parameter check, it reports no error even if a truck number outside the valid range is specified. An error will be reported when an actual read or write operation is performed.

Entry Name	SETSEC	Entry Address	WBOOT + 1EH
Function	Specifies the sector for subsequent read or write.		
Entry parameter	BC = sector No. (0 - 63)		
Return parameter	None.		
Explanation			

Valid sector numbers are 0 through 63. Although SETSEC does not check the entry parameter, an error will be signaled when an actual read or write is performed if a sector number beyond that range is specified.

Entry Name	SETDMA	Entry Address	WBOOT + 21H
Function	Specifies the DMA starting address for read or write.		
Entry parameter	BC = DMA starting address.		
Return parameter	None.		
Explanation			

SETDMA specifies the starting address of the area to be used as the memory buffer during read or write. Data is read from or written onto the drive in 128 byte (1 sector) units.

Entry Name	READ	Entry Address	WBOOT + 24H
Function	Reads the specified sector.		
Entry parameter	None.		
Return parameter	A = 00H: Normal termination. A ≠ 00H: Abnormal termination.		
Explanation			

READ reads the sector specified by SELDSK, SETTRK, and SETSEC and stores the contents in the 128 byte area starting at the address specified by SETDMA.

If the drive is FDD (D:, E:, F:, G:), one of the following codes is returned when an error occurred:

FAH: Read error.

FBH: Write error.   Only 0FAH or 0FCH is returned

FCH: Select error.   by READ.

FDH: Read only disk.

FEH: Read only file.

An error will be generated if a READ is executed for MCT (H:).
Use MIOS subroutines for MCT.

Entry Name	WRITE	Entry Address	WBOOT + 27H
Function	Writes the data to the specified sector.		

Entry parameter	C = Specifies how to write.
	
	    ØØH: Write standard format data (write after blocking).
	
	    Ø1H: Write unblocked data (write immediately without blocking).
	
	    Ø2H: Write to a sequential file.		

Return parameter	A = ØØH: Normal termination. A ≠ ØØH: Abnormal termination.		
Explanation			

WRITE writes the data from the 128 byte area starting at the address specified by SETDMA into the sector specified by SETTRK and SETSEC.

If the drive is FDD (D:, E:, F:, G:), one of the following codes is returned when an error occurred:

      FAH: Read error.

      FBH: Write error.   Only ØFBH, ØFCH, ØFDH, or ØFEH is

      FCH: Select error.   returned by WRITE.

      FDH: Read only disk.

      FEH: Read only file.

An error will be generated if a WRITE is specified for a drive other than RAM disk (A:) and FDD (D:, E:, F:, G:). Use MIOS subroutines for MCT.

Entry Name	LISTST	Entry Address	WBOOT + 2AH
Function	Returns the status of the list device.		
Entry parameter	None.		
Return parameter	A = FFH: Ready (sending data on the list device is allowed). A = 00H: Busy (sending data on the list device is disallowed).		
Explanation			

LISTST checks the LST: field (bits 7 and 6) of the I/O byte and returns the status of the corresponding device.

I/O byte

      Bit 7   Bit 6

      Ø   Ø   (TTY): Checks the serial port.

                                    ØFFH: Control In is high.

                                      ØØH: Control In is low.

      Ø   1   (CRT): Returns FFH because the device is always set to LCD.

      1   Ø   (LPT): Checks the RS-232C interface.

ØFFH: DSR is high.

ØØH: DSR is low.

1   1   (UL1): Always returns ØFFH if no

actual device is defined.

Entry Name	SECTRN	Entry Address	WBOOT + 2DH
Function	Translates a logical sector to a physical sector.		
Entry parameter	BC = Logical sector number.		
Return parameter	HL = Physical sector number.		
Explanation			

Actually, SECTRN performs no actual translation but returns the physical sector number identical to the logical sector number. This function is originally provided to perform skew processing to increase FD performance. Therefore, physical to logical sector translation is not necessary for drives other than FDD.

For FDD, SECTRN need not translate sector numbers because the FDD connected to MAPLE is intelligent to perform logical to physical sector translation.

Entry Name	PSET	Entry Address	WBOOT + 30H
Function	Converts graphics screen data for display.		

Entry parameter	B = Data to be converted.
	
	C = Function.
	
	  01H: AND, 02H: OR, 03H: XOR
	
	  In other cases, PSET loads the C reg. with the
	
	  data at the address specified by HL.
	
	HL = Graphics screen address of the data to be
	
	  converted. (0 - 3839)		

Return parameter	A = 00H: Normal termination.
	
	  = FFH: Screen is in character mode.
	
	  = Others: HL contains an address other than
	
	      graphics screen addresses (0 - 3839).
	
	C = Loaded with the operation result upon normal
	
	  termination.		

Explanation			

PSET processes the 1 byte data at the address specified by HL and

data in the B reg. on the graphics screen according to the data

in the C reg., then places the result to the C reg. An error

is reported in the following conditions:

- When the screen is not in graphics mode.
- When HL is loaded with an address other than the graphics
  screen addresses (0 - 3839).

PSET only loads the C reg. with the data at the specified address
on the graphics screen when the C reg. contains other than 01H,
02H, and 03H.

Each byte on the graphics screen is assigned an address as shown
below:
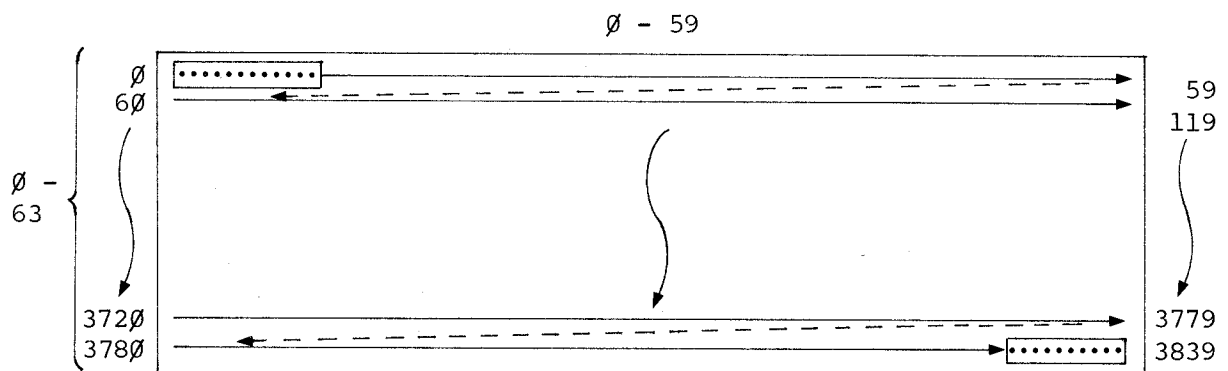
Entry Name	SCRNDUMP	Entry Address	WBOOT + 33H
Function	Takes a hard copy of the displayed data.		
Entry parameter	None.		
Return parameter	LSTERR (F69EH) = ØØH: Normal termination. = ØFFH: Terminated with CTRL/STOP key.		
Explanation			

SCRNDUMP checks the I/O byte and dumps (outputs) the current data on the LCD screen onto the device (serial, RS-232C) specified in the LST: field. However, it does nothing if the LST: field is set to CRT (LCD).

The dump operation can be terminated any number of times by pressing the CTRL/STOP key. LSTERR indicates whether the operation was terminated with the CTRL/STOP key.

SCRNDUMP sends the display data to the serial port or RS-232C interface as characters when character mode is selected. It checks the sixth DIP switch and converts special codes to spaces
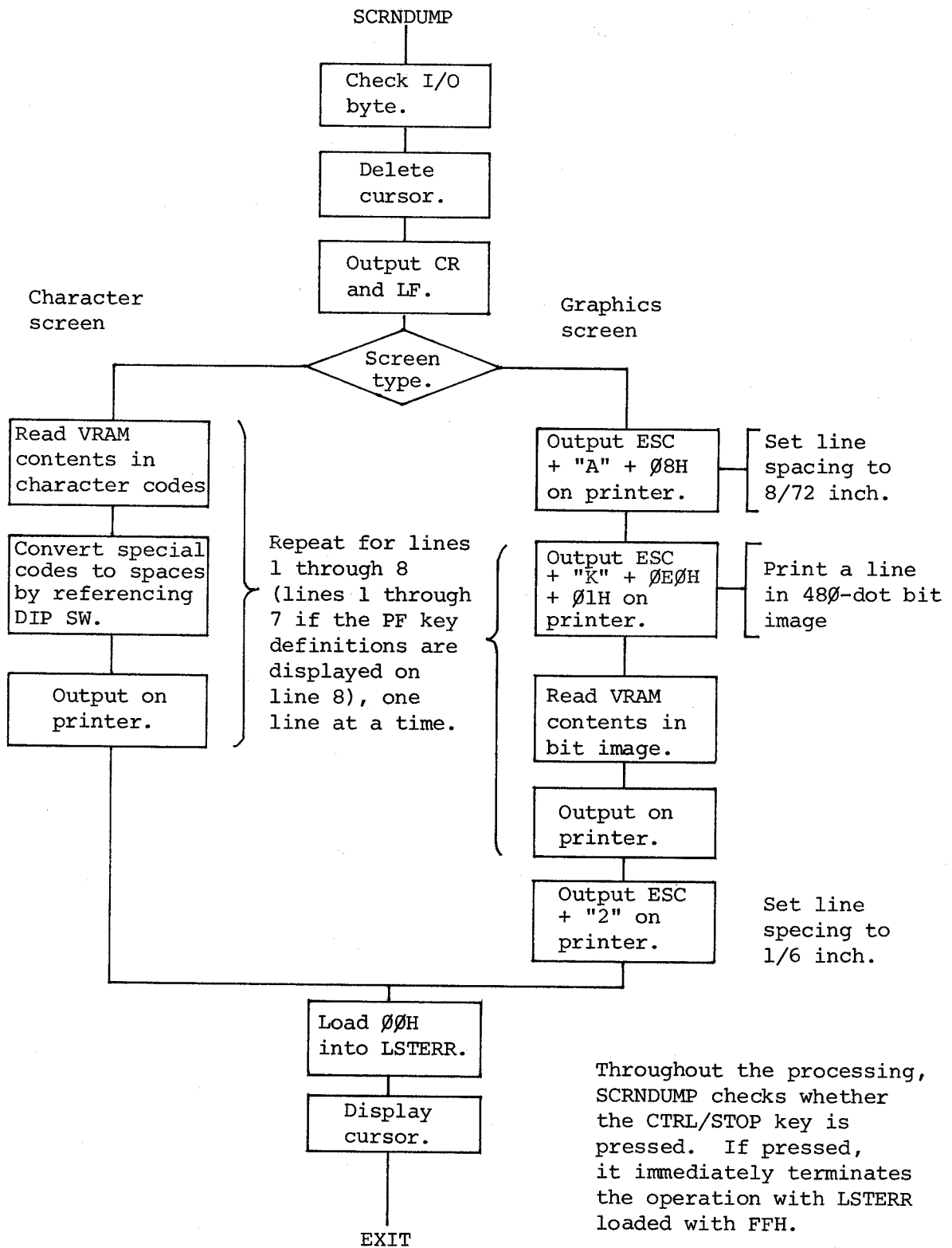
4-37

before output.

Sixth DIP switch

0: Converts 00H - 1FH, 7FH, and 0FFH to spaces.

1: Converts 00H - 1FH and 7FH - 0FFH to spaces.

The display data is output to the serial port or RS-232C interface in bit image when graphics mode is selected.

In either mode, seven lines from the top are output on the printer if the PF key definitions are displayed on line 8.

SCRNDUMP

```
┌─────────────┐
│ Check I/O   │
│ byte.       │
└─────────────┘

┌─────────────┐
│ Delete      │
│ cursor.     │
└─────────────┘

┌─────────────┐
│ Output CR   │
│ and LF.     │
└─────────────┘
```

Character screen ◄── Screen type. ──► Graphics screen

```
┌──────────────────┐
│ Read VRAM        │
│ contents in      │
│ character codes  │
└──────────────────┘

┌──────────────────┐
│ Convert special  │
│ codes to spaces  │
│ by referencing   │
│ DIP SW.          │
└──────────────────┘

┌──────────────────┐
│ Output on        │
│ printer.         │
└──────────────────┘
```

Repeat for lines 1 through 8 (lines 1 through 7 if the PF key definitions are displayed on line 8), one line at a time.

```
┌──────────────────┐        Set line
│ Output ESC       │        spacing to
│ + "A" + Ø8H      │ ────   8/72 inch.
│ on printer.      │
└──────────────────┘

┌──────────────────┐        Print a line
│ Output ESC       │        in 48Ø-dot bit
│ + "K" + ØEØH     │ ────   image
│ + Ø1H on         │
│ printer.         │
└──────────────────┘

┌──────────────────┐
│ Read VRAM        │
│ contents in      │
│ bit image.       │
└──────────────────┘

┌──────────────────┐
│ Output on        │
│ printer.         │
└──────────────────┘

┌──────────────────┐        Set line
│ Output ESC       │        specing to
│ + "2" on         │        1/6 inch.
│ printer.         │
└──────────────────┘
```

```
┌─────────────┐
│ Load ØØH    │
│ into LSTERR.│
└─────────────┘

┌─────────────┐
│ Display     │
│ cursor.     │
└─────────────┘
```

EXIT

Throughout the processing, SCRNDUMP checks whether the CTRL/STOP key is pressed. If pressed, it immediately terminates the operation with LSTERR loaded with FFH.

4-39

Entry Name	BEEP	Entry Address	WBOOT + 36H
Function	Sounds the speaker.		
Entry parameter	C = Specifies the duration of a beep in 100 ms units. BEEP does nothing if C = 0. DE = Specifies the period in 3.2 us units. $$\text{Frequency} = \frac{1}{3.2 \times (DE)} \times 106 \text{ Hz}$$		
Return parameter	None.		
Explanation			

BEEP generates a beep sound in the period specified by DE with the duration of time specified by C.

The processing can be terminated any number of times by pressing the CTRL/STOP key or turning the power switch off.

BEEP can be used as a 100 ms software timer because it waits for the length of time specified by the C reg. without generating sound if DE = 0000H.

Entry Name	RSOPEN	Entry Address	WBOOT + 39H
Function	Opens the RS-232C interface.		
Entry parameter	None.		
Return parameter	A = 00H: Normal termination. = 02H: Already open. = 04H: An invalid specification was found in the conditions set by CONFIG for RS-232C. This error causes no problem as long as CONFIG specifies the conditions for RS-232C but may cause a problem if the work area has been updated directly by the application program.		

Explanation

RSOPEN initializes the RS-232C interface based on the conditions set by CONFIG, turns RS-232C power on, enables RS-232C receive interrupts (8251 interrupts) for RS-232C communication.

RSOPEN must be executed before executing the following routines:

RSIN

RSINST

RSOUTST

RSOUT

Entry Name	RSCLOSE	Entry Address	WBOOT + 3CH
Function	Closes the RS-232C interface.		
Entry parameter	None.		
Return parameter	None.		
Explanation			

RSCLOSE turns RS-232C power off and disables RS-232C receive interrupts.

Entry Name	RSINST	Entry Address	WBOOT + 3FH
Function	Informs whether the RS-232C interface has received data.		
Entry parameter	None.		
Return parameter	See below.		
Explanation			

The status at termination is as follows:

1) Z flag = 1: Normal termination.

    A = FFH: Received data present.

    A = 00H: No received data present.

    BC = Number of received data bytes in the buffer.

2) Z flag = 00H: Abnormal termination.

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

Entry Name	RSOUTST	Entry Address	WBOOT + 42H
Function	Checks whether the RS-232C interface is ready for transmission.		
Entry parameter	None.		

Return parameter	A = 00H: Transmission disabled. (Z flag = 1)
	  = FFH: Transmission enabled. (Z flag = 1)
	  = 03H: RS-232C is not open. (Z flag = 0)		

Explanation			

The RS-232C interface is enabled for transmission when the

following two conditions are met:

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.

Entry Name	RSIN	Entry Address	WBOOT + 45H
Function	Receives one character from RS-232C.		
Entry parameter	None.		
Return parameter	Z flag = 1: Normal termination. A = Received data. Z flag = 0: Abnormal termination. A = 03H: RS-232C is not open. A = 04H: CTRL/STOP key is pressed.		
Explanation			

When no data is present at the RS-232C interface, RSIN waits
until data is received. Processing can be terminated by pressing
CTRL/STOP key.

If XON/XOFF control is specified, RSIN sends an XON when the
number of the received bytes in the buffer has reduced down to
1/4 of the buffer capacity after it sent an XOFF.
When SI/SO is specified, RSIN performs SI/SO processing on the
received data.

As explained above, XON/XOFF and SI/SO codes are processed by the
operating system and not returned to the application program as
data bytes.

Entry Name	RSOUT	Entry Address	WBOOT + 48H
Function	Transfers one character to RS-232C.		
Entry parameter	None.		
Return parameter	Z flag = 1: Normal termination. Z flag = 0: Abnormal termination. A = 03H: RS-232C is not open. A = 04H: CTRL/STOP key was pressed.		
Explanation			

RSOUT checks whether the RS-232C interface is enabled for output (conditions are the same as with RSOUTST) and, if it is disabled, waits until the interface is ready for transmission. Processing can be terminated by pressing the CTRL/STOP key.

RSOUT sends an SI or SO code before sending the pertinent data byte if SI/SO control is specified.

Entry Name	TIMDAT	Entry Address	WBOOT + 4BH
Function	Performs clock and alarm functions.		
Entry parameter	Described below.		
Return parameter	Described below.		
Explanation			

TIMDAT provides the following six functions:

1. Reads the time. (C = 00H)

2. Sets the time. (C = 0FFH)

3. Enables the alarm/wake function. (C = 80H)

4. Disables the alarm/wake function. (C = 81H)

5. Sets the alarm/wake time. (C = 82H)

6. Reads the alarm/wake time. (C = 84H)

The calling program must call TIMDAT after loading the C reg. with the code of the function to be performed and the D reg. with the starting address of the packet (time descriptor) for transferring time-related data. TIMDAT will do nothing if the C reg. is loaded with a code other than the above codes.

TIMDAT assumes the following clock specifications:

- Maximum time count is 23:59:59 12/31/1999.

- Leap year processing is performed automatically.

- The time is represented in the 24-hour system.

- The day of the week is not set automatically but updated when the day changes.

Time descriptor structure

The time descriptor consists of 11 bytes as shown below. Not all bytes are necessarily used by a function.

(DE)

№		
①	Loaded with the lowest two digits of the year in BCD code.	1 byte
②	Loaded with the month in BCD code.	1 byte
③	Loaded with the day in BCD code.	1 byte
④	Loaded with the hour in BCD code.	1 byte
⑤	Loaded with the minute in BCD code.	1 byte
⑥	Loaded with the second in BCD code.	1 byte
⑦	Loaded with the day of the week.	1 byte
⑧	Loaded with the alarm/wake type.	1 byte
⑨	Loaded with the address.	2 bytes
⑩	Loaded with the status.	1 byte

(1) - (6): Year, month, day, hour, minute, second

The time data 1984, 09, 14, 15, 53, 28 is loaded as follows:

84H, 09H, 14H, 15H, 53H, 28H

(1)   (2)   (3)   (4)   (5)   (6)

(7): Day of the week

00H, 01H, 02H, 03H, 04H, 05H, 06H

SUN. MON. TUE. WED. THU. FRI. SAT.

(8): Type

Specifies the alarm/wake type.

00H --- No specification.

01H --- Sets the alarm.

(Displays an alarm message at the specified time.)

02H --- Specifies wake1.

(Performs the function identified by the string at
the address specified in (9) at the specified
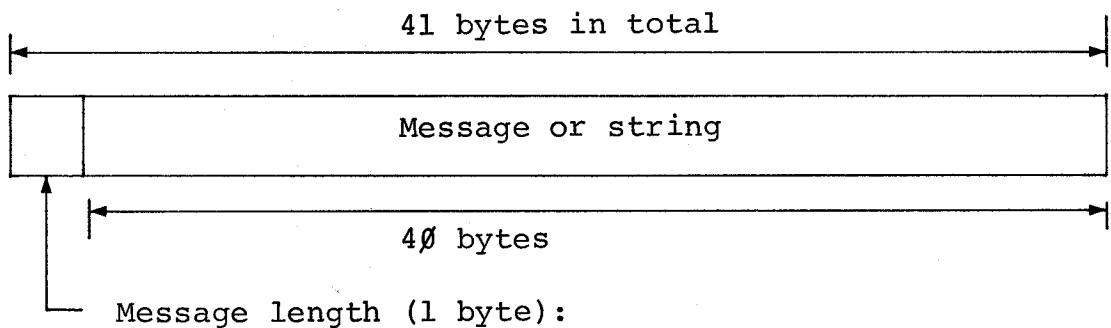time.)

03H --- Specifies wake2.

(Executes the subroutine at the address specified
in (9) at the specified time.)

(9): Address

The meaning of the address differs depending on the type specified in (8).

Type   Meaning

01H ----- Starting address of the alarm message.

02H ----- Starting address of the string identifying the function to be executed during wake1.

03H ----- Starting address of the subroutine (processing) to be executed during wake2.

The alarm message and wake1 string must be defined in the following format:

```
                    41 bytes in total
├──────────────────────────────────────────────┤
┌───┬────────────────────────────────────────────┐
│   │            Message or string                │
└───┴────────────────────────────────────────────┘
    │                40 bytes
    └─ Message length (1 byte):
```

Specify the actual message text or string length in binary from ØØH to 28H. ØØH indicates no message or null string.

(10): Status

Identifies the alarm/wake interrupt type.

Interrupt type   Status value

Alarm/wake time is specified. ------- 00H

(via BIOS TIMDAT).

Alarm/wake interrupt is generated. -- 01H

Alarm/wake time is read ------------- Set to 00H after

(via BIOS TIMDAT).   the current status

                                       is returned.

TIMDAT returns 01H only when it has read an alarm/wake time
for the first time after an alarm/wake interrupt occurs.
TIMDAT continues to return 00H whenever called until the
next interrupt occurs.