

3.8.2.2 Programming notes

The following points must be taken into consideration when reassign logical drives to physical drives:

- (1) Do not specify values 0BH through 0FFH as physical drive codes. Any drive assigned with one of these values cannot be selected.
- (2) The 11 bytes in DISKTBL must contain unique values.
- (3) When rewriting the DISKTBL table, also rewrite DISKROV (0F10CH), a vector identifying the R/O state of the drives. The reassignments are made valid when a warm boot is performed.
- (4) When rewriting the DISKTBL table, also rewrite the areas in the table that hold the addresses of ROM capsules 1 and 2 in (ROMCPNO1 at 0F10AH and ROMCPNO2 at 0F10BH).
- (5) Do not change the assignment of drive H:.

DISKROV (0F10CH) 2 bytes
- Disk R/O vector

	7	6	5	4	3	2	1	0	
+00H	H	G	F	E	D	C	B	A	Each bit
									= 0 : R/W
	7	6	5	4	3	2	1	0	= 1 : R/O
+01H	0	0	0	0	0	K	J	I	

The bits in the disk R/O vector are associated with logical drives A: through K: as shown above. Logical drives B:, C:, J:, and K: are initially set to R/O (06H, 06H).

ROMCPNO1 (0F10AH) 1 byte
- Drive code for ROM capsule 1
Loaded with the offset of ROM capsule 1 with respect to the beginning of DISKTBL. The initial value is 01H.

ROMCPNO2 (0F10BH) 1 byte
- Drive code for ROM capsule 2
Loaded with the offset of ROM capsule 2 with respect to the beginning of DISKTBL. The initial value is 02H.

CHANGE DISK TRANSLATE TABLE

NOTE : This sample program is changing disk
translate table.
Calling sequence.
SAMPLE12 (Logical):=(Physical)
(ex. SAMPLE12 A:=B)

<> assemble condition <>

.Z80

<> loading address <>

.PHASE 100H

<> constant values <>

BIOS entry

EB03	WBOOT	EQU	0EB03H	; Warm Boot entry
EB0C	CONOUT	EQU	WBOOT +09H	; Console out entry

System area

F0FF	DISKTBL	EQU	0F0FFH	; Top of User BIOS
F10C	DISKROV	EQU	0F10CH	; REIOS1 loading addr
F10A	ROMCPNOI	EQU	0F10AH	; REDOS1 loading addr

0080	INDATA	EQU	00080H	
------	--------	-----	--------	--

.start

000D	CR	EQU	0DH	
000A	LF	EQU	0AH	

MAIN PROGRAM

NOTE : This program is changing disk translate table.
First parameter is Logical drive.
Second parameter is Physical drive.

0100		LD	SP,1000H	; Set stack pointer.
0100	31 1000			
0103	CD 016E	CALL	GETDATA	; Get user input data.
0106	DA 0114	JP	C,ERROR_END	; If parameter end, then end.
0109	CD 019B	CALL	SETMSG	; Set return message data.
010C	CD 011D	CALL	CHANGE	; Change disk translate table.
010F	21 026E	LD	HL,MSG01	; Normal end message.
0112	18 03	JR	PEND	
0114		LD	HL,MSG02	; Error end message.
0114	21 0294	CALL	DSPMSG	; Display message.
0117	CD 0262	JP	WBOOT	
011A	C3 EB03			

CHANGE DISK DRIVE TABLE

NOTE :
<> entry parameter <>
B : Logical drive NO.
C : Physical drive NO.
<> return parameter <>
NON
<> preserved registers <>
NON

CAUTION :
If parameter error, then WBOOT.

011D		LD	E,B	; Get target logical drive NO. address.
011D	58	LD	D,00H	; DISKTBL + BC --> HL
011E	16 00	LD	HL,DISKTBL	
0120	21 F0FF	ADD	HL,DE	
0123	19	LD	(HL),C	; Set new physical drive No.
0124	71			
0125	1C	INC	E	; Calculate read only bit position.
0126	43	LD	B,E	; Drive No. (1 to 11)
0127	5A	LD	E,D	; DE is 0000H.
0128	37	SCF		; Carry on.
0129		RL	E	; Shift left by 1 bit.
0129	CB 13	RL	D	
012B	CB 12	DJNZ	LOOP10	; Loop by Drive No.
012D	10 FA			
012F	21 F10C	LD	HL,DISKROV	; Disk R/O table.
0132	79	LD	A,C	; If R/O disk (1,2,9,10)
0133	FE 01	CP	01H	; then set read only bit.

```

0135 28 0C
0137 FE 02
0139 28 08
013B FE 09
013D 28 04
013F FE 0A
0141 20 09

0143
0143 7E
0144 B3
0145 77
0146 23
0147 7E
0148 B2
0149 77
014A 18 0B

014C
014C 3E FF
014E AB
014F A6
0150 77
0151 23
0152 3E FF
0154 AA
0155 A6
0156 77

0157
0157 79
0158 B7
0159 C6
015A FE 03
015C D0

015D 06 0B
015F 21 F0FF
0162 CD 0252
0165 D8

0166 06 00
0168 21 F109
016B 09
016C 77

016D C9

```

```

JR Z,RO_DISK
CP 02H
JR Z,RO_DISK
CP 09H
JR Z,RO_DISK
CP 0AH
JR NZ,RW_DISK

RO_DISK:
LD A,(HL) ; Set R/O bit.
OR E
LD (HL),A
INC HL
LD A,(HL)
OR D
LD (HL),A
JR ROM_CAPSEL

RW_DISK:
LD A,OFFH ; Reset read only bit.
XOR E
AND (HL)
LD (HL),A
INC HL
LD A,OFFH
XOR D
AND (HL)
LD (HL),A

ROM_CAPSEL:
LD A,C ; Key code. (Disk code)
OR A ; RAM disk.
RET Z ; Yes.
CP 03H ; ROM capsel 1 or 2?
RET NC ; No.

LD B,11 ; Table length.
HL,DISKTBL ; Table address.
CALL SEARCH ; Search data.
RET C ; Not found.

LD B,00H ; Set position No. in table.
LD HL,ROMCPNO1-1
ADD HL,BC
LD (HL),A

RET

```

```

*****
GET PARAMETER FROM BUFFER
*****

```

NOTE : Get disk translate data from CCP input buffer.

```

<> entry parameter <>
NON
<> return parameter <>
CY-flag : Return parameter.
=0 -- Normal return
=1 -- Error return
(PParameter error)
Case of Normal return
B : Logical drive NO.
(O0H to 0AH)
C : Physical drive NO.
(O0H to 0AH)

```

```

<> preserved registers <>
NON

```

CAUTION : Case of normal return, this routine sets finishing message.

```

016E
016E 21 0080
0171 7E
0172 FE 05
0174 D8

0175 23
0176 23
0177 7E
0178 E6 DF
017A D8 41
017C D8
017D FE 0B
017F 3F
0180 D8

0181 47

0182 23
0183 7E
0184 FE 3A
0186 37
0187 C0
0188 23
0189 7E
018A FE 3D
018C 37
018D C0

018E 23
018F 7E

```

```

GETDATA:
LD HL,INDATA ; Parameter inputted address.
LD A,(HL) ; Get inputted byte number.
CP 05H ; Parameter count check.
RET C ; Error, then return with carry on.

INC HL ; Pointer update
INC HL
LD A,(HL) ; Get first data.
AND 0DFH ; Capital change.
SUB 'A' ; 'A' to 'K' check.
RET C ; If error,
CP 'L'-'A' ; then return with carry on.
CCF
RET C

LD B,A ; First parameter --> B

INC HL ; Pointer update.
LD A,(HL) ; '=' check.
CP '='
SCF
RET NZ
INC HL
LD A,(HL)
CP '='
SCF
RET NZ

INC HL ; Pointer update.
LD A,(HL) ; Get second parameter.

```

```

0190 E6 DF
0192 D6 41
0194 D8
0195 FE 0B
0197 3F
0198 D8

0199 4F
019A C9

```

```

AND ODFH ;
SUB 'A' ; 'A' to 'K' check.
RET C ;
CP 'L','A' ;
CCF ;
RET C ;

LD C,A ; Second parameter --> C
RET ;

```

```

*****
SET DRIVE NAME
*****

```

NOTE : Set drive name to message area.

```

<> entry parameter <>
    B : Drive No.
    C : Table No.
<> return parameter <>
    NON
<> preserved registers <>
    BC

```

CAUTION :

```

019B
019B C5
019C 78
019D C6 41
019F 32 027B

```

SETMSG;

```

PUSH BC ; Save register.
LD A,B ;
ADD A,'A' ; Get ASCII data.
LD (DRIVE),A ; Set drive code.

```

```

01A2 21 01C1
01A5 79
01A6 87
01A7 5F
01A8 16 00
01AA 19
01AB 5E
01AC 23
01AD 56

```

```

LD HL,DEVICETBL ; Data search table.
LD A,C ;
ADD A,A ;
LD E,A ;
LD D,00H ;
ADD HL,DE ; Get target address in table. (HL)
LD E,(HL) ;
INC HL ;
LD D,(HL) ;

```

```

01AE EB
01AF 11 0281
01B2 4E
01B3 23
01B4 06 00
01B6 ED B0
01B8 21 024E
01BB 0E 04
01BD ED B0

```

```

EX DE,HL ; Message top address. (HL)
LD DE,DEVICE ; Data setting address.
LD C,(HL) ; Get data count.
INC HL ;
LD B,00H ;
LDIR ;
LD HL,ENDMARK ; Set end data.
LD C,04H ;
LDIR ;

```

```

01BF C1
01C0 C9

```

```

POP BC ; Restore register.
RET ;

```

Message table & data

```

01C1
01C1 01D7
01C3 01E0
01C5 01ED
01C7 01FA
01C9 0200
01CB 0206
01CD 020C
01CF 0212
01D1 0220
01D3 022E
01D5 023E

```

DEVICETBL:

```

DW DV_A ; RAM disk
DW DV_B ; ROM capsul 1
DW DV_C ; ROM capsul 2
DW DV_D ; FDD 1
DW DV_E ; FDD 2
DW DV_F ; FDD 3
DW DV_G ; FDD 4
DW DV_H ; MCT cartridge
DW DV_I ; RAM cartridge
DW DV_J ; ROM cartridge 1
DW DV_K ; ROM cartridge 2

```

```

01D7
01D7 08 52 41 4D
01DB 20 64 69 73
01DF 6B
01E0
01E0 0C 52 4F 4D
01E4 20 63 61 70
01E8 73 65 6C 20
01EC 31
01ED
01ED 0C 52 4F 4D
01F1 20 63 61 70
01F5 73 65 6C 20
01F9 32
01FA
01FA 05 46 44 44
01FE 20 31
0200
0200 05 46 44 44
0204 20 32
0206
0206 05 46 44 44
020A 20 33
020C
020C 05 46 44 44
0210 20 34
0212
0212 0D 4D 43 54
0216 20 63 61 72
021A 74 72 69 64
021E 67 65
0220
0220 0D 52 41 4D
0224 20 63 61 72
0228 74 72 69 64
022C 67 65

```

```

DV_A: DB 8,'RAM disk'
DV_B: DB 12,'ROM capsul 1'
DV_C: DB 12,'ROM capsul 2'
DV_D: DB 5,'FDD 1'
DV_E: DB 5,'FDD 2'
DV_F: DB 5,'FDD 3'
DV_G: DB 5,'FDD 4'
DV_H: DB 13,'MCT cartridge'
DV_I: DB 13,'RAM cartridge'

```

```

022E
022E 0F 52 4F 4D
0232 20 63 61 72
0236 74 72 69 64
023A 67 65 20 31
023E
023E 0D 52 4F 4D
0242 20 63 61 72
0246 74 72 69 64
024A 67 65 20 32

024E
024E 2E 0D 0A 00

```

```

DV_J: DB 15,'ROM cartridge 1'

DV_K: DB 13,'ROM cartridge 2'

```

```

ENDMARK: DB ' ',CR,LF,00H

```

```

*****
SEARCH TABLE
*****

```

NOTE :

```

<> entry parameter <>
    A : Key code
    B : Length of table
    HL : Top of table address
<> return parameter <>
    CY : Return information
        =0 : Found
        =1 : Not found
    A : Key No.
    HL : Object item position in table
<> preserved registers <>
    BC,DE

```

CAUTION :

```

0252
0252 C5
0253 0E 00
0255
0255 BE
0256 28 07
0258 23
0259 0C
025A 10 F9
025C 0E FF
025E 37
025F
025F 79
0260 C1
0261 C9

```

```

SEARCH: PUSH BC ; Save register.
        LD C,00H ; Initialize key NO.

SEAR1: CP (HL) ; Compare key with item.
        JR Z,SEAR2 ; If found, then exit.
        INC HL ; else check next item.
        INC C ; Key NO. update.
        DJNZ SEAR1 ; If not end of table, then repeat,
        LD C,OFFH ; else set not found sign.
        SCF

SEAR2: LD A,C ; Set key NO.
        POP BC ; Restore register.
        RET

```

```

*****
DISPLAY MESSAGE
*****

```

NOTE :

```

Display message until find 00H.

<> entry parameter <>
    HL : Message data top address.
<> return parameter <>
    NON
<> preserved registers <>
    NON

```

CAUTION :

```

0262
0262 4E
0263 0C
0264 0D
0265 C8

0266 E5
0267 CD EBOC
026A E1
026B 23
026C 18 F4

```

```

DSPMSG: LD C,(HL) ; Get display data.
        INC C ; Data is 00H?
        DEC C ;
        RET Z ; Yes.

        PUSH HL ;
        CALL CONOUT ; Display data.
        POP HL ;
        INC HL ; Pointer update.
        JR DSPMSG ;

```

Message and Work area

```

026E
026E 43 68 61 6E
0272 67 65 20 64
0276 72 69 76 65
027A 20
027B
027C 3A 20 74 6F
0280 20
0281

0294
0294 50 61 72 61
0298 6D 65 74 65
029C 72 20 65 72
02A0 72 6F 72 21
02A4 0D 0A 00

```

```

MSG01: DB 'Change drive '

DRIVE: DS 1
        DB ': to '

DEVICE: DS 19

MSG02: DB 'Parameter error!',CR,LF,00H

```

END

3.8.3 Disk Details

3.8.3.1 ROM capsule

(1) General

A ROM capsule is made up of a ROM capsule carrier that carries ROM chips. The PINE main unit can house two ROM capsules. Data in these capsules, once loaded into PINE memory, can be accessed immediately through bank switching.

The ROM chips for the PINE ROM capsules may be 28-pin DIP CMOS mask ROM chips or CMOS EPROM chips that are compatible with 2764, 27128, or 27256.

ROM capsules are used to store application programs for the PINE. The application programs in ROM capsules may be executed as are or loaded into RAM for execution.

A ROM capsule can contain M-format (format for PX-8 ROM capsule) ROM chips or ROM cartridge chips. (Programs in such ROM are always loaded into RAM for execution or the ROM capsule is used as a data file.)

A ROM capsule has a maximum capacity of 32K bytes. Two ROM capsules may be chained to form a single ROM disk drive.

(2) Drive name and capacity

(a) Drive name

When two ROM capsules are used separately ----- Drives B: and C:
When two ROM capsules are chained ----- Drive B: or C:

(b) Capacity

	2764	27128	27256
Capacity	8 KB	16 KB	32 KB
Track	0	0 - 1	0 - 3
Sector	0 - 63	0 - 63	0 - 63
Maximum			
directory entries	31	31	31

Note: The values in the above table are obtained when the number of directory entries is 28 to 31. If the number of directory entries is less than 28, the number of logical tracks available on each ROM is greater than the above values. For example, the maximum logical track and sector numbers of 27256 are 4 and 6, respectively, if the number of directory entries is 1 to 3. This is because the PINE always assumes 31 logical entries though it can handle an arbitrary number of physical entries (maximum is 31).

(3) Formats

(a) General

Application programs are classified into two groups according to how they are executed by the PINE CP/M.

1. Programs loaded into TPA for execution as under the ordinary CP/M (for all drives) ----- Load-and-go programs
2. Programs in the ROM capsule that are executed immediately in ROM ----- ROM-based programs

The PINE supports the following two ROM formats for the ROM capsule:

1. M format

The M format is used for both PINE ROM capsules and ROM cartridges. Load-and-go programs and ROM file data must be stored in this format.

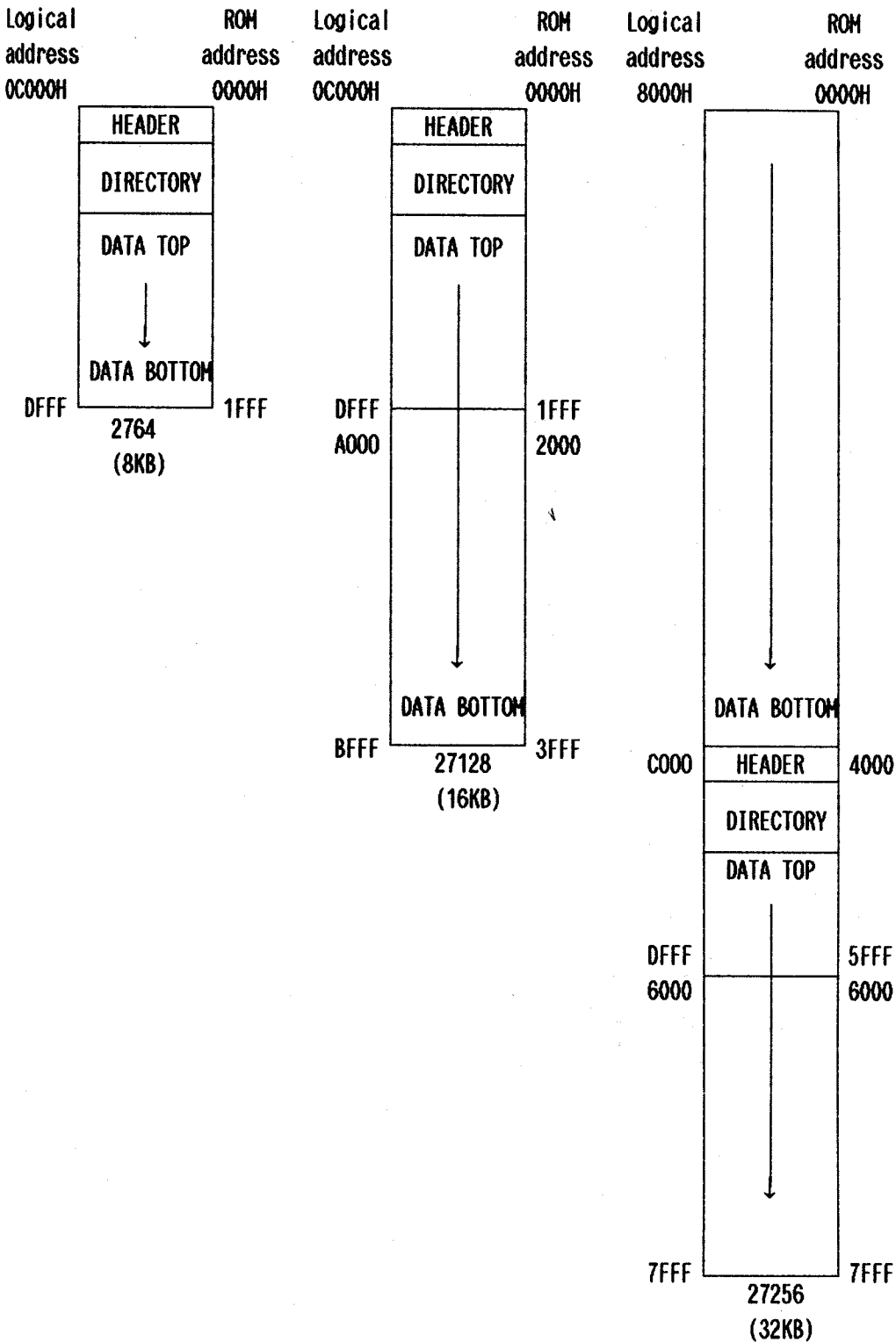
2. P format

The P format is supported only by the PINE. ROM-based programs must be stored in ROM in this format. This format is used only for ROM capsules.

The OS automatically identifies the ROM format and calculates the address of the ROM, so that user need not identify the ROM format by himself.

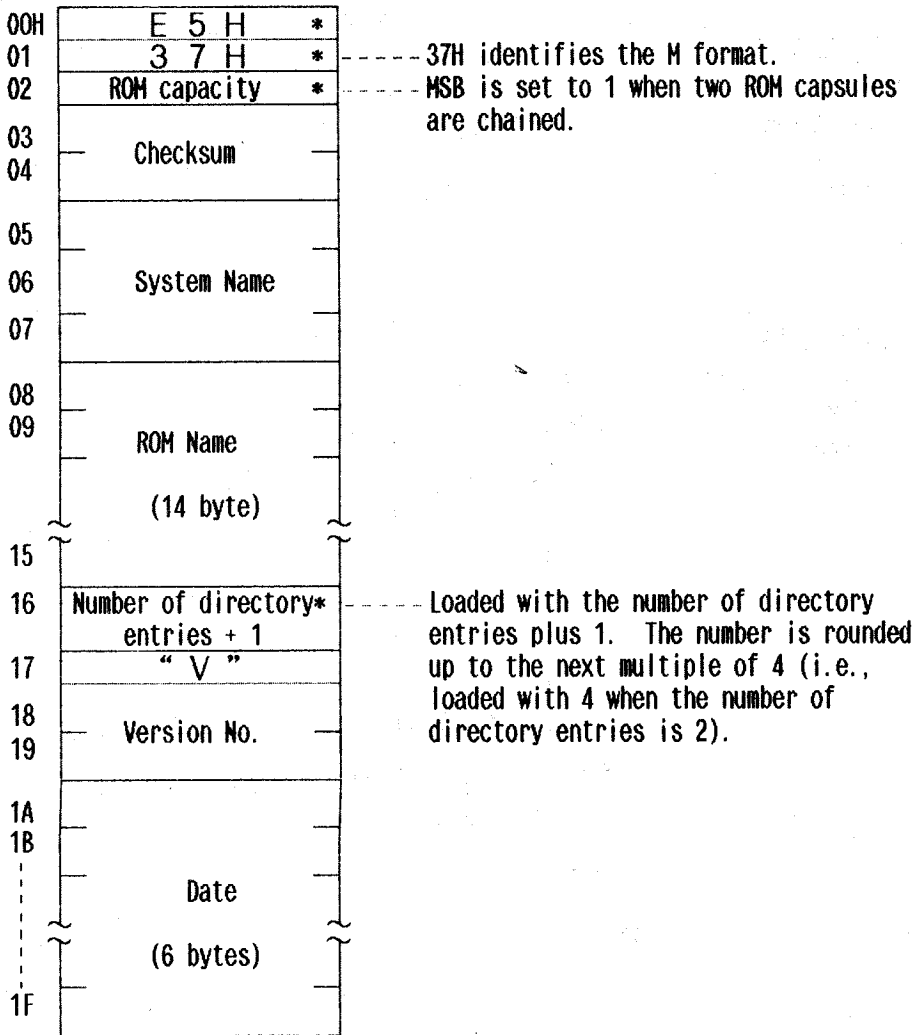
See Section 4.6, "Executing a ROM Program" for ROM-based programs.

(b) M format



1) Header area

The header area has the following format (32 bytes):



The fields marked by * must always be loaded.

2) Directory area

The directory area can contain a maximum of 31 entries. One directory entry is 32 bytes long. Its format is the same as that of the CP/M FCB (File Control Block).

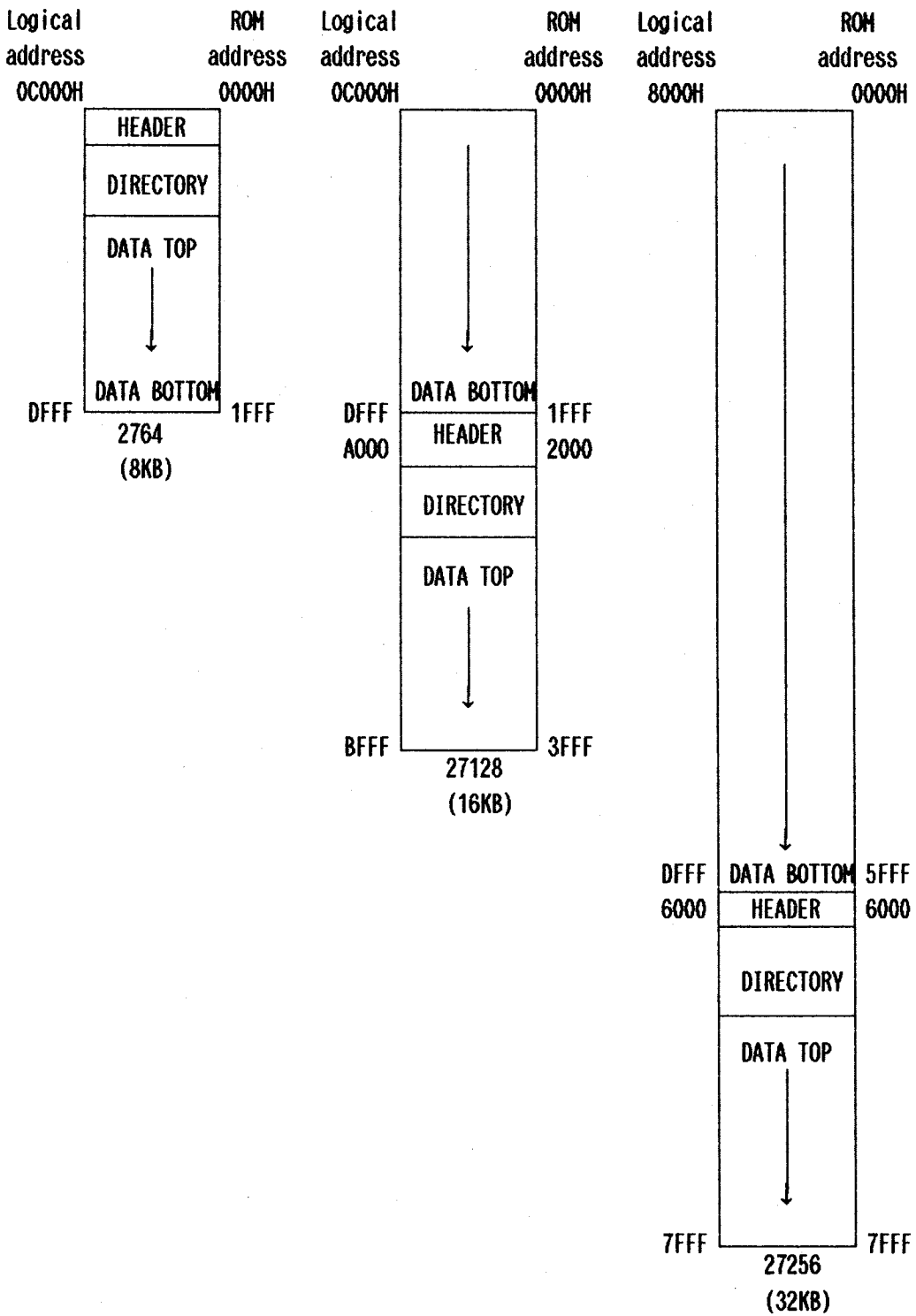
3) Data area

The data area start address varies depending on the number of directory entries. Assuming that the number of entries is n , the data area start address is found by adding $(20H \times m)$ to the header start address, where m is equal to $(n + 1)$ rounded up to the next multiple of 4.

Any accesses to the locations between (header start address + $20H \times m$) and (header start address + $3FFH$) logically return $0E5H$. Logically, therefore, the data area start address is set to (header start address + $400H$).

When two ROM capsules are used to form a single ROM drive (chained ROM capsules), the first ROM area that is left unused ($20H \times m$ bytes) is used as a data area together with the second ROM capsule. The second ROM capsule contains neither header nor directory and consists only of a data area.

(c) P format



1) Header area

The second byte is loaded with 50H when the P format is used. The other fields are the same as in the M format.

2) Directory area

Same as with the M format.

3) Data area

Same as in the M format. ROM-based program stored in this format, however, must have an ID identifying their program type (executed on ROM) at their beginning. See Section 4.6, "Executing a ROM Program" for more information about ROM-based programs.

Load-and-go programs may also be loaded into the P-format ROM.

4) Other

(a) When two ROMs are used in series

When two ROMs are used in series, it is necessary to use two ROMs of the same capacity.

(b) Preparing ROMs

For points to note when preparing ROMs, please refer to 7.3, Guide for Programming ROMs

3.8.3.2 ROM cartridge

(1) General

The PINE ROM cartridge is an optional read-only external storage. It is connected to the PINE main unit via the PINE cartridge interface.

The ROM cartridge can contain two capsule-type, 28-pin DIP CMOS or NMOS-type mask ROM, or EPROM chips compatible with 2764, 27128, or 27256. It can also house PX-8 ROM chips or M format ROM chips which are used as ROM capsules. The ROM cartridge ROM can store only load-and-go programs or data files; it cannot hold ROM-based programs.

A ROM cartridge has a maximum capacity of 64K bytes. Two ROM chips (capsule-type) may be put together for form a single storage drive.

(2) Drive name and capacity

(a) Drive name

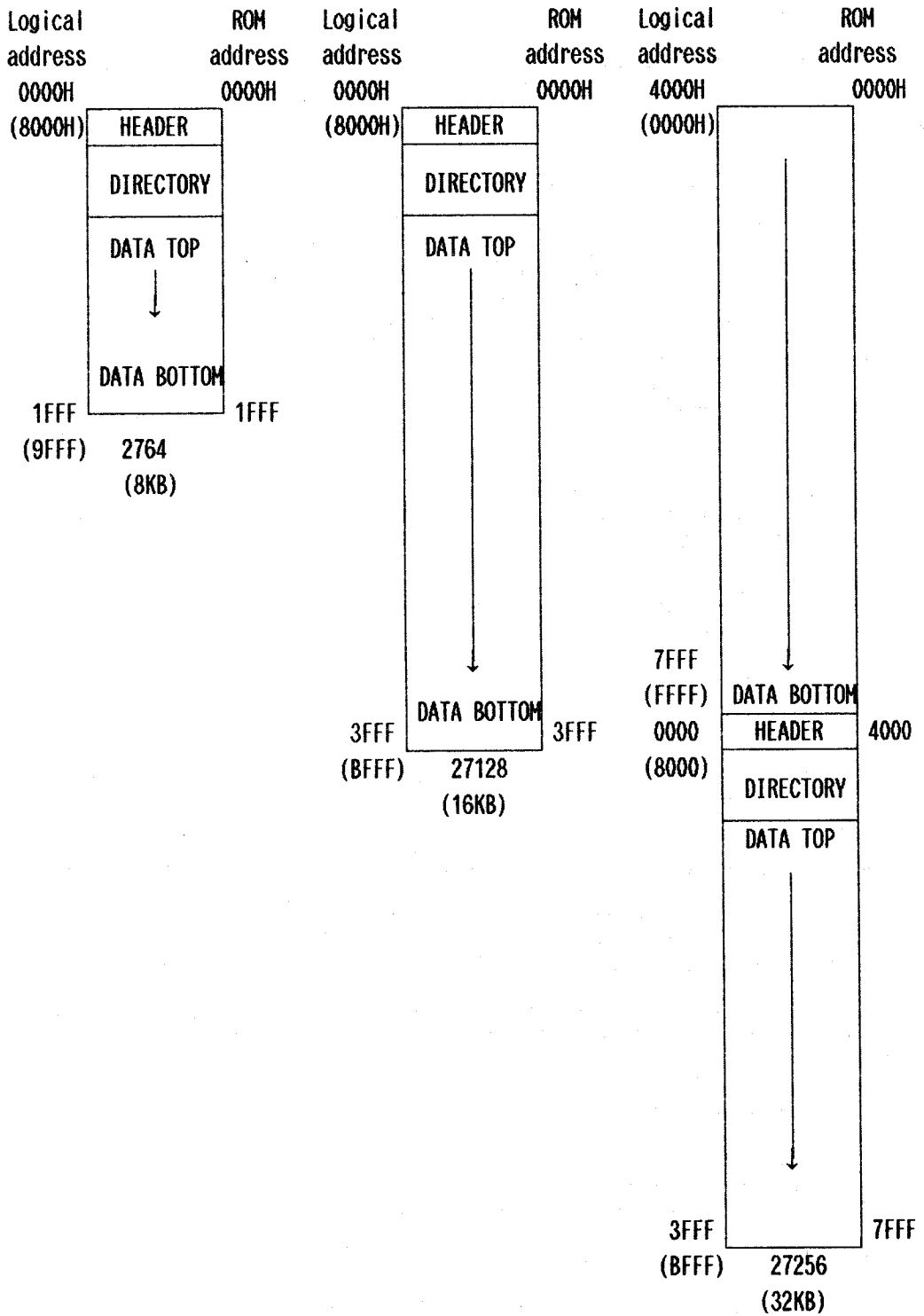
When two ROM chips are used separately ----- Drives J: and K:
When two ROM chips are chained ----- Drive J: or K:

(b) Capacity

Same as with the ROM capsule. (See 3.8.3.1.)

(3) Format

The format of the ROM cartridge ROM is shown in Figure 3.8.1. The formats of the header, directory, and data areas are the same as those of the M format ROM capsule ROM.



Note: Addresses enclosed in parentheses are logical addresses used when two ROM chips are used.

Fig. 3.8.1 ROM Cartridge ROM Format

(4) Miscellaneous Considerations

(a) Power control

The OS controls the power to the ROM cartridge to reduce power consumption. See Section 5.1, "Cartridges" for details.

(b) Notes on use of two ROM chips

The two ROM chips to be chained must have the same capacity.

3.8.3.3 Internal RAM disk

(1) General

The internal RAM disk is allocated in a part of the main RAM (bank 0) and allows the PINE main unit to operate as a CP/M machine by itself.

The internal RAM disk cannot be set up when an external RAM disk is installed. If an external RAM disk is installed, it is always selected when drive A: is specified.

The PINE performs checksum tests on the internal RAM disk to check for inadvertent data destruction.

(2) Drive name and capacity

(a) Drive name

A:

(b) Capacity

1) The internal RAM disk size may be specified in 1K byte units from 0K to 35K bytes (it cannot be specified in 1K-byte size) provided that the total size of the internal RAM disk and the user BIOS does not exceed 35.5K bytes. This restriction is placed to realize a 20K CP/M.

2) The internal RAM disk size can be established during system initialization or by the user using the CONFIG program.

3) The internal RAM disk can be established during system initialization only when no external RAM disk is installed. The default capacity is 26K bytes. It is set to 0K byte when the external RAM disk is installed.

4) Track: 0 - 4

Sector: 0 - 63

Number of directory entries: 16

(3) Format

Figure 3.8.2 shows the internal RAM disk format.

(a) Directory area

The directory area can contain as many as 16 entries with each directory entry being 32 bytes long. The directory area is 512 bytes long and always present.

(b) Checksum area

This area is loaded with the results of checksum tests made in 128-byte blocks. One byte in this area indicates the results of the test on a 128-byte block. Checksum data is updated during writes and checked during reads. The checksum area is fixed at 512 bytes and always reserved.

(c) Data area

The data area starts at (internal RAM disk start address + 1K byte). The size of the data area depends on the internal RAM disk size.

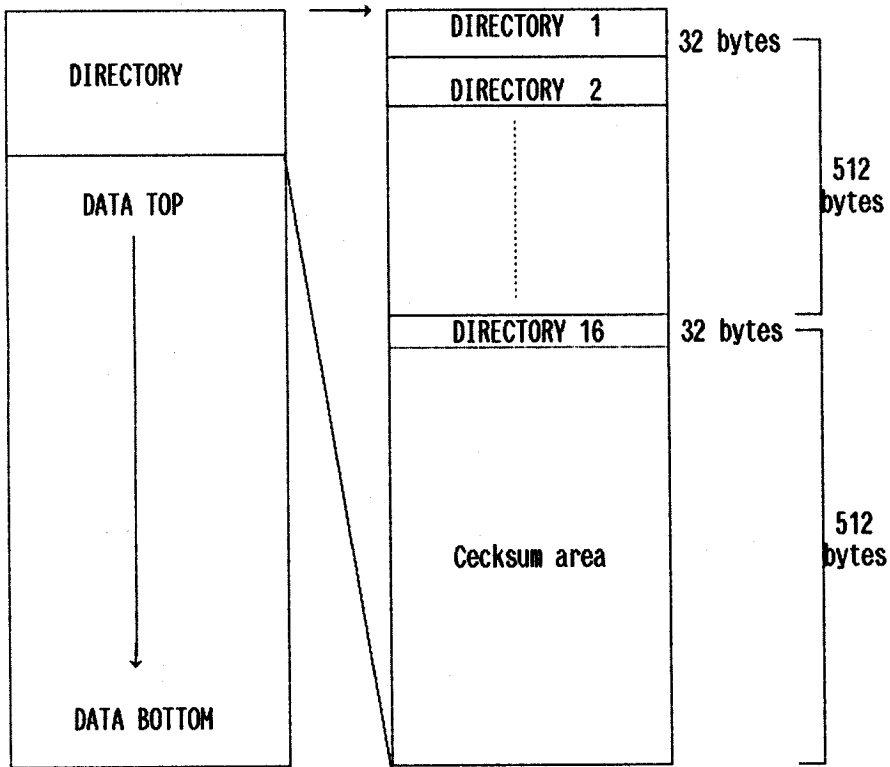


Fig. 3.8.2 Internal RAM Disk Format

(4) Miscellaneous considerations

(a) Since the internal RAM disk, in principle, is assigned to the background bank (bank 0) of the application program, bank switching is required for a ROM-based program to access it directly. When accessing it via CP/M, however, the program need not be aware of bank switching because the OS takes care of it.

(b) The location of the internal RAM disk in memory varies depending on the size of the user BIOS and the internal RAM disk.

3.8.3.4 External RAM disk

(1) General

The nonintelligent-type external RAM disk is an optional drive connected to the PINE system bus. Its functions include automatic update of address registers after memory accesses and close mode operation that decreases power consumption. See Section 5.4, "External RAM Disk" for details.

(2) Drive name and capacity

(a) Drive name

A:

(b) Capacity

Capacity: 128K bytes

Track: 0 - 7

Sector: 0 - 63

Number of directory entries: 32

(3) Format

Figure 3.8.3 shows the external RAM disk format.

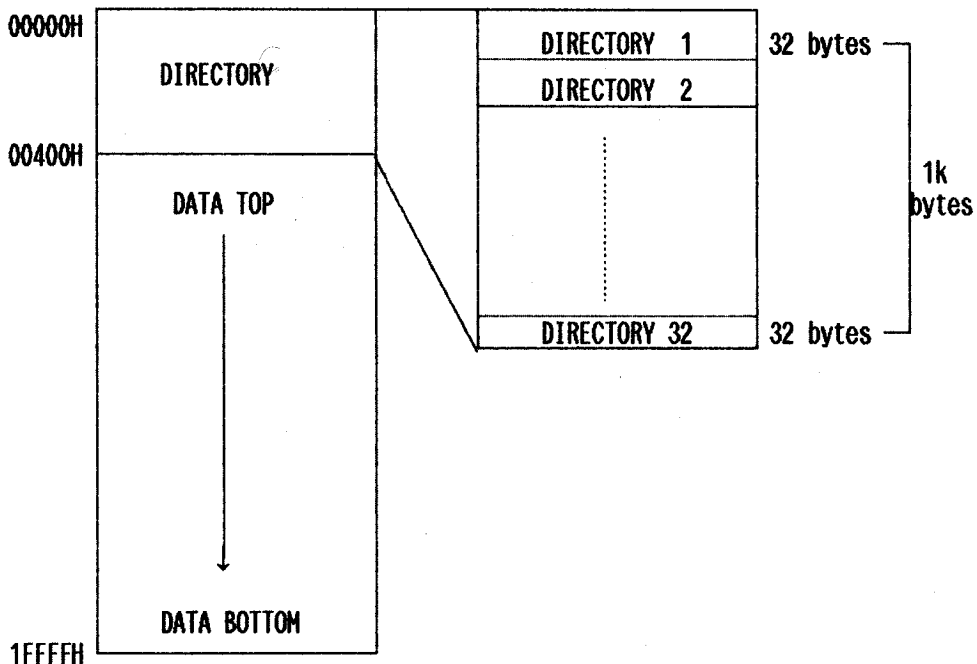


Fig. 3.8.3 External RAM Disk Format

(a) Directory area

The directory area can store a maximum of 32 entries of 32 bytes. This area is 1K bytes long and always reserved.

(b) Data area

The data area starts at address 00100H.

(4) Miscellaneous considerations

(a) When the external RAM disk is established at system initialization time, the OS automatically sets the internal RAM disk size to 0K byte.

(b) The external RAM disk connection status is determined at power-on. The system initialization is performed automatically when the PINE is switched on for the first time after the external RAM disk is connected or disconnected.

(c) Both the internal and external RAM disks are assigned to the same logical drive, i.e., drive A:. The two RAM disks cannot be used at the same time.

(d) The contents of the external RAM disk are not guaranteed if the reset button is pressed during a write to the external RAM disk. The system displays a

RAM DISK FORMAT (Y/N)

message in response to a reset made during a write so that the user can determine whether or not to format the external RAM disk.

3.8.3.5 RAM cartridge

(1) General

The PINE RAM cartridge is an optional external storage that allow both reads and writes. It is connected to the PINE main unit through the cartridge interface.

The RAM cartridge incorporates static RAM chips. Data on the static RAM chips can be accessed through I/O instructions issued from the PINE main unit. The RAM cartridge is battery-backed up and its data is sustained for up to three years even it is held removed from the main unit.

(2) Drive name and capacity

(a) Drive name

I:

(b) Capacity

Capacity: 16K bytes (The OS also supports 32K- and 64K-byte configuration.)

Track: 0 - 1 (for 16K-byte configuration)

0 - 3 (for 32K-byte configuration)

0 - 7 (for 64K-byte configuration)

Sector: 0 - 63

Number of directory entries: 16

(3) Format

The RAM cartridge has the format shown in Figure 3.8.4.

The formats of the directory, checksum, and data areas are the same as those of the internal RAM disk.

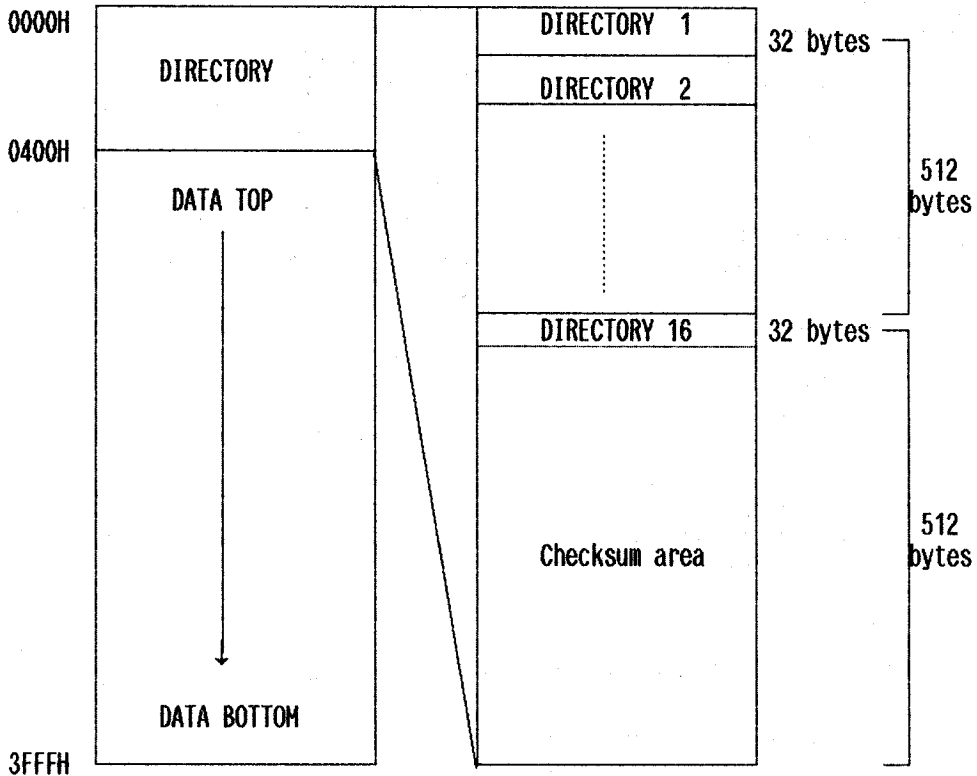


Fig. 3.8.4 RAM Cartridge Format

Note: The end address 7FFFH and 0FFFFH are for 32K-byte and 64K-byte configurations, respectively.

(4) Miscellaneous considerations

(a) The OS automatically identifies the RAM cartridge capacity when the PINE is powered on.

(b) RAM cartridges can be formatted by:

- 1) Selecting the item for RAM cartridge formatting from the system display.
- 2) Calling RAMCREMT in the OS ROM jump table.

3.8.3.6 Microcassette

(1) General

The PINE microcassette is an optional external storage that allows both reads and writes. It is connected to the PINE main unit through the cartridge interface.

Tape data is also directed to the buzzer in the main unit or an external loudspeaker. Tape can be manipulated manually or by using programs. Manual manipulation is via system display. (Manual manipulation is not possible when an item keyboard is installed.) The MIOS routines are used to manipulate tape through programs.

Microcassettes (MCT) are used in almost the same way as disk drives except that they support only sequential files. Each MCT has a directory file at its beginning which enables fast access to files. See Section 3.7, "MTOS/MIOS Operations" for details.

(2) Drive name and capacity

(a) Drive name

H: (Fixed)

(b) Capacity

The capacity of an MCT varies depending on the tape length. The concepts of track and sector are not used for MCTs. The number of directory entries is 12.

(3) Format

See Section 3.7, "MTOS/MIOS Operations" for details.

(4) Miscellaneous considerations

(a) Mount

A mount refers to reading in the tape directory into RAM (RAM directory). Once a mount is carried out, any directory update (caused by file creation, for example) is managed in the RAM directory. A mount operation must be performed whenever a file is accessed. (The OS supports the automatic mount feature.)

(b) Remove

A remove refers to writing the RAM directory into the tape directory and must be performed at the end of file processing. If tape is removed without performing a remove operation, the tape data is not guaranteed. Moreover, the contents of the tape that is mounted next may be destroyed.

(c) Tape mount/remove

The PINE microcassette is provided with a LED for tape mount/remove identification. That LED is turned off when a tape is mounted and turned on when the tape is removed. Tape must be mounted or removed by the user after confirming that the LED is on.

(d) Tape access

Data on an MCT is blocked and deblocked in 256 byte units. The portion of tape from which files are deleted cannot be reused to store new files. Only one file at a time can be accessed for a read or write. See Section 3.7, "MTOS/MIOS Operations" for details.

(e) Additional BDOS routines

The following BDOS (MTOS) routines are provided for microcassette processing in addition to the standard BDOS routines:

- 1) Verify
- 2) Remove
- 3) Mount
- 4) Read tape ID
- 5) Make tape directory

(f) BIOS routine for MCT processing

The BIOS routine provided for MCT processing is MCMTX.

3.8.3.7 External disk (floppy disk)

(1) General

External disks such as 5.25 inch and 3.5 inch floppy and compact disks can be connected to the PINE via a serial interface. Data on external disks are read or written in 1K byte units.

The following disk units are connectable:

- 1) TF-20
- 2) TF-15 (single or dual drives)
- 3) PF-10

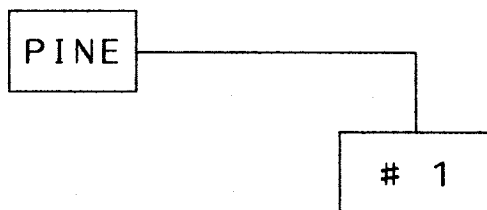
(2) Drive name and capacity

(a) Drive name

Drives D:, E:, F:, and G: are assigned to external disks. The association between a drive and a disk is determined by the DIP switches in that disk drive unit. Refer to manuals related to individual external disks for the procedure to assign a drive to a disk.

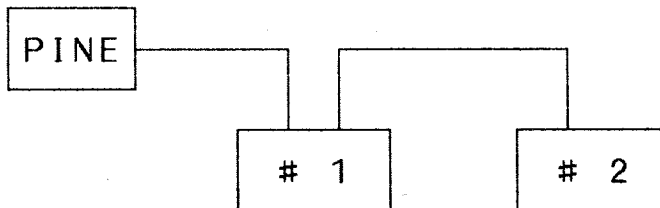
The PINE supports the following drive assignments:

- 1) When connecting an external disk



Model name	Drive name	Capacity
TF-20	D, E	320 KB x 2
TF-15 (single)	D	320 KB x 1
TF-15 (dual)	D, E	320 KB x 2
PF-10	D	320 KB x 1

2) When connecting two external disks



# 1		# 2	
Model name	Drive name	Model name	Drive name
TF-20	D, E	TF-20	F, G
TF-20	D, E	TF-15 (single)	F
TF-20	D, E	TF-15 (dual)	F, G
TF-20	D, E	PF-10	F
TF-15 (single)	D	TF-20	F, G
TF-15 (single)	D	TF-15 (single)	F
TF-15 (single)	D	TF-15 (dual)	F, G
TF-15 (single)	D	PF-10	F
TF-15 (dual)	D, E	TF-20	F, G
TF-15 (dual)	D, E	TF-15 (single)	F
TF-15 (dual)	D, E	TF-15 (dual)	F, G
TF-15 (dual)	D, E	PF-10	F
PF-10	D	TF-20	F, G
PF-10	D	TF-15 (single)	F
PF-10	D	TF-15 (dual)	F, G
PF-10	D	PF-10	F

(b) Capacity

Capacity/drive: 320K bytes
 Number of tracks/drive: 80
 Number of sectors/track: 16
 Storage capacity/sector: 256 bytes
 Number of directory entries: 64
 Storage available for user: 278K bytes

(c) Format

See Section 5.3, "Terminal Floppy."

(4) Miscellaneous considerations

(a) Disk drives and serial communication

A external disk drive is connected to the PINE through a serial interface. Although the PINE supports three types of interfaces, RS-232C, SIO, and cartridge SIO for serial communication, only one of them can be used at a time.

The OS supports automatic switching of the serial port. This function, for example, enables access to a disk while the printer (connected through the RS-232C) is being used.

(b) Communication mode

The communication parameters passed between the PINE and external disks are as follows:

Bit rate: 38,400 bps
Data length: 8 bits
Stop bits: 1 bit
Parity bit: None
Protocol: EPSP (Epson Serial Communication Protocol)
Control line: Not used

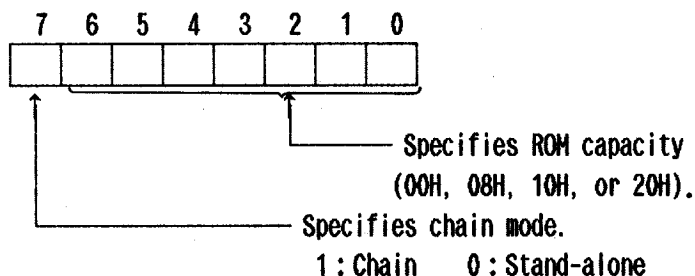
See Section 5.3, "Terminal Floppy" for more information.

(c) Buffer flushing

Disk units TF-20, TF-15 and PF-10 block or deblock data on tape to improve read/write efficiency. This leads to different processing speeds for the PINE and disk units. For example, when the PINE writes a record onto a floppy disk, the record may not be immediately written on the disk. To prevent the write data that is left unprocessed in the main unit buffer from being lost at the end of processing, the PINE OS automatically issues a flush buffer command to the disk unit at every warm boot or power-off.

3.8.4 Disk-related Work Areas

QT_ROM_CP1 (0EF97H) 1 byte
- ROM capsule 1 (drive B:) capacity



QT_ROM_CP2 (0EF98H) 1 byte
- ROM capsule 2 (drive C:) capacity
The format is the same as that of QT_ROM_CP1.

QT_ROM_CR1 (0EF99H) 1 byte
- ROM cartridge 1 (drive J:) capacity
The format is the same as that of QT_ROM_CP1.

QT_ROM_CR2 (0EF9AH) 1 byte
- ROM cartridge 2 (drive K:) capacity
The format is the same as that of QT_ROM_CP1.

QT_RAM_CR (0EF9BH) 1 byte
- RAM cartridge capacity
= 00H: No RAM cartridge installed.
10H: 16K-byte RAM cartridge
20H: 32K-byte RAM cartridge (not yet provided)
40H: 64K-byte RAM cartridge (not yet provided)

QT_RAM_IN (0EF9CH) 1 byte
- Internal RAM disk capacity
00H ≤ QT_RAM_IN ≤ 23H

QT_RAM_EX (0EF9DH) 1 byte
- External RAM disk capacity
= 00H: No external RAM disk installed.
40H: 64K-byte external RAM disk (not yet provided)
80H: 128K-byte external RAM disk

DR_ROM_CP1 (0EF9EH) 1 byte
- Number of directory entries for ROM capsule 1

DR_ROM_CP2 (0EF9FH) 1 byte
- Number of directory entries for ROM capsule 2

DR_ROM_CR1 (0EFA0H) 1 byte
- Number of directory entries for ROM cartridge 1

DR_ROM_CR2 (0EFA1H) 1 byte
- Number of directory entries for ROM cartridge 2

AD_ROM_CP1 (0EFA2H) 2 bytes
- ROM capsule 1 header address

AD_ROM_CP2 (0EFA4H) 2 bytes
- ROM capsule 2 header address

AD_ROM_CR1 (0EFA6H) 2 bytes
- ROM cartridge 1 header address
Always set to 0000H.

AD_ROM_CR2 (0EFA8H) 2 bytes
- ROM cartridge 2 header address
Always set to 0000H.

AD_RAM_IN (0EFAAH) 2 bytes
- Internal RAM disk header address

DISKTBL (0F0FFH) 11 bytes
- Logical/physical drives association table
See 3.8.2, "Logical to Physical Drive Assignment."

ROMCPNO1 (0F10AH) 1 byte
- ROM capsule 1 drive code
See 3.8.2, "Logical to Physical Drive Assignment."

ROMCPNO2 (0F10BH) 1 byte
- ROM capsule 2 drive code
See 3.8.2, "Logical to Physical Drive Assignment."

DISKROV (0F10CH) 2 bytes
- Disk R/O vector
See 3.8.2, "Logical to Physical Drive Assignment."

FTSTAB (0F10EH) 22 bytes
- Jump vector for first disk select
The jump vector used when BIOS SELDSK is called for the first time to select a disk.

READTAB (0F124H) 22 bytes
- Jump vector used during a disk read
Control is passed to the requested read routine through this jump vector during execution of a BIOS READ call.

WRTTAB (0F13AH) 22 bytes
- Jump vector used during a disk write
Control is passed to the requested write routine through this jump vector during execution of a BIOS WRITE call.

DPBASE (0F150H)
- Disk parameter header (DPH)
See a CP/M manual for the format of this header.

DPE0 16 bytes
- DPH for RAM disk

DPE1 16 bytes
- DPH for ROM capsule 1

DPE2 16 bytes
- DPH for ROM capsule 2

DPE3 16 bytes
- DPH for FDD (External floppy disk drive) 1

DPE4 16 bytes
- DPH for FDD 2

DPE5 16 bytes
- DPH for FDD 3

DPE6 16 bytes
- DPH for FDD 4

DPE7 16 bytes (microcassette)
- DPH for MCT

DPE8 16 bytes
- DPH for ROM cartridge 1

DPE9 16 bytes
- DPH for ROM cartridge 2

DPE10 16 bytes
- DPH for RAM cartridge

DPB0 (0F200H) 15 bytes
- DPB (Disk Parameter Block) for RAM disk

DPB1 (0F20FH) 15 bytes
- DPB for ROM capsule 1

DPB2 (0F21EH) 15 bytes
- DPB for ROM capsule 2

DPB3 (0F22DH) 15 bytes
- DPB for FDD

DPB7 (0F2CFH) 15 bytes
- DPB for MCT

DPB8 (0F24BH) 15 bytes
- DPB for ROM cartridge 1

DPB9 (0F25AH) 15 bytes
- DPB for ROM cartridge 2

DPB10 (0F269H) 15 bytes
- DPB for RAM cartridge

DIRBUF (0F5FBH) 128 bytes
- Directory access buffer

ALV0 (0F67BH) 32 bytes: RAM disk allocation area
CSV0 0 byte: RAM disk checksum area

ALV1 (0F69BH) 9 bytes: ROM capsule 1 allocation area
CSV1 0 byte: ROM capsule 1 checksum area

ALV2 (0F6A4H) 9 bytes: ROM capsule 2 allocation area
CSV2 0 byte: ROM capsule 2 checksum area

ALV3 (0F6ADH) 18 bytes: FDD 1 allocation area
CSV3 16 bytes: FDD 1 checksum area

ALV4 (0F6CFH) 18 bytes: FDD 2 allocation area
CSV4 16 bytes: FDD 2 checksum area

ALV5 (0F6F1H) 18 bytes: FDD 3 allocation area
CSV5 16 bytes: FDD 3 checksum area

ALV6 (0F713H) CSV6	18 bytes: FDD 4 allocation area 16 bytes: FDD 4 checksum area
ALV7 (0F735H) CSV7	8 bytes: MCT allocation area 16 bytes: MCT checksum area
ALV8 (0F74DH) CSV8	8 bytes: RAM cartridge allocation area 4 bytes: RAM cartridge checksum area
ALV9 (0F759H) CSV9	9 bytes: ROM cartridge 1 allocation area 0 byte: ROM cartridge 1 checksum area
ALV10 (0F761H) CAV10	9 bytes: ROM cartridge 2 allocation area 0 byte: ROM cartridge 2 checksum area
SYSFCB (0F88DH)	36 bytes: system FCB (File Control Block) area
SYSDMA (0F8B1H)	128 bytes: system DMA (Direct Memory Access) area

3.9 I/O Byte

3.9.1 General

The I/O byte is used to associate physical devices with four logical devices LST:, PUN:, RDR:, and CON:. It is located at address 0003H as with the ordinary CP/M.

BIOS routines making use of the I/O byte are CONST, CONIN, CONOUT, LIST, PUNCH, and READER.

3.9.2 Modifying the I/O Byte

The contents of the I/O byte can be modified by:

- (1) Using STAT.
- (2) Using BDOS.
- (3) Directly rewriting location 0003H.

3.9.3 I/O Byte Contents

Table 3.9.1 shows the bit assignments of the I/O byte.

Logical device	LST :	PUN :	RDR :	CON :	
Bit position	7, 6	5, 4	3, 2	1, 0	
Bit values \ I/O	Output	Output	Output	Output	Input
00	SIO	---	Keyboard	RS-232C	Keyboard
01	*Cartridge	LCD	---	*LCD	*Keyboard
10	RS-232C	*RS-232C	*RS-232C	LCD	RS-232C
11	Centronics	---	---	RS-232C	RS-232C

Table 3.9.1 I/O byte Bit Assignments

(1) The devices identified by * are defaults.

(2) The default device for LST: is determined by the state of DIP switches, bits 5 and 6.

DIP switches

Bits 8 - 7 Not used.
 6 - 5 = 00: SIO
 01: Cartridge
 10: RS-232C
 11: Centronics
 4 - 1 Keyboard selection

See Section 7.2, "Dip Switches" for details.

(3) When the PUN: or RDR: fields are set to the values indicated by "---" in the list

PUN: --- The device is always in the ready state and outputs no data when it is driven.

RDR: --- The device is always in the ready state and returns LAH (CTRL/Z) in response to reque.

(4) The I/O byte is initialized at cold boot time.