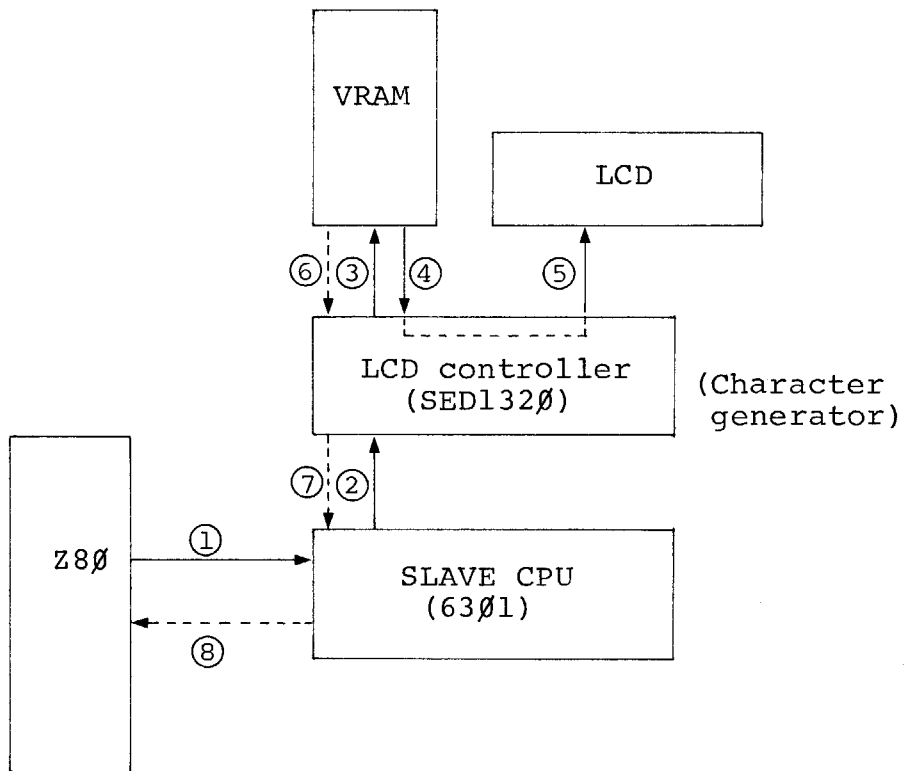# Chapter 6   CONOUT

6.1   Outline

The MAPLE is provided with an LCD of 8 lines by 80
columns as its primary display device.   The MAPLE OS
supports several screen modes to allow the user to make
effective use of this LCD.   This chapter describes the
screen modes in full detail.

6.2 Screen Configuration

The block diagram for the screen hardware and its
peripheral devices is shown on the next page. The
screen operation flow is as follows:

1) The program running on the Z80 CPU sends data to be
displayed to the SLAVE CPU, directly or using the CONOUT
function of BIOS (see Chapter 13, "SLAVE CPU"). (1 in
the diagram on the next page.)
2) The SLAVE CPU loads the data into VRAM via the LCD
controller (SED1320). (2 and 3 in the diagram.)
3) The LCD controller reads the data in VRAM and
displays it on the LCD. (4 and 5 in the diagram.)

```
            ┌──────────┐
            │          │
            │  VRAM    │           ┌──────────────┐
            │          │           │              │
            │          │           │     LCD      │
            └──────────┘           │              │
                                   └──────────────┘
        ⑥ ③   ④          ⑤
            ┌──────────────────────┐
            │   LCD controller     │      (Character
            │     (SED132Ø)        │       generator)
            └──────────────────────┘
                ⑦ ②
┌────────────┐
│            │   ①    ┌──────────────────────┐
│    Z8Ø     │ ─────► │   SLAVE CPU          │
│            │        │     (63Ø1)           │
│            │ ◄───── │                      │
│            │   ⑧    └──────────────────────┘
└────────────┘
```

The contents of VRAM can be read in the flow from 6 to
8.

Character fonts are stored in the LCD controller and
their corresponding codes in VRAM. The LCD controller
reads the codes for specified characters from VRAM (4),
converts them into fonts, and sends them to the LCD (5).

Fonts for external characters are defined at the
beginning of VRAM so, if the code read from VRAM is an
external character, the LCD controller reads the
corresponding font from VRAM and transfers it to the
LCD.

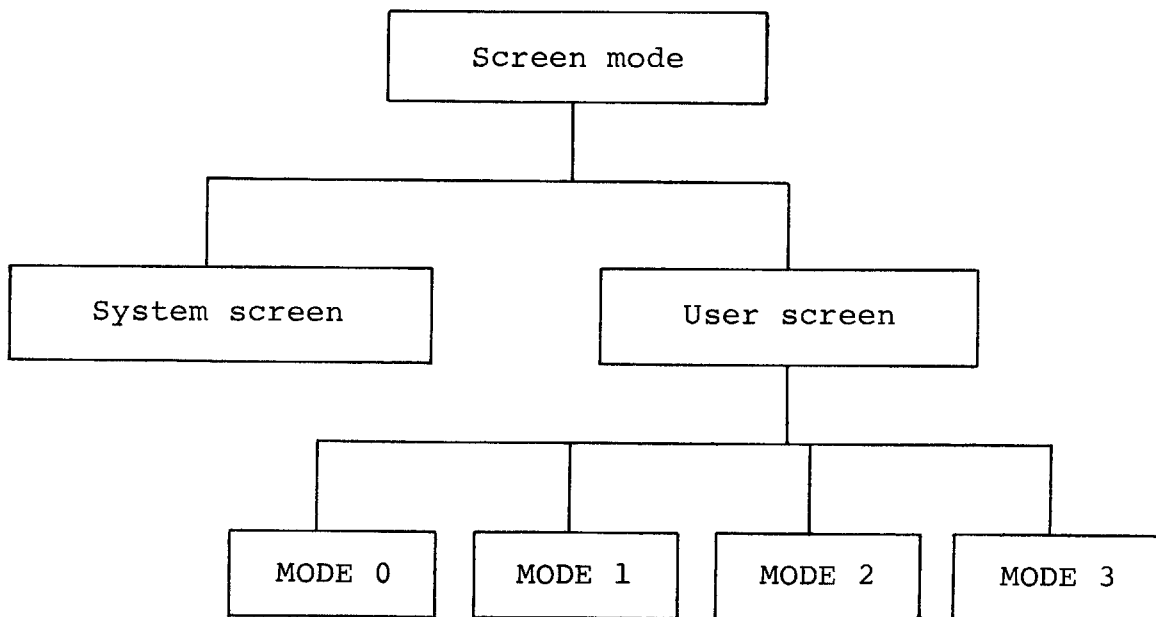See Chapter 13, "SLAVE CPU" for VRAM memory maps.

6-3

6.3 Screen Modes

1) System screen and user screen

MAPLE CP/M supports two types of screens; the system screen and the user screen. These two screens are independent of each other. That is, manipulating one screen does not affect the other screen at all.
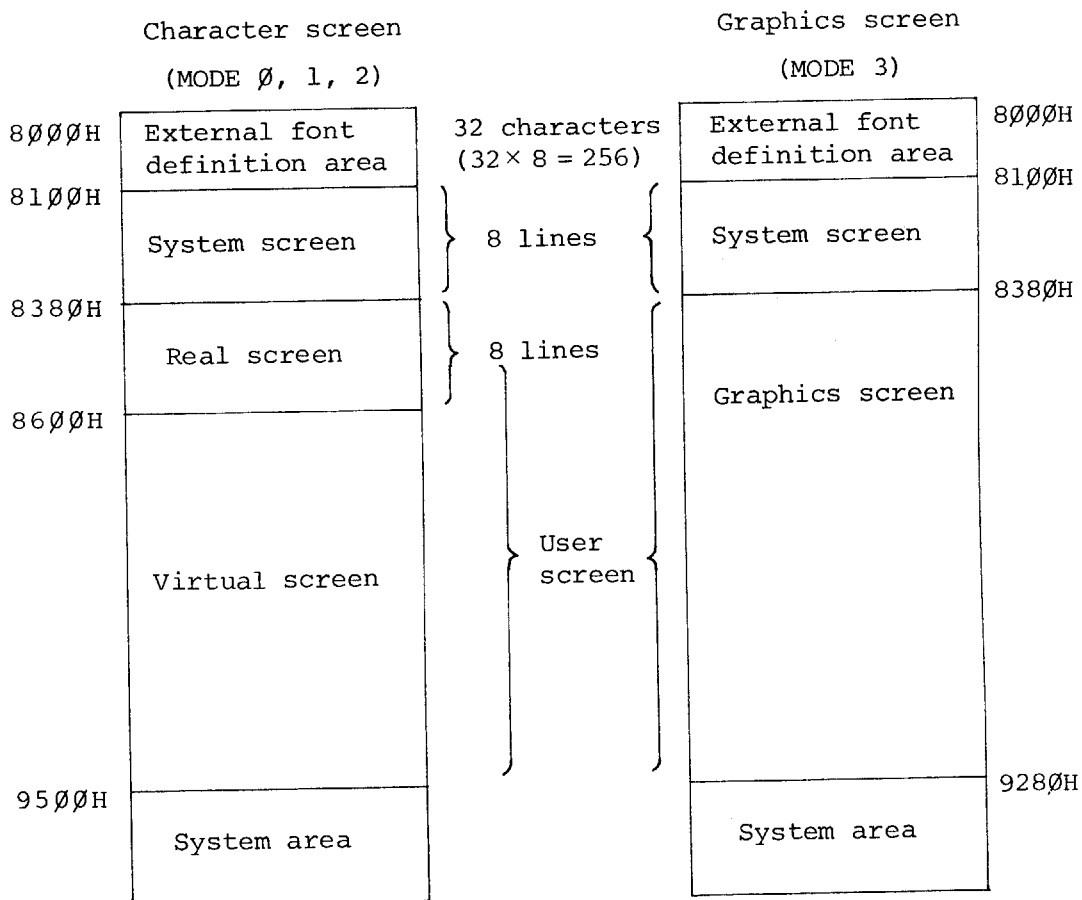
The user screen has four modes. (These modes are detailed later.)

A hierarchical diagram of the screen modes is shown below.

```
                    ┌─────────────────────┐
                    │     Screen mode     │
                    └──────────┬──────────┘
            ┌──────────────────┴──────────────────┐
  ┌─────────┴───────────┐            ┌─────────────┴──────────┐
  │   System screen     │            │      User screen       │
  └─────────────────────┘            └────────────┬───────────┘
                              ┌───────────┬────────┴──┬───────────┐
                        ┌─────┴────┐ ┌────┴─────┐ ┌───┴──────┐ ┌──┴───────┐
                        │ MODE  0  │ │ MODE  1  │ │ MODE  2  │ │ MODE  3  │
                        └──────────┘ └──────────┘ └──────────┘ └──────────┘
```

2) Relationship to VRAM

The screens are located in VRAM as shown below:

```
        Character screen                          Graphics screen
        (MODE Ø, 1, 2)                               (MODE 3)

8ØØØH ┌─────────────────────┐  32 characters  ┌─────────────────────┐ 8ØØØH
      │ External font       │  (32 × 8 = 256) │ External font       │
      │ definition area     │                 │ definition area     │
8100H ├─────────────────────┤                 ├─────────────────────┤ 8100H
      │                     │  ⎫              │                     │
      │ System screen       │  ⎬ 8 lines      │ System screen       │
      │                     │  ⎭              │                     │
8380H ├─────────────────────┤                 ├─────────────────────┤ 8380H
      │                     │  ⎫              │                     │
      │ Real screen         │  ⎬ 8 lines      │                     │
      │                     │  ⎭              │ Graphics screen     │
8600H ├─────────────────────┤                 │                     │
      │                     │                 │                     │
      │                     │    ⎫            │                     │
      │ Virtual screen      │    ⎬ User        │                     │
      │                     │    ⎭ screen      │                     │
      │                     │                 │                     │
9500H ├─────────────────────┤                 ├─────────────────────┤ 9280H
      │ System area         │                 │ System area         │
      └─────────────────────┘                 └─────────────────────┘
```

3) System screen

The system screen consists of 8 lines of 80 columns and
is used by the OS to display:

- System Display

- Alarm/wake message

- Password entry prompt message

- "CHARGE BATTERY"

Usually, no application programs normally can send data
onto the system screen.

4) User screen concept

The user screen is divided into three screens: real, virtual, and window screens.

(1) Real screen

The MAPLE LCD display can display 8 lines by 80 columns of data which makes up the real screen. The real screen consists of a window screen and a 1-line function key display area. The function key display area is optional. When no function key definitions are displayed, the real screen size equals the window screen size. From now on, the number of lines of the window screen is represented by h where h is 7 or 8.

(2) Window screen

The window screen is included in the real screen and made up of h lines of 80 columns. The window screen works like a window through which h lines of the virtual screen can be viewed. This window screen can scroll up and down over the virtual screen.

```
        ┌──────────────────────────────────────────┐
  ↑     │                                          │
  │     │                                          │
  │     │              Window screen               │
Real    │                                          │
screen  │                                          │
  │     ├──────────────────────────────────────────┤
  ↓     │      Function key display area (optional)│
        └──────────────────────────────────────────┘
```
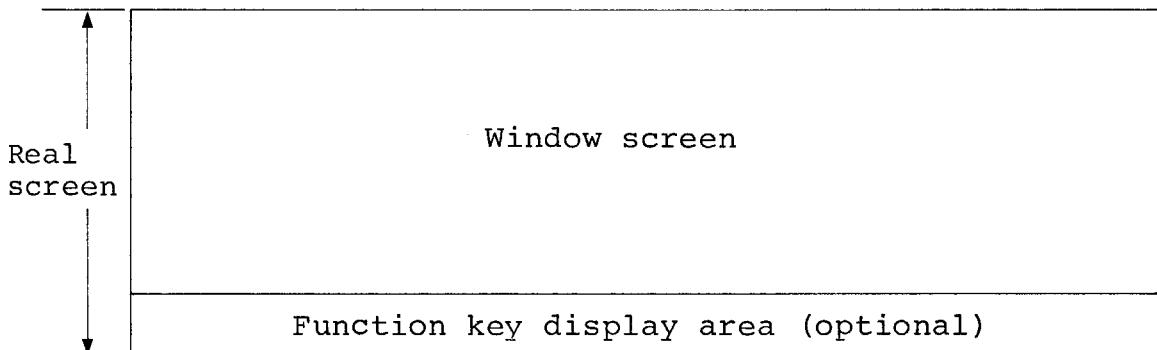
(3) Virtual screen

Although the MAPLE has a large (8 lines of 80 columns)
display screen for this type of display, it has
implemented the concept of virtual screen to meet the
needs of the applications programs which require
larger screens. The MAPLE provides two virtual
screens for the application programs. These screens
may be used for different purposes and displayed
alternately, e.g., displaying data on one screen while
writing display data onto the other screen. The two
screens may be displayed concurrently in some modes.
The sizes of the virtual screens are determined by the
screen mode and user specification.

The entire contents of a virtual screen can be viewed
by scrolling the window screen up and down over the
virtual screen. The window screen scrolls only
vertically.

Virtual screen

Window screen
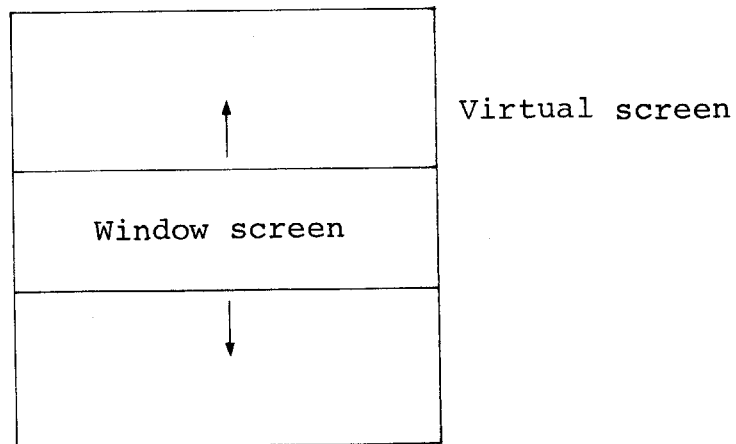
5) User screen modes

There are four user screen modes numbered from MODE 0 to
MODE 3. MODE 0 to MODE 2 are character screen modes in
which only characters can be displayed. MODE 3 is a
graphics screen mode in which both graphics and
characters can be displayed. Switching of screen modes
can be easily performed using the BIOS call CONFIG or
CONOUT.

a) MODE 0 (80-column mode)

Two 80-column wide virtual screens are available in this
mode. Their sizes may be defined as desired, as long as
one screen consists of at least eight lines and their
total number of lines do not exceed 48. The window
screen is always located on one of these virtual
screens.

Screen RAM structure in MODE 0

The window screen is always on one of the virtual screen. The contents of the window screen are transferred to the real screen for display on the LCD.

80 columns

System screen — 8 lines
Real screen — 8 lines
Virtual screen 1
Window screen (8 lines) — n1 lines
Virtual screen 2
Window screen (8 lines) — n2 lines

LCD ⟸ or

$h \leqq n1 \leqq 40$
$h \leqq n2 \leqq 40$

$n1 + n2 \leqq 48$
$(h = 7 \text{ or } 8)$
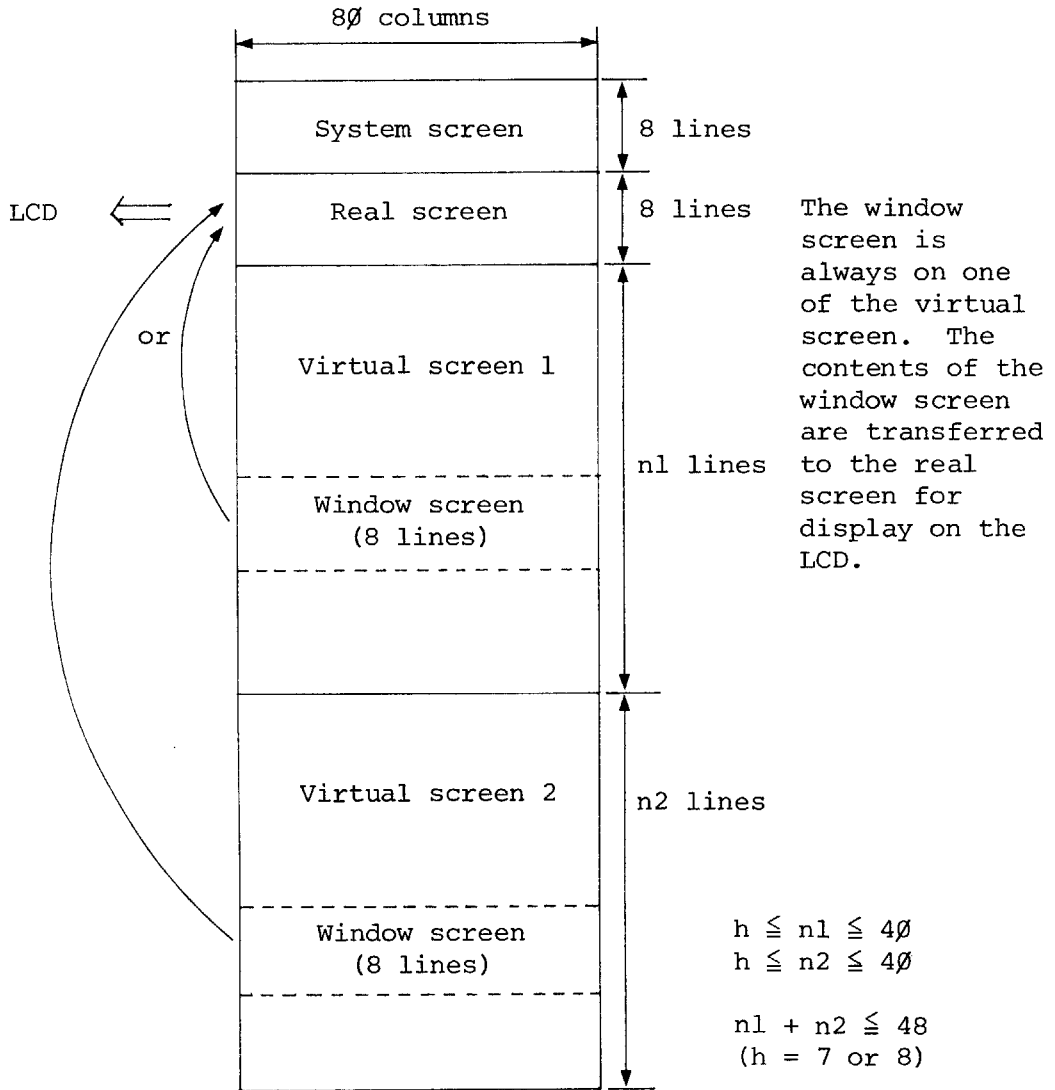
b) MODE 1 (39-column mode)

In this mode, either of the two virtual screens is 39 columns wide. The window screen (WS) has 2h lines. The first h lines of data is displayed on the left half of the real screen and the second h lines of data on the right half of the real screen.

The WS rests on one of the two virtual screens and can scroll up and down as required.

The two virtual screens can accommodate the same number

of lines in the range $2h \leq n \leq 48$.

Screen RAM structure in MODE 1

80 columns

System screen   8 lines

LCD

WS1   WS2   8 lines

Real screen

39 columns   2 columns   39 columns

Virtual screen 1   Virtual screen 2

WS 1 (h lines)

WS 2 (h lines)

2h lines

n lines

$2h \leq n \leq 48$
(h = 7 or 8)
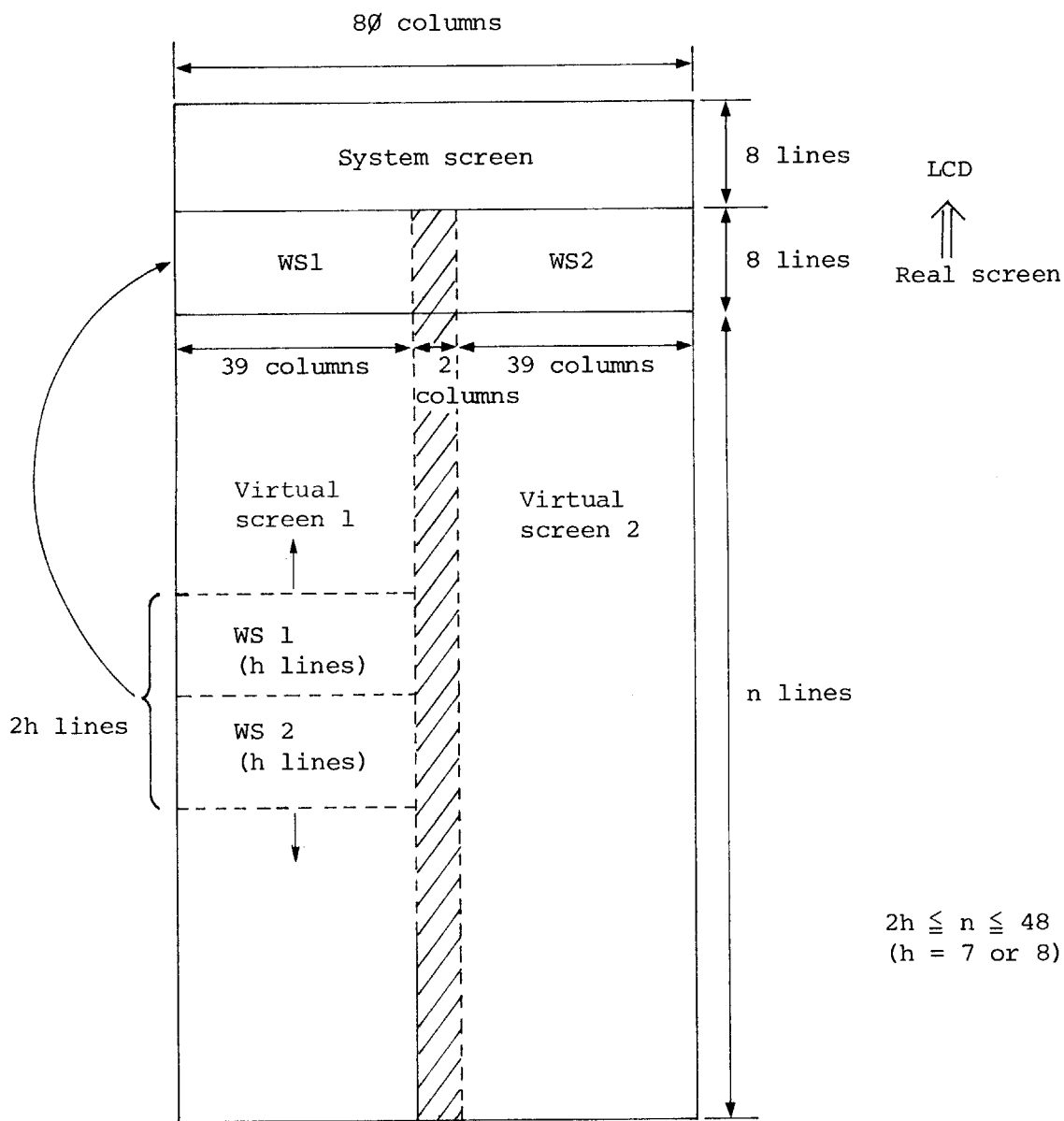
Used only to separate two virtual screens. Two blank columns are placed on the real screen.

c) MODE 2 (Split screen)

In this mode, two virtual screens are available. Either
of them may consist of any number of columns provided
that the total number of columns is 79. Since each
virtual screen is provided with its own window screen,
the contents of the two virtual screens can be displayed
on the real screen at the same time. These window
screens can scroll independently over the associated
virtual window.

Any character or symbol may be defined as the delimiter
to separate the two virtual screens on the real screen.

Screen RAM structure in MODE 2

Delimiter. Any character or symbol may be defined.

$$n1 + n2 = 79$$
$$2h \leqq n \leqq 48$$
$$(h = 7 \text{ or } 8)$$
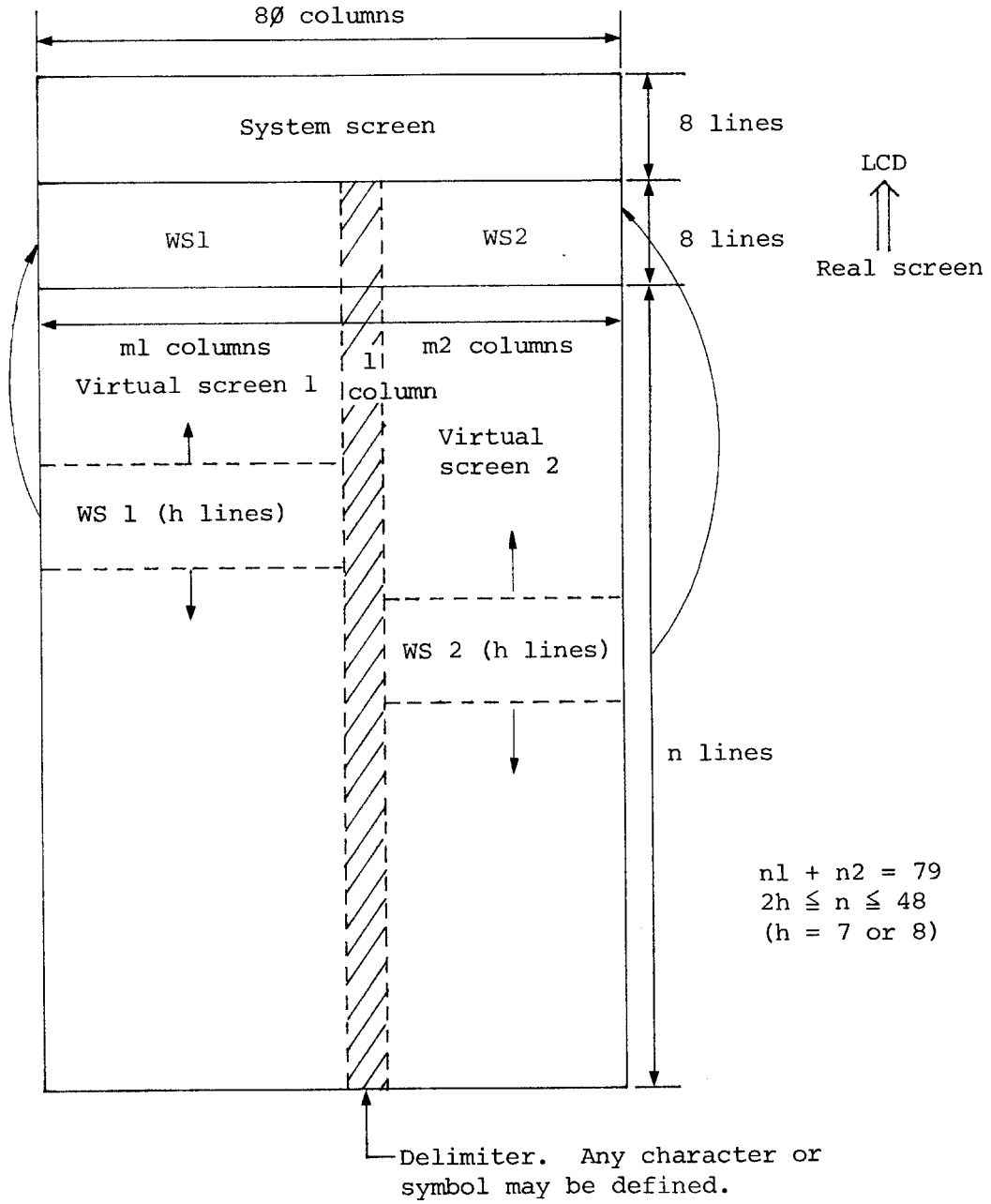
d) MODE 3 (Graphics mode)

This mode permits the application program to display
graphics patterns on a dot basis. In this mode, the
VRAM is occupied by the 480 dots by 64 dots graphics
screen. Accordingly, there can be only one user screen
at a time. No virtual screen is supported and only a
real screen is available in this mode.

Screen RAM structure in mode 3

```
┌──────────────────────────────┐ ▲
│        System screen         │ │ 8 lines
│                              │ ▼
├──────────────────────────────┤
│                              │
│       Graphics screen        │
│                              │
│                              │
│      480 dots x 64 dots      │
│                              │      Corresponds to
│                              │      the LCD screen of
│                              │      8 lines of 80
│                              │      columns.
│                              │
│                              │         ⇓
│                              │        LCD
│                              │
└──────────────────────────────┘
```
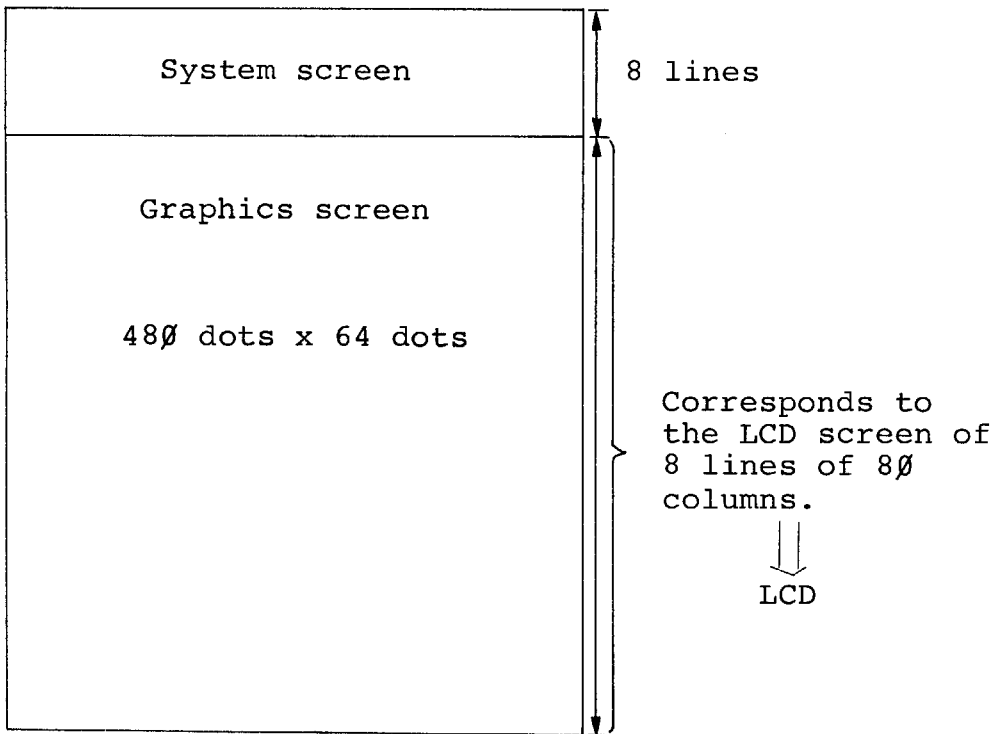
6.4 Special Screen Features

(1) Display screen and write screen

As explained in the previous section, two virtual screens
are supported in MODE 0 through MODE 2. However, data
can be written only onto the screen in which the cursor
is stationed (selected virtual screen). In MODE 0 and
MODE 1, only one of the two virtual screens holding the
cursor is displayed at a time.

Only one screen is supported in MODE 3 and used for both
display and write.

(2) Scroll mode

In character screen modes (MODE 0, 1, 2), the window
screen scrolls up and down over the virtual screen in
the tracking mode and non-tracking modes.

   1) Tracking mode

   In this mode, the window screen scrolls following the
   cursor. That is, the cursor is always in the window
   (real) screen.

2) Non-tracking mode

The window screen does not follow the cursor movement.
The WS remains in the current position even if the
cursor moves beyond the WS. The cursor stays in the
virtual screen and newly entered data is placed at
the cursor position.
When the mode is switched to the tracking mode, the
WS continues to scroll until the cursor appears in
the WS.

(3) Cursor functions and types
    1) Character screen
    Any combination of the following cursor functions and
    types are available on the character screen:
    - Cursor display: ON/OFF
    - Cursor type: Underline/block
    - Cursor blink: ON/OFF
    2) Graphics screen
    The cursor type is preset to underline and nonblink
    on the graphics screen. The user can control only
    the cursor display ON/OFF state.

(4) Block flash (blink)
    The displayed data can blink as the selected file
    name can on the MENU screen. Since this function is

supported by the slave CPU, the block flash command
must be issued directly to the slave CPU using the
BIOS SLAVE call. For further information, see
Chapter 13, "SLAVE CPU".

(5) User defined characters
   User characters can be defined under the following
   conditions:
   Number of user defined characters: 32 maximum
   User defined character codes: 0E0H - 0FFH

   0E0H and 0E1H are used by the OS. Although the user
   can overwrite the OS defined characters under these
   two codes with his own characters, the user defined
   characters will be overwritten again by the OS
   defined characters if the RESET switch is pressed.
   User defined characters under the other codes remain
   unchanged when the RESET switch is pressed.

   $$0E0H = " \triangle " \qquad 0E1H = " \hookleftarrow "$$

   Characters can be defined by sending the ESC + 0E0H
   sequence through the CONOUT routine.
   User defined characters are displayed on the screen
   but not printed on the printer.

(6) Character sets

The MAPLE supports the character sets for the
countries listed below. The user can select any of
them by sending the ESC + "C" sequence via CONOUT.
See the character set tables at the back of this
manual.

- USASCII
- FRANCE
- GERMANY
- ENGLAND
- DENMARK
- SWEDEN
- ITALY
- SPAIN
- NORWAY

6.5 How to Use CONOUT

Display of data on the screen is primarily accomplished
by the CONOUT BIOS call. Call the CONOUT using the
following calling sequence:

     Entry address = WBOOT + 09H

     Entry parameter = Load into C reg.

The CONOUT supports various control codes and ESC sequences.
The control codes and ESC sequences are fully described
in the next section. When entering more than one data
byte such as when sending an ESC sequence, call the CONOUT
the required number of times with each data byte loaded
in the C reg.

6.6 CONOUT Functions

The pages that follow list the CONOUT functions that the
MAPLE OS supports to handle control codes or ESC
sequences.

CONOUT SPECIFICATIONS (1)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
05H	ERASE END OF LINE	Deletes to the end of the line from the cursor position on the screen.	Deletes to the end of the line from the cursor position on the virtual screen in write operation.	Same as in system mode.
07H	BELL	Sounds the speaker at 440 Hz for 1 minute. In ASCII Ver. B, this function sounds the speaker for 200 ms.	Same as in system mode.	Same as in system mode.
08H	BACK SPACE	Moves the cursor one position to the left. The cursor does not move if it is in the home position.	Moves the cursor one position to the left in the currently selected virtual screen. The cursor does not move when it is in the home position.	Same as in system mode.
09H	TAB	Does nothing.	Searches for the next tab position following the cursor position to the right in the currently selected virtual screen and moves the cursor to the first tab position encountered. When no tab position is found on the current line, the function moves the cursor to the beginning of the next line. Tab position = $(1+8n)$ positions $n = 0, 1, 2 \ldots$	Searches for the next tab position starting at the current cursor position on the screen to the right and moves the cursor to the first tab position encountered. When no tab position is found on the line, the function moves the cursor to the beginning of the next line. Tab position = $(1+8n)$ positions $n = 0, 1, 2 \ldots$
0AH	LINE FEED	Does nothing.	Moves the cursor down one line in the currently selected virtual screen. The function scrolls up one line when the cursor is on the bottom line of the virtual screen.	Moves the cursor down one line on the screen. The function scrolls up one line when the cursor is on the bottom line of the screen.

CONOUT SPECIFICATIONS (1)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
OBH	HOME	Moves the cursor to the upper left corner on the screen.	Moves the cursor to the upper left corner in the currently selected virtual screen. How the cursor behaves depends on the tracking/non-tracking specification.	Same as in system mode.
OCH	CLEAR SCREEN & HOME	Clears the entire screen and moves the cursor to the beginning of the screen.	Clears the entire virtual screen in write operation and moves the cursor to the beginning of the virtual screen on which the cursor rests. How the cursor behaves depends on the tracking/non-tracking specification.	Same as in system mode.
ODH	CARRIAGE RETURN	Moves the cursor to the first column of the current line. When a character has been displayed in the last column on the line, this function moves the cursor to the first column of the previous line (the line on which the last character was displayed).	Moves the cursor to the first column of the current line. When a character has been displayed in the last column on a line on the virtual screen, this function moves the cursor to the first column of the previous line (the line on which the last character was displayed).	Same as in system mode.

CONOUT SPECIFICATIONS (2)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
10H	SCREEN UP	Does nothing.	Moves up the window screen a screenful of lines (h lines) over the currently selected virtual screen. Display starts at the home position when it moves up beyond the home position. The cursor is held in the original position on the virtual screen.	Does nothing.
11H	SCREEN DOWN	Does nothing.	Moves down the window screen a screenful of lines (h lines) over the currently selected virtual screen. The last line of the virtual screen is set to the bottom of the window screen when display moves down beyond the end of the virtual screen. The cursor is held in the original position on the virtual screen.	Does nothing.
1AH	ERASE END OF SCREEN	Clears to end of the screen from the current cursor position.	Clears to the end of the virtual screen from the current cursor position.	Same as in system mode.
1BH	ESC	Receives the next code as the second parameter of an ESC sequence.	Same as in system mode.	Same as in system mode.

CONOUT SPECIFICATIONS (3)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
1CH	CURSOR RIGHT	Moves the cursor one position to the right on the screen. If the cursor is in the last column on a line, moves it to the beginning of the next line. This function does nothing when the cursor is in the last column of the last line on the screen.	Moves the cursor one position to the right in the currently selected virtual screen. When the cursor is in the last column on a line, moves it to the beginning of the next line. When the cursor is in the last column on the last line of the window screen, display automatically scrolls up one line in the tracking mode and goes beyond the screen in the non-tracking mode. The function does nothing when the cursor is in the last column of the last line on the virtual screen.	Same as in system mode.
1DH	CURSOR LEFT	Moves the cursor one position to the left on the screen. When the cursor is in the first column on a line, moves it to the last column on the previous line. This function does nothing when the cursor is in the home position on the screen.	Moves the cursor one position to the left in the currently selected virtual screen. When the cursor is in the first column on a line, moves it to the last column on the previous line. When the cursor is in the first column on the first line of the window screen, the display automatically scrolls down one line in the tracking mode or it goes beyond the screen in the non-tracking mode. The function does nothing when the cursor is in the home position on the virtual screen.	Same as in system mode.

CONOUT SPECIFICATIONS (3)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
1EH	CURSOR UP	Moves the cursor up one line on the screen. This function does nothing when the cursor is on the first line of the screen.	Moves the cursor up one line in the currently selected virtual screen. When the cursor is on the first line of the window screen, display automatically scrolls down one line in the tracking mode and goes beyond the screen in the non-tracking mode. The function does nothing when the cursor is on the first line of the virtual screen.	Same as in system mode.
1FH	CURSOR DOWN	Moves the cursor down one line on the screen. This function does nothing when the cursor is on the last line of the screen.	Moves the cursor down one line in the currently selected virtual screen. When the cursor is on the last line of the window screen, display automatically scrolls down one line in the tracking mode and goes beyond the screen in the non-tracking mode. The function does nothing when the cursor is on the last line of the virtual screen.	Same as in system mode.

CONOUT SPECIFICATIONS (4)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC"%"	ACCESS CGROM DIRECTLY	Causes MAPLE CGROM to read the character associated with the specified code and displays it in the cursor position on the screen. (Command sequence) 1st byte: ESC 2nd byte: "%" 3rd byte: n $\varnothing \leq n \leq 255$	Causes MAPLE CGROM to read the character associated with the specified code and displays it in the cursor position in the currently selected virtual screen. (Command sequence) 1st byte: ESC 2nd byte: "%" 3rd byte: n $\varnothing \leq n \leq 255$	Same as in system mode.
ESC"(" ESC")"	PASS THROUGH	Does nothing.	Does nothing.	Does nothing.
ESC"*"	CLEAR SCREEN	Clears the screen and places the cursor in the home position.	Clears the currently selected virtual screen and places the cursor in the home position.	Same as in system mode.
ESC"0"	REVERSE ON	Does nothing.	Does nothing.	Does nothing.
ESC"1"	REVERSE OFF	Does nothing.	Does nothing.	Does nothing.
ESC"2"	CURSOR OFF	Suppresses the cursor display. The cursor can move, though invisible.	Same as in system mode.	Same as in system mode.
ESC"3"	CURSOR ON	Displays the cursor.	Same as in system mode.	Same as in system mode.
ESC"4"	UNDERLINE ON	Does nothing.	Does nothing.	Does nothing.
ESC"5"	UNDERLINE OFF	Does nothing.	Does nothing.	Does nothing.
ESC"6"	HIGHLIGHT ON	Does nothing.	Does nothing.	Does nothing.
ESC"7"	HIGHLIGHT OFF	Does nothing.	Does nothing.	Does nothing.

CONOUT SPECIFICATIONS (5)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC"8"	BLINK ON	Does nothing.	Does nothing.	Does nothing.
ESC"9"	BLINK OFF	Does nothing.	Does nothing.	Does nothing.
ESC"<"	PUSH CURSOR POSITION	Does nothing.	Does nothing.	Does nothing.
ESC"="	SET CURSOR POSITION	Specifies the cursor position on the screen. (Command sequence) 1st byte: ESC 2nd byte: "=" 3rd byte: Row position (m + 31) 4th byte: Column position (n + 31) m and n indicates the position on the virtual screen, where its home position is (1,1).	Specifies the cursor position in the currently selected virtual screen. If the cursor goes off the screen in the tracking mode, this function redisplays the screen so that the cursor appears in the center of the screen. (Command sequence) 1st byte: ESC 2nd byte: "=" 3rd byte: Row position (m + 31) 4th byte: Column position (n + 31) m and n indicates the position on the virtual screen, where its home position is (1,1).	Same as in system mode.
ESC">"	POP CURSOR POSITION	Does nothing.	Does nothing.	Does nothing.

CONOUT SPECIFICATIONS (5)

CODE	FUNCTION	SYSTEM MODE	MODE Ø/1/2	MODE 3
ESC"C"	SET CHARACTER-SET TABLE	Does nothing under the Japanese-language OS. Under the OS for other countries, this function sets up the specified character set. (Command sequence) 1st byte: ESC 2nd byte: "C" 3rd byte: Country Identifica-tion Character U: USASCII F: FRANCE G: GERMANY E: ENGLAND D: DENMARK W: SWEDEN I: ITALY S: SPAIN N: NORWAY	Same as in system mode.	Same as in system mode.
ESC"L"	CHANGE CRT COLOR	Does nothing. In new ASCII version (M25Ø3ØCB) kana mode, this function displays the parameters as they are (garbage data).	Same as in system mode.	Same as in system mode.

CONOUT SPECIFICATIONS (6)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC"p"	SCREEN DUMP	Produces a hard copy of the screen on the printer.	Produces a hard copy of the current window screen on the printer.	Same as in system mode.
ESC"T"	ERASE END OF LINE	Deletes to the end of the line from the cursor position on the screen.	Deletes to the end of the line from the cursor position in the currently selected virtual screen.	Same as in system mode.
ESC"Y"	ERASE END OF SCREEN	Clears to the end of the screen from the current cursor position.	Clears to the end of the virtual screen from the current cursor position.	Same as in system mode.
ESC 7BH	SECRET	Changes the subsequent output characters to spaces.	Same as in system mode.	Same as in system mode.
ESC 7DH	NON SECRET	Cancels the above mode.	Same as in system mode.	Same as in system mode.
ESC 80H	1BYTE CODE TO 2BYTE CODE (KATAKANA)	Does nothing. In new ASCII version (M25Ø3ØCB) kana mode, this function displays the parameters as they are (garbage data).	Same as in system mode.	Same as in system mode.
ESC 81H	1BYTE CODE TO 2BYTE CODE (HIRAGANA)	Does nothing. In new ASCII version (M25Ø3ØCB) kana mode, this function displays the parameters as they are (garbage data).	Same as in system mode.	Same as in system mode.
ESC 82H	2BYTE CODE TO 2BYTE CODE	Does nothing. In new ASCII version (M25Ø3ØCB) kana mode, this function displays the parameters as they are (garbage data).	Same as in system mode.	Same as in system mode.

CONOUT SPECIFICATIONS (7)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 90H	PARTIAL SCROLL UP	Does nothing.	(Command sequence) 1st byte: ESC 2nd byte: 90H 3rd byte: n-1 $(\emptyset \leq (n-1) \leq R-1)$ 4th byte: m $(1 \leq m \leq R)$ $(n-1) + (m-1) < R$, where R is the number of the lines on the virtual screen in mode 0, 1, or 2 and the number of the lines on the window screen in mode 3. One character line is made up of 8 dot lines. The m-line screen segment from the nth line scrolls up one line as the above command sequence is processed. The (n+m-1)th line is left blank.	
ESC 91H	PARTIAL SCROLL DOWN	Does nothing.	(Command sequence) 1st byte: ESC 2nd byte: 91H 3rd byte: n-1 $(\emptyset \leq (n-1) \leq R-1)$ 4th byte: m $(1 \leq m \leq R)$ $(n-1) + (m-1) < R$, where R is the number of the lines on the virtual screen in mode 0, 1, or 2 and the number of the lines on the window screen in mode 3. One character line is made up of 8 dot lines. The m-line screen segment from the nth line scrolls down one line as the above command sequence is processed. The nth line is left blank.	
ESC 92H	SCROLL RIGHT n character	Does nothing. In new ASCII version (M2503ØCB) kana mode, this function displays the parameters as they are (garbage data).	Same as in system mode.	Same as in system mode.
ESC 93H	SCROLL LEFT n character	Does nothing. In new ASCII version (M2503ØCB) kana mode, this function displays the parameters as they are (garbage data).	Same as in system mode.	Same as in system mode.

CONOUT SPECIFICATIONS (8)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 94H	SET SCROLL STEP	Does nothing.	Specifies the number of lines to scroll when the scroll up or down n line (ESC 96H or 97H) function is executed. (Command sequence) 1st byte: ESC 2nd byte: 94H 3rd byte: Number of lines to be scrolled (n) 1 ≤ n ≤ h (h = number of window lines)	Does nothing.
ESC 95H	SET SCROLL MODE	Does nothing.	Enables and disables the automatic scroll. The modes in which the automatic scroll is enabled and disabled are called "tracking mode" and "non-tracking mode," respectively. The default is tracking mode. (Command sequence) 1st byte: ESC 2nd byte: 95H 3rd byte: mode 0 (tracking mode) or 1 (non-tracking mode)	Does nothing
ESC 96H	SCROLL UP n LINES	Does nothing.	Moves up the window screen n lines over the virtual screen on which the cursor rests. Display starts in the home position when it moves up beyond the home position. The cursor is held in the original position on the virtual screen. The value of n is specified by the ESC 94H sequence. The default value is 1.	Does nothing.

CONOUT SPECIFICATIONS (8)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 97H	SCROLL DOWN n LINES	Does nothing.	Moves down the window screen a screenful of lines (n lines) over the virtual screen on which the cursor rests. The last line of the virtual screen is set to the bottom of the window screen when display moves down beyond the end of the virtual screen. The cursor is held in the original position on the virtual screen. The value of n is specified by the ESC 94H sequence. The default value is 1.	Does nothing.

CONOUT SPECIFICATIONS (9)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 0A0H	INS LED ON	Turns on the insert mode LED.	Same as in system mode.	Same as in system mode.
ESC 0A1H	INS LED OFF	Turns off the insert mode LED.	Same as in system mode	Same as in system mode.
ESC 0A2H	CAPS LOCK LED ON	Turns on the CAPS LOCK LED.	Same as in system mode.	Same as in system mode.
ESC 0A3H	CAPS LOCK LED OFF	Turns off the CAPS LOCK LED.	Same as in system mode.	Same as in system mode.
ESC 0A4H	NUM LED ON	Turns on the NUM input LED.	Same as in system mode.	Same as in system mode.
ESC 0A5H	NUM LED OFF	Turns off the NUM input LED.	Same as in system mode.	Same as in system mode.
ESC 0B0H	FUNCTION KEY CHECK MODE ON	Enables the programmable function key (PF keys) entry. (YPFCMFLG is set to 0FFH.) (Return code) When the C register is loaded with 0, the ASCII code corresponding to the pressed key other than PF keys is returned to the A register. When the C register is loaded with 0FFH, one of the following codes associated with PF1 to PF10, is returned to the A register. OS for the countries other than Japan: 0E0H – 0E9H OS for Japan: 0C0H – 0C9H	Same as in system mode.	Same as in system mode.
ESC 0B1H	FUNCTION KEY CHECK MODE OFF	Disables the programmable function key entry. (YPFCMFLG is set to 0.) When a programmable function key is pressed, the associated string is returned.	Same as in system mode.	Same as in system mode.

CONOUT SPECIFICATIONS (1∅)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 0C6H	DOT LINE WRITE	Does nothing.	Does nothing.	Draws a line of user-specified dot pattern on the LCD screen. (Command sequence) 1st byte: ESC 2nd byte: ∅C6H 3rd byte: Starting position (horizontal) $<H>$ ml 4th byte: Starting position (horizontal) $<L>$ 5th byte: Starting position (vertical) $<H>$ m2 6th byte: Starting position (vertical) $<L>$ 7th byte: Ending position (horizontal) $<H>$ nl 8th byte: Ending position (horizontal) $<L>$ 9th byte: Ending position (vertical) $<H>$ n2 1∅th byte: Ending position (vertical $<L>$ 11th byte: Mask pattern 1 12th byte: Mask pattern 2 13th byte: Operation where: $|m1-n1| <16383$ $|m2-n2| <16383$ The pattern must be specified in bit image in 16 bits from mask pattern 1, bit 7 through mask pattern 2, bit ∅. Line segments are masked sequentially by this mask pattern in the mode specified by operation. The dot coordinates pf a slanted line.

CONOUT SPECIFICATIONS (1∅)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
				are automatically calculated by the function. (Operation) Operation specifies the mode in which the dot coordinates corresponding to the 1 bits of the (16-bit) mask pattern are to be masked. The operation codes are: 1 = Off 2 = On 3 = Complement See separate sheets for dot coordinates.

6-34

CONOUT SPECIFICATIONS (11)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 0C7H	PSET/PRESET	Does nothing.	Does nothing.	Sets and resets the specified point on the LCD screen. (Command sequence) 1st byte: ESC 2nd byte: 0C7H 3rd byte: Function code 4th byte: Plot position (vertical) n1 5th byte: Plot position (horizontal) <H> 6th byte: Plot position (horizontal) <L>n2 0 ≤ n1 ≤ 63, 0 ≤ n2 ≤ 479 Function code: 1 = PSET (ON) 0 = PRESET (OFF)
ESC 0D0H	DISPLAY MODE SET	Does nothing.	Changes the screen mode and clears the screen. Mode 0 (Command sequence) 1st byte: ESC 2nd byte: 0D0H 3rd byte: Mode [0] 4th byte: Number of lines on VS1, n1 5th byte: Number of lines on VS2, n2 n1 + n2 ≤ 48, n1, n2 ≤ 8 The number of columns on VS1 and VS2 is 80. Mode 1 (Command sequence) 1st byte: ESC 2nd byte: 0D0H 3rd byte: Mode [1] 4th byte: Number of lines on VS1, n	Mode 2 (Command sequence) 1st byte: ESC 2nd byte: 0D0H 3rd byte: Mode [2] 4th byte: Number of lines on VS1, n 5th byte: Number of columns on VS1, m 6th byte: Screen delimiter 8 ≤ n ≤ 48, 1 ≤ m ≤ 78 VS1 and VS2 have the same number of lines. VS2 = 79 - (number of columns on VS1) Mode 3 (Command sequence) 1st byte: ESC 2nd byte: 0D0H

CONOUT SPECIFICATIONS (11)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
			$16 \leq n \leq 48$ The number of columns on VS1 and VS2 are 39.	3rd byte: Mode [3]
ESC 0D1H	SELECT DISPLAY SCREEN	Does nothing.	Specifies which virtual screen is to be displayed. (Command sequence) 1st byte: ESC 2nd byte: 0D1H 3rd byte: 0/1 0 = VS1 1 = VS2 The default is VS1.	Does nothing.

CONOUT SPECIFICATIONS (12)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 0D2H	DIRECT DISPLAY OF PHYSICAL SCREEN	Displays a specified character in the specified position on the real screen. (Command sequence) 1st byte: ESC 2nd byte: ØD2H 3rd byte: Row position (1-8) 4th byte: Column position (1-8Ø) The position must be specified with column and row numbers. This function displays a character directly in any location on the 8Ø × 8 screen. This function uses not internal but CG codes.	Same as in system mode.	Same as in system mode.
ESC 0D3H	SELECT FUNCTION KEY DISPLAY	Specifies whether or not function key definitions are to be displayed on the screen. (Command sequence) 1st byte: ESC 2nd byte: ØD3H 3rd byte: Ø/1 Ø = Displayed (The window screen has 7 lines.) 1 = Not displayed (The window screen has 8 lines.)	Same as in system mode.	Same as in system mode.
ESC 0D4H	LOCATE TOP OF SCREEN	Dos nothing.	Sets the window screen to the beginning of the virtual screen on which the cursor rests. The cursor is held in the original position.	Does nothing.

CONOUT SPECIFICATIONS (12)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 0D5H	LOCATE END OF SCREEN	Does nothing.	Sets the window screen to the end of the virtual screen on which the cursor rests. The cursor is held in the original position.	Does nothing.

CONOUT SPECIFICATIONS (13)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 0D6H	SELECT CURSOR KIND	Used to select the type of the cursor. (Command sequence) 1st byte: ESC 2nd byte: 0D6H 3rd byte: Type of the cursor [0/1/2/3] 0 = Block and blink 1 = Block and nonblink 2 = Underline and blink 3 = Underline and nonblink The default is 0.	Same as in system mode.	Does nothing. (Set to the nonblink underline cursor.)
ESC 0D7H	FIND CURSOR	Does nothing.	Moves the window screen to the cursor position over the virtual screen so that the cursor line will appear near the center of the screen.	Does nothing.
ESC 0E0H	SET DOWN LOAD CHARACTER	Does nothing.	Defines external characters with the codes 0E0H – 0FFH. (Command sequence) 1st byte: ESC 2nd byte: 0E0H 3rd byte: Character code (0E0H – 0FFH) 4th byte: Character pattern (1) 5th byte: Character pattern (2) 6th byte: Character pattern (3) 7th byte: Character pattern (4) 8th byte: Character pattern (5) 9th byte: Character pattern (6) 10th byte: Character pattern (7) 11th byte: Character pattern (8) (ROW SCAN)	Defines external characters with the codes 0E0H – 0FFH. ... 1 (Pattern 1) ... 2 (Pattern 2) ... 3 (Pattern 3) ... 4 (Pattern 4) ... 5 (Pattern 5) ... 6 (Pattern 6) ... 7 (Pattern 7) ... 8 (Pattern 8) 543210 (bit position) The lowest 6 bits are used to display a character pattern.

6-39

CONOUT SPECIFICATIONS (14)

CODE	FUNCTION	SYSTEM MODE	MODE Ø/1/2	MODE 3
ESC OFOH	KEYBOARD REPEAT ON/OFF	Does nothing.	Controls the keyboard repeat function (accepting inputs repeatedly while the key is held down). The default is REPEAT ON. (Command sequence) 1st byte: ESC 2nd byte: ØFØH 3rd byte: <Ø/1> where Ø = REPEAT OFF 1 = REPEAT ON	
ESC OF1H	SET KEYBOARD REPEAT START TIME	Does nothing.	Specifies the keyboard repeat start time (the interval between the time the first character is entered and the time the second character is taken in when a key is held down). (Command sequence) 1st byte: ESC 2nd byte: ØF1H 3rd byte: n where 1 ≤ n ≤ 127 time = n/64 SEC The default value is approx. 656 ms.	
ESC OF2H	SET KEYBOARD REPEAT INTERVAL TIME	Does nothing.	Specifies the keyboard repeat interval time. (Command sequence) 1st byte: ESC 2nd byte: ØF2H 3rd byte: n where 1 ≤ n ≤ 127 time = n/256 SEC The default is about 7Ø ms.	

CONOUT SPECIFICATIONS (14)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC 0F3H	SET ARROW KEY CODE	Does nothing.	Defines the arrow key codes. (Command sequence) 1st byte: ESC 2nd byte: 0F3H Default 3rd byte: Code of → --- 1CH 4th byte: Code of ← --- 1DH 5th byte: Code of ↑ --- 1EH 6th byte: Code of ↓ --- 1FH See "Arrow Key Function Chart" for details.	
ESC 0F4H	SET SCROLL KEY CODE	Does nothing.	Defines codes for SHIFT + arrow keys. (Command sequence) 1st byte: ESC 2nd byte: 0F4H Default 3rd byte: Code of SHIFT + → --- 80H 4th byte: Code of SHIFT + ← --- 80H 5th byte: Code of SHIFT + ↑ --- 0F8H 6th byte: Code of SHIFT + ↓ --- 0F9H See "Arrow Key Function Chart" for details.	

CONOUT SPECIFICATIONS (15)

CODE	FUNCTION	SYSTEM MODE	MODE 0/1/2	MODE 3
ESC ØF5H	SET CTRL KEY CODE	Does nothing.	Defines codes for CTRL + arrow keys. (Command sequence) 1st byte: ESC 2nd byte: ØF4H Default 3rd byte: Code of CTRL + → --- ØFFH 4th byte: Code of CTRL + ↓ --- ØFEH 5th byte: Code of CTRL + ↑ --- ØFAH 6th byte: Code of CTRL + → --- ØFBH See "Arrow Kay Function Chart" for details.	MODE 3
ESC ØF6H	CLEAR KEY BUFFER	Clears the keyboard buffer (Clears entire data previously entered.)	Same as in system mode.	Same as in system mode.
ESC ØF7H	SET KEY SHIFT	Defines the key shift code. (Command sequence) 1st byte: ESC 2nd byte: ØF7H 3rd byte: Shift Code Shift code Shift mode bit4 ON NUM bit3 ON HIRAGANA bit2 ON KATAKANA bit1 ON CAPS bitØ ON NORMAL The function of the CTRL and SHIFT keys differs depending on the current keyboard state. When two or more bits are ON, the mode corresponding to the highest bit is taken.	Same as in system mode.	Same as in system mode.

CONOUT SPECIFICATIONS (1)

CODE	FUNCTION	MODE 4	MODE 5	REMARK
05H	ERASE END OF LINE	Same as in system mode.	Same as in system mode.	
07H	BELL	Same as in system mode.	Same as in system mode	
08H	BACK SPACE	Same as in system mode.	Same as in system mode.	

09H	TAB	Moves the cursor to the next tab position on the screen. This function moves the cursor to the beginning of the next line when no tab position is found on the current line. Tab position:
		    Full width mode
		       $8n + 1$
		    Double width mode
		       $16n + 1$
		       $(n = 1, 2, \ldots)$
			Same as in mode 4.	

0AH	LINE FEED	Moves the cursor down one line on the screen. This function causes the screen to scroll up one line when the cursor is on the bottom line of the screen.	Same as in mode 4.	
0BH	HOME	Same as in system mode.	Same as in system mode.	
0CH	CLEAR SCREEN & HOME	Same as in system mode.	Same as in system mode.	
0DH	CARRIAGE RETURN	Same as in system mode.	Same as in system mode.	

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
10H	SCREEN UP	Does nothing.	Does nothing.	
11H	SCREEN DOWN	Does nothing.	Does nothing.	
1AH	ERASE END OF SCREEN	Same as in system mode.	Same as in system mode.	
1BH	ESC	Same as in system mode.	Same as in system mode.	
1CH	CURSOR RIGHT	Moves the cursor one character position (1 column in full width mode or 2 columns in double width mode) to the right on the screen. When the cursor is in the last column on a line, this function moves the cursor to the first column of the next line. This function does nothing when the cursor is in the last column of the last line on the screen.	Same as in mode 4.	
1DH	CURSOR LEFT	Moves the cursor one character position (1 column in full width mode or 2 columns in double width mode) to the left on the screen. When the cursor is in the first column on a line, this function moves the cursor to the last column of the previous line. This function does nothing when the cursor is in the first column of the first line on the screen.	Same as in mode 4.	
1EH	CURSOR UP	Same as in system mode.	Same as in system mode.	
1FH	CURSOR DOWN	Same as in system mode.	Same as in system mode.	

CONOUT SPECIFICATIONS (3)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC"%"	ACCESS CGROM DIRECTLY	Does nothing.	Does nothing.	(Command sequence) 1st byte: ESC 2nd byte: "%" 3rd byte: n ($\emptyset \le n \le 255$)
ESC"(" ESC")"	PASS THROUGH	Does nothing.	Does nothing.	
ESC"*"	CLEAR SCREEN	Same as in system mode.	Same as in system mode.	Same as $\emptyset$CH.
ESC"0"	REVERSE ON	Displays the subsequent output characters in reverse video.	Does nothing.	Reverse video character size = 1 character size + underline area size. The character size depends on the current character width.
ESC"1"	REVERSE OFF	Cancels the reverse video display function.	Does nothing.	
ESC"2"	CURSOR OFF	Same as in system mode.	Same as in system mode.	
ESC"3"	CURSOR ON	Same as in system mode.	Same as in system mode.	
ESC"4"	UNDERLINE ON	Displays characters with underlines.	Same as in mode 4.	
ESC"5"	UNDERLINE OFF	Cancels the underline display function.	Same as in mode 4.	
ESC"6"	HIGHLIGHT ON	Does nothing.	Does nothing.	
ESC"7"	HIGHLIGHT OFF	Does nothing.	Does nothing.	

CONOUT SPECIFICATIONS (4)

CODE		MODE 4	MODE 5	REMARKS
ESC"8"	BLINK ON	Does nothing.	Does nothing.	
ESC"9"	BLINK OFF	Does nothing.	Does nothing.	
ESC"<"	PUSH CURSOR POSITION	Does nothing.	Does nothing.	
ESC"="	SET CURSOR POSITION	Same as in system mode.	Same as in system mode.	
ESC">"	POP CURSOR POSITION	Does nothing.	Does nothing.	
ESC"C"	SET CHARACTER-SET TABLE	Does nothing.	Does nothing.	
ESC"L"	CHANGE CRT COLOR	Does nothing.	Does nothing.	Nothing is executed in old ASCII version (M25Ø3ØCA). In new ASCII version (M25Ø3ØCB), the data in the third parameter is displayed as is.
ESC"P"	SCREEN DUMP	Same as in system mode. (Only 2 or 3 lines (or entire screen) may be specified. See ESC + ØF8H and ESC + ØF9H descriptions for format.)	Same as in system mode.	
ESC"T"	ERASE END OF LINE	Same as in system mode.	Same as in system mode.	
ESC"Y"	ERASE END OF SCREEN	Same as in system mode.	Same as in system mode.	
ESC 7BH	SECRET	Same as in system mode.	Same as in system mode.	Same as ESC Ø5H.
ESC 7CH	CHANGE V-RAM	Does nothing.	Does nothing.	Same as ESC 1AH.
ESC 7DH	NON SECRET	Same as in system mode.	Same as in system mode.	

6-46

CONOUT SPECIFICATIONS (5)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 80H	1BYTE CODE TO 2BYTE CODE (KATAKANA)	Converts a JIS C6220 code (katakana) into a shift JIS code and places it in the DE registers. (Command sequence) 1st byte: ESC 2nd byte: 80H 3rd byte: JIS C6220 code (Return) DE registers: Shift JIS code See separate sheets for the character conversion chart.		Nothing is executed in old ASCII version (M2503CA). In new ASCII version (M2503CB), the data in the third para-meter is displayed as is.
ESC 81H	1BYTE CODE TO 2BYTE CODE (HIRAGANA)	Converts a JIS C6220 code (hiragana) into a shift JIS code and places it in the DE registers. (Command sequence) 1st byte: ESC 2nd byte: 81H 3rd byte: JIS C6220 code (Return) DE registers: Shift JIS code See separate sheets for the character conversion chart.		Nothing is executed in old ASCII version (M2503CA). In new ASCII version (M2503CB), the data in the third para-meter is displayed as is.
ESC 82H	2BYTE CODE TO 1BYTE CODE	Converts a shift JIS code into a JIS C6220 code and places it in the DE registers. (Command sequence) 1st byte: ESC 2nd byte: 82H 3rd byte: Shift JIS (High) 4th byte: Shift JIS (Low) (Return) DE registers: JIS C6220 code See separate sheets for the character conversion chart.		Nothing is executed in old ASCII version (M2503CA). In new ASCII version (M2503CB), the data in the third para-meter is displayed as is.
ESC 83H	JIS C6226 CODE TO SHIFT JIS CODE	Converts a JIS C6226 code into a shift JIS code and places it in the DE registers. (Command sequence) 1st byte: ESC 2nd byte: 83H 3rd byte: JIS C6226 (High) 4th byte: JIS C6226 (Low)		

CONOUT SPECIFICATIONS (5)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
		(Return) DE registers: Shift JIS code See separate sheets for the character conversion table.		
ESC 84H	SHIFT JIS CODE TO JIS C6226 CODE	Converts a shift JIS code into the JIS C6226 code and places it in the DE registers. (Command sequence) 1st byte: ESC 2nd byte: 84H 3rd byte: Shift JIS (High) 4th byte: Shift JIS (Low) (Return) DE registers: JIS C6226 code See separate sheets for the character conversion chart.		

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 90H	PARTIAL SCROLL UP	Does nothing.	Does nothing.	
ESC 91H	PARTIAL SCROLL DOWN	Does nothing.	Does nothing.	
ESC 92H	SCROLL RIGHT n CHARACTER	Scrolls the screen to the right by n columns. (Command sequence) 1st byte: ESC 2nd byte: 92H 3rd byte: Number of columns to be scrolled, n $1 \leq n \leq$ (maximum number of columns on the screen) Does nothing when an illegal number of columns are specified.	Does nothing.	Nothing is executed in old ASCII version (M25Ø3ØCA). In new ASCII version (M25Ø3ØCB), the data in the third parameter is displayed as is.
ESC 93H	SCROLL LEFT n CHARACTER	Scrolls the screen to the left by n columns. (Command sequence) 1st byte: ESC 2nd byte: 93H 3rd byte: Number of columns to scroll, n $1 \leq n \leq$ (maximum number of columns on the screen) Does nothing when an illegal number of columns is specified.	Does nothing.	Nothing is executed in old ASCII version (M25Ø3ØCA). In new ASCII version (M25Ø3ØCB), the data in the third parameter is displayed as is.

CONOUT SPECIFICATIONS (6)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 94H	SET SCROLL STEP	Does nothing. (The number of lines to be scrolled may be specified in the same way as for screen 5. This value is effective only when screen 5 is selected. On screen 4, the number of lines to scroll is always 1. Actual scrolling takes place when ESC + 96H or ESC + 97H is executed.	Specifies the number of lines to be scrolled. (Command sequence) 1st byte: ESC 2nd byte: 94H 3rd byte: Number of lines, n (1 ≤ n ≤ 15) Does nothing when an illegal number of columns is specified.	
ESC 95H	SET SCROLL MODE	Does nothing.	Does nothing.	
ESC 96H	SCROLL UP n LINE	Moves up the screen on which the cursor rests by one line. The cursor is held in the original position on the screen.	Moves up the screen on which the cursor rests by n lines. The cursor is held in the original position on the screen. The value of n equals the value specified in ESC + 94H. The default value is 1.	
ESC 97H	SCROLL DOWN n LINE	Moves down the screen on which the cursor rests by one line. The cursor is held in the original position on the screen.	Moves down the screen on which the cursor rests by n lines. The cursor is held in the original position on the screen. The value of n equals the value specified in ESC + 94H. The default value is 1.	

CONOUT SPECIFICATIONS (7)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 0A0H	INS LED ON	Same as in system mode.	Same as in system mode.	
ESC 0A1H	INS LED OFF	Same as in system mode.	Same as in system mode.	
ESC 0A2H	CAPS LOCK LED ON	Same as in system mode.	Same as in system mode.	
ESC 0A3H	CAPS LOCK LED OFF	Same as in system mode.	Same as in system mode.	
ESC 0A4H	NUM LED ON	Same as in system mode.	Same as in system mode.	
ESC 0A5H	NUM LED OFF	Same as in system mode.	Same as in system mode.	
ESC 0B0H	FUNCTION KEY CHECK MODE ON	Same as in system mode.	Same as in system mode.	
ESC 0B1H	FUNCTION KEY CHECK MODE OFF	Same as in system mode.	Same as in system mode.	

ESC
0C0H
	CHANGE KANJI MODE	Switches the screen mode between kanji and non-kanji.
		(Command sequence)
		1st byte: ESC
		2nd byte: 0C0H
		3rd byte: Mode
		    non-zero = kanji mode
		    zero = non-kanji mode
		If the mode is switched from non-kanji to kanji, the screen is initialized in the screen mode 4 in which 60 half-size character images (30 kanji images) can be displayed in a line and in which the screen is ready for phrase translation. When the mode is switched from kanji to non-kanji, screen mode 0, VS1=25, VS2=23, VS1, block blink cursor, and tracking mode are selected.
			Switches the screen mode between kanji and non-kanji.	

ESC 0C0H	SCREEN ALL CLEAR AND MASKING GUIDE	Clears the entire screen and masks all system areas.	Same as in mode 4.	

CONOUT SPECIFICATIONS (8)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 0C2H	HORIZONTAL DOT LINE WRITE	Draws a horizontal dot line on the LCD screen. (Command sequence) 1st byte: ESC 2nd byte: 0C2H 3rd byte: Starting point (vertical) n1 4th byte: Starting point (horizontal) < H > n2 5th byte: Starting point (horizontal) < L > 6th byte: Length < H > n3 7th byte: Length < L > $0 \leq n1 \leq 63$ $0 \leq n2 \leq 479$ $1 \leq n3 \leq 480$ This function does nothing when the line goes beyond the screen.	Same as in mode 4.	Nothing is executed in old ASCII version (M2503CA). In new ASCII version (M2503CB), the data specified in the third and subsequent parameters is displayed as is.
ESC 0C3H	HORIZONTAL DOT LINE ERASE	Draws a horizontal space dot line on the LCD screen. (Command sequence) 1st byte: ESC 2nd byte: 0C3H 3rd byte: Starting point (vertical) n1 4th byte: Starting point (horizontal) < H > n2 5th byte: Starting point (horizontal) < L > 6th byte: Length < H > n3 7th byte: Length < L > $0 \leq n1 \leq 63$ $0 \leq n2 \leq 479$ $1 \leq n3 \leq 480$	Same as in mode 4.	Nothing is executed in old ASCII version (M2503CA). In new ASCII version (M2503CB), the data specified in the third and subsequent parameters is displayed as is.

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
		This function does nothing when the line goes beyond the screen.		
ESC 0C4H	VERTICAL DOT LINE WRITE	Draws a vertical dot line on the LCD screen. (Command sequence) 1st byte: ESC 2nd byte: ØC4H 3rd byte: Starting point (vertical) n1 4th byte: Starting point (horizontal) < H > n2 5th byte: Starting point (horizontal) < L > 6th byte: Length < H > n3 $0 \leq n1 \leq 63$ $0 \leq n2 \leq 479$ $1 \leq n3 \leq 64$ This function does nothing when the line goes beyond the screen.	Same as in mode 4.	Nothing is executed in old ASCII version (M25Ø3ØCA). In new ASCII version (M25Ø3ØCB), the data specified in the third and subsequent parameters is displayed as is.

CONOUT SPECIFICATIONS (9)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 0C5H	VERTICAL DOT LINE ERASE	Draws a space dot line in the vertical direction on the LCD screen. (Command sequence) 1st byte: ESC 2nd byte: ØC5H 3rd byte: Starting point (vertical) nl 4th byte: Starting point (horizontal) < H > n2 5th byte: Starting point (horizontal) < L > 6th byte: Length < H > n3 7th byte: Length < L > Ø ≦ nl ≦ 63 Ø ≦ n2 ≦ 479 1 ≦ n3 ≦ 480 This function does nothing when the line goes beyond the screen.	Same as in mode 4.	Nothing is executed in old ASCII version (M25Ø3ØCA). In new ASCII version (M25Ø3ØCB), the data specified in the thrid and subsequent parameters is displayed as is.

6-54

CONOUT SPECIFICATIONS (9)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 0C6H	DOT LINE WRITE	Draws a line of user-specified dot pattern on the LCD screen. (Command sequence) 1st byte: ESC 2nd byte: 0C6H 3rd byte: Starting position (horizontal) < H > m1 4th byte: Starting position (horizontal) < L > 5th byte: Starting position (vertical) < H > m2 6th byte: Starting position (vertical) < L > 7th byte: Ending position (horizontal) < H > n1 8th byte: Ending position (horizontal) < L > 9th byte: Ending position (vertical) < H > 10th byte: Ending position (vertical) < L > 11th bute: Mask pattern 1 12th byte: Mask pattern 2 13th byte: Operation where |m1-n1| < 16383 |m2-n2| < 16383 The pattern must be specified in bit image in 16 bits from mask pattern 1, bit 7 through mask pattern 2, bit 0. Line segments are masked sequentially by this mask pattern in the mode specified by operation. The dot coordinates of a slanted line are automatically calculated by the function. (Operation) Operation specifies the mode in which the dot coordinates corresponding to the 1 bits of the (16) mask pattern are to be masked. The operation codes are: 01H = Off 02H = On 03H = Complement See separate sheets for dot coordinates.		

6-55

CONOUT SPECIFICATIONS (1∅)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS

ESC 0C7H	PSET/PRESET	Sets and resets the specified point on the LCD screen.
		(Command sequence)
		1st byte: ESC
		2nd byte: ∅C7H
		3rd byte: Function code
		4th byte: Plot position (vertical) nl
		5th byte: Plot position (horizontal) < H > n2
		6th byte: Plot position (horizontal) < L >
		
		    ∅ ≤ nl ≤ 63,
		    ∅ ≤ n2 ≤ 479
		Function code: l = PSET (ON)
		                ∅ = PRESET (OFF)
			Same as in mode 4.	

ESC 0C8H	CHARACTER SIZE WIDE	Sets the screen character display size to double width.	Same as in mode 4.	
ESC 0C9H	CHARACTER SIZE NARROW	Sets the screen character display size to full width.	Same as in mode 4.	
ESC 0CBH	SET KEISEN MODE	Sets and resets the ruler line mode for the ruler line data. (Command sequence) 1st byte: ESC 2nd byte: ∅CBH 3rd byte: nonzero = ruler line mode on. zero = ruler line mode off. The ruler line data includes the following: (SHIFT JIS CODE) 83F9H - 83FCH	Same as in mode 4.	

CONOUT SPECIFICATIONS (1∅)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
		849FH - 84AEH 84B1H - 84BBH 84C3H - 84F4H F74∅H - F747H		

CONOUT SPECIFICATIONS (11)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC OCCH	CONTROL GUIDE DISPLAY SET/RESET	Turns on or off display on the system area. (Command sequence) 1st byte: ESC 2nd byte: ØCCH 3rd byte: BIT 7 ... Shift mode (1-6 coloumns) BIT 6 ... Input data display (15-53 columns) BIT 5 ... Entry subguide (54-6Ø columns) BIT 4 ... Convert mode guide (11-12 columns) BIT 3 ... Guide line Data is masked if the corresponding bit is on, and displayed if the bit is off. More than one mode can be specified.		
ESC OCDH	CLEAR GUIDE	Clears vertical dot lines 46 through 64 (the area below the guide line).	Does nothing.	
ESC OCEH	CHANGE CONVERT MODE	Switches between the convert and nonconvert modes. (Command sequence) 1st byte: ESC 2nd byte: ØCEH 3rd byte: Change information BIT Ø ... ON: Convert mode OFF: Nonconvert mode BIT 6 ... ON: Enables convert mode switching from the keyboard. BIT 7 ... ON: Disables convert mode switching from the keyboard. Bit 7 takes precedence over bit 6 when both bits are on. The keyboard buffer is cleared each time the mode is switched from convert to nonconvert and vice versa. Actual switching takes place when CONIN or CONST is executed. The guide line display is also updated at this time.		

6-58

CONOUT SPECIFICATIONS (12)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 0D0H	DISPLAY MODE SET	Switches the screen mode between normal (screen 4) and zoom (screen 5). (Command sequence) 1st byte: ESC 2nd byte: 0D0H 3rd byte: Mode 04H: Normal mode 05H: Zoom mode 4th byte: Number of columns 1-60 in normal mode 1-100 in zoom mode Switching into the same mode is regarded as changing the number of columns and the cursor is placed in the home position.		
ESC 0D1H	SELECT DISPLAY SCREEN	Does nothing.	Does nothing.	
ESC 0D2H	DIRECT DISPLAY OF PHYSICAL SCREEN	Displays a character in the specified position on the real screen. (Command sequence) 1st byte: ESC 2nd byte: 0D2H 3rd byte: Row position (1-3 lines) 4th byte: Column position (1-60 columns) 5th byte: Attribute zero: Full width Nonzero: Double width 6th byte: Display specification zero: User area Nonzero: Guide area 7th byte: Character code HIGH 8th byte: Character code LOW When the 7th byte is a code from 20H to 7FH or from 0A0H to 0DFH, this function regards the code as consisting only of one byte and ignores the 8th byte. When the 7th byte is a code from 00H to 1FH, the function takes it as an error and does nothing. The display position may be set to any location on the screen with the line and column numbers.		

CONOUT SPECIFICATIONS (12)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
		When the 6th byte indicates the user area, this function displays the character in 16 × 22 dot matrix according to the specified attributes. When the 6th byte indicates the guide area, the function displays the character in 16 × 18 dot matrix ignoring the attribute. When the 3rd line is specified as the user area, it is treated as the guide line.		
ESC 0D3H	SELECT FUNCTION KEY DISPLAY	Does nothing.	Does nothing.	
ESC 0D4H	LOCATE TOP OF SCREEN	Does nothing.	Does nothing.	
ESC 0D5H	LOCATE END OF SCREEN	Does nothing.	Does nothing.	

CONOUT SPECIFICATIONS (13)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 0D6H	SELECT CURSOR KIND	Does nothing. (Set to the nonblink underline cursor.)	Does nothing. (Set to the 4×4 dot blinking block cursor.)	
ESC 0D7H	FIND CURSOR	Does nothing.	Does nothing.	
ESC 0E0H	SET DOWN LOAD CHARACTER	Does nothing.	Does nothing.	
ESC 0F0H	KEYBOARD REPEAT ON/OFF	Controls the keyboard repeat function (receiving key code repeatedly while a key is held). (Command sequence) 1st byte: ESC 2nd byte: 0F0H 3rd byte: 0 (repeat off) 1 (repeat on) The default is repeat on.	Same as in mode 4.	
ESC 0F1H	SET KEYBOARD REPEAT START TIME	Specifies the keyboard repeat start time (interval between the time the first data is entered and the time the second data is taken when the key is held). (Command sequence) 1st byte: ESC 2nd byte: 0F1H 3rd byte: n (n/64 SEC) $(1 \leq n \leq 127)$ The default is approx. 656 ms.	Same as in mode 4.	
ESC 0F2H	SET KEYBOARD REPEAT INTERVAL TIME	Specifies the keyboard repeat interval time. (Command sequence) 1st byte: ESC 2nd byte: 0F2H 3rd byte: n (n/256 SEC) $(1 \leq n \leq 127)$ The default is approx. 70 ms.	Same as in mode 4.	

6-61

CONOUT SPECIFICATIONS (14)
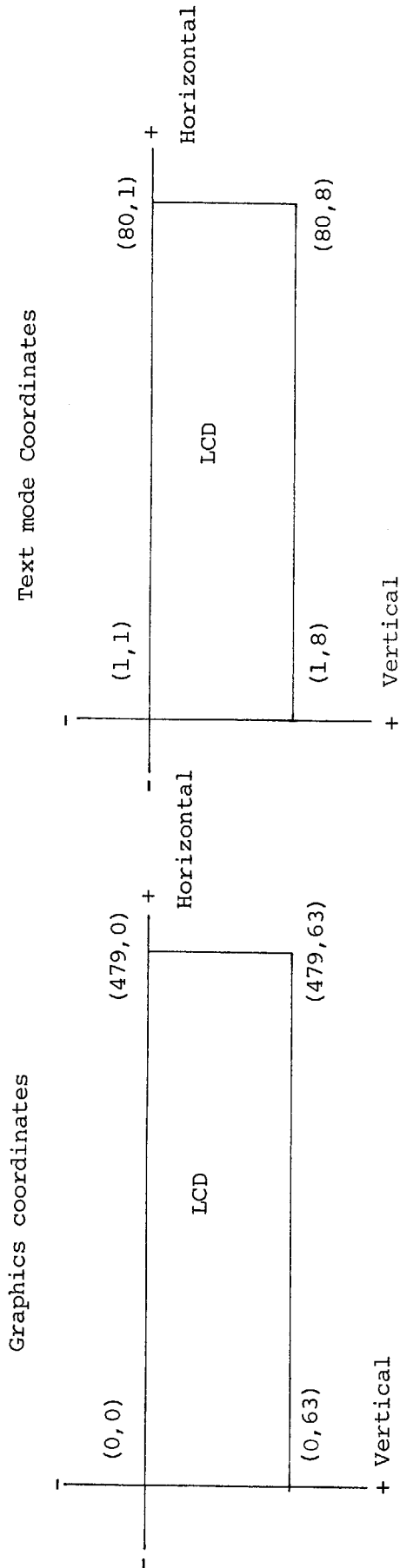
CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 0F3H	SET ARROW KEY CODE	Defines the arrow key codes. (Command sequence) 1st byte: ESC 2nd byte: 0F3H Default 3rd byte: Code of → --- 1CH 4th byte: Code of ← --- 1DH 5th byte: Code of ↑ --- 1EH 6th byte: Code of ↓ --- 1FH See "Arrow Key Function Chart" for details.		See "Arrow Key Function Chart".
ESC 0F4H	SET SCROLL KEY CODE	Defines codes for SHIFT + arrow key combinations. (Command sequence) 1st byte: ESC 2nd byte: 0F4H Default 3rd byte: Code of SHIFT + → --- 80H 4th byte: Code of SHIFT + ← --- 80H 5th byte: Code of SHIFT + ↑ --- 0F8H 6th byte: Code of SHIFT + ↓ --- 0F9H See "Arrow Key Function Chart" for details.		See "Arrow Key Function Chart".
ESC 0F5H	SET CTRL KEY CODE	Defines codes for CTRL + arrow key combinations. (Command sequence) 1st byte: ESC 2nd byte: 0F4H Default 3rd byte: Code of CTRL + → --- 0FFH 4th byte: Code of CTRL + ← --- 0FEH 5th byte: Code of CTRL + ↑ --- 0FAH 6th byte: Code of CTRL + ↓ --- 0FBH See "Arrow Key Function Chart" for details.		See "Arrow Key Function Chart".
ESC 0F6H	CLEAR KEY BUFFER	Same as in system mode.	Same as in system mode.	
ESC 0F7H	SET KEY SHIFT	Same as in system mode.	Same as in system mode.	

CONOUT SPECIFICATIONS (15)

CODE	FUNCTION	MODE 4	MODE 5	REMARKS
ESC 0F8H	SCREEN DUMP OF 44 DOT-LINE	Takes a dump of 1 to 44 vertical dot lines (not including the guide line) on the LCD screen.	This function only accepts the sequence but carries out no actual dump. (Actual dump is initiated when the user later sets the screen to mode 4 with ESC + ØDØH.)	
ESC 0F9H	SCREEN DUMP OF 64 DOT-LINE	Takes a dump of 1 to 64 vertical dot lines (including the guide line) on the LCD screen.	This function only accepts the sequence but carries out no actual dump. (Actual dump is initiated when the user later sets the screen to mode 4 with ESC + ØDØH.)	

Note: Starting and ending positions in the ESC Ø6CH (Dot Line Write sequence)

ESC+Ø6CH draws a line across the specified two points assuming that the LCD screen exists at the coordinates shown below. Values are taken to be negative when MSB is 1. Negative values are represented in two's complement form.

Graphics coordinates

```
       (0,0)          Horizontal     (479,0)
      ┌──────┬──────────────────────────────┐+
      │-     -│                              │
      │       │            LCD               │
      │       │                              │
      │-      │                              │
      └──────┴──────────────────────────────┘
       (0,63)                          (479,63)

      + Vertical
```

Text mode Coordinates

```
       (1,1)          Horizontal      (80,1)
      ┌──────┬──────────────────────────────┐+
      │-     -│                              │
      │       │            LCD               │
      │       │                              │
      │       │                              │
      └──────┴──────────────────────────────┘
       (1,8)                            (80,8)

      + Vertical
```

6-64

Arrow Key Function Chart

The arrow keys (including the shift keys) may be set to any codes using the above listed ESC sequences. Some special codes are used to control the screen directly. The codes and functions are listed below.

CODE	FUNCTION	DEFAULT KEY
00H . . 1BH	See the CONOUT Specifications	
1CH	CURSOR RIGHT	→
1DH	CURSOR LEFT	←
1EH	CURSOR UP	↑
1FH	CURSOR DOWN	↓
20H . . 7FH	See the CONOUT Specifications	
80H	Does nothing.	SHIFT/ → SHIFT/ ←
81H . . F7H	See the CONOUT Specifications	
F8H	Scroll Up One Line	SHIFT/ ↑
F9H	Scroll Down One Line	SHIFT/ ↓
FAH	Page Up	CTRL/ ↑
FBH	Page Down	CTRL/ ↓
FCH	TOP OF SCREEN	
FDH	BOTTOM OF SCREEN	
FEH	Switch To VS1	CTRL/ ←
FFH	Switch To VS2	CTRL/ →

Keys are disabled when they are set to 80H. When a key is set to one of the codes F8H through FFH, it is disabled and the screen can be controlled directly by the user.

ESC Sequence Parameter Table

The table below lists the ESC sequences for which the MAPLE does
nothing and discards any parameters.

(ESC sequences and the number of parameters that are ignored)

NUMBER	CODE	KANA MODE	KANJI MODE	REMARKS
1	ESC "%"	−	1	Access CG ROM directly
2	ESC "C"	1	1	Set character
3	ESC "L"	2	2	Change CRT Color In the new ASCII version (M2503ØCB) kana mode, the parameters are displayed as they are (garbage data).
4	ESC 80H	1	−	Code conversion In the new ASCII version (M2503ØCB) kana mode, the parameters are displayed as they are (garbage data).
5	ESC 81H	1	−	Code conversion In the new ASCII version (M2503ØCB) kana mode, the parameters are displayed as they are (garbage data).
6	ESC 82H	2	−	Code conversion In the new ASCII version (M2503ØCB) kana mode, the parameters are displayed as they are (garbage data).
7	ESC 83H	2	−	JIS C6226 → SHIFT JIS
8	ESC 84H	2	−	SHIFT JIS → JIS C6226
9	ESC 90H	−	2	Partial Scroll Up
10	ESC 91H	−	2	Partial Scroll Down
11	ESC 92H	1	−	Scroll Right In the new ASCII version (M2503ØCB) kana mode, the parameters are displayed as they are (garbage data).

NUMBER	CODE	KANA MODE	KANJI MODE	REMARKS
12	ESC 93H	1	-	Scroll Left In the new ASCII version (M25Ø3ØCB) kana mode, the parameters are displayed as they are (garbage data).
13	ESC 95H	-	1	Set Scroll Mode
14	ESC 0C2H	5	-	Horizontal Line Write In the new ASCII version (M25Ø3ØCB) kana mode, the parameters are displayed as they are (garbage data).
15	ESC 0C3H	5	-	Horizontal Line Erase In the new ASCII version (M25Ø3ØCB) kana mode, the parameters are displayed as they are (garbage data).
16	ESC 0C4H	4	-	Vertical Line Write In the new ASCII version (M25Ø3ØCB) kana mode, the parameters are displayed as they are (garbage data).
17	ESC 0C5H	4	-	Vertical Line Erase In the new ASCII version (M25Ø3ØCB) kana mode, the parameters are displayed as they are (garbage data).
18	ESC 0CBH	1	-	Ruler Line Mode On/Off
19	ESC 0CCH	1	-	Set/Reset Guide Display
20	ESC 0CEH	1	-	Change Convert Mode
21	ESC 0D1H	-	1	Switch between VS1 and VS2
22	ESC 0D3H	-	1	Display Function Key
23	ESC 0D6H	-	1	Cursor Type
24	ESC 0E0H	-	9	User Defined Character

Note: "-" indicates that the MAPLE takes some action.