

(1) The 7508 loads the buffer with the codes of the keys that are pressed while interrupts are disabled.

(2) When the buffer becomes full, the 7508 does not buffer subsequent key codes. However, it places the code associated with the STOP key at the end of the buffer if the key is pressed.

(3) The 7508 generates an interrupt when the STOP key is pressed. The data in the 7508 buffer is then read by the main CPU starting at the beginning of the buffer.

To disable/enable key interrupts use is made of BIOS MASKI.

When an item keyboard is installed, however, the user cannot enable only STOP key interrupts. Attempting to do so disables all keyboard interrupts.

#### 3.5.2.5 LEDs

Both the standard keyboard and the item keyboard have three LEDs.

On the standard keyboard, the LEDs are used to indicate the CAPS and NUM key status. On the item keyboard, the LEDs are used to indicate the SHIFT key status.

The LEDs can be controlled by the user using the BIOS CONOUT routine. The user, however, should not use the LEDs that are used by the system.

### 3.5.3 Standard Keyboard

#### 3.5.3.1 Outline

- (1) Number of keys: 72
- (2) Number of switch keys: 6 (CTRL, SHIFT x 2, CAPS, GRPH, or NUM)
- (3) Number of LEDs: 3

#### 3.5.3.2 Key Functions

- (1) Auto repeat keys

All keys except the following 15 keys are repeatable:

STOP, ESC, PAUSE, HELP, PF1 - PF5, CTRL, SHIFT (left and right), CAPS, GRPH, NUM

- (2) Auto repeat control

The auto repeat on/off state, repeat start time, and repeat interval time can be changed using the BIOS CONOUT routine.

### 3.5.3.3 Special Keys

The PINE keyboard is furnished with 14 special keys: STOP, ESC, PAUSE, HELP, PF1 - PF5, INS, ←, →, ↑, and ↓. These keys function in different ways depending on the keyboard mode (normal, shift, or control).

| Key   | Mode    | Code | Function  |
|-------|---------|------|---|
| STOP  | Normal  | 03H  | Clear key buffer.                               |
|       | Shift   | 03H  | Clear key buffer.                               |
|       | Control | 03H  | Clear key buffer.<br>Terminate I/O immediately. |
| ESC   | Normal  | 1BH  | None.   |
|       | Shift   | 1BH  | None.   |
|       | Control | None | Call subroutine.                                |
| PAUSE | Normal  | 13H  | None.   |
|       | Shift   | 13H  | None.   |
|       | Control | None | Call subroutine.                                |
| HELP  | Normal  | 00H  | None.   |
|       | Shift   | 00H  | None.   |
|       | Control | None | Call subroutine.                                |
| PF1   | Normal  | -    | Function 1                                      |
|       | Shift   | -    | Function 6                                      |
|       | Control | None | Call subroutine.                                |
| PF2   | Normal  | -    | Function 2                                      |
|       | Shift   | -    | Function 7                                      |
|       | Control | None | Call subroutine.                                |
| PF3   | Normal  | -    | Function 3                                      |
|       | Shift   | -    | Function 8                                      |
|       | Control | None | Call subroutine.                                |
| PF4   | Normal  | -    | Function 4                                      |
|       | Shift   | -    | Function 9                                      |
|       | Control | None | Call subroutine.                                |
| PF5   | Normal  | -    | Function 5                                      |
|       | Shift   | -    | Function 10                                     |
|       | Control | None | Call subroutine.                                |
| INS   | Normal  | 08H  | None.   |
|       | Shift   | None | Switch cursor tracking mode.                    |
|       | Control | None | Display the screen containing the cursor.       |
| →     | Normal  | 1CH  | None.   |
|       | Shift   | None | Scroll the screen 20 columns to the right.      |
|       | Control | None | Scroll the screen 40 columns to the right.      |
| ←     | Normal  | 1DH  | None.   |
|       | Shift   | None | Scroll the screen 20 columns to the left.       |
|       | Control | None | Scroll the screen 40 columns to the left.       |
| ↑     | Normal  | 1EH  | None.   |
|       | Shift   | None | Scroll the screen one line up.                  |
|       | Control | None | Scroll the screen eight lines up.               |
| ↓     | Normal  | 1FH  | None.   |
|       | Shift   | None | Scroll the screen one line down.                |
|       | Control | None | Scroll the screen eight lines down.             |

The functions here refer to those functions which are automatically started by the system. They are fully described on the following pages. The above codes can be read by the CONIN routine.

### 3.5.3.4 STOP and CTRL/STOP

STOP and CTRL/STOP keys are processed by the same interrupt processing routine and perform the following functions:

STOP: Clears the key buffer and returns ^C (03H).

CTRL/STOP: Clears the key buffer and returns ^C (03H).

Also terminates the current I/O operation.

The following flags are set when the STOP or CTRL/STOP key is pressed:

BRKFLG (0F019H) 1 byte

- Indicates that the STOP key has been pressed.
  - = 00H: STOP key not pressed.
  - = 0FFH: STOP key pressed.

This flag is reset to 00H if the BIOS CONIN or CONST routine is called when the key buffer is empty.

CSTOPFLG (0F01AH) 1 byte

- Indicates that the CTRL/STOP has been pressed.
  - = 00H: CTRL/STOP not pressed.
  - = 01H: CTRL/STOP pressed.

This flag is reset to 00H if the BIOS CONIN or CONST routine is called when the key buffer is empty.

CSTOPMCT (0F01BH) 1 byte

- Is the flag for terminating microcassette processing.
  - = 00H: CTRL/STOP key not pressed.
  - = 01H: CTRL/STOP key pressed.

This flag is reset to 00H by the PSTBIOS routine.

CSTOPPRN (0F309H) 1 byte

- Is the flag for terminating cartridge printer processing.
  - = 00H: CTRL/STOP not pressed.
  - = 01H: CTRL/STOP pressed.

This flag is reset to 00H by the PSTBIOS routine.

During an I/O operation that takes a long processing time or that does not terminate until some conditions are met, the PINE checks CSTOPFLG and terminates the operation if the flag is set. The PINE references CSTOPMCT or CSTOPPRN when handling I/O operations on the microcassette or cartridge printer.

### 3.5.3.5 Programmable Function Keys

The keys PF1 to PF5 at the top of the keyboard are programmable. They are used with or without the SHIFT key and are numbered PF6 (SHIFT/PF1) to PF10 (SHIFT/PF5) when used with the SHIFT key. Any string of up to 15 characters can be assigned to each of these keys.

PFKs can be defined in the system by:

- Using the PUTPFK (WBOOT + 6CH) BIOS function.
- Rewriting the system PF key table.
- Rewriting the pointer to the system PF key table.

A detailed description of the definition methods follows.

(1) Using the PUTPFK

See the description about BIOS PUTPFK.

(2) Rewriting the PF key table

The PINE uses two types of PF key tables. One is named PFKTAB and the other WPFKTBL. WPFKTBL is referenced by the system whenever a PF key is pressed. PFKTAB contains the data that is used by WBOOT to initialize WPFKTBL when resident is not specified.

The relationship between these tables is shown in Figure 3.5.1.

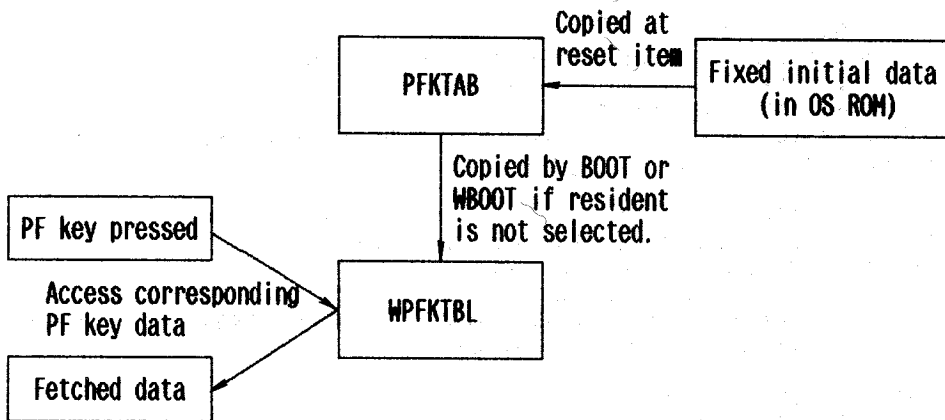


Fig. 3.5.1 Conceptual Diagram of the PF Key Tables

Change the data in WPFKTBL to make the PF key definitions valid only during execution of the application program. Change the data in PFKTAB to make the definition changes permanent (valid after warm boots).

(3) Rewriting the pointer to the PF key table

In the PINE OS, the starting address of the PF key table is stored in YPFKSTR (in Figure 3.5.1 the system references YPFKSTR to find the PF key table address each time it accesses PF key data). PF key definitions can be modified by changing the address in YPFKSTR.

The procedure for changing the pointer to the PF key table is given below.

- 1) Create a PF key table at location 8000H or higher in RAM. The table format is shown below.

|      | 1 byte               | 15 byte          |
|------|----------------------|------------------|
|      | Number of characters | Character string |
| PF1  |                      |                  |
| PF2  |                      |                  |
| PF3  | ⋮                    | ⋮                |
| ⋮    |                      |                  |
| PF10 | Number of characters | Character string |

2) Load 00H into the Number of characters field for unnecessary PF keys. No key code will be returned for these keys when they are pressed.

3) Load the table starting address into YPFKSTR.

This complete the table update process. YPFKSTR is initialized during a warm boot.

YPFKSTR (0F012H) 2 bytes

- PF key table starting address

The initial value is WPFKTBL (0F545H). (The table must be located at 8000H or higher location in RAM.)

PFKTAB (0F02DH) 160 bytes

- PF key initial data area

|      | 1 byte               | 15 byte          |
|------|----------------------|------------------|
|      | Number of characters | Character string |
| PF1  |                      |                  |
| PF2  |                      |                  |
| PF3  | ⋮                    | ⋮                |
| ⋮    |                      |                  |
| PF10 | Number of characters | Character string |

This table is copied into WPFKTBL during a cold or warm boot when resident is not specified.

WPFKTBL (0F545H) 160 bytes

- PF key table

|      | 1 byte               | 15 byte          |
|------|----------------------|------------------|
|      | Number of characters | Character string |
| PF1  |                      |                  |
| PF2  |                      |                  |
| PF3  | ⋮                    | ⋮                |
| ⋮    |                      |                  |
| PF10 | Number of characters | Character string |

### 3.5.3.6 Keys Calling a Subroutine

User-supplied service programs can be executed in the form of subroutine calls by pressing the specific keys.

When the ESC, PAUSE, HELP, or PF1 to PF5 is pressed with the CTRL key held down, the system calls the associated subroutine based on the call address stored in the subroutine table (YSUBRTN). The actual subroutine is executed when the BIOS CONIN or CONST is called.

Note the following points when modifying the table:

- 1) The modified call address must be 8000H or higher.
- 2) The called subroutine must reserve its own stack area when it is to use a large amount of stack area. The original stack must be restored on return.
- 3) In general, system calls such as BDOS and BIOS cannot be used in the subroutine. Call directly the corresponding module in OS ROM if necessary.
- 4) YSUBRTN is not initialized during a warm boot. Therefore, if the subroutine is to be used only in the application program, the table must be restored with the original values when the application program terminates. (Service program is not necessarily existed on the called address.)

YSUBRTN (0F0CDH) 16 bytes

- Subroutine table

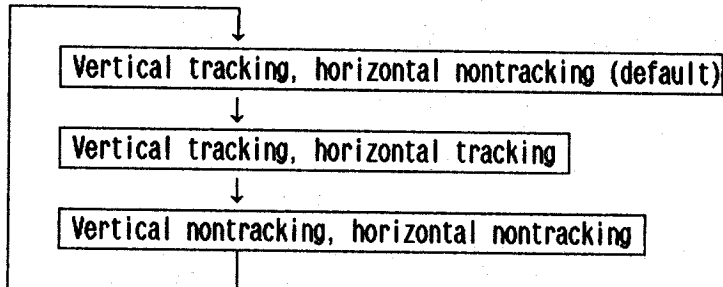
| Key sequence | Address | Default function                    |
|--------------|---------|-------------------------------------|
| CTRL/ESC     | 0F0CDH  | Does nothing but return.            |
| CTRL/PAUSE   | 0F0CFH  | Does nothing but return.            |
| CTRL/HELP    | 0F0D1H  | Performs system display processing. |
| CTRL/PF1     | 0F0D3H  | Does nothing but return.            |
| CTRL/PF2     | 0F0D5H  | Does nothing but return.            |
| CTRL/PF3     | 0F0D7H  | Does nothing but return.            |
| CTRL/PF4     | 0F0D9H  | Does nothing but return.            |
| CTRL/PF5     | 0F0DBH  | Performs screen dump processing.    |

### 3.5.3.7 Screen Control

The PINE uses the concept of virtual screen to implement CRT functions. Some keys are assigned special functions such as switching screen modes or defining the window to be displayed.

(1) SHIFT/INS (Switches the track mode.)

The PINE provides three scroll modes for the window screen. The scroll mode changes cyclically as the SHIFT/INS is pressed.



These modes control the behavior of the screen when the cursor goes out of the window. See Section 3.6, "LCD Display" for more information.

(2) CTRL/INS (Displays the cursor screen.)

This key sequence displays the portion of the screen in which the cursor is presently positioned regardless of the current scroll mode. It does nothing if the cursor is in the current screen.

(3) SHIFT/arrow keys, CTRL/arrow keys (Scrolls the screen.)

The SHIFT or CTRL key is pressed in combination with one of the arrow keys to display the portion of the virtual screen other than the window that is currently being displayed.

|         |   |
|---------|---|
| SHIFT/→ | Scrolls the window 20 columns to the right. Does nothing when the virtual screen is 40 columns wide or the right screen is already displayed. |
| SHIFT/← | Scrolls the window 20 columns to the left. Does nothing when the virtual screen is 40 columns wide or the left screen is already displayed.   |
| SHIFT/↑ | Scrolls the window one line up. Does nothing when the window is already at the top of the virtual screen.                                     |
| SHIFT/↓ | Scrolls the window one line down. Does nothing when the window is already at the bottom of the virtual screen.                                |
| CTRL/→  | Displays the right screen when the virtual screen is 80 columns wide. Does nothing if the right screen is already displayed.                  |
| CTRL/←  | Displays the left screen when the virtual screen is 80 columns wide. Does nothing if the left screen is already displayed.                    |
| CTRL/↑  | Scrolls up the window one screenful of lines. Display starts at the top of the virtual screen when it moves up beyond the top.                |
| CTRL/↓  | Scrolls down the window one screenful of lines. Display starts at the bottom of the virtual screen when it moves down the bottom.             |

### 3.5.3.8 Changing Codes

The key codes for arrow keys, SHIFT/arrow keys, and CTRL/arrow keys can be changed using BIOS CONOUT or by rewriting the associated areas. See Section 3.6 for BIOS CONOUT.

The user can assign any code to each of these keys. When 80H or 0F8H to 0FFH is entered, however, the OS takes special actions and does not return the code.

| Code | Function   |
|------|--|
| 80H  | Does nothing. The key entry is regarded as invalid.        |
| 0F6H | Changes the screen scroll mode.                            |
| 0F7H | Displays the screen containing the cursor.                 |
| 0F8H | Scrolls the window horizontally 20 columns to the right.   |
| 0F9H | Scrolls the window horizontally 20 columns to the left.    |
| 0FAH | Scrolls the window one line up.                            |
| 0FBH | Scrolls the window one line down.                          |
| 0FCH | Displays the right screen of the 80-column virtual screen. |
| 0FDH | Displays the left screen of the 80-column virtual screen.  |
| 0FEH | Scrolls up the window a screenful of lines.                |
| 0FFH | Scrolls down the window a screenful of lines.              |

The areas to be rewritten when changing key codes are listed below. The initial values in the list are the copy of the 14 bytes starting at DFLTRT (0F01FH). They are copied during a cold or warm boot if resident is not specified.

The changed key codes can be made permanent over a reset or warm boot by changing the contents of the 14-byte area starting at DFLTRT.

CHGRIGT (0F328H) 1 byte  
- Code for the ← key.  
Initial value is 1CH.

CHGLEFT (0F329H) 1 byte  
- Code for the → key.  
Initial value is 1DH.

CHGUP (0F32AH) 1 byte  
- Code for the ↑ key.  
Initial value is 1EH.

CHGDOWN (0F32BH) 1 byte  
- Code for the ↓ key.  
Initial value is 1FH.

CHGSRIGT (0F32CH) 1 byte  
- Code for the SHIFT/→ key.  
Initial value is 0F8H.

CHGSLEFT (0F32DH) 1 byte  
- Code for the SHIFT/← key.  
Initial value is 0F9H.

CHGSUP (0F32EH) 1 byte  
- Code for the SHIFT/↑ key.  
Initial value is 0FAH.

CHGSDOWN (0F32FH) 1 byte  
- Code for the SHIFT/↓ key.  
Initial value is 0FBH.



CHGSSCRN (0F330H) 1 byte  
- Code for the SHIFT/INS key.  
Initial value is 0F6H.

CHGCRIGT (0F331H) 1 byte  
- Code for the CTRL/→ key.  
Initial value is 0FCH.

CHGCLEFT (0F332H) 1 byte  
- Code for the CTRL/← key.  
Initial value is 0FDH.

CHGCUP (0F333H) 1 byte  
- Code for the CTRL/↑ key.  
Initial value is 0FEH.

CHGCDOWN (0F334H) 1 byte  
- Code for the CTRL/↓ key.  
Initial value is 0FFH.

CHGCSCRN (0F335H) 1 byte  
- Code for the CTRL/INS key.  
Initial value is 0F7H.

### 3.5.3.9 Key Interrupt Processing

The following table shows the actions taken by the keyboard interrupt processing routines, BIOS CONIN, and CONOUT in response to key entries.

#### STOP, CTRL/STOP

- Interrupt processing routine: Clears the key buffer and loads it with the position code.
- CONST: Returns control to the calling program with 0FFH in the A register.
- CONIN: Returns control to the calling program with 03H in the A register.

#### ESC, PAUSE, HELP, INS

- Interrupt processing routine: Loads the key buffer with the position code.
- CONST: Returns control to the calling program with 0FFH in the A register.
- CONIN: Returns control to the calling program with the key code in the A register.

#### Subroutine call

- Interrupt processing routine: Loads the key buffer with the position code.
- CONST: Performs the corresponding processing and checks for the presence of keyed data.
- CONIN: Performs the corresponding processing and waits for keyed data.

#### Code programmable keys (0F6H - 0FFH, 80H)

- Interrupt processing routine: Loads the key buffer with the position code.
- CONST: Performs the corresponding processing and checks for the presence of keyed data.
- CONIN: Performs the corresponding processing and waits for keyed data.

Code programmable keys (other than above codes)

- Interrupt processing routine: Loads the key buffer with the position code.
- CONST: Returns control to the calling program with 0FFH in the A register.
- CONIN: Returns control to the calling program with the key code in the A register.

PF keys

- Interrupt processing routine: Loads the key buffer with the position code.
- CONST: Returns control to the calling program with 0FFH in the A register.
- CONIN: Returns control to the calling program with the first one character in the A register.

Switch keys

- Interrupt processing routine: Loads the key buffer with a make or a break code.
- CONST: Determines the keyboard state and checks for keyed data again.
- CONIN: Determines the keyboard state and waits for keyed data again.

Other keys

- Interrupt processing routine: Loads the key buffer with the position code.
- CONST: Returns control to the calling program with 0FFH in the A register.
- CONIN: Returns control to the calling program with the key code in the A register.

### 3.5.3.10 Key Buffers

The PINE has two key buffers: one in the main CPU (Z-80) and the other in the 7508.

#### (1) 7508 key buffer

The 7508 key buffer may be seven characters (default) or one character wide. The size can be specified by sending the following commands to the 7508:

0CH: Reserves a 7-character area as the key code buffer.

1CH: Reserves a 1-character area as the key code buffer.

These commands are sent to the 7508 by the WRT7508 routine cataloged the system jump table. See Section 4.2, "Jump Tables" for details.

#### (2) Main key buffer

BIOS reserves a 33-byte area as the buffer to buffer key code of up to 32 characters. Figure 3.5.2 shows the structure of key buffer KBUF (0F860H).

The  
fr  
l)  
rou  
YIN  
po  
  
2)  
has  
the  
ad  
one  
  
KBU  
- K  
The  
byt  
  
YIN  
- K  
The  
pro  
inc  
  
E  
YKE  
- K  
The  
is  
pre  
  
T  
T  
C  
W  
K

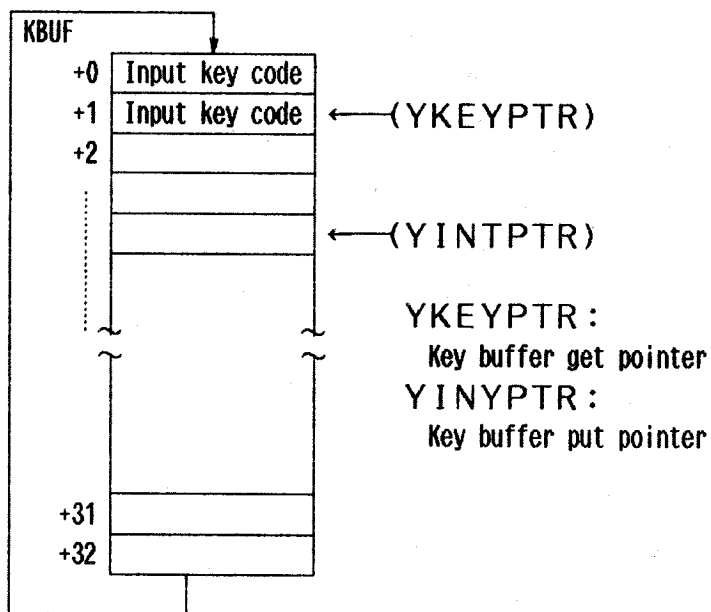


Fig. 3.5.2 Key buffer Structure

The key buffer is of the ring structure. Input or output to or from the key buffer is performed in the following sequence:

1) On receiving a keyboard interrupt, the keyboard interrupt processing routine loads the key code into the address pointed to by YINTPTR and increments the pointer by one. When the incremented pointer matches YKEYPTR, the routine discards the input key code.

2) The key entry processing routine determines that no key entry has been made if YKEYPTR and YINTPTR have the same value. If their values differ, the routine fetches the key code at the address pointed to by YKEYPTR and then increments the pointer by one.

KBUF (0F860H) 33 bytes

- Key buffer for key entries

The key buffer is 33 bytes long, it can actually hold only 32 bytes of data at most.

YINTPTR (0F00BH) 2 bytes

- Key buffer put pointer

The pointer to the buffer location into which the keyboard interrupt processing routine puts the received key code. The pointer is incremented by the keyboard interrupt processing routine.

YKEYPTR (0F00DH) 2 bytes

- Key buffer get pointer

The pointer to the buffer location from which the input key code is gotten. YINTPTR = YKEYPTR indicates that no key code is present in the buffer.

### 3.5.3.11 Automatic Key Entry at Power-on Time

The PINE provides special functions to enter a predefined string automatically at power-on time: the auto start function and the wake function. They load an auto start string and a wake string, respectively, when power is switched on in the restart mode.

Both strings may be as long as 32 characters. Any defined string is treated in the same way as keyed-in data and used as parameters to be placed on the menu or CCP command line or interpreted as parameters after a transient command is activated. The wake string can be defined using the system display or BIOS TIMDAT. The auto start string can be defined using the system display or BIOS AUTOST function. See Section 2.7, "Auto Start Function" and Section 2.9, "Alarm/Wake" for details.

### 3.5.3.12 Keyboard States

The keyboard state is determined by the combination of the switch key (CTRL, SHIFT, or GRPH) that is pressed and the current keyboard shift mode (Normal, Caps, Num).

The switch keys are assigned precedence in the decreasing order of CTRL, GRPH, and SHIFT.

Table 3.5.3 lists the keyboard states.

| Switch key \ Mode | NORMAL                      | CAPS       | NUM       |
|-------------------|-----------------------------|------------|-----------|
|                   | Normal                      | Caps       | Num       |
| CTRL              | Control                     |            |           |
| SHIFT             | Normal Shift                | Caps Shift | Num Shift |
| GRPH              | Graphic                     |            |           |
| CTRL+SHIFT        | Control Shift (Item mode) * |            |           |
| CTRL+GRPH         | Control                     |            |           |
| SHIFT+GRPH        | Graph Shift                 |            |           |
| CTRL+SHIFT+GRPH   | Control Shift (Item mode) * |            |           |

Table 3.5.3 Keyboard States

\*: See 3.5.3.14 for item mode.

### 3.5.3.13 Mode Transition

The mode transition diagrams for the PINE keyboard are shown in Figures 3.5.4.

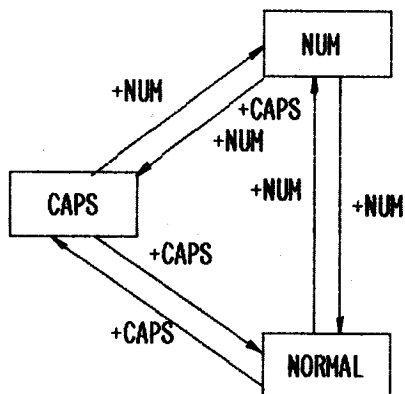


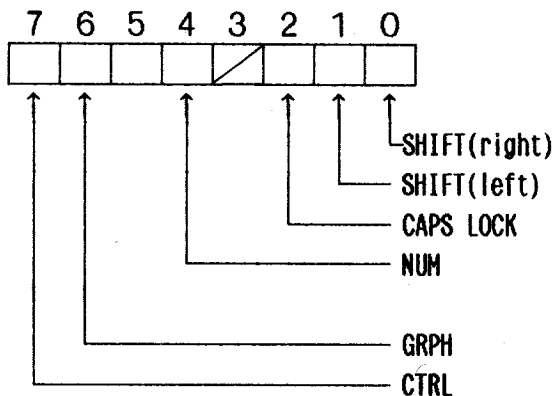
Fig. 3.5.4 Mode Transition (Export Version OS)

When power is turned on, the keyboard mode is first set to the mode that was established when power was turned off. The mode is set to Normal when a reset occurs.

The current keyboard state can be identified by checking the value of YSHFDT. YSHFDT contains the status of all switch keys. When referencing YSHFDT, it is necessary to set up the current keyboard state by running BIOS CONST in advance.

YSHFDT (0F00FH) 1 byte

- Indicates the current shift mode.



Bit n = 1 indicates that the corresponding mode is ON. Bit n = 0 indicates that the mode is OFF. This variable has a value of 00H when the keyboard is in the Normal mode.

### 3.5.3.14 Item Function

The PINE OS supports the item function on the standard keyboard. The item function defines character strings of up to 15 characters as key codes (40H - 7FH) and treats them in the same way as ordinary keyed-in strings. See Subsection 3.5.4, "Item Keyboard" for details.

### 3.5.4 Item Keyboard

#### 3.5.4.1 Outline

The PINE supports the item keyboard that allows the system to be used exclusively for specific applications.

The item keyboard consists of the item, numeric keypad, cursor movement key, and display sections. The keys in the item section can be redefined. Strings of up to 15 characters can be defined for a maximum of 62 items. This feature makes the item keyboard suitable for a wide variety of applications.

The user can easily replace the standard keyboard with the item keyboard. This means that the PINE can be used either for general or special purposes.

#### 3.5.4.2 Key configuration

The keys on the item keyboard are grouped as follows:

- Item section: 32 keys (including SHIFT)
- Numeric keypad section: 19 keys
- Cursor movement key section: 4 keys
- System key section: 3 keys
- LED section: 3 LEDs (one is used by the system)

The SHIFT key in the item section and the three keys in the system key section (STOP, INIT, and CTRL) are switch keys.

#### 3.5.4.3 Keyboard functions

The PINE keyboard has the following functions:

(1) Auto repeat function  
All keys except the SHIFT, STOP, INIT, and CTRL are repeatable. They are set to norepeat mode by default.

(2) Auto repeat control function  
The user can turn on and off the auto repeat function and specify the repeat start time and repeat interval time through BIOS CONOUT.

#### 3.5.4.4 Key specifications

##### (1) Keyboard layout

Figure 3.5.5 shows the layout of the item keyboard (furnished with an overlay sheet).

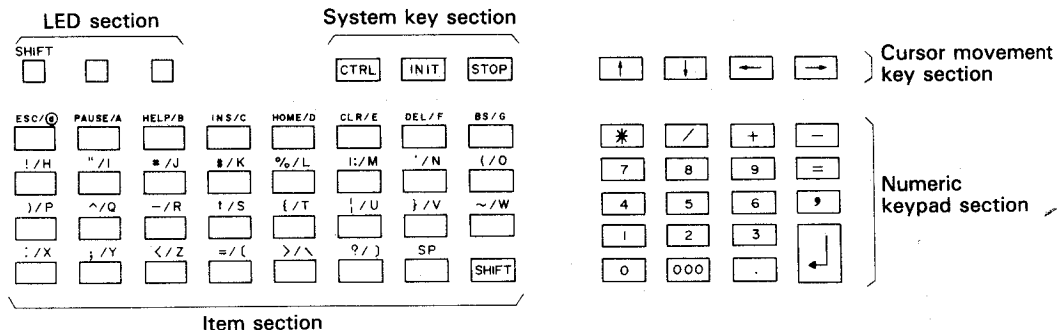


Fig. 3.5.5 Item Keyboard Layout

### 3.5.4.5 Item section

The item section consists of 32 keys including the SHIFT key. All item section keys other than the SHIFT key can be redefined. That is, 62 items can be defined using the combination of these 31 keys and the SHIFT key. The maximum length of the string defined for each key is 15 characters.

Except the SHIFT key, the 31 item keys have no label printed on their keytop.

The item keyboard is provided, as standard, with two types of overlay sheet: one for the system and the other for the user. The system overlay sheet contains the codes provided by the OS by default and is used for user program development and execution. The user overlay sheet is used during execution of user programs. The user must write the items on the sheet for himself.

The SHIFT key is a software lock key and alternates between the Normal and Shift modes. Its state is indicated by the left-most LED in the LED section. The shift status is held in IMSHFT (0F01CH).

IMSHFT (0F01CH) 1 byte  
- Item keyboard shift status  
  = 00H: Normal mode  
  = 20H: Shift mode

#### (1) Position codes

The key position code identifies the physical position of an item key. It is loaded in the C register when assigning an item function for a key using the BIOS PUTPFK routine.

Figure 3.5.6 shows the position codes in the Normal and Shift modes.

|     |     |     |     |     |     |     |       |
|-----|-----|-----|-----|-----|-----|-----|-------|
| 40H | 41H | 42H | 43H | 44H | 45H | 46H | 47H   |
| 48H | 49H | 4AH | 4BH | 4CH | 4DH | 4EH | 4FH   |
| 50H | 51H | 52H | 53H | 54H | 55H | 56H | 57H   |
| 58H | 59H | 5AH | 5BH | 5CH | 5DH | 5EH | SHIFT |

Normal mode

|     |     |     |     |     |     |     |       |
|-----|-----|-----|-----|-----|-----|-----|-------|
| 60H | 61H | 62H | 63H | 64H | 65H | 66H | 67H   |
| 68H | 69H | 6AH | 6BH | 6CH | 6DH | 6EH | 6FH   |
| 70H | 71H | 72H | 73H | 74H | 75H | 76H | 77H   |
| 78H | 79H | 7AH | 7BH | 7CH | 7DH | 7EH | SHIFT |

Shift mode

Fig. 3.5.6 Item Key Position Codes

(2) Default item key codes

The default item key codes are listed in Figure 3.5.7. The characters in the upper rows are those which printed on the overlay sheet. The numbers in the lower rows are key codes themselves.

|          |          |          |          |          |          |           |          |
|----------|----------|----------|----------|----------|----------|-----------|----------|
| @<br>40H | A<br>41H | B<br>42H | C<br>43H | D<br>44H | E<br>45H | F<br>46H  | G<br>47H |
| H<br>48H | I<br>49H | J<br>4AH | K<br>4BH | L<br>4CH | M<br>4DH | N<br>4EH  | O<br>4FH |
| P<br>50H | Q<br>51H | R<br>52H | S<br>53H | T<br>54H | U<br>55H | V<br>56H  | W<br>57H |
| X<br>58H | Y<br>59H | Z<br>5AH | [<br>5BH | \<br>5CH | ]<br>5DH | SP<br>20H | SHIFT    |

Normal mode

|            |              |             |            |             |            |            |           |
|------------|--------------|-------------|------------|-------------|------------|------------|-----------|
| ESC<br>1BH | PAUSE<br>13H | HELP<br>00H | INS<br>12H | HOME<br>0BH | CLR<br>0CH | DEL<br>7FH | BS<br>08H |
| !<br>21H   | ''<br>22H    | #<br>23H    | \$<br>24H  | %<br>25H    | &<br>26H   | '<br>27H   | (<br>28H  |
| )<br>29H   | ^<br>5EH     |             |            | {<br>7BH    | <br>7CH    | }<br>7DH   | ~<br>7EH  |
| :<br>3AH   | ;<br>3BH     | <<br>3CH    | =<br>3DH   | ><br>3EH    | ?<br>3FH   | SP<br>20H  | SHIFT     |

Shift mode

Fig. 3.5.7 Default Item Key Codes

Default key codes are set during system initialization, reset, or a warm boot started with resident not specified. They are also set when the keyboard is replaced by an item keyboard.

(3) Defining/cancelling item functions

Item functions can be defined or cancelled using the BIOS PUTPFK routine.

Item functions are set to their default values when an application program is activated. Therefore, the item functions must be defined or cancelled in the application program as required. It is desirable for some applications that unnecessary keys return no code. In such a case, the user should define only the necessary keys using the following steps:

- 1) Cancel all item functions.
- 2) Define only the necessary item functions.

See "BIOS PUTPFK" for the detailed procedure for defining and cancelling item functions.



### 3.5.4.6 Numeric keypad and cursor movement key sections

The numeric keypad and cursor movement key sections consist of 19 and 4 keys, respectively. When one of these keys is pressed, the code corresponding to the character printed on the keytop is returned.

Figure 3.5.8 shows the codes assigned to the keys in the numeric keypad and cursor movement key sections. The characters or symbols in the upper rows are also printed on the key top. The numbers in the lower rows are key codes.

|          |          |          |          |
|----------|----------|----------|----------|
| △<br>1EH | ▽<br>1FH | ◀<br>1DH | ▶<br>1CH |
| *<br>2AH | /<br>2FH | +<br>2BH | -<br>2DH |
| 7<br>37H | 8<br>38H | 9<br>39H | =<br>3DH |
| 4<br>34H | 5<br>35H | 6<br>36H | .<br>2CH |
| 1<br>31H | 2<br>32H | 3<br>33H | ↩<br>0DH |
| 0<br>30H | 000<br>  | .<br>2EH |          |

↪ 30H 30H 30H

Fig. 3.5.8 Key Codes of the Keys in the Numeric Keypad and Cursor Movement Sections

Basically, the functions of the keys in the numeric keypad and cursor movement key sections cannot be redefined (BIOS supports no routine for this purpose). To meet the requirements of a variety of applications, however, the PINE is designed so that the user can redefine the functions of the keys in these sections as he likes.

Redefinition of key functions is accomplished by directly rewriting the redefinition table (ITCDB) in the resident area in RAM. Any values from 00H through 0FFH can be specified. If 0FFH is specified for a key, three "0" (30H) codes are returned when that key is pressed.

ITCDB (0F0DDH) 24 bytes  
 - Item keyboard redefinition table

| Address | Key position | Printed character | Initial value |
|---------|--------------|-------------------|---------------|
| F0DEH   | 1            | △                 | 1EH           |
| F0DFH   | 2            | ▽                 | 1FH           |
| F0E0H   | 3            | ◀                 | 1DH           |
| F0E1H   | 4            | ▶                 | 1CH           |
| F0E2H   | 5            | *                 | 2AH           |
| F0E3H   | 6            | /                 | 2FH           |
| F0E4H   | 7            | +                 | 2BH           |
| F0E5H   | 8            | -                 | 2DH           |
| F0E6H   | 9            | 7                 | 37H           |
| F0E7H   | 10           | 8                 | 38H           |
| F0E8H   | 11           | 9                 | 39H           |
| F0E9H   | 12           | =                 | 3DH           |
| F0EAH   | 13           | 4                 | 34H           |
| F0EBH   | 14           | 5                 | 35H           |
| F0ECH   | 15           | 6                 | 36H           |
| F0EDH   | 16           | ,                 | 2CH           |
| F0EEH   | 17           | 1                 | 31H           |
| F0EFH   | 18           | 2                 | 32H           |
| F0F0H   | 19           | 3                 | 33H           |
| F0F1H   | 20           | 0                 | 30H           |
| F0F2H   | 21           | 000               | FFH           |
| F0F3H   | 22           | .                 | 2EH           |
| F0F4H   | 23           | ↶                 | 0DH           |

|    |    |    |    |
|----|----|----|----|
| 1  | 2  | 3  | 4  |
| 5  | 6  | 7  | 8  |
| 9  | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 23 |
| 20 | 21 | 22 |    |

### 3.5.4.7 System key section

Three system keys (CTRL, INIT, and STOP) make the standard system functions, except the system display and screen dump (they can be accomplished through BIOS), available to the user who uses an item keyboard. The procedures for using the standard system functions follow.

#### (1) System initialize

Press the reset switch while holding down the STOP and INIT keys. See Section 2.2, "System Initialize" for details of the system actions.

#### (2) Continue mode power off

Continue mode power-off is specified as standard when an item keyboard is installed. Even when this setting is changed to restart mode power-off, continue mode power-off will be restored if the power switch is turned off with the CTRL key held down. See Section 2.5, "Power-off" for system actions.

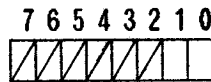
#### (3) STOP, CTRL/STOP

The STOP and CTRL/STOP keys on the item keyboard perform the same function as the STOP or CTRL/STOP key on the standard keyboard. These key functions can be disabled by setting the system function flag (ISYSREG) to 0.

#### (4) Programming notes

Both STOP and CTRL/STOP key functions are disabled when the PINE is enabled only for STOP key interrupts. The CTRL key on the item keyboard, unlike that on the standard keyboard, does not change the keyboard state.

ISYSREG (0F01EH) 1 byte  
- System function flag



STOP control bit

= 0 : Disable

= 1 : Enable (default)

CTRL / STOP control bit

= 0 : Disable

= 1 : Enable (default)

### 3.5.4.8 LED section

The item keyboard has three LEDs. One of them is used by the system to indicate the keyboard state. That LED indicates the Normal mode when it is lit and the Shift mode when it is turned off. The other two LEDs are available to the user.

Turning on and off the LED is controlled through BIOS CONOUT.

#### 3.5.4.9 Starting an item

When an item keyboard is installed, the PINE can be turned off, by default, in the continue mode when the power switch is turned off. Once an application is started, the PINE can resume processing in the mode that was established when power was turned off when the power switch is turned on. Therefore, the user need only consider the mode in which his application is to be started for the first time.

Application programs are stored in the following three storage devices:

1. External cassette (only BASIC)
2. Resident program
3. Drives (A: to K:)

(1) Starting an application program from the external cassette  
Select BASIC from the menu and load and execute the program from BASIC.

(2) Starting an application program in resident mode  
A BASIC or application program can be made resident by:  
1) starting it from the standard keyboard, 2) performing resident processing on the program, 3) turning off the power switch, and 4) switching to the item keyboard. The resident program is automatically activated when the power switch on the item keyboard is turned on.

(3) Starting an application program from a drive (A: - K:)  
Select the target program from the menu. This automatically starts the program. When an item keyboard is installed, only drives A: to D: and H: to K: are displayed in the menu.

#### 3.5.4.10 Terminating an item

(1) When an item keyboard is installed, the PINE, by default, is switched off in the continue mode only by turning off the power switch so that it can continue processing in the mode selected before power off when the power switch is turned on.

(2) Power must be turned off in the restart mode instead of continue mode depending on the application. The PINE power-off state are controlled through the continue flag. The continue flag can be manipulated by the application program using the BIOS CONTINUE routine or by directly rewriting the system area (IFRCECNT). See the descriptions about BIOS CONTINUE for details.

(3) In the restart mode, the application program can be automatically started (cold start) by defining an auto start string. After Resident processing, it can be warm-started simply by turning on the power switch.

### 3.5.4.11 Item functions on the standard keyboard.

The keys on the standard keyboard may also be used as item function keys by setting the item flag or pressing the CTRL and SHIFT keys simultaneously. In either case, the codes of the depressed keys are returned if no function is defined for them.

#### (1) Setting the item flag

Set the item flag (ITEMFLG) to 80H using the BIOS PUTPFK routine. The keys that are assigned a code from 40H to 7FH can be used as item function keys.

Note: Codes entered while pressing the shift key are not displayed even if defined.

#### (2) CTRL/SHIFT

Make key entries while pressing the CTRL and SHIFT keys (item mode) on the standard keyboard. The keys that will generate codes 40H to 7AH when pressed with neither CTRL nor SHIFT keys held down are defined as item function keys. Key codes 40H to 7AH correspond to the position codes of item functions.

In the CTRL SHIFT mode, keys with A to Z marked on the keytops return:

Uppercase characters 41H to 5AH in the CAPS mode.

Lowercase characters 61H to 7AH in the non-CAPS mode.

The keys associated with the position codes 40H and 5BH through 5EH return the same code regardless of the CAPS mode.

ITEMFLG (F01DH) 1 byte

- Item mode flag
- = 00H: Normal mode
- = 80H: Item mode

RWITEMTOP (0CC00H) 1024 bytes

- Item function key table

|     | 1byte                | 15byte           |
|-----|----------------------|------------------|
| 40H | Number of characters | Character string |
| 41H | Number of characters | Character string |
| ⋮   | ⋮                    | ⋮                |
| 7EH | Number of characters | Character string |
| 7FH | /                    | /                |

00H <= Number of characters <= 0FH

Keys whose number-of-characters field contains 00H are undefined and no code is returned when they are pressed.

### 3.5.4.12 Others

(1) When switching from standard keyboard to item keyboard, note that the old DIP switch settings specifying the character set are valid even after the keyboard is replaced (display character fonts vary depending on the specified character set).

(2) When the power switch is turned on, the PINE automatically determines whether an item keyboard is connected. If the keyboard is switched from standard to item keyboard when power is turned on, the system automatically performs a reset and initializes the item keyboard (loading default values).

(3) The default settings when the item keyboard is installed are as follows:

- Keys are not repeatable.
- Power is switched off in the continue mode when only the power switch is turned off.
- System function keys (STOP and CTRL/STOP) are enabled.
- Item function keys are assigned default codes.

(4) The functional differences between the item and standard keyboards are as follows:

- The screen dump subroutine cannot be called through the item keyboard CTRL and PF5 keys, though it can be called via BIOS.
- The system display function cannot be activated by the item keyboard CTRL and HELP keys. To execute this function, find the system display subroutine starting address from the CTRL/HELP address (0F0D1) in YSUBRTN and call the subroutine via the BIOS CALLX routine.

### 3.5.5 Work areas for keyboard processing

KBUF (0F860H) 33 bytes

- Input key buffer

Stores the key codes received from the 7508. The input key buffer is of the ring structure and controlled by the two pointers called YINTPTR and YKEYPTR. See 3.5.3.10 for the input key buffer.

DFLTRT (0F01FH) 14 bytes

- Loaded with the default codes of the cursor movement keys, SHIFT/INS, and CTRL/INS.

| Address | Default code | Key      | Address | Default code | Key       |
|---------|--------------|----------|---------|--------------|-----------|
| F01FH   | 1CH          | →        | F026H   | FBH          | SHIFT/ ↓  |
| F020H   | 1DH          | ←        | F027H   | F6H          | SHIFT/INS |
| F021H   | 1EH          | ↑        | F028H   | FCH          | CTRL/ →   |
| F022H   | 1FH          | ↓        | F029H   | FDH          | CTRL/ ←   |
| F023H   | F8H          | SHIFT/ → | F02AH   | FEH          | CTRL/ ↑   |
| F024H   | F9H          | SHIFT/ ← | F02BH   | FFH          | CTRL/ ↓   |
| F025H   | FAH          | SHIFT/ ↑ | F02CH   | F7H          | CTRL/INS  |

Even when this area is rewritten, the codes that the keys return are changed only at the next warm boot. See 3.5.3.8 for details.

PFKTAB (0F02DH) 160 bytes  
 - PF key default data table

|      | 1byte                | 15byte           |
|------|----------------------|------------------|
| PF1  | Number of characters | Character string |
| PF2  | Number of characters | Character string |
| PF3  | Number of characters | Character string |
| ⋮    | ⋮                    | ⋮                |
| PF10 | Number of characters | Character string |

The number of characters may be in the range from 00H to 0FH. 00H indicates that no string is defined. If this area is rewritten, the PF key data is changed at the next warm boot. See 3.5.3.5 for details.

YSUBRTN (0F0CDH) 16 bytes  
 - Subroutine call jump table

| Key for activating subroutine | Address | Default function         |
|-------------------------------|---------|--------------------------|
| CTRL/ESC                      | F0CDH   | Does nothing but return. |
| CTRL/PAUSE                    | F0CFH   | Does nothing but return. |
| CTRL/HELP                     | F0D1H   | System display.          |
| CTRL/PF1                      | F0D3H   | Does nothing but return. |
| CTRL/PF2                      | F0D5H   | Does nothing but return. |
| CTRL/PF3                      | F0D7H   | Does nothing but return. |
| CTRL/PF4                      | F0D9H   | Does nothing but return. |
| CTRL/PF5                      | F0DBH   | Screen dump.             |

See 3.5.3.6 for subroutine calls.



ITCDB (0F0DDH) 24 bytes

- Redefinition table for item keyboard keys

| Address | Character on keytop | Default code | Address | Character on keytop | Default code |
|---------|---------------------|--------------|---------|---------------------|--------------|
| F0DDH   | STOP                | 03H          | F0E9H   | =                   | 3DH          |
| F0DEH   | △                   | 1EH          | F0EAH   | 4                   | 34H          |
| F0DFH   | ▽                   | 1FH          | F0EBH   | 5                   | 35H          |
| F0E0H   | ◀                   | 1DH          | F0ECH   | 6                   | 36H          |
| F0E1H   | ▶                   | 1CH          | F0EDH   | ,                   | 2CH          |
| F0E2H   | *                   | 2AH          | F0EEH   | 1                   | 31H          |
| F0E3H   | /                   | 2FH          | F0EFH   | 2                   | 32H          |
| F0E4H   | +                   | 2BH          | F0F0H   | 3                   | 33H          |
| F0E5H   | -                   | 2DH          | F0F1H   | 0                   | 30H          |
| F0E6H   | 7                   | 37H          | F0F2H   | 000                 | FFH          |
| F0E7H   | 8                   | 38H          | F0F3H   | .                   | 2EH          |
| F0E8H   | 9                   | 39H          | F0F4H   | ↵                   | 0DH          |

See 3.5.4.6 for more information.

CHGRIGT (0F328H) 14 bytes

- Table containing key codes of the cursor movement keys, SHIFT/INS, and CTRL/INS.

| Address | Default code | Key      | Address | Default code | Key       |
|---------|--------------|----------|---------|--------------|-----------|
| F328H   | 1CH          | →        | F32FH   | FBH          | SHIFT/ ↓  |
| F329H   | 1DH          | ←        | F330H   | F6H          | SHIFT/INS |
| F32AH   | 1EH          | ↑        | F331H   | FCH          | CTRL/ →   |
| F32BH   | 1FH          | ↓        | F332H   | FDH          | CTRL/ ←   |
| F32CH   | F8H          | SHIFT/ → | F333H   | FEH          | CTRL/ ↑   |
| F32DH   | F9H          | SHIFT/ ← | F334H   | FFH          | CTRL/ ↓   |
| F32EH   | FAH          | SHIFT/ ↑ | F335H   | F7H          | CTRL/INS  |

14-byte data starting at DFLTRT is copied at every warm boot. The key entry routine references this table. See 3.5.3.8 for details.

RWITEMTOP (0CC00H) 1024 bytes  
- Item function table

|     | 1byte                | 15bytes          |
|-----|----------------------|------------------|
| 40H | Number of characters | Character string |
| 41H | Number of characters | Character string |
| 42H | Number of characters | Character string |
|     | ⋮                    | ⋮                |
| 7EH | Number of characters | Character string |
| 7FH |                      |                  |

See 3.5.4.5 and 3.5.4.11.

KEYF (0F008H) 1 byte

- Console buffer (KEYD) status flag
  - = 00H: No input data present
  - = 0FFH: Input data present

This flag is set by CONST and reset by BIOS CONIN.

KEYD (0F009H) 1 byte

- Console input buffer

Data is loaded by BIOS CONST into this area.

KEYS (0F00AH) 1 byte

- Console buffer key status flag
  - = 00H: Nocheck mode data
  - = 0FFH: Check mode data

This flag indicates whether the data in the console buffer is check mode data. It is valid when the current console is the standard keyboard. See the description about BIOS CONIN for check mode.

YINTPTR (0F00BH) 2 bytes

- Key buffer (KBUF) put pointer

This area contains the key buffer address into which the key interrupt handling routine is to load the code received from the 7508.

YKEYPTR (0F00DH) 2 bytes

- Key buffer (KBUF) get pointer

This area contains the key buffer address from which the key entry processing routine is to get input key codes.

YSHFDT (0F00FH) 1 byte

- Standard keyboard switch key status flag
  - Bit 7 -- CTRL (Set to 1 when pressed.)
  - 6 -- GRPH (Set to 1 when the mode is on.)
  - 5 -- Don't care
  - 4 -- NUM (Set to 1 when the mode is on.)
  - 3 -- Don't care
  - 2 -- CAPS (Set to 1 when the mode is on.)
  - 1 -- SHIFT (left) (Set to 1 when pressed.)
  - 0 -- SHIFT (right) (Set to 1 when pressed.)

See 3.5.3.13 for mode transition.

ZSHFT (0F010H) 2 bytes

- Loaded with the starting address of the key code conversion table selected based on the YSHFDT contents.

YPFKSTR (0F012H) 2 bytes

- Loaded with the starting address of the PF key table. See 3.5.3.5.

YPFKPTR (0F014H) 2 bytes

- The get pointer for PF key or item key strings.

YPFKCNT (0F016H) 1 byte

- The counter indicating the number of characters left in the buffer containing the PF key or item key string.

YPFMFLG (0F017H) 1 byte

- PF key check mode flag
  - = 00H: Nocheck mode
  - = 0FFH: Check mode

See the description about BIOS CONIN for check mode.

YSFCMFLG (0F018H) 1 byte

- Special key check mode flag
  - = 00H: Nocheck mode
  - = 0FFH: Check mode

See the description about BIOS CONIN.

BRKFLG (0F019H) 1 byte

- The flag indicating whether the STOP key has been pressed.
  - = 00H: STOP key pressed.
  - = 0FFH: STOP key not pressed.

See 3.5.3.4 for details.

CSTOPFLG (0F01AH) 1 byte

- The flag indicating whether the CTRL/STOP key has been pressed.
  - = 00H: CTRL/STOP key not pressed.
  - = 01H: CTRL/STOP key pressed.

See 3.5.3.4 for details.

CSTOPMCT (0F01BH) 1 byte

- Microcassette termination flag
  - = 00H: CTRL/STOP key not pressed.
  - = 01H: CTRL/STOP key pressed.

See 3.5.3.4 for details.

IMSHFT (0F01CH) 1 byte

- Item keyboard shift mode flag
  - = 00H: Normal mode
  - = 20H: Shift mode

See 3.5.3.5 for shift mode.

ITEMFLG (0F01DH) 1 byte

- Item mode flag (standard keyboard)
  - = 00H: Normal mode
  - = 80H: Item mode

See 3.5.3.11 for item mode.

ISYSREG (0F01EH) 1 byte

- System function flag (item keyboard)
  - Bits 7 - 2: Don't care.
  - Bit 1: CTRL/STOP control bit (disabled when this bit is 0).
  - 0: STOP control bit (disabled when this bit is 0).

See 3.5.3.7 for details.

CSTOPPRN (0F309H) 1 byte

- Cartridge printer termination flag
  - = 00H: CTRL/STOP key not pressed.
  - = 01H: CTRL/STOP key pressed.

See 3.5.3.4 for details.

KEYFLG (0F4D8H) 1 byte

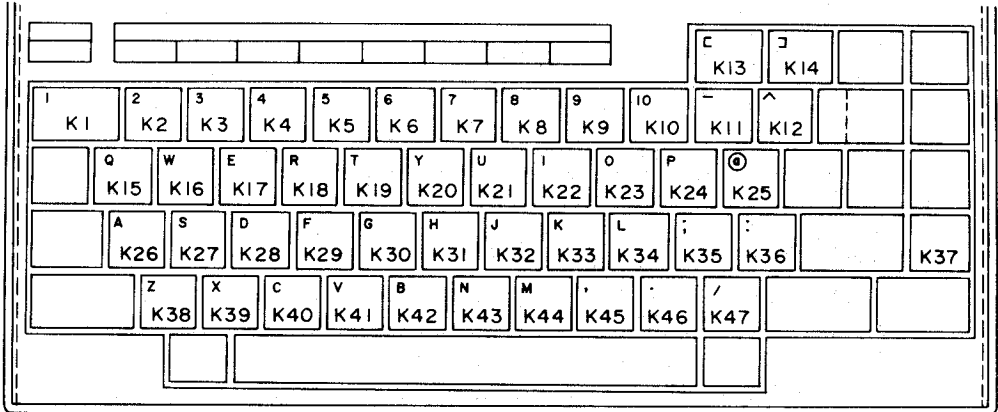
- Switch key status flag used for keyboard interrupt processing
  - Bits 7, 6 -- Don't care.
  - Bit 5 -- NUM
  - 4 -- SHIFT (right)
  - 3 -- GRPH
  - 2 -- CAPS
  - 1 -- SHIFT (left)
  - 0 -- CTRL

### 3.5.6 Key Code Charts

This subsection shows the key codes of the standard and item keyboards.

#### 3.5.6.1 Key code chart (standard keyboard)

The pages that follow contain key input code charts 1 to 10. The correspondence between the key numbers in the charts and the keyboard is shown below.



The relationship among key input code charts, keyboard types, and keyboard modes is as follows:

| key-board type \ Mode     | NORMAL<br>Normal<br>shift | CAPS<br>Caps<br>shift | NUM<br>Num<br>shift | GRAPH<br>Graph<br>shift | CONTROL<br>Control<br>shift |
|---------------------------|---------------------------|-----------------------|---------------------|-------------------------|-----------------------------|
| ASCII                     | Table 1                   | Table 1               | Table 2             | Table 4                 | Table 5                     |
| France                    | Table 1 +<br>Table 10     | Table 1 +<br>Table 10 | Table 3             | Table 4                 | Table 6                     |
| Germany                   | Table 1 +<br>Table 10     | Table 1 +<br>Table 10 | Table 3             | Table 4                 | Table 7                     |
| England                   | Table 1 +<br>Table 10     | Table 1 +<br>Table 10 | Table 2             | Table 4                 | Table 5                     |
| Denmark, Sweden<br>Norway | Table 1 +<br>Table 10     | Table 1 +<br>Table 10 | Table 3             | Table 4                 | Table 8                     |

Key input code Table 1

- (1) Type: ASCII
- (2) Mode: Normal, Normal shift, Caps, Caps shift
- (3) Display mode: Characters on keytops

|   | 0      | 1     | 2  | 3 | 4 | 5 | 6 | 7   | 8 | 9 | A | B | C | D | E | F |
|---|--------|-------|----|---|---|---|---|-----|---|---|---|---|---|---|---|---|
| 0 | HELP   |       | SP | 0 | @ | P | ' | p   |   |   |   |   |   |   |   |   |
| 1 |        |       | !  | 1 | A | Q | a | q   |   |   |   |   |   |   |   |   |
| 2 |        | INS   | '' | 2 | B | R | b | r   |   |   |   |   |   |   |   |   |
| 3 | STOP   | PAUSE | #  | 3 | C | S | c | s   |   |   |   |   |   |   |   |   |
| 4 |        |       | \$ | 4 | D | T | d | t   |   |   |   |   |   |   |   |   |
| 5 |        |       | %  | 5 | E | U | e | u   |   |   |   |   |   |   |   |   |
| 6 |        |       | &  | 6 | F | V | f | v   |   |   |   |   |   |   |   |   |
| 7 |        |       | ^  | 7 | G | W | g | w   |   |   |   |   |   |   |   |   |
| 8 | BS     |       | (  | 8 | H | X | h | x   |   |   |   |   |   |   |   |   |
| 9 | TAB    |       | )  | 9 | I | Y | i | y   |   |   |   |   |   |   |   |   |
| A |        |       | *  | : | J | Z | j | z   |   |   |   |   |   |   |   |   |
| B | HOME   | ESC   | +  | ; | K | [ | k | {   |   |   |   |   |   |   |   |   |
| C | CLR    | →     | ,  | < | L | \ | l |     |   |   |   |   |   |   |   |   |
| D | RETURN | ←     | -  | = | M | ] | m | }   |   |   |   |   |   |   |   |   |
| E |        | ↑     | .  | > | N | ^ | n | ~   |   |   |   |   |   |   |   |   |
| F |        | ↓     | /  | ? | O | - | o | DEL |   |   |   |   |   |   |   |   |

Notes:

- In the Caps shift mode, the relationship between the uppercase and lowercase characters are reversed.
- PF1 to PF10 return the predefined character string.
- SCRN (SHIFT/INS) returns no code.
- STOP, ESC, PAUSE, HELP, TAB, space bar, and return key return the same code whether the keyboard is in normal or shift mode.
- SHIFT/←, →, ↑, and ↓ return no code.

Key input code chart 2

- (1) Type: ASCII, England
- (2) Mode: Numeric, Numeric shift
- (3) Display mode: Characters on keytops

|   | 0      | 1     | 2     | 3    | 4 | 5 | 6 | 7   | 8 | 9 | A | B | C | D | E | F |
|---|--------|-------|-------|------|---|---|---|-----|---|---|---|---|---|---|---|---|
| 0 | HELP   |       | SP    | 0(H) |   |   |   |     |   |   |   |   |   |   |   |   |
| 1 |        |       |       | 1(J) |   |   |   |     |   |   |   |   |   |   |   |   |
| 2 |        |       | INS   | 2(K) |   |   |   |     |   |   |   |   |   |   |   |   |
| 3 | STOP   | PAUSE |       | 3(L) |   |   |   |     |   |   |   |   |   |   |   |   |
| 4 |        |       |       | 4(U) |   |   |   |     |   |   |   |   |   |   |   |   |
| 5 |        |       |       | 5(I) |   |   |   |     |   |   |   |   |   |   |   |   |
| 6 |        |       |       | 6(O) |   |   |   |     |   |   |   |   |   |   |   |   |
| 7 |        |       |       | 7    |   |   |   |     |   |   |   |   |   |   |   |   |
| 8 | BS     |       | ((I)  | 8    |   |   |   |     |   |   |   |   |   |   |   |   |
| 9 | TAB    |       | )(I)  | 9    |   |   |   |     |   |   |   |   |   |   |   |   |
| A |        |       | *(:)  |      |   |   |   |     |   |   |   |   |   |   |   |   |
| B | HOME   | ESC   | +(:)  |      |   |   |   |     |   |   |   |   |   |   |   |   |
| C | CLR    | →     | .     |      |   |   |   |     |   |   |   |   |   |   |   |   |
| D | RETURN | ←     | - (=) |      |   |   |   |     |   |   |   |   |   |   |   |   |
| E |        | ↑     | .     |      |   |   | ^ |     |   |   |   |   |   |   |   |   |
| F |        | ↓     | /     | ?(0) |   |   |   | DEL |   |   |   |   |   |   |   |   |

Notes:

- PF1 to PF10 returns the predefined character string.
- SCRN (SHIFT/INS) returns no code.
- STOP, ESC, PAUSE, HELP, TAB, space bar, and return key return the same code whether the keyboard is in normal or shift mode.
- SHIFT/←, →, ↑, and ↓ return no code.
- The characters in the parentheses are the uppercase characters for the corresponding keys.

Key input code Table 3

- (1) Type: France, German, Denmark, Sweden, Norway
- (2) Mode: Numeric, Numeric shift
- (3) Display mode: Key numbers, characters on keytops

|   | 0    | 1     | 2   | 3                | 4 | 5   | 6 | 7   | 8 | 9 | A | B | C | D | E | F |
|---|------|-------|-----|------------------|---|-----|---|-----|---|---|---|---|---|---|---|---|
| 0 | HELP |       | SP  | K10<br>K44<br>K1 |   |     |   |     |   |   |   |   |   |   |   |   |
| 1 |      |       |     | K32              |   |     |   |     |   |   |   |   |   |   |   |   |
| 2 |      | INS   |     | K2<br>K33        |   |     |   |     |   |   |   |   |   |   |   |   |
| 3 | STOP | PAUSE |     | K3<br>K34<br>K4  |   |     |   |     |   |   |   |   |   |   |   |   |
| 4 |      |       |     | K21              |   |     |   |     |   |   |   |   |   |   |   |   |
| 5 |      |       |     | K5<br>K22        |   |     |   |     |   |   |   |   |   |   |   |   |
| 6 |      |       |     | K6<br>K23        |   |     |   |     |   |   |   |   |   |   |   |   |
| 7 |      |       |     | K 7              |   |     |   |     |   |   |   |   |   |   |   |   |
| 8 | BS   |       | K11 | K 8              |   |     |   |     |   |   |   |   |   |   |   |   |
| 9 | TAB  |       | K12 | K 9              |   |     |   |     |   |   |   |   |   |   |   |   |
| A |      |       | K35 |                  |   |     |   |     |   |   |   |   |   |   |   |   |
| B | HOME | ESC   | K24 |                  |   |     |   |     |   |   |   |   |   |   |   |   |
| C | CLR  | →     | K45 |                  |   |     |   |     |   |   |   |   |   |   |   |   |
| D | RET  | ←     | K25 |                  |   |     |   |     |   |   |   |   |   |   |   |   |
| E |      | ↑     | K46 |                  |   | K14 |   |     |   |   |   |   |   |   |   |   |
| F |      | ↓     | K36 | K47              |   |     |   | DEL |   |   |   |   |   |   |   |   |

Notes:

- PF1 to PF10 return the predefined character string.
- SCRN (SHIFT/INS) returns no code.
- STOP, ESC, PAUSE, HELP, TAB, space bar, and return key return the same code whether the keyboard is in normal or shift mode.
- SHIFT/←, →, ↑, and ↓ return no code.



Key input code Table 4

- (1) Type: All types
- (2) Mode: Graph, Graph shift
- (3) Display mode: Key numbers, characters on keytops

|   | 0      | 1     | 2  | 3 | 4 | 5 | 6 | 7   | 8   | 9   | A | B | C | D | E     | F |
|---|--------|-------|----|---|---|---|---|-----|-----|-----|---|---|---|---|-------|---|
| 0 | HELP   |       | SP |   |   |   |   |     | K27 | K21 |   |   |   |   | K10   |   |
| 1 |        |       |    |   |   |   |   |     | K39 | K22 |   |   |   |   | K1    |   |
| 2 |        | INS   |    |   |   |   |   |     | K16 | K23 |   |   |   |   | K2    |   |
| 3 | STOP   | PAUSE |    |   |   |   |   |     | K28 | K24 |   |   |   |   | K3    |   |
| 4 |        |       |    |   |   |   |   |     | K26 | K25 |   |   |   |   | K4    |   |
| 5 |        |       |    |   |   |   |   |     | K19 | K33 |   |   |   |   | K5    |   |
| 6 |        |       |    |   |   |   |   |     | K18 | K41 |   |   |   |   | K6    |   |
| 7 |        |       |    |   |   |   |   |     | K15 | K45 |   |   |   |   | K7    |   |
| 8 | BS     |       |    |   |   |   |   |     | K17 | K44 |   |   |   |   | K8    |   |
| 9 | TAB    |       |    |   |   |   |   |     | K38 | K43 |   |   |   |   | K9    |   |
| A |        |       |    |   |   |   |   |     | K40 | K42 |   |   |   |   | K11   |   |
| B | HOME   | ESC   |    |   |   |   |   |     | K32 | K35 |   |   |   |   | K12   |   |
| C | CLR    | →     |    |   |   |   |   |     | K29 | K46 |   |   |   |   | K13   |   |
| D | RETURN | ←     |    |   |   |   |   |     | K30 | K36 |   |   |   |   | K37   |   |
| E |        | ↑     |    |   |   |   |   |     | K31 | K47 |   |   |   |   | K14   |   |
| F |        | ↓     |    |   |   |   |   | DEL | K20 | K34 |   |   |   |   | S/K14 |   |

Notes:

- PF1 to PF10 return the predefined character string.
- SCRN (SHIFT/INS) returns no code.
- STOP, ESC, PAUSE, HELP, TAB, space bar, and return key return the same code whether the keyboard is in normal or shift mode.
- SHIFT/←, →, ↑, and ↓ return no code.

Key input code Table 5

- (1) Type: ASCII, England
- (2) Mode: Control Control shift
- (3) Display mode: Characters on keytops

|   | 0                    | 1 | 2  | 3 | 4 | 5 | 6 | 7 | 8   | 9 | A | B | C | D | E | F |
|---|----------------------|---|----|---|---|---|---|---|-----|---|---|---|---|---|---|---|
| 0 | @                    | P | SP |   |   |   |   |   |     |   |   |   |   |   |   | 0 |
| 1 | A                    | Q |    |   |   |   |   |   |     |   |   |   |   |   |   | 1 |
| 2 | B                    | R |    |   |   |   |   |   |     |   |   |   |   |   |   | 2 |
| 3 | C                    | S |    |   |   |   |   |   |     |   |   |   |   |   |   | 3 |
| 4 | D                    | T |    |   |   |   |   |   |     |   |   |   |   |   |   | 4 |
| 5 | E                    | U |    |   |   |   |   |   |     |   |   |   |   |   |   | 5 |
| 6 | F                    | V |    |   |   |   |   |   |     |   |   |   |   |   |   | 6 |
| 7 | G                    | W |    |   |   |   |   |   |     |   |   |   |   |   |   | 7 |
| 8 | <sup>H</sup><br>BS   | X |    |   |   |   |   |   |     |   |   |   |   |   |   | 8 |
| 9 | <sup>I</sup><br>TAB  | Y |    |   |   |   |   |   |     |   |   |   |   |   |   | 9 |
| A | J                    | Z |    |   |   |   |   |   |     |   |   |   |   |   |   | - |
| B | <sup>K</sup><br>HOME | [ |    |   |   |   |   |   |     |   |   |   |   |   |   | ; |
| C | <sup>L</sup><br>CLR  | \ |    |   |   |   |   |   |     |   |   |   |   |   |   | : |
| D | <sup>M</sup><br>RET  | ] |    |   |   |   |   |   |     |   |   |   |   |   |   | , |
| E | N                    | ^ |    |   |   |   |   |   |     |   |   |   |   |   |   | . |
| F | O                    | / |    |   |   |   |   |   | DEL |   |   |   |   |   |   |   |

Notes:

- STOP, ESC, PAUSE, HELP, and PF1 to PF5 call a subroutine.
- TAB, return key, and space bar return the same code whether the keyboard is in normal or shift mode.
- INS, ←, →, ↑, and ↓ return no code.

Key input code Table 6

- (1) Type: France
- (2) Mode: Control Control shift
- (3) Display mode: Characters on keytops

|   | 0         | 1 | 2  | 3 | 4 | 5 | 6 | 7   | 8 | 9 | A | B | C | D | E | F |
|---|-----------|---|----|---|---|---|---|-----|---|---|---|---|---|---|---|---|
| 0 | à         | P | SP |   |   |   |   |     |   |   |   |   |   |   |   | ù |
| 1 | A         | Q |    |   |   |   |   |     |   |   |   |   |   |   |   | < |
| 2 | B         | R |    |   |   |   |   |     |   |   |   |   |   |   |   | # |
| 3 | C         | S |    |   |   |   |   |     |   |   |   |   |   |   |   | & |
| 4 | D         | T |    |   |   |   |   |     |   |   |   |   |   |   |   | é |
| 5 | E         | U |    |   |   |   |   |     |   |   |   |   |   |   |   | “ |
| 6 | F         | V |    |   |   |   |   |     |   |   |   |   |   |   |   | ’ |
| 7 | G         | W |    |   |   |   |   |     |   |   |   |   |   |   |   | ( |
| 8 | H<br>BS   | X |    |   |   |   |   |     |   |   |   |   |   |   |   | è |
| 9 | I<br>TAB  | Y |    |   |   |   |   |     |   |   |   |   |   |   |   | ! |
| A | J         | Z |    |   |   |   |   |     |   |   |   |   |   |   |   | ’ |
| B | K<br>HOME | ) |    |   |   |   |   |     |   |   |   |   |   |   |   | . |
| C | L<br>CLR  | Ç |    |   |   |   |   |     |   |   |   |   |   |   |   | ; |
| D | M<br>RET  | § |    |   |   |   |   |     |   |   |   |   |   |   |   | : |
| E | N         | ^ |    |   |   |   |   |     |   |   |   |   |   |   |   | = |
| F | O         | - |    |   |   |   |   | DEL |   |   |   |   |   |   |   |   |

Notes:

- STOP, ESC, PAUSE, HELP, and PF1 to PF5 call a subroutine.
- TAB, return key, and space bar return the same code whether the keyboard is in normal or shift mode.
- INS, ←, →, ↑, and ↓ return no code.

Key input code Table 7

- (1) Type: Germany
- (2) Mode: Control Control shift
- (3) Display mode: Characters on keytops

|   | 0         | 1 | 2  | 3 | 4 | 5 | 6 | 7 | 8   | 9  | A | B | C | D | E | F |
|---|-----------|---|----|---|---|---|---|---|-----|----|---|---|---|---|---|---|
| 0 | ß         | P | SP |   |   |   |   |   |     |    |   |   |   |   |   | 0 |
| 1 | A         | Q |    |   |   |   |   |   |     |    |   |   |   |   |   | 1 |
| 2 | B         | R |    |   |   |   |   |   |     |    |   |   |   |   |   | 2 |
| 3 | C         | S |    |   |   |   |   |   |     |    |   |   |   |   |   | 3 |
| 4 | D         | T |    |   |   |   |   |   |     |    |   |   |   |   |   | 4 |
| 5 | E         | U |    |   |   |   |   |   |     |    |   |   |   |   |   | 5 |
| 6 | F         | V |    |   |   |   |   |   |     |    |   |   |   |   |   | 6 |
| 7 | G         | W |    |   |   |   |   |   |     |    |   |   |   |   |   | 7 |
| 8 | H<br>BS   | X |    |   |   |   |   |   |     |    |   |   |   |   |   | 8 |
| 9 | I<br>TAB  | Y |    |   |   |   |   |   |     | \\ |   |   |   |   |   | 9 |
| A | J         | Z |    |   |   |   |   |   |     |    |   |   |   |   |   | < |
| B | K<br>HOME | Ä |    |   |   |   |   |   |     |    |   |   |   |   |   | # |
| C | L<br>CLR  | Ö |    |   |   |   |   |   |     |    |   |   |   |   |   | + |
| D | M<br>RET  | Ü |    |   |   |   |   |   |     |    |   |   |   |   |   | . |
| E | N         | ^ |    |   |   |   |   |   |     |    |   |   |   |   |   | . |
| F | O         | - |    |   |   |   |   |   | DEL |    |   |   |   |   |   |   |

Notes:

- STOP, ESC, PAUSE, HELP, and PF1 to PF5 call a subroutine.
- TAB, return key, and space bar return the same code whether the keyboard is in normal or shift mode.
- INS, ←, →, ↑, and ↓ return no code.

Key input code Table 8

- (1) Type: Sweden, Denmark, Norway
- (2) Mode: Control Control shift
- (3) Display mode: Characters on keytops

|   |           |      |    |  |  |  |  |  |     |  |  |  |  |  |  |  |   |
|---|-----------|------|----|--|--|--|--|--|-----|--|--|--|--|--|--|--|---|
| 0 | É         | P    | SP |  |  |  |  |  |     |  |  |  |  |  |  |  | 0 |
| 1 | A         | Q    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 1 |
| 2 | B         | R    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 2 |
| 3 | C         | S    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 3 |
| 4 | D         | T    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 4 |
| 5 | E         | U    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 5 |
| 6 | F         | V    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 6 |
| 7 | G         | W    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 7 |
| 8 | H<br>BS   | X    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 8 |
| 9 | I<br>TAB  | Y    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | 9 |
| A | J         | Z    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | < |
| B | K<br>HOME | (*1) |    |  |  |  |  |  |     |  |  |  |  |  |  |  | + |
| C | L<br>CLR  | (*1) |    |  |  |  |  |  |     |  |  |  |  |  |  |  | , |
| D | M<br>RET  | Å    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | . |
| E | N         | Ü    |    |  |  |  |  |  |     |  |  |  |  |  |  |  | . |
| F | O         | -    |    |  |  |  |  |  | DEL |  |  |  |  |  |  |  |   |

Notes:

- STOP, ESC, PAUSE, HELP, and PF1 to PF5 call a subroutine.
- TAB, return key, and space bar return the same code whether the keyboard is in normal or shift mode.
- INS, ←, →, ↑, and ↓ return no code.

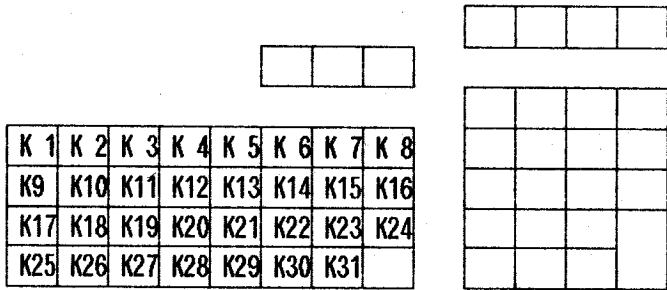
|      |         |        |        |
|------|---------|--------|--------|
| (*1) | Denmark | Sweden | Norway |
| LBH  | ? Æ     | Ä      | Æ      |
| LCH  | Ø       | Ö      | Ø      |

Table 9 Correspondence between the key codes and characters assorted by country

|     | U.S.A. | France | Germany | England | Denmark | Sweden | Italy | Spain | Norway |
|-----|--------|--------|---------|---------|---------|--------|-------|-------|--------|
| 23H | #      | #      | #       | £       | #       | #      | #     | Pt    | #      |
| 24H | \$     | \$     | \$      | \$      | \$      | ⌘      | \$    | \$    | ⌘      |
| 40H | @      | à      | §       | @       | @       | É      | @     | @     | É      |
| 5BH | [      | °      | Ä       | [       | Æ       | Ä      | °     | i     | Æ      |
| 5CH | \      | Ç      | Ö       | \       | Ø       | Ö      | \     | Ñ     | Ø      |
| 5DH | ]      | §      | Ü       | ]       | Å       | Å      | é     | ¿     | Å      |
| 5EH | ^      | ^      | ^       | ^       | ^       | Ü      | ^     | ^     | Ü      |
| 60H | ˆ      | ˆ      | ˆ       | ˆ       | ˆ       | é      | ù     | ˆ     | é      |
| 7BH | {      | é      | ä       | {       | æ       | ä      | à     | ˆ     | æ      |
| 7CH |        | ù      | ö       |         | ø       | ö      | ò     | ñ     | ø      |
| 7DH | }      | è      | ü       | }       | å       | å      | è     | }     | å      |
| 7EH | ~      | ˆ      | ß       | ~       | ~       | ü      | i     | ˆ     | ü      |

### 3.5.6.2 Key code chart (item keyboard)

The following pages contain key input code charts. The correspondence between the key numbers in the charts and the keyboard is shown below.



Key input code table 1

- (1) Mode: Normal
- (2) Display mode: Key numbers and characters on keytops

|   | 0    | 1 | 2   | 3                            | 4   | 5   | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|------|---|-----|------------------------------|-----|-----|---|---|---|---|---|---|---|---|---|---|
| 0 |      |   | K31 | <sup>0</sup> <sub>(*1)</sub> | K1  | K17 |   |   |   |   |   |   |   |   |   |   |
| 1 |      |   |     | 1                            | K2  | K18 |   |   |   |   |   |   |   |   |   |   |
| 2 |      |   |     | 2                            | K3  | K19 |   |   |   |   |   |   |   |   |   |   |
| 3 | STOP |   |     | 3                            | K4  | K20 |   |   |   |   |   |   |   |   |   |   |
| 4 |      |   |     | 4                            | K5  | K21 |   |   |   |   |   |   |   |   |   |   |
| 5 |      |   |     | 5                            | K6  | K22 |   |   |   |   |   |   |   |   |   |   |
| 6 |      |   |     | 6                            | K7  | K23 |   |   |   |   |   |   |   |   |   |   |
| 7 |      |   |     | 7                            | K8  | K24 |   |   |   |   |   |   |   |   |   |   |
| 8 |      |   |     | 8                            | K9  | K25 |   |   |   |   |   |   |   |   |   |   |
| 9 |      |   |     | 9                            | K10 | K26 |   |   |   |   |   |   |   |   |   |   |
| A |      |   | *   |                              | K11 | K27 |   |   |   |   |   |   |   |   |   |   |
| B |      |   | +   |                              | K12 | K28 |   |   |   |   |   |   |   |   |   |   |
| C |      | ▶ | .   |                              | K13 | K29 |   |   |   |   |   |   |   |   |   |   |
| D |      | ◀ | -   | =                            | K14 | K30 |   |   |   |   |   |   |   |   |   |   |
| E |      | △ | .   |                              | K15 |     |   |   |   |   |   |   |   |   |   |   |
| F |      | ▽ | /   |                              | K16 |     |   |   |   |   |   |   |   |   |   |   |

Note:  
- CTRL and INIT return no code.

(\*1) Three 30H codes are returned when the 000 key is pressed.

Key input code table 2

(1) Mode: Shift

(2) Display mode: Key numbers and characters on keytops

|   | 0    | 1  | 2                        | 3   | 4 | 5   | 6   | 7   | 8 | 9 | A | B | C | D | E | F |
|---|------|----|--------------------------|-----|---|-----|-----|-----|---|---|---|---|---|---|---|---|
| 0 | K3   |    | K31 <sup>0</sup><br>(+1) |     |   |     | K20 |     |   |   |   |   |   |   |   |   |
| 1 |      |    | K9                       | 1   |   |     |     |     |   |   |   |   |   |   |   |   |
| 2 |      | K4 | K10                      | 2   |   |     |     |     |   |   |   |   |   |   |   |   |
| 3 | STOP | K2 | K11                      | 3   |   |     |     |     |   |   |   |   |   |   |   |   |
| 4 |      |    | K12                      | 4   |   |     |     |     |   |   |   |   |   |   |   |   |
| 5 |      |    | K13                      | 5   |   |     |     |     |   |   |   |   |   |   |   |   |
| 6 |      |    | K14                      | 6   |   |     |     |     |   |   |   |   |   |   |   |   |
| 7 |      |    | K15                      | 7   |   |     |     |     |   |   |   |   |   |   |   |   |
| 8 | K8   |    | K16                      | 8   |   |     |     |     |   |   |   |   |   |   |   |   |
| 9 |      |    | K17                      | 9   |   |     |     |     |   |   |   |   |   |   |   |   |
| A |      |    | *                        | K25 |   |     |     |     |   |   |   |   |   |   |   |   |
| B | K5   | K1 | +                        | K26 |   |     |     | K21 |   |   |   |   |   |   |   |   |
| C | K6   | ▶  | ,                        | K27 |   |     |     | K22 |   |   |   |   |   |   |   |   |
| D | ↶    | ◀  | -                        | K28 |   |     |     | K23 | ✓ |   |   |   |   |   |   |   |
| E |      | △  | •                        | K29 |   | K18 |     | K24 |   |   |   |   |   |   |   |   |
| F |      | ▽  | /                        | K30 |   | K19 |     | K7  |   |   |   |   |   |   |   |   |

Notes:

- CTRL and INIT return no code.

(\*1) Three 30H codes are returned when the 000 key is pressed.



### 3.5.7 Hard Code Charts

This subsection describes the hard codes of the standard and item keyboards.

Hard codes refers to those codes which are used by the 7508 slave CPU to generate keyboard interrupts and send input key information to the main CPU when keys are pressed or shift keys are released.

The main program routine loads each hard code into the key buffer (KBUF). If the returned hard code is the make code for the STOP key (0B6H) and if the item key board is installed, the program converts the code to 10H, i.e., the STOP key code for the standard keyboard, and places the converted code into the key buffer.

#### (1) Hard code chart (standard keyboard)

The keys on the standard keyboard are assigned hard codes as shown below. Break and Make in the table indicate the code is generated at the time the corresponding key is released and pressed, respectively.

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

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

(2) Hard code chart (item keyboard)

The keys on the item keyboard are assigned the hard codes as shown below. Break and Make in the table indicate that the code is returned at the time the corresponding key is pressed and released, respectively.

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
|---|---|---|

|    |    |    |    |
|----|----|----|----|
| 36 | 37 | 38 | 39 |
|----|----|----|----|

|    |    |    |    |
|----|----|----|----|
| 40 | 41 | 42 | 43 |
| 44 | 45 | 46 | 47 |
| 48 | 49 | 50 | 51 |
| 52 | 53 | 54 | 58 |
| 55 | 56 | 57 |    |

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 4  | 5  | 6  | 7  | 8  | 9  | 10 | 11 |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 |

| High<br>Low order<br>order | 0 | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8 | 9 | A           | B          |
|----------------------------|---|----|----|----|----|----|----|----|---|---|-------------|------------|
| 0                          |   |    | 43 | 51 | 4  | 20 | 12 | 28 |   |   |             |            |
| 1                          |   | 36 | 44 | 52 | 5  | 21 | 13 | 29 |   |   |             |            |
| 2                          |   | 37 | 45 | 53 | 6  | 22 | 14 | 30 |   |   | 1<br>Break  | 1<br>Make  |
| 3                          |   | 38 | 46 | 54 | 7  | 23 | 15 | 31 |   |   |             |            |
| 4                          |   | 39 | 47 | 55 | 8  | 24 | 16 | 32 |   |   |             |            |
| 5                          |   | 40 | 48 | 56 | 9  | 25 | 17 | 33 |   |   | 2<br>Break  | 2<br>Make  |
| 6                          |   | 41 | 49 | 57 | 10 | 26 | 18 | 34 |   |   | 3<br>Break  | 3<br>Make  |
| 7                          |   | 42 | 50 | 58 | 11 | 27 | 19 |    |   |   | 35<br>Break | 35<br>Make |

\*\*\*\*\*  
 DISPLAY KEY STATUS  
 \*\*\*\*\*

NOTE : This sample program is displaying the current key status.

<> assemble condition <>

.Z80

<> loading address <>

.PHASE 100H

<> constant values <>

BIOS entry

|      |        |     |            |   |                      |
|------|--------|-----|------------|---|----------------------|
| EB03 | WBOOT  | EQU | 0EB03H     | : | Warm Boot entry      |
| EB06 | CONST  | EQU | WBOOT +03H | : | Console status entry |
| EB09 | CONIN  | EQU | WBOOT +06H | : | Console input entry  |
| EB0C | CONOUT | EQU | WBOOT +09H | : | Console out entry    |

System area

|      |           |     |        |   |                            |
|------|-----------|-----|--------|---|----------------------------|
| F00F | YSHFDT    | EQU | 0F00FH | : | Normal keyboard key status |
| F01F | IMSHFT    | EQU | 0F01FH | : | ITEM keyboard key status   |
| F775 | YKCOUNTRY | EQU | 0F775H | : | Keyboard country           |

|      |      |     |     |   |                   |
|------|------|-----|-----|---|-------------------|
| 0003 | STOP | EQU | 03H | : | Stop code         |
| 0005 | EOL  | EQU | 05H | : | Erase end of line |
| 000D | CR   | EQU | 0DH | : | Carriage return   |
| 000A | LF   | EQU | 0AH | : | Line feed         |
| 000B | HOME | EQU | 0BH | : | Home code         |
| 0012 | CLS  | EQU | 12H | : | Clear screen      |
| 001B | ESC  | EQU | 1BH | : | ESC code          |

\*\*\*\*\*  
 MAIN PROGRAM  
 \*\*\*\*\*

NOTE : This program is displaying the current shift key status.

|      |         |         |           |   |                     |
|------|---------|---------|-----------|---|---------------------|
| 0100 |         | LD      | SP,1000H  | : | Set stack pointer.  |
| 0100 | 31 1000 |         |           |   |                     |
| 0103 | 0E 12   | LD      | C,CLS     | : | Clear screen & home |
| 0105 | CD EBOC | CALL    | CONOUT    | : |                     |
| 0108 | CD 0186 | CALL    | CUSROFF   | : | Cursor off.         |
| 010B |         | MAIN10: |           |   |                     |
| 010B | 76      | HALT    |           | : | Halt (sleep mode)   |
| 010C | CD 0122 | CALL    | KEYST     | : | Display key status. |
| 010F | CD EB06 | CALL    | CONST     | : | Input any key?      |
| 0112 | 3C      | INC     | A         | : |                     |
| 0113 | 20 F6   | JR      | NZ,MAIN10 | : | No.                 |
| 0115 | CD EB09 | CALL    | CONIN     | : | Get inputted key.   |
| 0118 | FE 03   | CP      | STOP      | : | STOP?               |
| 011A | 20 EF   | JR      | NZ,MAIN10 | : | No.                 |
| 011C | CD 017B | CALL    | CUSRON    | : | Cursor on.          |
| 011F | C3 EB03 | JP      | WBOOT     | : | End.                |

\*\*\*\*\*  
 DISPLAY KEY STATUS  
 \*\*\*\*\*

NOTE : Display current key status.

<> entry parameter <>

NON

<> return parameter <>

NON

<> preserved registers <>

NON

CAUTION :

|      |         |        |               |   |                                      |
|------|---------|--------|---------------|---|--------------------------------------|
| 0122 |         | KEYST: |               |   |                                      |
| 0122 | 21 01DE | LD     | HL,SVSHFT     | : | Old key status.                      |
| 0125 | 11 F00F | LD     | DE,YSHFDT     | : | Current key status (Normal keyboard) |
| 0128 | 3A F775 | LD     | A,(YKCOUNTRY) | : | ITEM keyboard?                       |
| 012B | E6 80   | AND    | 80H           | : |                                      |
| 012D | 28 03   | JR     | Z,KEY10       | : | No.                                  |
| 012F | 11 F01F | LD     | DE,IMSHFT     | : | Current key status (ITEM)            |
| 0132 |         | KEY10: |               |   |                                      |
| 0132 | 1A      | LD     | A,(DE)        | : | Get current status.                  |
| 0133 | BE      | CP     | (HL)          | : | Same as old one?                     |
| 0134 | C8      | RET    | Z             | : | Yes.                                 |
| 0135 | 77      | LD     | (HL),A        | : | Save current key status.             |
| 0136 | 0E 0B   | LD     | C,HOME        | : | Move cursor to home position.        |
| 0138 | CD 016F | CALL   | SVCONOUT      | : |                                      |

```

013B 21 0191      LD      HL,SHFTTBL      ; Shift data table.
013E 06 08        LD      B,08H          ; Loop counter.

0140
0140 0E 05      KEY20: LD      C,EOL          ; Erase end of line.
0142 CD 016F     CALL   SVCONOUT

0145 1F          RRA          ; LSB --> CY
0146 DC 0159     CALL   C,DSPMSG      ; If ON, then dsp message.
0149 23          INC      HL          ; Pointer update.
014A 23          INC      HL
014B 05          DEC      B          ; Counter decrement.
014C C8          RET      Z          ; End.

014D 0E 0D        LD      C,CR          ; Move cursor to next line.
014F CD 016F     CALL   SVCONOUT
0152 0E 0A        LD      C,LF
0154 CD 016F     CALL   SVCONOUT
0157 18 E7        JR      KEY20        ; Loop.

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

NOTE :      Display message

<> entry parameter <>
          HL : Data table top address.
          (HL*0) -- Byte number
          (HL*1) -- Display data

<> return parameter <>
          NON

<> preserved registers <>
          All registers

CAUTION :

0159
0159 F5      DSPMSG: PUSH   AF          ; Save registers
015A C5      PUSH   BC
015B D5      PUSH   DE
015C E5      PUSH   HL

015D 5E      LD      E,(HL)      ; Get data top address
015E 23      INC      HL
015F 56      LD      D,(HL)
0160 EB      EX      DE,HL

0161 46      LD      B,(HL)      ; Get displaying data number.
0162 23      INC      HL        ; HL %s displaying data top address.

0163
0163 4E      DSP10: LD      C,(HL)      ; Get display data.
0164 CD 016F  CALL   SVCONOUT      ; Display.
0167 23      INC      HL        ; Pointer update.
0168 10 F9   DJNZ   DSP10      ; Loop.

016A E1      POP    HL          ; Restore registers.
016B D1      POP    DE
016C C1      POP    BC
016D F1      POP    AF
016E C9      RET

          Conout data with unchanging all registers

016F
016F F5      SVCONOUT: PUSH   AF
0170 C5      PUSH   BC
0171 D5      PUSH   DE
0172 E5      PUSH   HL
0173 CD EB0C CALL   CONOUT
0176 E1      POP    HL
0177 D1      POP    DE
0178 C1      POP    BC
0179 F1      POP    AF
017A C9      RET

          Cursor on

017B
017B 0E 1B   CUSRON: LD      C,ESC
017D CD EB0C CALL   CONOUT
0180 0E 33   LD      C,'3'
0182 CD EB0C CALL   CONOUT
0185 C9      RET

          Cursor off

0186
0186 0E 1B   CUSROFF: LD      C,ESC
0188 CD EB0C CALL   CONOUT
018B 0E 32   LD      C,'2'
018D CD EB0C CALL   CONOUT
0190 C9      RET

0191
0191 01A1    SHFTTBL: DW     SHFTR
0193 01AE    DW     SHFTL
0195 01BA    DW     CAPS
0197 01C4    DW     NON
0199 01C6    DW     NUM
019B 01C4    DW     NON

```

```

019D 01CE
019F 01D6

01A1
01A1 0C
01A2 53 48 49 46
01A6 54 28 72 69
01AA 67 68 74 29
01AE
01AE 0B
01AF 53 48 49 46
01B3 54 28 6C 65
01B7 66 74 29
01BA
01BA 09
01BB 43 41 50 53
01BF 20 4C 4F 43
01C3 4B
01C4
01C4 01
01C5 00
01C6
01C6 07
01C7 4E 55 4D 45
01CB 52 49 43
01CE
01CE 07
01CF 47 52 41 50
01D3 48 49 43
01D6
01D6 07
01D7 43 4F 4E 54
01DB 52 4F 4C

```

```

01DE
01DE FF

```

```

DW GRPH
DW CTRL

SHFTR: DB 12
        DB 'SHIFT(right)'

SHFTL: DB 11
        DB 'SHIFT(left)'

CAPS:  DB 9
        DB 'CAPS LOCK'

NON:   DB 1
        DB 00H

NUM:   DB 7
        DB 'NUMERIC'

GRPH:  DB 7
        DB 'GRAPHIC'

CTRL:  DB 7
        DB 'CONTROL'

SVSHFT: DB OFFH

END

```

\*\*\*\*\*  
 KEY IN BY USING HALT  
 \*\*\*\*\*

NOTE : This sample program is how to input key  
 by using halt.

<> assemble condition <>

.Z80

<> loading address <>

.PHASE 100H

<> constant values <>

BIOS entry

|      |          |     |            |                        |
|------|----------|-----|------------|------------------------|
| EB03 | WBOOT    | EQU | 0EB03H     | : Warm Boot entry      |
| EB06 | CONST    | EQU | WBOOT +03H | : Console status entry |
| EB09 | CONIN    | EQU | WBOOT +06H | : Console input entry  |
| EB0C | CONOUT   | EQU | WBOOT +09H | : Console out entry    |
| EB7E | POWEROFF | EQU | WBOOT +7BH | : Power off entry      |

System area

|      |           |     |        |                                    |
|------|-----------|-----|--------|------------------------------------|
| EF40 | ATSHUTOFF | EQU | 0EF40H | : Auto power off time (minute)     |
| EF41 | ATSOTIME  | EQU | 0EF41H | : Auto power off time (second)     |
| EF8F | TIMER0    | EQU | 0EF8FH | : 1 sec counter                    |
| F77C | TIMEEND   | EQU | 0F77CH | : Auto power off time setting area |

|      |      |     |     |             |
|------|------|-----|-----|-------------|
| 0003 | STOP | EQU | 03H | : Stop code |
|------|------|-----|-----|-------------|

\*\*\*\*\*  
 MAIN PROGRAM  
 \*\*\*\*\*

NOTE : This program is using CONST routine.  
 And at the time of auto power off,  
 power off by user.

|      |         |      |               |                                      |
|------|---------|------|---------------|--------------------------------------|
| 0100 | MAIN:   | LD   | SP,1000H      | : Set stack pointer.                 |
| 0100 |         |      |               |                                      |
| 0103 | MAIN10: | LD   | HL,(ATSOTIME) | : Set auto power off time.           |
| 0103 |         | LD   | DE,(TIMER0)   | : (ATSOTIME)+(TIMER0) --> (TIMEEND)  |
| 0106 |         | ADD  | HL,DE         |                                      |
| 010A |         | LD   | (TIMEEND),HL  |                                      |
| 010B |         |      |               |                                      |
| 010E | MAIN20: | LD   | A,(ATSHUTOFF) | : Check auto power off.              |
| 010E |         | OR   | A             | : Disable?                           |
| 0111 |         | JR   | Z,MAIN30      | : Yes.                               |
| 0112 |         |      |               |                                      |
| 0114 |         | LD   | HL,(TIMEEND)  | : Check power off time.              |
| 0117 |         | LD   | DE,(TIMER0)   | : (TIMEEND)-(TIMER0) < 0 ?           |
| 011B |         | OR   | A             |                                      |
| 011C |         | SBC  | HL,DE         |                                      |
| 011E |         | LD   | C,00H         | : Yes, then continue mode power off. |
| 0120 |         | CALL | M,POWEROFF    |                                      |
| 0123 | MAIN30: | HALT |               | : Halt (Sleep mode)                  |
| 0123 |         |      |               |                                      |
| 0124 |         | CALL | CONST         | : Input any key?                     |
| 0127 |         | JNC  | A             |                                      |
| 0128 |         | JR   | NZ,MAIN20     | : No.                                |
| 012A |         | CALL | CONIN         | : Get inputted key.                  |
| 012D |         | CP   | STOP          | : Stop code?                         |
| 012F |         | JP   | Z,WBOOT       | : Yes, then end.                     |
| 0132 |         | LD   | C,A           | : Display inputted key.              |
| 0133 |         | CALL | CONOUT        |                                      |
| 0136 |         | JR   | MAIN10        | : Loop.                              |

END

### 3.6 LCD Display (CONOUT Details)

#### 3.6.1 General

##### 3.6.1.1 Hardware general description

The PINE LCD displays 240 x 64 dots of information in a uniform matrix using 1/64-duty dynamic drives. The basic specifications for the PINE LCD controller are given below.

- (1) LCD panel ----- 240 x 64 dots (40 characters x 8 lines)
- (2) LCD driver ----- X driver: SED 1120 x 4  
Y driver: SED 1130 x 1
- (3) VRAM area ----- Reserved in main RAM (in 2K-byte unit)  
Valid display area: 30 x 64 = 1920 bytes
- (4) Display mode --- Bit image (character generator not provided)
- (5) Scroll ----- Vertical dot scroll
- (6) Cursor display - Not provided.
- (7) Frame frequency- Variable.
- (8) Display on/off  
function -- Provided.
- (9) Low power consumption achieved by use of speed-up signals.

VRAM is located in the main RAM on bank 0. The LCD controller reads and displays the VRAM contents in bit image.

##### 3.6.1.2 Software general description

- (1) The PINE OS supports two types of screens: the system screen and the user screen. These two screens can be controlled independently.
- (2) The PINE OS uses the concept of virtual screen and provides a screen of 80 characters by 25 lines.
- (3) The cursor is displayed under software control. Cursor blinking is controlled by overflow interrupts.
- (4) The character generator is located in OS ROM and displays 6 x 8 dot characters.
- (5) The display functions are implemented in the form of ESC sequences.

### 3.6.2 Screen Modes

#### 3.6.2.1 Screen configuration

The PINE uses two types of screen: the system screen (40 characters x 8 lines) and the user screen (a maximum of 80 characters x 25 lines). These screens are independent of each other. That is, manipulating one screen does not affect the other screen at all.

The user can use only the user screen. The system screen is used by the OS functions such as system display, alarm, and power fail.

Figure 3.6.1 illustrates the relationship between the system screen and the user screen.

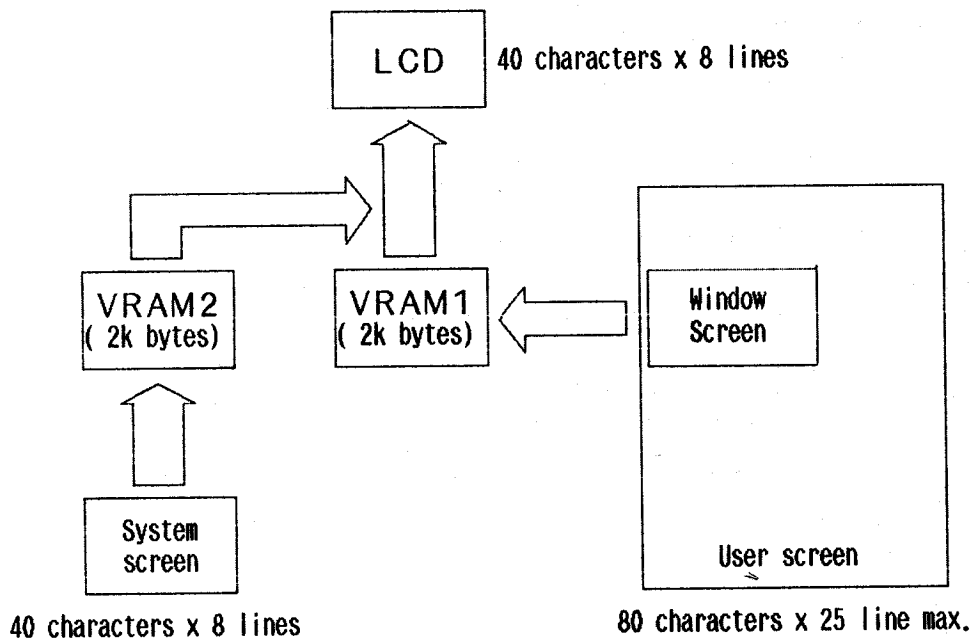


Fig. 3.6.1 System Screen and User Screen