# Chapter 5 Keyboard

## 5.1 General

The MAPLE is furnished with a typewriter keyboard which contains special keys such as cursor movement keys (arrow keys) and programmable function keys. I/O operations concerning the keyboard is controlled by the 7508 sub-CPU. When a key entry is made, the 7508 informs the Z80 CPU of the presence of the key entry by generating an interrupt. The OS, on receipt of the interrupt, fetches information from the 7508 identifying the key location and takes the corresponding action. In addition to this key entry function, a number of MAPLE keyboard functions are supported at the OS level. Those keyboard functions are fully discussed in this chapter (see Chapter 11 for the 7508 CPU).

## 5.2 Keys and Keyboard Types

- Number of keys: 72 (73 keys for Japanese-language keyboard)
- Number of switch keys: 6

    * What is a switch key?

    When an ordinary key is pressed, the 7508 CPU provides only the information that indicates the

depression of the key. When a switch key is pressed, however, it provides two types of information, that is the information indicating the depression of a key and the information indicating the release of the key. This kind of keys include SHIFT and CTRL are used to switch the keyboard mode. These keys are all controlled by the OS and application programs need not concern about this.

- Keyboard types

  The MAPLE supports twelve types of keyboards to accommodate various languages. Keyboard and OS key entry routine assignments are defined by DIP-SW 1 through 4 in the MAPLE's ROM compartment. DIP-SW settings are shown on the next page (see the end of this mannual for key assignments for different countries).

| Keyboard type | DIP-SW | | | | Object OS |
|---|---|---|---|---|---|
| | 4 | 3 | 2 | 1 | |
| USASCII | 1 | 1 | 1 | 1 | ASCII OS |
| France | 1 | 1 | 1 | Ø | |
| Germany | 1 | 1 | Ø | 1 | |
| England | 1 | 1 | Ø | Ø | |
| Denmark | 1 | Ø | 1 | 1 | |
| Sweden | 1 | Ø | 1 | Ø | |
| Italy | 1 | Ø | Ø | 1 | |
| Spain | 1 | Ø | Ø | Ø | |
| Norway | Ø | 1 | 1 | Ø | |
| Kana | Ø | Ø | Ø | Ø | Japanese-language JIS OS |
| Japanese-language JIS keyboard | Ø | Ø | Ø | 1 | |
| Japanese-language touch type keyboard | Ø | Ø | Ø | 1 | Japanese-language touch type OS |

Ø --- OFF
1 --- ON

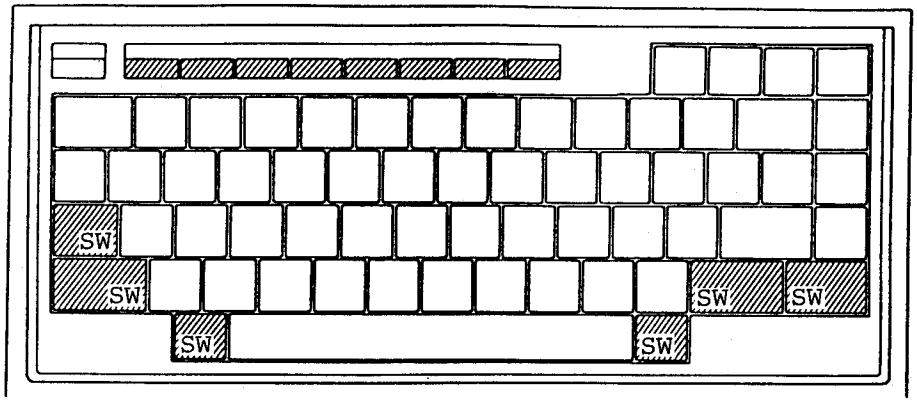
Auto repeat keys and switches

Auto repeat keys 

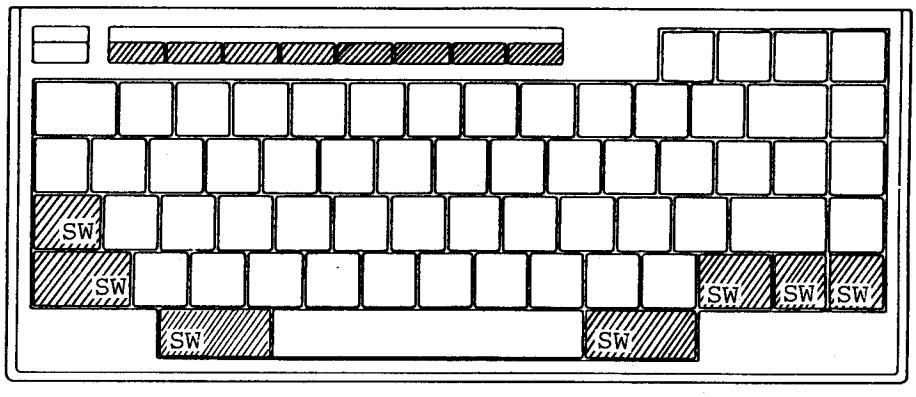
(Keys other than shaded keys)

Switch keys 


Keyboard other than Japanese-language keyboard




Auto repeat keys and switches



## 5.3 OS Key Routine Functions

- Keyboard buffers: 32 (The 7508 sub-CPU has 7 unique buffers own.)
- N-key rollover feature: Provided.
- Auto repeat feature: Provided. (See the previous page for auto repeat keys.)
  Repeat start time -- 656 ms
  Repeat period ---- 70 ms
- Auto repeat setting:
  Auto repeat ON/OFF state, repeat start time, and repeat period can be changed using the BIOS CONOUT routine.
- The CAPS, NUM, and GRAPH keyboard modes are indicated by LEDs on the keyboard.

## 5.4 Operation Flow

The steps below show the sequence of operations from key depression to transfer of the key data to the application program.

(1) A key is pressed.

(2) The 7508 scans the keyboard every 30 ms and, if a key entry is sensed, loads the corresponding hardware code into its own buffer.

(3) The 7508 reports the Z80 of a key entry via the interrupt line.

(4)(5)(6) The Z80 takes data from the 7508 buffer via the 7508 port and stores the data into the keyboard buffer. Any data overflowing the keyboard buffer is discarded.

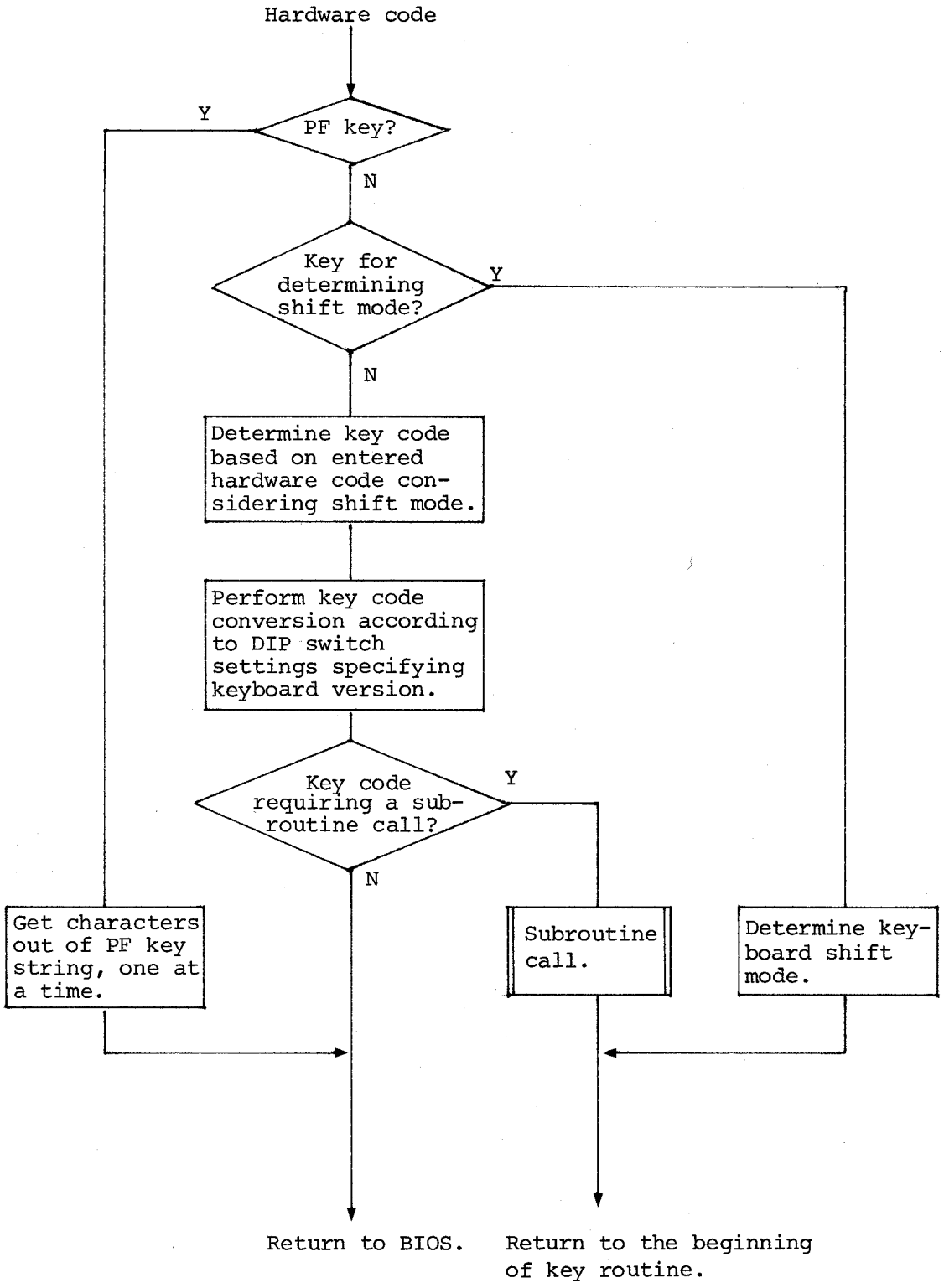
(7) The key routine takes hardware codes out of the keyboard buffer, one at a time, and returns the corresponding key codes to the application program after making the following checks:

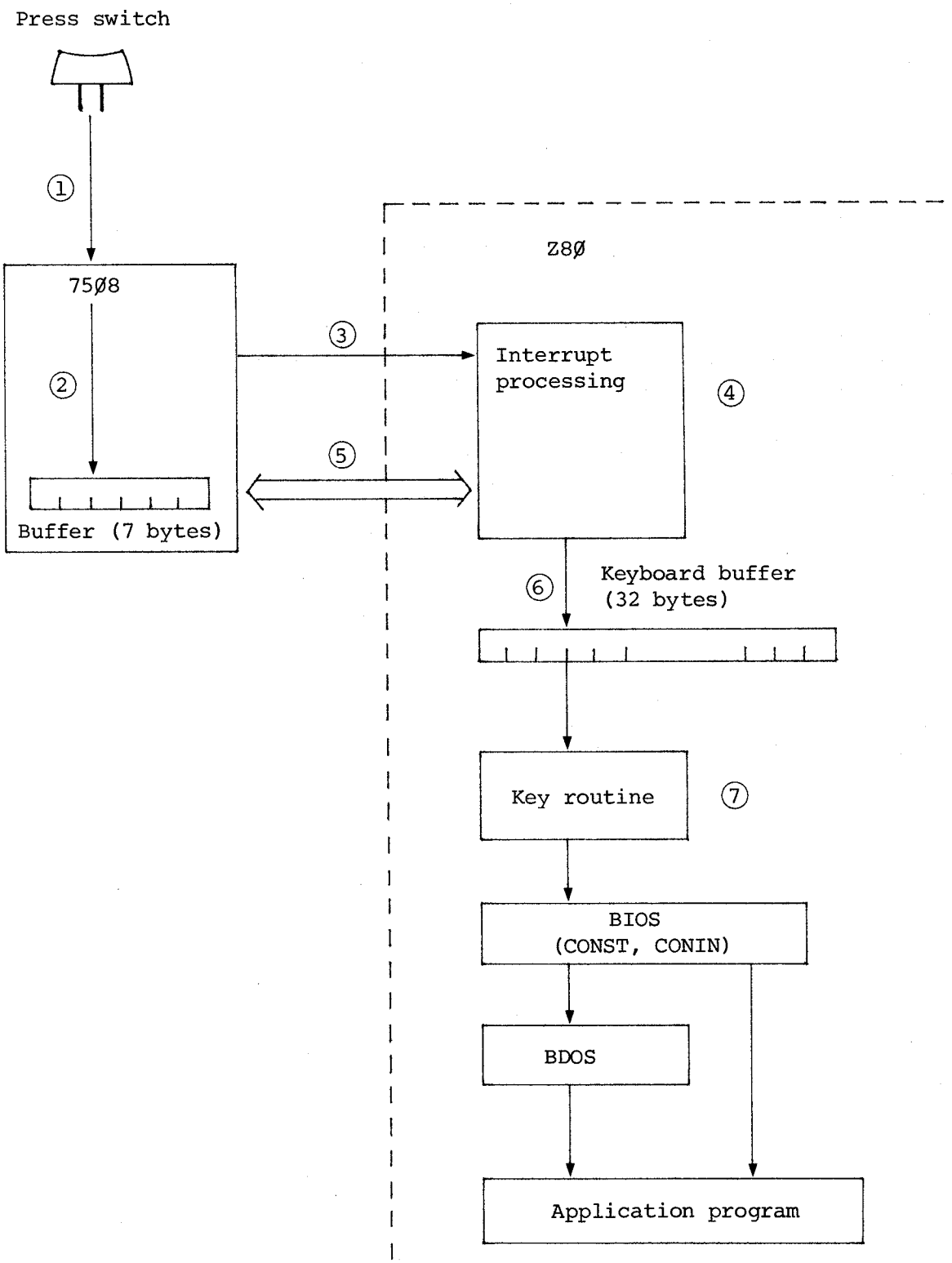
- Code for changing the keyboard mode ?
  (SHIFT key, CTRL key, etc.)
- PF key ?
- Subroutine call required?
  (CTRL/ESC - CTRL/PFK)

The above steps are illustrated in flowchart form on the next page.

* 7508 hardware codes

The 7508 hardware codes only identify the corresponding key on the keyboard and have no relation with the keyboard shift mode. Consequently, the key routine determines what code is actually entered according to the previously established state of the SHIFT, GRAPH, or CTRL key. (See Chapter 11 for details on hardware codes.)





## 5.5 Keyboard States

### 5.5.1 Keyboard Mode Transition

The MAPLE ASCII keyboard operates in three modes: Normal, CAPS and NUM. The Japanese-language OS supports the Kana mode in addition to these modes.

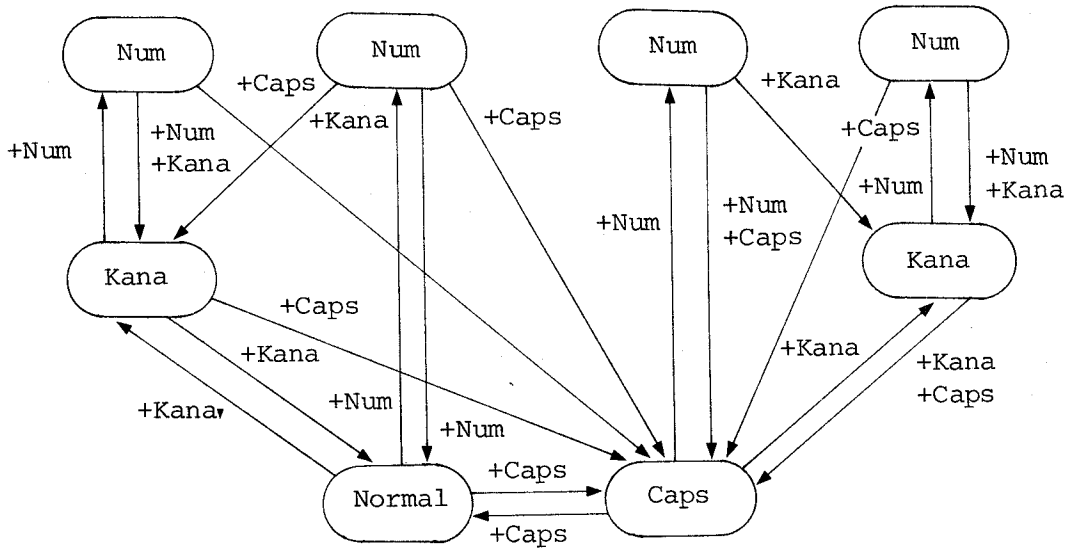
Normal: Unshifted letters are input in lowercase. For keys which have two characters on their keytop, lower letters are input.

CAPS: The same as the Normal mode except that unshifted letters are accepted in upper case.

NUM: Numbers are input from the numeric keys which are aligned horizontally on the top of the keyboard or from the keys having a number indicated at the upper right of the keytop. Some symbols are also input. Other keys are ignored.

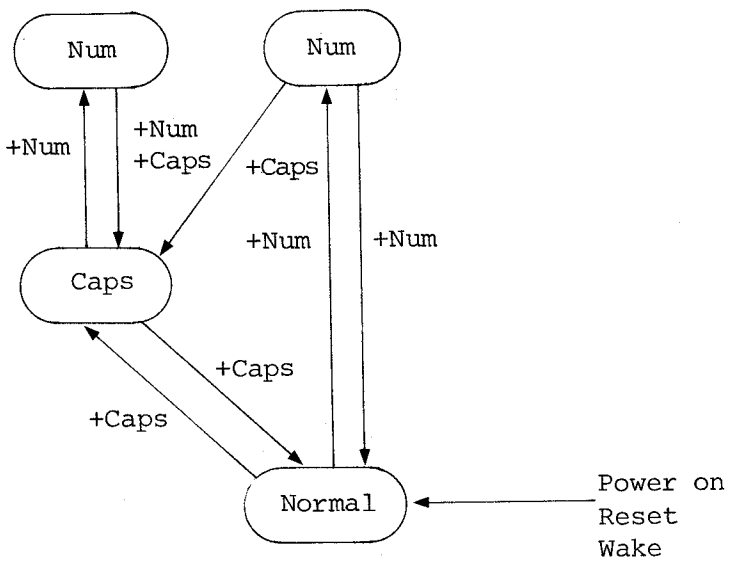
Kana: Kana characters are input.

Mode transition diagram (Kana keyboard)




Power on, Reset
Wake

Mode transition diagram (Keyboards except Kana and Japanese-language keyboards)

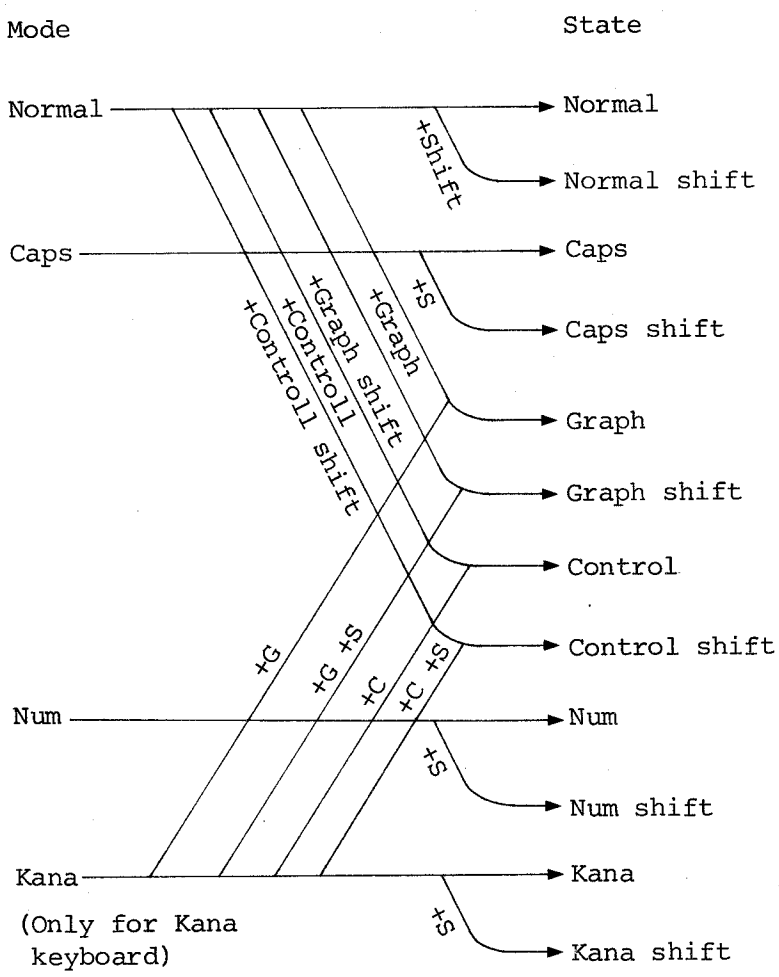


### 5.5.2 Keyboard State Transition

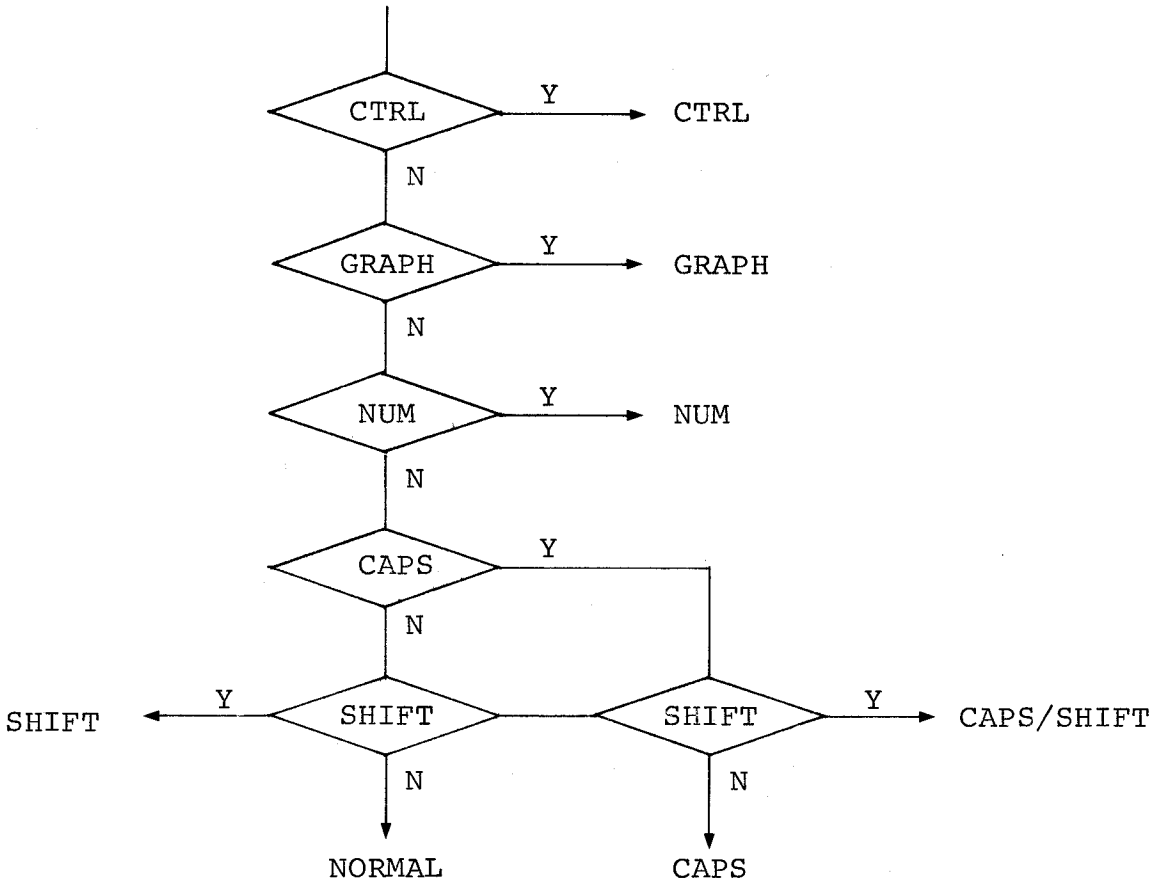
In any of the keyboard modes given in the previous subsection, a depression of a key returns different codes depending on whether the key is pressed singly or together with the SHIFT, GRPH, or CTRL key. The state transition diagram for the MAPLE keyboard is shown on the next page. The codes here refer to those codes which the application program receives from the keyboard through the BIOS CONIN function or a BDOS function.

Keyboard states (for Non-Japanese-language keyboards)

The keyboard state is determined by the combination of the keyboard mode and the state of the SHIFT, GRPH, and CTRL keys. The CTRL key has a higher priority than the GRPH key; i.e., if the CTRL and GRPH keys are pressed simultaneously, only the CTRL key is validated.




The codes received from keyboards may differ depending on the state in which the keyboard is.




The precedence of the mode keys are as follows:

1. CTRL
2. GRAPH
3. NUM
4. CAPS
5. SHIFT

The shift mode of a higher precedence is honored when two or more shift mode keys are pressed at the same time.

## 5.6 Special Keys

There are some keys which perform special functions besides returning a code when pressed. They are called special keys. The functions of the special keys are described below.

(1) STOP: Clears the key buffer and places only ^C (Ø3H) code into the buffer. Since the STOP key is normally used to interrupt program execution, when pressed, it clears all existing key codes except the ^C code off the buffer so that the MAPLE can respond immediately to this key. You can also enter ^C by typing C while holding down the CTRL key. In this case, the key buffer is not cleared at all.

(2) CTRL/STOP: This key sequence not only performs the above functions but also interrupt the current I/O operation such as an RS-232C receive operation. For example, press these keys to interrupt a program which is stalling, waiting for data from the RS-232C interface. The execution of the RS-232C receive routine is then interrupted and control is returned to the application program, which can then terminates itself by monitoring the ^C code.

Since both STOP and CTRL/STOP load the key buffer with 03H, it is impossible to tell which key was pressed from the contents of the key buffer alone. The key can be identified, however, by checking the following flags in the system work area:

CSTOPFLG --- Overseas version = F10BH

Japanese-language version = EE25H

= 00H: CTRL/STOP not pressed.

≠ 00H: CTRL/STOP pressed.

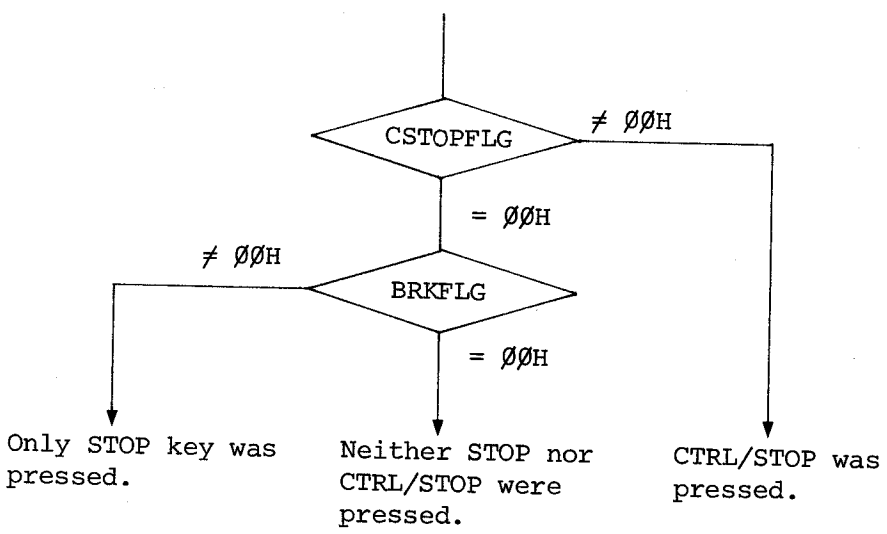
BRKFLG --- Overseas version = F10AH

Japanese-language version = EE24H

= 00H: STOP or CTRL/STOP not pressed.

≠ 00H: STOP or CTRL/STOP pressed.

Both CSTOPFLG and BRKFLG are set to 00H when the key buffer is cleared by CONIN.



(3) SHIFT/INS: Toggle between the tracking mode and non-tracking mode.

(4) CTRL/INS: Display the portion of the screen on which the cursor is currently positioned. This key sequence is used in non-tracking mode to scroll the screen up to the cursor position.

(5) Cursor movement keys (arrow keys)

There are four cursor movement keys: ↑, ↓, ←, and →. Since each of them may be pressed independently or in combination with the SHIFT or CTRL key, it may be assumed that there are logically twelve movement keys. The user can assign a code from 00H to 0FFH to each of these keys. Especially, the OS takes special actions when codes 80H and 0F8H to 0FFH are entered. These codes can be set by the application program sending ESC + F3H, ESC + F4H, and ESC + F5H via the CONOUT BIOS call.

| Code | OS action |
|---|---|
| 00H<br>\|<br>7FH | The key routine returns the code.<br>(Same as with ordinary keys.) |
| 80H | No action. |
| 81H<br>\|<br>0F7H | The key routine returns the code.<br>(Same as with ordinary keys.) |
| 0F8H | Scrolls the screen one line up. |
| 0F9H | Scrolls the screen one line down. |
| 0FAH | Scrolls the screen one page (8 lines) up. |

0FBH ---------- Scrolls the screen one page (8 lines) down.

0FCH ---------- Scrolls the screen to the top of the virtual screen.

0FDH ---------- Scrolls the screen to the bottom of the virtual screen.

0FEH ---------- Displays virtual screen 1.

0FFH ---------- Displays virtual screen 2.

The table below lists the initial values for the cursor movement keys.

| Cursor movement keys | Initial value |
|---|---|
| → | 1CH |
| ← | 1DH |
| ↑ | 1EH |
| ↓ | 1FH |
| SHIFT/ → | 8ØH |
| SHIFT/ ← | 8ØH |
| SHIFT/ ↑ | F8H |
| SHIFT/ ↓ | F9H |
| CTRL/ → | FFH |
| CTRL/ ← | FEH |
| CTRL/ ↑ | FAH |
| CTRL/ ↓ | FBH |

(6) Programmable function keys

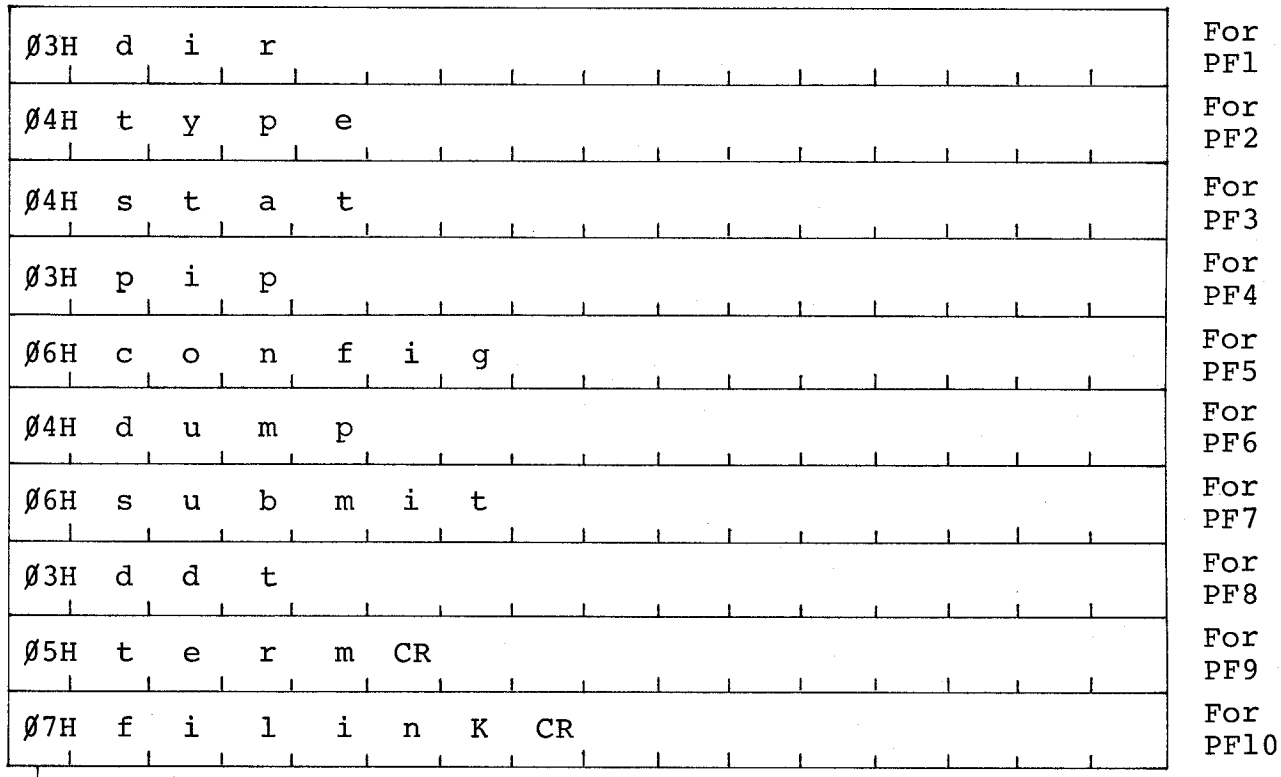
The keys PF1 to PF5 at the top of the keyboard are programmable function keys. They are used with or without the SHIFT key and are numbered PF6 (SHIFT/PF1) to PF10 (SHIFT/PF5) when used with the SHIFT key. Any string of up to 15 characters can be assigned to each of these keys in the following ways:

1) Using the BIOS PUTPFK (WBOOT + 6CH) function.

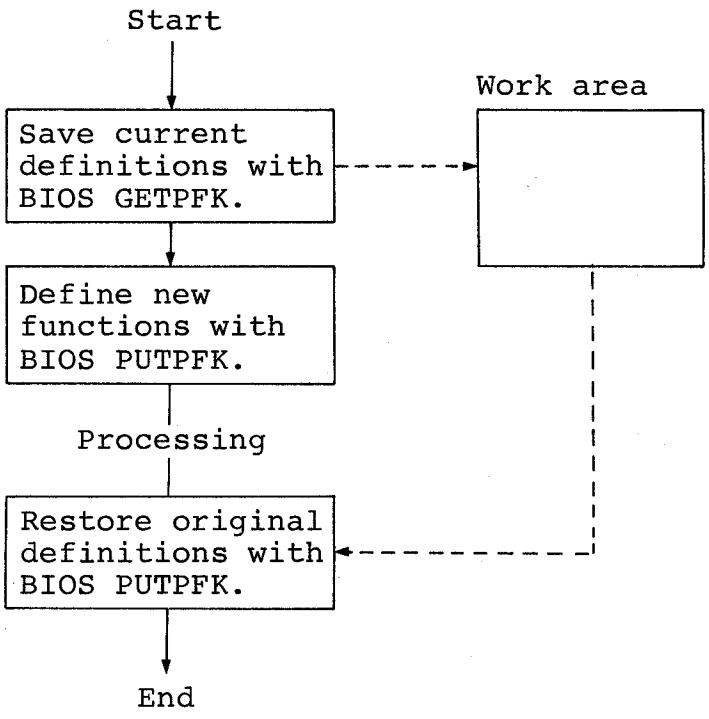
2) Defining a programmable function key table (160 bytes) having the same structure as that owned by the OS and storing its starting address in the first two bytes at YPFKSTR (0F103H) or at 0EDE9H for Japanese-language OS.

The user using the ASCII OS Version 1.0 must define the programmable function key table in a user area between addresses 8000H to 0FFFFH. If he is using the CP/M CCP area, however, he can define it in any user area at locations 8000H or higher. Those who use other operating systems may define the table anywhere in the user area.

Programmable function key table



Definitions made using the first method will be reserved until the next CBOOT (depression of the RESET switch) is executed. Therefore, if programmable function keys definitions are modified in an application program, the program must restore the table with the original definitions before terminating processing.




The second method causes the OS to restore the contents of YPFKSTR into its original programmable function key table when a WBOOT is executed. Application programs need not perform any special processing before termination.

The user is recommended to employ the second method because some application programs may unconditionally call WBOOT when a BDOS error occurs.

(7) Keys calling a subroutine

Predefined processing can be performed in the form of a subroutine call by pressing the ESC, PAUSE, HELP, or PF1 to PF5 key while holding down the CTRL key. Since the entries for such subroutines are managed in a table form, application programs can use their own key routines via key entries by changing the entry values in that table.

The starting address of the table is 0F1BAH for the overseas version and 0EED3H for the Japanese-language version.

Table structure

```
DW  XXXX : CTRL/ESC subroutine entry
DW  XXXX : CTRL/PAUSE subroutine entry
DW  XXXX : CTRL/HELP subroutine entry
DW  XXXX : CTRL/PF1 subroutine entry
DW  XXXX : CTRL/PF2 subroutine entry
DW  XXXX : CTRL/PF3 subroutine entry
DW  XXXX : CTRL/PF4 subroutine entry
DW  XXXX : CTRL/PF5 subroutine entry
```

The OS use the following two key subroutines:

CTRL/HELP: System display processing

CTRL/PF5: Hard copy processing