

8042 Keyboard Controller

(From IBM Technical Reference Manual)

Status Register

The status register is an 8-bit read-only register at I/O address hex 64. It has information about the state of the keyboard controller (8042) and interface. It may be read at any time.

Status-Register Bit Definition

- Bit 0** Output Buffer Full—A 0 indicates that the keyboard controller's output buffer has no data. A 1 indicates that the controller has placed data into its output buffer but the system has not yet read the data. When the system reads the output buffer (I/O address hex 60), this bit will return to a 0.
- Bit 1** Input Buffer Full—A 0 indicates that the keyboard controller's input buffer (I/O address hex 60 or 64) is empty. A 1 indicates that data has been written into the buffer but the controller has not read the data. When the controller reads the input buffer, this bit will return to 0.
- Bit 2** System Flag—This bit may be set to 0 or 1 by writing to the system's flag bit in the keyboard controller's command byte. It is set to 0 after a power on reset.
- Bit 3** Command/Data—The keyboard controller's input buffer may be addressed as either I/O address hex 60 or 64. Address hex 60 is defined as the data port, and address hex 64 is defined as the command port. Writing to address hex 64 sets this bit to 1; writing to address hex 60 sets this bit to 0. The controller uses this bit to determine if the byte in its input buffer should be interpreted as a command byte or a data byte.
- Bit 4** Inhibit Switch—This bit is updated whenever data is placed in the keyboard controller's output buffer. It reflects the state of the keyboard-inhibit switch. A 0 indicates the keyboard is inhibited.
- Bit 5** Transmit Time-Out—A 1 indicates that a transmission started by the keyboard controller was not properly completed. If the transmit byte was not clocked out within the specified time limit, this will be the only error. If the transmit byte was clocked out but a response was not received within the programmed time limit, the transmit time-out and receive time-out error bits are set On. If the transmit byte was clocked out but the response was received with a parity error, the transmit time-out and parity error bits are set On.
- Bit 6** Receive Time-Out—A 1 indicates that a transmission was started by the keyboard but did not finish within the programmed receive time-out delay.
- Bit 7** Parity Error—A 0 indicates the last byte of data received from the keyboard had odd parity. A 1 indicates the last byte had even parity. The keyboard should send with odd parity.

Commands (I/O Address hex 64)

- 20** Read Keyboard Controller's Command Byte—The controller sends its current command byte to its output buffer.
- 60** Write Keyboard Controller's Command Byte—The next byte of data written to I/O address hex 60 is placed in the controller's command byte. Bit definitions of the command byte are as follows:
- Bit 7** Reserved—Should be written to a 0.
- Bit 6** IBM Personal Computer Compatibility Mode—Writing a 1 to this bit causes the controller to convert the scan codes received from the keyboard to those used by the IBM Personal Computer. This includes converting a two-byte break sequence to the one-byte IBM Personal Computer format.
- Bit 5** IBM Personal Computer Mode—Writing a 1 to this bit programs the keyboard to support the IBM Personal Computer keyboard interface. In this mode the controller does not check parity or convert scan codes.
- Bit 4** Disable Keyboard—Writing a 1 to this bit disables the keyboard interface by driving the 'clock' line low. Data is not sent or received.
- Bit 3** Inhibit Override—Writing a 1 to this bit disables the keyboard inhibit function.
- Bit 2** System Flag—The value written to this bit is placed in the system flag bit of the controller's status register.
- Bit 1** Reserved—Should be written to a 0.
- Bit 0** Enable Output-Buffer-Full Interrupt—Writing a 1 to this bit causes the controller to generate an interrupt when it places data into its output buffer.
- AA** Self-Test—This commands the controller to perform internal diagnostic tests. A hex 55 is placed in the output buffer if no errors are detected.
- AB** Interface Test—This commands the controller to test the keyboard clock and data lines. The test result is placed in the output buffer as follows:
- 00** No error detected.
- 01** The 'keyboard clock' line is stuck low.
- 02** The 'keyboard clock' line is stuck high.
- 03** The 'keyboard data' line is stuck low.

04 The 'keyboard data' line is stuck high.

- AC** Diagnostic Dump—Sends 16 bytes of the controller's RAM, the current state of the input port, the current state of the output port, and the controller's program status word to the system. All items are sent in scan-code format.
- AD** Disable Keyboard Feature—This command sets bit 4 of the controller's command byte. This disables the keyboard interface by driving the clock line low. Data will not be sent or received.
- AE** Enable Keyboard Interface—This command clears bit 4 of the command byte, which releases the keyboard interface.
- C0** Read Input Port—This commands the controller to read its input port and place the data in its output buffer. This command should be used only if the output buffer is empty.
- D0** Read Output Port—This command causes the controller to read its output port and place the data in its output buffer. This command should be issued only if the output buffer is empty.
- D1** Write Output Port—The next byte of data written to I/O address hex 60 is placed in the controller's output port.
- Note:** Bit 0 of the controller's output port is connected to System Reset. This bit should not be written low.
- E0** Read Test Inputs—This command causes the controller to read its T0 and T1 inputs. This data is placed in the output buffer. Data bit 0 represents T0, and data bit 1 represents T1.
- F0–FF** Pulse Output Port—Bits 0 through 3 of the controller's output port may be pulsed low for approximately 6 microseconds. Bits 0 through 3 of this command indicate

Keyboard Commands

Commands from the System

The commands described below may be sent to the keyboard at any time. The keyboard will respond within 20 milliseconds.

Note: The following commands are those sent by the system. They have a different meaning when issued by the keyboard.

SECTION 4

Reset (Hex FF)

The system issues a Reset command to start a program reset and a keyboard internal self-test. The keyboard acknowledges the command with an 'acknowledge' signal (ACK) and ensures the system accepts the 'ACK' before executing the command. The system signals acceptance of the 'ACK' by raising the clock and data for a minimum of 500 microseconds. The keyboard is disabled from the time it receives the Reset command until the 'ACK' is accepted or until another command overrides the previous one. Following acceptance of the 'ACK', the keyboard

begins the reset operation, which is similar to a power-on reset. The keyboard clears the output buffer and sets up default values for typematic and delay rates.

Resend (Hex FE)

The system can send this command when it detects an error in any transmission from the keyboard. It can be sent only after a keyboard transmission and before the system enables the interface to allow the next keyboard output. Upon receipt of Resend, the keyboard sends the previous output again unless the previous output was Resend. In this case, the keyboard will resend the last byte before the Resend command.

No-Operation (NOP) (Hex FD through F7)

These commands are reserved and are effectively no-operation or NOP. The system does not use these codes. If sent, the keyboard will acknowledge the command and continue in its prior scanning state. No other operation will occur.

Set Default (Hex F6)

The Set Default command resets all conditions to the power-on default state. The keyboard responds with 'ACK', clears its output buffer, sets default conditions, and continues scanning (only if the keyboard was previously enabled).

Default Disable (Hex F5)

This command is similar to Set Default, except the keyboard stops scanning and awaits further instructions.

Enable (Hex F4)

Upon receipt of this command, the keyboard responds with 'ACK', clears its output buffer, and starts scanning.

Set Typematic Rate/Delay (Hex F3)

The system issues this command, followed by a parameter, to change the typematic rate and delay. The typematic rate and delay parameters are determined by the value of the byte following the command. Bits 6 and 5 serve as the delay parameter and bits 4, 3, 2, 1, and 0 (the least-significant bit) are the rate parameter. Bit 7, the most-significant bit, is always 0. The delay is equal to 1 plus the binary value of bits 6 and 5 multiplied by 250 milliseconds $\pm 20\%$. The period (interval from one typematic output to the next) is determined by the following equation:

Period = $(8 + A) \times (2^B) \times 0.00417$ seconds, where A = binary value of bits 2, 1, and 0 and B = binary value of bits 4 and 3.

The typematic rate (make code per second) is $1/\text{period}$. The period is determined by the first equation above. The following table results.

Bit Rate	Bit Rate
00000 30.0	10000 7.5
00001 26.7	10001 6.7
00010 24.0	10010 6.0
00011 21.8	10011 5.5
00100 20.0	10100 5.0
00101 18.5	10101 4.6
00110 17.1	10110 4.3
00111 16.0	10111 4.0
01000 15.0	11000 3.7
01001 13.3	11001 3.3
01010 12.0	11010 3.0
01011 10.9	11011 2.7
01100 10.0	11100 2.5
01101 9.2	11101 2.3
01110 8.6	11110 2.1
01111 8.0	11111 2.0

Typematic Rate

The keyboard responds to the Set Typematic Rate Delay command with an 'ACK', stops scanning, and waits for the rate parameter. The keyboard responds to the rate parameter with another 'ACK', sets the rate and delay, and continues scanning (if the keyboard was previously enabled). If a command is received instead of the rate parameter, the set-typematic-rate

function ends with no change to the existing rate, and the new command is processed. However, the keyboard will not resume scanning unless instructed to do so by an Enable command.

The default rate for the system keyboard is as follows:

The typematic rate = 10 characters per second $\pm 20\%$ and the delay = 500 ms $\pm 20\%$.

No-Operation (NOP) (Hex F2 through EF)

These commands are reserved and are effectively no-operation (NOP). The system does not use these codes. If sent, the keyboard acknowledges the command and continues in its prior scanning state. No other operation will occur.

Echo (Hex EE)

Echo is a diagnostic aide. When the keyboard receives this command, it issues a hex EE response and continues scanning if the keyboard was previously enabled.

Set/Reset Mode Indicators (Hex ED)

Three mode indicators on the keyboard are accessible to the system. The keyboard activates or deactivates these indicators when it receives a valid command from the system. They can be activated or deactivated in any combination.

It is up to the using system to remember the previous state of an indicator. This is in case its setting does not change when a command sequence is issued to change the state of another indicator.

The system remembers the previous state of an indicator so that its setting does not change when a command sequence is issued to change the state of another indicator.

The command has the following format:

Command	Options
---------	---------

Set/Reset Command

A Set/Reset Mode Indicators command consists of two bytes. The first is the command byte and has the following bit setup:

11101101 – hex ED

The second byte is an option byte. It has a list of the indicators to be acted upon. The format of the option byte is as follows:

- Bit 7** Reserved
- Bit 6** Reserved
- Bit 5** Reserved
- Bit 4** Reserved
- Bit 3** Reserved
- Bit 2** Caps Lock indicator
- Bit 1** Numeric Lock indicator
- Bit 0** Scroll Lock indicator

Note: Bit 7 is the most-significant bit; bit 0 is the least-significant.

The keyboard will respond to the Set/Reset Mode Indicators command with an 'ACK', discontinue scanning, and wait for the option byte. The keyboard will respond to the option byte with an Ack, set the indicators, and continue scanning if the keyboard was previously enabled. If another command is received in place of the option byte, execution of the function of the Set/Reset Mode Indicators command is stopped with no change to the indicator states, and the new command is processed. Then scanning is resumed.

Initializing the 8042 Keyboard Controller

```
STATUS_PORT      EQU      064H
KBD_OUT_BUF      EQU      060H
KBD_INPT_BUF     EQU      060H
KBD_CMD_BUF      EQU      064H
KBD_DATA_BUF     EQU      060H
PORT_A           EQU      060H
PORT_B           EQU      061H
OUT_BUF_FULL     EQU      01H
INPT_BUF_FULL    EQU      02H
SYS_FLAG         EQU      04H
CMD_DATA         EQU      08H
KEYBD_INH       EQU      10H
TRANS_TMOUT      EQU      20H
RCV_TMOUT        EQU      40H
PARITY_EVEN      EQU      80H
INH_KEYBOARD     EQU      10H
```

```
KBD_ENA          EQU      0AEH
KBD_DIS          EQU      0ADH
```

; Keyboard Commands

```
KB_MENU          EQU      0F1H
KB_ENABLE        EQU      0F4H
KB_MAKEBREAK     EQU      0F7H
KB_ECHO          EQU      0FEH
KB_RESET         EQU      0FFH
KB_LED_CMD       EQU      0EDH
```

; Keyboard responses

```
KB_OK            EQU      0AAH
KB_ACK           EQU      0FAH
KB_OVERRUN       EQU      0FFH
KB_RESEND        EQU      0FEH
KB_BREAK         EQU      0F0H
KB_FA            EQU      010H
KB_FE            EQU      020H
KB_PR_LED        EQU      040H
```

```

;*****
;INITKBD
; This procedure intializes the 8042
;*****

INITKBD    PROC    FAR

            PUSH    AX
            PUSH    DX
            PUSH    CX

KBD1:      SUB     CX, CX
            IN     AL, STATUS_PORT    ; wait for input buffer empty
            TEST   AL, INPT_BUF_FULL
            LOOPNZ KBD1

            MOV    AL, 0AAH          ;Send self test command
            OUT    KBD_CMD_BUF, AL

KBD2:      SUB     CX, CX            ; Wait for test to complete
            IN     AL, STATUS_PORT
            TEST   AL, OUT_BUF_FULL
            LOOPZ  KBD2

            IN     AL, KBD_OUT_BUF   ;Check self test result
            CMP    AL, 055H
            JNE    KBDF

            MOV    AL, 0ABH          ;Send test Interface command
            OUT    KBD_CMD_BUF, AL

KBD3:      SUB     CX, CX            ; wait for test to complete
            IN     AL, STATUS_PORT
            TEST   AL, OUT_BUF_FULL
            LOOPZ  KBD3

            IN     AL, KBD_OUT_BUF   ;Display Interface test results
            CMP    AL, 000H          ;Should be 00
            JNE    KBDF

            MOV    AL, 060H          ;Send command byte
            OUT    KBD_CMD_BUF, AL
            SUB    CX, CX

KBD4:      IN     AL, STATUS_PORT    ;Wait until input buffer
            TEST   AL, INPT_BUF_FULL ;is not full
            LOOPNZ KBD4

            MOV    AL, 01101001B    ;Send command byte
            OUT    KBD_DATA_BUF, AL

KBD5:      SUB     CX, CX
            IN     AL, STATUS_PORT    ;Wait until input buffer is not full
            TEST   AL, INPT_BUF_FULL
            LOOPNZ KBD5

            MOV    AL, 0FFH          ; Send Keyboard Reset command
            OUT    KBD_DATA_BUF,AL
            CALL   DLY1

KBD6:      SUB     CX, CX
            IN     AL, STATUS_PORT    ;Wait until output buffer is full
            TEST   AL, OUT_BUF_FULL
            LOOPZ  KBD6
            IN     AL, KBD_OUT_BUF

            MOV    AL, 0EEH          ; Send Echo EE Command
            OUT    KBD_DATA_BUF,AL
            CALL   DLY1

```

```

KBD7:      SUB     CX, CX
           IN      AL, STATUS_PORT      ;Wait until output buffer is full
           TEST    AL, OUT_BUF_FULL
           LOOPZ   KBD7

           IN      AL, KBD_OUT_BUF      ; Read Echo from Keyboard
           CALL    DLY1
           CMP     AL, 0EEH
           JNE     KBDF

           MOV     AL, 0F4H             ; Enable the Keyboard
           OUT     KBD_DATA_BUF, AL

KBD8:      SUB     CX, CX
           IN      AL, STATUS_PORT      ;Wait until output buffer is full
           TEST    AL, OUT_BUF_FULL
           LOOPZ   KBD8

           IN      AL, KBD_OUT_BUF      ; Clear the output buffer

           POP     CX
           POP     DX
           POP     AX
           RET

KBDF:      MOV     AL, 0FFH
           AL2DPLY
           JMP     $

INITKBD    ENDP

;*****
C8042      PROC FAR
           OUT     STATUS_PORT, AL

C42:      SUB     CX, CX
           IN      AL, STATUS_PORT
           TEST    AL, INPT_BUF_FULL
           LOOPNZ  C42
           RET

C8042      ENDP

```