

Appendix C: Keyboard Scan Codes

Table 90: PC Keyboard Scan Codes (in hex)

| Key | Down | Up | Key | Down | Up | Key | Down | Up | Key | Down | Up |
|------|------|----|---------|------|----|-------------|------|----|---------------|-------------------------|----|
| Esc | 1 | 81 | [{ | 1A | 9A | , < | 33 | B3 | <i>center</i> | 4C | CC |
| 1! | 2 | 82 |]} | 1B | 9B | . > | 34 | B4 | <i>right</i> | 4D | CD |
| 2@ | 3 | 83 | Enter | 1C | 9C | /? | 35 | B5 | <i>+</i> | 4E | CE |
| 3# | 4 | 84 | Ctrl | 1D | 9D | R shift | 36 | B6 | <i>end</i> | 4F | CF |
| 4\$ | 5 | 85 | A | 1E | 9E | *PrtSc | 37 | B7 | <i>down</i> | 50 | D0 |
| 5% | 6 | 86 | S | 1F | 9F | alt | 38 | B8 | <i>pgdn</i> | 51 | D1 |
| 6^ | 7 | 87 | D | 20 | A0 | space | 39 | B9 | <i>ins</i> | 52 | D2 |
| 7& | 8 | 88 | F | 21 | A1 | CAPS | 3A | BA | <i>del</i> | 53 | D3 |
| 8* | 9 | 89 | G | 22 | A2 | F1 | 3B | BB | / | E0 35 | B5 |
| 9(| 0A | 8A | H | 23 | A3 | F2 | 3C | BC | <i>enter</i> | E0 1C | 9C |
| 0) | 0B | 8B | J | 24 | A4 | F3 | 3D | BD | F11 | 57 | D7 |
| -_ | 0C | 8C | K | 25 | A5 | F4 | 3E | BE | F12 | 58 | D8 |
| =+ | 0D | 8D | L | 26 | A6 | F5 | 3F | BF | ins | E0 52 | D2 |
| Bksp | 0E | 8E | :: | 27 | A7 | F6 | 40 | C0 | del | E0 53 | D3 |
| Tab | 0F | 8F | ‘“ | 28 | A8 | F7 | 41 | C1 | home | E0 47 | C7 |
| Q | 10 | 90 | `~ | 29 | A9 | F8 | 42 | C2 | end | E0 4F | CF |
| W | 11 | 91 | L shift | 2A | AA | F9 | 43 | C3 | pgup | E0 49 | C9 |
| E | 12 | 92 | \ | 2B | AB | F10 | 44 | C4 | pgdn | E0 51 | D1 |
| R | 13 | 93 | Z | 2C | AC | NUM | 45 | C5 | left | E0 4B | CB |
| T | 14 | 94 | X | 2D | AD | SCRL | 46 | C6 | right | E0 4D | CD |
| Y | 15 | 95 | C | 2E | AE | <i>home</i> | 47 | C7 | up | E0 48 | C8 |
| U | 16 | 96 | V | 2F | AF | <i>up</i> | 48 | C8 | down | E0 50 | D0 |
| I | 17 | 97 | B | 30 | B0 | <i>pgup</i> | 49 | C9 | R alt | E0 38 | B8 |
| O | 18 | 98 | N | 31 | B1 | - | 4A | CA | R ctrl | E0 1D | 9D |
| P | 19 | 99 | M | 32 | B2 | <i>left</i> | 4B | CB | Pause | E1 1D 45 E1 9D C5 | - |

Table 91: Keyboard Codes (in hex)

| Key | Scan Code | ASCII | Shift ^a | Ctrl | Alt | Num | Caps | Shift Caps | Shift Num |
|--------|-----------|-------|--------------------|-------------|-------------|-----|------|-------------|-------------|
| Esc | 01 | 1B | 1B | 1B | | 1B | 1B | 1B | 1B |
| 1 ! | 02 | 31 | 21 | | 7800 | 31 | 31 | 31 | 31 |
| 2 @ | 03 | 32 | 40 | 0300 | 7900 | 32 | 32 | 32 | 32 |
| 3 # | 04 | 33 | 23 | | 7A00 | 33 | 33 | 33 | 33 |
| 4 \$ | 05 | 34 | 24 | | 7B00 | 34 | 34 | 34 | 34 |
| 5 % | 06 | 35 | 25 | | 7C00 | 35 | 35 | 35 | 35 |
| 6 ^ | 07 | 36 | 5E | 1E | 7D00 | 36 | 36 | 36 | 36 |
| 7 & | 08 | 37 | 26 | | 7E00 | 37 | 37 | 37 | 37 |
| 8 * | 09 | 38 | 2A | | 7F00 | 38 | 38 | 38 | 38 |
| 9 (| 0A | 39 | 28 | | 8000 | 39 | 39 | 39 | 39 |
| 0) | 0B | 30 | 29 | | 8100 | 30 | 30 | 30 | 30 |
| - _ | 0C | 2D | 5F | 1F | 8200 | 2D | 2D | 5F | 5F |
| = + | 0D | 3D | 2B | | 8300 | 3D | 3D | 2B | 2B |
| Bksp | 0E | 08 | 08 | 7F | | 08 | 08 | 08 | 08 |
| Tab | 0F | 09 | 0F00 | | | 09 | 09 | 0F00 | 0F00 |
| Q | 10 | 71 | 51 | 11 | 1000 | 71 | 51 | 71 | 51 |
| W | 11 | 77 | 57 | 17 | 1100 | 77 | 57 | 77 | 57 |
| E | 12 | 65 | 45 | 05 | 1200 | 65 | 45 | 65 | 45 |
| R | 13 | 72 | 52 | 12 | 1300 | 72 | 52 | 72 | 52 |
| T | 14 | 74 | 54 | 14 | 1400 | 74 | 54 | 74 | 54 |
| Y | 15 | 79 | 59 | 19 | 1500 | 79 | 59 | 79 | 59 |
| U | 16 | 75 | 55 | 15 | 1600 | 75 | 55 | 75 | 55 |
| I | 17 | 69 | 49 | 09 | 1700 | 69 | 49 | 69 | 49 |
| O | 18 | 6F | 4F | 0F | 1800 | 6F | 4F | 6F | 4F |
| P | 19 | 70 | 50 | 10 | 1900 | 70 | 50 | 70 | 50 |
| [{ | 1A | 5B | 7B | 1B | | 5B | 5B | 7B | 7B |
|]} | 1B | 5D | 7D | 1D | | 5D | 5D | 7D | 7D |
| enter | 1C | 0D | 0D | 0A | | 0D | 0D | 0A | 0A |
| ctrl | 1D | | | | | | | | |
| A | 1E | 61 | 41 | 01 | 1E00 | 61 | 41 | 61 | 41 |
| S | 1F | 73 | 53 | 13 | 1F00 | 73 | 53 | 73 | 53 |
| D | 20 | 64 | 44 | 04 | 2000 | 64 | 44 | 64 | 44 |
| F | 21 | 66 | 46 | 06 | 2100 | 66 | 46 | 66 | 46 |
| G | 22 | 67 | 47 | 07 | 2200 | 67 | 47 | 67 | 47 |
| H | 23 | 68 | 48 | 08 | 2300 | 68 | 48 | 68 | 48 |
| J | 24 | 6A | 4A | 0A | 2400 | 6A | 4A | 6A | 4A |
| K | 25 | 6B | 4B | 0B | 2500 | 6B | 4B | 6B | 4B |
| L | 26 | 6C | 4C | 0C | 2600 | 6C | 4C | 6C | 4C |
| ; : | 27 | 3B | 3A | | | 3B | 3B | 3A | 3A |
| ' " | 28 | 27 | 22 | | | 27 | 27 | 22 | 22 |
| ~ | 29 | 60 | 7E | | | 60 | 60 | 7E | 7E |
| Lshift | 2A | | | | | | | | |
| \ | 2B | 5C | 7C | 1C | | 5C | 5C | 7C | 7C |
| Z | 2C | 7A | 5A | 1A | 2C00 | 7A | 5A | 7A | 5A |
| X | 2D | 78 | 58 | 18 | 2D00 | 78 | 58 | 78 | 58 |
| C | 2E | 63 | 43 | 03 | 2E00 | 63 | 43 | 63 | 43 |
| V | 2F | 76 | 56 | 16 | 2F00 | 76 | 56 | 76 | 56 |
| B | 30 | 62 | 42 | 02 | 3000 | 62 | 42 | 62 | 42 |
| Key | Scan Code | ASCII | Shift | Ctrl | Alt | Num | Caps | Shift Caps | Shift Num |

Table 91: Keyboard Codes (in hex)

| Key | Scan Code | ASCII | Shift ^a | Ctrl | Alt | Num | Caps | Shift Caps | Shift Num |
|----------------|-----------|-------------|--------------------|-----------------|-------------|-------------|-------------|-------------|-------------|
| N | 31 | 6E | 4E | 0E | 3100 | 6E | 4E | 6E | 4E |
| M | 32 | 6D | 4D | 0D | 3200 | 6D | 4D | 6D | 4D |
| , < | 33 | 2C | 3C | | | 2C | 2C | 3C | 3C |
| . > | 34 | 2E | 3E | | | 2E | 2E | 3E | 3E |
| / ? | 35 | 2F | 3F | | | 2F | 2F | 3F | 3F |
| Rshift | 36 | | | | | | | | |
| * PrtSc | 37 | 2A | INT 5 ^b | 10 ^c | | 2A | 2A | INT 5 | INT 5 |
| alt | 38 | | | | | | | | |
| space | 39 | 20 | 20 | 20 | | 20 | 20 | 20 | 20 |
| caps | 3A | | | | | | | | |
| F1 | 3B | 3B00 | 5400 | 5E00 | 6800 | 3B00 | 3B00 | 5400 | 5400 |
| F2 | 3C | 3C00 | 5500 | 5F00 | 6900 | 3C00 | 3C00 | 5500 | 5500 |
| F3 | 3D | 3D00 | 5600 | 6000 | 6A00 | 3D00 | 3D00 | 5600 | 5600 |
| F4 | 3E | 3E00 | 5700 | 6100 | 6B00 | 3E00 | 3E00 | 5700 | 5700 |
| F5 | 3F | 3F00 | 5800 | 6200 | 6C00 | 3F00 | 3F00 | 5800 | 5800 |
| F6 | 40 | 4000 | 5900 | 6300 | 6D00 | 4000 | 4000 | 5900 | 5900 |
| F7 | 41 | 4100 | 5A00 | 6400 | 6E00 | 4100 | 4100 | 5A00 | 5A00 |
| F8 | 42 | 4200 | 5B00 | 6500 | 6F00 | 4200 | 4200 | 5B00 | 5B00 |
| F9 | 43 | 4300 | 5C00 | 6600 | 7000 | 4300 | 4300 | 5C00 | 5C00 |
| F10 | 44 | 4400 | 5D00 | 6700 | 7100 | 4400 | 4400 | 5D00 | 5D00 |
| num | 45 | | | | | | | | |
| scrl | 46 | | | | | | | | |
| home | 47 | 4700 | 37 | 7700 | | 37 | 4700 | 37 | 4700 |
| up | 48 | 4800 | 38 | | | 38 | 4800 | 38 | 4800 |
| pgup | 49 | 4900 | 39 | 8400 | | 39 | 4900 | 39 | 4900 |
| _d | 4A | 2D | 2D | | | 2D | 2D | 2D | 2D |
| left | 4B | 4B00 | 34 | 7300 | | 34 | 4B00 | 34 | 4B00 |
| center | 4C | 4C00 | 35 | | | 35 | 4C00 | 35 | 4C00 |
| right | 4D | 4D00 | 36 | 7400 | | 36 | 4D00 | 36 | 4D00 |
| + ^e | 4E | 2B | 2B | | | 2B | 2B | 2B | 2B |
| end | 4F | 4F00 | 31 | 7500 | | 31 | 4F00 | 31 | 4F00 |
| down | 50 | 5000 | 32 | | | 32 | 5000 | 32 | 5000 |
| pgdn | 51 | 5100 | 33 | 7600 | | 33 | 5100 | 33 | 5100 |
| ins | 52 | 5200 | 30 | | | 30 | 5200 | 30 | 5200 |
| del | 53 | 5300 | 2E | | | 2E | 5300 | 2E | 5300 |
| Key | Scan Code | ASCII | Shift | Ctrl | Alt | Num | Caps | Shift Caps | Shift Num |

a. For the alphabetic characters, if capslock is active then see the shift-capslock column.

b. Pressing the PrtSc key does not produce a scan code. Instead, BIOS executes an int 5 instruction which should print the screen.

c. This is the control-P character that will activate the printer under MS-DOS.

d. This is the minus key on the keypad.

e. This is the plus key on the keypad.

Table 92: Keyboard Related BIOS Variables

| Name | Address ^a | Size | Description |
|---------------------------------|----------------------|------|---|
| KbdFlags1 (modifier flags) | 40:17 | Byte | This byte maintains the current status of the modifier keys on the keyboard. The bits have the following meanings: bit 7: Insert mode toggle bit 6: Capslock toggle (1=capslock on) bit 5: Numlock toggle (1=numlock on) bit 4: Scroll lock toggle (1=scroll lock on) bit 3: Alt key (1=alt is down) bit 2: Ctrl key (1=ctrl is down) bit 1: Left shift key (1=left shift is down) bit 0: Right shift key (1=right shift is down) |
| KbdFlags2 (Toggle keys down) | 40:18 | Byte | Specifies if a toggle key is currently down. bit 7: Insert key (currently down if 1) bit 6: Capslock key (currently down if 1) bit 5: Numlock key (currently down if 1) bit 4: Scroll lock key (currently down if 1) bit 3: Pause state locked (ctrl-Numlock) if one bit 2: SysReq key (currently down if 1) bit 1: Left alt key (currently down if 1) bit 0: Left ctrl key (currently down if 1) |
| AltKpd | 40:19 | Byte | BIOS uses this to compute the ASCII code for an alt-Key-pad sequence. |
| BufStart | 40:80 | Word | Offset of start of keyboard buffer (1Eh). Note: this variable is not supported on many systems, be careful if you use it. |
| BufEnd | 40:82 | Word | Offset of end of keyboard buffer (3Eh). See the note above. |
| KbdFlags3 | 40:96 | Byte | Miscellaneous keyboard flags. bit 7: Read of keyboard ID in progress bit 6: Last char is first kbd ID character bit 5: Force numlock on reset bit 4: 1 if 101-key kbd, 0 if 83/84 key kbd. bit 3: Right alt key pressed if 1 bit 2: Right ctrl key pressed if 1 bit 1: Last scan code was E0h bit 0: Last scan code was E1h |
| KbdFlags4 | 40:97 | Byte | More miscellaneous keyboard flags. bit 7: Keyboard transmit error bit 6: Mode indicator update bit 5: Resend receive flag bit 4: Acknowledge received bit 3: Must always be zero bit 2: Capslock LED (1=on) bit 1: Numlock LED (1=on) bit 0: Scroll lock LED (1=on) |

a. Addresses are all given in hexadecimal

Table 93: On-Board Keyboard Controller Commands (Port 64h)

| Value (hex) | Description |
|-------------|--|
| 20 | Transmit keyboard controller's command byte to system as a scan code at port 60h. |
| 60 | The next byte written to port 60h will be stored in the keyboard controller's command byte. |
| A4 | Test if a password is installed (PS/2 only). Result comes back in port 60h. 0FAh means a password is installed, 0F1h means no password. |
| A5 | Transmit password (PS/2 only). Starts receipt of password. The next sequence of scan codes written to port 60h, ending with a zero byte, are the new password. |
| A6 | Password match. Characters from the keyboard are compared to password until a match occurs. |
| A7 | Disable mouse device (PS/2 only). Identical to setting bit five of the command byte. |
| A8 | Enable mouse device (PS/2 only). Identical to clearing bit five of the command byte. |
| A9 | Test mouse device. Returns 0 if okay, 1 or 2 if there is a stuck clock, 3 or 4 if there is a stuck data line. Results come back in port 60h. |
| AA | Initiates self-test. Returns 55h in port 60h if successful. |
| AB | Keyboard interface test. Tests the keyboard interface. Returns 0 if okay, 1 or 2 if there is a stuck clock, 3 or 4 if there is a stuck data line. Results come back in port 60h. |
| AC | Diagnostic. Returns 16 bytes from the keyboard's microcontroller chip. Not available on PS/2 systems. |
| AD | Disable keyboard. Same operation as setting bit four of the command register. |
| AE | Enable keyboard. Same operation as clearing bit four of the command register. |
| C0 | Read keyboard input port to port 60h. This input port contains the following values: bit 7: Keyboard inhibit keyswitch (0 = inhibit, 1 = enabled). bit 6: Display switch (0=color, 1=mono). bit 5: Manufacturing jumper. bit 4: System board RAM (always 1). bits 0-3: undefined. |
| C1 | Copy input port (above) bits 0-3 to status bits 4-7. (PS/2 only) |
| C2 | Copy input port (above) bits 4-7 to status port bits 4-7. (PS/2 only). |
| D0 | Copy microcontroller output port value to port 60h (see definition below). |
| D1 | Write the next data byte written to port 60h to the microcontroller output port. This port has the following definition: bit 7: Keyboard data. bit 6: Keyboard clock. bit 5: Input buffer empty flag. bit 4: Output buffer full flag. bit 3: Undefined. bit 2: Undefined. bit 1: Gate A20 line. bit 0: System reset (if zero). Note: writing a zero to bit zero will reset the machine. Writing a one to bit one combines address lines 19 and 20 on the PC's address bus. |

Table 93: On-Board Keyboard Controller Commands (Port 64h)

| Value (hex) | Description |
|-------------|---|
| D2 | Write keyboard buffer. The keyboard controller returns the next value sent to port 60h as though a keypress produced that value. (PS/2 only). |
| D3 | Write mouse buffer. The keyboard controller returns the next value sent to port 60h as though a mouse operation produced that value. (PS/2 only). |
| D4 | Writes the next data byte (60h) to the mouse (auxiliary) device. (PS/2 only). |
| E0 | Read test inputs. Returns in port 60h the status of the keyboard serial lines. Bit zero contains the keyboard clock input, bit one contains the keyboard data input. |
| Ex | Pulse output port (see definition for D1). Bits 0-3 of the keyboard controller command byte are pulsed onto the output port. Resets the system if bit zero is a zero. |

Table 94: Keyboard to System Transmissions

| Value (hex) | Description |
|-----------------|--|
| 00 | Data overrun. System sends a zero byte as the last value when the keyboard controller's internal buffer overflows. |
| 1..58 81..D8 | Scan codes for key presses. The positive values are down codes, the negative values (H.O. bit set) are up codes. |
| 83AB | Keyboard ID code returned in response to the F2 command (PS/2 only). |
| AA | Returned during basic assurance test after reset. Also the up code for the left shift key. |
| EE | Returned by the ECHO command. |
| F0 | Prefix to certain up codes (N/A on PS/2). |
| FA | Keyboard acknowledge to keyboard commands other than resend or ECHO. |
| FC | Basic assurance test failed (PS/2 only). |
| FD | Diagnostic failure (not available on PS/2). |
| FE | Resend. Keyboard requests the system to resend the last command. |
| FF | Key error (PS/2 only). |

Table 95: Keyboard Microcontroller Commands (Port 60h)

| Value (hex) | Description |
|-------------|--|
| ED | Send LED bits. The next byte written to port 60h updates the LEDs on the keyboard. The parameter (next) byte contains: bits 3-7: Must be zero. bit 2: Capslock LED (1 = on, 0 = off). bit 1: Numlock LED (1 = on, 0 = off). bit 0: Scroll lock LED (1 = on, 0 = off). |
| EE | Echo commands. Returns 0EEh in port 60h as a diagnostic aid. |
| F0 | Select alternate scan code set (PS/2 only). The next byte written to port 60h selects one of the following options: 00: Report current scan code set in use (next value read from port 60h). 01: Select scan code set #1 (standard PC/AT scan code set). 02: Select scan code set #2. 03: Select scan code set #3. |
| F2 | Send two-byte keyboard ID code as the next two bytes read from port 60h (PS/2 only). |
| F3 | Set Autorepeat delay and repeat rate. Next byte written to port 60h determines rate: bit 7: must be zero bits 5,6: Delay. 00- 1/4 sec, 01- 1/2 sec, 10- 3/4 sec, 11- 1 sec. bits 0-4: Repeat rate. 0- approx 30 chars/sec to 1Fh- approx 2 chars/sec. |
| F4 | Enable keyboard. |
| F5 | Reset to power on condition and wait for enable command. |
| F6 | Reset to power on condition and begin scanning keyboard. |
| F7 | Make all keys autorepeat (PS/2 only). |
| F8 | Set all keys to generate an up code and a down code (PS/2 only). |
| F9 | Set all keys to generate an up code only (PS/2 only). |
| FA | Set all keys to autorepeat and generate up and down codes (PS/2 only). |
| FB | Set an individual key to autorepeat. Next byte contains the scan code of the desired key. (PS/2 only). |
| FC | Set an individual key to generate up and down codes. Next byte contains the scan code of the desired key. (PS/2 only). |
| FD | Set an individual key to generate only down codes. Next byte contains the scan code of the desired key. (PS/2 only). |
| FE | Resend last result. Use this command if there is an error receiving data. |
| FF | Reset keyboard to power on state and start the self-test. |

Table 96: BIOS Keyboard Support Functions

| Function # (AH) | Input Parameters | Output Parameters | Description |
|-----------------|---|--|--|
| 0 | | a1- ASCII character ah- scan code | Read character. Reads next available character from the system's type ahead buffer. Wait for a keystroke if the buffer is empty. |
| 1 | | ZF- Set if no key. ZF- Clear if key available. a1- ASCII code ah- scan code | Checks to see if a character is available in the type ahead buffer. Sets the zero flag if not key is available, clears the zero flag if a key is available. If there is an available key, this function returns the ASCII and scan code value in ax. The value in ax is undefined if no key is available. |
| 2 | | a1- shift flags | Returns the current status of the shift flags in a1. The shift flags are defined as follows: bit 7: Insert toggle bit 6: Capslock toggle bit 5: Numlock toggle bit 4: Scroll lock toggle bit 3: Alt key is down bit 2: Ctrl key is down bit 1: Left shift key is down bit 0: Right shift key is down |
| 3 | a1 = 5 bh = 0, 1, 2, 3 for 1/4, 1/2, 3/4, or 1 second delay b1= 0..1Fh for 30/sec to 2/sec. | | Set auto repeat rate. The bh register contains the amount of time to wait before starting the autorepeat operation, the b1 register contains the autorepeat rate. |
| 5 | ch = scan code c1 = ASCII code | | Store keycode in buffer. This function stores the value in the cx register at the end of the type ahead buffer. Note that the scan code in ch doesn't have to correspond to the ASCII code appearing in c1. This routine will simply insert the data you provide into the system type ahead buffer. |
| 10h | | a1- ASCII character ah- scan code | Read extended character. Like ah=0 call, except this one passes all key codes, the ah=0 call throws away codes that are not PC/XT compatible. |
| 11h | | ZF- Set if no key. ZF- Clear if key available. a1- ASCII code ah- scan code | Like the ah=01h call except this one does not throw away keycodes that are not PC/XT compatible (i.e., the extra keys found on the 101 key keyboard). |

Table 96: BIOS Keyboard Support Functions

| Function # (AH) | Input Parameters | Output Parameters | Description |
|--------------------|---------------------|--|--|
| 12h | | al- shift flags ah- extended shift flags | Returns the current status of the shift flags in ax. The shift flags are defined as follows: bit 15: SysReq key pressed bit 14: Capslock key currently down bit 13: Numlock key currently down bit 12: Scroll lock key currently down bit 11: Right alt key is down bit 10: Right ctrl key is down bit 9: Left alt key is down bit 8: Left ctrl key is down bit 7: Insert toggle bit 6: Capslock toggle bit 5: Numlock toggle bit 4: Scroll lock toggle bit 3: Either alt key is down (some machines, left only) bit 2: Either ctrl key is down bit 1: Left shift key is down bit 0: Right shift key is down |

