**INT 09h (9) Keyboard**

 The keyboard generates an INT 9 every time a key is pushed or

 released.

 **Notes:** This is a hardware interrupt (IRQ 1) activated by

 the make or break of every keystroke.

 The default INT 9 handler in the ROM reads the make and break scan

 codes from the keyboard and converts them into actions or key codes as

 follows:

 ş For ASCII keys, when a make code is encountered, the ASCII code

 and the scan code for the key are placed in the 32-byte keyboard

 buffer, which is located at 0:41Eh. The ASCII code and scan code

 are placed in the buffer at the location addressed by the

 Keyboard Buffer Tail Pointer (0:041Ch). The Keyboard Buffer Tail

 Pointer is then incremented by 2, and if it points past the end

 of the buffer, it is adjusted so that it points to the beginning

 of the buffer.

 ş If Ctrl, Alt, or Shift has been pressed, the Shift Status

 (0:0417h) and Extended Shift Status (0:0418h) bytes are updated.

 ş If the Ctrl-Alt-Del combination has been pressed, the Reset

 Flag (0:0472h) is set to 1234h and control is given to the

 power-on self test (POST). Because the Reset Flag is 1234h, the

 POST routine bypasses the memory test.

 ş If the Pause key sequence has been entered, this interrupt

 enters an indefinite loop. The loop is broken as soon as a valid

 ASCII keystroke is entered. (The PC Convertible issues an INT

 15h, Service 41h (Wait on External Event), to execute its pause

 loop.)

 ş If the Print Screen key sequence is entered, an INT 05h (Print

 Screen) is executed.

 ş If the Control-Break key sequence is entered, an INT 1Bh

 (Control-Break) is executed.

 ş For XTs dated 1/10/86 and after, ATs, XT-286s, and PC

 Convertibles, the INT 9h handler generates an INT 15h, function

 91h (Interrupt Complete) to signal that a keystroke is

 available. Also, on these machines, a make or break of the Sys

 Req key generates an INT 15h, function 85h (System Request Key

 Pressed).

 ş For ATs dated 6/10/85 and after, XT-286s, and PC Convertibles,

 an INT 15h, function 4Fh (Keyboard Intercept) is executed after

 the scan code has been read from the keyboard port (60h). This

 allows the user to redefine or remove a keystroke.

 INT 16 provides a standard way to read characters from the keyboard

 buffer that have been placed there by the INT 9 handler in ROM.

**INT 16h, 11h (17) Extended Keyboard Status many**

 Checks to see if a character is available in the buffer. This service

 is provided to support the extended keyboard (101/102-key keyboard).

 **On entry:** AH 11h

 **Returns:** Zero 0, if character is available

 1, if character is not available

 AX If character is available, then set to

 character as in Service 10h

 --------------------------------------------------------------------------

 **Notes:** This service is supported only on the AT dated

 11/15/85 and after, the XT dated 1/10/86 and after,

 and the XT-286.

 If a character is available, the Zero Flag is

 cleared and AX contains the ASCII value in AL and

 the scan code in AH. The character is not removed

 from the buffer. Use Service 10h to remove the

 character from the buffer. See Service 10h for a

 complete description of the meaning of AX if a

 character is available.

 This service is excellent for clearing the keyboard

 or allowing a program to be interruptable by a

 specific key sequence.

 See Service 01h for an equivalent service that works

 with all keyboards, not supporting the new keys on

 the enhanced (101/102-key) keyboard.

**Keyboard Codes**

 This tables lists all the keyboard codes that are returned by calls to

 INT 16 functions 0 through 2 (the second and third columns) and 10h

 through 12h (the last column).

 **83/84-Key[1] 101/102-Key[2] 101/102-Key**

 **Standard Standard Extended**

 **Keystroke Function Function[3] Function[4]**

 Esc 01/1B same same

 1 02/31 same same

 2 03/32 same same

 3 04/33 same same

 4 05/34 same same

 5 06/35 same same

 6 07/36 same same

 7 08/37 same same

 8 09/38 same same

 9 0A/39 same same

 0 0B/30 same same

 - 0C/2D same same

 = 0D/3D same same

 Backspace 0E/08 same same

 Tab 0F/09 same same

 q 10/71 same same

 w 11/77 same same

 e 12/65 same same

 r 13/72 same same

 t 14/74 same same

 y 15/79 same same

 u 16/75 same same

 i 17/69 same same

 o 18/6F same same

 p 19/70 same same

 [ 1A/5B same same

 ] 1B/5D same same

 Enter 1C/0D same same

 Ctrl \*\* \*\* \*\*

 a 1E/61 same same

 s 1F/73 same same

 d 20/64 same same

 f 21/66 same same

 g 22/67 same same

 h 23/68 same same

 j 24/6A same same

 k 25/6B same same

 l 26/6C same same

 ; 27/3B same same

 ' 28/27 same same

 ` 29/60 same same

 Shift \*\* \*\* \*\*

 \ 2B/5C same same

 z 2C/7A same same

 x 2D/78 same same

 c 2E/63 same same

 v 2F/76 same same

 b 30/62 same same

 n 31/6E same same

 m 32/6D same same

 , 33/2C same same

 . 34/2E same same

 / 35/2F same same

 Gray \* 37/2A same same

 Alt \*\* \*\* \*\*

 Space 39/20 same same

 Caps Lock \*\* \*\* \*\*

 F1 3B/00 same same

 F2 3C/00 same same

 F3 3D/00 same same

 F4 3E/00 same same

 F5 3F/00 same same

 F6 40/00 same same

 F7 41/00 same same

 F8 42/00 same same

 F9 43/00 same same

 F10 44/00 same same

 F11 no key -- 85/00

 F12 no key -- 86/00

 Num Lock \*\* \*\* \*\*

 Scroll Lock \*\* \*\* \*\*

 White Home 47/00 same same

 White Up Arrow 48/00 same same

 White PgUp 49/00 same same

 Gray - 4A/2D same same

 White Left Arrow 4B/00 same same

 Center Key -- -- 4C/00

 White Right Arrow 4D/00 same same

 Gray + 4E/2B same same

 White End 4F/00 same same

 White Down Arrow 50/00 same same

 White PgDn 51/00 same same

 White Ins 52/00 same same

 White Del 53/00 same same

 SysReq no key \*\* \*\*

 Key 45 [5] no key 56/5C same

 Enter (number keypad) no key 1C/0D E0/0D

 Gray / no key 35/2F E0/2F

 PrtSc no key \*\* \*\*

 Pause no key \*\* \*\*

 Gray Home no key 47/00 47/E0

 Gray Up Arrow no key 48/00 48/E0

 Gray Page Up no key 49/00 49/E0

 Gray Left Arrow no key 4B/00 4B/E0

 Gray Right Arrow no key 4D/00 4D/E0

 Gray End no key 4F/00 4F/E0

 Gray Down Arrow no key 50/00 50/E0

 Gray Page Down no key 51/00 51/E0

 Gray Insert no key 52/00 52/E0

 Gray Delete no key 53/00 53/E0

 Shift Esc 01/1B same same

 ! 02/21 same same

 @ 03/40 same same

 # 04/23 same same

 $ 05/24 same same

 % 06/25 same same

 ^ 07/5E same same

 & 08/26 same same

 \* (white) 09/2A same same

 ( 0A/28 same same

 ) 0B/29 same same

 \_ 0C/5F same same

 + (white) 0D/2B same same

 Shift Backspace 0E/08 same same

 Shift Tab (Backtab) 0F/00 same same

 Q 10/51 same same

 W 11/57 same same

 E 12/45 same same

 R 13/52 same same

 T 14/54 same same

 Y 15/59 same same

 U 16/55 same same

 I 17/49 same same

 O 18/4F same same

 P 19/50 same same

 { 1A/7B same same

 } 1B/7D same same

 Shift Enter 1C/0D same same

 Shift Ctrl \*\* \*\* \*\*

 A 1E/41 same same

 S 1F/53 same same

 D 20/44 same same

 F 21/46 same same

 G 22/47 same same

 H 23/48 same same

 J 24/4A same same

 K 25/4B same same

 L 26/4C same same

 : 27/3A same same

 " 28/22 same same

 ~ 29/7E same same

 | 2B/7C same same

 Z 2C/5A same same

 X 2D/58 same same

 C 2E/43 same same

 V 2F/56 same same

 B 30/42 same same

 N 31/4E same same

 M 32/4D same same

 < 33/3C same same

 > 34/3E same same

 ? 35/3F same same

 Shift Gray \* \*\* \*\* 37/2A

 Shift Alt \*\* \*\* \*\*

 Shift Space 39/20 same same

 Shift Caps Lock \*\* \*\* \*\*

 Shift F1 54/00 same same

 Shift F2 55/00 same same

 Shift F3 56/00 same same

 Shift F4 57/00 same same

 Shift F5 58/00 same same

 Shift F6 59/00 same same

 Shift F7 5A/00 same same

 Shift F8 5B/00 same same

 Shift F9 5C/00 same same

 Shift F10 5D/00 same same

 Shift F11 no key -- 87/00

 Shift F12 no key -- 88/00

 Shift Num Lock \*\* \*\* \*\*

 Shift Scroll Lock \*\* \*\* \*\*

 Shift 7 (number pad) 47/37 same same

 Shift 8 (number pad) 48/38 same same

 Shift 9 (number pad) 49/39 same same

 Shift Gray - 4A/2D same same

 Shift 4 (number pad) 4B/34 same same

 Shift 5 (number pad) 4C/35 same same

 Shift 6 (number pad) 4D/36 same same

 Shift Gray + 4E/2B same same

 Shift 1 (number pad) 4F/31 same same

 Shift 2 (number pad) 50/32 same same

 Shift 3 (number pad) 51/33 same same

 Shift 0 (number pad) 52/30 same same

 Shift . (number pad) 53/2E same same

 Shift SysReq no key \*\* \*\*

 Shift Key 45 [5] no key 56/7C same

 Shift Enter (number pad) no key 1C/0D E0/0D

 Shift Gray / no key 35/2F E0/2F

 Shift PrtSc no key \*\* \*\*

 Shift Pause no key \*\* \*\*

 Shift Gray Home no key 47/00 47/E0

 Shift Gray Up Arrow no key 48/00 48/E0

 Shift Gray Page Up no key 49/00 49/E0

 Shift Gray Left Arrow no key 4B/00 4B/E0

 Shift Gray Right Arrow no key 4D/00 4D/E0

 Shift Gray End no key 4F/00 4F/E0

 Shift Gray Down Arrow no key 50/00 50/E0

 Shift Gray Page Down no key 51/00 51/E0

 Shift Gray Insert no key 52/00 52/E0

 Shift Gray Delete no key 53/00 53/E0

 Ctrl Esc 01/1B same same

 Ctrl 1 -- -- --

 Ctrl 2 (NUL) 03/00 same same

 Ctrl 3 -- -- --

 Ctrl 4 -- -- --

 Ctrl 5 -- -- --

 Ctrl 6 (RS) 07/1E same same

 Ctrl 7 -- -- --

 Ctrl 8 -- -- --

 Ctrl 9 -- -- --

 Ctrl 0 -- -- --

 Ctrl - 0C/1F same same

 Ctrl = -- -- --

 Ctrl Backspace (DEL) 0E/7F same same

 Ctrl Tab -- -- 94/00

 Ctrl q (DC1) 10/11 same same

 Ctrl w (ETB) 11/17 same same

 Ctrl e (ENQ) 12/05 same same

 Ctrl r (DC2) 13/12 same same

 Ctrl t (DC4) 14/14 same same

 Ctrl y (EM) 15/19 same same

 Ctrl u (NAK) 16/15 same same

 Ctrl i (HT) 17/09 same same

 Ctrl o (SI) 18/0F same same

 Ctrl p (DEL) 19/10 same same

 Ctrl [ (ESC) 1A/1B same same

 Ctrl ] (GS) 1B/1D same same

 Ctrl Enter (LF) 1C/0A same same

 Ctrl a (SOH) 1E/01 same same

 Ctrl s (DC3) 1F/13 same same

 Ctrl d (EOT) 20/04 same same

 Ctrl f (ACK) 21/06 same same

 Ctrl g (BEL) 22/07 same same

 Ctrl h (Backspace) 23/08 same same

 Ctrl j (LF) 24/0A same same

 Ctrl k (VT) 25/0B same same

 Ctrl l (FF) 26/0C same same

 Ctrl ; -- -- --

 Ctrl ' -- -- --

 Ctrl ` -- -- --

 Ctrl Shift \*\* \*\* \*\*

 Ctrl \ (FS) 2B/1C same same

 Ctrl z (SUB) 2C/1A same same

 Ctrl x (CAN) 2D/18 same same

 Ctrl c (ETX) 2E/03 same same

 Ctrl v (SYN) 2F/16 same same

 Ctrl b (STX) 30/02 same same

 Ctrl n (SO) 31/0E same same

 Ctrl m (CR) 32/0D same same

 Ctrl , -- -- --

 Ctrl . -- -- --

 Ctrl / -- -- --

 Ctrl Gray \* -- -- 96/00

 Ctrl Alt \*\* \*\* \*\*

 Ctrl Space 39/20 same same

 Ctrl Caps Lock -- -- --

 Ctrl F1 5E/00 same same

 Ctrl F2 5F/00 same same

 Ctrl F3 60/00 same same

 Ctrl F4 61/00 same same

 Ctrl F5 62/00 same same

 Ctrl F6 63/00 same same

 Ctrl F7 64/00 same same

 Ctrl F8 65/00 same same

 Ctrl F9 66/00 same same

 Ctrl F10 67/00 same same

 Ctrl F11 no key -- 89/00

 Ctrl F12 no key -- 8A/00

 Ctrl Num Lock -- -- --

 Ctrl Scroll Lock -- -- --

 Ctrl White Home 77/00 same same

 Ctrl White Up Arrow -- -- 8D/00

 Ctrl White PgUp 84/00 same same

 Ctrl Gray - -- -- 8E/00

 Ctrl White Left Arrow 73/00 same same

 Ctrl 5 (number pad) -- -- 8F/00

 Ctrl White Right Arrow 74/00 same same

 Ctrl Gray + -- -- 90/00

 Ctrl White End 75/00 75/00 same

 Ctrl White Down Arrow -- -- 91/00

 Ctrl White PgDn 76/00 same same

 Ctrl White Ins -- -- 92/00

 Ctrl White Del -- -- 93/00

 Ctrl SysReq no key \*\* \*\*

 Ctrl Key 45 [5] no key -- --

 Ctrl Enter (number pad) no key 1C/0A E0/0A

 Ctrl / (number pad) no key -- 95/00

 Ctrl PrtSc no key 72/00 same

 Ctrl Break no key 00/00 same

 Ctrl Gray Home no key 77/00 77/E0

 Ctrl Gray Up Arrow no key -- 8D/E0

 Ctrl Gray Page Up no key 84/00 84/E0

 Ctrl Gray Left Arrow no key 73/00 73/E0

 Ctrl Gray Right Arrow no key 74/00 74/E0

 Ctrl Gray End no key 75/00 75/E0

 Ctrl Gray Down Arrow no key -- 91/E0

 Ctrl Gray Page Down no key 76/00 76/E0

 Ctrl Gray Insert no key -- 92/E0

 Ctrl Gray Delete no key -- 93/E0

 Alt Esc -- -- 01/00

 Alt 1 78/00 same same

 Alt 2 79/00 same same

 Alt 3 7A/00 same same

 Alt 4 7B/00 same same

 Alt 5 7C/00 same same

 Alt 6 7D/00 same same

 Alt 7 7E/00 same same

 Alt 8 7F/00 same same

 Alt 9 80/00 same same

 Alt 0 81/00 same same

 Alt - 82/00 same same

 Alt = 83/00 same same

 Alt Backspace -- -- 0E/00

 Alt Tab -- -- A5/00

 Alt q 10/00 same same

 Alt w 11/00 same same

 Alt e 12/00 same same

 Alt r 13/00 same same

 Alt t 14/00 same same

 Alt y 15/00 same same

 Alt u 16/00 same same

 Alt i 17/00 same same

 Alt o 18/00 same same

 Alt p 19/00 same same

 Alt [ -- -- 1A/00

 Alt ] -- -- 1B/00

 Alt Enter -- -- 1C/00

 Alt Ctrl \*\* \*\* \*\*

 Alt a 1E/00 same same

 Alt s 1F/00 same same

 Alt d 20/00 same same

 Alt f 21/00 same same

 Alt g 22/00 same same

 Alt h 23/00 same same

 Alt j 24/00 same same

 Alt k 25/00 same same

 Alt l 26/00 same same

 Alt ; -- -- 27/00

 Alt ' -- -- 28/00

 Alt ` -- -- 29/00

 Alt Shift \*\* \*\* \*\*

 Alt \ -- -- 2B/00

 Alt z 2C/00 same same

 Alt x 2D/00 same same

 Alt c 2E/00 same same

 Alt v 2F/00 same same

 Alt b 30/00 same same

 Alt n 31/00 same same

 Alt m 32/00 same same

 Alt , -- -- 33/00

 Alt . -- -- 34/00

 Alt / -- -- 35/00

 Alt Gray \* -- -- 37/00

 Alt Space 39/20 same same

 Alt Caps Lock \*\* \*\* \*\*

 Alt F1 68/00 same same

 Alt F2 69/00 same same

 Alt F3 6A/00 same same

 Alt F4 6B/00 same same

 Alt F5 6C/00 same same

 Alt F6 6D/00 same same

 Alt F7 6E/00 same same

 Alt F8 6F/00 same same

 Alt F9 70/00 same same

 Alt F10 71/00 same same

 Alt F11 no key -- 8B/00

 Alt F12 no key -- 8C/00

 Alt Num Lock \*\* \*\* \*\*

 Alt Scroll Lock \*\* \*\* \*\*

 Alt Gray - -- -- 4A/00

 Alt Gray + -- -- 4E/00

 Alt 7 (number pad) # # #

 Alt 8 (number pad) # # #

 Alt 9 (number pad) # # #

 Alt 4 (number pad) # # #

 Alt 5 (number pad) # # #

 Alt 6 (number pad) # # #

 Alt 1 (number pad) # # #

 Alt 2 (number pad) # # #

 Alt 3 (number pad) # # #

 Alt Del -- -- --

 Alt SysReq no key \*\* \*\*

 Alt Key 45 [5] no key -- --

 Alt Enter (number pad) no key -- A6/00

 Alt / (number pad) -- -- A4/00

 Alt PrtSc no key \*\* \*\*

 Alt Pause no key \*\* \*\*

 Alt Gray Home no key -- 97/00

 Alt Gray Up Arrow no key -- 98/00

 Alt Gray Page Up no key -- 99/00

 Alt Gray Left Arrow no key -- 9B/00

 Alt Gray Right Arrow no key -- 9D/00

 Alt Gray End no key -- 9F/00

 Alt Gray Down Arrow no key -- A0/00

 Alt Gray Page Down no key -- A1/00

 Alt Gray Insert no key -- A2/00

 Alt Gray Delete no key -- A3/00

 --------------------------------------------------------------------------

**Footnotes**

 [1] The 83-key and 84-key keyboards are the original PC keyboard

 and the original-layout AT keyboard, respectively.

 [2] The 101/102-key keyboard is the current IBM standard

 ("Enhanced") keyboard.

 [3] "Standard Function" refers to Interrupt 16h Services 0, 1,

 and 2.

 [4] "Extended Function" refers to Interrupt 16h Services 10h,

 11h, and 12h.

 [5] In the United States, the 101/102-key keyboard is shipped

 with 101 keys. Overseas versions have an additional key

 sandwiched between the left Shift key and the Z key. This

 additional key is identified by IBM (and in this table) as

 "Key 45."

 [\*\*] Keys and key combinations marked \*\* are used by the ROM BIOS

 but do not put values into the keyboard buffer.

 [--] Keys and key combinations marked -- are ignored by the ROM

 BIOS.

**INT 16h, 10h (16) Extended Keyboard Read many**

 Returns the next character in the keyboard buffer; if no character is

 available, this service waits until one is available. This service is

 provided to support the enhanced (101/102-key) keyboard.

 **On entry:** AH 10h

 **Returns:** AL ASCII character code

 AH Scan code

 --------------------------------------------------------------------------

 **Notes:** This service is supported only on the AT dated

 11/15/85 and after, PC XT dated 1/10/86 and after,

 and XT-286.

 The scan codes are the numbers representing the

 location of the key on the keyboard. As new keys

 have been added and the keyboard layout rearranged,

 this numbering scheme has not been consistent with

 its original purpose. See the list of scan codes.

 If the character is a special character, AL will be

 0 and the value in AH will be the extended scan code

 for the key. See the list of scan codes.

 Use the scan codes to differentiate between keys

 representing the same ASCII code, such as the plus

 key across the top of the keyboard and the gray plus

 key.

 After the character has been removed from the

 keyboard buffer, the keyboard buffer start pointer

 (at 0:041Ah) is increased by 2. If the start pointer

 is beyond the end of the buffer, the start pointer

 is reset to the start of the keyboard buffer.

 If no character is available at the keyboard, an INT

 15h, Service 90h (Device Busy), will be issued for

 the keyboard, informing the operating system that

 there is a keyboard loop taking place and thereby

 allowing the operating system to perform another

 task.

 After every character is typed, an INT 15h, Service

 91h (Interrupt Complete), is issued. This allows the

 operating system to switch back to a task that is

 waiting for a character at the keyboard.

 See Service 00h for an equivalent service that works

 with all keyboards, not supporting the new keys on

 the enhanced (101/102-key) keyboard.

**INT 16h, 05h (5) Keyboard Write many**

 Puts a scan code/character code combination in the keyboard buffer.

 **On entry:** AH 05h

 CH Scan code

 CL ASCII character code

 **Returns:** AL 00h - Operation successful

 01h - Buffer full

 --------------------------------------------------------------------------

 **Notes:** This service is available only for ATs dated

 11/15/85 and after, and XT 286s.

 This service "fools" a program into thinking that

 keys have come from the keyboard. A good example of

 this would be the operation needed by a program that

 implements keyboard macros.

 For a description and list of scan codes, see "Scan

 Codes."

**INT 16h, 12h (18) Get Extended Shift Status many**

 Returns the current keyboard shift status. This service is provided to

 support the enhanced (101/102-key) keyboard.

 **On entry:** AH 12h

 **Returns:** AL Shift status

 AH Extended shift status

 --------------------------------------------------------------------------

 **Shift Status (AL)**

 7 6 5 4 3 2 1 0

 1 . . . . . . . Insert locked

 . 1 . . . . . . Caps Lock locked

 . . 1 . . . . . Num Lock locked

 . . . 1 . . . . Scroll Lock locked

 . . . . 1 . . . Alt key is pressed

 . . . . . 1 . . Ctrl key is pressed

 . . . . . . 1 . Left Shift key is pressed

 . . . . . . . 1 Right Shift key is pressed

 **Extended Shift Status (AH)**

 7 6 5 4 3 2 1 0

 1 . . . . . . . Sys Req key is pressed

 . 1 . . . . . . Caps Lock key is pressed

 . . 1 . . . . . Num Lock key is pressed

 . . . 1 . . . . Scroll Lock key is pressed

 . . . . 1 . . . Right Alt key is pressed

 . . . . . 1 . . Right Ctrl key is is pressed

 . . . . . . 1 . Left Alt key is pressed

 . . . . . . . 1 Left Ctrl key is pressed

 **Notes:** This service is supported only on the AT dated

 11/15/85 and after, the XT dated 1/10/86 and after,

 and the XT-286.

 The shift status byte return in AL is obtained from

 the byte at memory location 0:0417h. Note that this

 is the same value that is returned by Service 02h.

 Note that the Extended Shift Status (value returned

 in AH) reports if keys are being pressed, not

 locked.

 See Service 02h for an equivalent service that works

 with all keyboards, not supporting the new keys on

 the enhanced (101/102-key) keyboard.

**INT 1Ah, 00h (0) Read System-Timer Time Counter all**

 Reports the current time of day, and whether 24 hours has passed since

 1) the last power-on, 2) the last system reset, or 3) the last system-

 timer time read or set.

 **On entry:** AH 00h

 **Returns:** CX High-order part of clock count

 DX Low-order part of clock count

 AL 0 if 24 hours has not passed; else 1

 --------------------------------------------------------------------------

 **Notes:** The following formulas convert the clock count to

 the time of day:

 Hour = Clock / 65543 (1007h)

 Remainder = Clock MOD 65543

 Minutes = Remainder / 1092 (444h)

 Remainder = Remainder MOD 1092

 Second = Remainder / 18.21

 Remainder = Remainder MOD 18.21

 Hundredths = CINT(Remainder \* 100)

 The "system timer" (as distinguished from the real-

 time clock) is the timer that's set when the system

 is started. This time is temporary, lasting only as

 long as the system is turned on.

 The clock count may also be read as a 4-byte integer

 at memory location 0:046C. This 4-byte value is

 equal to the 4-byte integer in CX:DX after Service

 00h has been called.

 After the call, the flag (at 0:0470h) stating

 whether 24 hours has passed or not, is cleared.

 When TIME is typed at the command line, DOS gets the

 time by means of this service.

 Counts occur at the rate of 18.2 per second.

**INT 1Ah, 01h (1) Set System-Timer Time Counter all**

 Sets the current time of day.

 **On entry:** AH 01h

 CX High-order part of clock count

 DX Low-order part of clock count

 **Returns:** None

 --------------------------------------------------------------------------

 **Notes:** The following formula converts the time of day to a

 clock count:

 Count = (Hout \* 65543.33) + (Minutes \* 1092.38) +

 (Seconds \* 18.21) + (Hundreths \* .182)

 The "system timer" (as distinguished from the real-

 time clock) is the timer that's set when the system

 is started. This time is temporary, lasting only as

 long as the system is turned on.

 The clock count may also be set as a 4-byte integer

 at memory location 0:046C. This 4-byte value will be

 set to the 4-byte integer in CX:DX after the call.

 After the set, the flag (at 0:0470h) stating whether

 24 hours has passed or not, is cleared.

 When TIME is typed at the command line, DOS gets the

 time by means of this service. Setting a new time

 will call this service.

 Counts occur at the rate of 18.2 per second.

**INT 1Ah, 02h (2) Read Real-Time Clock Time many**

 Reads the time from the computer's real-time clock.

 **On entry:** AH 02h

 **Returns:** CF Set if clock not operating; else cleared

 CH Hours (BCD)

 CL Minutes (BCD)

 DH Seconds (BCD)

 DL 1 if daylight saving time option; else 0

 --------------------------------------------------------------------------

 **Notes:** All times are in Binary Coded Decimal (BCD). For

 example, 9:04.12 am will be reported as CX = 0904;

 DH = 12.

 The real-time clock is the clock that runs even when

 the computer is turned off. A CMOS battery is used

 so that, even while the computer is off, the date,

 time, and alarm time are maintained.

 This service is available only for ATs, XT-286s, and

 PC Convertibles. Previous machines will report

 unpredictable results (including the state of the

 Carry Flag).

 The daylight saving time option is not reported on

 ATs with dates prior to 6/10/85.

 The information returned from this service could

 differ from the system time. This is because the

 real-time clock is read once upon system startup,

 updating the system timer. All other references to

 time use the system timer. If either the real-time

 clock is changed (via Service 03h) or the system

 timer is changed (via Service 01h), the other clock

 is not changed accordingly.

 Even though the system uses the system timer for

 timing information, both the real-time clock and the

 system timer are updated continuously.

**INT 1Ah, 03h (3) Set Real-Time Clock Time many**

 Sets the time for the computer's real-time clock.

 **On entry:** AH 03h

 CH Hours (BCD)

 CL Minutes (BCD)

 DH Seconds (BCD)

 DL 1 if daylight saving time option; else 0

 **Returns:** None

 --------------------------------------------------------------------------

 **Notes:** All times are in Binary Coded Decimal (BCD). For

 example, 9:04.12 am would be set as CX = 0904; DH =

 12.

 The real-time clock is the clock that runs even when

 the computer is turned off. A CMOS battery is used

 so that, even while the computer is off, the date,

 time, and alarm time are maintained.

 This service is available only for ATs, XT-286s, and

 PC Convertibles. Previous machines will report

 unpredictable results (including the state of the

 Carry Flag).

 Setting the real-time clock will not affect the

 system time. This is because the real-time clock is

 read once upon system startup, updating the system

 timer. All other references to time use the system

 timer. If either the real-time clock is changed (via

 this service) or the system timer is changed (via

 Service 01h), the other clock is not changed

 accordingly.

 Even though the system uses the system-timer for

 timing information, both the real-time clock and the

 system timer are updated continuously.

**INT 1Ah, 04h (4) Read Real-Time Clock Date many**

 Reads the date from the computer's real-time clock.

 **On entry:** AH 04h

 **Returns:** CF Set if clock not operating; else cleared

 CH Century (19 or 20) (BCD)

 CL Year (BCD)

 DH Month (BCD)

 DL Day (BCD)

 --------------------------------------------------------------------------

 **Notes:** All dates are in Binary Coded Decimal (BCD). For

 example, November 15, 1987 will be reported as CX =

 1987; DX = 1115.

 The real-time clock is the clock that runs even when

 the computer is turned off. A CMOS battery is used

 so that, even while the computer is off, the date,

 time, and alarm time are maintained.

 This service is available only for ATs, XT-286s, and

 PC Convertibles. Previous machines will report

 unpredictable results (including the state of the

 Carry Flag).

 The information returned from this service could

 differ from the system date. This is because the

 real-time clock is read once upon system startup,

 updating the system timer. All other references to

 date use the system timer. If either the real-time

 clock date is changed (via Service 05h) or the

 system timer is changed (via Service 01h), the other

 date is not changed accordingly.

 Even though the system uses the system timer for

 date information, both the real-time clock and the

 system timer are updated continuously.

**INT 1Ah, 05h (5) Set Real-Time Clock Date many**

 Sets the date on the computer's real-time clock.

 **On entry:** AH 05h

 CH Century (19 or 20) (BCD)

 CL Year (BCD)

 DH Month (BCD)

 DL Day (BCD)

 --------------------------------------------------------------------------

 **Notes:** All dates are in Binary Coded Decimal (BCD). For

 example, January 15, 1987 will be set as CX = 1987;

 DX = 0115.

 The real-time clock is the clock that runs even when

 the computer is turned off. A CMOS battery is used

 so that, even while the computer is off, the date,

 time, and alarm time are maintained.

 This service is available only for ATs, XT-286s, and

 PC Convertibles. Previous machines will report

 unpredictable results (including the state of the

 Carry Flag).

 Setting the real-time clock date will not affect the

 system date. This is because the real-time clock is

 read once upon system startup, updating the system

 timer. All other references to date use the system

 timer. If either the real-time clock date is changed

 (via this service) or the system timer is changed

 (via Service 01h), the other date is not changed

 accordingly.

 Even though the system uses the system timer for

 date information, both the real-time clock and the

 system timer are updated continuously.

**INT 08h (8) System Timer**

 This interrupt is a hardware interrupt (IRQ 0) activated by the system

 timer 18.2 times per second (every 55 ms). The default handler does

 the following:

 ş Keeps a count of the number of timer ticks at memory location

 0:46Ch (System Timer Counter--long integer). After 24 hours of

 operation, a flag is set at memory location 0:470h to signal

 this condition and the System Timer Counter (0:46Ch) is reset to

 0.

 ş Decrements the Diskette Drive Motor Off Counter at memory

 location 0:440h if it is not 0. When this location reaches 0,

 the diskette drive motor is turned off and the Diskette Drive

 Motor Status byte at memory location 0:43Fh is updated to

 reflect that the motor has been turned off.

 ş Generates INT 1Ch.

 ş For the PC Convertible, generates INT 4Ah if an alarm interrupt

 occurs.

**INT 1Ch (28) User Timer Tick**

 Called at every timer tick (18.2 times per second, or every 55 ms) by

 INT 08h (System Timer).

 --------------------------------------------------------------------------

 By default, the interrupt handler for this interrupt IRETs back to the

 caller. The user can supply his own handler so he can attain control

 at every timer tick.

 **Notes:** Note that this interrupt is called by the hardware

 timer interrupt (INT 08h), which has not signaled

 the end of the interrupt back to the interrupt

 controller. Therefore, this routine is limited in

 the actions it can take, since all interrupts will

 be disabled. A better approach to this problem is to

 intercept INT 08h. The new INT 08h handler will

 first call the standard INT 08h, which will handle

 the interrupt controller completion signals

 mentioned above. Then after the standard INT 08h is

 completed, the new handler can do whatever it has to

 do at every timer tick.

**INT 70h (112) Real-Time Clock XT-286, AT, Convertible**

 Handles the alarm and periodic interrupts for the real-time clock.

 --------------------------------------------------------------------------

 This is a hardware interrupt (IRQ 8) activated by the CMOS timer 1024

 times per second.

 **Notes:** This interrupt is available only on the AT and XT-

 286. The real-time clock on the PC Convertible

 generates an INT 02h (Non-Maskable Interrupt) to

 handle the periodic and alarm functions.

 This interrupt provides two services. The first is

 the periodic interrupt that is generated when a time

 delay is specified. See INT 15h, Service 83h (Event

 Wait), and Service 86h (Wait), for more information.

 The second service is the Alarm Interrupt. When the

 CMOS alarm time matches the current CMOS time, then

 INT 4Ah (User Alarm) is executed.

**INT 1Bh (27) Keyboard Break**

 Called when a Control-Break key sequence is encountered.

 --------------------------------------------------------------------------

 When the machine is powered on, the interrupt handler for this

 interrupt just IRETs back to the caller. DOS, however, installs a

 Ctrl-Break handler that sets an internal flag to signal that a

 keyboard Break condition has occurred.

 **Notes:** DOS tests the state of the internal flag

 occasionally, depending on the state of the Control-

 Break Check (INT 21h, Function 33h). If the flag is

 on during a test, then the string "^C", followed by

 a Carriage-Return/ Line-Feed combination, is

 displayed on the screen, the DOS internal flag is

 reset, and INT 23h (Control-Break Exit Address) is

 called.

 Note that INT 09h (Keyboard) sets the byte at

 0:0471h to signal that a Control-Break key has been

 entered.

**INT 15h, 85h (133) System Request Pressed many**

 Provides an interface for special processing of the Sys Req key.

 **On entry:** AH 85h

 AL Subservice

 00h - System Request key Make (pressed)

 01h - System Request key Break (released)

 **Returns:** Carry Set if service not supported, else cleared

 --------------------------------------------------------------------------

 By default, this service does nothing but clear the Carry Flag. It is

 up to the operating system or user to provide a handler to process the

 Sys Req key. This service is called by INT 09h (Keyboard) after the

 Sys Req key has been pressed ("made") or released ("broken").

 **Notes:** The PC, PCjr, and the XT dated 11/08/82 do not

 support this service.

 Currently there is no standard use for the System

 Request key. Future operating systems will probably

 use it to switch between tasks.

**INT 15h, 83h (131) Event Wait many**

 Modifies a byte in the user's memory after a specified time interval.

 Control is transferred to the caller immediately after this service is

 called.

 **On entry:** AH 83h

 AL Subservice

 00h - Set interval

 01h - Cancel set interval

 ES:BX - Pointer to byte that will be posted

 CX:DX - Microseconds until posting (long integer)

 **Returns:** Carry Set if service not supported, else cleared

 --------------------------------------------------------------------------

 After this service is called, control is returned immediately to the

 caller. This service does not work by delaying and then returning

 control to the user, but returns control immediately; the user must

 periodically look at the byte specified (pointed to by ES:BX) to

 determine when the interval has elapsed.

 **Notes:** The PC, PCjr, and the XT dated 11/08/82 do not

 support this service. The AT dated 1/10/84 is only

 able to set the interval; thus the value in AL is

 not used.

 The byte is posted by setting the high bit (bit 7).

 To test the end of the interval, make sure the high

 bit of the byte is clear before calling Subservice 0

 (Set Interval).

 The real-time clock is used to count the number of

 microseconds.

 The granularity of the time is 976 microseconds.

 CX is the high-order word of the count. For example,

 if CX is 98h and DX is 9680h, then a 10-second delay

 would be specified.

**INT 15h, 91h (139) Interrupt Complete many**

 Signals the system that a device is ready to be serviced.

 **On entry:** AH 91h

 AL Device type (See service 90h)

 **Returns:** Carry Set if error, else cleared

 --------------------------------------------------------------------------

 **Notes:** This service is not available for the PC, PCjr, and

 XT dated 11/08/82.

 When a device is needed by a process, but access to

 it is delayed, the system can perform another task

 to achieve maximum performance until the device is

 freed up for the process to use. In a multi-tasking

 environment, that is the purpose for this service.

 The operating system can keep track of which process

 is waiting for which device, and perform another

 task while the device is busy. After a device is

 ready to be serviced, then this function is called

 to notify the system that a device is ready.

 For example, INT 09h will call this service after a

 key has been entered, to signal the operating system

 that a key has been entered. Thus if the system was

 performing another task while a previous process was

 waiting for a keyboard character, then the previous

 process could regain control of the system and get

 the keyboard character just entered.

**INT 14h, 00h (00) Initialize Serial Port Parameters**

 Initializes the baud rate, parity, stop-bit, and word length

 parameters for a serial port, and returns the status for the port.

 **On entry:** AH 00h

 AL Communications parameters (see below)

 DX Serial port number (0 - COM1, 1 - COM2, etc.)

 **Returns:** AX Line and modem status (see Service 03h)

 --------------------------------------------------------------------------

 **Baud rate Parity Stop bits Word Length**

 **Bit:** 7 6 5 4 3 2 1 0

 0 0 0 110 0 0 None 0 One 1 0 7 bits

 0 0 1 150 0 1 Odd 1 Two 1 1 8 bits

 0 1 0 300 1 0 None

 0 1 1 600 1 1 Even

 1 0 0 1,200

 1 0 1 2,400

 1 1 0 4,800

 1 1 1 9,600

**INT 14h, 03h (3) Get Serial Port Status**

 Returns line status and modem status information for a specified

 serial port.

 **On entry:** AH 03h

 DX Serial port number (0 - COM1, 1 - COM2, etc.)

 **Returns:** AX Line and Modem status (See below)

 --------------------------------------------------------------------------

 Status information is returned in AX, as follows:

  **AH (Line status)**

 7 6 5 4 3 2 1 0

 1 . . . . . . . Time-out error

 . 1 . . . . . . Transfer shift register empty

 . . 1 . . . . . Transfer holding register empty

 . . . 1 . . . . Break-detect error

 . . . . 1 . . . Framing error

 . . . . . 1 . . Parity error

 . . . . . . 1 . Overrun error

 . . . . . . . 1 Data ready

  **AL (Modem status)**

 7 6 5 4 3 2 1 0

 1 . . . . . . . Received line signal detect

 . 1 . . . . . . Ring indicator

 . . 1 . . . . . Data set ready

 . . . 1 . . . . Clear to send

 . . . . 1 . . . Change in receive line signal detected

 . . . . . 1 . . Trailing edge ring detector

 . . . . . . 1 . Change in data set ready

 . . . . . . . 1 Change in clear to send

 If bit 7 of the Line status byte (AH) is set, then the rest of the

 bits in AH and AL are unpredictable.

 **Notes:** Early versions of the ROM-BIOS for the original PC

 had a programming error that would cause "time-out"

 errors to be reported as "transfer shift register

 empty" and "break-detect" errors. This has been

 corrected in all other versions of the ROM-BIOS.

**INT 14h, 01h (1) Send One Character**

 Sends one character to the specified serial port.

 **On entry:** AH 01h

 AL Character

 DX Serial port number (0 - COM1, 1 - COM2, etc.)

 **Returns:** AH Line status (see Service 03h)

 --------------------------------------------------------------------------

 If an error occurs, bit 7 of AH will be set. SInce bit 7 is used as a

 general error flag, this service is unable to identify a time-out

 error. For complete diagnostic information, use Service 03h (Get

 serial port status).

**INT 14h, 02h (2) Receive One Character**

 Receives one character at the specified serial port.

 **On entry:** AH 02h

 DX Serial port number (0 - COM1, 1 - COM2, etc.)

 **Returns:** AL Character

 AH Line status (See service 03h)

 --------------------------------------------------------------------------

 **Notes:** This service waits for a character. If no character

 is available or an error occurs, no character is

 returned and bit 7 of AH is set.

 If an error occurs, bit 7 of AH is set. SInce bit 7

 is used as a general error flag, this service is

 unable to identify a time-out error. For complete

 diagnostic information, use Service 03h (Get serial

 port status).