**DOS INT 21h - DOS Function Codes**

The follow abridged list of DOS interrupts has been extracted from a large list compiled by Ralf Brown. These are available on any Simtel mirror (e.g. [sunsite.anu.edu.au](http://sunsite.anu.edu.au/archives)) under the directory ms-dos/info/interNNp.zip

|  |  |  |  |
| --- | --- | --- | --- |
| **AH** | **Description** | **AH** | **Description** |
| 01 | [Read character from STDIN](http://spike.scu.edu.au/~barry/interrupts.html#ah01) | 02 | [Write character to STDOUT](http://spike.scu.edu.au/~barry/interrupts.html#ah02) |
| 05 | [Write character to printer](http://spike.scu.edu.au/~barry/interrupts.html#ah05) | 06 | [Console Input/Output](http://spike.scu.edu.au/~barry/interrupts.html#ah06) |
| 07 | [Direct char read (STDIN), no echo](http://spike.scu.edu.au/~barry/interrupts.html#ah07) | 08 | [Char read from STDIN, no echo](http://spike.scu.edu.au/~barry/interrupts.html#ah08) |
| 09 | [Write string to STDOUT](http://spike.scu.edu.au/~barry/interrupts.html#ah09) | 0A | [Buffered input](http://spike.scu.edu.au/~barry/interrupts.html#ah0a) |
| 0B | [Get STDIN status](http://spike.scu.edu.au/~barry/interrupts.html#ah0b) | 0C | [Flush buffer for STDIN](http://spike.scu.edu.au/~barry/interrupts.html#ah0c) |
| 0D | [Disk reset](http://spike.scu.edu.au/~barry/interrupts.html#ah0d) | 0E | [Select default drive](http://spike.scu.edu.au/~barry/interrupts.html#ah0e) |
| 19 | [Get current default drive](http://spike.scu.edu.au/~barry/interrupts.html#ah19h) | 25 | [Set interrupt vector](http://spike.scu.edu.au/~barry/interrupts.html#ah25) |
| 2A | [Get system date](http://spike.scu.edu.au/~barry/interrupts.html#ah2a) | 2B | [Set system date](http://spike.scu.edu.au/~barry/interrupts.html#ah2b) |
| 2C | [Get system time](http://spike.scu.edu.au/~barry/interrupts.html#ah2c) | 2D | [Set system time](http://spike.scu.edu.au/~barry/interrupts.html#ah2D) |
| 2E | [Set verify flag](http://spike.scu.edu.au/~barry/interrupts.html#ah2e) | 30 | [Get DOS version](http://spike.scu.edu.au/~barry/interrupts.html#ah30) |
| 35 | [Get Interrupt vector](http://spike.scu.edu.au/~barry/interrupts.html#ah35) |  |  |
| 36 | [Get free disk space](http://spike.scu.edu.au/~barry/interrupts.html#ah36) | 39 | [Create subdirectory](http://spike.scu.edu.au/~barry/interrupts.html#ah39) |
| 3A | [Remove subdirectory](http://spike.scu.edu.au/~barry/interrupts.html#ah3a) | 3B | [Set working directory](http://spike.scu.edu.au/~barry/interrupts.html#ah3b) |
| 3C | [Create file](http://spike.scu.edu.au/~barry/interrupts.html#ah3c) | 3D | [Open file](http://spike.scu.edu.au/~barry/interrupts.html#ah3d) |
| 3E | [Close file](http://spike.scu.edu.au/~barry/interrupts.html#ah3e) | 3F | [Read file](http://spike.scu.edu.au/~barry/interrupts.html#ah3f) |
| 40 | [Write file](http://spike.scu.edu.au/~barry/interrupts.html#ah40) | 41 | [Delete file](http://spike.scu.edu.au/~barry/interrupts.html#ah41) |
| 42 | [Seek file](http://spike.scu.edu.au/~barry/interrupts.html#ah42) | 43 | [Get/Set file attributes](http://spike.scu.edu.au/~barry/interrupts.html#ah43) |
| 47 | [Get current directory](http://spike.scu.edu.au/~barry/interrupts.html#ah47) | 4C | [Exit program](http://spike.scu.edu.au/~barry/interrupts.html#ah4c) |
| 4D | [Get return code](http://spike.scu.edu.au/~barry/interrupts.html#ah4d) | 54 | [Get verify flag](http://spike.scu.edu.au/~barry/interrupts.html#ah54) |
| 56 | [Rename file](http://spike.scu.edu.au/~barry/interrupts.html#ah56) | 57 | [Get/Set file date](http://spike.scu.edu.au/~barry/interrupts.html#ah57) |

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 01h - READ CHARACTER FROM STANDARD INPUT, WITH ECHO**

Return: AL = character read

Notes:

* ^C/^Break are checked
* ^P toggles the DOS-internal echo-to-printer flag
* ^Z is not interpreted, thus not causing an EOF if input is redirected character is echoed to standard output

SeeAlso: AH=06h,AH=07h,AH=08h,AH=0Ah

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 02h -WRITE CHARACTER TO STANDARD OUTPUT**

Entry: DL = character to write

Return: AL = last character output

Notes:

* ^C/^Break are checked
* the last character output will be the character in DL unless DL=09h on entry, in which case AL=20h as tabs are expanded to blanks
* if standard output is redirected to a file, no error checks (write- protected, full media, etc.) are performed

SeeAlso: AH=06h,AH=09h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 05h - WRITE CHARACTER TO PRINTER**

Entry: DL = character to print

Notes:

* keyboard checked for ^C/^Break
* STDPRN is usually the first parallel port, but may be redirected under DOS 2+
* if the printer is busy, this function will wait

SeeAlso: INT 17/AH=00h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 06h - DIRECT CONSOLE OUTPUT**

Entry: DL = character (except FFh)

Return: AL = character output

Notes: does not check ^C/^Break

SeeAlso: AH=02h,AH=09h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 06h - DIRECT CONSOLE INPUT**

Entry: AH = 06h DL = FFh

Return:

* ZF *set* if no character available and AL = 00h
* ZF *clear* if character available AL = character read

Notes:

* ^C/^Break are NOT checked
* if the returned character is 00h, the user pressed a key with an extended keycode, which will be returned by the next call of this function
* although the return of AL=00h when no characters are available is not documented, some programs rely on this behavior

SeeAlso: AH=0Bh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH=07h - DIRECT CHARACTER INPUT, WITHOUT ECHO**

Return: AL = character read from standard input

Notes: does not check ^C/^Break

SeeAlso: AH=01h,AH=06h,AH=08h,AH=0Ah

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 08h - CHARACTER INPUT WITHOUT ECHO**

Return: AL = character read from standard input

Notes: ^C/^Break are checked

SeeAlso: AH=01h,AH=06h,AH=07h,AH=0Ah,AH=64h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 09h - WRITE STRING TO STANDARD OUTPUT**

Entry: DS:DX -> '$'-terminated string

Return: AL = 24h

Notes: ^C/^Break are checked

SeeAlso: AH=02h,AH=06h"OUTPUT"

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 0Ah - BUFFERED INPUT**

Entry: DS:DX -> [buffer (see below](http://spike.scu.edu.au/~barry/interrupts.html#dosbuf))

Return: buffer filled with user input

Notes:

* ^C/^Break are checked
* reads from standard input

SeeAlso: AH=0Ch

Format of DOS input buffer:

|  |  |  |
| --- | --- | --- |
| **Offset** | **Size** | **Description** |
| 00 | 1 | maximum characters buffer can hold |
| 01 | 1 | number of chars from last input which may be recalled OR number of characters actually read, excluding CR |
| 02 | n | actual characters read, including the final carriage return |

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH=0Bh - GET STDIN STATUS**

Return:

* AL = 00h if no character available
* AL = FFh if character is available

Notes: ^C/^Break are checked

SeeAlso: AH=06h"INPUT"

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 0Ch - FLUSH BUFFER AND READ STANDARD INPUT**

Entry:

* AL = STDIN input function to execute after flushing buffer
* other registers as appropriate for the input function

Return: as appropriate for the specified input function

Note: if AL is not one of 01h,06h,07h,08h, or 0Ah, the buffer is flushed but no input is attempted

SeeAlso: AH=01h,AH=06h"INPUT",AH=07h,AH=08h,AH=0Ah

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 0Dh - DISK RESET**

Notes: This function writes all modified disk buffers to disk, but does not update the directory information

SeeAlso: AX=5D01h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 0Eh - SELECT DEFAULT DRIVE**

Entry: DL = new default drive (0=A:, 1=B:, etc)

Return: AL = number of potentially valid drive letters

Notes: the return value is the highest drive present

SeeAlso: AH=19h,AH=3Bh,AH=DBh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 19h - GET CURRENT DEFAULT DRIVE**

Return: AL = drive (0=A:, 1=B:, etc)

SeeAlso: AH=0Eh,AH=47h,AH=BBh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 25h - SET INTERRUPT VECTOR**

Entry:

* AL = interrupt number
* DS:DX -> new interrupt handler

Notes: this function is preferred over direct modification of the interrupt vector table

SeeAlso: AX=2501h,AH=35h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 2Ah - GET SYSTEM DATE**

Return: CX = year (1980-2099) DH = month DL = day AL = day of week (00h=Sunday)

SeeAlso: AH=2Bh"DOS",AH=2Ch,AH=E7h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 2Bh - SET SYSTEM DATE**

Entry: CX = year (1980-2099) DH = month DL = day

Return:

* AL = 00 successful
* FFh invalid date, system date unchanged

Note: DOS 3.3+ also sets CMOS clock

SeeAlso: AH=2Ah,AH=2Dh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 2Ch - GET SYSTEM TIME**

Return: CH = hour CL = minute DH = second DL = 1/100 seconds

Note: on most systems, the resolution of the system clock is about 5/100sec, so returned times generally do not increment by 1 on some systems, DL may always return 00h

SeeAlso: AH=2Ah,AH=2Dh,AH=E7h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 2Dh - SET SYSTEM TIME**

Entry: CH = hour CL = minute DH = second DL = 1/100 seconds

Return:

* AL = 00h successful
* FFh if invalid time, system time unchanged

Note: DOS 3.3+ also sets CMOS clock

SeeAlso: AH=2Bh"DOS",AH=2Ch

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 2Eh - SET VERIFY FLAG**

Entry: AL = new state of verify flag (00 off, 01h o)

Notes:

* default state at system boot is OFF
* when ON, all disk writes are verified provided the device driver supports read-after-write verification

SeeAlso: AH=54h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH=30h - GET DOS VERSION**

Entry: AL = what to return in BH (00h OEM number, 01h version flag)

Return:

* AL = major version number (00h if DOS 1.x)
* AH = minor version number
* BL:CX = 24-bit user serial number (most versions do not use this) if DOS <5 or AL=00h
* BH = MS-DOS OEM number if DOS 5+ and AL=01h
* BH = version flag bit 3: DOS is in ROM other: reserved (0)

Notes:

* DOS 4.01 and 4.02 identify themselves as version 4.00
* MS-DOS 6.21 reports its version as 6.20; version 6.22 returns the correct value
* Windows95 returns version 7.00 (the underlying MS-DOS)

SeeAlso: AX=3000h/BX=3000h,AX=3306h,AX=4452h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH=35h - GET INTERRUPT VECTOR**

Entry: AL = interrupt number

Return: ES:BX -> current interrupt handler

SeeAlso: AH=25h,AX=2503h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 36h - GET FREE DISK SPACE**

Entry: DL = drive number (0=default, 1=A:, etc)

Return:

* AX = FFFFh if invalid drive
* AX = sectors per cluster BX = number of free clusters CX = bytes per sector DX = total clusters on drive

Notes:

* free space on drive in bytes is AX \* BX \* CX
* total space on drive in bytes is AX \* CX \* DX
* "lost clusters" are considered to be in use
* this function does not return proper results on CD-ROMs; use AX=4402h"CD-ROM" instead

SeeAlso: AH=1Bh,AH=1Ch,AX=4402h"CD-ROM"

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 39h - "MKDIR" - CREATE SUBDIRECTORY**

Entry: DS:DX -> ASCIZ pathname

Return:

* CF clear if successful AX destroyed
* CF set on error AX = error code (03h,05h)

Notes:

* all directories in the given path except the last must exist
* fails if the parent directory is the root and is full
* DOS 2.x-3.3 allow the creation of a directory sufficiently deep that it is not possible to make that directory the current directory because the path would exceed 64 characters

SeeAlso: AH=3Ah,AH=3Bh,AH=6Dh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 3Ah - "RMDIR" - REMOVE SUBDIRECTORY**

Entry: DS:DX -> ASCIZ pathname of directory to be removed

Return:

* CF clear if successful, AX destroyed
* CF set on error AX = error code (03h,05h,06h,10h)

Notes: directory must be empty (contain only '.' and '..' entries)

SeeAlso: AH=39h,AH=3Bh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 3Bh - "CHDIR" - SET CURRENT DIRECTORY**

Entry: DS:DX -> ASCIZ pathname to become current directory (max 64 bytes)

Return:

* CF clear if successful, AX destroyed
* CF set on error AX = error code (03h)

Notes: if new directory name includes a drive letter, the default drive is not changed, only the current directory on that drive

SeeAlso: AH=47h,AH=71h,INT 2F/AX=1105h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 3Ch - "CREAT" - CREATE OR TRUNCATE FILE**

Entry:

* CX = [file attributes](http://spike.scu.edu.au/~barry/interrupts.html#fattr)
* DS:DX -> ASCIZ filename

Return:

* CF clear if successful, AX = file handle
* CF set on error AX = error code (03h,04h,05h)

Notes: if a file with the given name exists, it is truncated to zero length

SeeAlso: AH=16h,AH=3Dh,AH=5Ah,AH=5Bh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 3Dh - "OPEN" - OPEN EXISTING FILE**

Entry:

* AL = access and sharing modes
* DS:DX -> ASCIZ filename

Return:

* CF clear if successful, AX = file handle
* CF set on error AX = error code (01h,02h,03h,04h,05h,0Ch,56h)

Notes:

* file pointer is set to start of file
* file handles which are inherited from a parent also inherit sharing and access restrictions
* files may be opened even if given the hidden or system attributes

SeeAlso: AH=0Fh,AH=3Ch,AX=4301h,AX=5D00h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 3Eh - "CLOSE" - CLOSE FILE**

Entry: BX = file handle

Return:

* CF clear if successful, AX destroyed
* CF set on error, AX = error code (06h)

Note: if the file was written to, any pending disk writes are performed, the time and date stamps are set to the current time, and the directory entry is updated

SeeAlso: AH=10h,AH=3Ch,AH=3Dh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 3Fh - "READ" - READ FROM FILE OR DEVICE**

Entry:

* BX = file handle
* CX = number of bytes to read
* DS:DX -> buffer for data

Return:

* CF clear if successful - AX = number of bytes actually read (0 if at EOF before call)
* CF set on error AX = error code (05h,06h)

Notes:

* data is read beginning at current file position, and the file position is updated after a successful read
* the returned AX may be smaller than the request in CX if a partial read occurred
* if reading from CON, read stops at first CR

SeeAlso: AH=27h,AH=40h,AH=93h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH=40h - "WRITE" - WRITE TO FILE OR DEVICE**

Entry:

* BX = file handle
* CX = number of bytes to write
* DS:DX -> data to write

Return:

* CF clear if successful -AX = number of bytes actually written
* CF set on error - AX = error code (05h,06h)

Notes:

* if CX is zero, no data is written, and the file is truncated or extended to the current position
* data is written beginning at the current file position, and the file position is updated after a successful write
* the usual cause for AX < CX on return is a full disk

SeeAlso: AH=28h,AH=3Fh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 41H - "UNLINK" - DELETE FILE**

Entry:

* DS:DX -> ASCIZ filename (no wildcards, but see notes)
* CL = attribute mask for deletion (server call only, see notes)

Return:

* CF clear if successful, AX destroyed (DOS 3.3) AL seems to be drive of deleted file
* CF set on error AX = error code (02h,03h,05h)

Notes:

* (DOS 3.1+) wildcards are allowed if invoked via AX=5D00h, in which case the filespec must be canonical (as returned by AH=60h), and only files matching the attribute mask in CL are deleted
* DOS does not erase the file's data; it merely becomes inaccessible because the FAT chain for the file is cleared
* deleting a file which is currently open may lead to filesystem corruption.

SeeAlso: AH=13h,AX=4301h,AX=4380h,AX=5D00h,AH=60h,AH=71h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH=42h - "LSEEK" - SET CURRENT FILE POSITION**

Entry:

* AL = origin of move 00h start of file 01h current file position 02h end of file
* BX = file handle
* CX:DX = offset from origin of new file position

Return:

* CF clear if successful, DX:AX = new file position in bytes from start of file
* CF set on error, AX = error code (01h,06h)

Notes:

* for origins 01h and 02h, the pointer may be positioned before the start of the file; no error is returned in that case, but subsequent attempts at I/O will produce errors
* if the new position is beyond the current end of file, the file will be extended by the next write (see AH=40h)

SeeAlso: AH=24h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH=43 - GET FILE ATTRIBUTES**

Entry:

* AL = 00h
* DS:DX -> ASCIZ filename

Return:

* CF clear if successful CX = [file attributes](http://spike.scu.edu.au/~barry/interrupts.html#fattr)
* CF set on error, AX = error code (01h,02h,03h,05h)

BUG: Windows for Workgroups returns error code 05h (access denied) instead of error code 02h (file not found) when attempting to get the attributes of a nonexistent file.

SeeAlso: AX=4301h,AX=4310h,AX=7143h,AH=B6h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH=43 - "CHMOD" - SET FILE ATTRIBUTES**

Entry:

* AL = 01h
* CX = new [file attributes](http://spike.scu.edu.au/~barry/interrupts.html#fattr)
* DS:DX -> ASCIZ filename

Return:

* CF clear if successful, AX destroyed
* CF set on error, AX = error code (01h,02h,03h,05h)

Notes:

* will not change volume label or directory attribute bits, but will change the other attribute bits of a directory
* MS-DOS 4.01 reportedly closes the file if it is currently open

SeeAlso: AX=4300h,AX=4311h,AX=7143h,INT 2F/AX=110Eh

Bitfields for file attributes:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bits** | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Description** | shareable | - | archive | directory | vol. label | system | hidden | read-only |

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 47h - "CWD" - GET CURRENT DIRECTORY**

Entry:

* DL = drive number (00h = default, 01h = A:, etc)
* DS:SI -> 64-byte buffer for ASCIZ pathname

Return:

* CF clear if successful
* CF set on error, AX = error code (0Fh)

Notes:

* the returned path does not include a drive or the initial backslash
* many Microsoft products for Windows rely on AX being 0100h on success

SeeAlso: AH=19h,AH=3Bh,AH=71h

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 4Ch - "EXIT" - TERMINATE WITH RETURN CODE**

Entry: AL = return code

Return: never returns

Notes: unless the process is its own parent, all open files are closed and all memory belonging to the process is freed

SeeAlso: AH=00h,AH=26h,AH=4Bh,AH=4Dh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 4Dh - GET RETURN CODE (ERRORLEVEL)**

Return:

* AH = termination type (00=normal, 01h control-C abort, 02h=critical error abort, 03h terminate and stay resident)
* AL = return code

Notes:

* the word in which DOS stores the return code is cleared after being read by this function, so the return code can only be retrieved once
* COMMAND.COM stores the return code of the last external command it executed as ERRORLEVEL

SeeAlso: AH=4Bh,AH=4Ch,AH=8Ah

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 54h - GET VERIFY FLAG**

Return: AL = verify flag (00h=off, 01h=on, i.e. all disk writes verified after writing)

SeeAlso: AH=2Eh

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 56h - "RENAME" - RENAME FILE**

Entry:

* DS:DX -> ASCIZ filename of existing file (no wildcards, but see below)
* ES:DI -> ASCIZ new filename (no wildcards)
* CL = attribute mask (server call only, see below)

Return:

* CF clear if successful
* CF set on error, AX= error code (02h,03h,05h,11h)

Notes:

* allows move between directories on same logical volume
* this function does not set the archive attribute
* open files should not be renamed
* (DOS 3.0+) allows renaming of directories

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 57h - GET FILE'S LAST-WRITTEN DATE AND TIME**

Entry:

* AL = 00h (Get attribute)
* BX = file handle

Return:

* CF clear if successful, CX = file's time DX = file's date
* CF set on error, AX = error code (01h,06h)

SeeAlso: AX=5701h

Bitfields for file time:

|  |  |  |  |
| --- | --- | --- | --- |
| **Bits** | 15-11 | 10-5 | 4-0 |
| **Description** | hours | minutes | seconds |

Bitfields for file date:

|  |  |  |  |
| --- | --- | --- | --- |
| **Bits** | 15-9 | 8-5 | 4-0 |
| **Description** | year (1980-) | month | day |

[[Index]](http://spike.scu.edu.au/~barry/interrupts.html#Index)**AH = 57h - SET FILE'S LAST-WRITTEN DATE AND TIME**

Entry:

* AL =01h (Set attributes)
* BX = file handle
* CX = [new time](http://spike.scu.edu.au/~barry/dt)
* DX = [new date](http://spike.scu.edu.au/~barry/interrupts.html#dt)

Return:

* CF clear if successful
* CF set on error AX = error code (01h,06h)

SeeAlso: AX=5700h

*This page is maintained by*[*Barry Wilks*](http://spike.scu.edu.au/~barry/index.html)*.*