

# BDOS system calls (8-bit and 16-bit)

So far, this list covers CP/M 1, 2 and 3. In general, "CP/M 3 and above" means CP/M 3 and DR's DOS Plus:

LD	DE,parameter
LD	C,function
CALL	5

becomes:

MOV	DX,parameter
MOV	CL,function
INT	0E0h

while values are returned in AX and BX rather than BA and HL. This system is supported in DR Multiuser DOS.

Note that DOS + is a CP/M extended BDOS, DOS Plus is a PC operating system compatible with CP/M-86 and PCDOS 2.

If you have detailed information on MP/M, CP/Net, DOS Plus v2.1, CP/M-86 or Concurrent CP/M - please let me know! (See note on the homepage for my email address).

---

## BDOS function 0 - System Reset

Supported by: CP/M 1, 2, 3.

*Entered with C=0. Does not return.*

Quit the current program, return to command prompt. This call is hardly ever used since the RST 0 instruction does the same thing and saves four bytes.

## BDOS function 0 - System Reset

Supported by: DOS Plus.

*Entered with CL=0, DL=0 or 1. Does not return.*

Quit the current program, return to command prompt. If DL is 0, the memory used by the program is deallocated; if DL is 1, it remains resident.

---

## BDOS function 1 - Console input

Supported by: All versions

*Entered with C=1. Returns A=L=character.*

Wait for a character from the keyboard; then echo it to the screen and return it.

---

## BDOS function 2 - Console output

Supported by: All versions

*Entered with C=2, E=ASCII character.*

Send the character in E to the screen. Tabs are expanded to spaces. Output can be paused with ^S and restarted with ^Q (or any key under versions prior to CP/M 3). While the output is paused, the program can be terminated with ^C.

---

## **BDOS function 3 - Auxiliary (Reader) input**

**Supported by:** All versions

*Entered with C=3. Returns A=L=ASCII character*

Note that this call can hang if the auxiliary input never sends data.

---

## **BDOS function 4 - Auxiliary (Punch) output**

**Supported by:** All versions

*Entered with C=4, E=ASCII character.*

If the device is permanently not ready, this call can hang.

---

## **BDOS function 5 - Printer output**

**Supported by:** All versions

*Entered with C=2, E=ASCII character.*

If the printer is permanently offline or busy, this call can hang.

---

## **BDOS function 6 - Direct console I/O**

**Supported by:** All versions, with variations

*Entered with C=6, E=code. Returned values (in A) vary.*

E=0FFh

Return a character without echoing if one is waiting; zero if none is available.

E=0FEh

[CP/M3, NovaDOS, Z80DOS, DOS+] Return console input status. Zero if no character is waiting, nonzero otherwise.

E=0FDh

[CP/M3, DOS+] Wait until a character is ready, return it without echoing.

E=0FCh

[DOS+] One-character lookahead - return the next character waiting but leave it in the buffer.

Values of E not supported on a particular system will output the character. **Direct console functions may interact undesirably with non-direct ones, since certain buffers may be bypassed. Try not to mix them.**

---

## **BDOS function 7 - Get I/O byte**

Supported by: CP/M 2 and lookalikes

*Entered with C=7. Returns I/O byte.*

[Here](#)'s a description of how the IOBYTE works.

## **BDOS function 7 - Auxiliary Input status**

Supported by: CP/M 3 and above

*Entered with C=7. Returns A=0 or 0FFh.*

0FFh is returned if the Auxiliary Input device has a character ready; otherwise 0 is returned.

---

## **BDOS function 8 - Set I/O byte**

Supported by: CP/M 2 and lookalikes

*Entered with C=8, E=I/O byte.*

[Here](#)'s a description of how the IOBYTE works.

## **BDOS function 8 - Auxiliary Output status**

Supported by: CP/M 3 and above

*Entered with C=8. Returns A=0 or 0FFh.*

0FFh is returned if the Auxiliary Output device is ready for characters; otherwise 0 is returned.

---

## **BDOS function 9 - Output string**

Supported by: All versions

*Entered with C=9, DE=address of string.*

Display a string of ASCII characters, terminated with the \$ character. Thus the string may not contain \$ characters - so, for example, the VT52 cursor positioning command ESC Y y+32 x+32 will not be able to use row 4.

Under CP/M 3 and above, the terminating character can be changed using BDOS function [110](#).

---

## **BDOS function 10 - Buffered console input**

Supported by: All versions, with variations

*Entered with C=0Ah, DE=address or zero.*

This function reads characters from the keyboard into a memory buffer until RETURN is pressed. The Delete key is handled correctly. In later versions, more features can be used at this point; ZPM3 includes a full line editor with recall of previous lines typed.

On entry, DE is the address of a buffer. If DE=0 (in CP/M-86, DOS Plus etc, DX=0FFFFh), the [DMA address](#) is used (CP/M 3 and later) and the buffer already contains data:

DE=address:		DE=0 / DX=0FFFFh:	
buffer: DEFB	size	buffer: DEFB	size
DEFB	?	DEFB	len
DEFB	bytes	DEFB	bytes

The value at buffer+0 is the amount of bytes available in the buffer. Once the limit has been reached, no more can be added, although the line editor can still be used.

If DE=0 (in CP/M-86, DOS Plus etc, DX=0FFFFh) the next byte contains the number of bytes already in the buffer; otherwise this is ignored. On return from the function, it contains the number of bytes present in the buffer.

The bytes typed then follow. There is no end marker.

---

## BDOS function 11 - Console status

Supported by: All versions

*Entered with C=0Bh. Returns A=L=status*

Returns A=0 if no characters are waiting, nonzero if a character is waiting.

---

## BDOS function 12 - Lift head

Supported by: Version 1

*Entered with C=0Ch. Returns HL=0*

Has no effect except to return HL=0.

## BDOS function 12 - Return version number

Supported by: Versions 2.0 and later

*Entered with C=0Ch. Returns B=H=system type, A=L=version number.*

The system type is subdivided into a machine type and a CP/M type. The machine type occupies the high nibble of the byte; the CP/M type occupies the low nibble.

Machine types:	CP/M types:	Version numbers:
0 - 8080	0 - CP/M	00h - Version 1 (see above)
1 - 8086	1 - MP/M	20h - Version 2.0
	2 - CP/Net	21h - Version 2.1
		22h - Version 2.2
		25h - Version 2.5 (DOS +)
		30h - Version 3.0
		31h - Version 3.1
		41h - Version 4.1 (DOS Plus 1)
		60h - Version 6.0 (DOS Plus 2)

It is interesting to note that the version numbers returned by DRDOS and Novell DOS follow this system; DRDOS 5 is version 6.5, DRDOS 6 is version 6.7 and Novell DOS 7 is version 7.2.

---

## BDOS function 13 - Reset disks

Supported by: All versions.

*Entered with C=0Dh. Returned values vary.*

Resets disk drives. Logs out all disks and empties disk buffers. Sets the currently selected drive to A:. Any drives set to Read-Only in software become Read-Write; replacement BDOSses tend to leave them Read-Only.

In versions 1 and 2, logs in drive A: and returns 0FFh if there is a file present whose name begins with a \$, otherwise 0. Replacement BDOSses may modify this behaviour.

---

## BDOS function 14 - Select disk

Supported by: All versions

*Entered with C=0Eh, E=drive number. Returns L=A=0 or 0FFh.*

The drive number passed to this routine is 0 for A:, 1 for B: up to 15 for P:.

Sets the currently selected drive to the drive in A; logs in the disc. Returns 0 if successful or 0FFh if error. Under CP/M 3 and later, H can contain a physical error number.

---

## BDOS function 15 - Open file

Supported by: All versions

*Entered with C=0Fh, DE=FCB address. Returns error codes in BA and HL.*

This function opens a file to read or read/write. The FCB is a 36-byte data structure, most of which is maintained by CP/M. [Look here](#) for details.

The FCB should have its DR, Fn and Tn fields filled in, and the four fields EX, S1, S2 and RC set to zero. Under CP/M 3 and later, if CR is set to 0FFh then on return CR will contain the [last record byte count](#). Note that CR should normally be reset to zero if sequential access is to be used.

Under CP/M 3 and later, a password can be supplied to this function by pointing the [DMA address](#) at the password.

On return from this function, A is 0FFh for error, or 0-3 for success. Some versions (including CP/M 3) always return zero; others return 0-3 to indicate that an image of the directory entry is to be found at (80h+20h\*A).

If A=0FFh, CP/M 3 returns a [hardware error](#) in H and B. It also sets some bits in the FCB:

- F7' is set if the file is read-only because writing is password protected and no password was supplied;
  - F8' is set if the file is read-only because it is a User 0 system file opened from another user area.
- 

## BDOS function 16 - Close file

Supported by: All versions

*Entered with C=10h, DE=FCB address. Returns error codes in BA and HL.*

This function closes a file, and writes any pending data. This function should always be used when a file has been written to.

On return from this function, A is 0FFh for error, or 0-3 for success. Some versions always return zero; others return 0-3 to indicate that an image of the directory entry is to be found at (80h+20h\*A).

Under CP/M 3, if F5' is set to 1 then the pending data are written and the file is made consistent, but it remains open.

If A=0FFh, CP/M 3 returns a [hardware error](#) in H and B.

---

## **BDOS function 17 - search for first**

**Supported by:** All versions

*Entered with C=11h, DE=address of FCB. Returns error codes in BA and HL.*

Search for the first occurrence of the specified file; the filename should be stored in the supplied FCB. The filename can include ? marks, which match anything on disc. If the first byte of the FCB is ?, then any directory entry (including disc labels, date stamps etc.) will match. The EX byte is also checked; normally it should be set to zero, but if it is set to ? then all suitable extents are matched.

Returns A=0FFh if error (CP/M 3 returns a [hardware error](#) in H and B), or A=0-3 if successful. The value returned can be used to calculate the address of a memory image of the directory entry; it is to be found at [DMA](#)+A\*32.

---

## **BDOS function 18 - search for next**

**Supported by:** All versions

*Entered with C=12h, (DE=address of FCB)?. Returns error codes in BA and HL.*

This function should only be executed immediately after function 17 or another invocation of function 18. No other disc access functions should have been used.

Function 18 behaves exactly as number 17, but finds the next occurrence of the specified file after the one returned last time. The FCB parameter is not documented, but Jim Lopushinsky states in LD301.DOC:

In none of the official Programmer's Guides for any version of CP/M does it say that an FCB is required for Search Next (function 18). However, if the FCB passed to Search First contains an unambiguous file reference (i.e. no question marks), then the Search Next function requires an FCB passed in reg DE (for CP/M-80) or DX (for CP/M-86).

---

## **BDOS function 19 - delete file**

**Supported by:** All versions

*Entered with C=13h, DE=address of FCB. Returns error codes in BA and HL.*

Deletes all directory entries matching the specified filename. The name can contain ? marks. Returns A=0FFh if error, otherwise 0-3 (CP/M 3 returns a [hardware error](#) in H and B).

Under CP/M 3, if bit F5' is set to 1, the file remains but any password protection is removed. If the file has any password protection at all, the [DMA address](#) must be pointing at the password when this function is called.

---

## BDOS function 20 - read next record

*Entered with C=14h, DE=address of FCB. Returns error codes in BA and HL.*

Supported by: All versions

Load a record (normally 128 bytes, but under CP/M 3 this can be a multiple of 128 bytes) at the previously specified [DMA address](#). Values returned in A are:

0	OK,
1	end of file,
9	invalid FCB,
10	media changed,
0FFh	hardware error.

If on return A is not 0FFh, H contains the number of 128-byte records read before the error (CP/M 3 only).

---

## BDOS function 21 - write next record

Supported by: All versions

*Entered with C=15h, DE=address of FCB. Returns error codes in BA and HL.*

Write a record (normally 128 bytes, but can be a multiple of 128 bytes) to be found at previously specified [DMA address](#). Values returned in A are:

0	OK,
1	directory full,
2	disk full,
9	invalid FCB,
10	media changed,
0FFh	hardware error.

If on return A is not 0FFh, H contains the number of 128-byte records written before the error (CP/M 3 only).

---

## BDOS function 22 - create file

Supported by: All versions

*Entered with C=16h, DE=address of FCB. Returns error codes in BA and HL.*

Creates the file specified. Returns A=0FFh if the directory is full.

If the file exists already, then the default action is to return to the command prompt, but CP/M 3 may return a [hardware error](#) instead. Under CP/M 3, set F6' to create the file with a password; the [DMA address](#) should point at a 9-byte buffer:

DEFS	8	;Password
DEFB	1	; <a href="#">Password mode</a>

---

## BDOS function 23 - Rename file

Supported by: All versions

*Entered with C=17h, DE=address of FCB. Returns error codes in BA and HL.*

Renames the file specified to the new name, stored at FCB+16. This function cannot rename across drives so the "drive" bytes of both filenames should be identical. Returns A=0-3 if successful; A=0FFh if error. Under CP/M 3, if H is zero then the file could not be found; if it is nonzero it contains a [hardware error](#) number.

---

## BDOS function 24 - Return bitmap of logged-in drives

Supported by: All versions

*Entered with C=18h. Returns bitmap in HL.*

Bit 7 of H corresponds to P: while bit 0 of L corresponds to A:. A bit is set if the corresponding drive is logged in.

---

## BDOS function 25 - Return current drive

Supported by: All versions

*Entered with C=19h. Returns drive in A. Returns currently selected drive. 0 => A:, 1 => B: etc.*

---

## BDOS function 26 - Set DMA address

Supported by: All versions

*Entered with C=1Ah, DE=address.*

Set the Direct Memory Access address; a pointer to where CP/M should read or write data. Initially used for the transfer of 128-byte records between memory and disc, but over the years has gained many more functions.

---

## BDOS function 27 - Return address of allocation map

Supported by: All versions, but differs under v3.



*Entered with C=1Bh. Returns address in HL.*

Return the address of the allocation bitmap (which blocks are used and which are free) in HL. Under banked CP/M 3, this will be an address in bank 0 (the system bank) and not easily accessible.

Under previous versions, the format of the bitmap is a sequence of bytes, with bit 7 of the byte representing the lowest-numbered block on disc, and counting starting at block 0 (the directory). A bit is set if the corresponding block is in use.

Under CP/M 3, the allocation vector may be of this form (single-bit) or allocate two bits to each block (double-bit). This information is stored in the [SCB](#).

---

## **BDOS function 28 - Software write-protect current disc**

Supported by: All versions, with differences

*Entered with C=1Ch.*

Temporarily set current drive to be read-only; attempts to write to it will fail. Under genuine CP/M systems, this continues until either call [13 \(disc reset\)](#) or call [37 \(selective disc reset\)](#) is called; in practice, this means that whenever a program returns to the command prompt, all drives are reset to read/write. Newer BDOS replacements only reset the drive when function [37](#) is called.

---

## **BDOS function 29 - Return bitmap of read-only drives**

Supported by: All versions

*Entered with C=1Dh. Returns bitmap in HL.*

Bit 7 of H corresponds to P: while bit 0 of L corresponds to A:. A bit is set if the corresponding drive is set to read-only in software.

---

## **BDOS function 30 - set file attributes**

Supported by: All versions

*Entered with C=1Eh, DE=address of FCB. Returns error codes in BA and HL.*

Set and reset the bits required. Standard CP/M versions allow the bits F1', F2', F3', F4', T1' (read-only), T2' (system) and T3' (archive) to be changed. Some alternative BDOSses allow F5', F6', F7' and F8' to be set, but this is not to be encouraged since setting these bits can cause CP/M 3 to behave differently.

Under CP/M 3, the [Last Record Byte Count](#) is set by storing the required value at FCB+32 (FCB+20h) and setting the F6' bit.

The code returned in A is 0-3 if the operation was successful, or 0FFh if there was an error. Under CP/M 3, if A is 0FFh and H is nonzero, H contains a [hardware error](#).

---

## **BDOS function 31 - get DPB address**

Supported by: CP/M 2 and later.

*Entered with C=1Fh. Returns address in HL.*

Returns the address of the Disc Parameter Block for the current drive. See the [formats listing](#) for details of the DPBs under various CP/M versions.

---

## **BDOS function 32 - get/set user number**

Supported by: CP/M 2 and later.

*Entered with C=20h, E=number. If E=0FFh, returns number in A.*

Set current user number. E should be 0-15, or 255 to retrieve the current user number into A. Some versions can use user areas 16-31, but these should be avoided for compatibility reasons.

DOS+ returns the number set in A.

---

## **BDOS function 33 - Random access read record**

Supported by: CP/M 2 and later.

*Entered with C=21h, DE=FCB address. Returns error codes in BA and HL.*

Read the record specified in the random record count area of the FCB, at the [DMA address](#). The pointers in the FCB will be updated so that the next record to read using the sequential I/O calls will be the record just read. Error numbers returned are:

0	
1	OK
4	Reading unwritten data
6	Reading unwritten extent (a 16k portion of file does not exist)
9	Record number out of range
10	Invalid FCB
0FFh	Media changed
	[CP/M 3] <a href="#">hardware error</a> in H.

---

## **BDOS function 34 - Random access write record**

Supported by: CP/M 2 and later.

*Entered with C=22h, DE=FCB address. Returns error codes in BA and HL.*

Write the record specified in the random record count area of the FCB, at the [DMA address](#). The pointers in the FCB will be updated so that the next record to write using the sequential I/O calls will be the record just written. Error numbers returned are:

0	OK
---	----

2	Disc full
3	Cannot close extent
5	Directory full
6	Record number out of range
9	Invalid FCB
10	Media changed
0FFh	[CP/M 3] <a href="#">hardware error</a> in H.

If the record indicated is beyond the end of the file, the record will be written and the file may contain a gap; attempting to read this gap may give "reading unwritten data" errors, or nonsense.

---

## BDOS function 35 - Compute file size

Supported by: CP/M 2 and later.

*Entered with C=23h, DE=FCB address. Returns error codes in BA and HL.*

Set the random record count bytes of the FCB to the number of 128-byte records in the file. Returns A=0FFh if error (file not found, or CP/M 3 [hardware error](#)); otherwise A=0.

---

## BDOS function 36 - Update random access pointer

Supported by: CP/M 2 and later.

*Entered with C=24h, DE=FCB address.*

Set the random record count bytes of the FCB to the number of the last record read/written by the sequential I/O calls.

---

## BDOS function 37 - Selectively reset disc drives

Supported by: CP/M 2 and later.

*Entered with C=25h, DE=bitmap of drives to reset. Returns A=0.*

Bit 7 of D corresponds to P: while bit 0 of E corresponds to A:.. A bit is set if the corresponding drive should be reset. Resetting a drive removes its software read-only status.

---

## BDOS function 38 - Access drives

Supported by: MP/M, Concurrent CP/M.

---

## BDOS function 39 - Free drive

Supported by: MP/M, Concurrent CP/M.

---

## BDOS function 40 - Write random with zero fill

Supported by: CP/M 2 and later.

*Entered with C=28h, DE=FCB address. Returns error codes in BA and HL.*

As [function 34](#), but if the write is to a newly allocated disc block the remainder of the block is filled with zeroes.

---

## BDOS function 41 - Test and write record

Supported by: MP/M, Concurrent CP/M.

---

## BDOS function 42 - Lock record

Supported by: MP/M, Concurrent CP/M.

---

## BDOS function 43 - Unlock record

Supported by: MP/M, Concurrent CP/M.

---

## BDOS function 44 - Set number of records to read/write at once

Supported by: CP/M 3 and later.

*Entered with C=2Ch, E=number of records. Returns A=0 or 0FFh.*

Sets the number of records that functions 14, 15, 31, 32 and 40 should attempt to read at once. E should be 1-127. Returns A=0 if E was valid, 0FFh otherwise. The random access counters still operate with 128 bytes/record.

---

## BDOS function 45 - Set action on hardware error

Supported by: CP/M 3 and later.

*Entered with C=2Dh, E=action.*

Instructs CP/M 3 what action to take if there is a [hardware error](#):

E < 254

Compatibility mode; program is terminated and an error message printed.

E = 254

Error code is returned in H, error message is printed.

E = 255

Error code is returned in H, no error message is printed.

Note that the messages (if printed) are not followed by a carriage return or linefeed.

---

## BDOS function 46 - Find free space on a drive

Supported by: CP/M 3.

*Entered with C=2Eh, E=drive. Returns error codes in BA and HL.*

Sets the three bytes at the [DMA address](#) to the number of free 128-byte records on the disc, low byte first. E is the drive (0 for A:, 1 for B: etc.). If on return A=0FFh, there is a [hardware error](#) in H.

---

## BDOS function 47 - Chain to program

Supported by: CP/M 3 and later.

*Entered with C=2Fh, E=chain flag. Does not return.*

Execute the command line at 80h. The [error return code](#) is not changed, so the new program can discover the status returned by the previous one.

If E=255, then the currently set drive and user (as set by calls [25](#) and [32](#) become those used by the CCP; otherwise the CCP drive and user remain the same.

---

## BDOS function 48 - Empty disc buffers

Supported by: CP/M 3 and later.

*Entered with C=30h, E=flag. Returns error codes in BA and HL.*

Forces all pending disc writes to be executed. If E=255, then all disc read buffers are emptied as well (so that information must be read from the disc - used by file verification programs).

Returns A=0FFh if there is a [hardware error](#) in H.

---

## BDOS function 49 - Access the System Control Block

Supported by: CP/M 3.

*Entered with C=31h, DE=address of parameter area. Returned values vary.*

The [SCB](#) is a 100-byte area of memory (officially) used for storing system settings. The parameter area is formed:

pb+0:	DB	offset	;0-99
pb+1:	DB	command	;0 => Read byte at <i>offset</i> into A, and word at <i>offset</i> into HL.
			;0FEh => Write word at pb+2 to SCB at <i>offset</i> and <i>offset</i> +1
			;0FFh => Write byte at pb+2 to SCB at <i>offset</i>

## BDOS function 49 - Access the system variables

Supported by: DOS Plus.

*Entered with CL=31h, DX=address of parameter area. Returned values vary.*

The DOS Plus system variables contain various system settings. The parameter area is formed:

```
pb+0:  DB      system variable ;0-5
pb+1:  DB      command        ;0   => Read the specified variable into AL
                                   ;0FFh => Write data at pb+2 to specified variable.
```

Variables are:

Number	Length	Description
0	1	screen width-1
1	1	screen height-1
2	1	page mode - 0 if programs like TYPE should give paged output
3	1	system clock speed in Hz
4	1	temporary file drive, 0 for current or 1-16 for A: to P:
5	5	date and time
		DW      day count, day 1 is 1/1/78
		DB      BCD hours
		DB      BCD minutes
		DB      BCD seconds

---

## BDOS function 50 - Use the BIOS

Supported by: CP/M 3 and later.

*Entered with C=32h, DE=address of parameter area. Returned values vary.*

Under CP/M 3, the BIOS should not be directly called, except possibly the character I/O and [USERF](#) calls. Instead, this function should be used. The parameter area is formed:

```
pb+0:  DB      bios-function ;0-32
pb+1:  DB      bios-a       ;Value for A register
pb+2:  DB      bios-c       ;Value for C register
pb+3:  DB      bios-b       ;Value for B register
pb+4:  DB      bios-e       ;Value for E register
pb+5:  DB      bios-d       ;Value for D register
pb+6:  DB      bios-l       ;Value for L register
pb+7:  DB      bios-h       ;Value for H register
```

Under DOS Plus or CP/M-86, this function should be used for all BIOS calls; only character I/O BIOS calls are permitted.

---

## BDOS function 51 - Set DMA segment

Supported by: CP/M-86, DOS Plus.

*Entered with CL=33h, DX=segment value.*

---

## BDOS function 52 - Get DMA address

Supported by: DOS Plus.

*Entered with CL=34h. Returns address in ES:BX.*

---

## BDOS function 53 - Allocate maximum memory

Supported by: DOS Plus.

*Entered with CL=35h, DX=address of [MCB](#)*

Set MCB\_EXT to 2 if the memory should remain allocated after program termination. Set MCB\_LENGTH to maximum required length.

Returns MCB\_EXT=0 if no additional memory is available, 1 if there is. Sets MCB\_SEGMENT and MCB\_LENGTH to the values for the memory block allocated, which will be less than or equal to the size requested. Returns AX=0 if OK, 0FFFFh if error; CX=error code:

0 OK  
2 Illegal call number  
3 Out of memory  
12 Out of descriptors  
43 Bad Parameter

---

## **BDOS function 54 - Allocate absolute maximum memory**

Supported by: DOS Plus.

*Entered with CL=36h, DX=address of [MCB](#)*

Set MCB\_EXT to 2 if the memory should remain allocated after program termination. Set MCB\_LENGTH to maximum required length, and MCB\_SEGMENT to the required start paragraph.

Returns MCB\_EXT=0 if no additional memory is available, 1 if there is. Sets MCB\_SEGMENT and MCB\_LENGTH to the values for the memory block allocated, which will be less than or equal to the size requested. Returns AX=0 if OK, 0FFFFh if error; [CX=error code](#).

---

## **BDOS function 55 - Allocate memory**

Supported by: DOS Plus.

*Entered with CL=37h, DX=address of [MCB](#)*

Set MCB\_EXT to 2 if the memory should remain allocated after program termination. Set MCB\_LENGTH to maximum required length.

Returns MCB\_EXT=0 if no additional memory is available, 1 if there is. Sets MCB\_SEGMENT and MCB\_LENGTH to the values for the memory block allocated, which will be exactly the size requested. Returns AX=0 if OK, 0FFFFh if error; [CX=error code](#).

---

## **BDOS function 56 - Allocate absolute memory**

Supported by: DOS Plus.

*Entered with CL=38h, DX=address of [MCB](#)*

Set MCB\_EXT to 2 if the memory should remain allocated after program termination. Set MCB\_LENGTH to maximum required length, and MCB\_SEGMENT to the required start paragraph.

Returns MCB\_EXT=0 if no additional memory is available, 1 if there is. Sets MCB\_SEGMENT and MCB\_LENGTH to the values for the memory block allocated, which will be exactly the size requested.

Returns AX=0 if OK, 0FFFFh if error; [CX=error code](#).

---

## BDOS function 57 - Free memory

Supported by: DOS Plus.

*Entered with CL=39h, DX=address of [MCB](#)*

Set MCB\_SEGMENT to start of area, MCB\_LENGTH to length. It is not possible to deallocate the middle or start of an area which was allocated as one piece. Set MCB\_EXT to 0 to free the specified MCB, or 0FFh to free all this program's memory. Returns AX=0 if OK, 0FFFFh if error; [CX=error code](#).

---

## BDOS function 58 - Free all memory

Supported by: DOS Plus, CP/M-86.

*Entered with CL=3Ah. This function is obsolete in DOS Plus.*

---

## BDOS function 59 - Load overlay

Supported by: CP/M 3 and higher Loaders.

*Entered with C=3Bh, DE=FCB address. Returns error code in A.*

Load a program or RSX. The FCB should have been opened before making this call; the load address should be stored at FCB+21h. If the program is a PRL (Paged Relocatable), the original CP/M loader will relocate it; some replacement loaders, such as that in [CCP+](#) need a special flag set to indicate this.

If DE=0, the Loader will not load anything, but it will remove from the RSX chain any RSXs marked as due for deletion.

Returns A=0 if OK, 0FEh if there was an error, or 0FFh if the Loader is not present.

---

## BDOS function 60 - Call to RSX

Supported by: CP/M 3 and later RSXs.

*Entered with C=3Ch, DE=RSXPB address*

This call is provided for programs to communicate with Resident System Extensions. The format of the RSXPB is:

```
DEFB    function, 0-127 ;(128-255 used internally by CP/M)
DEFB    numwords       ;Number of parameter words passed to RSX.
DEFW    parameters
```

A popular convention is for the first parameter word to point to a copy within the program of the name of the intended RSX; for example:

```
RSXPB:  DEFB    100      ;Function
        DEFB    1        ;1 Parameter
        DEFW    RNAME    ;RSX name
RNAME:  DEFB    'GETERL  '
```



This call returns BA=HL=00FFh if the requested RSX is not resident; otherwise, the values are those returned by the RSX.

---

## **BDOS function 64 - Log in**

Supported by: CP/Net.

---

## **BDOS function 65 - Log off**

Supported by: CP/Net.

---

## **BDOS function 66 - Send message**

Supported by: CP/Net.

---

## **BDOS function 67 - Receive message**

Supported by: CP/Net.

---

## **BDOS function 68 - Get network status**

Supported by: CP/Net.

---

## **BDOS function 69 - Get configuration table address**

Supported by: CP/Net.

---

## **BDOS function 70 - Set compatibility attributes**

Supported by: CP/Net.

---

## **BDOS function 71 - Get server configuration**

Supported by: CP/Net.

---

## **BDOS function 98 - Clean up disc**

Supported by: CP/M 3.

*Entered with C=62h. Returns A=0 if OK, 0FFh if current drive is invalid*

This function is called by the CCP when a program terminates. Open files are closed; buffered data are not written, so files opened for writing may be destroyed if not properly closed.

---

## BDOS function 99 - Truncate file

Supported by: CP/M 3 and later.

*Entered with C=63h, DE=FCB address. Returns error codes in BA and HL.*

Set the random record count of the FCB to the required file size in records. If the file is password protected, point the [DMA](#) address at the password. Returns A=0-3 for success, 0FFh for failure; H may contain a [hardware error](#) code.

---

## BDOS function 100 - Set directory label

Supported by: CP/M 3 and later.

*Entered with C=64h, DE=FCB. Returns error codes in BA and HL.*

The EX byte (FCB+0Ch) of the FCB should contain a flags byte:

Bit	Meaning if set
-----	
7	Passwords enabled
6	Stamp on access
5	Stamp on update
4	Stamp on creation
0	Assign password at DMA+8 to label.

This function may need a password at the [DMA](#) address, if there is a label on the disc which has a password.

This function returns A=0 if OK, 0FFh for failure. H may contain a [hardware error](#) code.

---

## BDOS function 101 - Get directory label byte

Supported by: CP/M 3 and later.

*Entered with C=65h, E=drive. Returns A=byte*

The byte returned has the following bits set:

Bit	Meaning if set
-----	
7	Passwords enabled
6	Stamp on access
5	Stamp on update
4	Stamp on creation
0	Directory label exists

If bit 0 is zero, the other bits are meaningless.

If a label does exist, it can be found using functions [17](#) and [18](#) with the first 13 bytes of the FCB set to "?", and checking for a file whose user number is 32.

---

## BDOS function 102 - Get file date and time

Supported by: CP/M 3 and later.

*Entered with C=66h, DE=FCB address. Returns error codes in BA and HL.*

Gets the file date and time stamps, and the password mode. The stamps end up at FCB+18h (create or access), FCB+1Ch (update); the password mode ends up at FCB+0Ch. This returns A=0-3 for success, or 0FFh for failure. To tell whether a stamp is for creation or access, check the [directory label byte](#).

For information on the format of date and time stamps, see [the date and time calls](#). The password mode is documented in the next function.

---

## BDOS function 103 - Set file password and protection

Supported by: CP/M 3 and later.

*Entered with C=67h, DE=FCB address. Returns error codes in BA and HL.*

Sets file password and protection mode. Set FCB+0Ch to the required password mode, and point the [DMA address](#) at the current password. The password mode byte should have the following bits set:

Bit	Meaning if set
-----	
7	Reading requires password
6	Writing requires password
5	Deletion requires password
0	Assign password at DMA+8 to file.

---

## BDOS function 104 - Set date and time

Supported by: CP/M 3 and later; some CP/M 2 upgrades.

*Entered with C=68h, DE=address of time stamp.*

The format of the time stamp is:

DW	day	;Day 1 is 1 January 1978
DB	hour	;Packed BCD
DB	minute	;Packed BCD

The "seconds" counter will be reset to zero when this function is called.

---

## BDOS function 105 - Get date and time

Supported by: CP/M 3 and later; some CP/M 2 upgrades.

*Entered with C=69h, DE=address of time stamp. Returns A=seconds (packed BCD).*

The format of the time stamp is:

DW	day	;Day 1 is 1 January 1978
DB	hour	;Packed BCD
DB	minute	;Packed BCD

The "seconds" counter will be returned in A.

---

## BDOS function 106 - Set default password

Supported by: CP/M 3 and above.

*Entered with C=6Ah, DE=address of password.*

Sets the default password, used for all file operations requiring passwords when no password is given explicitly.

---

## BDOS function 107 - Get serial number

Supported by: CP/M 3 and above.

*Entered with C=6Bh, DE=address of buffer.*

Stores a 6-byte serial number at the address given by DE. Usually this number will be printable ASCII.

---

## BDOS function 108 - Get/put program return code

Supported by: CP/M 3 and above.

*Entered with C=6Ch, DE=code. Returns HL=code.*

If DE=0FFFFh, then the current code is returned in HL. Otherwise, it is set to the value in DE. Allowable values are:

00000h - 0FEFFh

No fatal error

0FF00h - 0FF7Fh

Fatal error

0FF80h - 0FFFCCh

Reserved

0FFFDh

Program terminated because of a [hardware error](#).

0FFFEh

Program terminated by Control-C.

If a program was chained by [function 47](#), an error code stored by the previous program will be available to it. Otherwise the CCP sets the return code to zero when it executes a program (some replacement CCPs do not do this).

If the error code is 0FF00h or above, and the next command begins with the character : then it will not be run.

---

## BDOS function 109 - Set or get console mode

Supported by: CP/M 3 and above

*Entered with C=6Dh, DE=mode or 0FFFFh. Returns HL=mode.* The mode is a bitmapped value. The significant bits are:

- 0 If set, function [11](#) detects only ^C.
- 1 If set, ^S does not pause screen output.
- 2 If set, output does not expand tabs, nor is it echoed to the printer even if ^P was pressed.
- 3 If set, ^C does not cause the program to terminate.

```
8 } These form a 2-bit number controlling GET and SUBMIT:
9 }
    0 Return "conditional" status to function 11.
      Programs using function 11 to test for interruption (eg SID) are not
      interrupted, but those using it to test for input do get the input.
1  Function 11 always returns "true".
2  Function 11 always returns "false".
3  Temporarily disable GET/SUBMIT, input comes from keyboard.
```

---

## BDOS function 110 - Get/set string delimiter

Supported by: CP/M 3 and above

*Entered with C=6Eh, DE=0FFFFh or ASCII value. Returns ASCII value in A.*

This function gets or sets the string delimiter (used in function [9](#) to mark the end of the string to print). If DE=0FFFFh, this gets the current ASCII value into A; otherwise it sets the delimiter to the value in E.

---

## BDOS function 111 - Send block of text to console

Supported by: CP/M 3 and above

*Entered with C=6Fh, DE=address of character control block.*

For 8-bit CP/M, the character control block is formed:

DW	text_begin
DW	text_length

For 16-bit CP/M, it is formed:

DW	text_begin_offset
DW	text_begin_segment
DW	text_length

---

## BDOS function 112 - Send block of text to printer

Supported by: CP/M 3 and above

*Entered with C=70h, DE=address of character control block.*

[Format of character control block](#)

---

## BDOS function 113 - Direct screen functions

Supported by: Personal CP/M.

---

## BDOS function 115 - Reserved for GSX

Supported by: GSX (Graphics System Extension)

*Entered with C=73h, DE=parameter block address.*

---

## BDOS function 116 - Set file date & time

Supported by: DOS Plus

*Entered with CL=74h, DX=FCB address. Returns AX=BX=error code.*

The required stamps should be put at the [DMA address](#). Returns AL = 0-3 for success, or 0FFh for failure.

---

## BDOS function 124 - Byte block copy

Supported by: Personal CP/M.

---

## BDOS function 125 - Byte block alter

Supported by: Personal CP/M.

---

## BDOS function 128 - Absolute memory request

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 129 - Relocatable memory request

Supported by: MP/M

---

## BDOS function 130 - Free memory

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 131 - Poll I/O device

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 132 - Wait on system flag

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 133 - Set system flag

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 134 - Create message queue

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 135 - Open message queue

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 136 - Delete message queue

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 137 - Read from message queue

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 138 - Conditionally read from message queue

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 139 - Write to message queue

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 140 - Conditionally write to message queue

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 141 - Delay

Supported by: MP/M, Concurrent CP/M, DOS Plus

*Entered with C=8Dh, DE=no. of ticks to wait.*

Delays calling process out for DE ticks of the system clock (system dependent; usually 50Hz or 60Hz). Returns A=0FFh if this function is unsupported.

---

## BDOS function 142 - Call dispatcher

Supported by: MP/M, Concurrent CP/M, DOS Plus

*Entered with C=8Eh.*

Relinquishes the CPU. The process restarts when its next time slice falls due. Returns A=0FFh if this function is unsupported.

---

## BDOS function 143 - Terminate process

Supported by: MP/M, Concurrent CP/M, DOS Plus

*Entered with C=8Fh.*

As [function 0](#).

---

## BDOS function 144 - Create a process

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 145 - Set process priority

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 146 - Attach console

Supported by: MP/M, Concurrent CP/M

*Entered with C=92h.*

Blocks the calling process until the system console is available. This is automatically called if any console I/O function is used while the current process is not attached to the console.

---

## BDOS function 147 - Detach console

Supported by: MP/M, Concurrent CP/M, DOS Plus

*Entered with C=93h.*

The program is disconnected from the screen and keyboard and must run in the background.

Under DOS Plus, attempting to use the keyboard will cause the program to be terminated. Note that backgrounded programs cannot use the INT 2nh DOS-like calls, only the INT E0h CP/M calls. Limited screen output can be achieved using BIOS function 8 (write to status line).

Under MP/M and CCP/M, attempting to output to the screen will cause function [146](#) to be invoked.

---

## BDOS function 148 - Set console

Supported by: MP/M

---



## BDOS function 149 - Assign console

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 150 - Send CLI command

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 151 - Call resident system process

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 152 - Parse filename

Supported by: MP/M, CP/M 3 and higher.

*Entered with C=98h, DE=address of PFCB. Returns codes in HL.*

This parses an ASCII filename to a FCB. The format of the PFCB is:

```
DEFW    ascii-address
DEFW    fcb-address
```

The value returned in HL is 0FFFFh if the ASCII filename was invalid; 0 if the ASCII filename was terminated with a zero or a carriage return; and otherwise the address of the next character after the filename.

The filename can be of any form up to the full *d:filename.typ;password* (password only under CP/M 3 and above). When the filename is parsed, the password is copied to FCB+10h, with its length at FCB+1Ah.

Under ZPM3, the filename may be of the form *du:filename.typ;password*; FCB+0Dh will be 0 if no user number was specified, or 1+number otherwise. If a Z-System is loaded, ZPM3 can also parse filenames of the form *dir:filename.typ;password*.

**BUG:** ZPM3 (version N10) puts the password at FCB+0Ch rather than FCB+10h, overwriting the user number stored at FCB+0Dh.

The filename can be terminated by any of: space tab cr nul ; = > < . : , [ ] / | .

---

## BDOS function 153 - Return console number

Supported by: MP/M, Concurrent CP/M

---

## BDOS function 154 - System data address

Supported by: MP/M, Concurrent CP/M, DOS Plus

*Entered with C=9Ah. Returns HL (or ES:BX) = system data address.*

---

## **BDOS function 155 - Get date and time**

Supported by: MP/M, Concurrent CP/M

---

## **BDOS function 156 - Return address of process descriptor**

Supported by: MP/M 2, Concurrent CP/M

---

## **BDOS function 157 - Abort a process**

Supported by: MP/M 2, Concurrent CP/M

---

## **BDOS function 158 - Attach printer**

Supported by: MP/M 2, Concurrent CP/M

*Entered with C=9Eh.*

Blocks the calling process until the default printer is available. This is automatically called if any printer function is used while the current process is not attached to the printer.

---

## **BDOS function 159 - Detach printer**

Supported by: MP/M 2, Concurrent CP/M

*Entered with C=9Fh.*

Detach this process from the printer,

---

## **BDOS function 160 - Select printer**

Supported by: MP/M 2, Concurrent CP/M

---

## **BDOS function 161 - Conditionally attach printer**

Supported by: MP/M 2, Concurrent CP/M

---

## **BDOS function 162 - Conditionally attach console**

Supported by: MP/M 2, Concurrent CP/M

---

## **BDOS function 163 - Return version number**

## BDOS function 164 - Return printer device number

Supported by: MP/M 2, Concurrent CP/M

---

## DOS+ function 210 - Return system information

Supported by: DOS+

*Entered with C=0D2h, E=number, 0-5. Returns values in BA and HL.*

Suitable arguments are:

0. Return base address for BDOS module. Used for dynamic reconfiguration.
1. Return the current DMA setting.
2. Flush any console input pending, emptying all buffers. Returns 0.
3. Returns list device status (ready/not ready)
4. Returns punch " " "
5. Returns reader " " "

Arguments 3 thru 5 represent ready as 0ffh, not ready as 0. If no custom installation for punch/reader device status has been made their status is returned 0 (not ready). Other arguments return 0.

---

## DOS+ function 211 - Print decimal number

Supported by: DOS+

*Entered with C=0D3h, DE=decimal number.*

Outputs the number as an unsigned 16-bit integer, with leading zero suppression.

---

- [Back to the CP/M archive listing](#)
- [Back to Home Page](#)

