



This is part of **Family API** which allow to create dual-os version of program runs under OS/2 and DOS

**Note:** This is legacy API call. It is recommended to use 32-bit equivalent

2021/09/17 04:47 · prokushev · [0 Comments](#)

2021/08/20 03:18 · prokushev · [0 Comments](#)

## DosDevIOCtl

This call performs control functions on a device specified by an opened device handle.

### Syntax

DosDevIOCtl (Data, ParmList, Function, Category, DevHandle)

### Parameters

;Data (PVOID) - input : Address of the data area. ;ParmList (PVOID) - input : Address of the command-specific argument list. ;Function (USHORT) - input : Device-specific function code. ;Category (USHORT) - input : Device category. ;DevHandle (HFILE) - input : Device handle returned by DosOpen or a standard (open) device handle.

### Return Code

rc (USHORT) - return Return code descriptions are: \* 0 NO\_ERROR \* 1 ERROR\_INVALID\_FUNCTION \* 6 ERROR\_INVALID\_HANDLE \* 15 ERROR\_INVALID\_DRIVE \* 31 ERROR\_GEN\_FAILURE \* 87 ERROR\_INVALID\_PARAMETER \* 115 ERROR\_PROTECTION\_VIOLATION \* 117 ERROR\_INVALID\_CATEGORY \* 119 ERROR\_BAD\_DRIVER\_LEVEL \* 163 ERROR\_UNCERTAIN\_MEDIA \* 165 ERROR\_MONITORS\_NOT\_SUPPORTED

### Remarks

Values returned in the range hex FF00 through FFFF are user dependent error codes. Values returned in the range hex FE00 through FEFF are device driver dependent error codes.

Refer to the IBM Operating System/2 Version 1.2 I/O Subsystems And Device Support Volume 1 for a complete listing of control functions (DevHlp calls).

### Family API Considerations

Some options operate differently in the DOS mode than in the OS/2 mode. Therefore, the following considerations apply to DosDevIOctl when coding for the DOS mode.

The level of support for DosDevIOctl is identified by category and function code with a noted restriction if it is not supported by DOS 2.X or DOS 3.X Functions tend to be more restrictive in lower version numbers of DOS.

\*Category 1 supported as follows: **41H Set Baud Rate** 42H Set Line Control **All other category 1 functions are not supported for DOS 2.X and DOS 3.X.** \*Category 2 not supported in FAPI \*Category 3 not supported in FAPI \*Category 4 not supported in FAPI \*Category 5 supported in FAPI as follows: 42H Set Frame control - supports IBM Graphics Printers only **44H Set Infinite Retry - for DOS 2.X and DOS 3.X, the function is in effect only for the duration of the calling program** 46H Initialize printer **62H Get Frame Control - not supported for DOS 2.X and DOS 3.X** 64H Get Infinite Retry **66H Get Printer Status.** \*Category 6 not supported in FAPI \*Category 7 not supported in FAPI \*Category 8 supported in FAPI as follows: 00H Lock Drive - not supported for versions below DOS 3.2 **01H Unlock Drive - not supported for versions below DOS 3.2** 02H Redetermine Media - not supported for versions below DOS 3.2 **03H Set Logical Map - not supported for versions below DOS 3.2** 20H Block Removable - not supported for versions below DOS 3.2 **21H Get Logical Map - not supported for versions below DOS 3.2** 43H Set Device Parameters - not supported for DOS 2.X and DOS 3.X **44H Write Track - not supported for DOS 2.X and DOS 3.X** 45H Format Track - not supported for DOS 2.X and DOS 3.X **63H Get Device Parameters - not supported for DOS 2.X and DOS 3.X** 64H Read Track - not supported for DOS 2.X and DOS 3.X \*\*65H Verify Track - not supported for DOS 2.X and DOS 3.X. \*Category 9 is reserved \*Category 10 (0AH) not supported in FAPI \*Category 11 (0BH) not supported in FAPI.

#### Example Code

#### C Binding

```
<PRE> #define INCL_DOSDEVICES
```

```
USHORT rc = DosDevIOctl(Data, ParmList, Function, Category, DevHandle);
```

```
PVOID Data; /* Data area */ PVOID ParmList; /* Command arguments */ USHORT Function; /* Device function */ USHORT Category; /* Device category */ HFILE DevHandle; /* Specifies the device */
```

```
USHORT rc; /* return code */ </PRE>
```

#### MASM Binding

```
<PRE> EXTRN DosDevIOctl:FAR INCL_DOSDEVICES EQU 1
```

```
PUSH@ OTHER Data ;Data area PUSH@ OTHER ParmList ;Command arguments PUSH WORD Function ;Device function PUSH WORD Category ;Device category PUSH WORD DevHandle ;Device handle CALL DosDevIOctl
```

```
Returns WORD </PRE>
```

# Note

Text based on [http://www.edm2.com/index.php/DosDevIOCtl\\_\(FAPI\)](http://www.edm2.com/index.php/DosDevIOCtl_(FAPI))

Family API		
DOS	Process Manager	<a href="#">DosBeep</a> <a href="#">DosExit</a> <a href="#">DosSleep</a> <a href="#">DosExecPgm</a>
	File Manager	<a href="#">DosChDir</a> <a href="#">DosChgFilePtr</a> <a href="#">DosClose</a> <a href="#">DosDelete</a> <a href="#">DosDupHandle</a> <a href="#">DosMkDir</a> <a href="#">DosMove</a> <a href="#">DosQCurDir</a> <a href="#">DosQCurDisk</a> <a href="#">DosSetFileMode</a> <a href="#">DosOpen</a> <a href="#">DosQFileInfo</a> <a href="#">DosRead</a> <a href="#">DosQFileMode</a> <a href="#">DosQFSInfo</a> <a href="#">DosQVerify</a> <a href="#">DosRmDir</a> <a href="#">DosSelectDisk</a> <a href="#">DosFindClose</a> <a href="#">DosFindFirst</a> <a href="#">DosFindNext</a> <a href="#">DosSetFileInfo</a> <a href="#">DosSetVerify</a> <a href="#">DosWrite</a> <a href="#">DosFileLocks</a> <a href="#">DosSetFHandState</a> <a href="#">DosNewSize</a> <a href="#">DosBufReset</a> <a href="#">DosQFHandState</a> <a href="#">DosSetFSinfo</a> <a href="#">DosShutdown</a>
	Memory Manager	<a href="#">DosFreeSeg</a> <a href="#">DosSubAlloc</a> <a href="#">DosSubFree</a> <a href="#">DosSubSet</a> <a href="#">DosAllocHuge</a> <a href="#">DosAllocSeg</a> <a href="#">DosReallocHuge</a> <a href="#">DosReallocSeg</a> <a href="#">DosGetHugeShift</a> <a href="#">DosCreateCSAlias</a>
	NLS	<a href="#">DosCaseMap</a> <a href="#">DosGetCtryInfo</a> <a href="#">DosGetDBCSEv</a> <a href="#">DosSetCtryCode</a> <a href="#">DosGetCollate</a> <a href="#">DosGetMessage</a> <a href="#">DosInsMessage</a> <a href="#">DosPutMessage</a>
	Date and Time	<a href="#">DosSetDateTime</a> <a href="#">DosGetDateTime</a>
	Devices	<a href="#">DosDevConfig</a> <a href="#">DosDevIOCtl</a> <a href="#">DosDevIOCtl2</a>
	Signals	<a href="#">DosHoldSignal</a> <a href="#">DosSetSigHandler</a>
	Misc	<a href="#">BadDynLink</a> <a href="#">DosGetEnv</a> <a href="#">DosGetMachineMode</a> <a href="#">DosGetVersion</a> <a href="#">DosError</a> <a href="#">DosErrClass</a> <a href="#">DosSetVec</a>
KBD		<a href="#">KbdCharIn</a> <a href="#">KbdFlushBuffer</a> <a href="#">KbdGetStatus</a> <a href="#">KbdSetStatus</a> <a href="#">KbdStringIn</a> <a href="#">KbdPeek</a>
VIO		<a href="#">VioGetBuf</a> <a href="#">VioGetConfig</a> <a href="#">VioGetCurPos</a> <a href="#">VioGetCurType</a> <a href="#">VioGetPhysBuf</a> <a href="#">VioReadCellStr</a> <a href="#">VioReadCharStr</a> <a href="#">VioScrollUp</a> <a href="#">VioScrollDn</a> <a href="#">VioScrollLf</a> <a href="#">VioScrollRt</a> <a href="#">VioScrUnLock</a> <a href="#">VioSetCurPos</a> <a href="#">VioSetCurType</a> <a href="#">VioSetMode</a> <a href="#">VioGetMode</a> <a href="#">VioShowBuf</a> <a href="#">VioWrtCellStr</a> <a href="#">VioWrtCharStr</a> <a href="#">VioWrtCharStrAtt</a> <a href="#">VioWrtNAttr</a> <a href="#">VioWrtNCell</a> <a href="#">VioWrtNChar</a> <a href="#">VioWrtTTY</a> <a href="#">VioScrLock</a> <a href="#">VioPopUp</a>
Tools		<a href="#">BIND</a>
Modules		<a href="#">DOSCALLS.DLL</a> <a href="#">VIOCALLS.DLL</a> <a href="#">KBDCALLS.DLL</a> <a href="#">MSG.DLL</a>
Libraries		<a href="#">API.LIB</a> <a href="#">OS2386.LIB</a> <a href="#">FAPI.LIB</a> <a href="#">DOSCALLS.LIB</a> <a href="#">SUBCALLS.LIB</a>

2018/08/25 15:05 · [prokushev](#) · [0 Comments](#)

From: <https://www.osfree.ru/doku/> - **osFree wiki**

Permanent link: <https://www.osfree.ru/doku/doku.php?id=en:docs:fapi:dosdevioctl&rev=1629434125>

Last update: **2021/08/20 04:35**

