 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

DosSetFHandState

This call sets the state of the specified file.

Specifies whether a handle of a named pipe can be inherited, and write-behind is allowed.

Syntax

```
DosSetFHandState (FileHandle, FileHandleState)
```

Parameters

- FileHandle (HFILE) - input : File handle to be set.
- FileHandleState (USHORT) - input : Contents of the open mode word defined in a previous DosOpen or DosOpen2 call.
    - 15 - Zero Bit field. This bit must be set to zero.
    - 14 - Write-Through flag:
        - 0 = Writes to the file may be run through the system buffer cache.
        - 1 = Writes to the file may go through the system buffer cache, but the data is written (actual file I/O completed) before a synchronous write call returns. This state of the file defines it as a synchronous file. For synchronous files, this is a mandatory bit in that the data must be written out to the medium for synchronous write operations.

This bit is not inherited by child processes.

- 13 - Fail-Errors flag. Media I/O errors are handled as follows:
    - 0 = Reported through the system critical error handler.
    - 1 = Reported directly to the caller by way of a return code.

Media I/O errors generated through an IOCTL category 8 function always get reported directly to the caller by way of a return code. The Fail-Errors function applies only to non-IOCTL handle-based type file I/O calls.

This bit is not inherited by child processes.

- 12 - No-Cache/Cache flag. The file is opened as follows:

    - 0 = It is advisable for the disk driver to cache the data in I/O operations on this file.
    - 1 = I/O to the file need not be done through the disk driver cache.

This bit is an advisory bit, and is used to advise FSDs and device drivers on whether it is worth caching the data or not. This bit, like the write-through bit, is a per-handle bit. This bit is not inherited by child processes.

- 11-8 - Reserved Bits. These bits are reserved and should be set to the values returned by DosQFHandState in these positions.
- 7 - Inheritance flag:
    - 0 = File handle is inherited by a spawned process resulting from a DosExecPgm call.
    - 1 = File handle is private to the current process.
- 6-4 - Zero Bit field. These bits must be set to zero. Any other values are invalid.
- 3 - Reserved Bit. This bit is reserved and should be set to the value returned by DosQFHandState for this position.
- 2-0 - Zero Bit field. These bits must be set to zero. Any other values are invalid.

Return Code

rc (USHORT) - return:Return code descriptions are:

- 0 NO_ERROR
- 6 ERROR_INVALID_HANDLE
- 87 ERROR_INVALID_PARAMETER

Remarks

OS/2 does not guarantee the order that sectors are written for multiple sector writes. If an application requires several sectors written in a specific order, the operator should issue them as separate synchronous write operations. Setting the write-through flag does not affect any previous writes. That data may remain in the buffers. When a critical error occurs that the application cannot handle, it may reset critical error handling to be performed by the system. This is done by calling `DosSetFHandState` to turn off the fail/errors bit and reissuing the I/O call. The expected critical error reoccurs and control is passed to the system critical error handler. The precise time that the effect of this function is visible at the application level is unpredictable when asynchronous I/O is pending.

The file handle state bits set by this function can be queried by DosQFHandState.

Family API Considerations

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

- The validity of the file handle is not checked.
- The Inheritance flag must be set equal to zero. (This flag is not supported in DOS 2.X.)
- The Write-Through flag must be set equal to zero.
- The Fail-Errors flag must be set equal to zero.

Named Pipe Considerations

DosSetFHandState allows setting of the inheritance (I) and Write-Through (W) bits. Setting W to 1 prevents write-behind operations on remote pipes.

Bindings

C

```
#define INCL_DOSFILEMGR

USHORT  rc = DosSetFHandState(FileHandle, FileHandleState);
HFILE   FileHandle;      /* File handle */
USHORT  FileHandleState; /* File handle state */
USHORT  rc;              /* return code */
```

MASM

```
EXTRN  DosSetFHandState:FAR
INCL_DOSFILEMGR     EQU 1

PUSH   WORD    FileHandle      ;File handle
PUSH   WORD    FileHandleState ;File handle state
CALL   DosSetFHandState

Returns WORD
```

Family API		
DOS	Process Manager	DosBeep DosExit DosSleep DosExecPgm
	File Manager	DosChDir DosChgFilePtr DosClose DosDelete DosDupHandle DosMkDir DosMove DosQCurDir DosQCurDisk DosSetFileMode DosOpen DosQFileInfo DosRead DosQFileMode DosQFSInfo DosQVerify DosRmDir DosSelectDisk DosFindClose DosFindFirst DosFindNext DosSetFileInfo DosSetVerify DosWrite DosFileLocks DosSetFHandState DosNewSize DosBufReset DosQFHandState DosSetFSinfo DosShutdown
	Memory Manager	DosFreeSeg DosSubAlloc DosSubFree DosSubSet DosAllocHuge DosAllocSeg DosReallocHuge DosReallocSeg DosGetHugeShift DosCreateCSAlias
	NLS	DosCaseMap DosGetCtryInfo DosGetDBCSEv DosSetCtryCode DosGetCollate DosGetMessage DosInsMessage DosPutMessage
	Date and Time	DosSetDateTime DosGetDateTime
	Devices	DosDevConfig DosDevIOCtl DosDevIOCtl2
	Signals	DosHoldSignal DosSetSigHandler
	Misc	BadDynLink DosGetEnv DosGetMachineMode DosGetVersion DosError DosErrClass DosSetVec

Family API	
KBD	KbdCharIn KbdFlushBuffer KbdGetStatus KbdSetStatus KbdStringIn KbdPeek
VIO	VioGetBuf VioGetConfig VioGetCurPos VioGetCurType VioGetPhysBuf VioReadCellStr VioReadCharStr VioScrollUp VioScrollDn VioScrollLf VioScrollRt VioScrUnLock VioSetCurPos VioSetCurType VioSetMode VioGetMode VioShowBuf VioWrtCellStr VioWrtCharStr VioWrtCharStrAtt VioWrtNAttr VioWrtNCell VioWrtNChar VioWrtTTY VioScrLock VioPopUp
Tools	BIND
Modules	DOSCALLS.DLL VIOCALLS.DLL KBDCALLS.DLL MSG.DLL
Libraries	API.LIB OS2386.LIB FAPI.LIB DOSCALLS.LIB SUBCALLS.LIB

2018/08/25 15:05 · prokushev · 0 Comments