



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](#)

VioGetPhysBuf

This call gets addressability to the physical display buffer.

Syntax

```
VioGetPhysBuf (DisplayBuf, Reserved)
```

Parameters

- DisplayBuf(PVIOPHYSBUF) - input/output : Address of the data structure that contains the physical display buffer address and length on input and returns the selectors used to address the display buffer.
 - displaybufaddr (PBYTE): Address of the 32 bit start address (selector:offset) of the physical display buffer passed as input. If displaybuflen is 0, then displaybufaddr is the far address of the PhysBuf Block described below.
 - displaybuflen (ULONG): 32 bit length of the physical display buffer. If displaybuflen is 0, then displaybufaddr is treated as the far address of the PhysBuf Block described below and the Selector List is not present.
 - selectors (SEL): Selector list.

Returns the selectors (each of word-length) that address the physical display buffer. The first selector returned in the list, addresses the first 64KB of the physical display buffer or displaybuflen, whichever is smaller. If displaybuflen is greater than 64KB, the second selector addresses the second 64KB. The last selector returned in the list, addresses the remainder of the display buffer. The application is responsible for ensuring enough space is reserved for the selector list to accommodate the specified buffer length.

- PhysBuf Block (PhysBuf): Address of the data structure. The PhysBuf Block is a variable length data structure. The first word is the Length of the PhysBuf Block in bytes. The remaining words of the structure are the selectors that address the physical video buffer. If Length is specified as 2, the required length of the PhysBuf Block is returned in its place.
- PhysBuf Block (USHORT) : Length of PhysBuf structure in bytes

selector (SEL)

```

        First selector
        selector (SEL)
        Next selector
        selector (SEL)
        ...
        selector (SEL)
        Last selector
* Reserved (USHORT) - input : Reserved word of 0s.

```

Return Code

rc (USHORT) - return

Return code descriptions are:

- 0 NO_ERROR
- 350 ERROR_VIO_PTR
- 429 ERROR_VIO_IN_BG
- 430 ERROR_VIO_ILLEGAL_DURING_POPUP
- 436 ERROR_VIO_INVALID_HANDLE
- 465 ERROR_VIO_DETACHED
- 494 ERROR_VIO_EXTENDED_SG

Remarks

If displaybuflen = 0, VioGetPhysBuf returns a selector that addresses the physical display buffer corresponding to the current mode. One selector is returned in Selector List. If a VioGetPhysBuf is issued after a VioGetBuf, then all VioWrtXX calls will no longer be written to the LVB. They will only be written to the physical display buffer. An application uses VioGetPhysBuf to get addressability to the physical display buffer. The selector returned by VioGetPhysBuf may be used only when an application program is executing in the foreground. When an application wants to access the physical display buffer, the application must call VioScrLock. VioScrLock either waits until the program is running in the foreground or returns a warning when the program is running in the background. For more information refer to VioScrLock and VioScrUnLock.

The buffer range specified for the physical screen buffer must fall between hex 'A0000' and 'BFFFF' inclusive. An application may issue VioGetPhysBuf only when it is running in the foreground. An application may issue VioGetPhysBuf more than once.

Bindings

C Binding

```

typedef struct _VIOPHYSBUF { /* viopb */
    PBYTE pBuf;           /* Buffer start address */

```

```

    ULONG      cb;           /* Buffer length */
    SEL       asel[1];       /* Selector list */
} VIOPHYSBUF;

#define INCL_VIO

USHORT   rc = VioGetPhysBuf(Structure, Reserved);

PVIOPHYSBUF     Structure;    /* Data structure */
USHORT          Reserved;    /* Reserved (must be zero) */

USHORT          rc;          /* return code */

```

MASM Binding

```

VIOPHYSBUF  struct
    viopb_pBuf dd ? ;Buffer start address
    viopb_cb   dd ? ;buffer length
    viopb_asel dw 1 dup (?) ;selector list
VIOPHYSBUF ends

EXTRN  VioGetPhysBuf:FAR
INCL_VIO          EQU 1

PUSH@ OTHER  Structure      ;Data structure
PUSH    WORD   Reserved      ;Reserved (must be zero)
CALL    VioGetPhysBuf

Returns WORD

```

Note

Text based on <http://www.edm2.com/index.php/VioGetPhysBuf>

Family API	
DOS	Process Manager DosBeep DosExit DosSleep DosExecPgm
	File Manager DosChDir DosChgFilePtr DosClose DosDelete DosDupHandle DosMkDir DosMove DosQCurDir DosQCurDisk DosSet FileMode DosOpen DosQFileInfo DosRead DosQ FileMode DosQFSInfo DosQVerify DosRmDir DosSelectDisk DosFindClose DosFindFirst DosFindNext DosSet FileInfo DosSet Verify DosWrite DosFileLocks DosSet FHandState DosNewSize DosBufReset DosQFHandState DosSet FInfo DosShutdown
	Memory Manager DosFreeSeg DosSubAlloc DosSubSet DosAllocHuge DosAllocSeg DosReallocHuge DosReallocSeg DosGet Huge Shift 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
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	DOSCALS.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

From:
<https://cocorico.osfree.org/doku/> - **osFree wiki**



Permanent link:
<https://cocorico.osfree.org/doku/doku.php?id=en:docs:fapi:viogetphysbuf>

Last update: **2021/09/19 02:17**