



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](#)

## VioGetConfig

This call returns the video display configuration.

### Syntax

```
VioGetConfig (ConfigID, ConfigData, VioHandle)
```

### Parameters

- ConfigID ([USHORT](#)) - input: Identifies for which display configuration information is being requested:
  - 0 - Current configuration
  - 1 - Primary configuration
  - 2 - Secondary configuration

For OS/2 1.2, when ConfigID = 0, the current configuration is returned rather than the primary configuration (as was returned in OS/2 1.0 and 1.1). This change makes the OS/2 mode version of VioGetConfig match the family API version that has returned the current configuration starting with OS/2 1.0. OS/2 1.0 and 1.1 applications that issued VioGetConfig to determine the display configuration benefit from this change. The application can run on the configuration selected by the operator (by issuing the MODE command before invoking the application) rather than switching away from the operator selected display.

- ConfigData ([PVIICONFIGINFO](#)) - output: Address of structure where the display configuration is returned.
- VioHandle ([HVIO](#)) - input : This must be zero unless the caller is a Presentation Manager application, in which case it must be the value returned by VioGetPs.

### Return Code

rc ([USHORT](#)) - return

Return code descriptions are:

- 0 NO\_ERROR
- 421 ERROR\_VIO\_INVALID\_PARMS
- 436 ERROR\_VIO\_INVALID\_HANDLE
- 438 ERROR\_VIO\_INVALID\_LENGTH
- 465 ERROR\_VIO\_DETACHED

## Remarks

The values returned may not be correct if the adapter cannot be properly identified by the Base Video Handler (BVH) selected at system installation time. It can also be incorrect if the physical setup does not match that indicated by the presence of the adapter or by adapter switches. For example, it is impossible to detect the absence of a display on a CGA or the display attached to an EGA, despite the setup switches.

## Bindings

### C

```
typedef struct _VIOCONFIGINFO { /* voin */
    USHORT cb; /* Length of this data structure */
    USHORT adapter; /* Display adapter type */
    USHORT display; /* Display/monitor type */
    ULONG cbMemory; /* Amount of memory on the adapter
                    in bytes */

    USHORT Configuration;
    USHORT VDHVersion;
    USHORT Flags;
    ULONG HWBufferSize;
    ULONG FullSaveSize;
    ULONG PartSaveSize;
    USHORT EMAdaptersOFF; /* Offset to emulated adapter types */
    USHORT EMDisplaysOFF; /* Offset to emulated display types */
} VIOCONFIGINFO;

#define INCL_VIO

USHORT rc = VioGetConfig(ConfigID, ConfigData, VioHandle);

USHORT ConfigID; /* Configuration ID */
P_VIOCONFIGINFO ConfigData; /* Configuration data */
HVIO VioHandle; /* Vio handle */

USHORT rc; /* return code */
```

# MASM

```
VIOCONFIGINFO struct
    vioin_cb          dw ? ;Length of this data structure
    vioin_adapter     dw ? ;Display adapter type
    vioin_display     dw ? ;Display/monitor type
    vioin_cbMemory    dd ? ;Amount of memory on the adapter in bytes
    vioin_Configuration dw ? ;
    vioin_VDHVersion  dw ? ;
    vioin_Flags       dw ? ;
    vioin_HWBufferSize dd ? ;
    vioin_FullSaveSize dd ? ;
    vioin_PartSaveSize dd ? ;
    vioin_EMAdaptersOFF dw ? ;Offset to emulated adapter types
    vioin_EMDisplaysOFF dw ? ;Offset to emulated display types
VIOCONFIGINFO ends
```

```
EXTRN VioGetConfig:FAR
INCL_VIO EQU 1
```

```
PUSH WORD ConfigID ;Configuration ID
PUSH@ OTHER ConfigData ;Configuration data
PUSH WORD VioHandle ;Vio handle
CALL VioGetConfig
```

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
KBD	KbdCharIn KbdFlushBuffer KbdGetStatus KbdSetStatus KbdStringIn KbdPeek	

<b>Family API</b>	
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:viogetconfig>

Last update: **2021/09/19 05:21**

