



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](#)

## KbdGetStatus

This call gets the current state of the keyboard.

### Syntax

```
KbdGetStatus (StatData, KbdHandle)
```

### Parameters

- StatData (PKBDINFO) - output : Address of the keyboard status structure:
  - length (USHORT) : Length, in bytes, of this data structure, including length.
    - 10 Only valid value.
  - sysstate (USHORT) : State as follows:

^ Bit ^ Description ^

15-9	Reserved, set to zero.		
8	Shift return is on.		
7	Length of the turn-around character (meaningful only if bit 6 is on).		
6	Turn-around character is modified.		
5	Interim character flags are modified.		
4	Shift state is modified.		
3	ASCII mode is on.		
2	Binary mode is on.		
1	Echo off.		

0	Echo on.	* turnchardef (USHORT) : Definition of the turn-around character. In ASCII and extended-ASCII format, the turn-around character is defined as the carriage return. In ASCII format only, the turn-around character is defined in the low-order byte. * intcharflag (USHORT) : Interim character flags:	<b>Bit Description</b>
15-8	NLS shift state.		
7	Interim character flag is on.		
6	Reserved, set to zero.		
5	Application requested immediate conversion.		
4-0	Reserved, set to zero.	* shiftstate (USHORT) : Shift state as follows:	<b>Bit Description</b>
15	SysReq key down		
14	CapsLock key down		
13	NumLock key down		
12	ScrollLock key down		
11	Right Alt key down		
10	Right Ctrl key down		
9	Left Alt key down		
8	Left Ctrl key down		
7	Insert on		
6	CapsLock on		
5	NumLock on		
4	ScrollLock on		
3	Either Alt key down		
2	Either Ctrl key down		
1	Left Shift key down		
0	Right Shift key down		

- KbdHandle (HKBD) - input : Default keyboard or the logical keyboard.

## Return Code

rc (USHORT) - return

Return code descriptions are:

- 0 NO\_ERROR
- 376 ERROR\_KBD\_INVALID\_LENGTH
- 439 ERROR\_KBD\_INVALID\_HANDLE
- 445 ERROR\_KBD\_FOCUS\_REQUIRED

- 447 ERROR\_KBD\_KEYBOARD\_BUSY
- 464 ERROR\_KBD\_DETACHED
- 504 ERROR\_KBD\_EXTENDED\_SG

## Remarks

The initial state of the keyboard is established by the system at application load time. Some default states may be modified by the application through KbdSetStatus. KbdGetStatus returns only those keyboard parameters initially set by KbdSetStatus. The returned parameters are:

- Input Mode
- Interim Character Flags
- Shift State
- Echo State
- Turnaround Character

KbdGetStatus completes only when the handle has access to the physical keyboard (focus) or the handle is 0 and no other handle has the focus.

## Family API Considerations

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

- Interim character is not supported
- TurnAround character is not supported
- NLS\_SHIFT\_STATE is always NULL.
- KbdHandle is ignored.

## Bindings

### C Binding

```
typedef struct _KBDINFO {           /* kbst */
    USHORT cb;                       /* length in bytes of this structure */
    USHORT fsMask;                   /* bit mask of functions to be altered */
    USHORT chTurnAround;             /* define TurnAround character */
    USHORT fsInterim;               /* interim character flags */
    USHORT fsState;                 /* shift states */
}KBDINFO;

#define INCL_KBD

USHORT rc = KbdGetStatus(Structure, KbdHandle);

PKBDINFO Structure; /* Data structure */
```

```
HKBD          KbdHandle;      /* Keyboard handle */

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

## MASM Binding

```
KBDINFO struc
    kbst_cb          dw ? ;length in bytes of this structure
    kbst_fsMask      dw ? ;bit mask of functions to be altered
    kbst_chTurnAround dw ? ;define TurnAround character
    kbst_fsInterim   dw ? ;interim character flags
    kbst_fsState     dw ? ;shift states
KBDINFO ends

EXTRN KbdGetStatus:FAR
INCL_KBD EQU 1

PUSH@ OTHER Structure ;Data structure
PUSH WORD KbdHandle ;Keyboard handle
CALL KbdGetStatus

Returns WORD
```

## Note

This text based on [http://www.edm2.com/index.php/KbdGetStatus\\_\(FAPI\)](http://www.edm2.com/index.php/KbdGetStatus_(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>	

<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:  
<http://www.osfree.su/doku/> - **osFree wiki**

Permanent link:  
<http://www.osfree.su/doku/doku.php?id=en:docs:fapi:kbdgetstatus&rev=1631978976>

Last update: **2021/09/18 15:29**

