# KbdOpen

**Bindings**:

**C**:

```
#define INCL_KBD

USHORT  rc = KbdOpen(KbdHandle);

PHKBD            KbdHandle;     /* Keyboard handle */

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

**MASM**:

```
EXTRN  KbdOpen:FAR
INCL_KBD            EQU 1

PUSH@  WORD    KbdHandle     ;Keyboard handle
CALL   KbdOpen

Returns WORD
```

This call creates a new logical keyboard.

KbdOpen (KbdHandle)

*KbdHandle* (**PHKBD**) - output Address of the logical keyboard.

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

| | |
|---|---|
| 0 | NO_ERROR |
| 440 | ERROR_KBD_NO_MORE_HANDLE |
| 441 | ERROR_KBD_CANNOT_CREATE_KCB |
| 464 | ERROR_KBD_DETACHED |
| 504 | ERROR_KBD_EXTENDED_SG |

**Remarks**

*KbdOpen* blocks while another thread has the keyboard focus (by way of *KbdGetFocus*) until the thread with the focus issues *KbdFreeFocus*. Therefore, to prevent *KbdOpen* from blocking, it is recommended that *KbdOpen* be issued only while the current thread has the focus. For example:

*KbdGetFocus* wait until focus available on handle 0 *KbdOpen* get a logical keyboard handle *KbdOpen* get another logical keyboard handle *KbdOpen* get yet another logical keyboard handle *KbdFreeFocus* give up the focus on handle 0.

From:
http://osfree.su/doku/ - **osFree wiki**

Permanent link:
**http://osfree.su/doku/doku.php?id=en:ibm:prcp:kbd:open&rev=1400261880**

Last update: **2014/05/16 17:38**