

Note: This API calls are shared between DOS and Win16 personality.

DPMI is a shared interface for DOS applications to access Intel 80286+ CPUs services. DOS DMPI host provides core services for protected mode applications. Multitasking OS with DOS support also provides DMPI in most cases. Windows standard and extended mode kernel is a DPMI client app. Standard and extended mode kernel differs minimally and shares common codebase. Standard Windows kernel works under DOSX extender. DOSX is a specialized version of 16-bit DPMI Extender (but it is standard DPMI host). Standard mode is just DPMI client, exnhanced mode is DPMI client running under Virtual Machime Manager (really, multitasker which allow to run many DOS sessions). Both modes shares DPMI interface for kernel communication. The OS/2 virtual DOS Protected Mode Interface (VDPMI) device driver provides Version 0.9 DPMI support for virtual DOS machines. Win16 (up to Windows ME) provides Version 0.9 DPMI support. Windows in Standard Mode provides DPMI services only for Windows Applications, not DOS sessions.

DPMI host often merged with DPMI extender. Usually DPMI extender provide DPMI host standard services and DOS translation or True DPMI services.

2021/08/05 10:15 · prokushev · 0 Comments

Int 31H, AH=00H, AL=0FH

Version

1.0

Brief

Set Multiple Descriptors

Input

```
AX = 000FH
CX = number of descriptors to copy
ES:(E)DI = selector:offset of a buffer in the following format:
```

Offset	Length	Contents
00H	2	Selector #1
02H	8	Descriptor #1
0AH	2	Selector #2
0CH	8	Descriptor #2
...	...	...

```
if function successful
Carry flag = clear

if function unsuccessful
Carry flag = set
AX = error code
8021H   invalid value (access rights/type bytes invalid)
8022H   invalid selector
8025H   invalid linear address (descriptor references a linear address range
outside that allowed for DPMI clients)
CX = number of descriptors successfully copied
```

Notes

Copies one or more descriptors from a client buffer into the local descriptor table (LDT).

If an error occurs because of an invalid selector or descriptor, the function returns the number of descriptors which were successfully copied in CX. All of the descriptors which were copied prior to the one that failed are valid. All descriptors from the invalid entry to the end of the table are not updated.

32-bit programs must use ES:EDI to point to the buffer. 16-bit programs should use ES:DI.

A descriptor's access rights/type byte (byte 5) follows the same format and restrictions as the access rights/type parameter (in CL) for the Set Descriptor Access Rights function (Int 31H Function 0009H). On 80386 (or later) machines, the descriptor's extended access rights/type byte (byte 6) follows the same format and restrictions as the extended access rights/type parameter (in CH) for the same function, except the low-order 4 bits (marked “reserved”) are used to set the upper 4 bits of the descriptor's limit.

If the descriptor's present bit is not set, then the only error checking is that the client's CPL must be equal to the descriptor's DPL field and the “must be 1” bit in the descriptor's byte 5 must be set.

A DPMI 1.0 host will reload any segment register which contains a selector specified in the data structure supplied to this function. It is suggested that DPMI 0.9 hosts also implement this.

Refer to the rules for descriptor usage in Appendix D.

See also

Note

Text based on http://www.delorie.com/djgpp/doc/dpmi/

DPMI	
Process manager	**INT 2FH** 1680H, 1687H
Signals	

DPMI	
Memory manager	
Misc	**INT 2FH** 1686H, 168AH
Devices	

2021/08/13 14:23 · prokushev · 0 Comments

From:
http://ftp.osfree.org/doku/ - **osFree wiki**

Permanent link:
http://ftp.osfree.org/doku/doku.php?id=en:docs:dpmi:api:int31:00:0f

Last update: **2021/08/27 01:56**