

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=09H

Version

0.9

Brief

Set Descriptor Access Rights

Input

```
AX = 0009H
BX = selector
CL = access rights/type byte
CH = 80386 extended access rights/type byte
```

Return

```
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 (changing the access rights/type bytes would
cause the descriptor to reference a linear address range outside that
allowed for DPMI clients.)
```

Notes

Modifies the access rights and type fields in the LDT descriptor for the specified segment.

The access rights/type byte passed to the function in CL has the following format:

```
+---+---+---+---+---+---+---+---+
| P |  DPL  | 1 |C/D|E/C|W/R| A |
+-+-+---+---+-+-+-+-+-+-+-+-+-+-+
  |     |     |   |   |   |   |
  |     |     |   |   |   |   +- 0=not accessed, 1=accessed
  |     |     |   |   |   +----- data: 0=read, 1=>read/write
  |     |     |   |   |          code: must be 1 (readable)
  |     |     |   |   +--------- data: 0=expand-up, 1=expand-down
  |     |     |   |              code: must be 0 (non-conforming)
  |     |     |   +------------- 0=data, 1=code
  |     |     +----------------- must be 1
  |     +----------------------- must equal caller's CPL
  +----------------------------- 0=absent, 1=present
```

If the Present bit is not set in the descriptor, the DPMI host allows any values except in the DPL and “must be 1” bit fields.

On 80386 (and later) machines, the DPMI host interprets the value passed to the function in CH as follows:

```
+---+---+---+---+---+---+---+---+
| G |B/D| 0 |Avl|    Reserved   |
+-+-+-+-+-+-+-+-+---+---+---+---+
  |   |   |   |         |
  |   |   |   |         +-- ignored
  |   |   |   +------------ can be 0 or 1
  |   |   +---------------- must be 0
  |   +-------------------- 0=default 16-bit, 1=default 32-bit
  +------------------------ 0=byte granular, 1=page granular
```

A DPMI 1.0 host will reload any segment registers which contain the selector specified in register BX. It is suggested that DPMI 0.9 hosts also implement this.

Client programs should use the LAR instruction to examine the access rights of a descriptor.

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

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

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

Permanent link:
https://ftp.osfree.org/doku/doku.php?id=en:docs:dpmi:api:int31:00:09

Last update: **2021/08/27 01:38**