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=00H

Version

0.9

Brief

Allocate LDT Descriptors

Input

```
AX = 0000H
CX = number of descriptors to allocate
```

Return

```
if function successful
Carry flag = clear
AX = base selector
```

```
if function unsuccessful
Carry flag = set
AX = error code
8011H   descriptor unavailable
```

Notes

Allocates one or more descriptors in the task's Local Descriptor Table (LDT). The descriptor(s) allocated must be initialized by the application with other function calls.

If more than one descriptor was requested, the function returns a base selector referencing the first of a contiguous array of descriptors. The selector values for subsequent descriptors in the array can be calculated by adding the value returned by Int 31H Function 0003H.

The allocated descriptor(s) will be set to “data” with the present bit set and a base and limit of zero. The privilege level of the descriptor(s) will match the application's code segment privilege level.

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

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

Last update: **2021/08/27 01:23**