



**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, enhanced mode is DPMI client running under Virtual Machine 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](#)

**Note: This API implemented by DPMI extenders under DOS and by DOSX.EXE under Windows. Under OS/2 MVDM this API implemented by VDPMI**

Interrupt	AX	Description	Version	32-bit host	16-bit host
2f	<a href="#">1680</a>	Release Current Virtual Machine's Time Slice	1.0	Yes	Yes
2f	<a href="#">1686</a>	Get CPU Mode	0.9		
2f	<a href="#">1687</a>	Obtain Real-to-Protected Mode Switch Entry Point	0.9		
2f	<a href="#">168A</a>	Get Vendor-Specific API Entry Point	1.0		
31	<a href="#">0000</a>	Allocate LDT Descriptors	0.9		
31	<a href="#">0001</a>	Free LDT Descriptor	0.9		
31	<a href="#">0002</a>	Segment to Descriptor	0.9		
31	<a href="#">0003</a>	Get Selector Increment Value	0.9		
31	<a href="#">0006</a>	Get Segment Base Address	0.9		
31	<a href="#">0007</a>	Set Segment Base Address	0.9		
31	<a href="#">0008</a>	Set Segment Limit	0.9		
31	<a href="#">0009</a>	Set Descriptor Access Rights	0.9		
31	<a href="#">000A</a>	Create Alias Descriptor	0.9		
31	<a href="#">000B</a>	Get Descriptor	0.9		
31	<a href="#">000C</a>	Set Descriptor	0.9		
31	<a href="#">000D</a>	Allocate Specific LDT Descriptor	0.9		
31	<a href="#">000E</a>	Get Multiple Descriptors	1.0		
31	<a href="#">000F</a>	Set Multiple Descriptors	1.0		
31	<a href="#">0100</a>	Allocate DOS Memory Block	0.9		
31	<a href="#">0101</a>	Free DOS Memory Block	0.9		
31	<a href="#">0102</a>	Resize DOS Memory Block	0.9		
31	<a href="#">0200</a>	Get Real Mode Interrupt Vector	0.9		

Interrupt	AX	Description	Version	32-bit host	16-bit host
31	0201	Set Real Mode Interrupt Vector	0.9		
31	0202	Get Processor Exception Handler Vector	0.9		
31	0203	Set Processor Exception Handler Vector	0.9		
31	0204	Get Protected Mode Interrupt Vector	0.9		
31	0205	Set Protected Mode Interrupt Vector	0.9		
31	0210	Get Extended Processor Exception Handler Vector (Protected Mode)	1.0		
31	0211	Get Extended Processor Exception Handler Vector (Real Mode)	1.0		
31	0212	Set Extended Processor Exception Handler Vector (Protected Mode)	1.0		
31	0213	Set Extended Processor Exception Handler Vector (Real Mode)	1.0		
31	0300	Simulate Real Mode Interrupt	0.9		
31	0301	Call Real Mode Procedure With Far Return Frame	0.9		
31	0302	Call Real Mode Procedure With IRET Frame	0.9		
31	0303	Allocate Real Mode Callback Address	0.9		
31	0304	Free Real Mode Callback Address	0.9		
31	0305	Get State Save/Restore Addresses	0.9		
31	0306	Get Raw Mode Switch Addresses	0.9		
31	0400	Get Version	0.9		
31	0401	Get DPMI Capabilities	1.0		
31	0500	Get Free Memory Information	0.9		
31	0501	Allocate Memory Block	0.9		
31	0502	Free Memory Block	0.9		
31	0503	Resize Memory Block	0.9		
31	0504	Allocate Linear Memory Block	1.0		
31	0505	Resize Linear Memory Block	1.0		
31	0506	Get Page Attributes	1.0		
31	0507	Set Page Attributes	1.0		
31	0508	Map Device in Memory Block	1.0		
31	0509	Map Conventional Memory in Memory Block	1.0		
31	050A	Get Memory Block Size and Base	1.0		
31	050B	Get Memory Information	1.0		
31	0600	Lock Linear Region	0.9		
31	0601	Unlock Linear Region	0.9		
31	0602	Mark Real Mode Region as Pageable	0.9		
31	0603	Relock Real Mode Region	0.9		
31	0604	Get Page Size	0.9		
31	0702	Mark Page as Demand Paging Candidate	0.9		
31	0703	Discard Page Contents	0.9		
31	0800	Physical Address Mapping	0.9		
31	0801	Free Physical Address Mapping	1.0		
31	0900	Get and Disable Virtual Interrupt State	0.9		
31	0901	Get and Enable Virtual Interrupt State	0.9		

Interrupt	AX	Description	Version	32-bit host	16-bit host
31	<a href="#">0902</a>	Get Virtual Interrupt State	0.9		
31	<a href="#">0A00</a>	Get Vendor-Specific API Entry Point	0.9		
31	<a href="#">0B00</a>	Set Debug Watchpoint	0.9		
31	<a href="#">0B01</a>	Clear Debug Watchpoint	0.9		
31	<a href="#">0B02</a>	Get State of Debug Watchpoint	0.9		
31	<a href="#">0B03</a>	Reset Debug Watchpoint	0.9		
31	<a href="#">0C00</a>	Install Resident Service Provider Callback	1.0		
31	<a href="#">0C01</a>	Terminate and Stay Resident	1.0		
31	<a href="#">0D00</a>	Allocate Shared Memory	1.0		
31	<a href="#">0D01</a>	Free Shared Memory	1.0		
31	<a href="#">0D02</a>	Serialize on Shared Memory	1.0		
31	<a href="#">0D03</a>	Free Serialization on Shared Memory	1.0		
31	<a href="#">0E00</a>	Get Coprocessor Status	1.0		
31	<a href="#">0E01</a>	Set Coprocessor Emulation	1.0		

From:

<https://www.osfree.org/doku/> - **osFree wiki**

Permanent link:

<https://www.osfree.org/doku/doku.php?id=en:docs:dpmi&rev=1662044678>

Last update: **2022/09/01 15:04**

