2025/11/04 11:36 1/2 DOSKRNL

# **DOSKRNL**

DOSKRNL is a specialized DOS kernel changed to support DOS in MVM.

## **Startup**

On DOSKRNL start SS:BP contains some structure which contains initial information for DOS kernel. Exact structure format not know. SS:SP contains stack pointer (initially point to area after init structure, size around 400 bytes)

Offset	Size	Description
0	2	First free segment after DOSKRNL
2	2	Size of memory - first free segment (paragraphs)
4	2	Size of init area (paragraphs)
6	2	Value of BREAK setting
8	2	Value of DOS setting
10	4	Far pointer to list of DOS DEVICE setting
14	4	Far pointer to SHELL (filepath only)
18	4	Far pointer to SHELL (arguments)
22	4	Pointer to linked list of VDD
25	1	Current drive (0-A, 1-B,)
26	1	Boot drive (0-A, 1-B,)
????	???	????

Memory map on DOSKRNL start:

Init area	[BP+0+BP+2-BP+4]:0
Free	[BP+0]:0
DOSKRNL	60:0
CMOS Data	40:0
Interrupt vectors	0:0

As in most DOS kernels, DOSKRNL moves initialization code to higher memory. But uses init structure, not BIOS INT 12h, for memory information.

VDDs linked list (in init area) in standard DOS Device drivers format

offset	Size	Description		
0	4	Pointer to next device (Must be set to -1 for last device)		
		Attributes		
		Bit 15 = 1 if char device 0 if blk		
		if bit 15 is 1		
		Bit 0 = 1 if Current sti device		
1		Bit 1 = 1 if Current sto output		
4		Bit 2 = 1 if Current NUL device		
		Bit 3 = 1 if Current CLOCK dev		
osFree wiki - https://ftp.osfree.org/doku/  Bit 4 = 1 if SPECIAL				
		Bit 14 is the IOCTL bit		
		Rit 12 is the NON IRM FORMAT hit		

offset	Size	Description		
6	2	Pointer to Device strategy entry point		
8	2	Pointer to Device interrupt entry point		
10		character device name field. Character devices set a device name. For block devices the first byte is the number of units		

DOSKRNL search and initialize XMS driver and moves some parts to HMA. Most other initialization things is same. DOSKRNL initializes standard device drivers and adds and initializes VDDs, passed via init structure. Some things (at least COMMAND.COM, device drivers) also passed via init structure. not CONFIG.SYS. At the present time interface not investigated well.

### **CONFIG.SYS**

CONFIG.SYS device information and some other settings not parsed by DOSKRNL, but passed via init structure. Here list of DOSKRNL options:

- RMSIZE
- DEVICE
- SHELL
- DOS
- BREAK

todo add more info here

Also most of these setting can be adjusted via DOS Properties (aka DOS Settings). Refer OS/2 Version 2.0 Volume 2: DOS and Windows Environment for more information.

#### API

According different sources IBM DOSKRNL implements API on level of MS-DOS 5.x. osFree DOSKRNL supports API as in FreeDOS kernel. More information about implemented functions available here. DOSKRNL also implements some extensions to API mostly aimed to communicate with host system.

## **Supervisor calls**

DOSKRNL uses SVC interface to call host functions.

From:

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

Permanent link:

https://ftp.osfree.org/doku/doku.php?id=en:docs:kernel:doskrnl&rev=1701012370

Last update: 2023/11/26 15:26

