TUGAS PRAKTIKUM SISTEM OPERASI

Nama : Andika Wirapala Fathin Azfar Kelas : A

NIM : L200180014 Modul : 3

1. Pemetaan Memori

"Low" memory (< 1MB)

start	end	size	type	description
Low Memory (the first MiB)				
0x00000000	0x000003FF	1 KiB	RAM - partially unusable (see above)	Real Mode IVT (Interrupt Vector Table)
0x00000400	0x000004FF	256 bytes	RAM - partially unusable (see above)	BDA (BIOS data area)
0x00000500	0x00007BFF	almost 30 KiB	RAM (guaranteed free for use)	Conventional memory
0x00007C00 (typical location)	0x00007DFF	512 bytes	RAM - partially unusable (see above)	Your OS BootSector
0x00007E00	0x0007FFFF	480.5 KiB	RAM (guaranteed free for use)	Conventional memory
0x00080000	0x0009FFFF	128 KiB	RAM - partially unusable (see above)	EBDA (Extended BIOS Data Area)
0x000A0000	0x000FFFFF	384 KiB	various (unusable)	Video memory, ROM Area

BIOS Data Area (BDA)

address (size)	description
0x0400 (4 words)	IO ports for COM1-COM4 serial (each address is 1 word, zero if none)
0x0408 (3 words)	IO ports for LPT1-LPT3 parallel (each address is 1 word, zero if none)
0x040E (word)	EBDA base address >> 4 (**usually!**)

0x0410 (word)	packed bit flags for detected hardware
0x0417 (word)	keyboard state flags
0x041E (32 bytes)	keyboard buffer
0x0449 (byte)	Display Mode
0x044A (word)	number of columns in text mode
0x0463 (2 bytes, taken as a word)	base IO port for video
0x046C (word)	# of IRQ0 timer ticks since boot
0x0475 (byte)	# of hard disk drives detected
0x0480 (word)	keyboard buffer start
0x0482 (word)	keyboard buffer end
0x0497 (byte)	last keyboard LED/Shift key state

Extended BIOS Data Area (EBDA)

You may see "maps" of the EBDA if you search the web. However, those maps are for the original IBM BIOS EBDA. They do not apply to any current EBDA, used by any current BIOS. The EBDA area is not standardized. It **does** contain data that your OS will need, but you must do a bytewise pattern search to find those tables. (See Plug-and-Play.)

ROM Area

start	end	size	region/exception	description
Standard usage of the ROM Area				
0x000A0000	0x000BFFFF	128 KiB	video RAM	VGA display memory
0x000C0000	0x000C7FFF	32 KiB (typically)	ROM	Video BIOS
0x000C8000	0x000EFFFF	160 KiB (typically)	ROMs and unusable space	Mapped hardware & Misc.
0x000F0000	0x000FFFFF	64 KiB	ROM	Motherboard BIOS

“Upper Memory” (>1MB)

start	end	size	region/exception	description
High Memory				
0x00100000	0x00EFFFFF	0x00E00000 (14 MiB)	RAM -- free for use (if it exists)	Extended memory [1, 2]
0x00F00000	0x00FFFFFF	0x00100000 (1 MiB)	Possible memory mapped hardware	ISA Memory Hole 15-16MB [3]
0x01000000	????????	???????? (whatever exists)	RAM -- free for use	More Extended memory [1]
0xC0000000 (sometimes, depends on motherboard and devices)	0xFFFFFFFF	0x40000000 (1 GiB)	various (typically reserved for memory mapped devices)	Memory mapped PCI devices, PnP NVRAM?, IO APIC/s, local APIC/s, BIOS, ...
0x0000000100000000 (possible memory above 4 GiB)	????????????????	???????????????? (whatever exists)	RAM -- free for use (PAE/64bit)	More Extended memory [1]
????????????????	????????????????	????????????????	Possible memory mapped hardware	Potentially usable for memory mapped PCI devices in modern hardware (but typically not, due to backward compatibility)

2. Perbedaaan real mode dan protect mode

- Real mode: Dalam Real-mode, tidak ada proteksi ruang alamat memori, sehingga tidak dapat melakukan multi-tasking. Inilah sebabnya, mengapa program-program DOS bersifat single-tasking. Jika dalam modus real terdapat multi-tasking, maka kemungkinan besar antara dua program yang sedang berjalan, terjadi tabrakan (crash) antara satu dengan lainnya.
- Protected mode: Modus terproteksi (protected mode) adalah sebuah modus di mana terdapat proteksi ruang alamat
- memori yang ditawarkan oleh mikroprosesor untuk digunakan oleh sistem operasi. Modus ini datang dengan mikroprosesor Intel 80286 atau yang lebih tinggi. Karena memiliki proteksi ruang alamat memori, maka dalam modus ini sistem operasi dapat melakukan multitasking.