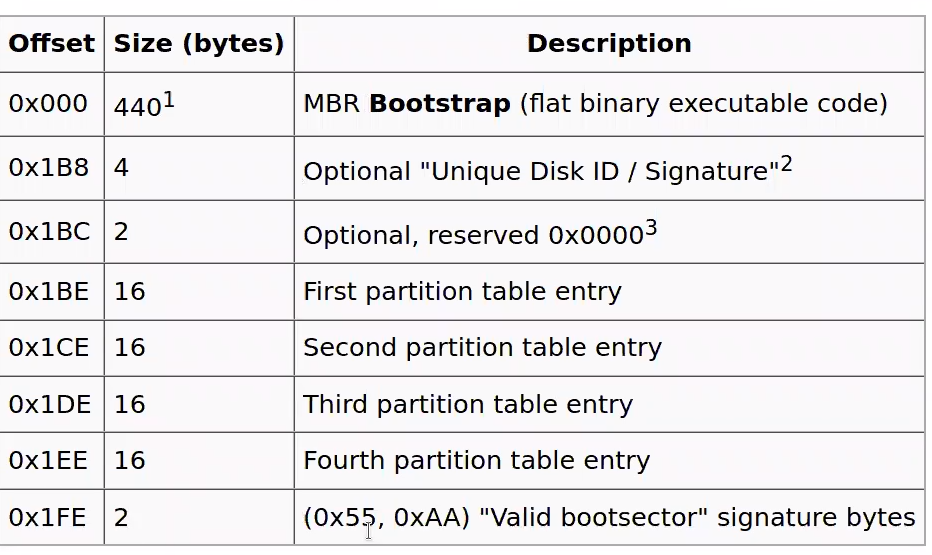
INT 0x15 : Interrupt Call

EAX = 0xE820 is a register

ES:DI is filled with list entries of memory

manually update DI everytime you store a list entry

Bare OS in Master Boot Record aka Real Mode Operating System.



If we look at the 0x1FE, 0x55 in least significant bytes, and 0xAA in most significant bytes identify “Valid..”

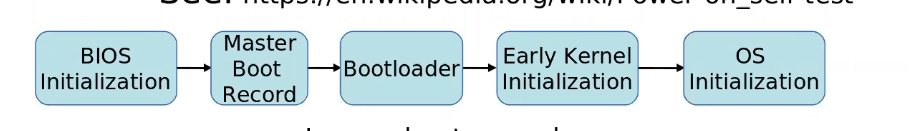
4 valid partitions, starting sector and the length of partition

Reserved bytes, leaves 440 bytes to write master boot record code

Grub stage1: 512 in mBr

Grub stage 1.5: device driver for the file system that you’re asking grub to read to load into the kernel

EXTD4 builds on EXTD2



Real mode

20-bit addressability ( <1MB RAM)

16-bit registers:

Main – AccumulateX, BaseX, CountX, DisplacementX

Index, StackP, BaseP, Source Index and Destination Index

Segment – CS,DS,SS,ES

Instruction pointer – IP

Main registers have 8-bit subfields shown in brackets:

AX (AH, AL)

BX (BH, BL)

CX (CH, CL)

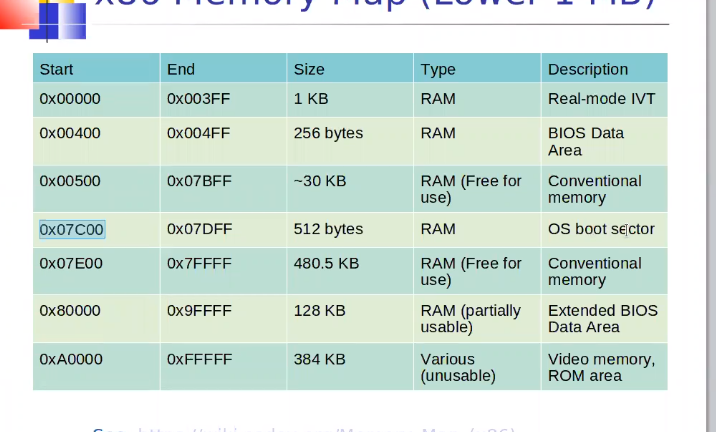
DX (DH,DL)

Segment is a 64-bit memory, with a 16-bit offset

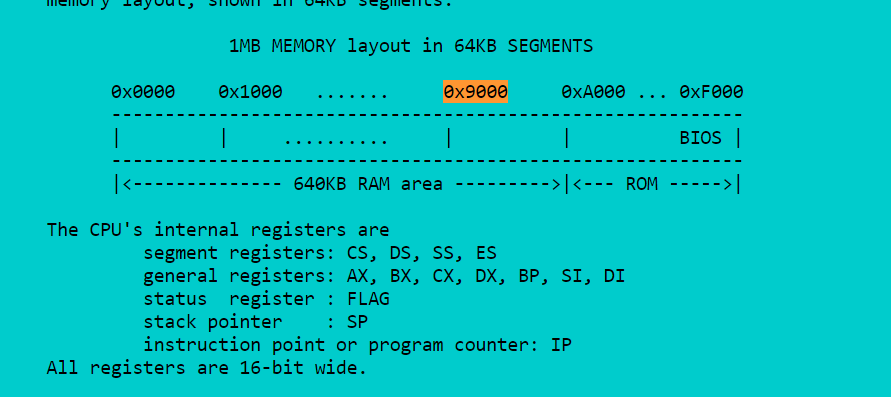
No GDT (hardware-based memory protection

Protected Mode available on newer x86 processors

movw = WORD 4 bytes b = long 8 bytes long bytes



0x07C00 is where it gets loaded into the BIOS



0x30 = 48

0x186a0 = 100000

Sall $

0xE801 = 59393