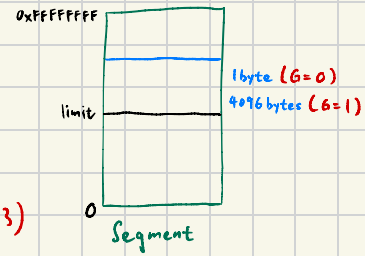
# Segment Descriptor (i386)




## S=1 Application (bit 44)

| bit 11 = 0 → data | bit 11 = 1 → code |
|---|---|
| bit 8 = Accessed | bit 8 = Accessed |
| bit 9 = Write-enable | bit 9 = Read-enable |
| bit 10 = Expand-direction | bit 10 = Conforming |

(bit 40 ~ 43)

CPL is same as current DPL of CS other than transfer controls into Conforming with its own.

DPL of data segments, call gates, and TSSs mean the lowest privilege of processor to get accesses.

DPL of code segments mean the highest privilege of processor to transfer controls.

DPL should be same as CPL without going from a call gate.

Whenever segments accessed, the lower privilege is chosen either CPL or RPL from a selector which may be received by some interface invoked by somewhere other segments would have differ privilege.

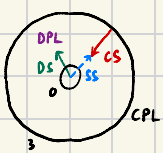
SS should be same as CPL whenever stack switch occurs.

Using Call gates, $\max\{CPL, RPL\} \le$ gate DPL

CALL or JMP when Conforming, callee DPL $\le$ CPL

JMP when Non-Conforming, callee DPL = CPL

RET, caller DPL $\ge$ CPL




## S=0 System (bit 44)

LDT = 0010 Task Gate = 0101 (bit 11~8)

bit 11 = 0 → 16-bit bit 11 = 1 → 32-bit

TSS(Available) = 001

TSS(Busy) = 011

Call Gate = 100 (bit 10~8)

Interrupt Gate = 110

Trap Gate = 111


direct call without stack switch