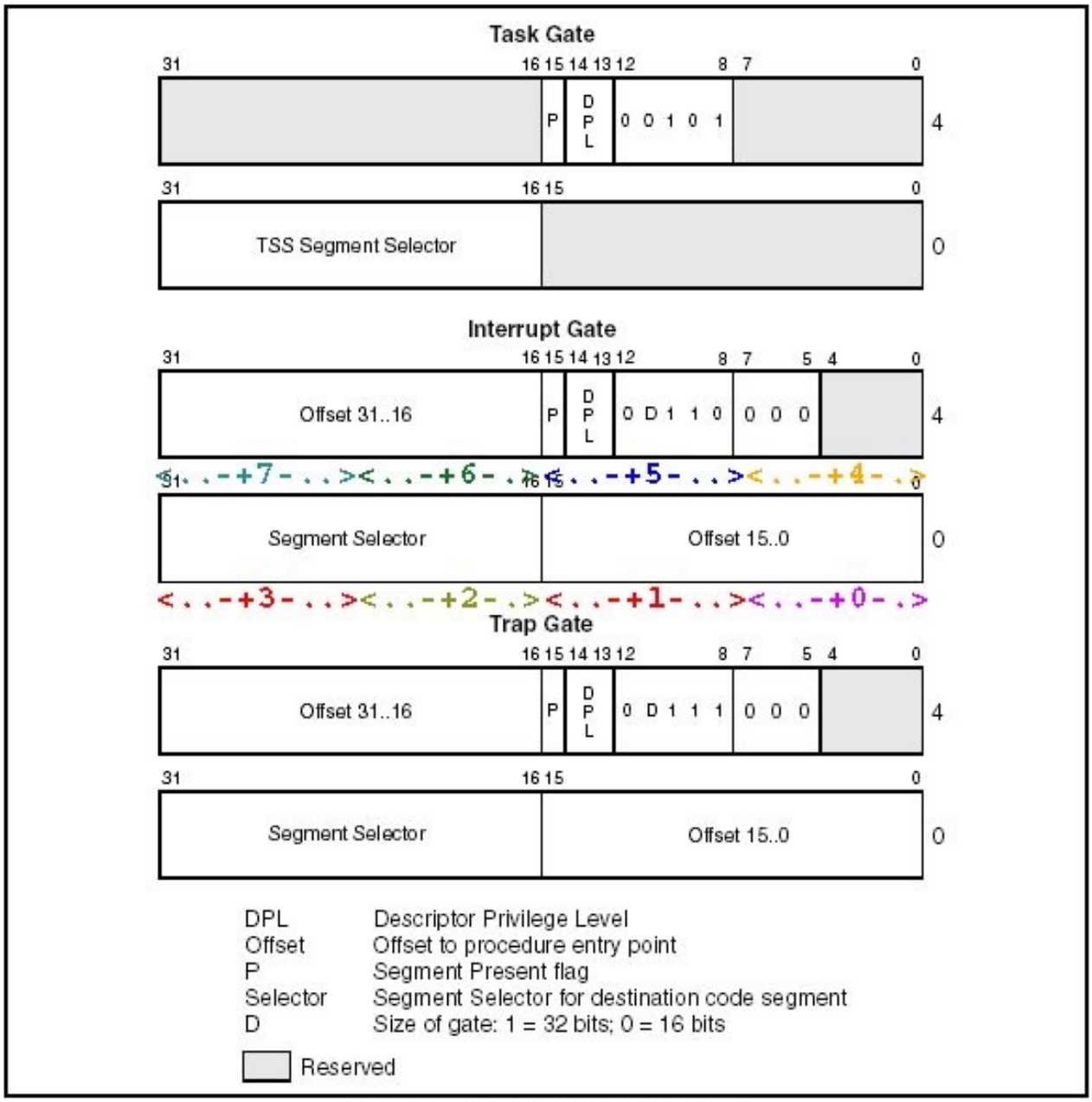

Figure 5-2. IDT Gate Descriptors

Interrupt Gates

SIDT - Retrieve idt base

```
sidt fword ptr [the_idt] ; <Size><Base>
```

each descriptor is 8bytes i.e. 2 dwords, so to find INT1,

```
mov eax,[the_idt+2]
add ebx,1*8
```

the address that handles the interrupt is split over the 2 dwords, as the diagram shows. to get the current address,

```
mov dx,word ptr [ebx+6] ; get first word
shl edx,10h ; mov the first word to the start of edx
mov dx,word ptr [ebx] ; get lower word, EDX now contains the address
```

overwritting a new address should be simple enough, remember to save the old one so you can restore or dasiy chain your interrupt.

DPL is the Descriptor Priveldge Level, this is the ring level the interupt can be called from, 0 2 3 etc

To get the dpl we'll read the 5th byte, but this also contains the present and Interrupt gate marker, so

```
mov byte ptr [ebx+5],dpl
```

32bit present interrupt gates have the value of, 08Eh

```
1 00  01110
P DPL 32bit INT
```

The DPL is just 2 bits of this byte, so we have a max value of 3, simple set it and replace the byte at +5

```
DPL 0 = 8Eh
DPL 1 = AEh
DPL 2 = CEh
DPL 3 = EEh
```

Thats the basics of an IDT Entry,

yates.
01/DEC/02