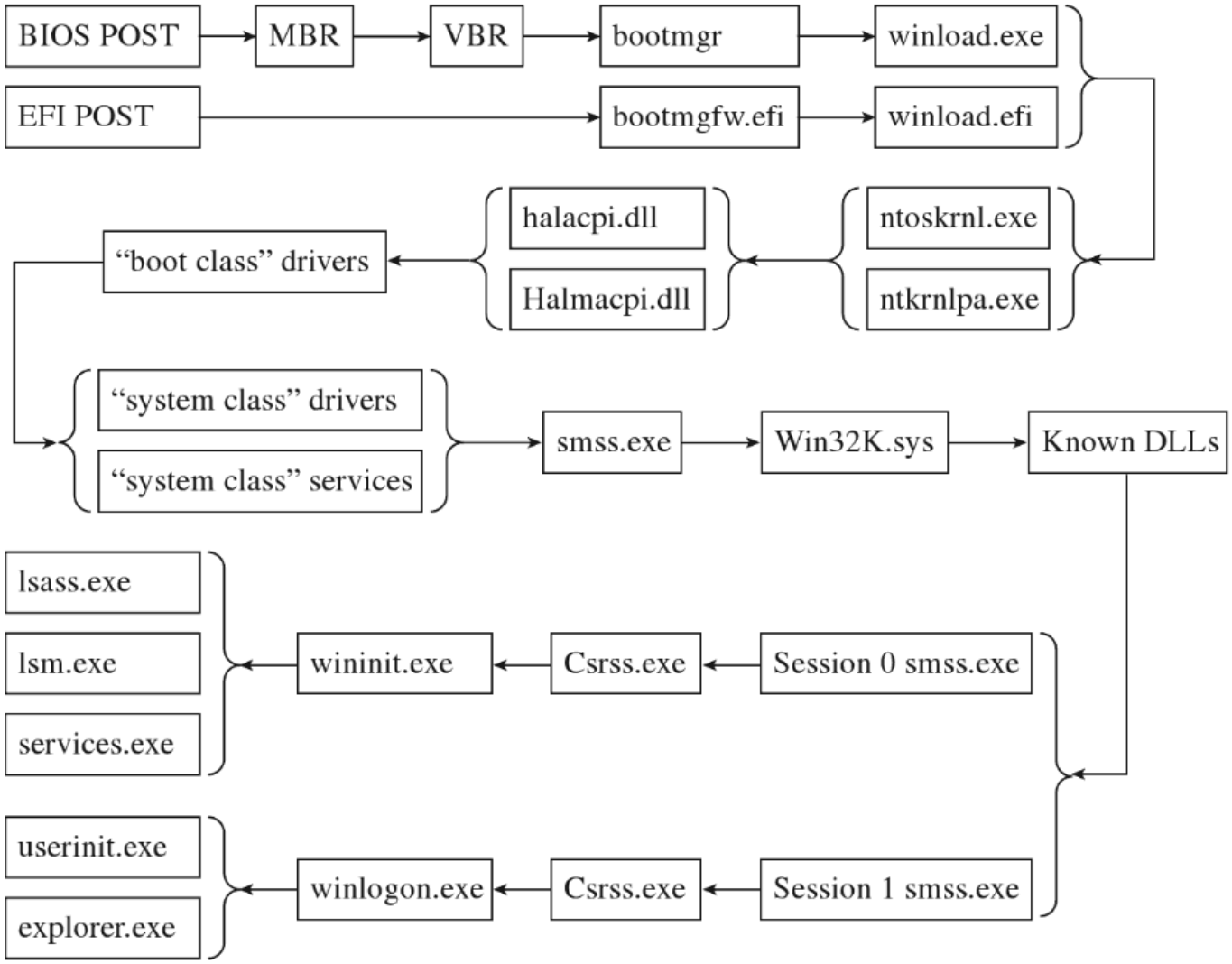
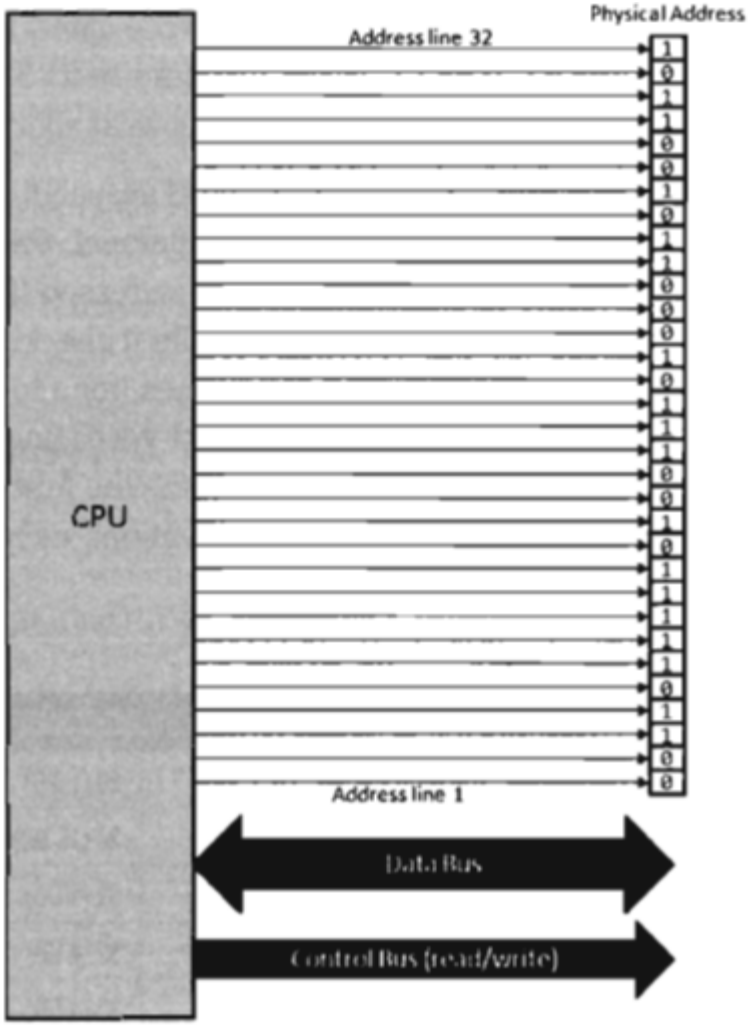
Proces uruchamiania SO Win7 (Boot Process)



Od czego więc zacząć ?



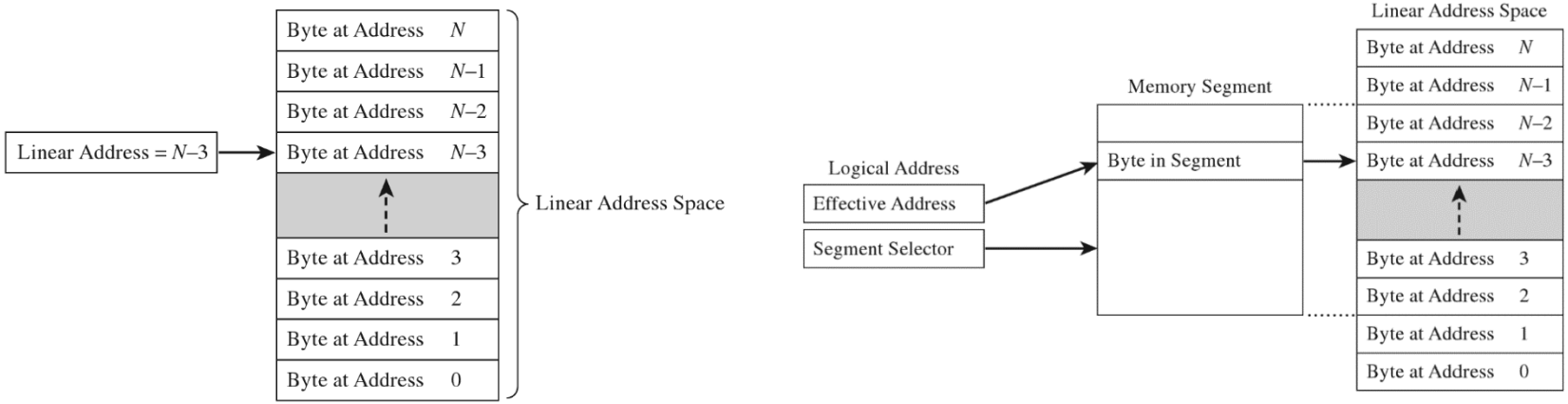
- Poznanie sprzętu, na którym działa SO:
  - Jaka jest architektura procesora?
  - jakie są obszary, w których procesor wspiera twórcę SO?

- Istnieją dwa modele pamięci fizyczny (rzeczywisty, sprzętowy) i liniowy (abstrakcyjny, widziany powyżej warstwy sprzętowej).
- PAE (Physical Address Extension) – adresacja 36 bitowa (do 52 bit)
- Sprawdzenie MAXPHYADDR zwrócony przez CPUID (funkcja 80000008h)

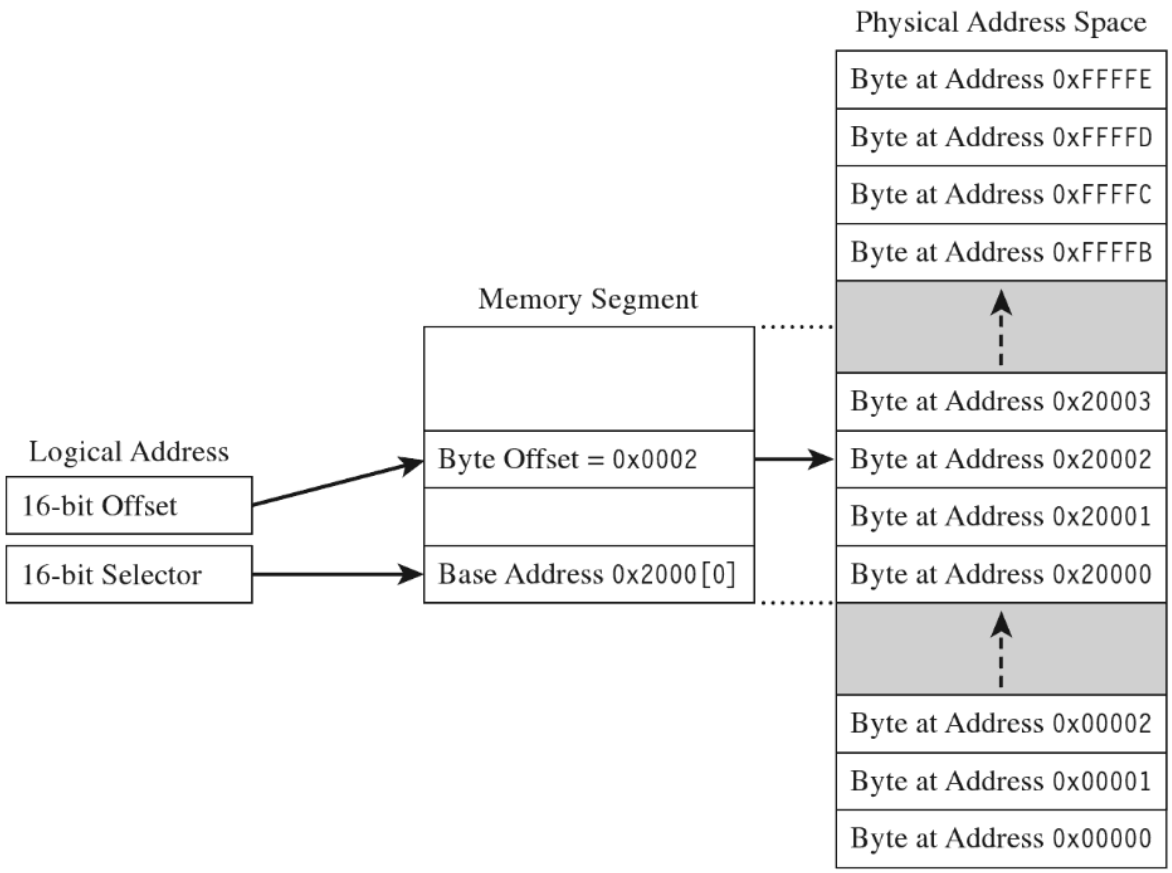
Płaski i segmentowy model pamięci

Adres fizyczny – adres wskazujący na położenie komórki w przestrzeni pamięci fizycznej (adres musi wskazywać na istniejącą komórkę)
Adres liniowy – adres wskazujący na położenie komórki pamięci w przestrzeni adresów liniowych (wirtualnych)

Adres logiczny – adres komórki w przestrzeni adresów liniowych jest ustalany na podstawie pewnego zestawu identyfikatorów (np. selektora i offsetu)

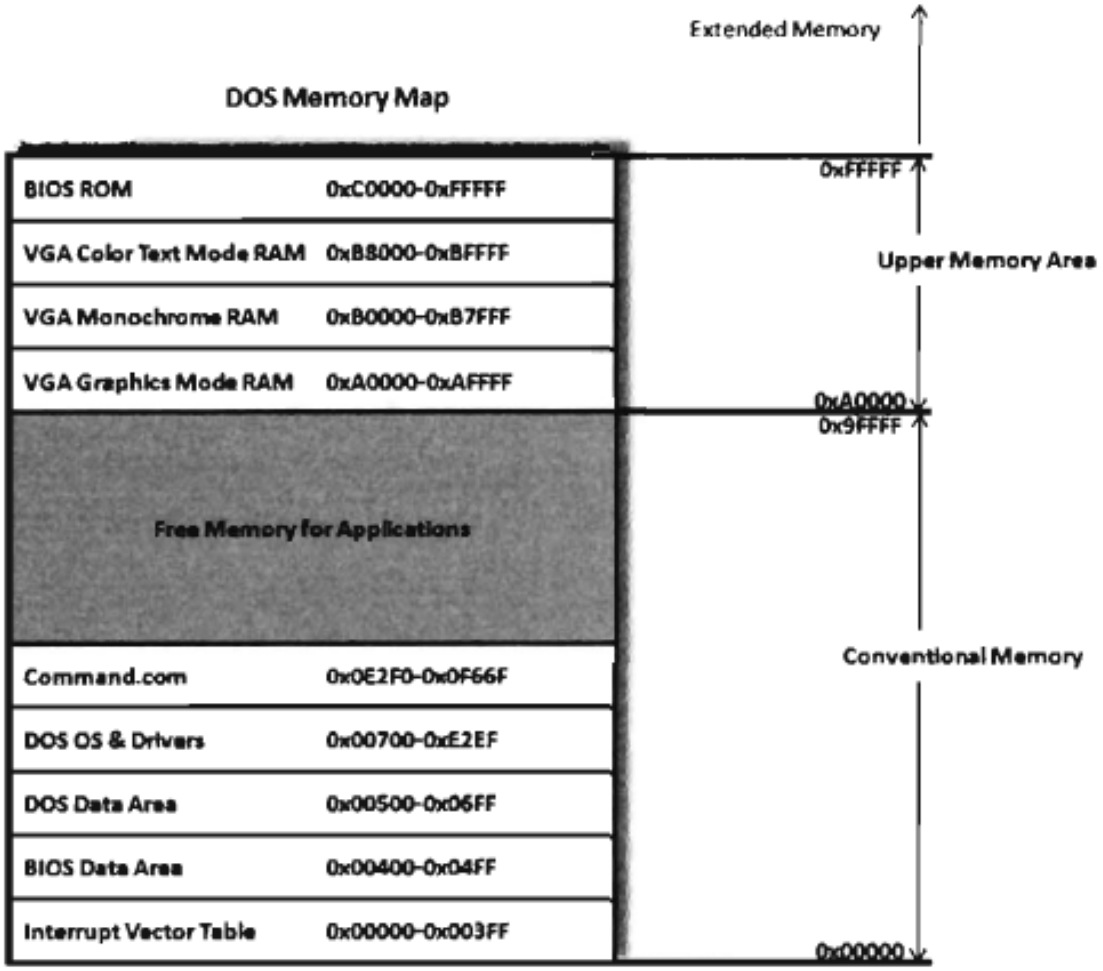


Tryby pracy procesora



- Tryb rzeczywisty (real mode) - używa segmentacji
- Tryb chroniony (protected mode)
- System management mode (SMM) – wywoływanie specjalnego kod umieszczonego w firmware – shutdown - BSDaemon, "System Management Mode Hacks," Phrack, Volume 12, Issue 65.

DOS (Memory Map/Layout)

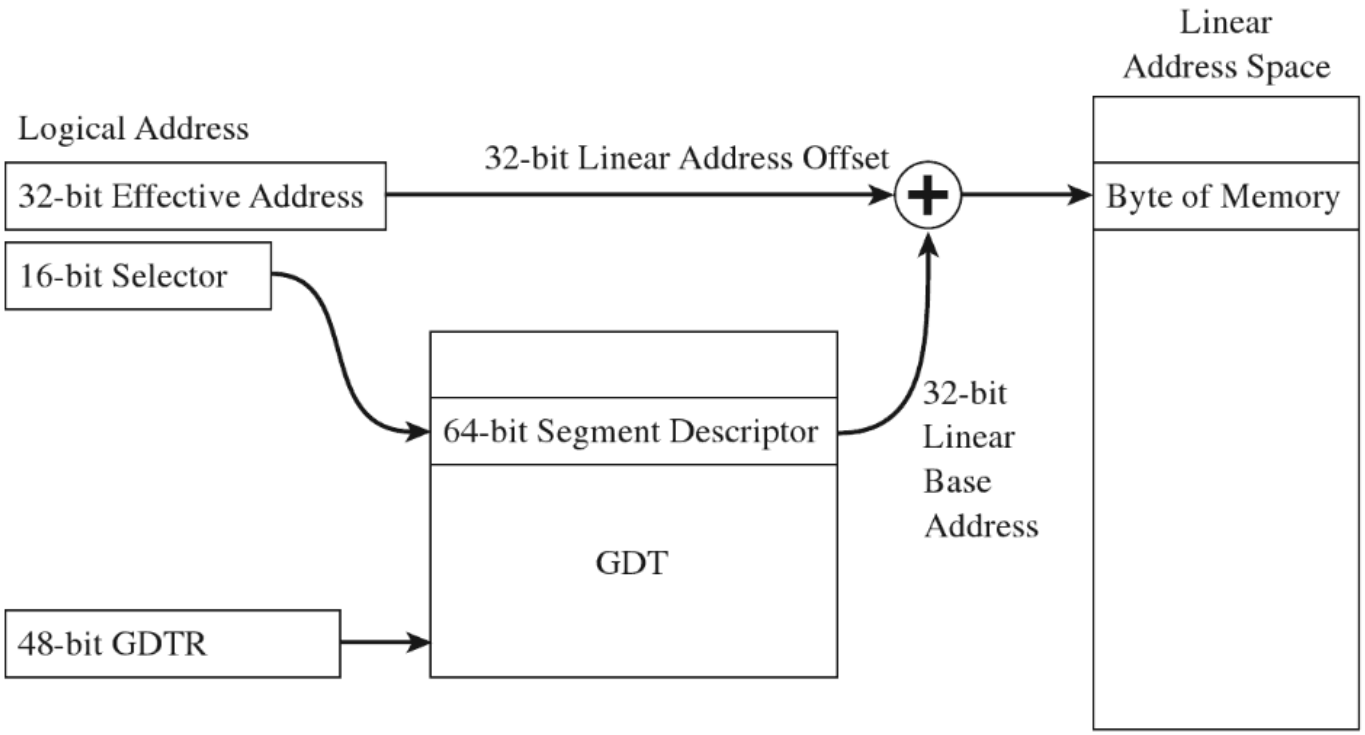


- BIOS
- Tryb rzeczywisty jest podstawą przejścia w tryb chroniony

Wady trybu rzeczywistego

- Modyfikowanie adresów funkcji systemowych (hooking)
- Zdolność istniejących sterowników do przejęcia dowolnej informacji
- Manipulowanie strukturami danych systemowych do ukrycia procesu
- Modyfikacja programów wykonywalnych w celu zmiany przepływu sterowania

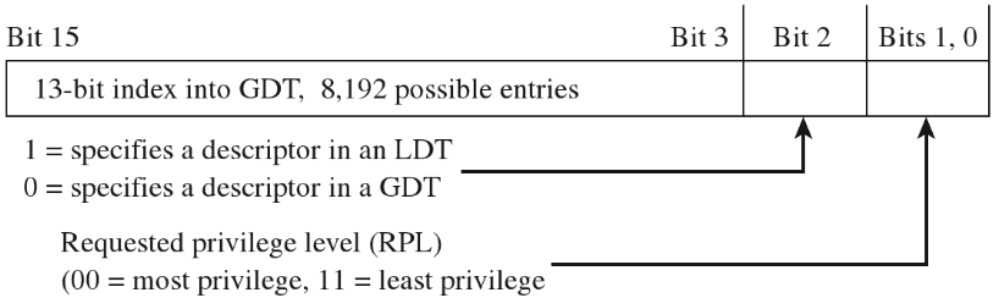
Tryby chroniony z segmentacją (bez stronicowania)



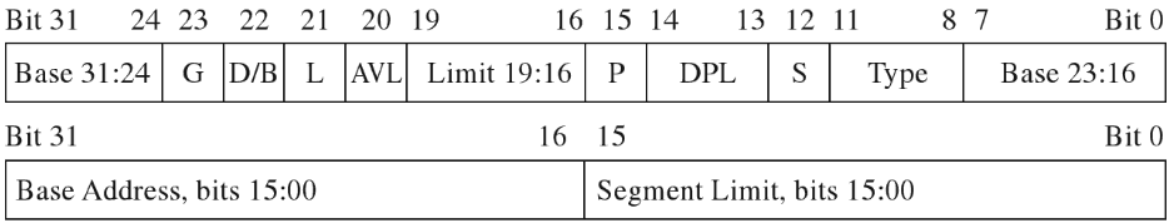
Segmentacja jest obowiązkowa w x86!

Selektor i deskryptor

16-bit Segment Selector

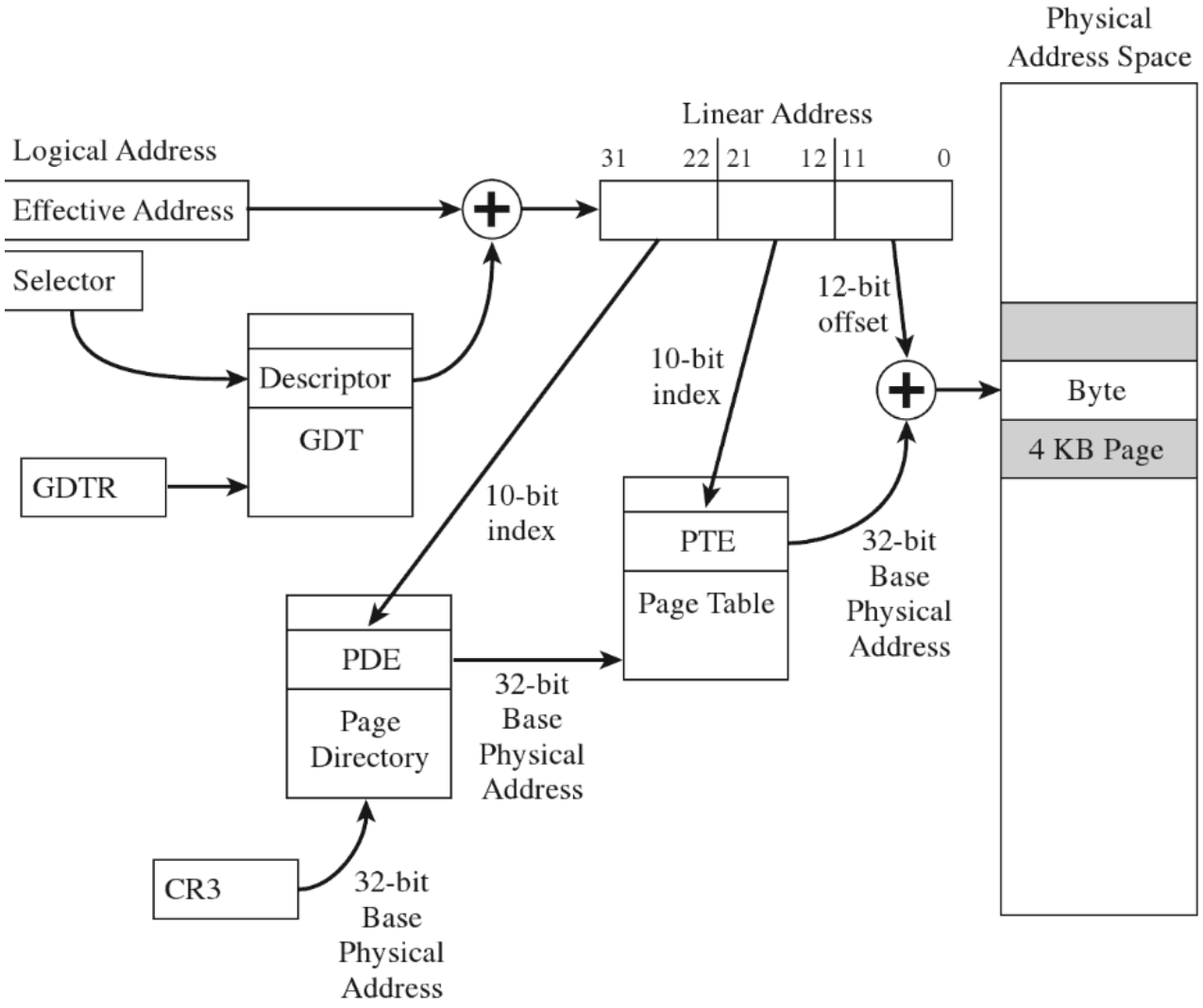


64-bit Segment Descriptor

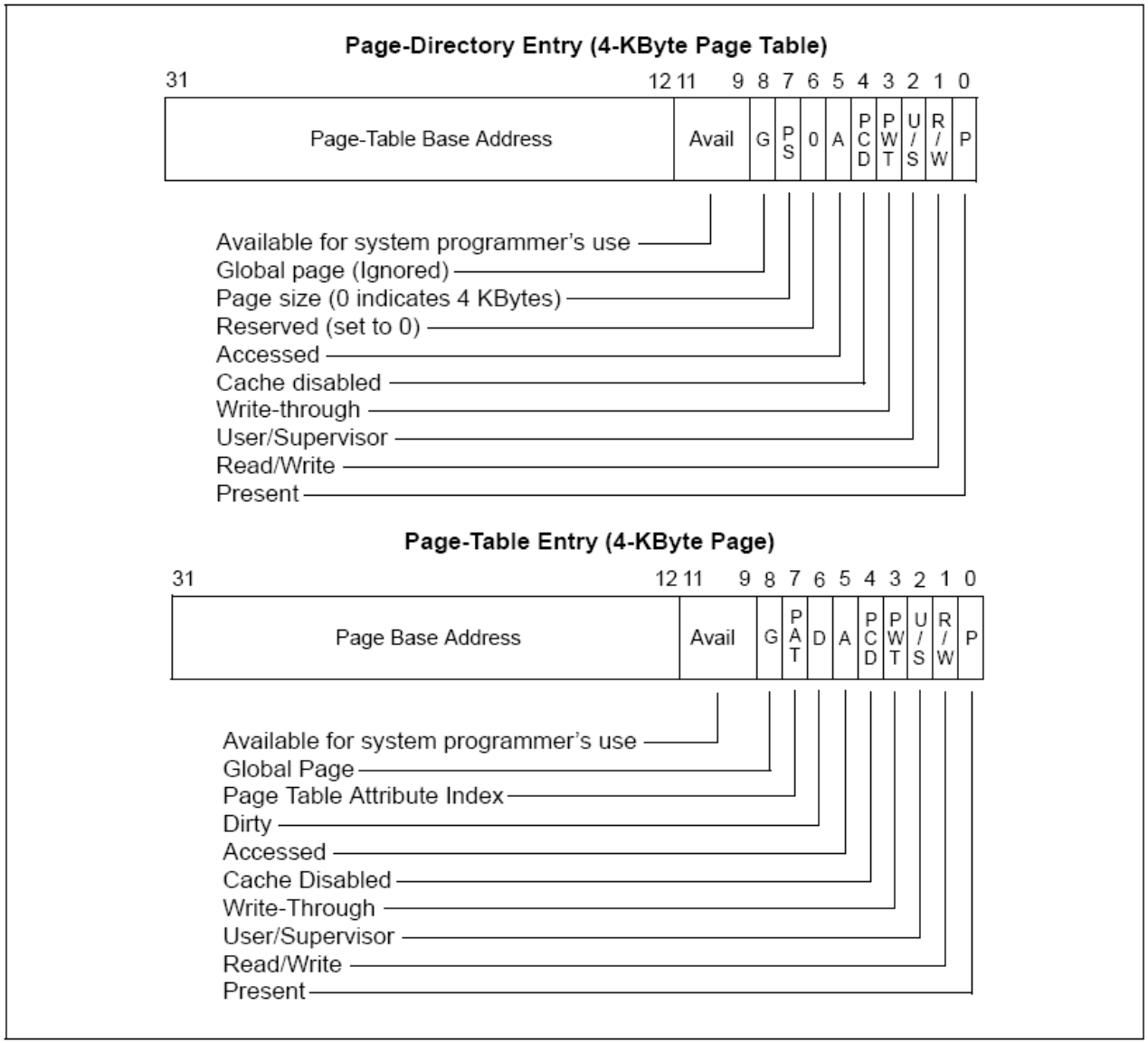


Segment Limit (20-bits)	Segment size (if G = 0: 1 byte – 1 MB, if G = 1: 4 KB to 4 GB)
Base Address (32-bits)	Base linear address used to form the final linear address
Type Field	Type of segment (code or data), access, and growth direction
S Flag	If S is clear, system segment. If S is set, application segment
DPL	Descriptor privilege level (00 = Ring 0, 11 = Ring 3)
P Flag	If P is set, segment is resident in memory
AVL	No explicit purpose, available for use by operating system
L Flag	IA-32 processors set this bit to zero (indicates 64-bit code)
D/B	Meaning varies according to segment type (code, data, or stack)
G Flag	See description of segment limit field

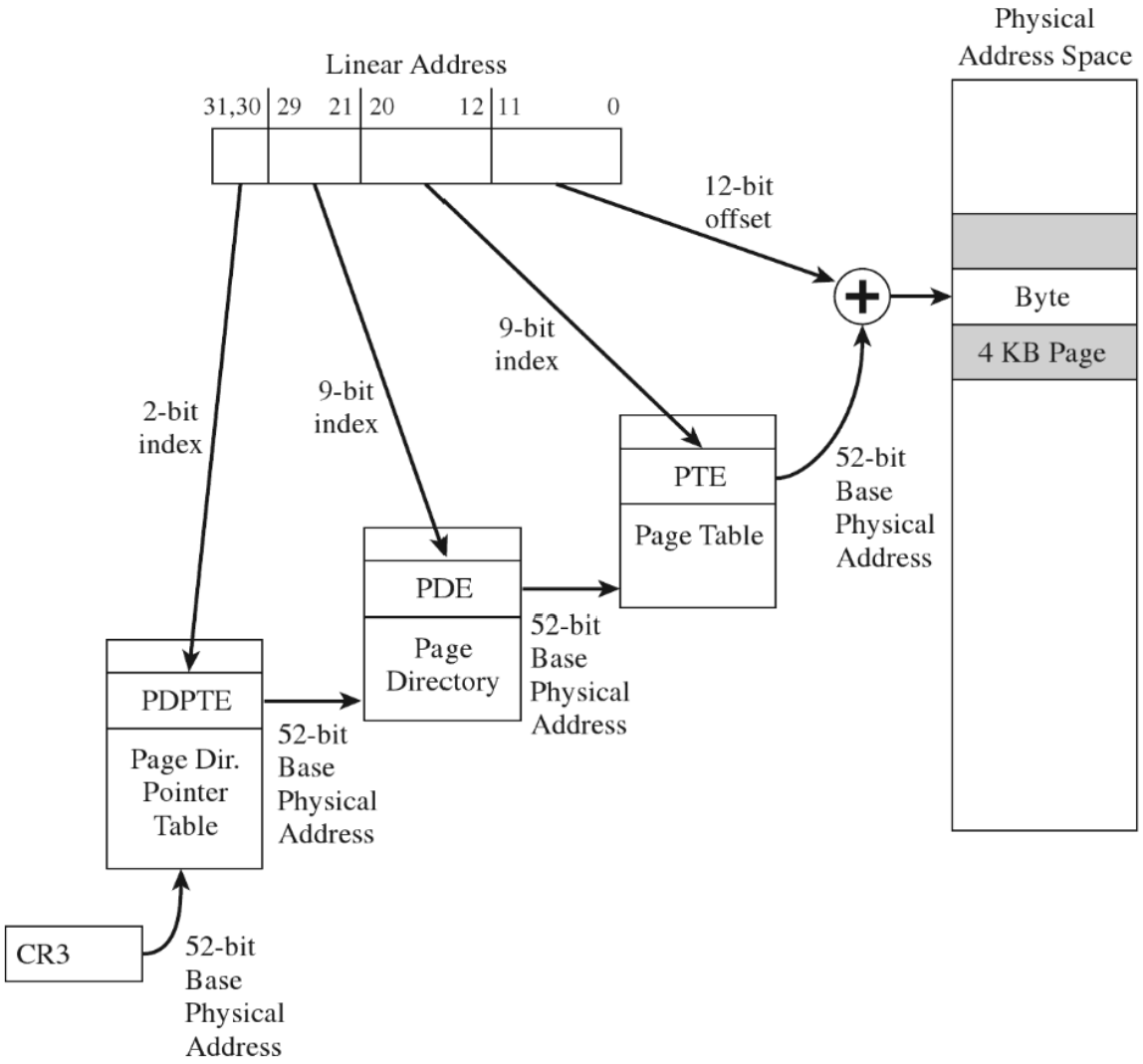
Tryb chroniony z segmentacją i stronicowaniem



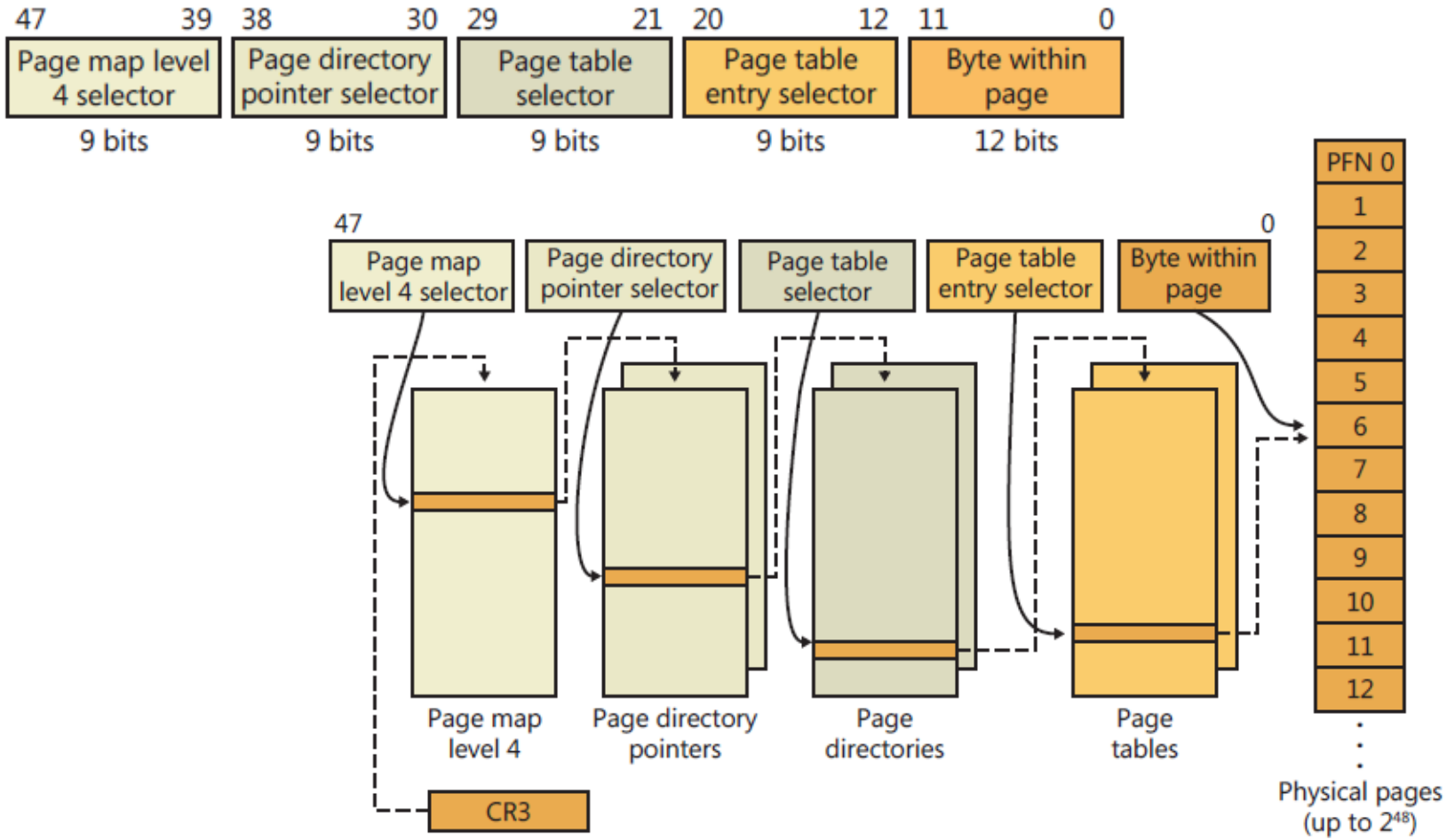
Wpis w PDE i PTE



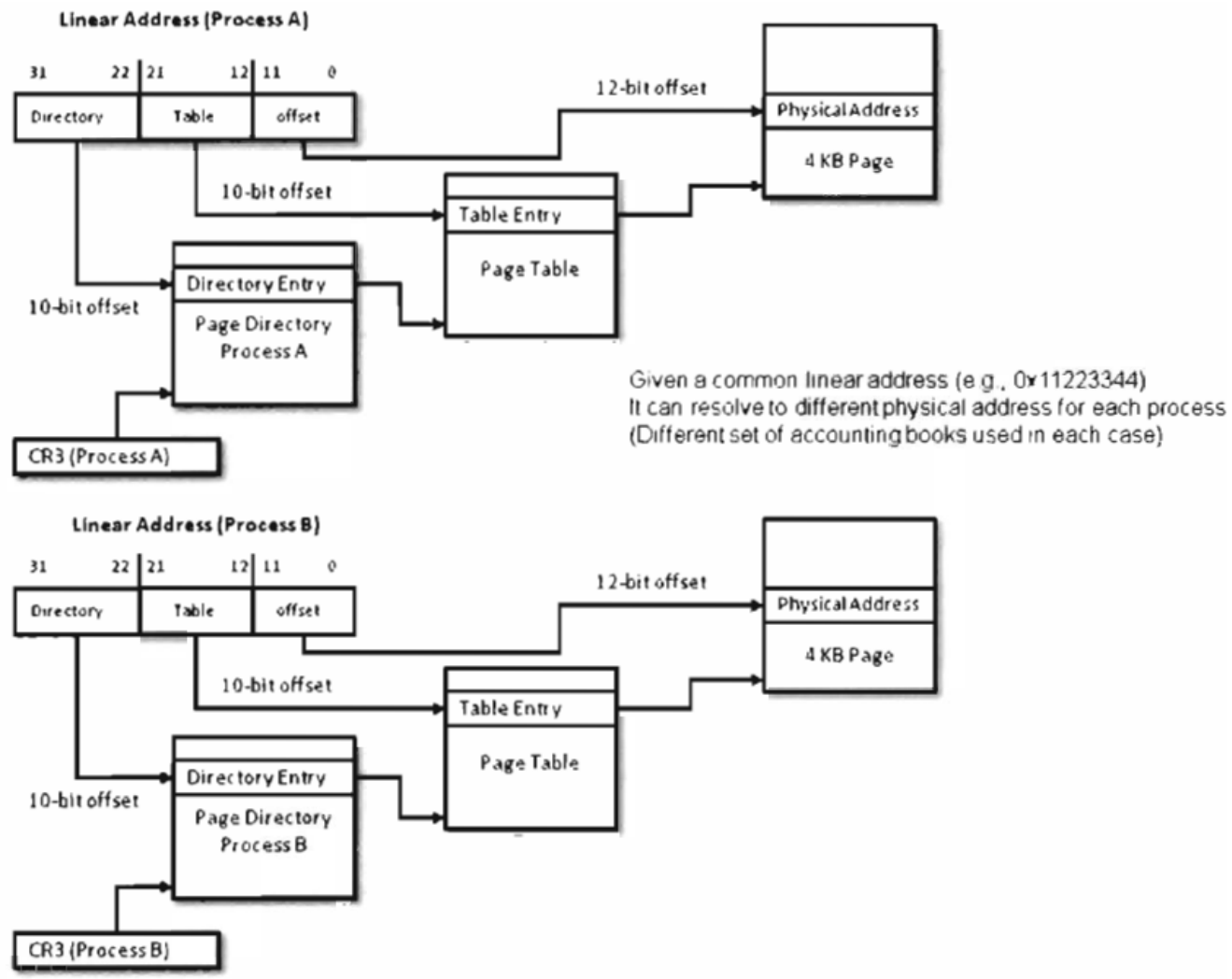
Tryb chroniony z segmentacją i stronicowaniem dla PAE



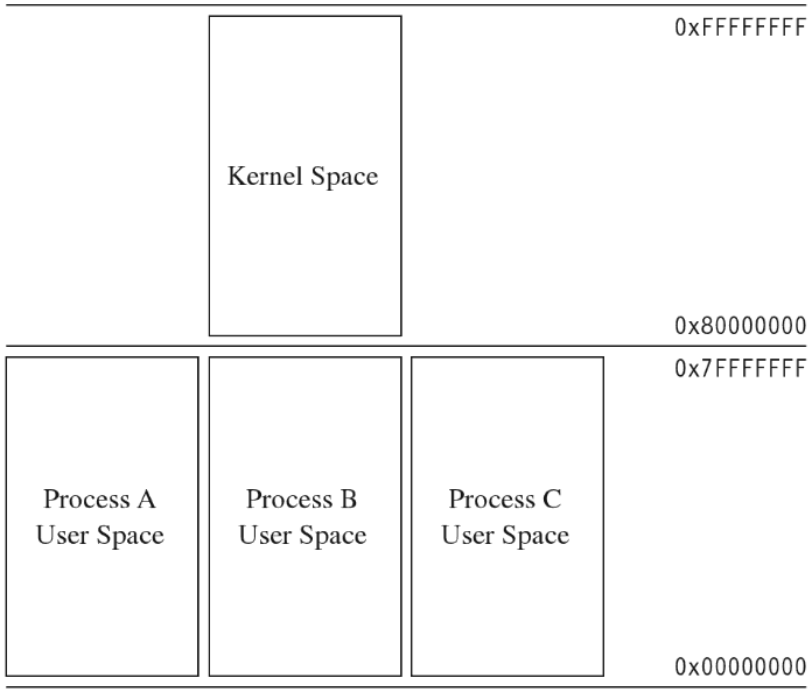
Tryb chroniony z segmentacją i stronicowaniem dla x64

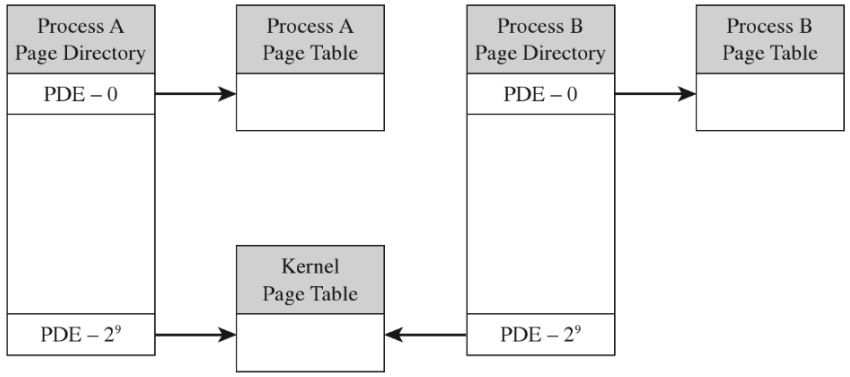


Separacja przestrzeni adresowej

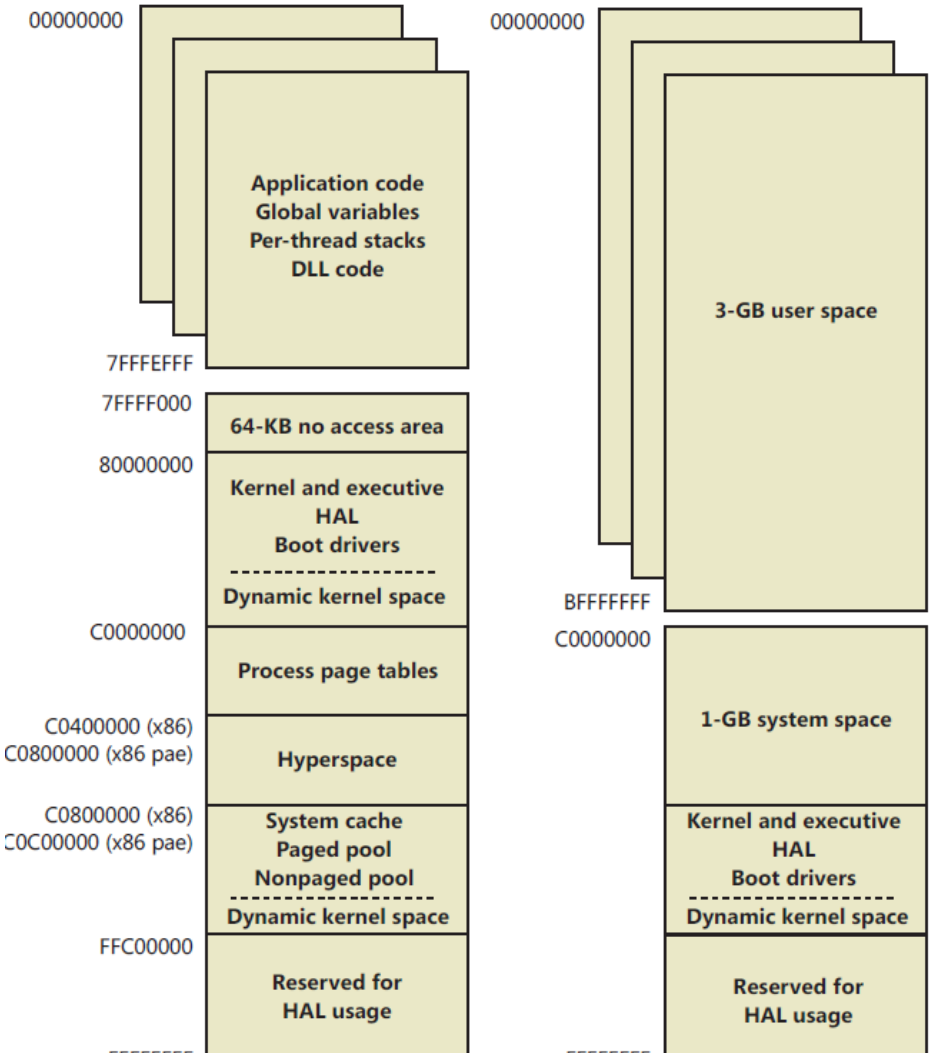


Podział liniowej przestrzeni adresowej





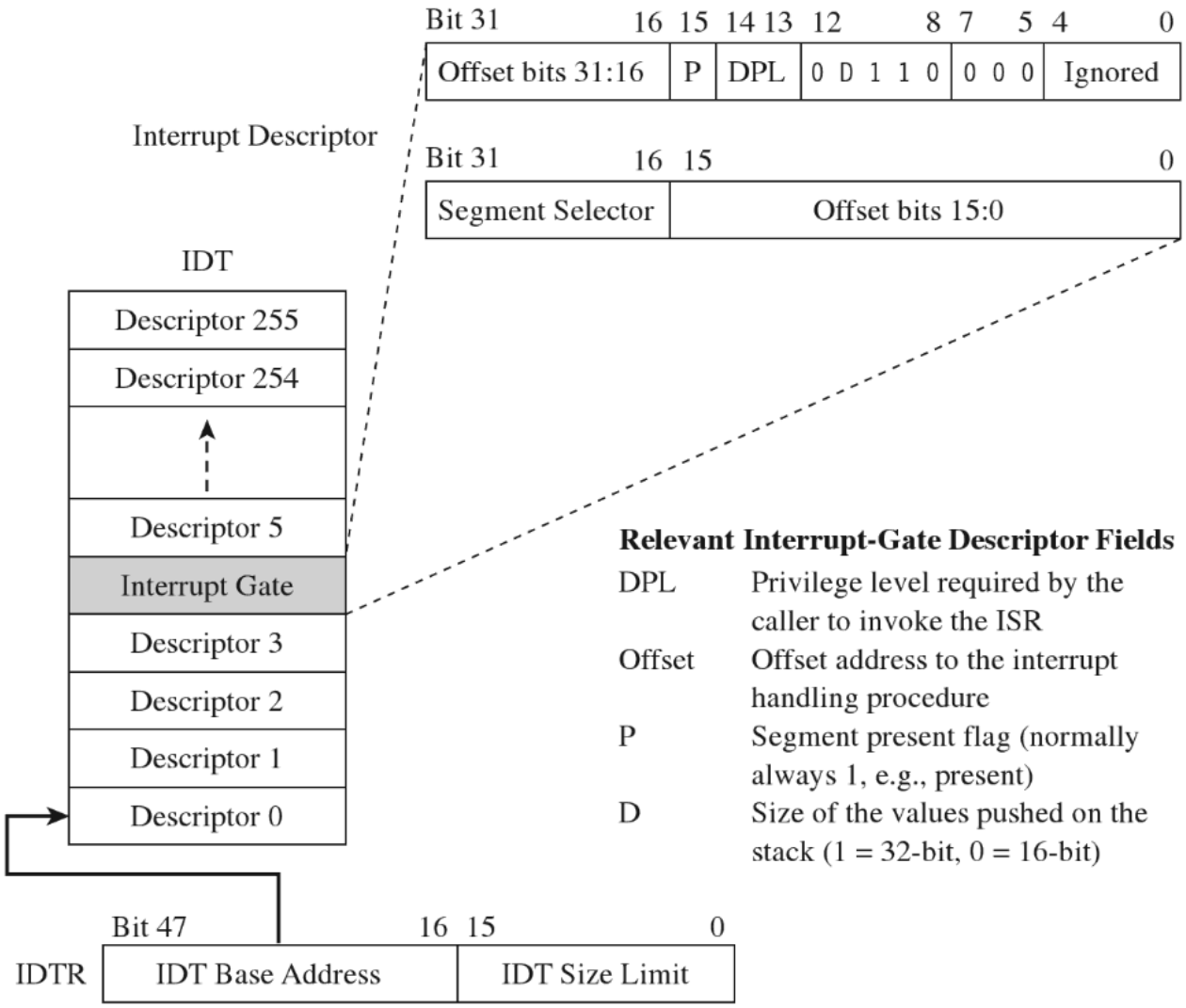
Mapa przestrzeni adresowej w trybie 32-bitowym



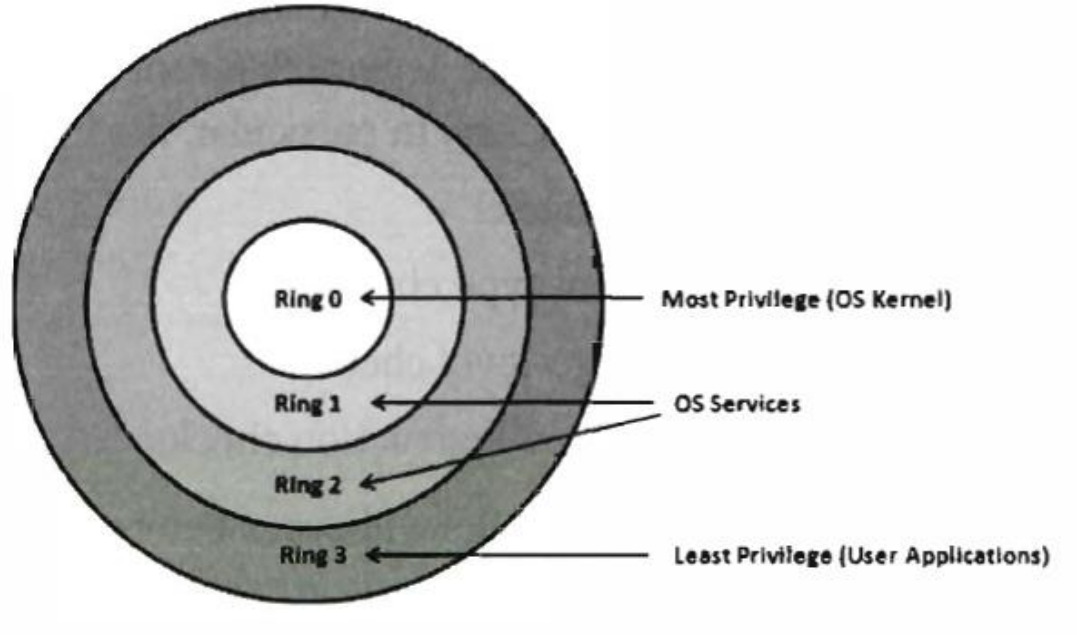
Mapa przestrzeni adresowej w trybie 64-bitowym

Start	End	Size	Description
FFFF0800`00000000	FFFFF67F`FFFFFFFF	238TB	Unused System Space
FFFFF680`00000000	FFFFF6FF`FFFFFFFF	512GB	PTE Space
FFFFF700`00000000	FFFFF77F`FFFFFFFF	512GB	HyperSpace
FFFFF780`00000000	FFFFF780`00000FFF	4K	Shared System Page
FFFFF780`00001000	FFFFF7FF`FFFFFFFF	512GB-4K	System Cache Working Set
FFFFF800`00000000	FFFFF87F`FFFFFFFF	512GB	Initial Loader Mappings
FFFFF880`00000000	FFFFF89F`FFFFFFFF	128GB	Sys PTEs
FFFFF8a0`00000000	FFFFF8bF`FFFFFFFF	128GB	Paged Pool Area
FFFFF900`00000000	FFFFF97F`FFFFFFFF	512GB	Session Space
FFFFF980`00000000	FFFFFa70`FFFFFFFF	1TB	Dynamic Kernel VA Space
FFFFFa80`00000000	*nt!MmNonPagedPoolStart-1	6TB Max	PFN Database
*nt!MmNonPagedPoolStart	*nt!MmNonPagedPoolEnd	512GB Max	Non-Paged Pool
FFFFFFFF`FFc00000	FFFFFFFF`FFFFFFFF	4MB	HAL and Loader Mappings

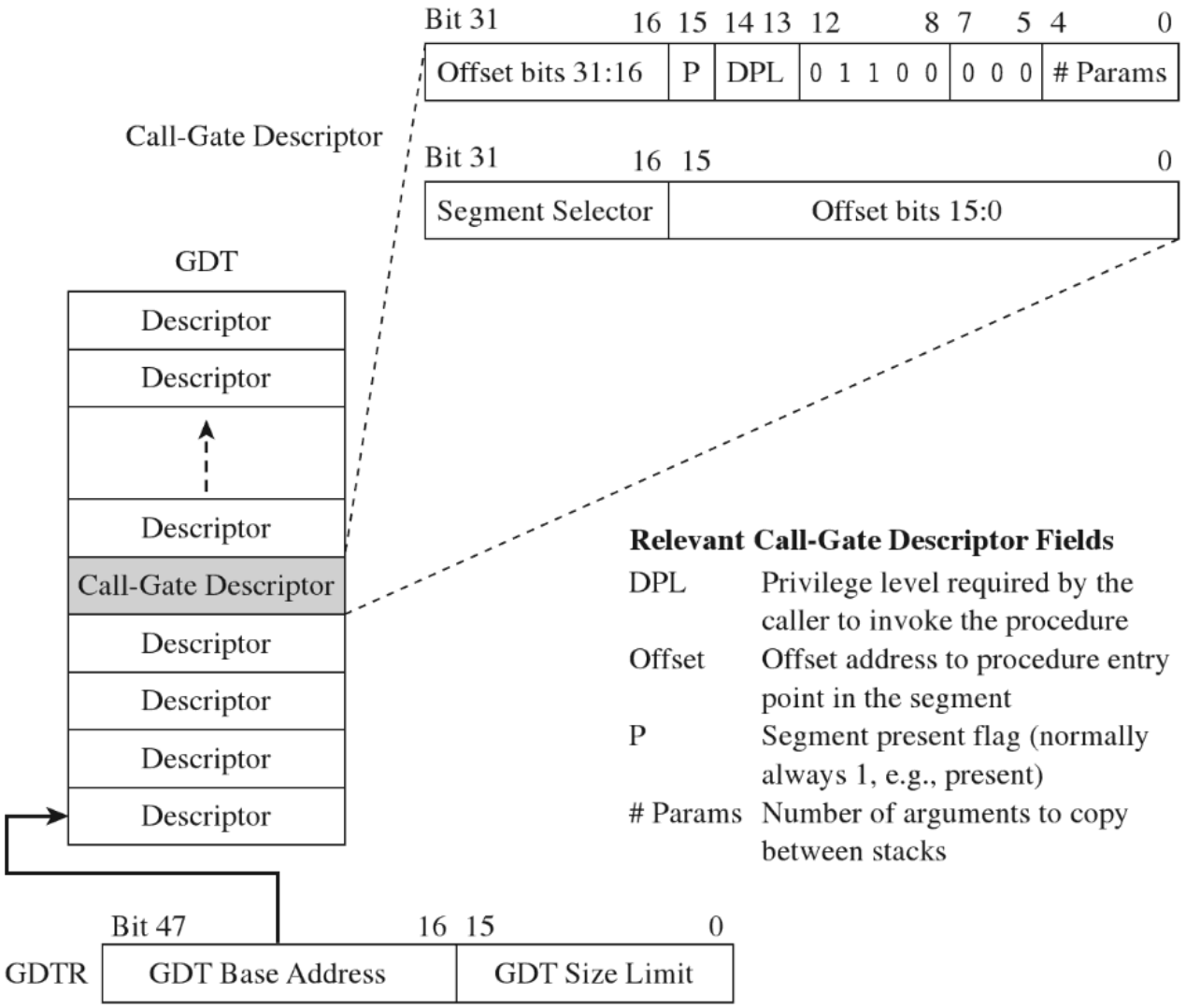
Przerwania


Bit 31
16 15 14 13 12 8 7 5 4 0
Offset bits 31:16
P
DPL
0 D 1 1 0
0 0 0
Ignored
Interrupt Descriptor
Bit 31
16 15 0
Segment Selector
Offset bits 15:0
IDT
Descriptor 255
Descriptor 254
Descriptor 5
Interrupt Gate
Descriptor 3
Descriptor 2
Descriptor 1
Descriptor 0
Relevant Interrupt-Gate Descriptor Fields
DPL Privilege level required by the caller to invoke the ISR
Offset Offset address to the interrupt handling procedure
P Segment present flag (normally always 1, e.g., present)
D Size of the values pushed on the stack (1 = 32-bit, 0 = 16-bit)
Bit 47
16 15 0
IDTR
IDT Base Address
IDT Size Limit

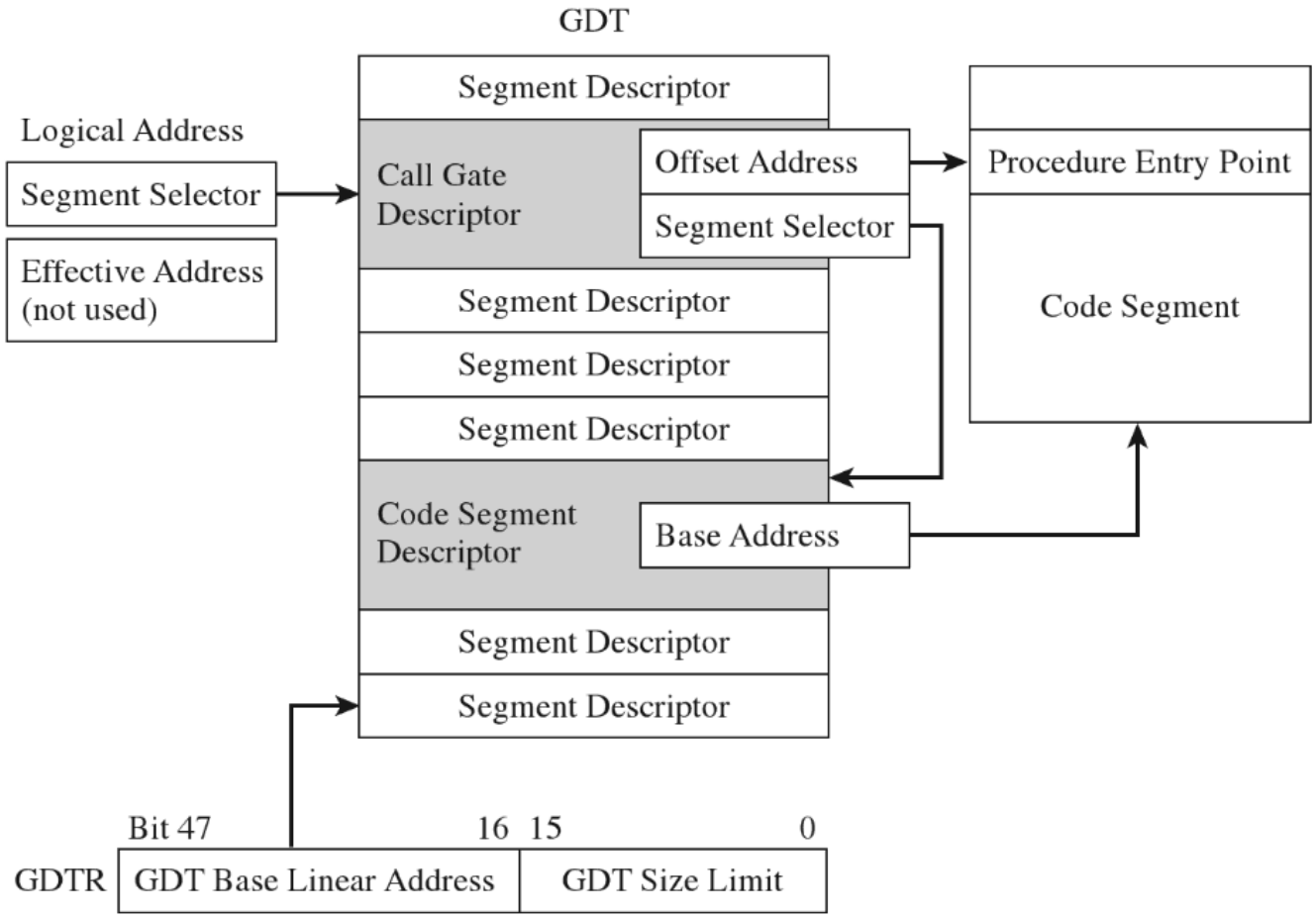
Poziomy uprzywilejowania

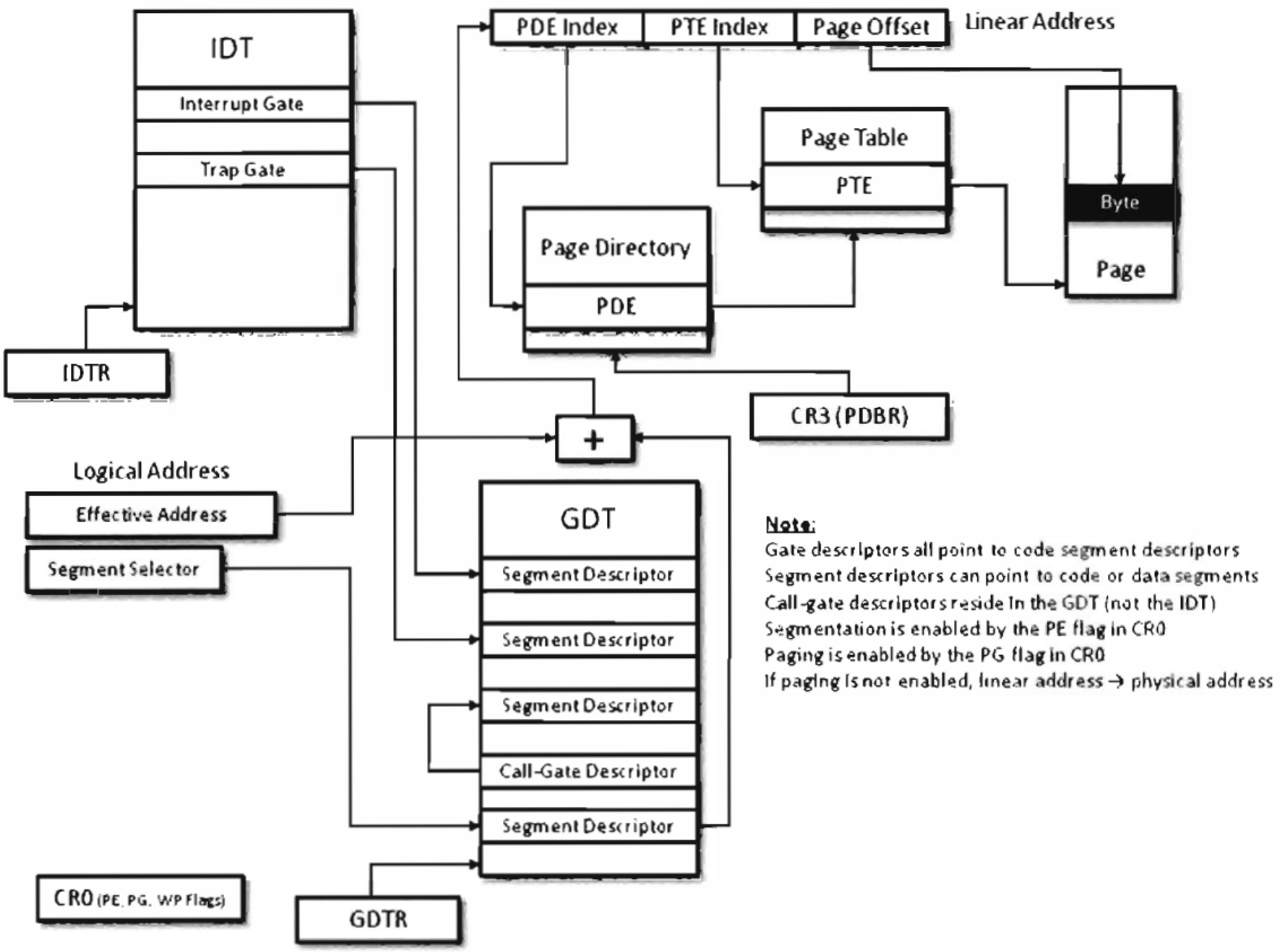


Deskryptory bram



Wywołanie funkcji systemowej

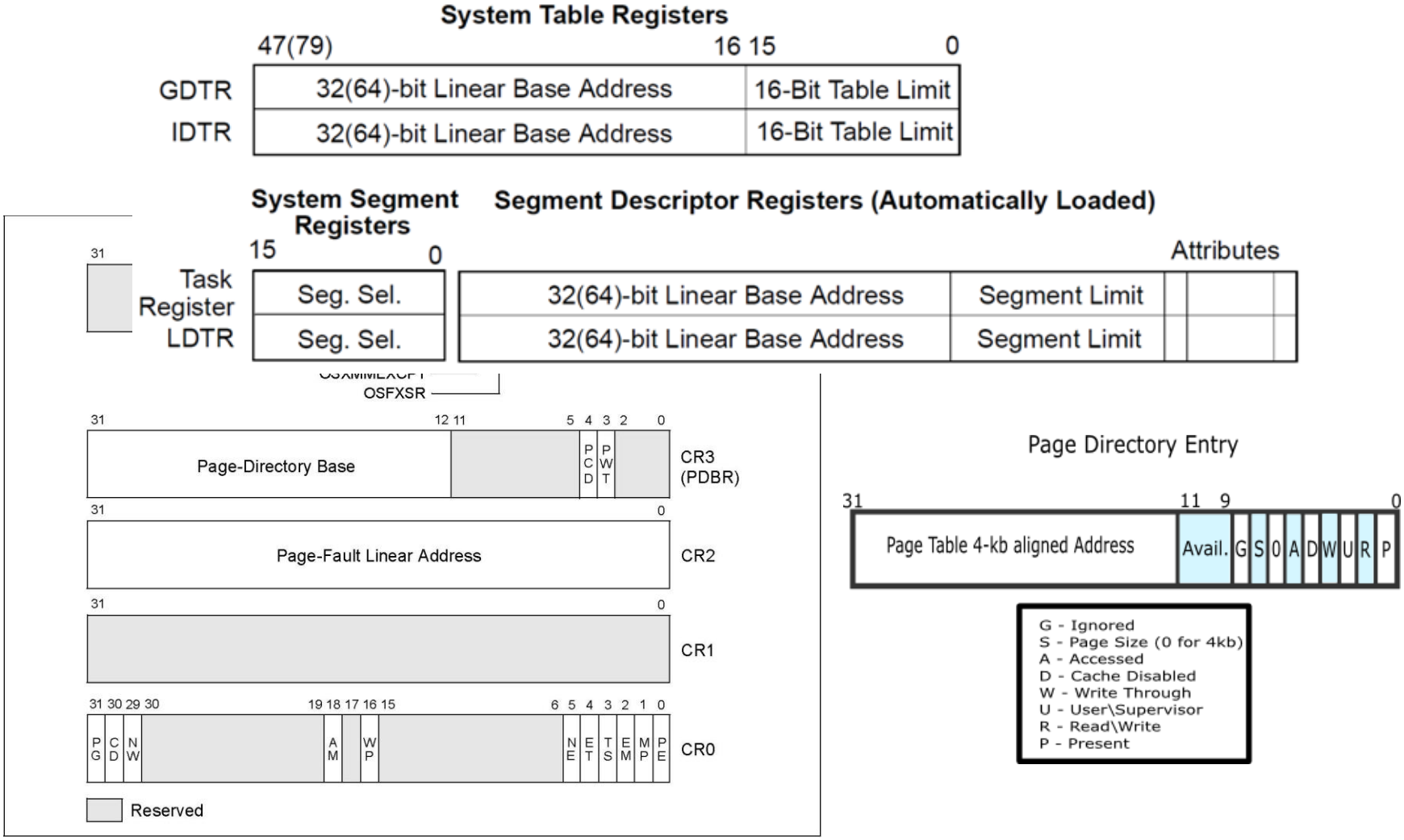




Note:

Gate descriptors all point to code segment descriptors
Segment descriptors can point to code or data segments
Call-gate descriptors reside in the GDT (not the IDT)
Segmentation is enabled by the PE flag in CR0
Paging is enabled by the PG flag in CR0
If paging is not enabled, linear address → physical address

Pola rejestrów



Opis komponentów adresowania w x86_64