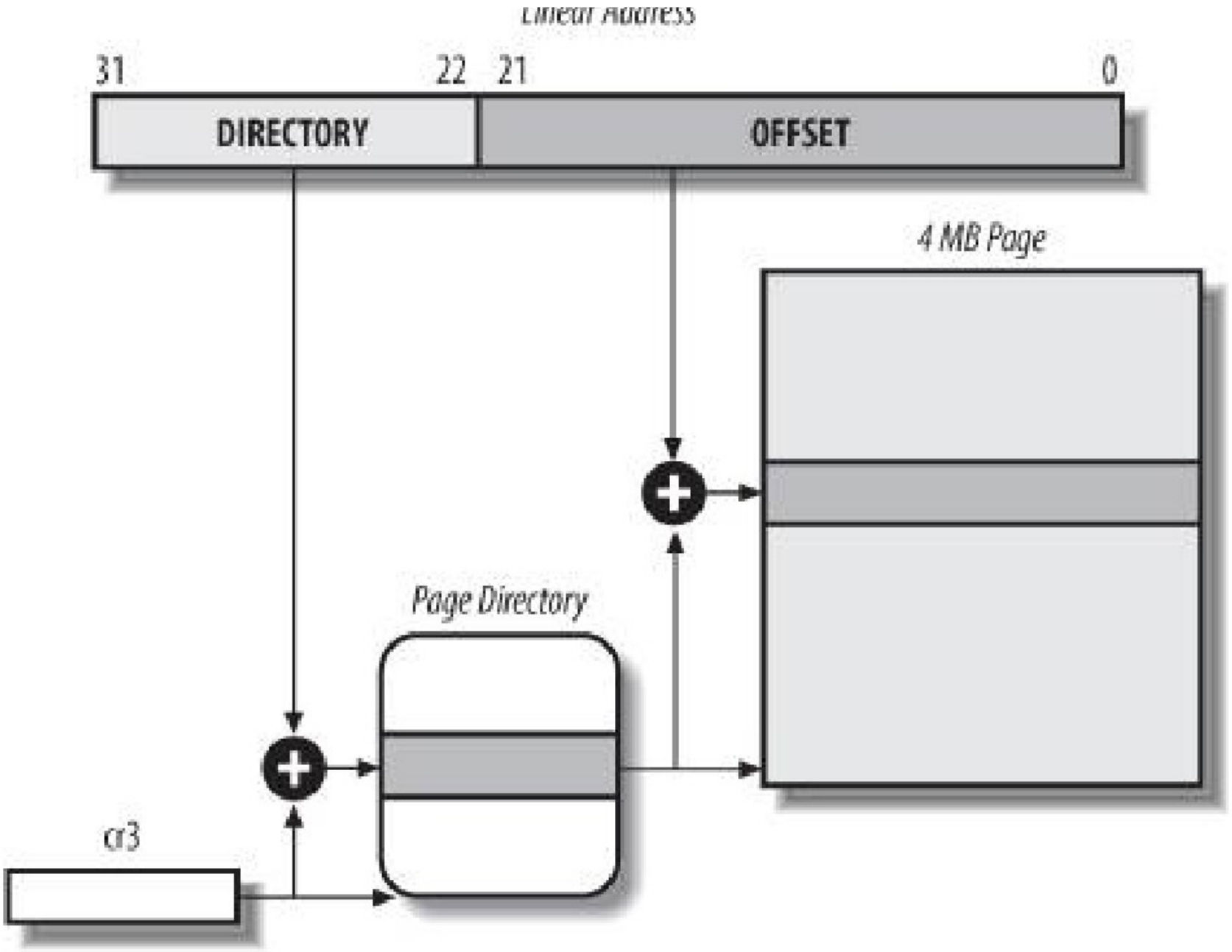
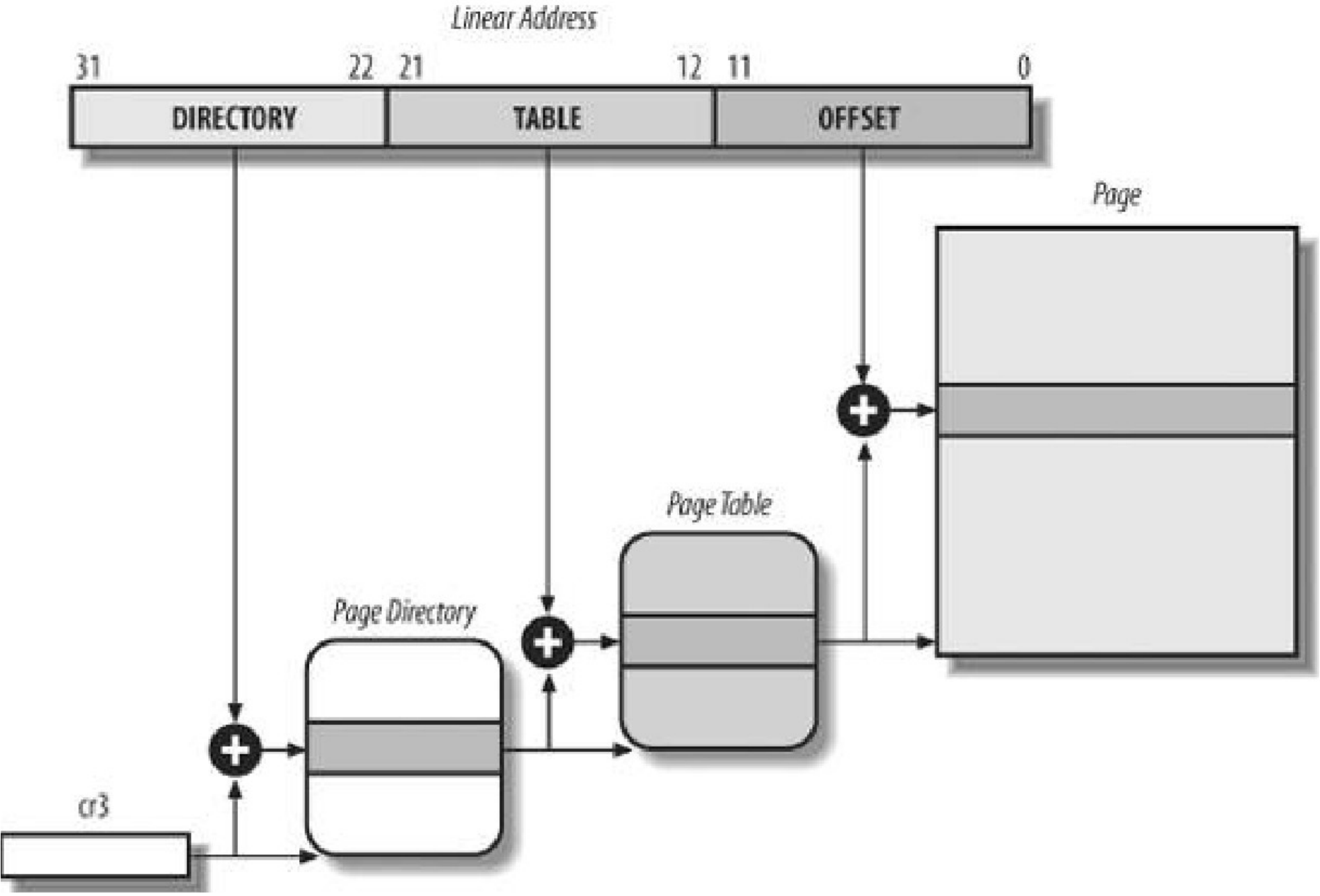
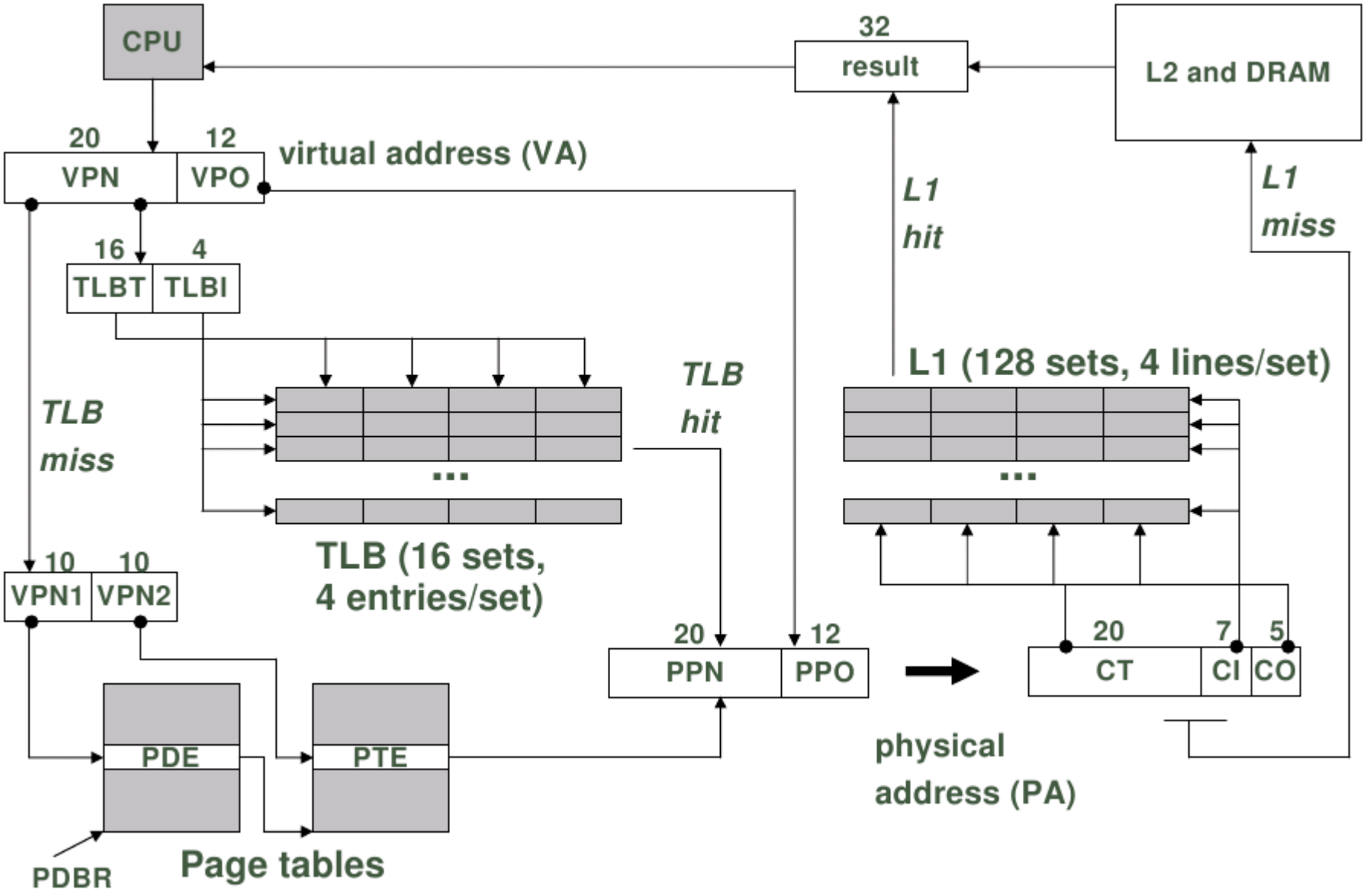
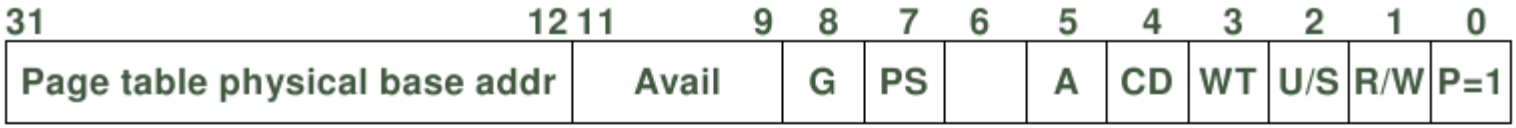

Linear Address
31
22
21
0
DIRECTORY
OFFSET
4 MB Page
Page Directory
cr3


Linear Address
31
22
21
12
11
0
DIRECTORY
TABLE
OFFSET
Page
Page Table
Page Directory
cr3


CPU
32
result
L2 and DRAM
20
12
VPN
VPO
virtual address (VA)
L1
hit
L1
miss
16
4
TLBT
TLBI
TLB
hit
L1 (128 sets, 4 lines/set)
TLB
miss
TLB (16 sets,
4 entries/set)
10
10
VPN1
VPN2
20
12
PPN
PPO
20
7
5
CT
CI
CO
PDE
PTE
physical
address (PA)
PDBR
Page tables



Page table physical base address: 20 most significant bits of physical page table address (forces page tables to be 4KB aligned)

Avail: These bits available for system programmers

G: global page (don't evict from TLB on task switch)

PS: page size 4K (0) or 4M (1)

A: accessed (set by MMU on reads and writes, cleared by software)

CD: cache disabled (1) or enabled (0)

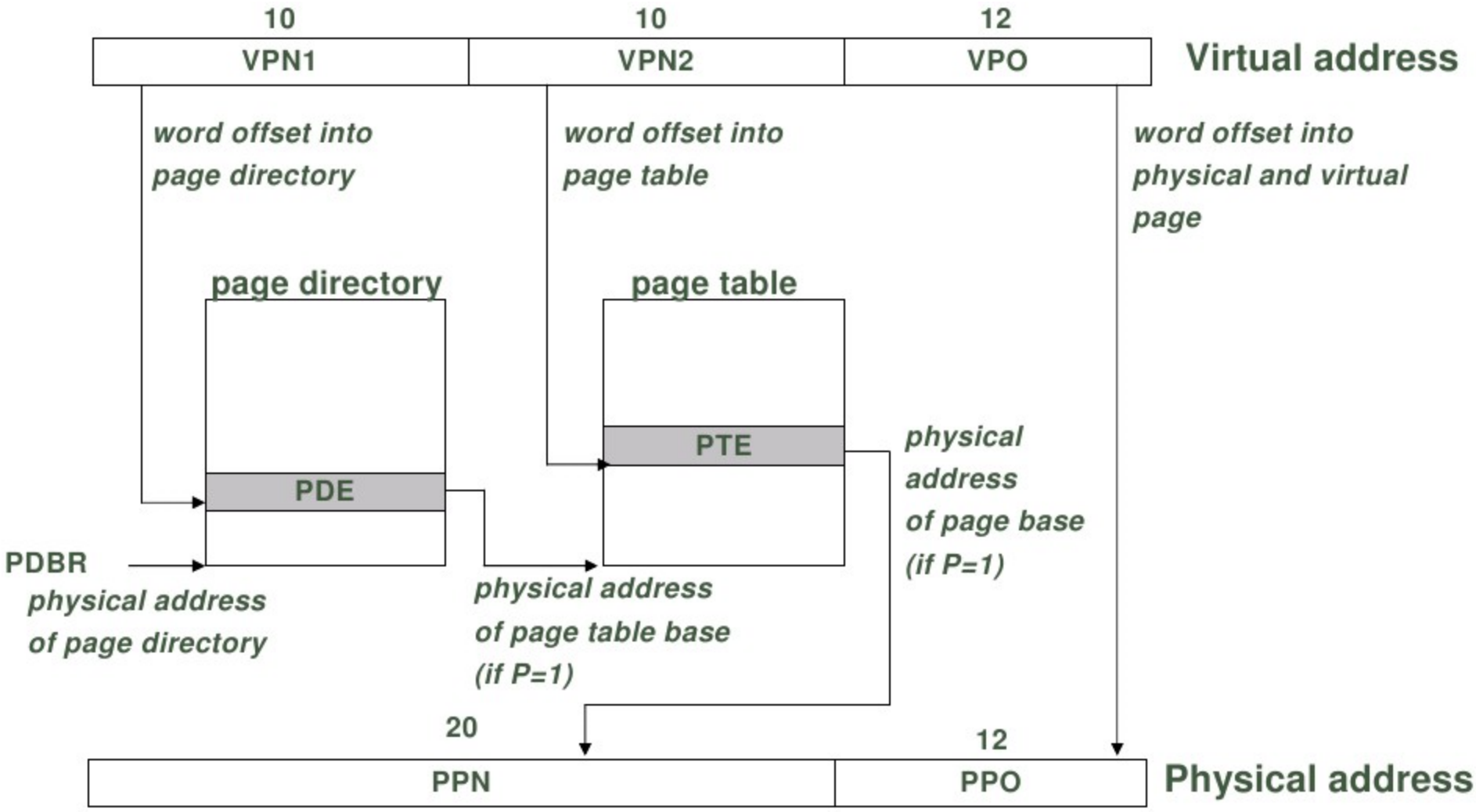
WT: write-through or write-back cache policy for this page table

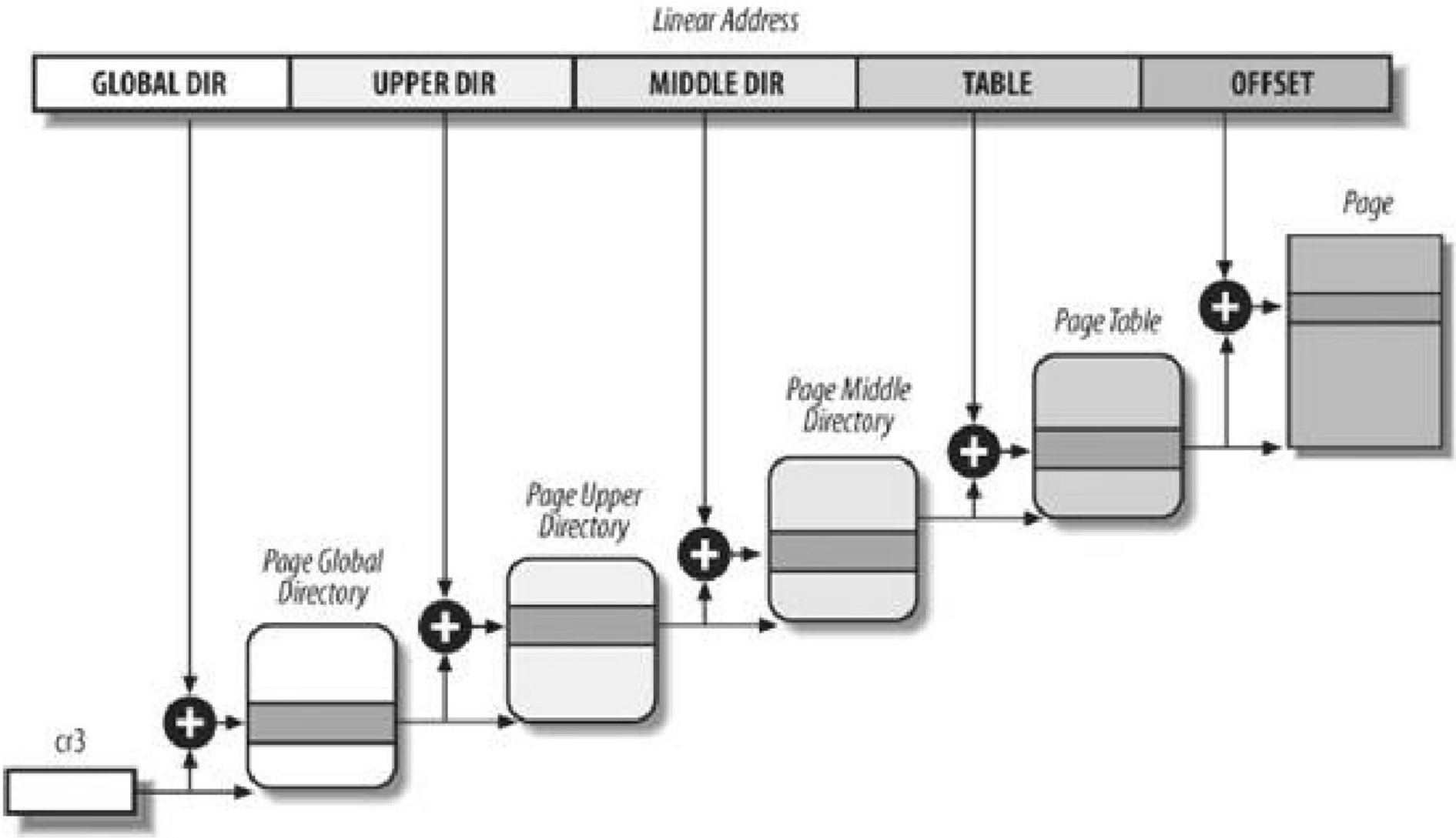
U/S: user or supervisor mode access

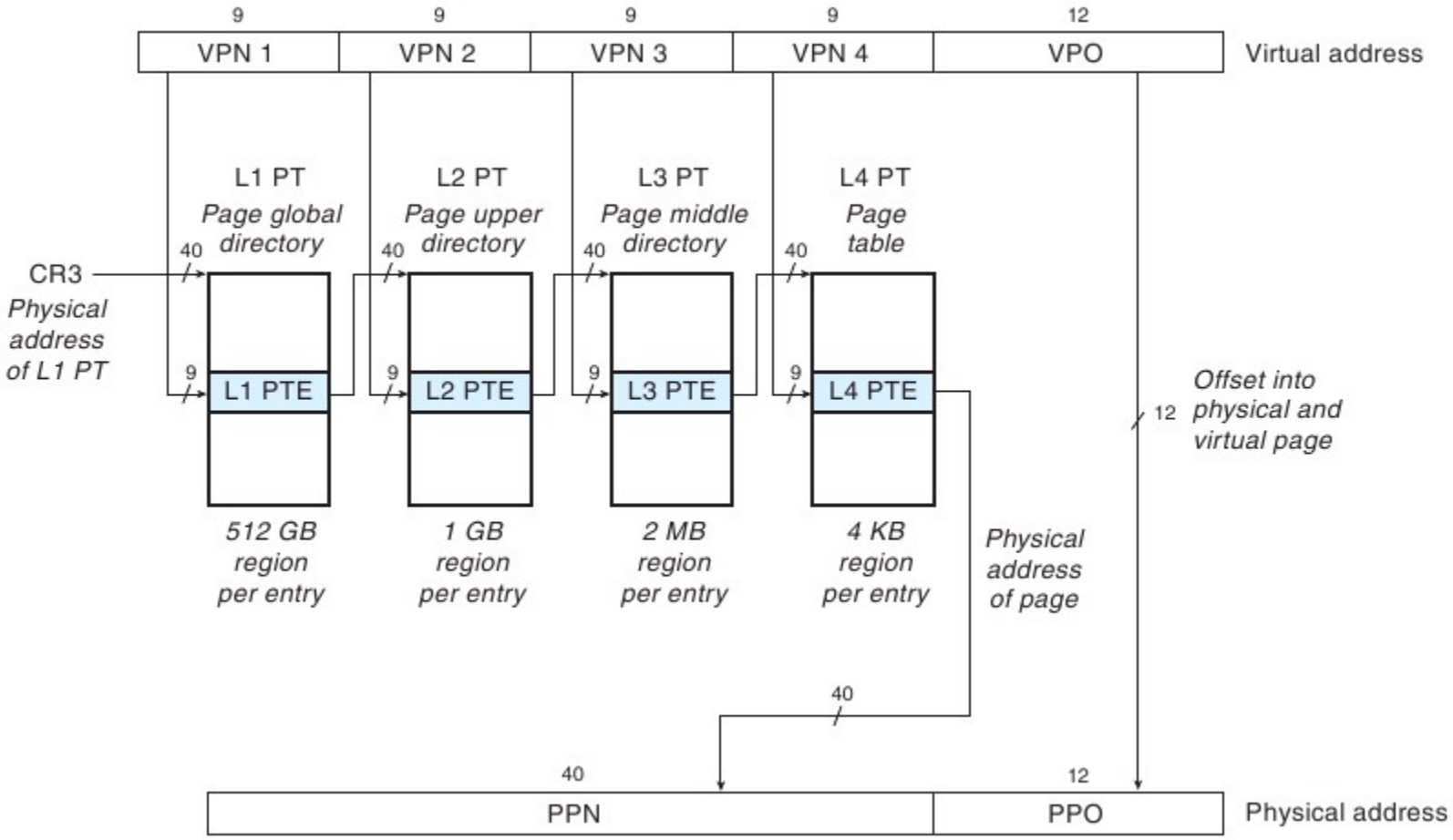
R/W: read-only or read-write access

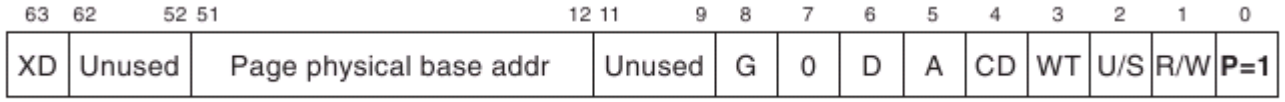
P: page table is present in memory (1) or not (0)

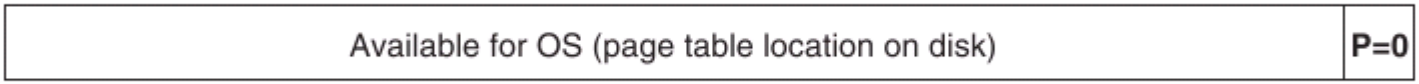

31
1
0
Available for OS (page table location in secondary storage)
P=0


10
10
12
VPN1
VPN2
VPO
Virtual address
word offset into page directory
word offset into page table
word offset into physical and virtual page
page directory
page table
PTE
PDE
physical address of page base (if P=1)
PDBR
physical address of page directory
physical address of page table base (if P=1)
20
12
PPN
PPO
Physical address

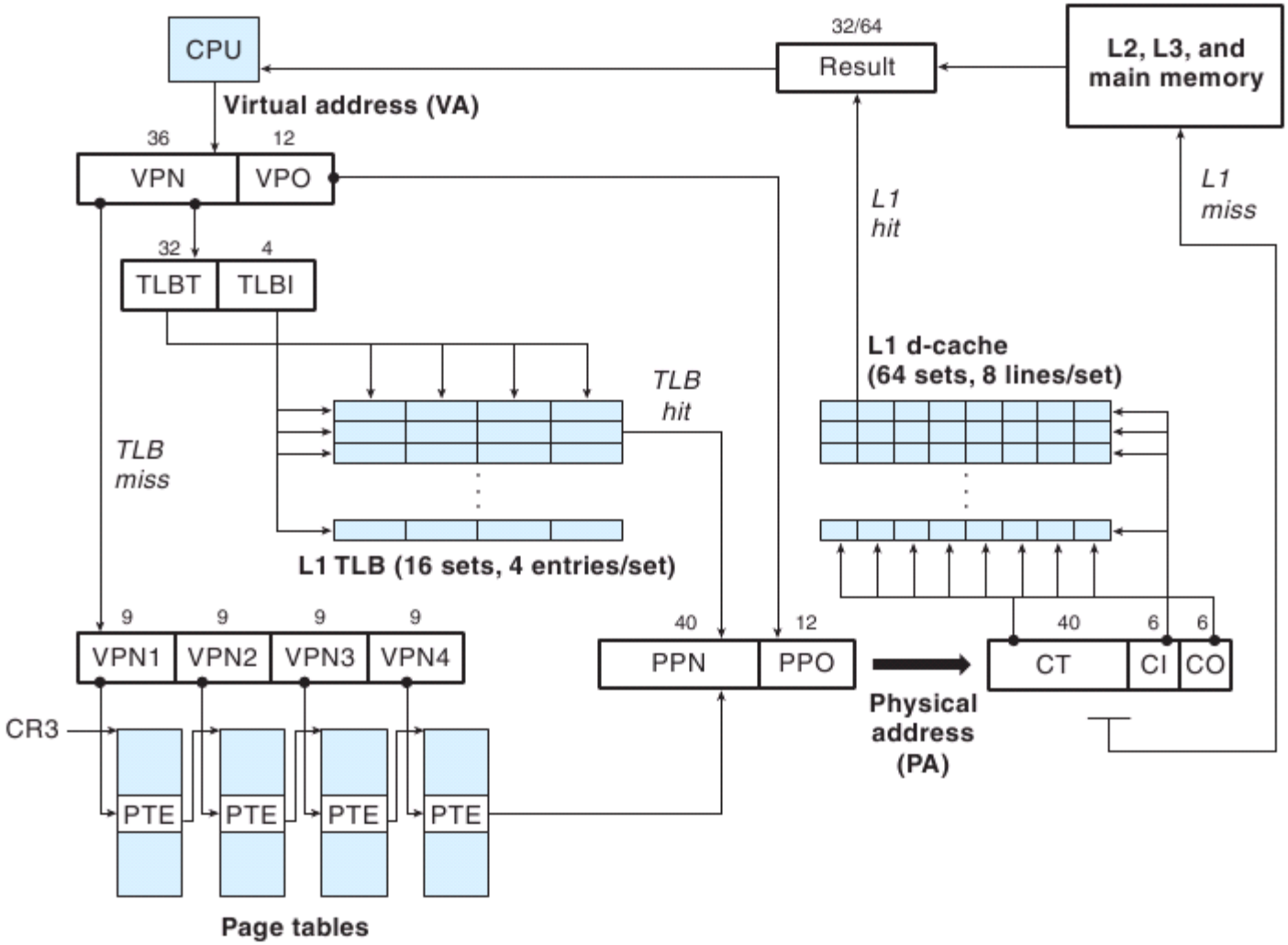

Linear Address
GLOBAL DIR
UPPER DIR
MIDDLE DIR
TABLE
OFFSET
Page
Page Table
Page Middle Directory
Page Upper Directory
Page Global Directory
cr3

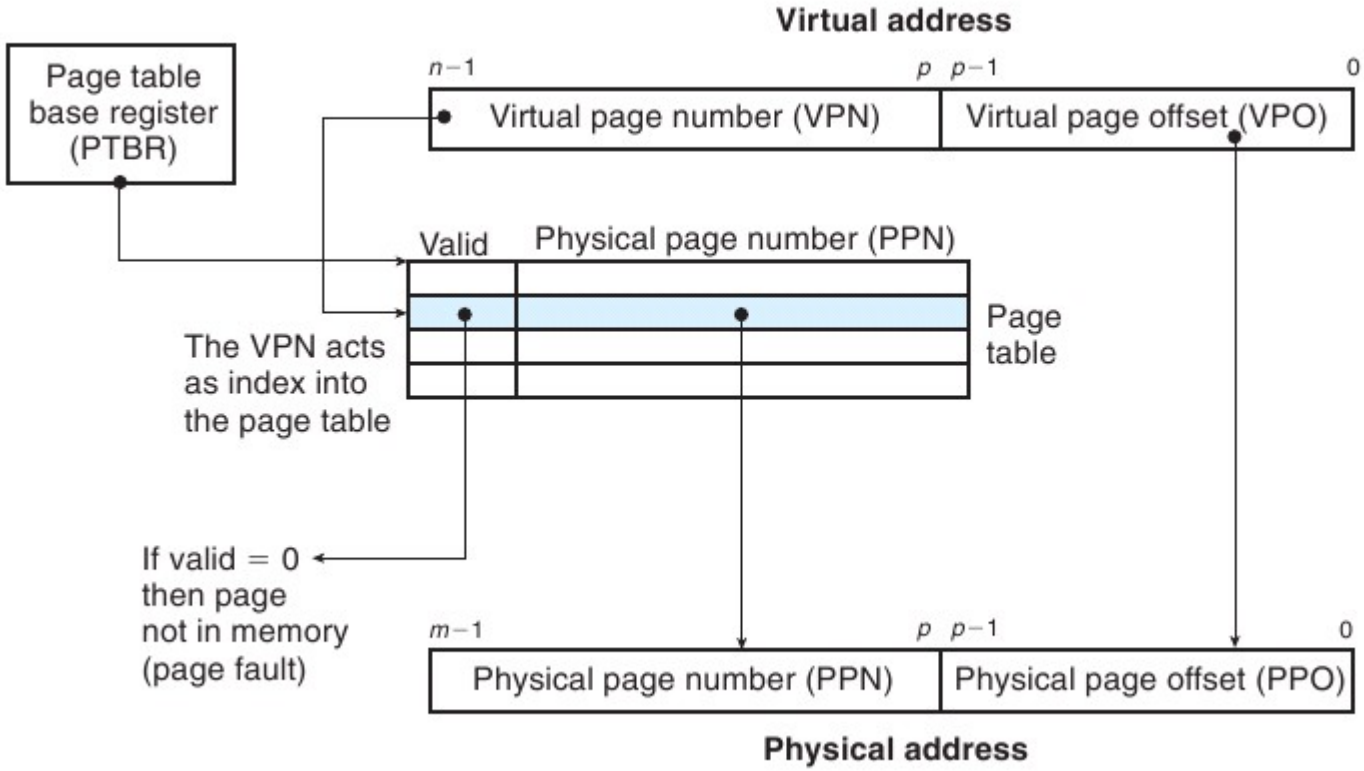

9
9
9
9
12
VPN 1
VPN 2
VPN 3
VPN 4
VPO
Virtual address
L1 PT
Page global directory
L2 PT
Page upper directory
L3 PT
Page middle directory
L4 PT
Page table
40
40
40
40
CR3
Physical address of L1 PT
9
9
9
9
L1 PTE
L2 PTE
L3 PTE
L4 PTE
Offset into physical and virtual page
12
512 GB region per entry
1 GB region per entry
2 MB region per entry
4 KB region per entry
Physical address of page
40
40
12
PPN
PPO
Physical address

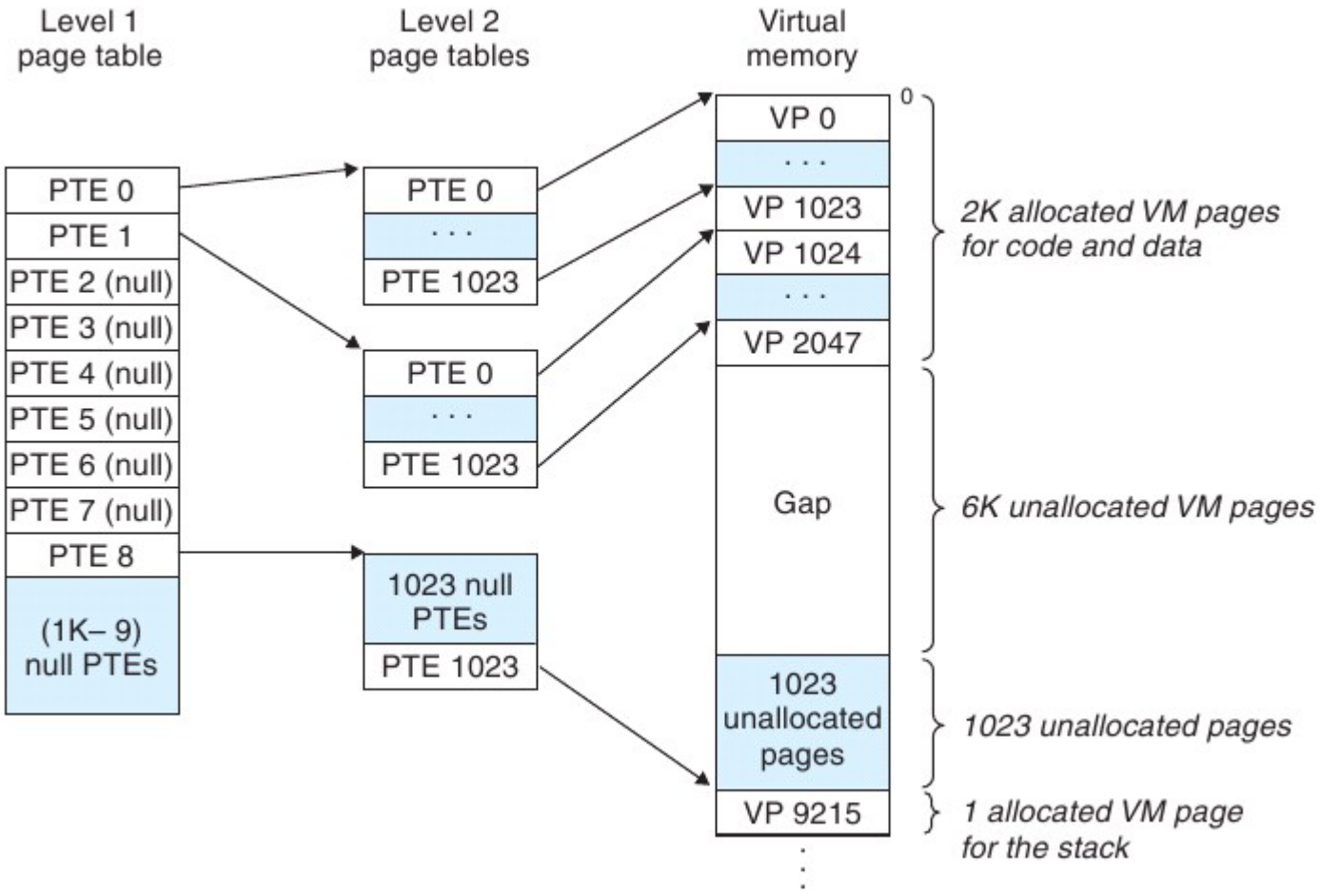

63
62
52
51
12
11
9
8
7
6
5
4
3
2
1
0
XD
Unused
Page physical base addr
Unused
G
0
D
A
CD
WT
U/S
R/W
P=1

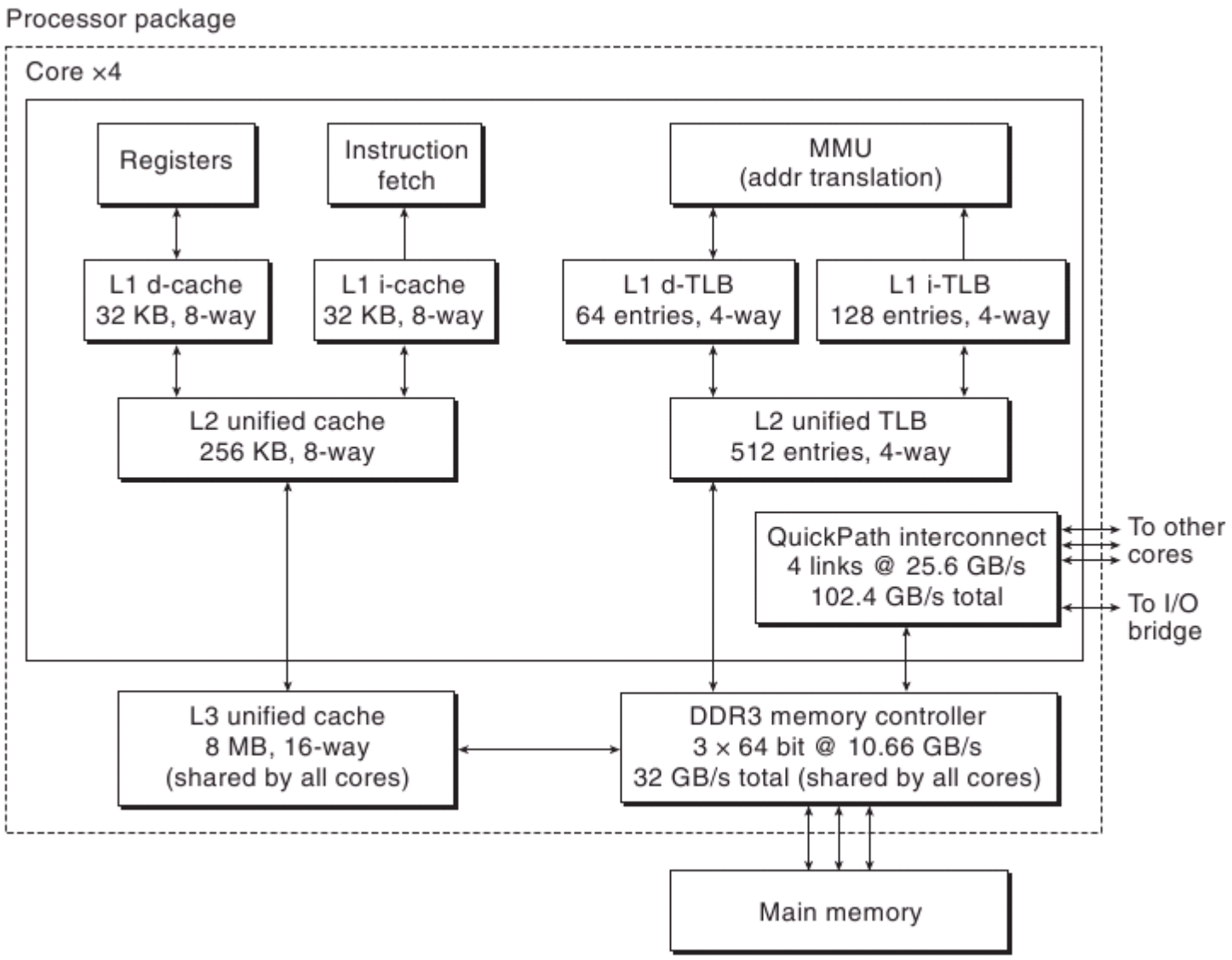

Available for OS (page table location on disk)
P=0

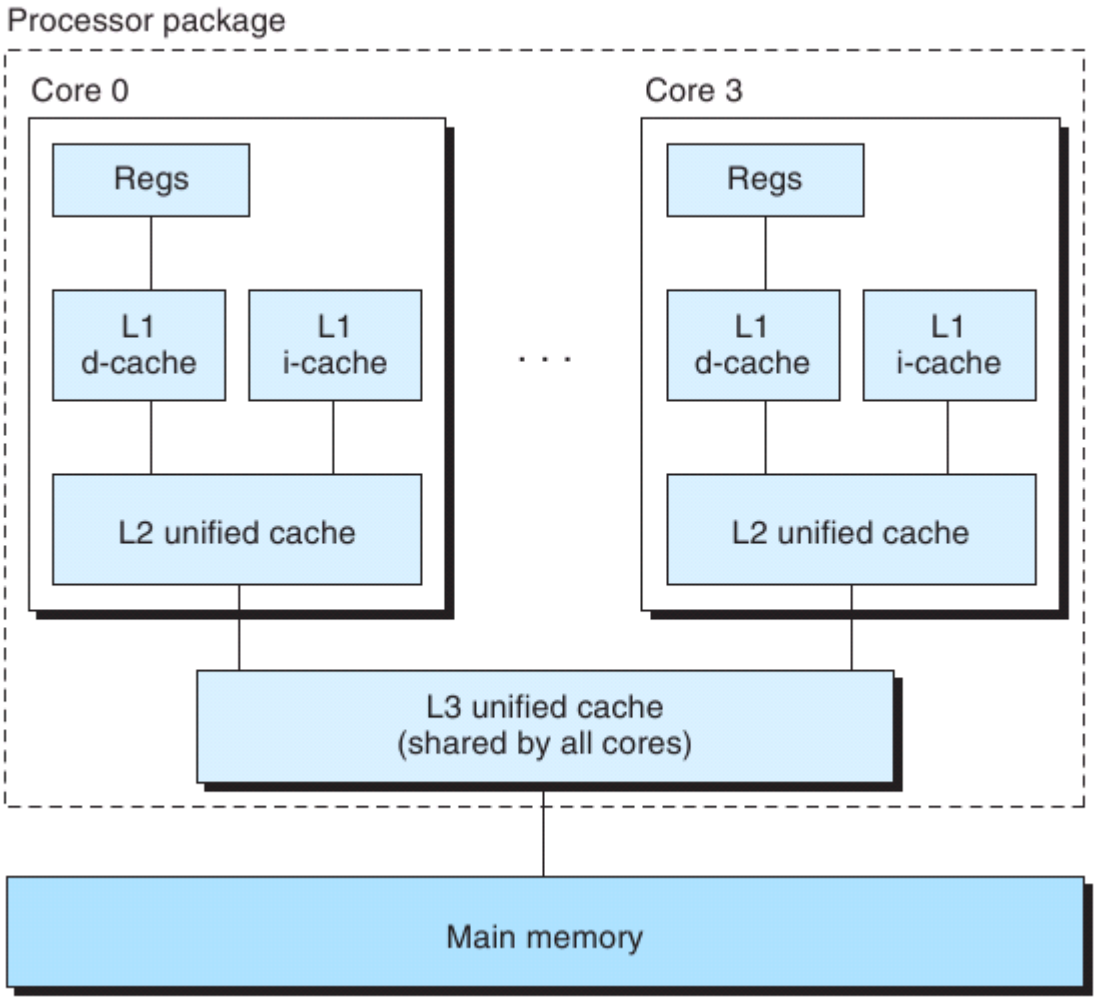
Field	Description
P	Child page table present in physical memory (1) or not (0).
R/W	Read-only or read-write access permission for all reachable pages.
U/S	User or supervisor (kernel) mode access permission for all reachable pages.
WT	Write-through or write-back cache policy for the child page table.
CD	Caching disabled or enabled for the child page table.
A	Reference bit (set by MMU on reads and writes, cleared by software).
PS	Page size either 4 KB or 4 MB (defined for Level 1 PTEs only).
Base addr	40 most significant bits of physical base address of child page table.
XD	Disable or enable instruction fetches from all pages reachable from this PTE.

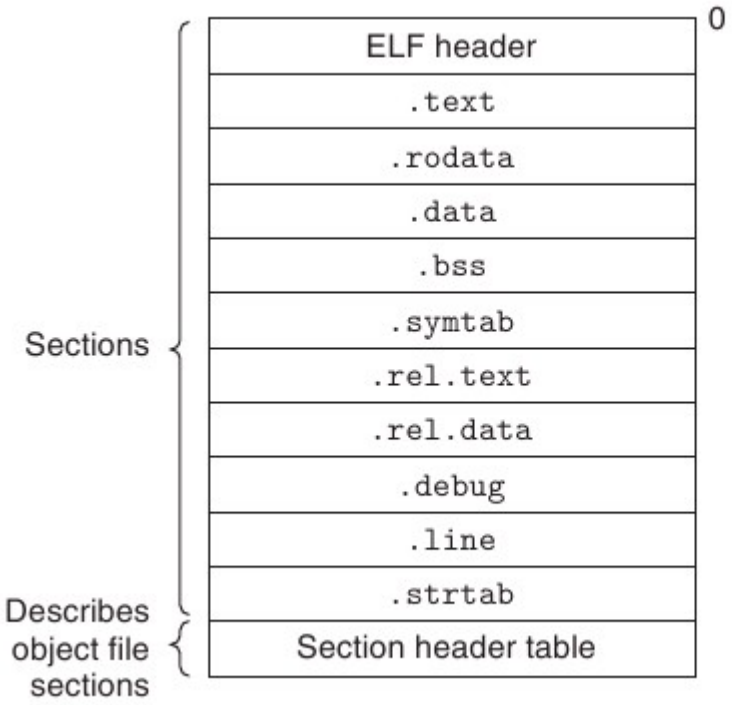

CPU
32/64
Result
L2, L3, and main memory
Virtual address (VA)
36
12
VPN
VPO
L1 hit
L1 miss
32
4
TLBT
TLBI
L1 d-cache (64 sets, 8 lines/set)
TLB hit
TLB miss
L1 TLB (16 sets, 4 entries/set)
9
9
9
9
40
12
40
6
6
VPN1
VPN2
VPN3
VPN4
PPN
PPO
CT
CI
CO
Physical address (PA)
CR3
PTE
PTE
PTE
PTE
Page tables

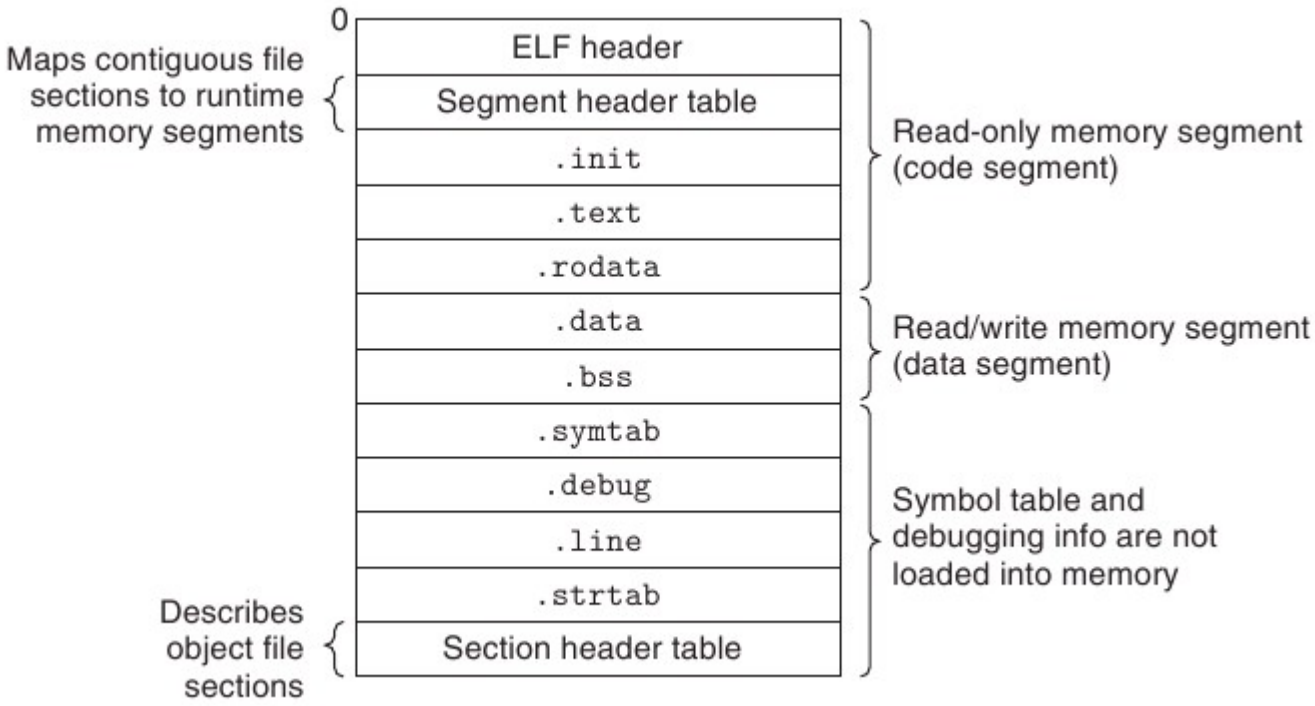

Virtual address
Page table base register (PTBR)
n−1
p
p−1
0
Virtual page number (VPN)
Virtual page offset (VPO)
Valid
Physical page number (PPN)
Page table
The VPN acts as index into the page table
If valid = 0 then page not in memory (page fault)
m−1
p
p−1
0
Physical page number (PPN)
Physical page offset (PPO)
Physical address


Level 1
page table
Level 2
page tables
Virtual
memory
PTE 0
PTE 1
PTE 2 (null)
PTE 3 (null)
PTE 4 (null)
PTE 5 (null)
PTE 6 (null)
PTE 7 (null)
PTE 8
(1K– 9)
null PTEs
PTE 0
. . .
PTE 1023
PTE 0
. . .
PTE 1023
1023 null
PTEs
PTE 1023
0
VP 0
. . .
VP 1023
VP 1024
. . .
VP 2047
Gap
1023
unallocated
pages
VP 9215
2K allocated VM pages
for code and data
6K unallocated VM pages
1023 unallocated pages
1 allocated VM page
for the stack


Processor package
Core ×4
Registers
Instruction fetch
MMU (addr translation)
L1 d-cache 32 KB, 8-way
L1 i-cache 32 KB, 8-way
L1 d-TLB 64 entries, 4-way
L1 i-TLB 128 entries, 4-way
L2 unified cache 256 KB, 8-way
L2 unified TLB 512 entries, 4-way
QuickPath interconnect 4 links @ 25.6 GB/s 102.4 GB/s total
To other cores
To I/O bridge
L3 unified cache 8 MB, 16-way (shared by all cores)
DDR3 memory controller 3 × 64 bit @ 10.66 GB/s 32 GB/s total (shared by all cores)
Main memory


Processor package
Core 0
Regs
L1
d-cache
L1
i-cache
L2 unified cache
. . .
Core 3
Regs
L1
d-cache
L1
i-cache
L2 unified cache
L3 unified cache
(shared by all cores)
Main memory


0
ELF header
.text
.rodata
.data
.bss
.symtab
.rel.text
.rel.data
.debug
.line
.strtab
Section header table
Sections
Describes object file sections

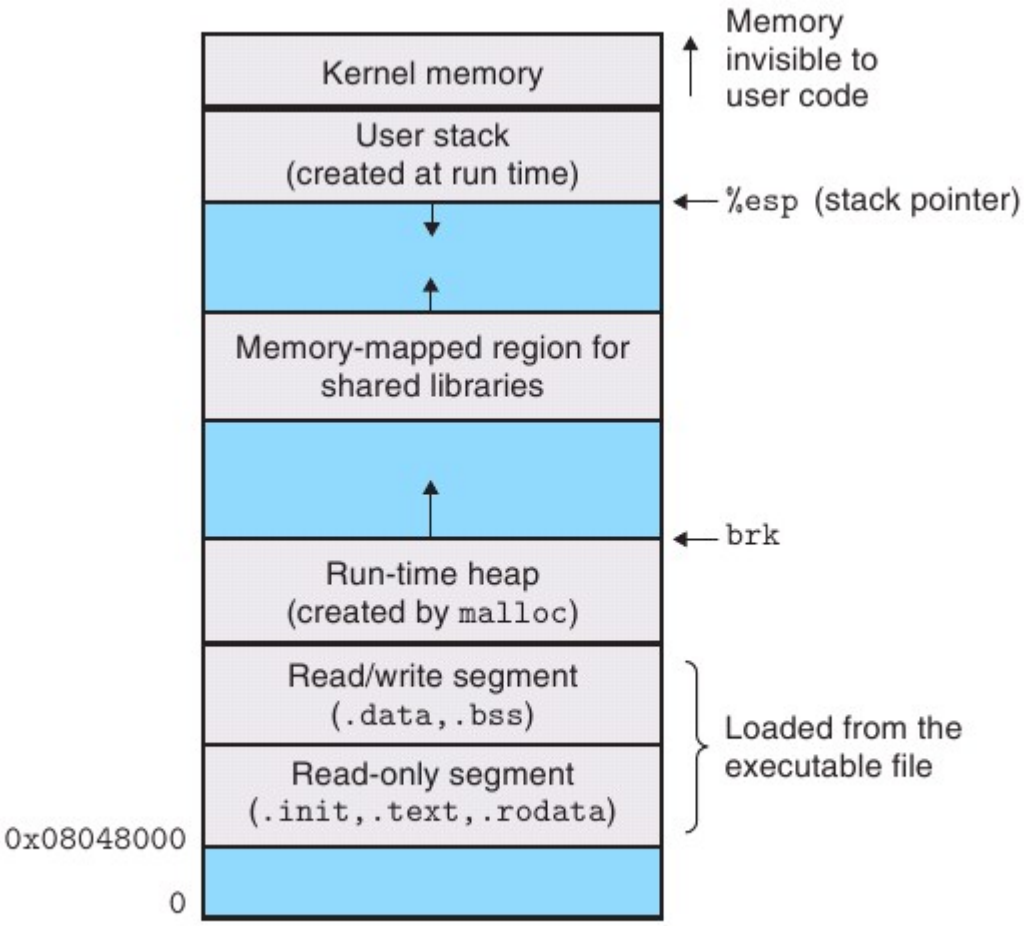

0
ELF header
Segment header table
.init
.text
.rodata
.data
.bss
.symtab
.debug
.line
.strtab
Section header table
Maps contiguous file sections to runtime memory segments
Describes object file sections
Read-only memory segment (code segment)
Read/write memory segment (data segment)
Symbol table and debugging info are not loaded into memory

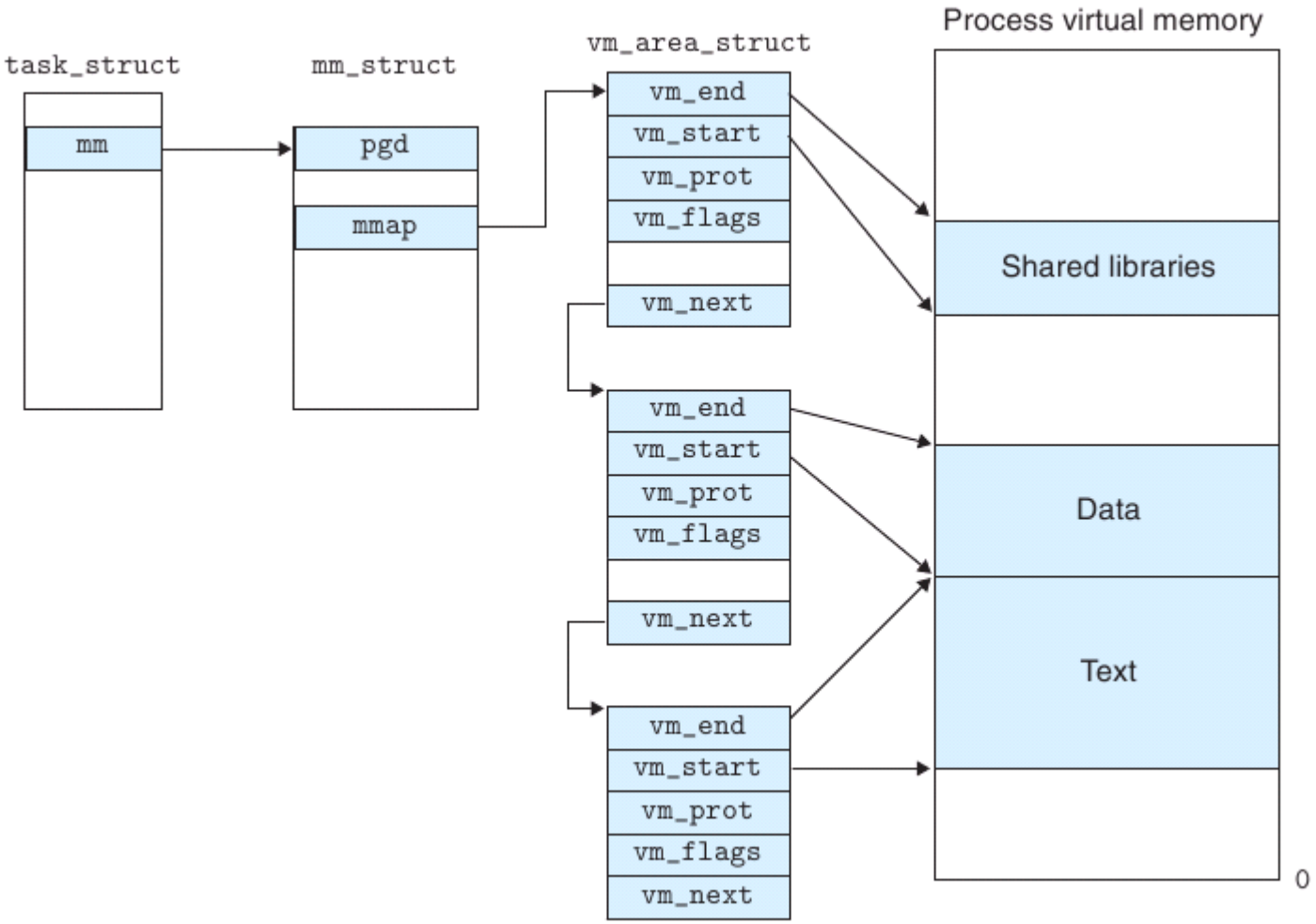
Read-only code segment

```
  LOAD off    0x00000000 vaddr 0x08048000 paddr 0x08048000 align 2**12
       filesz 0x00000448 memsz 0x00000448 flags r-x
```

Read/write data segment

```
  LOAD off    0x00000448 vaddr 0x08049448 paddr 0x08049448 align 2**12
       filesz 0x000000e8 memsz 0x00000104 flags rw-
```


Kernel memory
Memory invisible to user code
User stack (created at run time)
%esp (stack pointer)
Memory-mapped region for shared libraries
brk
Run-time heap (created by malloc)
Read/write segment (.data, .bss)
Read-only segment (.init, .text, .rodata)
Loaded from the executable file
0x08048000
0


task_struct
mm
mm_struct
pgd
mmap
vm_area_struct
vm_end
vm_start
vm_prot
vm_flags
vm_next
vm_end
vm_start
vm_prot
vm_flags
vm_next
vm_end
vm_start
vm_prot
vm_flags
vm_next
Process virtual memory
Shared libraries
Data
Text
0