Virtual Memory: Systems

Adapted form Randal E. Bryant and David R. O'Hallaron ， Computer Systems: A Programmer's Perspective, 3/E (CS:APP3e ）, 即 "深入理解计算机系统（中文版）" Chapter 9.

Qianni Deng

Department of Computer Science and Engineering

Shanghai Jiao Tong University

Spring 2018

Today

- **Simple memory system example**
- Case study: Core i7/Linux memory system

Review of Symbols

- **Basic Parameters**
  - **N = 2^n** : Number of addresses in virtual address space
  - **M = 2^m** : Number of addresses in physical address space
  - **P = 2^p** : Page size (bytes)
- **Components of the virtual address (VA)**
  - **TLBI**: TLB index
  - **TLBT**: TLB tag
  - **VPO**: Virtual page offset
 - **VPN**: Virtual page number
- **Components of the physical address (PA)**
  - **PPO**: Physical page offset (same as VPO)
  - **PPN:** Physical page number
  - **CO**: Byte offset within cache line
  - **CI:** Cache index
  - **CT**: Cache tag

Simple Memory System Example

- **Addressing**
  - 14-bit virtual addresses
  - 12-bit physical address
  - Page size = 64 bytes

13	12	11	10	9	8	7	6	5	4	3	2	1	0

← — — — — — VPN — — — — — → ← — — VPO — — →

Virtual Page Number   **Virtual Page Offset**

11	10	9	8	7	6	5	4	3	2	1	0

← — — — — PPN — — — — → ← — — PPO — — →

Physical Page Number   **Physical Page Offset**
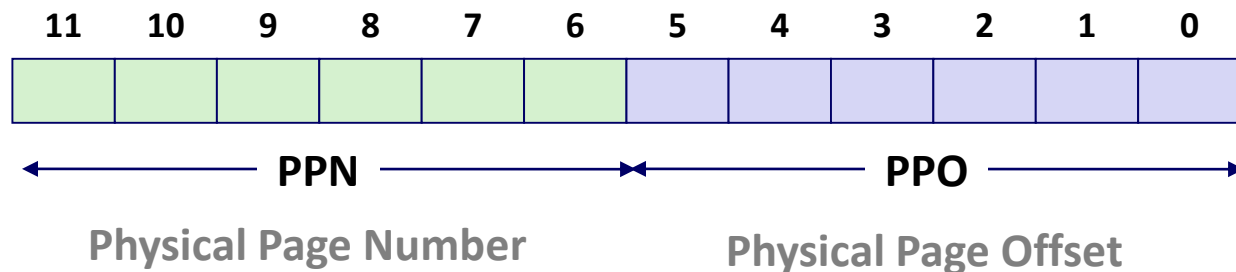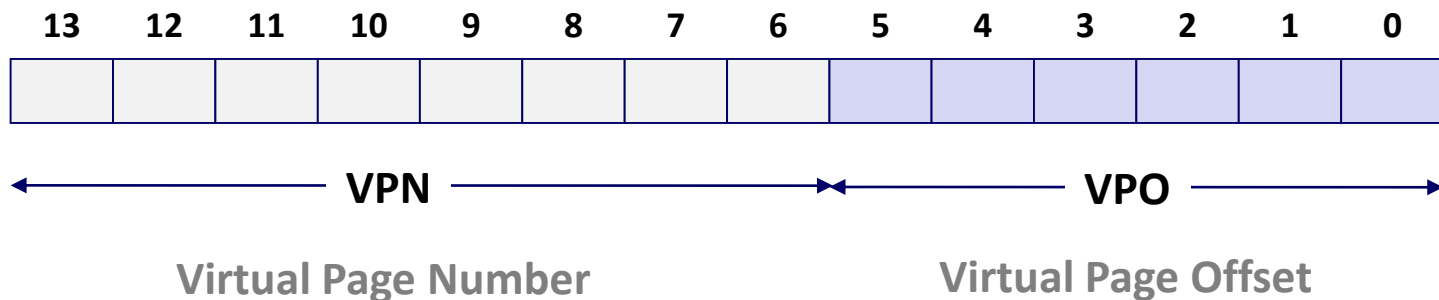
1. Simple Memory System TLB

- **16 entries**
- **4-way associative**

TLBT ← → TLBI →

| 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

← VPN → ← VPO →
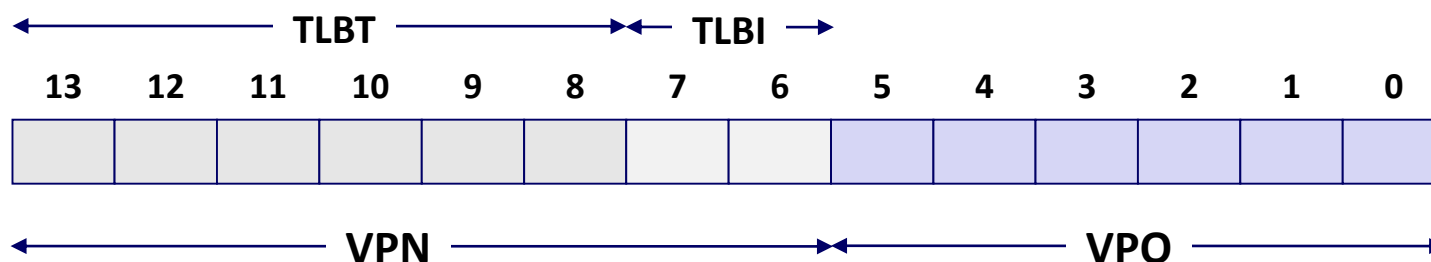
Set	Tag	PPN	Valid	Tag	PPN	Valid	Tag	PPN	Valid	Tag	PPN	Valid
0	03	–	0	09	0D	1	00	–	0	07	02	1
1	03	2D	1	02	–	0	04	–	0	0A	–	0
2	02	–	0	08	–	0	06	–	0	03	–	0
3	07	–	0	03	0D	1	0A	34	1	02	–	0

2. Simple Memory System Page Table

Only show first 16 entries (out of 256)

VPN	PPN	Valid
00	28	1
01	–	0
02	33	1
03	02	1
04	–	0
05	16	1
06	–	0
07	–	0

VPN	PPN	Valid
08	13	1
09	17	1
0A	09	1
0B	–	0
0C	–	0
0D	2D	1
0E	11	1
0F	0D	1

3. Simple Memory System Cache

- **16 lines, 4-byte block size**
- **Physically addressed**
- **Direct mapped**

	CT						CI				CO	
11	10	9	8	7	6	5	4	3	2	1	0

PPN | | | | | | PPO
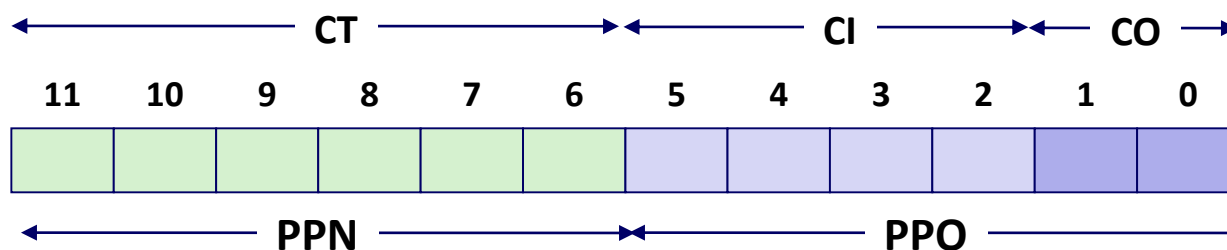
Idx	Tag	Valid	B0	B1	B2	B3
0	19	1	99	11	23	11
1	15	0	–	–	–	–
2	1B	1	00	02	04	08
3	36	0	–	–	–	–
4	32	1	43	6D	8F	09
5	0D	1	36	72	F0	1D
6	31	0	–	–	–	–
7	16	1	11	C2	DF	03

Idx	Tag	Valid	B0	B1	B2	B3
8	24	1	3A	00	51	89
9	2D	0	–	–	–	–
A	2D	1	93	15	DA	3B
B	0B	0	–	–	–	–
C	12	0	–	–	–	–
D	16	1	04	96	34	15
E	13	1	83	77	1B	D3
F	14	0	–	–	–	–

Address Translation Example #1

Virtual Address: 0x03D4

	TLBT						TLBI						
13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	0	1	0	1	0	0

← VPN → ← VPO →
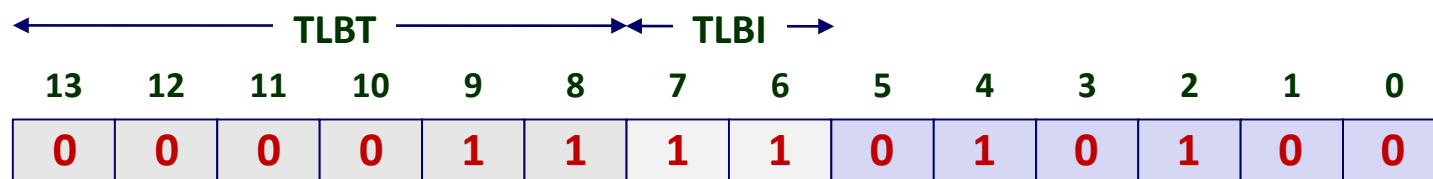
VPN **0x0F**   TLBI **0x3**   TLBT **0x03**   TLB Hit? **Y**   Page Fault? **N**   PPN: **0x0D**

Physical Address

	CT						CI				CO	
11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	1	0	1	0	1	0	1	0	0

← PPN → ← PPO →
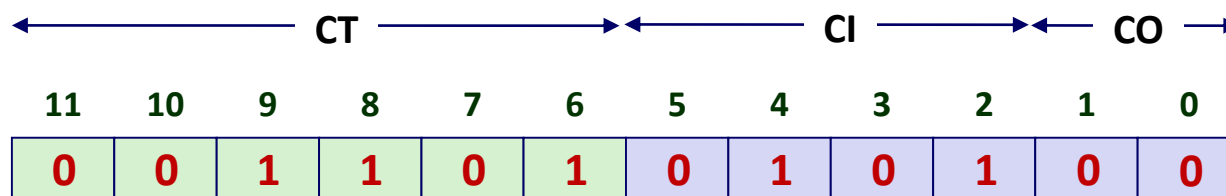
CO **0**   CI **0x5**   CT **0x0D**   Hit? **Y**   Byte: **0x36**

Address Translation Example #2

Virtual Address: 0x0020
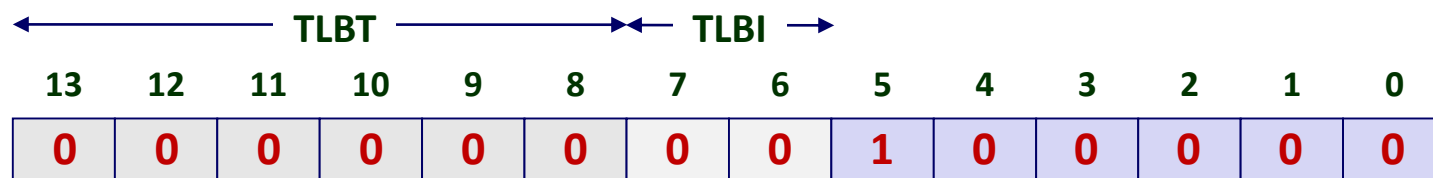
	TLBT						TLBI						
13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0

VPN ↔ VPO

VPN **0x00**   TLBI **0**   TLBT **0x00**   TLB Hit? **N**   Page Fault? **N**   PPN: **0x28**

Physical Address

	CT						CI				CO	
11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	0	0	0	1	0	0	0	0	0

PPN ↔ PPO
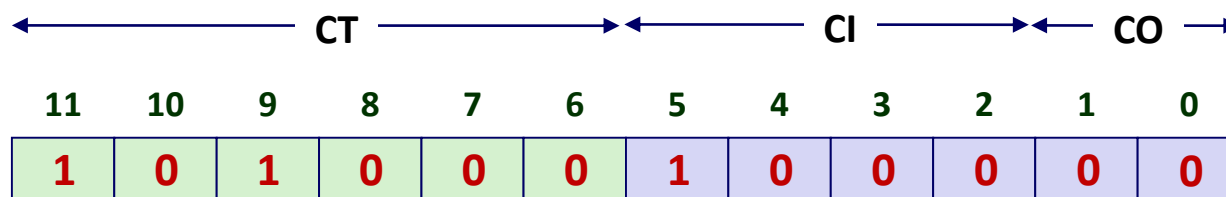
CO **0**   CI **0x8**   CT **0x28**   Hit? **N**   Byte: **Mem**

Address Translation Example #3

Virtual Address: 0x0020
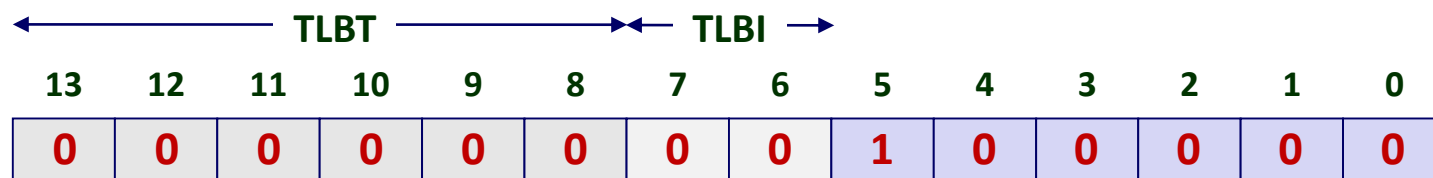
	TLBT							TLBI					
13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0

VPN → | VPO

VPN **0x00**   TLBI **0**   TLBT **0x00**   TLB Hit? **N**   Page Fault? **N**   PPN: **0x28**

Physical Address

	CT						CI				CO	
11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	0	0	0	1	0	0	0	0	0

PPN → | PPO
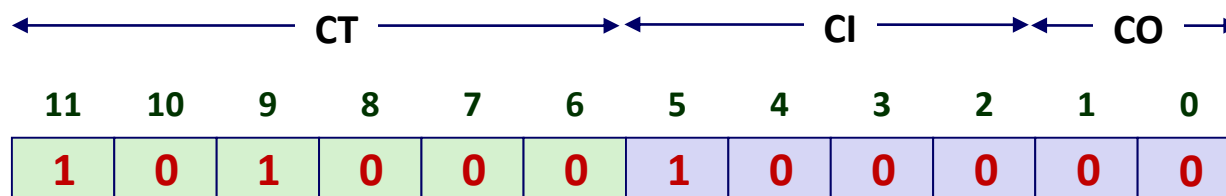
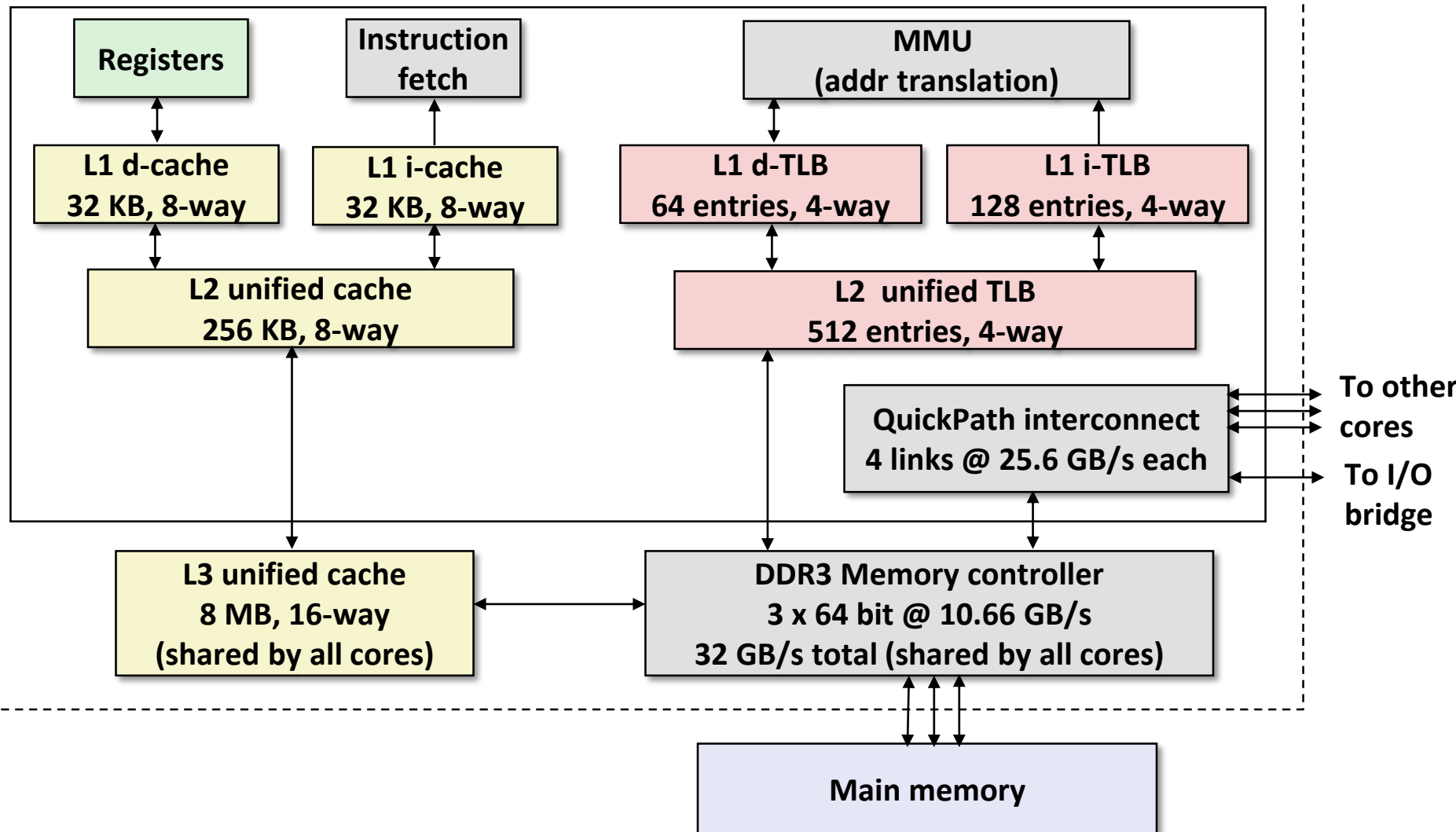CO **0**   CI **0x8**   CT **0x28**   Hit? **N**   Byte: **Mem**

Today

- **Simple memory system example**

- **Case study: Core i7/Linux memory system**

- **Memory mapping**

Intel Core i7 Memory System

Processor package

Core x4

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 each

To other cores

To I/O bridge

L3 unified cache 8 MB, 16-way (shared by all cores)

DDR3 Memory controller 3 x 64 bit @ 10.66 GB/s 32 GB/s total (shared by all cores)

Main memory

Review of Symbols

- **Basic Parameters**
  - **N = 2^n** : Number of addresses in virtual address space
  - **M = 2^m** : Number of addresses in physical address space
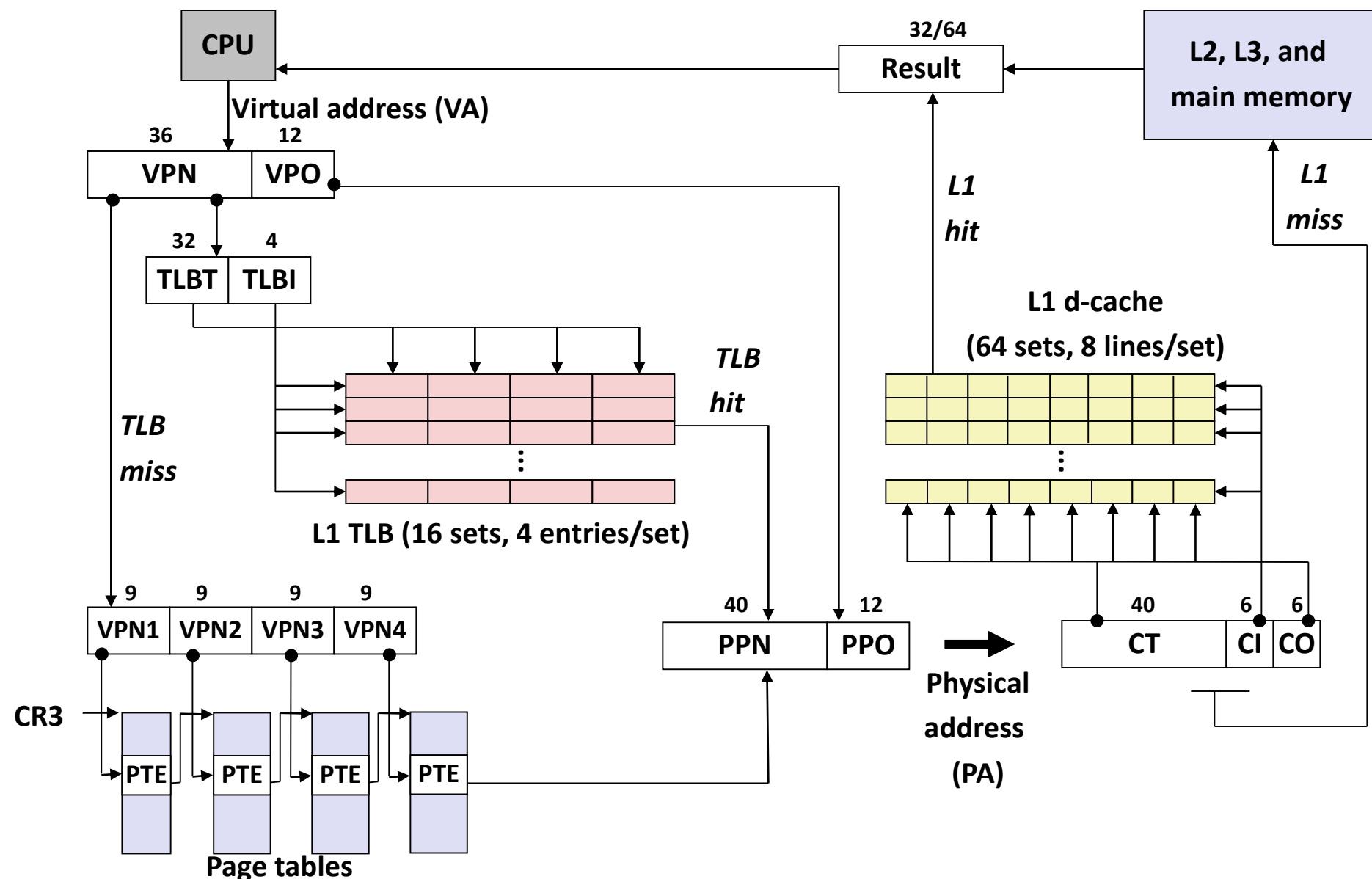 - **P = 2^p** : Page size (bytes)
- **Components of the virtual address (VA)**
  - **TLBI**: TLB index
  - **TLBT**: TLB tag
  - **VPO**: Virtual page offset
  - **VPN**: Virtual page number
- **Components of the physical address (PA)**
  - **PPO**: Physical page offset (same as VPO)
  - **PPN:** Physical page number
  - **CO**: Byte offset within cache line
  - **CI:** Cache index
  - **CT**: Cache tag

End-to-end Core i7 Address Translation

Core i7 Level 1-3 Page Table Entries

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

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

Each entry references a 4K child page table. Significant fields:

P: Child page table present in physical memory (1) or not (0).

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

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).

Page table physical base address: 40 most significant bits of physical page table address (forces page tables to be 4KB aligned)

XD: Disable or enable instruction fetches from all pages reachable from this PTE.

Core i7 Level 4 Page Table Entries

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

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

Each entry references a 4K child page. Significant fields:

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

R/W: Read-only or read-write access permission for child page

U/S: User or supervisor mode access

WT: Write-through or write-back cache policy for this page
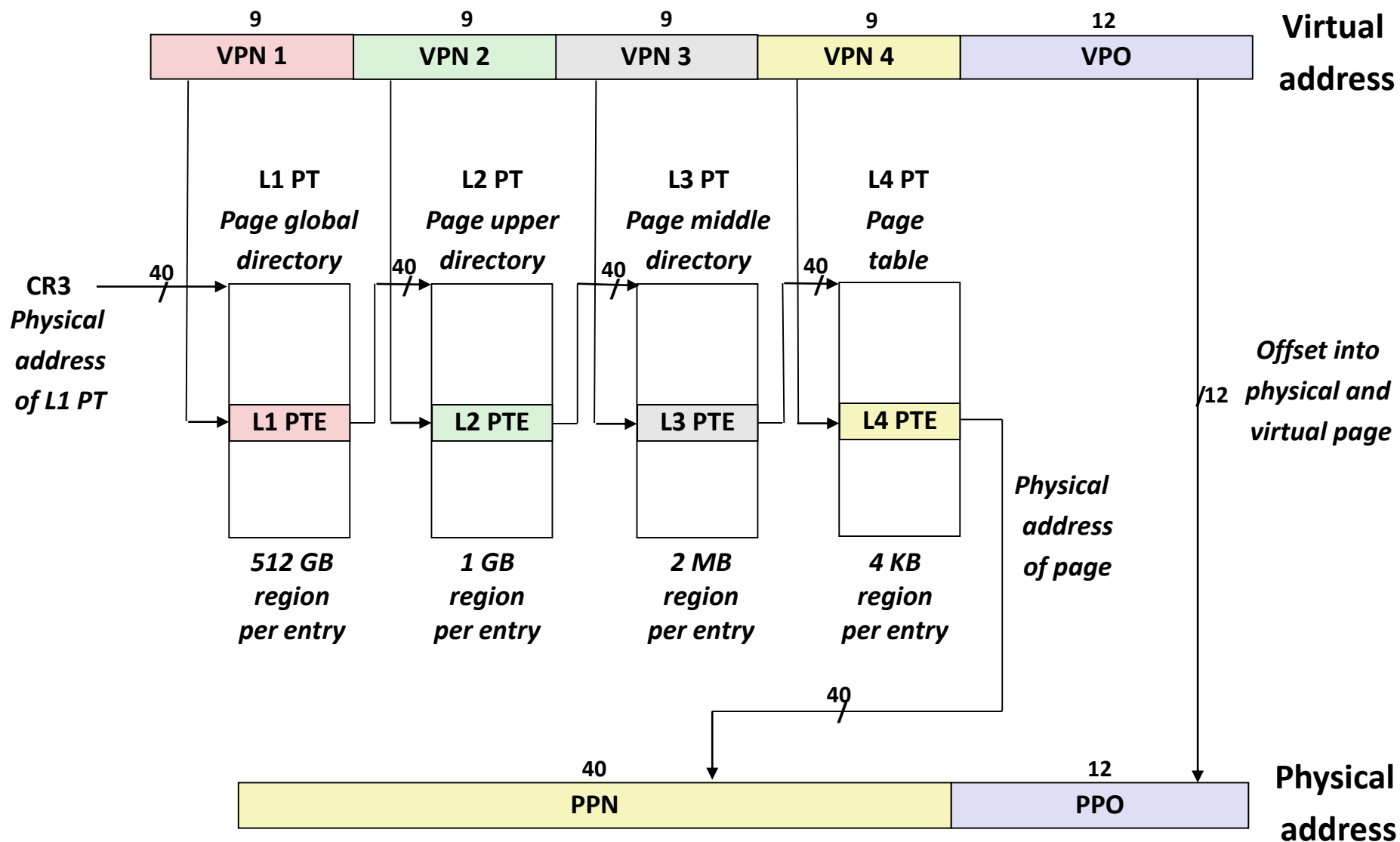
A: Reference bit (set by MMU on reads and writes, cleared by software)

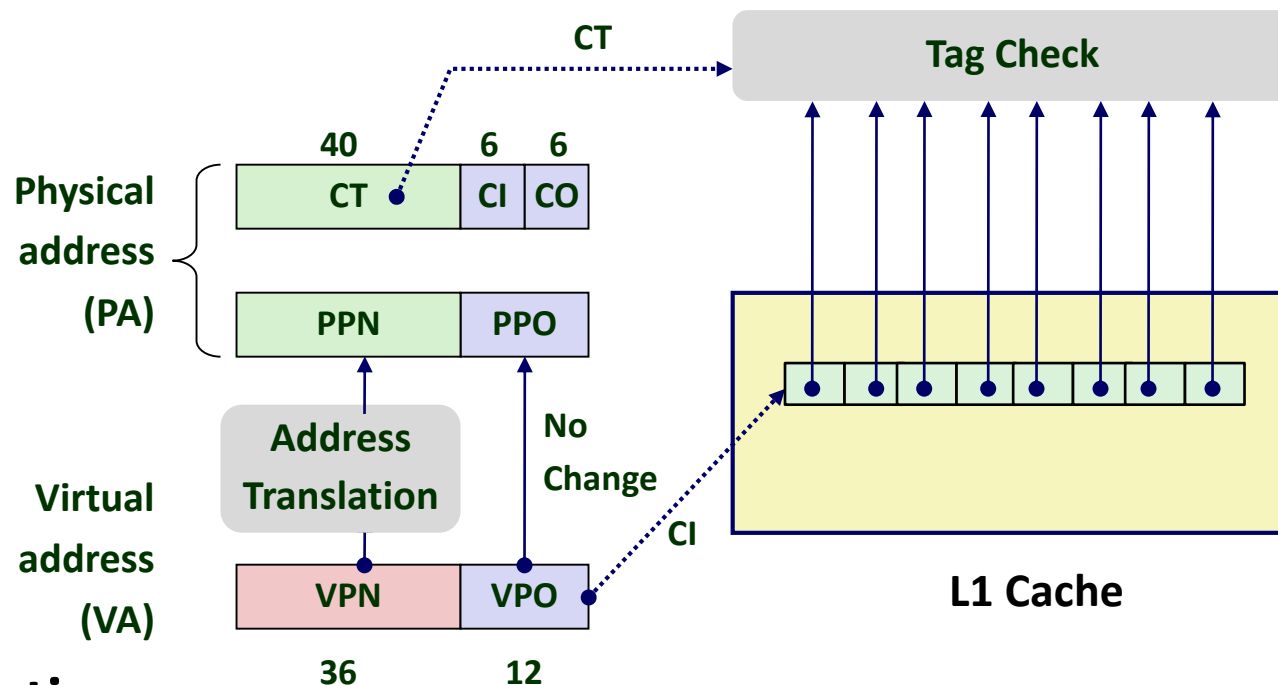**D:** Dirty bit (set by MMU on writes, cleared by software)

Page physical base address: 40 most significant bits of physical page address (forces pages to be 4KB aligned)

XD: Disable or enable instruction fetches from this page.

Core i7 Page Table Translation

9	9	9	9	12	**Virtual**
VPN 1	VPN 2	VPN 3	VPN 4	VPO	**address**

L1 PT
Page global directory

L2 PT
Page upper directory

L3 PT
Page middle directory

L4 PT
Page table

CR3
Physical address of L1 PT

40

40

40

40

Offset into physical and virtual page

L1 PTE

L2 PTE

L3 PTE

L4 PTE

Physical address of page

/12

512 GB region per entry

1 GB region per entry

2 MB region per entry

4 KB region per entry

40

40	12	**Physical**
PPN	PPO	**address**

Cute Trick for Speeding Up L1 Access

CT → **Tag Check**

40   6   6

Physical address (PA)

| CT | CI | CO |

| PPN | PPO |

Address Translation

No Change

Virtual address (VA)

| VPN | VPO |

36   12

CI

L1 Cache

- ■ **Observation**
  - ▪ Bits that determine CI identical in virtual and physical address
  - ▪ Can index into cache while address translation taking place
  - ▪ Generally we hit in TLB, so PPN bits (CT bits) available next
  - ▪ "Virtually indexed, physically tagged"
  - ▪ Cache carefully sized to make this possible