Virtual Memory Examples

Problem 1:

This problem concerns the way virtual addresses are translated into physical addresses. Imagine a system has the following parameters:

- Virtual addresses are 20 bits wide.
- Physical addresses are 18 bits wide.
- The page size is 1024 bytes.
- The TLB is 2-way set associative with 16 total entries.

The contents of the TLB and the first 32 entries of the page table are shown as follows. **All numbers are given in hexadecimal**.

TLB			
Index	Tag	PPN	Valid
0	03	C3	1
	01	71	0
1	00	28	1
	01	35	1
2	02	68	1
	3A	F1	0
3	03	12	1
	02	30	1
4	7F	05	0
	01	A1	0
5	00	53	1
	03	4E	1
6	1B	34	0
	00	1F	1
7	03	38	1
	32	09	0

Page Table					
VPN	PPN	Valid	VPN	PPN	Valid
000	71	1	010	60	0
001	28	1	011	57	0
002	93	1	012	68	1
003	AB	0	013	30	1
004	D6	0	014	0D	0
005	53	1	015	2B	0
006	1F	1	016	9F	0
007	80	1	017	62	0
008	02	0	018	C3	1
009	35	1	019	04	0
00A	41	0	01A	F1	1
00B	86	1	01B	12	1
00C	A1	1	01C	30	0
00D	D5	1	01D	4E	1
00E	8E	0	01E	57	1
00F	D4	0	01F	38	1

Part 1

1. The diagram below shows the format of a virtual address. Please indicate the following fields by labeling the diagram: (If a field does not exist, do not draw it on the diagram.)

VPO	The virtual page offset
VPN	The virtual page number
TLBI	The TLB index
TLBT	The TLB tag

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

2. The diagram below shows the format of a physical address. Please indicate the following fields by labeling the diagram: (If a field does not exist, do not draw it on the diagram.)

PPO	The physical page offset
PPN	The physical page number

17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Part 2

For the given virtual addresses, please indicate the TLB entry accessed and the physical address. Indicate whether the TLB misses and whether a page fault occurs. If there is a page fault, enter “-” for “PPN” and leave the physical address blank.

Virtual address: `078E6`

1. Virtual address (one bit per box)

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

2. Address translation

Parameter	Value	Parameter	Value
VPN	0x	TLB Hit? (Y/N)	
TLB Index	0x	Page Fault? (Y/N)	
TLB Tag	0x	PPN	0x

3. Physical address(one bit per box)

17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Virtual address: `04AA4`

1. Virtual address (one bit per box)

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

2. Address translation

Parameter	Value	Parameter	Value
VPN	0x	TLB Hit? (Y/N)	
TLB Index	0x	Page Fault? (Y/N)	
TLB Tag	0x	PPN	0x

3. Physical address(one bit per box)

17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Problem 2:

Consider the Pentium memory system (Figure 1). PTEs and PDEs are 32 bits wide.

1. How many entries can the level-1 page table hold?

2. Assume there is a single task running on the system. The task's heap area is allocated in the *physical* range `0x660000 - 0x666600`. The task's stack area is allocated in the *physical* range `0x7999400 - 0x8000000`. The task's `text` area is allocated in the *physical* range `0x1000 - 0x1400`. The task has no other sections.

   (a) How many valid PTEs are there?

   (b) How many valid PDEs are there?

   (c) How much memory is in use strictly by the Page Directory and Page Tables?

3. If the Pentium used a flat page table, how much space would that page table take up?

4. The Pentium can overlap some of the translation of a virtual address with the cache lookup for that address. If Intel wants to preserve this behavior but also double the size of the cache, how must they change the cache?

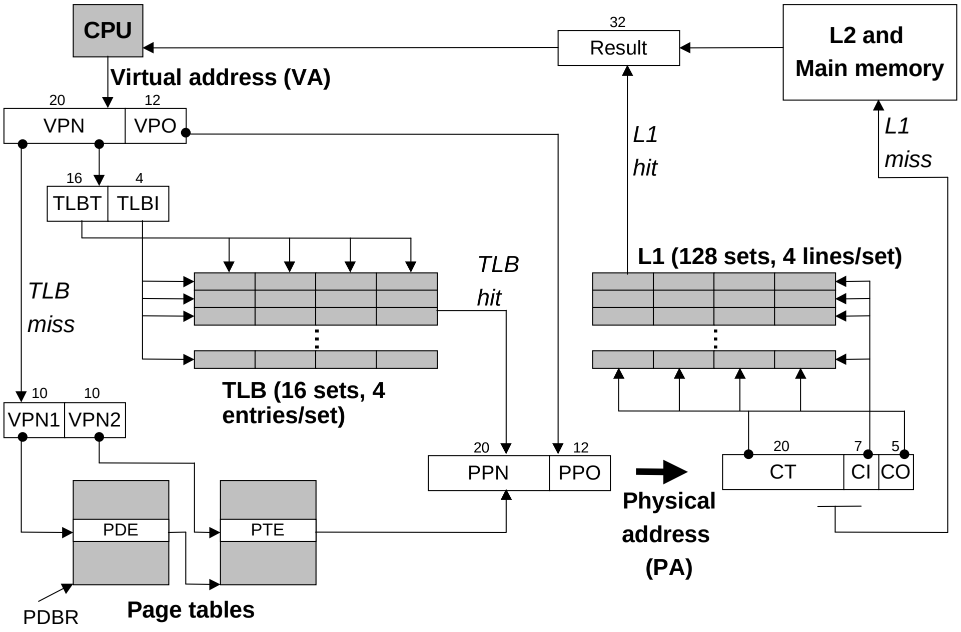

Figure 1: Summary of Pentium address translation).