

Final Exam in Computer Architecture

Answer the following questions. Total number of points is 140.

Exam period is two hours (120 minutes).

Write your answers clearly and then submit by email as PDF to wimers@biu.ac.il

Good luck!

Question 1

A system with physical memory of 128Gbyte, 38-bit virtual address space and 64-bit (8 bytes) of word length is given. Its Virtual Memory (VM) supports 8Kbyte page size.

Please answer the following questions and explain your answers.

1. (10 Pts) How many pages are in the VM space?

Answer: 38-bits VM address defines 2^{38} VM accessible bytes. With page size of $8K=2^{13}$ bytes there is a total of $2^{38}/2^{13} = 2^{25}$ pages.

2. (10 Pts) Assuming that any process can access any address of the VM, and an entry of the page table is a whole word. What is the size in bytes of a page table?

Answer: Since the VM space has 2^{25} pages, and the memory has 64-bit (8 bytes) word lengths, a page table comprises $2^{25} \times 8 = 2^{28} = 256\text{Mbyte}$.

3. (10 Pts) It is known that the average VM size of a process running simultaneously with others is 1Gbyte. What is then the average number of processes that the system can support simultaneously?

Answer: The average number of pages used by a process is $1\text{Gbyte}/8\text{Kbyte} = 2^{30}/2^{13} = 2^{17}$ pages. From 1 the entire VM has 2^{25} pages; hence the average number of processes running in the system is $2^{25}/2^{17} = 2^8 = 256$ processes.

This also follows from the VM size divided by process size $2^{38}\text{byte}/2^{30}\text{byte} = 256$ processes.

4. (10 Pts) Can all the tables in 3 be stored in the main memory?

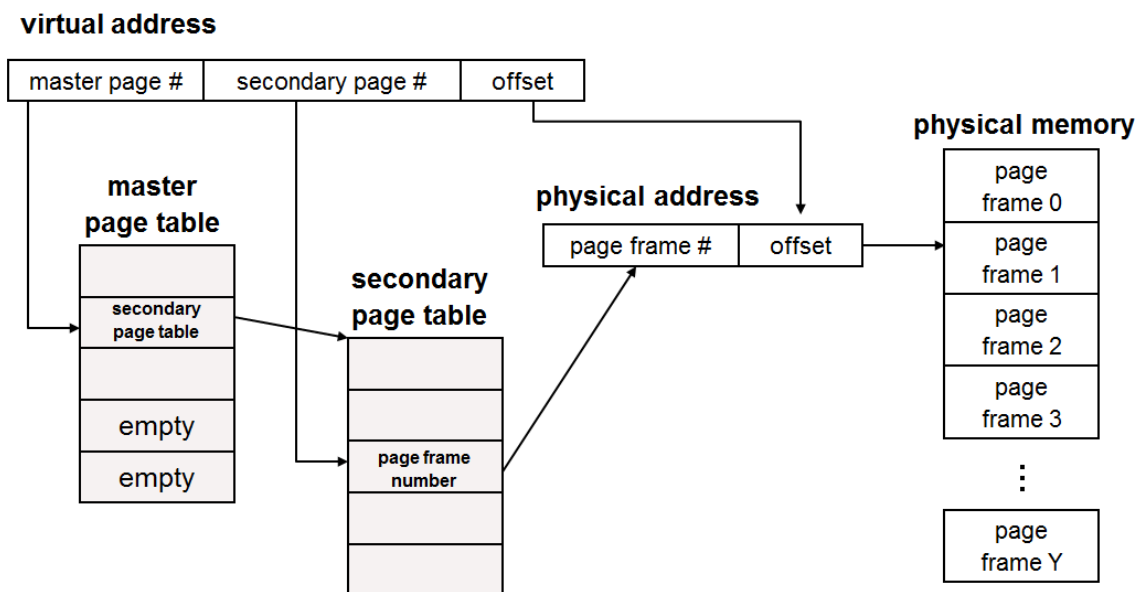
Answer: Yes. From 3 there are 256 processes on average, with table size 256Mbyte for each, yielding a total of $256 \times 256\text{Mbyte} = 2^8 \times 2^{28}\text{bytes} =$

2^{36} bytes = 64Gbyte. It turns out that half of the 128Gbyte physical memory is just page tables.

5. (10 Pts) What are the implications on user's programs performance of increasing the number of processes in 3 running simultaneously?

Answer: From 4, on the average 64Gbyte/128Gbyte = 50% of the physical memory is occupied by page table. Increasing the number of simultaneous processes will leave very little physical memory for user's program. This will enforce the operating system to perform extensive page swaps which kills user's programs performance.

6. (20 Pts) A fresh Tokyo Tech graduate engineer proposed to organize the page tables as shown below. He proposed to divide the 25-bit page index into 12-bit master page and 13-bit secondary page.
- What is the advantage of such VM organization?
 - Where in the memory hierarchy the master page tables exist and where the secondary page tables exist?



Answer:

There are two advantages. First, the decomposition of a page table of a single process into many secondary tables allows the OS to better utilize the physical memory. A contiguous memory for a page table is not required any more. Secondly, it enables to store a huge portion of the secondary page tables on the disk, thus leaving more physical memory for user's programs. Master page table must still be stored in the main memory.

Question 2

The following MIPS program increases the elements of an array by 1.

```
Loop:   LD      R2,0(R1)      ;R2=array element
        DADDIU   R2,R2,#1     ;increment R2
        SD      R2,0(R1)     ;store result
        DADDIU   R1,R1,#8     ;increment pointer
        BNE     R2,R3,LOOP    ;branch if not last element
```

A CPU with the following specifications is given:

- Supports dynamic scheduling (Tomasulo).
- Can issue in same clock cycle any two instructions.
- Can send on CDB in same clock cycle results of any two completed instructions.
- Its register file and memory are dual-ported, allowing any two read/write operations in same clock cycle.

The CPU supports the following pipeline units:

- One address calculation unit with 2 clock-cycles latency.
- One integer unit with 3 clock-cycles latency.
- One branch detection unit with 1 clock-cycle latency.

Each unit can maintain only a single instruction at a time.

Instructions can be issued even if not all preceding branches are decided, but their execution must wait for until all preceding branches are decided.

The above program is executed by the CPU. The progression of the program in the first two iterations is filled in the table below, where each pipeline stage specifies its clock cycle time.

1. (40 Pts) Complete filling the table below. Recall that store execution needs both address calculation and data ready to store in memory.

Answer: Notice the structural hazards occurring by the adder denoted by DADDIU#.

Loop #	Instruction	Issue	EX			MEM	Write CDB
			Start	End	Wait for		
1	LD R2, 0(R1)	1	2	3		4	5
1	DADDIU R2, R2, #1	1	6	8	LW		9
1	SD R2, 0(R1)	2	3	9	DADDIU	10	

1	DADDIU R1, R1, #8	2	3	5			6
1	BNE R1, R3, Loop	3	7	7	DADDIU		
2	LD R2, 0(R1)	4	8	9	BNE	10	11
2	DADDIU R2, R2, #1	4	12	14	LW		15
2	SD R2, 0(R1)	5	8	15	BNE, DADDIU	16	
2	DADDIU R1, R1, #8	5	9	11	BNE, DADDIU#1		12
2	BNE R1, R3, Loop	6	13	13	DADDIU		
3	LD R2, 0(R1)	7	14	15	BNE	16	17
3	DADDIU R2, R2, #1	7	18	20	LW		21
3	SD R2, 0(R1)	8	14	21	BNE, DADDIU	22	
3	DADDIU R1, R1, #8	8	15	17	BNE, DADDIU#2		18
3	BNE R1, R3, Loop	9	19	19	DADDIU		
4	LD R2, 0(R1)	10	20	21	BNE	22	23
4	DADDIU R2, R2, #1	10	24	26	LW		27
4	SD R2, 0(R1)	11	20	27	BNE, DADDIU	28	
4	DADDIU R1, R1, #8	11	21	23	BNE, DADDIU#3		24
4	BNE R1, R3, Loop	12	25	25	DADDIU		

2. (15 Pts) What is the effective CPI?

Answer: Iteration ends when the item is stored back into the array. From the 3rd iteration on the distance between two stores is 6 clock cycles, where the loop comprises 5 instructions. Hence $CPI=1.2$.

3. (15 Pts) Adding the pipeline any decode and execution units as needed, what is the smallest possible CPI?

Answer: The instruction completion is limited by:

Issue rate = $5/3$, and completion rate = 2, so the bottleneck is the issue rate. Hence smallest $CPI = 3/5 = 0.6$.