컴퓨터구조 기말고사 (80점 만점)

2014. 12. 16

1. [표 1]은 LRU replacement algorithm을 사용하는 virtual memory system의 page table 초기상태이다. Page size는 2KB=2^{11}B이고, physical page 0에 들어 있는 page가 사용한지 가장 오래 된 것이고, 그 다음에 1, 2에 있는 것, physical page 3에 들어 있는 page가 가장 최근에 사용한 것이다.
   (1) Physical page 1에는 virtual address 몇 번지부터 몇 번지까지가 들어가 있는가? 16진수로 답하라. (1점)
   (2) [표 2]를 완성하라. 단 Page fault일 때만 Y로 표기하고, page number는 10진수로 표시하라. (6점)
   (3) [표 2]를 다 실행한 후의 final page table을 [표 3]에 보여라. (4점)

2. 1번 문제와 같은 조건인데, page size가 256B=2^8B이고 fully-associative TLB가 있다. TLB의 초기상태는 [표 4]와 같고, TLB도 LRU를 사용한다.
   (1) 다음 reference string을 실행한 후 TLB의 최종 상태를 보여라. (4점)
   338, 770, 0C4, 788, CA0, 5F0
   (2) 최종적으로 main memory에 남아 있는 virtual page들의 번호는? (2점)
   (3) TLB miss를 발생시키는 virtual addresses는? (2점)
   (4) Page fault를 발생시키는 virtual addresses는? (2점)

Valid	Tag	PPN
1	7	1
0		

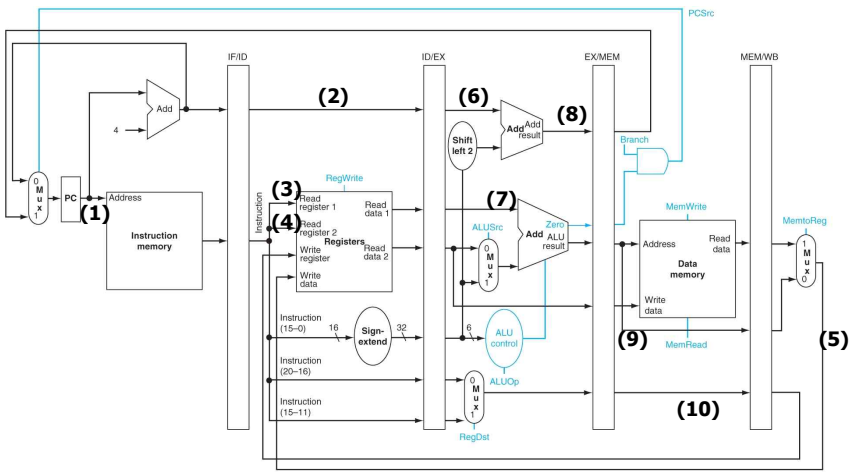
[표 4] TLB

3. [표 5]는 block size=8 bytes인 2-way set associative data cache의 초기상태이다. 이 컴퓨터의 replacement policy는 Tag 값이 **작은** 것을 replace하는 것이다.
   (1) 각 data 부분에 memory 몇 번지의 내용이 들어있는지 M[050]~M[051]과 같이 [표 5]에 표시하되 주소는 16진수로 표기하라. Valid한 data가 없을 때는 빈 간 그대로 둔다. (3점)
   (2) [표 6]을 완성하라. 단 H/M는 Hit일 때만 H로 표시하라. (6점)
   (3) [표 7]에 cache의 최종 상태를 보이되, [표 5]와 달라진 부분만 표시하라. (2점)

4. [프로그램 1]이 [그림 1]의 pipeline에서 실행된다. addi 명령어가 WB stage에 있을 때 (1)~(10)의 값을 십진수로 표시하라. 단 addi 명령어는 308 번지에 있으며, memory m 번지 ($0 \leq m < 80_{hex}$)의 값은 m*2이고, register $r($0<r<32_{ten}$)에는 r+1이 저장되어 있다고 하자. Hazard는 없다고 가정한다. (각 0.5점)

```
addi  $1,$2,$3
add   $3,$4,$5
beq   $6,$7,-3
sw    $8,50($9)
slt   $10,$11,$12
```

[프로그램 1]



[그림 1]

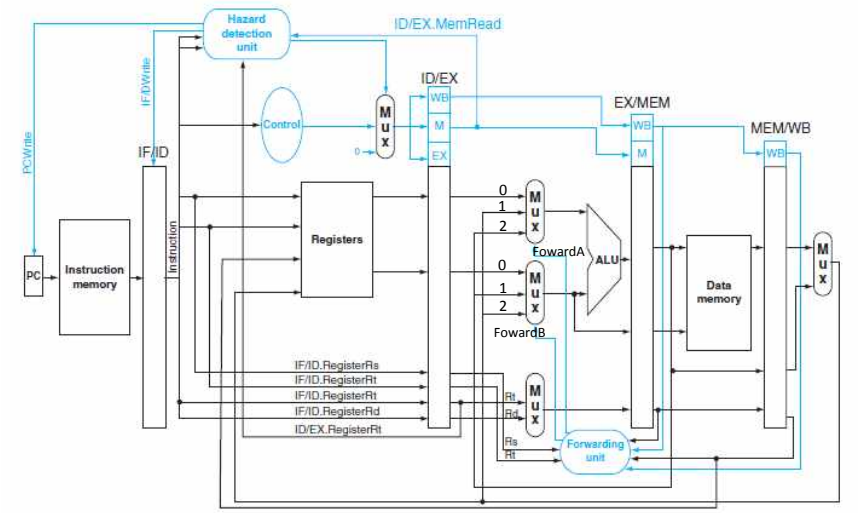
5. [프로그램 2]가 5-stage MIPS pipeline에서 실행될 때, (각 2점)
   (1) 이 프로그램에 존재하는 data hazards의 종류(1a, 1b, 2a, 2b)를 모두 말하라. 각 hazard와 관련된 명령어 쌍도 함께 보일 것.
   (2) Forwarding unit이 없을 때, 제대로 실행하기 위해서는 어느 명령어와 어느 명령어 사이에 최소 몇 개의 bubble을 삽입해야 하는지 모두 써라.
   (3) Forwarding unit이 있을 때, 제대로 실행하기 위해서는 어느 명령어와 어느 명령어 사이에 최소 몇 개의 bubble을 삽입해야 하는지 모두 써라.

```
add  $1,$2,$3
or   $2,$3,$4
slt  $5,$1,$6
lw   $7,31($8)
sub  $9,$4,$7
and  $11,$12,$5
```

[프로그램 2]

6. [프로그램 2]가 5-stage MIPS pipeline에서 실행될 때, 다음 각 경우에 pipeline의 각 stage에 있는 명령어의 Opcode는? bubble이 있는 경우에는 (bubble)로 표시하고, 알 수 없는 경우는 빈 간으로 둔다. (각 2점)
   (1) Forwarding unit은 없고, or 명령어가 WB stage에 있을 때
   (2) Forwarding unit은 없고, lw 명령어가 WB stage에 있을 때
   (3) Forwarding unit이 있고, add 명령어가 WB stage에 있을 때
   (4) Forwarding unit이 있고, slt 명령어가 WB stage에 있을 때

7. [프로그램 2]가 [그림 2]와 같이 forwarding unit과 hazard detection unit이 있는 5-stage MIPS pipeline에서 실행될 때, (각 1점)
   (1) `slt` 명령어가 EX stage에 있을 때, ForwardA와 ForwardB의 값은?
   (2) `sub` 명령어가 EX stage에 있을 때, ForwardA와 ForwardB의 값은?



[그림 2]
Hazard detection unit과 Forwarding unit

8. [프로그램 3]이 5-stage MIPS pipeline에서 실행될 때, 다음 각 경우의 실행 과정을 multicycle 또는 single-cycle pipeline diagram으로 보여라. 단 `lw`가 IF stage에 있을 때를 CC0라 하고, CC2~CC12을 보여라. (각 4점)
   (1) Branch는 EX stage에서 실행되며, assume-branch-not-taken prediction이고 no delay slot.
   (2) Branch는 ID stage에서 실행되며, assume-branch-not-taken prediction이고 delay slot 하나 사용.
   (3) Branch는 ID stage에서 실행되며, not-taken으로 초기화 되어 있는 1-bit predictor를 갖는 branch history table 사용. no delay slot.

```
XX:  lw   $1,40($6)
     beq  $2,$3,YY  ; At first taken,
     add  $4,$5,$6  ; then not taken
     sub  $7,$8,$9
YY:  slt  $12,$13,$14
     bne  $10,$11,XX ; Taken
     and  $15,$16,$17
     or   $18,$19,$20
     xor  $23,$24,$25
     sll  $26,$27,3
```

[프로그램 3]

9. (1) Programmed I/O와 interrupt-driven I/O의 가장 큰 차이점은? (2점)
   (2) RAID-3와 RAID-5의 차이점 두 가지는? (2점)
   (3) 현재 service 받고 있는 것보다 우선순위가 낮은 interrupt들을 금지시키기 위해 사용하는 것은? (1점)

10. Virtual address=35 bits, Physical memory=1G Bytes, Page size=4K Bytes
    Physically indexed and physically tagged cache, cache size=256K Bytes(data 부분만), cache block=32 Bytes
   (1) Virtual page number는 몇 bits인가? (1점)
   (2) Physical page number는 몇 bits인가? (1점)
   (3) Page table의 크기는 몇 bits인가? Page table entry에는 valid bit과 page number만 있다. (2점)
   (4) Direct cache라면 cache tag는 몇 bits인가? (2점)
   (5) 8-way set associative cache라면 number of sets는? (2점)
   (6) Fully associative cache라면 cache 전체의 크기는? (2점)