

컴퓨터구조 중간고사 (73점 만점)

2018. 10. 24

1. A, B, C 세 종류의 명령어가 있는 컴퓨터에서 어떤 프로그램을 실행시켰다. 각 명령어의 CPI와 실행된 명령어 개수가 오른쪽 표와 같을 때 (계산 과정을 보일 것)

Instruction class	A	B	C
CPI	3	4	6
실행 명령어 개수	4×10^8	3×10^8	2×10^8

- (1) IC(instruction count)는? (1점)
 (2) 평균 CPI는? (1점)
 (3) 2 GHz clock을 사용한다면 실행시간은? (1점)
 (4) 하드웨어를 개선해서 C 명령어를 5 clocks에 실행하게 바꾸었다. 이때의 speedup은? (2점)
2. Multiplicand = 110001_{two}, Multiplier = 110011_{two} 이다.
 (1) 두 수를 교재의 최종 algorithm으로 곱할 때 덧셈과 뺄셈은 각각 몇 번씩 하는가? 답만 쓸 것. (2점)
 (2) 두 수를 Booth's algorithm으로 곱할 때 덧셈과 뺄셈은 각각 몇 번씩 하는가? 답만 쓸 것. (3점)
3. Dividend = 00000 11010_{two}, Divisor = 00100_{two} 이다.
 (1) 두 수를 교재의 최종 algorithm(restoring division)으로 나누는 과정을 lecture slide와 같은 표로 보여라. (5점)
 (2) 두 수를 nonrestoring division algorithm으로 나누는 과정을 lecture slide와 같은 표로 보여라. (5점)
4. IEEE 754 single precision floating point format의 지수부분은 8 bits이고 bias는 127이다. 유효자리는 23 bits이고 hidden bit을 사용한다. 28.25_{ten}를 IEEE 754 single precision floating point format으로 표현하되, 16진수 8자리로 표시하라. 과정을 보일 것. (5점)

5. 다음 각 명령어를 assembler가 번역한 후 생성되는 MIPS 기계어를 **16진수 8자리**로 나타내라. Assemble error가 발생한다면 그 이유를 설명하라. 답만 쓸 것. (각 2점)
 (1) and \$5,\$6,\$7 (and는 function code=100100)
 (2) addi \$1,\$2,3 (andi는 opcode=001000)
 (3) sll \$16,\$17,5 (sll은 function code=000000)
 (4) sw \$2,7(\$zero) (sw는 opcode=101011)
 (5) beq \$1,\$3,AA (이 명령어는 200_{ten} 번지에 있고 AA는 216_{ten}번지이다. beq는 opcode=000100)
 (6) bne \$1,\$3,BB (이 명령어는 0x03FA0200 번지에 있고 BB는 0x03000000번지이다. bne는 opcode=000101)
 (7) jal CC (이 명령어는 0x00003000 번지에 있고 CC는 0x70006000 번지이다. jal은 opcode=000011)
 (8) j DD (이 명령어는 0x70000000 번지에 있고 DD는 0x70006000 번지이다. j는 opcode=000010)

6. MIPS CPU를 갖는 컴퓨터에서 메모리 m 번지($0 \leq m < FE_{hex}$)의 값은 m+2이고, register \$r($0 \leq r \leq 31_{ten}$)에는 r*2가 저장되어 있다고 하자. 다음 각 명령어의 **실행 결과를 설명하되**, "몇 번 register 혹은 memory 몇 번지 내용이 얼마로 바뀐다."와 같이 표현하라. PC가 PC+4가 아닌 다른 값으로 바뀔 때는 "PC가 얼마로 바뀐다."로 표기하라. 만일 이 명령어를 실행할 수 없다면 그 이유를 설명하라. (6)번을 제외하고 이 문제의 숫자는 모두 2진수이다. 답을 쓸 때 10진수는 그냥 쓰면 되지만, 16진수를 사용할 때는 접두어 0x를 붙일 것. 답만 쓸 것. (각 2점)

- (1) opcode = 000000, function code = 100000 이면 add 이다.

000000	00001	00010	00000	00000	100000
--------	-------	-------	-------	-------	--------

- (2) opcode = 100011 이면 lw(load word) 이다. Little-endian을 가정하라.

100011	00011	00010	0000 0000 0000 0011
--------	-------	-------	---------------------

- (3) opcode = 100011 이면 lw(load word) 이다. Big-endian을 가정하라.

100011	00101	00001	0000 0000 0000 0010
--------	-------	-------	---------------------

(4) opcode = 101011 이면 sw(store word) 이다. Big-endian을 가정하라.

101011	00100	01100	0000 0000 0000 0010
--------	-------	-------	---------------------

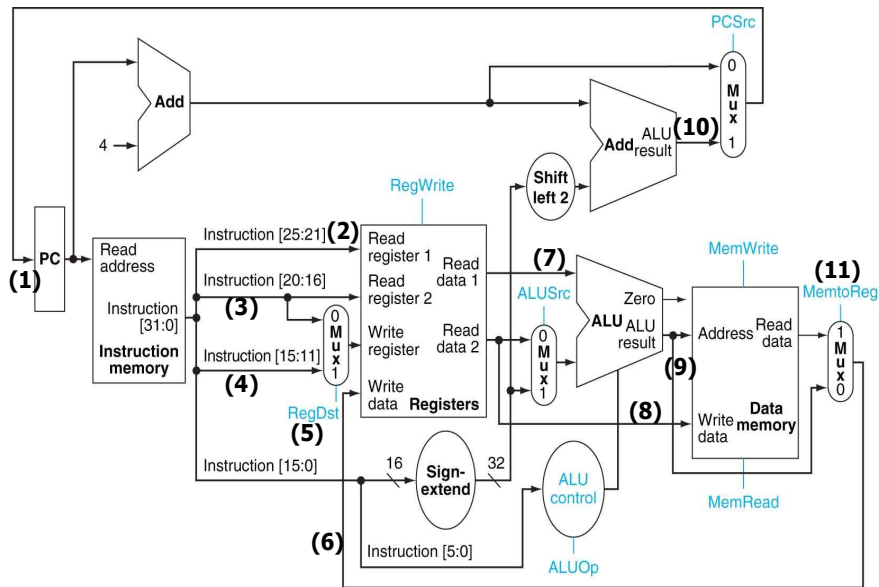
(5) opcode = 101011 이면 sw(store word) 이다. Little-endian을 가정하라.

101011	00111	00010	0000 0000 0000 0010
--------	-------	-------	---------------------

(6) opcode = 000011 이면 jal(jump and link) 이고, 이 명령어는 0xA6000050 번지에 있다.

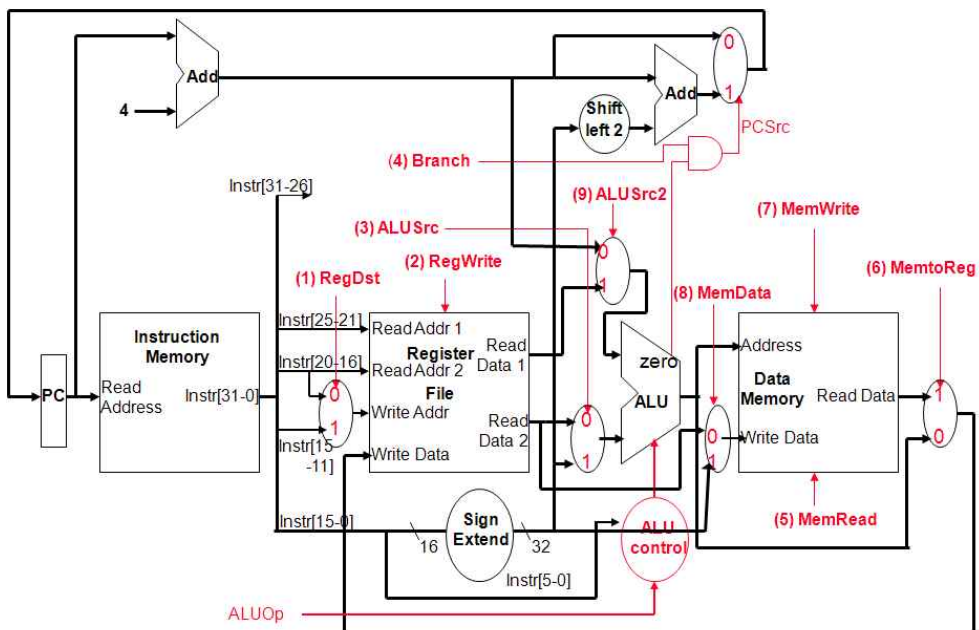
000011	1000 0000 0000 0000 0000 0000 00
--------	----------------------------------

7. [그림 1]의 MIPS datapath에서 00000030_{hex}번지에 저장되어 있는 명령어 addi \$5,\$3,7을 실행할 때 (1)~(11)의 값을 **16진수**로 표시하라. 부득이 10진수를 사용할 때는 아랫첨자 _{ten} 사용. (5)와 (11)은 제어신호이고, register와 memory의 내용은 문제 6번과 같다. 어떤 값이 될지 알 수 없는 경우에는 ?로 표시하라. 답만 쓸 것. (각 1점)



[그림 1]
MIPS datapath

8. [그림 2]는 MIPS의 single-cycle datapath를 수정한 것이다. 이 datapath에서 다음과 같이 동작하는 새로운 명령어를 실행시키려면 다음 제어신호 (1)~(9)의 값은 어떻게 되어야 하는가? Don't care는 X로 표시하라. 답만 쓸 것. (각 1점)
- $$\text{Reg}[rt] \leftarrow \text{Mem}[(PC+4)+\text{sign-extended}(\text{offset})]$$



[그림 2] Modified
MIPS datapath