

학과		학번		이름	
----	--	----	--	----	--

학번 마지막 번호의 인덱스에 해당하는 행의 문제를 푼다.

	푸는 문제 번호		푸는 문제 번호
0	1, 4, 5, 8, 9, 11, 13, 16	5	2, 4, 6, 8, 9, 12, 13, 16
1	1, 2, 6, 7, 10, 11, 14, 15	6	1, 3, 5, 7, 9, 11, 13, 16
2	1, 3, 4, 7, 9, 12, 14, 15	7	1, 2, 3, 5, 10, 12, 14, 15
3	3, 6, 7, 8, 10, 12, 13, 16	8	1, 2, 4, 7, 9, 12, 14, 15
4	1, 2, 6, 7, 9, 11, 13, 15	9	1, 2, 3, 6, 10, 11, 14, 16

1. 다음의 두 boolean expression에 대하여 물음에 답하라. [총 15점]

$$(x + y) \cdot (x + z) = x + y \cdot z$$

1) 두 식이 동일한 boolean expression임을 truth table을 이용하여 증명하라. [4점]

--

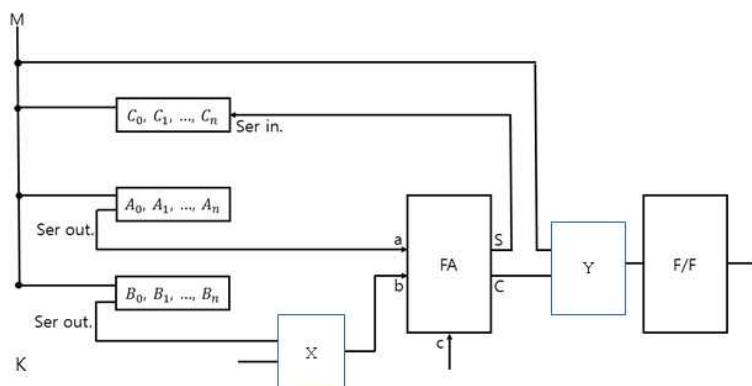
2) 두 식이 동일한 boolean expression임을 boolean algebra를 이용하여 증명하라. [5점]

--

3) 좌변과 같은 수식을 우변과 같은 수식으로 최적화하는 장점을 6가지 제시하라. [각 1점]

2. 두 shift register(A, B)에 저장된 수의 덧셈과 뺄셈을 수행하여 결과를 shift register(C)에 저장하는 Binary Serial Adder(BSA)를 Full Adder와 Flip-Flop을 이용하여 만들려고 한다. M은 shift left 기능에 대한 제어선을 의미하며, K는 0이면 덧셈, 1이면 뺄셈을 수행하게 하는 제어선을 의미한다. [총 15점]

1) 아래 배치도에서 비어 있는 연결 부분을 완성하라.[3점]



2) 두 개의 빈칸(X, Y)에 들어갈 게이트 명을 써라. [각 2점]

X		Y	
---	--	---	--

3) BSA가 수행되려면 Flip-Flop의 초기 값에 대한 어떤 가정이 필요한가? [4점]

--

4) Full Adder의 출력 값 S와 C를 계산하는 식을 작성하라. (입력 값은 a, b, c로 표현) [각 2점]

S	
C	

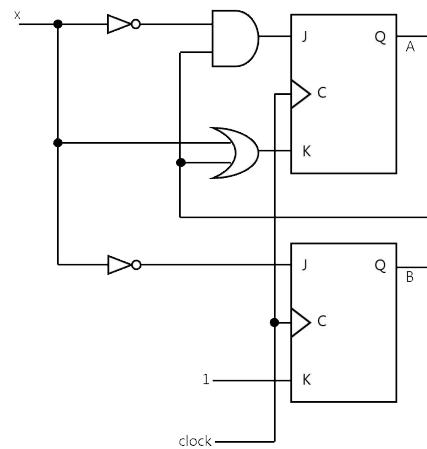
3. 4개의 register가 common bus로 연결된 디지털 컴퓨터가 있다. 각 register는 16 bit 크기를 가지며 common bus는 MUX(multiplexer)와 디코더로 구성되어 있다. [총 15점]

- (1) 각 MUX에는 몇 개의 selection input이 필요한가? [3점]
- (2) 어떤 종류의 MUX가 필요한가? (예를 들어, 16×1) [3점]
- (3) Common bus를 구성하기 위해 몇 개의 MUX가 필요한가? [3점]
- (4) 어떤 종류의 디코더가 필요한가? [3점]
- (5) 마이크로프로세서가 생기면서 상대적으로 간단하게 버스시스템을 구성할 수 있다. 이 경우 어떤 방법으로 버스시스템을 만들 수 있는가? [3점]

(1)		(2)		(3)
(4)		(5)		

4. 오른 쪽에 주어진 그림은 두 개의

JK flip-flop A, B와 external input x를 이용하여 어떠한 동작을 수행하는 회로를 나타낸다. 이 회로가 어떻게 동작하는지를 조사하기 위해 sequential circuit을 설계하는 과정을 거꾸로 따라가 보려고 한다.
물음에 답하여라. [총 15점]



- 1) 문제에 주어진 회로에서 flip-flop A, B의 입력 값 J , K 를 boolean function 식으로 각각 표현 하라. [각 1점]

J_A	
K_A	
J_B	
K_B	

- 2) 1)에서 도출한 boolean function 식을 근거로 하여 아래에 주어진 state transition table의 next state 부분과 flip-flop inputs 부분을 채우시오. [4점]

Present state	Input	Next state		Flip-flop inputs						
		A	B	x	A	B	J_A	K_A	J_B	K_B
0	0	0								
0	0	1								
0	1	0								
0	1	1								
1	0	0								
1	0	1								
1	1	0								
1	1	1								

- 3) 2)의 table에서 도출한 present state와 next state, x 정보를 참고로 하여 해당 회로의 state diagram을 그리시오. [3점]

- 4) 결론적으로 문제에서 주어진 회로는 어떠한 기능을 수행하는가? $x = 0$ 일 때와 $x = 1$ 일 때 회로가 수행하는 기능을 아래 빈 칸에 작성하시오. [각2점]

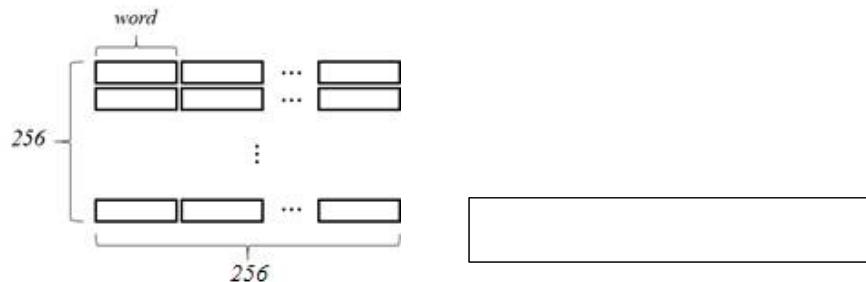
$x = 0$	
$x = 1$	

5. 아래 물음에 답하시오. [총 15점]

- 1) 64K-word RAM($2^{16} \times 16$)을 위한 주소는 최소 몇 비트가 필요한가? [4점]

- 2) 위 RAM에 direct(random) access 하기 위한 회로를 4×16 decoder만으로 만들기 위해서는 총 몇 개의 4×16 decoder가 필요한가? [4점]

- 3) 아래 그림과 같이 특정 word의 주소가 (x, y) 의 형태를 지니는 2차원 addressing 구조로 위 RAM을 설계하였을 때, 총 몇 개의 4×16 decoder가 필요한가? (단, 4×16 decoder만을 사용한다고 가정) [4점]



- 4) word의 주소가 (x, y, z) 로 표현되는 3차원 addressing 구조로 위 RAM을 설계하고자 한다. 필요한 4×16 decoder의 개수가 최소가 되도록 RAM을 구성한다고 했을 때, 주소 비트 x, y, z 의 크기는 얼마가 되는가? (단, 4×16

decoder만을 사용한다고 가정하며, $x >= y >= z$ 가 되도록 한다) [각 1점]

x		y		z	
---	--	---	--	---	--

6. 부동소수점(Floating point) 표현법은 사인부(Sign field), 지수부(Exponent Field), 유효숫자부(Mantissa Field) 세 개의 부분으로 이루어진다.[15점]

(1) 유효숫자부의 범위를 표현하는 다음 칸을 채워라.[각 1점]

	<=	M	<	
--	----	---	---	--

(2) 사인부의 값을 s, 지수부(의 값을 e, 유효숫자부의 값을 M으로 가정하였을 때, 나타내는 수를 위 세 개의 값으로 위 세 변수(s, e, M)로 표현하라. 단 지수부의 베이스는 2로 한다. [2점]

--

(3) 각 부분이 크기가 늘어나는 경우, 발생되는 효과를 기술하라.[각2점]

지수부	
유효숫자부	

(4) 지수부(Exponent Field)의 크기를 3 비트 e, 유효숫자부 (Mantissa Field)의 크기를 4비트로 가정하고, 지수부의 베이스를 2로 하였을 때, 0(Zero)은 어떻게 정의하는가?[3점]

--

(5) (3)번의 경우로 가정하고, 0.75 더하기 0.55의 연산을 이진수로 수행하는 것을 보여라.[4점]

--

7. 현대의 컴퓨터에서 정수를 표현하는 방법에는 Ⓐ Signed Magnitude Ⓑ 1의 보수 (1's Complement) Ⓒ 2의 보수 (2's Complement)와 같이 세 가지 방법이 있다 [15점]

(1) n개 비트로 이루어진 정수를 위 세 가지 방법으로 표현하였을 때, 각 방법으로 표현할 수 있는 수의 범위를 써라. [각 줄 1점]

(A)		~	signed magnitude	~	
-----	--	---	------------------	---	--

(B)		~	1's complement	~	
-----	--	---	----------------	---	--

(C)		~	2's complement	~	
-----	--	---	----------------	---	--

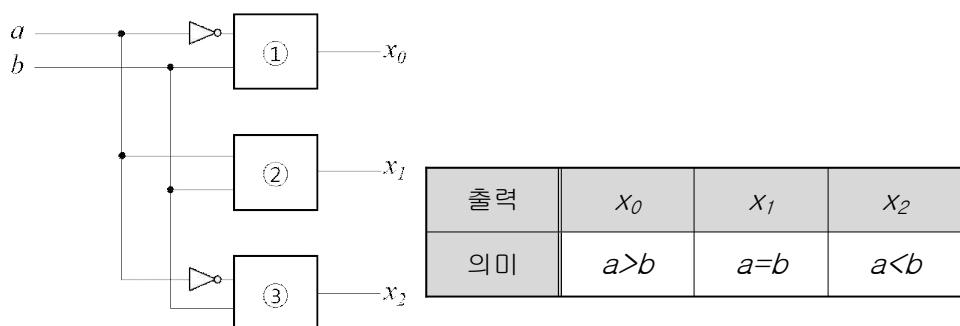
(2) n이 4로 가정하여, 7 빼기 2의 연산과정을 각 방법으로 보여라

(A)	2점	(B)	2점
-----	----	-----	----

(C)	2점
-----	----

(3) (C)의 방법이 (A) (B) 방법보다 더 좋은 점을 세 가지 기술하라 [각 1점]

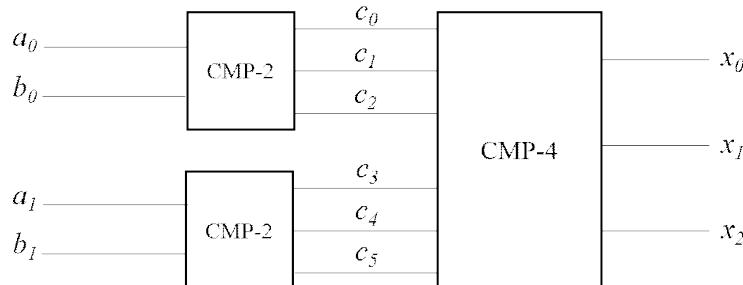
8. 2개의 input(a, b)에 대하여 다음과 같은 세 개의 output(x_0, x_1, x_2)를 출력하는 combinational circuit CMP-2를 설계하였을 때, 아래 물음에 답하시오. [15점]



(1) ①, ②, ③에 들어갈 gate명을 쓰시오. [각 1점]

①		②		③
---	--	---	--	---

(2) (1)번에서 설계된 두 개의 CMP-2를 이용하여, 2-bit 크기의 두 수 $A(a_1, a_0)$, $B(b_1, b_0)$ 를 비교한 뒤 $x_0(A>B)$, $x_1(A=B)$, $x_2(A<B)$ 를 출력하는 CMP-4를 아래 그림과 같이 설계하였을 때, x_0 , x_1 , x_2 를 $c_0 \sim c_5$ 로 표현하시오. [각 2점]



\mathbf{x}_0		\mathbf{x}_1		\mathbf{x}_2
----------------	--	----------------	--	----------------

(3) 위 회로와 같은 방식으로, 32-bit 크기의 두 수 $A(a_{31} \dots a_0)$, $B(b_{31} \dots b_0)$ 를 비교한 뒤 세 개의 output $x_0(A>B)$, $x_1(A=B)$, $x_2(A<B)$ 를 출력하는 회로를 설계하였을 때, 필요한 CMP-2와 CMP-4의 개수를 써라. [각 2점]

CMP-2		CMP-4
-------	--	-------

(4) 위 회로가 없는 경우, 동일한 기능을 수행할 수 있는 방법을 쓰라. [2점]

--

9. 아래의 문장들에 대하여 O/X로 답하라(각 1점).

- (1) NAND(Not AND) gate의 한쪽 입력 값을 1로 고정하면 출력은 항상 다른 쪽 입력 값의 Complement가 된다.
- (2) ROM(Read-Only Memory)은 메모리 내의 특정 word에 접근하는데 소요되는 시간이 항상 동일하다.
- (3) ROM(Read-Only Memory)은 Flip-Flop이 포함될 수 없는 Combinational Circuit이다.
- (4) 2의 보수를 shift left 연산할 때 overflow는 serial output으로 알아낼 수 있다.
- (5) Gray-code는 각 비트의 위치가 값을 지닌 weighted code이다.
- (6) Demultiplexer(DeMUX)를 따로 설계할 필요 없이 Multiplexer(MUX)의 입력과 출력을 바꾸어서 사용하면 된다.
- (7) 레지스터들을 serial로 연결하려면 레지스터들이 shift 레지스터여야 한다.
- (8) 버스를 MUX와 DEC를 이용하여 설계할 때, MUX의 개수는 레지스터의 크기에 따라

라 달라진다.

(9) PLA(Programmable Logic Array)를 이용하여 모든 Circuit을 설계할 수 있다.

(10) 전송 중 한 비트가 변환 확률을 $\frac{1}{2}$ 이라고 할 때, Parity Bit를 포함하여 4비트를 전송하는 경우 에러를 검출하지 못할 확률은 $\left(\frac{1}{2}\right)^4 = \frac{1}{16}$ 이다.

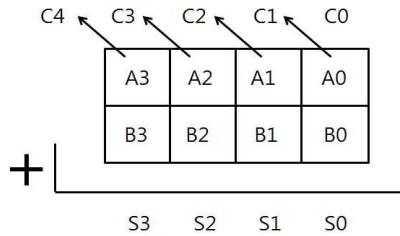
1	2	3	4	5	6	7	8	9	10

10. 아래의 문장들에 대하여 O/X로 답하라(각 1점).

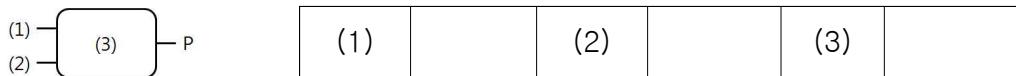
- (1) 하나의 Truth table에 대응되는 Boolean expression은 하나이다.
- (2) RAM(Random Access Memory)는 메모리 내의 특정 word를 access하기 위한 Decoder를 포함하는 Combinational Circuit이다.
- (3) Floating point 표현의 경우 Mantissa field의 크기를 늘리면 두 정수 사이에 표현 할 수 있는 수의 가지 수가 늘어나게 된다.
- (4) Arithmetic shift left에서 serial input은 sign bit, serial output은 2로 나눈 나머지이다.
- (5) 12×4096 디코더를 4×16 의 디코더들만으로 설계가 가능하다.
- (6) Excitation table은 flip-flop의 상태 변화를 위한 입력 값을 정의한다.
- (7) 3-state buffer gate는 일정 시간의 지연을 필요로 하는 경우에 사용할 수 있다
- (8) A 레지스터에 4, B 레지스터의 모든 비트들은 10이 저장된 상태에서 두 레지스터를 더한 결과는 30이 된다.
- (9) RTL(Register Transfer Language)로 작성된 프로그램을 수행한 결과는 하드웨어 회로도이다.
- (10) A 레지스터의 특정 비트 값을 1로 만들려면, B 레지스터의 해당 위치에 1을 주고 나머지 비트들은 0을 저장한 후에 두 레지스터를 XOR한다.

1	2	3	4	5	6	7	8	9	10

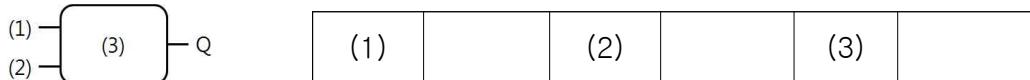
11. 두 개의 binary number에 대한 덧셈을 수행할 때, overflow의 발생 여부는 더 하려는 두 수가 signed number인지, unsigned number인지에 따라 달라질 수 있다. 다음 그림은 4bit 크기의 두 binary number A, B에 대한 덧셈을 수행하는 과정을 나타낸다. 3번 위치는 most significant bit 위치를 나타낸다. C1이 0번 위치에서 1번 위치로 올라가는 output carry를 나타낸다고 하였을 때, 다음 질문에 답하여라. [총 10점]



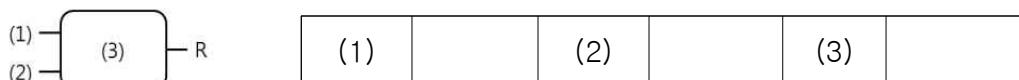
- (1) 두 숫자가 unsigned number일 때, 덧셈을 수행한 후 overflow detection을 할 수 있는 회로를 설계한다고 하자. output P는 overflow가 발생했을 때 1을 출력 한다. 빈칸을 채우시오. [각 1점]



- (2) 두 숫자가 signed number일 때, 덧셈을 수행한 후 overflow detection을 할 수 있는 회로를 설계한다고 하자. output Q는 overflow가 발생했을 때 1을 출력 한다. 빈칸을 채우시오. [각 1점]



- (3) 현재 A가 4bit 크기의 레지스터 안에 저장되어 있다고 하자. A가 저장된 레지스터에 대해 Arithmetic Shift Left 연산을 수행할 때, overflow detection을 할 수 있는 회로를 설계하려고 한다. output R은 overflow가 발생했을 때 1을 출력 한다. 빈칸을 채우시오. (단, 해당 회로는 Arithmetic Shift Left 연산을 수행하기 전에 동작한다.) [각 1점]



- (4) A 레지스터를 shift left 할 때도 Overflow가 발생할 수 있다. 0) overflow는 어떻게 알아 낼 수 있는가? [1점]

--

12. 아래 글을 읽고 물음에 답하라[10점].

The process of communication is done by programmed control transfer which is implemented by ①the loop of instructions in Mano's architecture. ②The computer keeps checking the flag bits for input and output alternately, and

when if finds one of them set, it initiates an information transfer. The difference of information flow rate between the computer and that of the input-output devices makes this type of transfer inefficient. To see why this is inefficient, consider a computer that can executes ③10 Mega instructions per second. Usually man can types keyboard at a maximum rate of 10 ④ CPS. The printer can transfer information at a maximum rate of 3 ⑤LPM with 20 characters per line. This means that at the maximum rate, the computer will check the FGI or FGO flags ⑥ times between each transfer. The computer is ⑦wasting time while checking these flags instead of doing some other useful processing task.

(1) ①에 해당하는 입력을 위한 프로그램 모듈을 작성하라.[2점]

(2) ②에 해당하는 용어를 무엇이라고 하는가?[1점]

(3) ③을 근거로 instruction cycle time을 계산하라. [1점]

(4) ④의 약자가 아닌 전체 용어로 작성하라.[1점]

(5) 한 개의 문자를 입력하는 동안 몇 번의 instruction cycle이 수행되는가?[2점]

(7) ⑦처럼 하나의 글자를 입출력하기 위하여 낭비되는 instruction cycle은 얼마인가 ?[2점]

(8) 이런 낭비를 줄이는 방법은 무엇인가?[1점]

13. 다음 프로그램은 원소가 4인 배열 A와 B를 더하여서 배열 C에 저장하는 루틴이다. 배열의 시작주소와 원소의 개수는 파라미터로 서브루틴에 전달된다. 해당번지의 instruction이 실행된 후 레지스터와 메모리의 결과를 작성하라.[각 1점]

주소				주소		
4		BSA	MAT	6B		ISZ PT2
5		HEX	100	6C		ISZ PT3
6		HEX	200	6D		ISZ CTR
7		HEX	300	6E		BUN LOP I
8		DEC	-4	6F		BUN MAT I
9		HLT		70	PT1	DEC 0
				71	PT2	DEC 0
		ORG	50	72	PT3	DEC 0
50	MAT,	HEX	0	73	CTR	DEC 0
51		LDA	MAT I			ORG 100
52		STA	PT1	100		DEC 6
53		ISZ	MAT	101		DEC 4
54		LDA	MAT I	102		DEC 7
55		STA	PT2	103		DEC 9
56		ISZ	MAT			ORG 200
57		LDA	MAT I	200		DEC 4
58		STA	PT3	201		DEC 6
59		ISZ	MAT	202		DEC 3
5A		LDA	MAT I	203		DEC 1
5B		STA	CTR			ORG 300
5C		ISZ	MAT	300		DEC 0
5D	LOP,	LDA	PT1 I	301		DEC 0
5E		ADD	PT2 1	302		DEC 0
5F		STA	PT3 I	303		DEC 0
6A		ISZ	PT1			

명령어	4	54	6F
AC			
PC			
50번지			
70번지			
302번지			

14. 다음 질문에 답하라 [10점].

번호	프로그램			번호	프로그램		
1		ORG	3000	22		SZE	
2		LDA	N	23		BUN	ONE
3		BSA	MUL	24		BUN	ZRO
4	FST,	DEC	0003	25	ONE,	LDA	X
5	SND,	DEC	0005	26		ADD	P
6	TRD,	HEX	0	27		STA	P
7	END	HLT		28	ZRO,	CLE	
8	N,	DEC	-2	29		LDA	X
9		ORG	2000	30		CIL	
10	MUL,	HEX	0	31		STA	X
11		STA	CTR	32		ISZ	CTR
12		LDA	MUL I	33		BUN	LOP
13		STA	X	34		LDA	P
14		ISZ	MUL	35		STA	MUL I
15		LDA	MUL I	36		ISZ	MUL
16		STA	Y	37		BUN	MUL I
17		ISZ	MUL	38	CTR,	DEC	0
18	LOP,	CLE		39	X,	HEX	0
19		LDA	Y	40	Y,	HEX	0
20		CIR		41	P.	HEX	0
21		STA	Y	42			

(1) 변수 X, Y가 프로그램 실행되는 과정에 변화되는 값을 작성하라. [각 칸 당 0.5점]

Y	0	->		->		->	
X	0	->		->		->	

(2) 서브프로그램이 시작될 때, 메인프로그램에서 무엇을 통하여 데이터를 전달하고, 그 값은 얼마인가? [각 1점]

방법		값	
----	--	---	--

(3) 최종 결과가 저장되는 주소는 얼마이고, 그 결과 값은 얼마인가?[각 1점]

주소	
결과값	

(4) 34, 35번 명령어를 빼고, 6번과 7번 명령어 사이에 하나의 명령어를 넣어서 동일한 결과를 얻는 프로그램으로 수정하여라. [1점]

LINE	프로그램		
7			
7.5	END	HLT	

(5) 이 프로그램이 수행되는 과정에서 오버플로우가 나는 경우를 처리하기위하여 26번과 27번 사이에 다음 명령어를 끼워넣었다. 서브루틴 **ERR**의 빈곳을 메워라. 단 4번과 5번의 데이터는 항상 양수라고 가정한다.[각1점]

26~27		BSA	ERR
41		ORG	3500
42	ERR	HEX	
43		①	
44		BUN	②
45		BUN	END

15. 다음 성질들이 RISC와 CISC로 구분하여 R과 C로 각각 기입하라 [각1점]

Characteristic	C or R
대부분의 명령어들의 사용빈도가 유사하다.	
다양한 Addressing Mode를 채택하고 있다	
Operand가 메모리인 명령어들을 많이 가진다.	
매 사이클마다 하나의 명령어를 수행할 수 있다.	
Controller는 Logic circuit로 구현한다.	
프로그램의 성능은 컴파일러의 성능에 좌우된다.	
파라미터 전달을 두 모듈 간의 공유 레지스터로 수행된다.	
명령어 파이프라인을 사용할 수 있다.	
상대적으로 많은 레지스터를 확보하고 있다.	
명령어의 operand를 다양하게 가져간다.	

16. 4단계- fetch, decode, operand fetch, execution -로 이루어진 instruction pipeline에 대하여 물음에 답하라.

(1) 각 단계를 수행하는 데 걸리는 시간을 δ 라고 할 때, 이와 같은 pipeline의 setup time을 δ 를 이용하여 표현하라.[2점]

(2) 이 pipeline의 이상적인 speedup은 얼마인가? [2점]

(3) 이 instruction pipeline이 실질적으로 이상적인 병렬처리 효과를 발휘할 수 없는 이유를 세가지 쓰라. [각 2점]