

## 연습문제

01 다음 수식을 1-주소 명령어, 2-주소 명령어, 3-주소 명령어로 프로그램을 각각 작성하시오.

$$Y = (A \times B + C) \div (D - E)$$

여기서 사용하는 명령어 니모닉(mnemonic)은 다음과 같다.

ADD	덧셈
SUB	뺄셈
MUL	곱셈
DIV	나눗셈
MOV	레지스터와 기억 장소 간의 데이터 이동
LOAD	기억 장소에서 AC로 데이터 적재
STOR	AC 데이터를 기억 장소에 저장

02 클럭 주파수가 2GHz인 CPU가 'ADD addr' 명령어를 인출하여 실행하는 데 걸리는 시간을 구하시오. 단, 직접 주소 지정 방식이 사용되며 인터럽트는 발생하지 않는다고 가정한다.

03 다음과 같은 주소 지정 방식을 사용하는 경우, 명령어 인출과 실행에 기억 장치 액세스가 몇 번 필요한가?

- (a) 즉시 주소 지정 방식
- (b) 레지스터 주소 지정 방식
- (c) 인덱스 주소 지정 방식
- (d) 간접 주소 지정 방식

04 상대 주소 지정 방식을 사용하는 분기 명령어가 450번지에 저장되어 있다. 명령어 길이는 16비트이며 연산 코드는 6비트다. 또한 기억 장치의 주소는 바이트 단위로 지정된다고 할 때 다음 물음에 답하시오.

- (a) 분기 명령어가 실행된 다음에 874번지로 분기되도록 하기 위해 분기 명령어의 주소 필드에 넣어야 할 내용을 2의 보수로 표현하시오.
- (b) 230번지로 분기되도록 하는 경우 분기 명령어의 주소 필드에 넣어야 할 내용을 2의 보수로 표현하시오.

05 다음 그림에서 해당 프로그램을 수행한 후 (a) 기억 장치에 저장된 170번지 데이터와 200번지 데이터를 인출하여 ALU에서 곱셈을 수행하였다(b). 이 결과를 레지스터 R1에 임시로 저장한 후 그 다음 명령어(c)로 레지스터 간접 주소 지정 방식이 수행되었다. 이 메모리에 저장된 최종 주소와 데이터 값은?

(a)

10	LOAD	150
11	ADD	160
12	STOR	170
13	LOAD	180
14	MUL	190
15	STOR	200

주 기억 장치	
35	50
36	190
37	200
38	170
⋮	⋮
150	0004
160	0002
170	0003
180	0003
190	0002
200	0001

레지스터	
R1	
R2	100
R3	150

(b) 170번지 Data × 200번지 Data = ALU

(c)

opcode	R1
--------	----

06 컴퓨터의 상태가 그림과 같을 때 다음 명령어들이 연속적으로 수행된 후 스택의 최종 상태를 오른쪽에 나타내시오.

POP R3  
PUSH R0  
PUSH R2  
PUSH R1  
POP R0

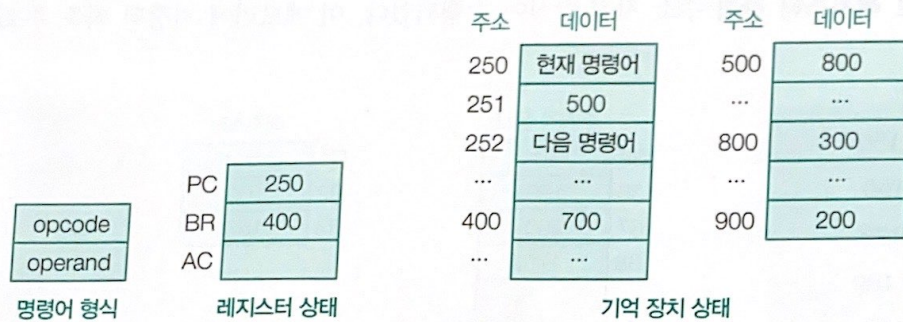
R3	3333H
R2	2222H
R1	1111H
R0	0000H
SP	1004H

100EH	89ABH
100CH	789AH
100AH	6789H
1008H	5678H
1006H	4567H
1004H	3456H
1002H	2345H
1000H	1234H

R3		100EH	
R2		100CH	
R1		100AH	
R0		1008H	
		1006H	
		1004H	
		1002H	
		1000H	
SP			



- 07 그림과 같이 컴퓨터의 명령어 형식은 워드를 2개 차지하고, opcode와 한 개의 주소 필드를 가지고 있다. BR은 베이스 레지스터고 데이터 레지스터는 AC 한 개다.



- (a) 명령어가 "load immediate 500"일 때 AC에 적재되는 값은? 500
- (b) 명령어가 "load direct 500"일 때 AC에 적재되는 값은? 800
- (c) 명령어가 "load indirect 500"일 때 AC에 적재되는 값은? 300
- (d) 명령어가 "load base register relative 500"일 때 AC에 적재되는 값은? 200
- (e) 명령어가 "branch PC relative 500"일 때 분기 목적지 주소는? 750
- 08 상대 주소 지정 방식의 분기 명령어가 기억 장치 620<sub>(10)</sub> 번지에 저장되어 있다. 이 명령어는 기억 장치 한 개의 장소를 차지한다. 분기 목적어 주소는 530<sub>(10)</sub> 번지다. 명령어에 분기 목적지 주소는 10비트가 할당되어 있다. 명령어에 표현되어 있는 오퍼랜드 필드 값을 다음 순서에 따라 구하시오.
- (a) 이 명령어가 실행될 때 PC 값은 10진수로 얼마인가?
- (b) PC 값과 분기 목적지까지의 차이는 10진수로 얼마인가?
- (c) 명령어에 표현되어 있는 오퍼랜드 필드 값을 10비트의 2진수로 변환하면?

- 09 레지스터 값이 그림과 같을 때 다음 명령어의 유효 주소를 구하시오. 괄호 안은 주소 지정 방식이다. R0, R1은 범용 레지스터고, IX는 인덱스 레지스터, BR은 베이스 레지스터다.

R0	120	IX	212
R1	211	BR	820

- (a) Load R0 (register indirect addressing) 120
- (b) Load IX+4 (indexing) 216
- (c) Load BR-8 (base register addressing) 812
- (d) Load IX+R1 (indexing) 423

- 10 오른쪽 그림은 인터럽트 서비스 루틴의 마지막 명령어인 인터럽트 복귀 명령어를 수행할 때 컴퓨터의 상태다. 주소는 16비트고 기억 장치는 8비트 단위로 구성되어 있으며 little-endian 방식으로 값을 저장한다. 인터럽트 복귀 명령어를 실행한 후 변경되는 레지스터 값을 구하시오. SR은 상태 레지스터다.

SR	1200
PC	5420
SP	6504

주소	데이터
6500	12
6501	23
6502	56
6503	7B
6504	23
6505	45
6506	75

- 11 AC와 기본 주소 레지스터 B에 저장되어 있는 값과 주기억 장치 내의 워드 값이 다음 그림과 같이 주어졌을 때, "ADD <sup>40</sup>~~400~~"과 같은 명령어에 대하여 주소 40이 각각 다음 주소인 경우 유효 주소 값과 명령어 수행 후의 AC 값을 기술하시오.

- (a) 직접 주소  
(b) 간접 주소  
(c) B에 대한 상대 주소  
(d) 데이터 자신immediate addressing

B	125
AC	100

	주기억 장치
40	72
72	275
165	124

- 12 버스 사이클 주기가 16비트 마이크로프로세서와 32비트 마이크로프로세서가 같다고 하자. 평균적으로 명령어와 오퍼랜드의 20%는 길이가 32비트고, 40%는 길이가 16비트이며, 40%는 8비트라고 가정하자. 32비트 마이크로프로세서를 이용하여 명령어와 오퍼랜드를 인출할 때, 어느 정도의 성능 개선을 얻을 수 있는지 구하시오.
- 13 많은 종류의 명령 코드를 갖도록 명령어를 설계한다면 프로그래밍과 CPU 내부에 어떤 영향을 주는가?
- 14 대부분의 RISC 구조에서 3-주소 명령어를 주로 사용하는 이유를 설명하시오.
- 15 주소 해상도는 무엇인가? 명령어와 데이터를 위해 동일한 주소 해상도를 사용해야 하는가? 비트 단위의 주소 해상도를 사용하지 않는 이유를 설명하시오.