2장 CPU의 구조 및 기능 (9점)

1. 다음 수식을 1-주소명령어, 2-주소명령어 3-주소명령어로 프로그램을 각각 작성하시오 (3점)

Y = (A+B+C) ÷ (D-E)

단, 여기서 사용하는 명령어 니모닉 코드는 다음과 같다

ADD: 덧셈 SUB: 뺄셈 MUL: 곱셈 DIV: 나눗셈  
MOV: 레지스터와 기억장소간의 데이터 이동  
 LOAD: 기억장소에서 AC로 데이터 적재  
 STOR: AC 데이터를 기억장소에 저장

1. 1-주소명령어들을 사용한 프로그램

LOAD A ; AC <- M[A]

ADD B ; AC <- AC + M[B]

ADD C ; AC <- AC + M[C]

STOR T ; M[T] <- AC

LOAD D ; AC <- M[D]

SUB E ; AC <- AC - M[E]

DIV T ; AC <- M[T] / AC

STOR Y ; M[Y] <- AC

1. 2-주소명령어들을 사용한 프로그램

MOV R1, A ; R1 <- M[A]

ADD R1, B ; R1 <- R1 + M[B]

ADD R1, C ; R1 <- R1 + M[C]

MOV R2, D ; R2 <- M[D]

SUB R2, E ; R2 <- R2 – M[E]

DIV R1, R2 ; R1 <- R1 / R2

MOV Y, R1 ; M[Y] <- R1

1. 3-주소명령어를 사용한 프로그램

ADD R1, A, B ; R1 <- M[A] + M[B]

ADD R1, R1, C ; R1 <- R1 + M[C]

SUB R2, D, E ; R2 <- M[D] – M[E]

DIV Y, R1, R2 ; M[Y] <- R1 / R2

1. 클럭 주파수가 2GHz 인 CPU가 ‘ ADD *addr* ’ 명령어를 인출하여 실행하는 데 걸리는 시간을 구하시오. 단 직접주소지정방식이 사용되며 인터럽트는 발생하지 않는다고 가정한다 (1점)

답: 3ns

1. 다음과 같은 주소지정방식을 사용하는 경우, 명령어 인출과 실행에   
   기억장소 액세스가 몇 번 필요한가? (3점)

* 즉시 주소 지정 방식: 기억장소에 액세스할 필요가 없다. 즉, 0번.
* 레지스터 주소 지정 방식: 0번.
* 간접 주소 지정 방식: 1번.

1. 상대주소지정방식을 사용하는 분기명령어가 450 번지에 저장되어 있다. 명령어 길이는 16비트이며 연산 코드는 6비트다. 또한, 기억장치의 주소는 바이트 단위로 지정된다고 할 때, 다음 물음에 답하시오. (2점)

* 분기 명령어가 실행된 다음에 874번지로 분기되도록 하기 위해 분기 명령어의 주소 필드에 넣어야 할 내용을 10진수로 표현 하시오

정답: 423

* 230번지로 분기되도록 하는 경우, 분기 명령어의 주소 필드에 넣어야 할 내용을 10진수로 표현 하시오

정답: -221