



1회 ▶ 산 05-2

중앙처리장치가 무엇을 하고 있는가를 나타내는 것으로서 기억장치의 사이클을 단위로 하여 해당 사이클 동안에 무엇을 위해 기억장치를 접근하는가를 나타내 주는 것은?

- ① 메이저 상태(major state)
- ② 마이너 상태(minor state)
- ③ 홀드 상태(hold state)
- ④ 대기 상태(ready state)

핵심이론

메이저 스테이트(Major State)

- CPU가 무엇을 하고 있는가를 나타내는 것이다.
- 기억장치의 사이클을 단위로 하여 해당 사이클 동안에 무엇을 위해 기억장치를 접근하는가를 나타낸다.
- Fetch, Indirect, Execute, Interrupt의 4가지 상태가 있다.

유사문제

2회 ▶ 14-3, 07-1

1. CPU의 메이저 상태(Major State)로 볼 수 없는 것은?

- ① Fetch
- ② Indirect
- ③ Execute
- ④ Direct

1회 ▶ 산 07-1

2. 다음 중 명령어 주기에 속하지 않는 것은?

- ① Fetch Cycle
- ② Direct Cycle
- ③ Indirect C
- ④ Execution Cycle

3회 ▶ 산 14-3, 06-4, 04-4

3. 명령어 사이클(Instruction Cycle)이 아닌 것은?

- ① Fetch Cycle
- ② Control Cycle
- ③ Indirect Cycle
- ④ Interrupt Cycle

2회 ▶ 산 04-2, 03-1

4. 인스트럭션(instruction) 사이클에 해당되지 않는 것은?

- ① FETCH cycle
- ② INDIRECT cycle
- ③ DECODE cycle
- ④ EXECUTE cycle

2회 ▶ 05-2, 01-2

5. 전자계산기의 중앙처리장치(CPU)는 4가지 단계를 반복적으로 거치면서 동작한다. 4가지 단계에 속하지 않는 것은?

- ① Fetch cycle
- ② Branch cycle
- ③ Interrupt cycle
- ④ Execute cycle

1회 ▶ 산 00-2

6. CPU가 명령어를 수행하는데 필요한 동작이 아닌 것은?

- ① buffer
- ② fetch
- ③ decode
- ④ execute

2회 ▶ 산 03-2, 99-1

7. 인스트럭션(instruction)의 수행 과정이 아닌 것은?

- ① 주소 변환
- ② 명령 인출
- ③ 오퍼랜드 인출
- ④ 사이클 실행



1회 ▶ 08-1

인스트럭션을 수행하기 위한 메이저 상태에 대한 설명으로 옳은 것은?

- ① 명령어를 가져오기 위해 기억장치에 접근하는 것을 Fetch 상태라 한다.
- ② Execute 상태는 간접 주소지정 방식의 경우 수행된다.
- ③ CPU의 현재 상태를 보관하기 위한 기억장치 접근을 Indirect 상태라 한다.
- ④ 명령어 종류를 판별하는 것을 Indirect 상태라 한다.

핵심이론

메이저 스테이트의 종류

- 인출 단계(Fetch Cycle) : 다음에 실행할 명령어를 기억장치로부터 CPU로 가져와 해독하는 단계이다.
- 간접 단계(Indirect Cycle) : 인출 단계에서 해석된 명령의 Operand부가 간접 주소일 경우 유효 주소를 구하는 단계이다.
- 실행 단계(Execute Cycle) : 실제로 명령을 실행하는 단계이다.
- 인터럽트 단계(Interrupt Cycle) : 인터럽트 발생 시 복귀 주소를 저장시키고, 제어 순서를 인터럽트 처리 프로그램의 첫 번째 명령으로 옮기는 단계이다.

유사문제

3회 ▶ 산 14-1, 10-2, 07-2

1. 명령 수행을 위한 메이저 상태에 대한 설명 중 옳은 것은?

- ① 실행 상태는 간접 주소 방식의 경우에만 수행된다.
- ② 기억장치 내의 명령어를 가져오는 것을 인출(fetch) 상태라 한다.
- ③ CPU의 현재 상태를 보관하기 위한 기억장치 접근을 indirect 상태라 한다.
- ④ 명령어의 종류를 판별하는 것을 indirect 상태라 한다.

1회 ▶ 산 08-4

2. 다음은 메이저 스테이트의 4가지 상태에 대한 설명이다. 옳지 않은 것은?

- ① Fetch State: 명령을 읽고 해독한다.
- ② Indirect State: 유효 주소를 계산하여 실제 데이터를 읽어온다.
- ③ Execute State: 실제 데이터를 읽어 명령을 실행한다.
- ④ Interrupt State: 인터럽트 처리를 수행한다.

1회 ▶ 산 08-1

3. CPU가 명령어를 실행할 때의 메이저 상태에 대한 설명 중 옳은 것은?

- ① 실행 사이클은 간접 주소 방식의 경우에만 수행된다.
- ② 명령어의 종류를 판별하는 것을 Indirect 사이클이라 한다.
- ③ 기억장치 내의 명령어를 CPU로 가져오는 것을 패치(Fetch) 사이클이라 한다.
- ④ 인터럽트 사이클 동안 데이터를 기억장치에서 읽어낸다.

1회 ▶ 산 09-1

4. 인스트럭션의 수행 시간에 관한 설명으로 옳지 않은 것은?

- ① memory read/write cycle이 인스트럭션 수행 시간에 지배적 영향을 준다.
- ② 수행 시간은 인스트럭션 종류에 따라 다르다.
- ③ OP-code만으로 인스트럭션 수행 시간을 모두 알 수 있다.
- ④ 인스트럭션 수행 시간은 여러 개의 machine cycle로 구성된다.

1회 ▶ 12-2

5. 명령어 사이클에 대한 설명 중 옳지 않은 것은?

- ① 간접 사이클은 피연산 데이터가 있는 기억장치의 유효 주소를 계산하는 과정이다.
- ② 인터럽트 사이클은 요청된 서비스 프로그램을 수행하여 완료할 때까지의 과정이다.
- ③ 실행 사이클은 연산자 코드의 내용에 따라 연산을 수행하는 과정이다.
- ④ 패치 사이클은 주기억장치로부터 명령어를 꺼내어 디코딩하는 과정이다.



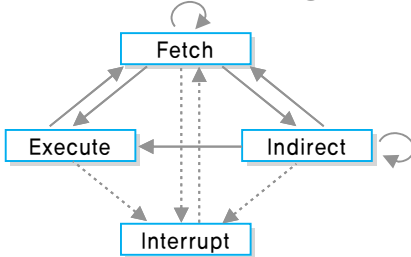
1회 ▶ 05-1주

메이저 상태(major state)에 대한 설명 중 옳은 것은?

- ① execute state가 끝나면 항상 fetch state로 간다.
- ② 특정한 명령에 대해서는 indirect state가 필요하다.
- ③ 메이저 사이클은 fetch, indirect, execute, interrupt 과정을 반드시 수행해야 한다.
- ④ indirect state는 데이터의 유효번지를 얻기 위해 기억장치에 접근하는 상태이다.

핵심이론

메이저 스테이트 변화 과정



유사문제

2회 ▶ 산 09-1, 05-4

1. 한 명령의 실행 사이클(execute cycle) 중에 인터럽트 요청이 있어 인터럽트를 처리한 후 CPU가 다음에 수행하는 cycle은?

- ① fetch cycle ② indirect cycle
- ③ execute cycle ④ direct cycle

1회 ▶ 02-4

2. 인터럽트를 처리한 후 다음으로 전환해야 할 메이저 상태는?

- ① Fetch 메이저 상태 ② Direct 메이저 상태
- ③ Indirect 메이저 상태 ④ Execute 메이저 상태

1회 ▶ 산 13-3

3. 컴퓨터의 메이저 사이클에서 인터럽트 사이클 후 처리되는 사이클은?

- ① 실행(execute) ② 간접(indirect)
- ③ 인출(fetch) ④ 직접(direct)

1회 ▶ 산 10-4

4. 하나의 인스트럭션을 수행하기 위한 메이저 상태의 변천 과정 중 잘못된 것은?

- ① fetch → execute ② fetch → indirect → execute
- ③ fetch → indirect ④ indirect → fetch → execute

1회 ▶ 06-4

5. 컴퓨터의 메이저 상태에 대한 설명 중 옳지 않은 것은?

- ① EXECUTE 상태가 끝나면 항상 FETCH 상태로만 간다.
- ② memory reference인 간접 주소 인스트럭션을 수행하기 위해서는 fetch-indirect-execute 순서로 진행되어야 한다.
- ③ 특정한 인스트럭션에 대해서는 INDIRECT 상태가 필요 없다.
- ④ FETCH 상태에서는 기억장치에서 인스트럭션을 읽어 중앙연산처리 장치로 가져온다.

1회 ▶ 12-3

6. 컴퓨터의 메이저 상태에 대한 설명으로 틀린 것은?

- ① 실행 상태가 끝나면 항상 패치 상태로만 간다.
- ② 간접 주소 명령어 형식인 경우 패치-간접-실행 순서로 진행되어야 한다.
- ③ 실행 상태는 연산자 코드의 내용에 따라 연산을 수행하는 과정이다.
- ④ 패치 상태에서는 기억장치에서 인스트럭션을 읽어 중앙처리장치로 가져온다.

1회 ▶ 09-4

7. 다음 메이저 상태(Major State)에 대한 설명으로 틀린 것은?

- ① fetch 상태는 명령을 메모리로부터 읽어 이를 해독하는 상태이다.
- ② fetch 상태의 다음 상태는 반드시 indirect 상태가 되어야 한다.
- ③ execute 상태는 처리하기 위한 실제 데이터를 읽어 decode된 연산을 수행하는 상태이다.
- ④ Interrupt 상태가 종료되면 fetch 상태로 분기한다.

1회 ▶ 산 14-1

8. 명령 수행 사이클에 대한 설명 중 옳지 않은 것은?

- ① 모든 cycle에는 그 끝부분에서 interrupt 사이클로 들어갈 것인가의 여부를 판단하게 되어 있다.
- ② indirect 사이클이 끝난 뒤에는 반드시 execute 사이클이 이어진다.
- ③ interrupt 사이클이 끝나면 반드시 fetch 사이클로 들어간다.
- ④ fetch 사이클 다음에는 indirect 사이클이나 execute 사이클이 이어진다.

1회 ▶ 13-2

9. 메이저 상태(major state)에서 제어 데이터에 대한 설명으로 잘못된 것은?

- ① FETCH state에서 중앙처리장치의 제어점을 제어하기 위한 제어 데이터는 명령어이다.
- ② INDIRECT state에서 다음 상태로 변천하는 것을 제어하는 데이터는 간접 주소와 직접 주소를 구별하는 비트이다.
- ③ EXECUTE state에서 다음 상태로 변천하는 것을 제어하는 데이터는 인터럽트 요청 신호이다.
- ④ INTERRUPT state에서는 제어 데이터에 의하여 fetch state로 변한다.

[정답] 핵심문제 ④ / 유사문제 1. ① 2. ① 3. ③ 4. ④ 5. ① 6. ① 7. ② 8. ① 9. ④



144 인출 단계(Fetch Cycle)

1회 ▶ 06-1

컴퓨터 시스템이 작동되면 먼저 프로그램 카운터의 초기 주소값이 결정되고 주소에 의하여 명령어가 기억장치로부터 읽혀지는 것을 무엇이라 하는가?

- ① 인출(Fetch)
- ② 실행(Execute)
- ③ 간접(Indirect)
- ④ 인터럽트(Interrupt)

핵심이론

인출 단계(Fetch Cycle)

- 다음에 실행할 명령어를 기억장치로부터 CPU로 가져와 해독하는 단계이다.
- 인스트럭션의 종류에 대한 판단이 이루어지는 상태이다.
- 인터럽트를 처리한 후 실행되는 단계이다.

유사문제

1회 ▶ 산 03-1

1. 기억장치에서 instruction을 읽어서 CPU로 가져오는 상태를 무엇이라 하는가?

- ① Interrupt 상태 ② Indirect 상태
- ③ Execute 상태 ④ Fetch 상태

1회 ▶ 산 06-4

2. 기억장치에서 명령어를 읽어 CPU로 가져오는 것을 무엇이라 하는가?

- ① Reference ② fetch
- ③ Execute ④ Major state

2회 ▶ 산 13-2, 04-1

3. 인스트럭션은 중앙처리장치를 이용하여 수행되는데 다음 중 명령을 읽어내는 사이클(cycle)은?

- ① fetch 사이클 ② Execute 사이클
- ③ Indirect 사이클 ④ Timing 사이클

1회 ▶ 03-1

4. 기억장치에서 인스트럭션을 읽어서 중앙처리장치로 가져올 때 중앙처리장치와 제어기는 어떤 상태인가?

- ① 인출(fetch) 상태
- ② 실행(execute) 상태
- ③ 간접(indirect) 상태
- ④ 인터럽트(interrupt) 상태

1회 ▶ 산 02-1

5. 메이저 상태에서 인스트럭션의 종류에 대한 판단이 이루어지는 상태는?

- ① fetch
- ② execute
- ③ interrupt
- ④ indirect

1회 ▶ 11-2

6. 컴퓨터에서 명령어를 처리하기 위해서 명령어를 CPU에 옮긴 후 명령 레지스터(IR)에 셋(set)시켜 해독하는 단계는?

- ① Fetch Cycle
- ② Indirect Cycle
- ③ Execute Cycle
- ④ Interrupt Cycle

4회 ▶ 10-4, 04-2, 01-2, 99-3

7. 한 명령의 Execute Cycle 중에 Interrupt 요청을 받아 Interrupt를 처리한 후 실행되는 사이클은?

- ① Fetch Cycle
- ② Indirect Cycle
- ③ Execute Cycle
- ④ Direct Cycle



2회 ▶ 00-3, 산 07-4

다음 일련의 Micro-Operation들은 어느 명령어 Cycle을 나타내고 있는가?

T1 : MAR ← PC
 T2 : MBR ← M, PC ← PC + 1
 T3 : MBR(OP), I ← MBR(I)

- ① Fetch ② Indirect ③ Execution ④ Interrupt

핵심이론

인출 단계 마이크로 오퍼레이션

T0 : MAR ← PC
 T1 : MBR ← M[MAR],
 PC ← PC+1
 T2 : OPR ← MBR[OP],
 I ← MBR[I]
 T3 : F ← 1 또는 R ← 1

유사문제

1회 ▶ 02-4

1. 인출 사이클(fetch cycle)의 첫 마이크로 오퍼레이션은?

- ① MAR←PC ② AC←AC+MBR ③ MAR←MBR ④ IR←MBR

3회 ▶ 11-1, 05-2, 02-1

2. fetch cycle에서 일어나는 micro instruction이다. 실행 순서가 옳은 것은? (단, MAR : Memory Address Register, MBR : Memory Buffer Register, PC : Program Counter, OPR : Operation Code Register)

ㄱ. MAR ← PC ㄴ. MBR ← M(MAR)
 ㄷ. PC ← PC+1 ㄹ. OPR ← MBR(OP)

- ① ㄴ→ㄱ→ㄷ→ㄹ ② ㄱ→ㄴ→ㄷ→ㄹ
 ③ ㄴ→ㄷ→ㄱ→ㄹ ④ ㄷ→ㄱ→ㄴ→ㄹ

2회 ▶ 산 05-4, 99-2

3. 컴퓨터의 fetch 사이클 시퀀스를 옳게 나타낸 것은?

ㄱ. 명령을 읽고 PC를 하나 증가 ㄴ. 실행 사이클로 간다.
 ㄷ. 명령의 번지를 전송 ㄹ. OP 코드를 전송
 ㅁ. 간접 사이클로 간다.

- ① ㄴ→ㄱ→ㄷ→ㄹ→ㅁ ② ㄷ→ㄱ→ㄷ→ㄹ→ㄴ
 ③ ㄷ→ㄹ→ㄱ→ㄴ→ㅁ ④ ㄱ→ㄷ→ㄷ→ㄹ→ㅁ

1회 ▶ 05-4

4. 명령을 수행하는 과정에서 가장 먼저 수행되어야 하는 마이크로 오퍼레이션은?

- ① PC+1→PC ② MBR→IR ③ PC→MAR ④ PC→MBR

1회 ▶ 산 99-2

5. 메이저 스테이트(Major state)를 명령이 패치(Fetch) 사이클에서 사용하지 않는 것은?

- ① Program Counter ② Stack Pointer
 ③ Memory Buffer Register ④ Memory Address Register

1회 ▶ 산 13-3

6. 인출(fetch) 명령 사이클 상태를 나타낸 것으로 적합하지 않은 것은?

- ① ADD X : MBR(OP) → IR ② AND X : MBR(OP) → IR
 ③ ADD X : MBR+AC → AC ④ JMP X : MBR(PC) → IR

2회 ▶ 산 01-3, 99-3

7. program counter의 기능을 설명한 것 중 옳은 것은?

- ① PC의 내용은 fetch cycle 동안에 1 증가된다.
 ② PC의 내용은 execute cycle 동안에 1 증가된다.
 ③ PC의 내용은 fetching, executing 과 관계없다.
 ④ PC의 내용은 변화하지 않는다.

1회 ▶ 산 01-3

8. 페치 사이클(fetch cycle)에 해당하지 않는 것은?

- ① 주기억장치의 지정 장소(address)로부터 명령을 끄집어내어 CPU에 옮긴다.
 ② 명령의 오퍼레이션(operation)부를 명령 레지스터(Instruction Register)에 세트(set)시켜 해독시킨다.
 ③ 다음에 실행할 명령의 기억장소(address)를 세트(set)시킨다.
 ④ 실제로 명령을 이행한다.

[정답] 핵심문제 ① / 유사문제 1. ① 2. ② 3. ② 4. ③ 5. ② 6. ③ 7. ① 8. ④



146 간접 단계(Indirect Cycle)

2회 ▶ 10-1, 03-4

인스트럭션 수행시 유효 주소를 구하기 위한 메이저 상태는?

- ① FETCH 상태 ② EXECUTE 상태
③ INDIRECT 상태 ④ INTERRUPT 상태

핵심이론

간접 단계(Indirect Cycle)

- 인출 단계에서 해석된 명령의 Operand부가 간접 주소일 경우 유효 주소를 구하는 단계이다.
- 기억장치로부터 Operand의 주소를 읽는다.
- 마이크로 오퍼레이션

T0 : MAR ← MBR[AD]
T1 : MBR ← M[MAR]
T2 : No Operation
T3 : F ← 1, R ← 0

유사문제

2회 ▶ 05-1주, 99-3

1. indirect cycle 동안에 컴퓨터는 무엇을 하는가?

- ① 명령을 읽는다.
② 오퍼랜드(operand)를 읽는다.
③ 인터럽트(interrupt)를 처리한다.
④ 오퍼랜드(operand)의 어드레스(address)를 읽는다.

2회 ▶ 10-4, 06-4

2. 간접 사이클 동안에는 어떤 동작이 수행되는가?

- ① 기억장치로부터 명령어의 주소를 인출한다.
② 기억장치로부터 데이터를 인출한다.
③ 기억장치로부터 데이터의 주소를 인출한다.
④ 기억장치로부터 명령어를 인출한다.

1회 ▶ 산 10-1

3. indirect cycle 동안에 컴퓨터가 수행하는 작업은?

- ① 명령을 읽는다.
② 오퍼랜드의 번지를 읽는다.
③ 오퍼랜드를 읽는다.
④ 인터럽트를 처리한다.

2회 ▶ 07-2, 05-1

4. 간접 상태(indirect state) 동안에 수행되는 것은?

- ① 명령어를 읽는다.
② 오퍼랜드의 주소를 읽는다.
③ 오퍼랜드를 읽는다.
④ 인터럽트를 처리한다.

2회 ▶ 산 05-1주, 99-4

5. 메모리로부터 읽은 내용이 오퍼랜드(operand)의 번지일 경우 컴퓨터의 사이클(cycle)은?

- ① 인터럽트 사이클 ② 페치 사이클 ③ 실행 사이클 ④ 간접 사이클

2회 ▶ 산 09-4, 99-1

6. 오퍼랜드(operand)의 번지를 읽어내는 컴퓨터 사이클은?

- ① 간접 사이클 ② 실행 사이클 ③ 직접 사이클 ④ 인터럽트 사이클

1회 ▶ 09-4

7. 다음과 같은 마이크로 오퍼레이션과 관련 있는 사이클은?

- ① MAR ← MBR(addr)
② MBR ← M(MAR)
③ 실행

- ① FETCH CYCLE ② EXECUTE CYCLE
③ INDIRECT CYCLE ④ INTERRUPT CYCLE

1회 ▶ 09-1

8. 간접 사이클(Indirect cycle)을 옳게 나타낸 마이크로 오퍼레이션은? (단, MAR : memory address register, MBR : memory buffer register, IEN : interrupt enable)

- ① MAR ← MBR(AD),
 MBR ← M(MAR)
② MAR ← PC,
 MBR ← M(MAR), PC ← PC+1
 OPR ← MBR(OP), I ← MBR(I)
③ MAR ← MBR(AD),
 MBR ← AC
 M ← MBR
④ MAR(AD) ← PC, PC ← 0,
 MAR ← PC, PC ← PC+1
 M ← MBR, IEN ← 0

[정답] 핵심문제 ③ / 유사문제 1. ④ 2. ③ 3. ② 4. ② 5. ④ 6. ① 7. ③ 8. ①



3회 ▶ 11-3, 05-1, 00-1

메이저 스테이트 중 하드웨어로 실현되는 서브루틴의 호출이라고 볼 수 있는 것은?

- ① FETCH 스테이트 ② INDIRECT 스테이트 ③ EXECUTE 스테이트 ④ INTERRUPT 스테이트

핵심이론

인터럽트 단계(Interrupt Cycle)

- 인터럽트 발생 시 복귀 주소를 저장시키고, 제어 순서를 인터럽트 처리 프로그램의 첫 번째 명령으로 옮기는 단계이다.
- 하드웨어로 실현되는 서브루틴의 호출이라고 볼 수 있다.
- 인터럽트를 처리한 후에는 항상 인출 단계로 변천한다.
- 마이크로 오퍼레이션

T0 : MBR[AD] ← PC, PC ← 0
T1 : MAR ← PC, PC ← PC+1
T2 : M[MAR] ← MBR, IEN ← 0
T3 : F ← 0, R ← 0

유사문제

1회 ▶ 산 12-1

1. 메이저 상태(Major State) 중 인스트럭션의 수행과는 상대적으로 무관한 것은?

- ① Fetch Major State ② Indirect Major State
③ Execute Major State ④ Interrupt Major State

2회 ▶ 11-3, 08-4

2. Interrupt cycle에 대한 micro-operation 중에서 관계가 없는 것은? (단, MAR : Memory Address Register, PC : Program Counter, M : memory, MBR : Memory Buffer Register, IEN : Interrupt Enable이며, Interrupt Handler는 0번지에 저장되어 있다.)

- ① MAR ← PC, PC ← PC + 1 ② MBR ← MAR, PC ← 0
③ M ← MBR, IEN ← 0 ④ GO TO fetch cycle

1회 ▶ 05-1추

3. 인터럽트 사이클을 위한 마이크로 연산이 아닌 것은?

- ① MAR←PC, PC←PC+1 ② MBR(AD)←PC, PC←0
③ M←MAR, IEN←0 ④ F←0, R←0

1회 ▶ 산 08-1

4. 다음 마이크로 오퍼레이션은 무슨 사이클에 해당하는가? (단, IEN: Interrupt Enable Flip-Flop, AR: Address Register, TR: Temporary Register, R: Interrupt Flip-Flop, SC: Sequence Counter)

- ① : AR←0, TR←PC
② : M[AR]←TR, PC←0
③ : PC←PC + 1, IEN←0, R←0, SC←0

- ① Fetch Cycle ② Interrupt Cycle
③ Indirect Cycle ④ Execute Cycle

1회 ▶ 산 00-3

5. 인터럽트 사이클에 대한 마이크로 동작 중 관계없는 것은? (단, Interrupt handler는 0번지에 저장되어 있다.)

- ① MBR←PC, PC←0 ② MAR←PC, PC←0
③ M←MBR, IEN←0 ④ fetch cycle로 간다.

3회 ▶ 09-2, 03-2, 01-3

6. 다음과 같은 마이크로 오퍼레이션이 일어나는 상태는?

1. MBR(AD) ← PC, PC ← 0
2. MAR ← PC, PC ← PC+1
3. M ← MBR, IEN ← 0
4. F ← 0, R ← 0

- ① Fetch ② Indirect ③ Interrupt ④ execute

1회 ▶ 12-3

7. 인터럽트 서비스가 진행되면 다른 인터럽트를 배제시켜야 하는데 이 때 변경시켜야 하는 flag는 무엇이며, 어떻게 변경하여야 하는가?

- ① IEN ← 1 ② IEN ← 0 ③ VAD ← 0 ④ VAD ← 1

[정답] 핵심문제 ④ / 유사문제 1. ④ 2. ② 3. ③ 4. ② 5. ② 6. ③ 7. ②



1회 ▶ 산 08-4

제어 유닛의 내부 구조 구성 요소들의 설명으로 틀린 것은?

- ① 명령어 해독기 : 다음에 실행할 마이크로 명령어의 주소를 저장하는 레지스터이다.
- ② 제어 기억장치 : 마이크로 명령어들로 이루어진 마이크로 프로그램을 저장하는 내부 기억장치이다.
- ③ 순서 제어 모듈 : 마이크로 명령어의 실행 순서를 결정하는 회로들의 집합이다.
- ④ 제어 버퍼 레지스터 : 제어 기억장치로부터 읽혀진 마이크로 명령어 비트들을 일시적으로 저장하는 레지스터이다.

핵심이론

제어기(Control Unit)의 구성 요소

- 명령어 해독기(Instruction Decoder)
- 제어 기억장치(Control Memory)
- 서브루틴 레지스터(Subroutine Register)
- 제어 주소 레지스터(CAR ; Control Address Register)
- 제어 버퍼 레지스터(CBR ; Control Buffer Register)
- 순서제어 모듈(Sequencing Module)

제어 데이터

- 제어장치가 제어 신호를 발생시키기 위해 필요한 자료이다.
- 제어 데이터의 종류
 - 메이저 스테이트 사이의 변환을 제어하는 제어 데이터
 - 중앙처리장치의 제어점을 제어하는 데 필요한 제어 데이터
 - 인스트럭션 수행 순서를 결정하는 데 필요한 제어 데이터
- 메이저 스테이트별 제어점 제어
 - 인출 단계 : 명령어
 - 간접 단계 : 유효 주소
 - 실행 단계 : 명령어의 OP-Code
 - 인터럽트 단계 : 인터럽트 체제에 따라 다름

유사문제

1회 ▶ 12-2

1. 마이크로 프로그램을 이용하는 제어장치의 구성 요소가 아닌 것은?

- ① 순서 제어 모듈
- ② 서브루틴 레지스터
- ③ 명령 레지스터
- ④ 제어 버퍼 레지스터

1회 ▶ 00-2

2. 명령 레지스터에 호출된 OP Code를 해독하여 그 명령을 수행 시키는데 필요한 각종 제어 신호를 만들어 내는 장치는?

- ① Instruction Decoder
- ② Instruction Encoder
- ③ Instruction Counter
- ④ Instruction Multiplexer

2회 ▶ 02-1, 01-1

3. 제어 데이터가 될 수 없는 것은?

- ① 연산자의 종류
- ② 연산을 위한 수치 데이터
- ③ 인스트럭션의 주소지정 방식
- ④ 연산 결과에 대한 상태 플래그 내용

2회 ▶ 산 01-1, 00-2

4. CPU 내 제어기의 제어 데이터 중에 포함되지 않는 것은?

- ① 각 메이저 스테이트 사이의 변환을 제어하는 제어 데이터
- ② 중앙처리장치의 제어점을 제어하는데 필요한 제어 데이터
- ③ 인스트럭션의 수행순서를 결정하는데 필요한 제어 데이터
- ④ 입·출력 장치의 제어점을 제어하는 제어 데이터

2회 ▶ 산 01-3, 00-4

5. 제어 데이터의 종류를 열거한 것으로 옳지 않은 것은?

- ① 메이저 스테이트 사이의 변환을 제어하는 제어 데이터
- ② 입·출력 장치의 제어점을 제어하는데 필요한 제어 데이터
- ③ 중앙처리장치의 제어점을 제어하는데 필요한 제어 데이터
- ④ 인스트럭션 수행 순서를 결정하는데 필요한 제어 데이터

[정답] 핵심문제 ① / 유사문제 1. ③ 2. ① 3. ② 4. ④ 5. ②



1회 ▶ 02-2

하드와이어드(hard-wired) 방식이 마이크로 프로그래밍 방식보다 좋은 점은?

- ① 다양한 어드레스 모드를 갖는다.
- ② 인스트럭션 세트를 변경하기가 쉽다.
- ③ 컴퓨터의 속도가 향상된다.
- ④ 비교적 복잡한 명령 세트를 가진 시스템에 적합하다.

핵심이론

고정 배선(Hard-wired) 방식

- 하드웨어적인 방법으로 제어기를 구현하는 방식이다.
- 소프트웨어 방식에 비해 속도가 빠르다.
- 소프트웨어 방식에 비해 비싸다.
- 명령어 세트를 변경할 수 없다.
- 비교적 간단한 명령 세트를 가진 시스템에 적합하다.

유사문제

1회 ▶ 99-3

1. 하드 와이어드(hard-wired) 방식이 마이크로 프로그래밍 방식보다 좋은 점은?

- ① 구조화된 제어 구조를 제공한다.
- ② 인스트럭션 세트를 변경할 수 있다.
- ③ 컴퓨터의 속도를 감할 수 있다.
- ④ 비교적 복잡한 명령 세트를 가진 시스템에 적합하다.

1회 ▶ 13-1

2. 하드와이어 제어 방식이 마이크로 프로그램을 이용한 제어 방식보다 좋은 점은?

- ① 비교적 복잡한 명령어들로 구성된 시스템 구현에 적합
- ② 마이크로 명령어를 추가하기 위해 설계 변경이 용이
- ③ 비교적 명령어 설계에 유연성과 자율성을 보장
- ④ 프로그램 실행 속도가 비교적 빠름

1회 ▶ 산 08-1

3. 논리회로 방식에 의한 제어기(Control Logic Unit)를 설명한 것 중 옳지 않은 것은?

- ① 고속 제어가 가능하다.
- ② 제어기의 변경이 쉽다.
- ③ 하드웨어적인 방법으로 제어장치를 구성한다.
- ④ 제어장치에 의해 제어 신호를 발생한다.

1회 ▶ 14-3

4. 고정 배선 제어 방식과 비교하여 마이크로 프로그램을 이용한 제어 방식의 특징으로 볼 수 없는 것은?

- ① 구조적이고 임의적인 설계가 가능하다.
- ② 경제적이며 시스템의 설계비용을 줄일 수 있다.
- ③ 보다 용이한 유지보수 관리가 가능하다.
- ④ 처리 속도가 빠르고 시스템이 간단할 때 유리하다.

1회 ▶ 산 11-2

5. hardwired control 방법으로 제어장치를 구현할 때 설명이 잘못된 것은?

- ① 논리회로 설계기법에 의해서 제어 신호를 생성하는 방법이다.
- ② RISC 구조를 기본으로 하는 컴퓨터에서 주로 많이 사용된다.
- ③ 동작 속도를 빠르게 할 수 있다.
- ④ instruction set을 갱신(update)하기가 용이하다.

1회 ▶ 산 12-2

6. 제어장치의 구현 방법 중 고정 배선식 제어장치(Hard Wired Control Unit)에 대한 설명으로 틀린 것은?

- ① 하드웨어적으로 구현한 방법으로 제어 신호를 발생시킨다.
- ② 마이크로 프로그램 제어 방식보다 속도가 빠르다.
- ③ 한 번 만들어진 명령어 세트를 변경할 수 없다.
- ④ 제작은 어렵지만 제작비용은 저렴하다.



3회 ▶ 08-2, 00-2, 산 10-2

어떤 명령을 수행할 수 있도록 된 일련의 제어 워드가 특수한 기억장치 속에 저장된 것을 무엇이라 하는가?

- ① 제어 메모리 ② 제어 데이터 ③ 마이크로 프로그램 ④ 고정 배선 제어

핵심이론

마이크로 프로그래밍(Micro Programming) 방식

- 소프트웨어적인 방법으로 제어기를 구현하는 방식이다.
- 하드웨어 방식에 비해 저렴하지만, 속도가 느리다.
- 명령어 세트를 변경하기가 쉽다.

마이크로 프로그램(Micro Program)

- CPU 내의 제어기를 설계하는 프로그램이다.
- 어떤 명령을 수행할 수 있도록 된 일련의 제어 워드가 특수한 기억장치 속에 저장된 것이다.
- 마이크로 프로그램을 저장하는 제어 메모리는 주로 ROM을 사용한다.
- 전자계산기의 제작 단계에서 컨트롤 스토리지(Control Storage) 속에 저장한다.
- 마이크로 프로그램은 각종 제어 신호를 발생시킨다.
- 마이크로 프로그램은 마이크로 명령어들의 집합이다.

유사문제

2회 ▶ 산 09-1, 99-2

1. 마이크로 프로그램에 대한 설명 중 옳지 않은 것은?

- ① 마이크로 프로그램은 소프트웨어라고 하는 것보다 하드웨어적인 요소가 많아 펌웨어(firmware)라고도 불린다.
 ② 제어기를 구성하는 방법으로 마이크로 프로그램이 이용될 수 있다.
 ③ 마이크로 프로그램은 전자계산기의 제작 단계에서 컨트롤 스토리지(control storage) 속에 저장한다.
 ④ 마이크로 프로그램은 마이크로 명령어로 형성되어 있다.

4회 ▶ 산 03-2, 01-3, 00-4, 99-3

2. 마이크로 프로그램(micro program)에 대한 설명 중 옳지 않은 것은?

- ① 마이크로 프로그램은 보통 RAM에 저장한다.
 ② 마이크로 프로그램은 각종 제어 신호를 발생시킨다.
 ③ 마이크로 프로그램은 마이크로 명령어로 형성되어 있다.
 ④ 마이크로 프로그램은 CPU 내의 제어장치를 설계하는 프로그램이다.

1회 ▶ 산 00-1

3. 프로세서의 제어장치에 대한 설명 중 옳지 않은 것은?

- ① 고정 배선 방법과 마이크로 프로그램 방식이 있다.
 ② 마이크로 프로그램 방식은 고정 배선 방법보다 더 비싸다.
 ③ 고정 배선 방법은 부품의 수는 최소화되기는 하나 작동 속도를 높이는 데 목표가 있다.
 ④ 마이크로 프로그램 방식에서는 마이크로 프로그램을 저장하기 위한 제어 메모리가 필요하다.

2회 ▶ 산 10-1, 99-3

4. 마이크로 프로그램을 저장하는 제어 메모리는 주로 어떤 메모리를 사용하는가?

- ① ROM ② CAM ③ RAM ④ virtual memory

2회 ▶ 06-4, 산 10-2

5. 제어 기억장치는 보통 어느 기억장치 소자를 이용하여 구현되는가?

- ① CAM ② DISK ③ ROM ④ RAM

2회 ▶ 14-2, 09-4

6. 마이크로 프로그램 제어기가 다음에 수행할 마이크로 인스트럭션의 주소를 결정하는데 사용하는 정보가 아닌 것은?

- ① 인스트럭션 레지스터(IR)
 ② 타이밍 신호
 ③ CPU의 상태 레지스터
 ④ 마이크로 인스트럭션에 나타난 주소

1회 ▶ 산 10-4

7. 마이크로 프로그램 제어기가 다음에 수행할 마이크로 인스트럭션의 주소를 결정하는데 필요한 정보가 아닌 것은?

- ① 인스트럭션 레지스터
 ② CPU 상태 플래그
 ③ 마이크로 인스트럭션에 나타난 주소
 ④ 메모리 주소

[정답] 핵심문제 ③ / 유사문제 1. ③ 2. ① 3. ② 4. ① 5. ③ 6. ② 7. ④