



2회 ▶ 산 14-1, 07-2

기억장치로부터 명령어를 인출하여 해독하고, 해독된 명령어를 실행하기 위해 제어 신호를 발생시키는 각 단계의 세부 동작을 무엇이라 하는가?

- ① Fetch operation      ② Control operation      ③ Macro operation      ④ Micro operation

## 핵심 이론

## 마이크로 오퍼레이션(Micro Operation)의 정의

- 프로그램에 의한 명령의 수행은 마이크로 오퍼레이션의 수행으로 이루어지는데, 마이크로 오퍼레이션은 명령을 수행하기 위해 CPU 내의 레지스터와 플레그의 상태 변환을 일으키는 동작이다.
- 레지스터에 저장된 데이터에 의해서 이루어지는 동작이다.
- 기억장치로부터 명령어를 인출하여 해독하고, 해독된 명령어를 실행하기 위해 제어 신호를 발생시키는 각 단계의 세부 동작이다.
- 하나의 클록 펄스(Clock Pulse)에 기준을 두고 실행된다.
- 마이크로 오퍼레이션이 순서적으로 일어나게 하려면 제어 신호가 필요하다.

## 유사문제

2회 ▶ 03-4, 산 13-3

1. 명령을 수행하기 위해 CPU 내의 레지스터와 플래그의 상태 변환을 일으키는 작업을 무엇이라 하는가?

- ① fetch
- ② program operation
- ③ micro operation
- ④ count operation

2회 ▶ 산 12-3, 05-4

2. 다음이 설명하고 있는 것은?

- 하나의 클럭펄스 동안에 실행되는 기본적인 동작을 의미한다.
- 명령을 수행하기 위하여 CPU 내의 레지스터와 플래그의 상태 변환을 일으키는 동작을 의미한다.

- ① Program Operation
- ② Fetch
- ③ Count Operation
- ④ Micro Operation

1호 ▶ 04-2

3. 명령을 수행하기 위한 CPU의 상태 변환을 무엇이라 하는가?

- ① fetch
- ② program operation
- ③ micro operation
- ④ count operation

1호 ▶ 10-1

4. 인스트럭션 실행 과정에서 한 단계씩 이루어지는 동작은?

- ① micro operation                      ② fetch  
③ control routine                      ④ automation

3호 ▶ 08-4 06-4 00-3

5. 마이크로 오퍼레이션은 어디에 기준을 두고 실행되는가?

- ① flag                      ② 펄스                      ③ 메모리                      ④ RAM

3회 ▶ 06-2, 01-3, 01-2

6. 마이크로 오퍼레이션을 순서적으로 발생시키는데 필요한 것은?

- ① 스위치      ② 레지스터      ③ 누산기      ④ 제어 신호

9회 ▶ 04-2, 03-1, 02-4, 00-3, 99-3, 산 08-4, 05-2, 04-4, 04-1

7. 중앙연산 처리장치에서 micro-operation이 순서적으로 일어나게 하려면 무엇이 필요한가?

- ① 스위치(switch)                      ② 레지스터(register)  
③ 누산기(accumulator)              ④ 제어 신호(control signal)

1호 ▶ 13-1

8. 중앙연산 처리장치에서 micro-operation이 실행되도록 하는 것은?

- ① 스위치(switch)                      ② 레지스터(register)  
③ 누산기(accumulator)              ④ 제어 신호(control signal)

2회 ▶ 산 00-4, 99-2

9. 마이크로 동작의 시퀀스를 결정하여 주는 신호는?

- ① 사이클 신호      ② 누산기      ③ 레지스터      ④ 제어 신호

2회 ▶ 산 14-3, 06-1

10. 중앙처리장치에서 마이크로 동작의 실행이 순서적으로 발생할 수 있도록 역할을 담당하는 것은?

- ① 레지스터(REGISTER)
- ② 제어(CONTROL) 신호
- ③ 누산기(ACCUMULATOR)
- ④ 프로그램 카운터(PROGRAM COUNTER)

1회 ▶ 산 00-3

11. '메모리가 제대로 동작하려면 어드레스 신호, 데이터 신호 및 ( ) 신호가 상호간 시간적 관계가 잘 유지되어야 한다.' ( )에 해당하는 올바른 신호는?

- ① 제어                      ② 호출  
③ 액티브(active)        ④ 상태(state)

[정답] 핵심문제 ④ / 유사문제 1. ③ 2. ④ 3. ③ 4. ① 5. ② 6. ④ 7. ④ 8. ④ 9. ④ 10. ② 11. ①



3회 ▶ 14-1, 10-2, 06-4

마이크로 오퍼레이션에 대한 설명 중 옳지 않은 것은?

- ① 마이크로 오퍼레이션은 CPU 내의 레지스터들과 연산장치에 의해서 이루어진다.
- ② 프로그램에 의한 명령의 수행은 마이크로 오퍼레이션의 수행으로 이루어진다.
- ③ 마이크로 오퍼레이션 중에 CPU 내부의 연산 레지스터, 인덱스 레지스터는 프로그램으로 레지스터의 내용을 변경할 수 없다.
- ④ 마이크로 오퍼레이션이 실행될 때마다 CPU 내부의 상태는 변하게 된다.

## 유사 문제

1회 ▶ 08-2

1. 마이크로 오퍼레이션에 대한 설명으로 옳지 않은 것은?

- ① 마이크로 오퍼레이션이란 컴퓨터의 모든 명령을 구성하고 있는 몇 가지 종류의 기본 동작이다.
- ② 컴퓨터에서 수행이 가능한 마이크로 오퍼레이션의 종류는 그 컴퓨터 내에 존재하는 레지스터들과 연산기의 종류, 그들 서로 간에 연결된 형태로 의해 결정된다.
- ③ 일반적으로 마이크로 오퍼레이션은  $F(R, R) \rightarrow R$  마이크로 오퍼레이션과  $R \rightarrow R$  마이크로 오퍼레이션으로 구분하며 이 때 F는 처리기를 의미한다.
- ④  $F(R, R) \rightarrow R$  마이크로 오퍼레이션은 자료의 처리나 변형 없이 다른 레지스터로 자료가 옮겨지는 마이크로 오퍼레이션이다.

1회 ▶ 11-3

2. 마이크로 오퍼레이션(micro-operation)의 설명으로 옳지 않은 것은?

- ① 레지스터에 저장된 데이터에 의해 이루어지는 동작이다.
- ② 한 개의 클럭(clock) 펄스 동안 실행되는 기본동작이다.
- ③ 한 개의 instruction은 여러 개의 마이크로 오퍼레이션이 동작되어 실행된다.
- ④ 현재 CPU가 무엇을 하고 있는가를 나타내는 상태 동작이다.

3회 ▶ 산 05-1, 02-3, 01-3

3. 마이크로 동작(Micro-operation)에 대한 정의로서 옳은 것은?

- ① 컴퓨터의 빠른 계산 동작
- ② 2진수 계산에 쓰이는 동작
- ③ 플립플롭 내에서 기억되는 동작
- ④ 레지스터에 저장된 데이터에 의해서 이루어지는 동작

1회 ▶ 13-3

4. 마이크로 오퍼레이션에 대한 설명 중 옳은 것은?

- ① 레지스터 전달 명령은 마이크로 오퍼레이션을 기술할 수 없다.
- ② 마이크로 오퍼레이션 수행을 위해서 제어 함수는 필요 없다.
- ③ 마이크로 오퍼레이션은 1클럭 동안에 수행된다.
- ④ 마이크로 오퍼레이션 실행에서 워드 타임과 비트 타임은 같아야만 한다.

6회 ▶ 06-4, 03-1, 02-2, 산 09-2, 02-3, 01-1

5. 다음 중 어떤 명령(instruction)이 수행되기 위해 가장 우선적으로 이루어져야 하는 마이크로 오퍼레이션은?

- ①  $MBR \leftarrow PC$
- ②  $PC \leftarrow PC + 1$
- ③  $IR \leftarrow MBR$
- ④  $MAR \leftarrow PC$

2회 ▶ 산 12-1, 06-2

6. 다음과 같이 세 개의 마이크로 동작(micro operation)이 이루어졌을 경우에 이 동작이 끝났을 때 A 레지스터 상태는?

$$\begin{aligned} T_1 : B &\leftarrow \overline{B} \\ T_2 : B &\leftarrow B + 1 \\ T_3 : A &\leftarrow A + B \end{aligned}$$

- ①  $A \leftarrow \overline{A} + B + 1$
- ②  $A \leftarrow A + \overline{B} + 1$
- ③  $A \leftarrow \overline{A} + B + \overline{B} + 1$
- ④  $A \leftarrow B + \overline{B} + 1$

1회 ▶ 산 10-4

7. 다음 중 논리 마이크로 오퍼레이션이 아닌 것은?

- ①  $A \leftarrow 0$  : clear
- ②  $A \leftarrow 1$  : set
- ③  $A \leftarrow A \cdot B$  : AND( $A \cap B$ )
- ④  $A \leftarrow +1$  : 2의 보수

1회 ▶ 산 09-4

8. 다음 중 논리 마이크로 동작을 표현한 것은? (단, R1, R2, R3은 레지스터를 의미한다.)

- ①  $R3 \leftarrow R2 + R1$
- ②  $R1 \leftarrow \overline{R2} + 1$
- ③  $R1 \leftarrow R1 \oplus R2$
- ④  $R1 \leftarrow R1 - 1$

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



3회 ▶ 13-3, 03-4, 01-2

다음의 마이크로 오퍼레이션과 관련 있는 것은?

MAR ← MBR(ADDR)  
MBR ← M(MAR)  
EAC ← AC+MBR

- ① AND
- ② ADD
- ③ JMP
- ④ BSA

### 핵심이론

#### ADD

- AC(누산기)와 메모리의 내용을 더하여 결과를 AC에 저장하는 연산 명령이다.
- 마이크로 오퍼레이션

T0 : MAR ← MBR[AD]  
T1 : MBR ← M[MAR]  
T2 : AC ← AC+MBR

### 유사문제

1회 ▶ 산 05-2

1. 다음의 마이크로 오퍼레이션은 무엇을 수행하는가?

t0 : MAR ← MBR(AD)  
t1 : MBR ← M  
t2 : EAC ← AC + MBR

- ① AND 명령
- ② JMP 명령
- ③ ADD 명령
- ④ BSA 명령

1회 ▶ 12-3

2. 다음은 ADD 명령어의 마이크로 오퍼레이션이다. t2 시간에 가장 알맞은 동작은? (단, MAR : Memory Address Register, MBR : Memory Buffer Register, M(addr) : Memory, AC : 누산기)

t0 : MAR ← MBR(addr)  
t1 : MBR ← M(MAR)  
t2 :

- ① AC ← MBR
- ② MBR ← AC
- ③ M(MBR) ← MBR
- ④ AC ← AC+MBR

1회 ▶ 산 11-3

3. 다음과 같은 마이크로 사이클의 명령어는?

T1 : MAR ← IR(address)  
T2 : MBR ← M[MAR]  
T3 : AC ← AC + MBR

- ① ADD
- ② SUB
- ③ MUL
- ④ STORE



THEME

# 134 LDA (Load to Accumulator)

8회 ▶ 09-4, 07-2, 06-2, 05-4, 04-4, 02-4, 산 06-1, 99-3

다음 마이크로 연산이 나타내는 동작은?

```
MAR ← MBR(AD)
MBR ← M(MAR), AC ← 0
AC ← AC + MBR
```

- ① ADD to AC
- ② OR to AC
- ③ STORE to AC
- ④ LOAD to AC

## 핵심이론

### LDA(Load to AC)

- 메모리의 내용을 AC로 가져오는 명령이다.
- 마이크로 오퍼레이션

```
T0 : MAR ← MBR[AD]
T1 : MBR ← M[MAR], AC ← 0
T2 : AC ← AC + MBR
```

## 유사문제

1회 ▶ 10-4

1. 다음은 어떤 마이크로 명령에 의해서 수행되는 경우인가?  
(단, AC는 누산기임)

```
MAR ← MBR(AD)
MBR ← M(MAR)
AC ← MBR
```

- ① BUN 명령
- ② STA 명령
- ③ ISZ 명령
- ④ LDA 명령

1회 ▶ 13-2

2. 다음 내용은 LOAD 기능을 수행하는 마이크로 오퍼레이션이다. 이 가운데 어떤 명령어든지 수행되기 위해서는 반드시 거쳐야 하는 단계끼리 나열한 것은? (단, Rs1, Rd, S2 : 레지스터 주소)

단계	마이크로 오퍼레이션
1	MAR ← PC, R(read)
2	IR ← MBR, PC ← PC+4(명령어 크기)
3	IR Decoding
4	MAR ← Rs1+S2, R(read)
5	Rd ← MBR
6	PC ← 다음에 수행할 명령어의 주소

- ① 1, 2, 3, 6
- ② 2, 3, 4, 6
- ③ 3, 4, 5, 6
- ④ 1, 3, 5, 6

[정답] 핵심문제 ④ / 유사문제 1. ④ 2. ①



## 1회 ▶ 08-4

다음 마이크로 연산이 나타내는 동작은?

```
MAR ← MBR(AD)
MBR ← AC
M ← MBR
```

- ① Branch AC
- ② Store to AC
- ③ Add AC
- ④ Load to AC

## 핵심이론

## STA(Store AC)

- AC의 내용을 메모리에 저장하는 명령이다.
- 마이크로 오퍼레이션

```
T0 : MAR ← MBR[AD]
T1 : MBR ← AC
T2 : M[MAR] ← MBR
```

## 유사문제

## 1회 ▶ 산 02-1

1. 다음과 같은 마이크로 동작은 어떠한 명령의 수행 과정을 나타내는 것인가?

```
MBR ← AC (MBR에 데이터를 전송)
M ← MBR (메모리에 워드를 저장)
단, MAR ← MBR(AD) 유효번지를 전송
```

- ① load to AC(accumulator)
- ② branch unconditionally
- ③ AND to AC
- ④ store AC

## 1회 ▶ 산 03-4

2. 다음과 같은 마이크로 동작은 어떠한 명령의 수행 과정을 나타내는 것인가?

```
MAR ← MBR(AD) ; 유효번지를 전송
MBR ← AC ; MBR에 데이터를 전송
M ← MBR ; 메모리에 워드를 저장
```

- ① load to AC(accumulator)
- ② branch unconditionally
- ③ AND to AC
- ④ store AC

## 1회 ▶ 14-1

3. 다음과 같은 마이크로 동작은 어떤 명령의 수행 과정을 나타내는 것인가?

```
MAR ← MBR(ADDR) : 유효 주소 전송
MBR ← AC : MBR에 데이터를 전송
M(MAR) ← MBR : M은 메모리
```

- ① Load to AC
- ② AND to AC
- ③ Branch Unconditionally
- ④ Store AC

[정답] 핵심문제 ② / 유사문제 1. ④ 2. ④ 3. ④



## 1회 ▶ 04-2

BUN(Branch UNconditionally) 명령을 마이크로 동작으로 표시한 것은?

- ①  $PC \leftarrow MAR$
- ②  $PC \leftarrow MBR(AD)$
- ③  $MAR \leftarrow MBR(AD), PC \leftarrow M(MAR)$
- ④  $MBR \leftarrow M(MBR), PC \leftarrow MBR$

## 핵심이론

## BUN(Branch UNconditionally)

- PC에 특정한 주소를 전송하여 실행 명령의 위치를 변경하는 무조건 분기 명령이다.
- 마이크로 오퍼레이션

T0 :  $PC \leftarrow MBR[AD]$

## BSA(Branch and Save return Address)

- 복귀 주소를 저장하고 서브프로그램으로 분기하는 명령이다.
- 마이크로 오퍼레이션

T0 :  $MAR \leftarrow MBR[AD], MBR[AD] \leftarrow PC, PC \leftarrow MAR$   
 T1 :  $M[MAR] \leftarrow MBR[AD]$   
 T2 :  $PC \leftarrow PC+1$

## 유사문제

## 2회 ▶ 09-1, 05-1추

1. BSA(Branch and Save return Address)의 마이크로 동작 중 시간 T0에서 실행하는 동작이 아닌 것은? (단, T0는 sequencer 출력을 나타냄)

- ①  $PC \leftarrow PC+1$
- ②  $MAR \leftarrow MBR(AD)$
- ③  $MBR(AD) \leftarrow PC$
- ④  $PC \leftarrow MBR(AD)$

## 1회 ▶ 10-1

2. 다음은 어떤 마이크로 명령에 의해서 수행되는 경우인가?

$MAR \leftarrow MBR(addr)$   
 $MBR(addr) \leftarrow PC$   
 $PC \leftarrow MAR$   
 $M(MAR) \leftarrow MBR$   
 $PC \leftarrow PC + 1$

- ① BSA 명령
- ② STA 명령
- ③ ISZ 명령
- ④ ADD 명령



## 1회 ▶ 산 02-2

다음의 예는 실행 주기(execution cycle) 중에서 어떤 명령을 나타내는 것인가?

```
MAR ← MBR(AD)
MBR ← M
MBR ← MBR + 1
M ← MBR, iF(MBR=0) then (PC ← PC+1)
```

- |       |       |
|-------|-------|
| ① JMP | ② AND |
| ③ ISZ | ④ BSA |

## 핵심이론

## ISZ(Increment and Skip if Zero)

- 메모리의 값을 읽고 그 값을 1 증가시킨 값이 0이면 현재 명령을 건너뛰고 다음 명령으로 이동하는 명령이다.
- 마이크로 오퍼레이션

```
T0 : MAR ← MBR[AD]
T1 : MBR ← M[MAR]
T2 : MBR ← MBR+1
T3 : M[MBR] ← MBR,
      IF(MBR=0) THEN (PC←PC+1)
```

## 유사문제

## 1회 ▶ 산 13-3

1. 주소의 변경이나 프로그램 루프의 실행 횟수를 계산하는데 유용한 명령으로 지정된 주소에 저장된 워드의 내용을 1 증가시킨 다음 그 결과가 0이면 다음 명령을 skip하고, 0이 아니면 그대로 다음 명령을 실행하는 것은?

- |          |
|----------|
| ① BUN 명령 |
| ② BSA 명령 |
| ③ JMP 명령 |
| ④ ISZ 명령 |



T H E M E

# 138 마이크로 사이클 타임

2회 ▶ 산 13-2, 09-2

한 개의 마이크로 오퍼레이션 수행에 필요한 시간을 무엇이라 하는가?

- ① micro cycle time
- ② seek time
- ③ search time
- ④ access time

## 핵심이론

### 마이크로 사이클 타임(Micro Cycle Time)

- 한 개의 마이크로 오퍼레이션을 수행하는 데 필요한 시간이다.
- CPU의 속도를 나타내는 척도로 이용된다.
- CPU Clock Time 또는 CPU Cycle Time이라고도 한다.

## 유사문제

3회 ▶ 산 13-1, 10-4, 08-2

1. 마이크로 오퍼레이션 수행에 필요한 시간을 무엇이라 하는가?

- ① search time
- ② seek time
- ③ access time
- ④ CPU clock time

2회 ▶ 산 05-1, 01-2

2. "동기 디지털 시스템에 내장되어 있는 모든 레지스터의 타이밍은( )에 의하여 제어된다." ( )에 올바른 용어는?

- ① 마스터 클록 발생기
- ② 프로그램 카운터
- ③ 스택 포인터
- ④ 플립플롭





## 1회 ▶ 산 10-4

동기 고정식에서 마이크로 사이클 타임(micro cycle time)은 어떻게 정의 되는가?

- ① 마이크로 오퍼레이션들의 수행 시간 중 가장 긴 것을 마이크로 사이클 타임으로 정한다.
- ② 마이크로 오퍼레이션들의 수행 시간 중 가장 짧은 것을 마이크로 사이클 타임으로 정한다.
- ③ 마이크로 오퍼레이션들의 수행 시간 중 가장 짧은 것과 긴 것의 평균시간을 마이크로 사이클 타임으로 정한다.
- ④ 중앙처리장치의 클럭주기와 마이크로 사이클 타임은 항상 일치된다.

## 핵심이론

## 동기 고정식(Synchronous Fixed) 마이크로 사이클 타임

- 모든 마이크로 오퍼레이션 중에서 수행 시간이 가장 긴 것을 마이크로 사이클 타임으로 정의하는 방식이다.
- 모든 마이크로 오퍼레이션의 수행 시간이 유사한 경우 유리한 방식이다.
- 여러 종류의 마이크로 오퍼레이션의 수행 시 마이크로 사이클 타임이 실제적인 오퍼레이션 시간보다 길다.
- 제어장치의 구현이 간단하지만, CPU의 시간 낭비가 심하다.

## 유사문제

## 2회 ▶ 산 14-2, 11-1

1. 마이크로오퍼레이션에 관한 설명 중 옳은 것은?

- ① 마이크로 오퍼레이션을 동기시키는 방법으로 동기 고정식과 동기 가변식이 있다.
- ② 동기 고정식은 CPU 시간의 효율적 이용은 가능하나 제어가 복잡하다.
- ③ 동기 가변식은 CPU 시간의 낭비를 초래하지만 제어회로가 간단하다.
- ④ 마이크로 사이클은 마이크로 오퍼레이션과 무관하다.

## 1회 ▶ 09-1

2. 동기 고정식 마이크로 오퍼레이션 제어의 특성을 설명한 것이 아닌 것은?

- ① 제어장치의 구현이 간단하다.
- ② 중앙처리장치의 시간 이용이 비효율적이다.
- ③ 여러 종류의 마이크로 오퍼레이션의 수행시 CPU 사이클 타임이 실제적인 오퍼레이션 시간보다 길다.
- ④ 마이크로 오퍼레이션이 끝나고 다음 오퍼레이션이 수행될 때까지 시간 지연이 있게 되어 CPU 처리 속도가 느려진다.

## 2회 ▶ 11-3, 10-1

3. 동기 고정식 마이크로 오퍼레이션 제어의 특성이 아닌 것은?

- ① 제어장치의 구현이 간단하다.
- ② 여러 종류의 마이크로 오퍼레이션 수행시 CPU 사이클 타임이 실제적인 오퍼레이션 시간보다 길다.
- ③ 마이크로 오퍼레이션들의 수행 시간의 차이가 큰 경우에 적합한 제어이다.
- ④ 중앙처리장치의 시간 이용이 비효율적이다.

## 2회 ▶ 00-2, 99-1

4. 다음 마이크로 사이클에 대한 내용 중 가장 관계가 적은 것은?

- ① 마이크로 오퍼레이션 수행에 필요한 시간을 마이크로 사이클 타임이라 한다.
- ② 마이크로 오퍼레이션 중에서 수행 시간이 가장 긴 것을 정의한 방식이 동기 고정식이다.
- ③ 마이크로 오퍼레이션에 따라서 수행 시간을 다르게 하는 것을 동기 가변식이라 한다.
- ④ 마이크로 오퍼레이션 중에서 수행 시간의 차이가 큰 것을 약 30개로 구분해서 사용한다.



1회 ▶ 07-1

마이크로 사이클에 대한 설명으로 옳지 않은 것은?

- ① 마이크로 오퍼레이션 수행에 필요한 시간을 마이크로 사이클 타임이라 한다.
- ② 마이크로 오퍼레이션 중에서 수행 시간이 가장 긴 것을 정의한 방식이 동기 고정식이다.
- ③ 마이크로 오퍼레이션에 따라서 수행 시간을 다르게 하는 것을 동기 가변식이라 한다.
- ④ 모든 마이크로 오퍼레이션들의 수행 시간이 유사한 경우에 유리한 방식은 동기 가변식이다.

핵심이론

동기 가변식(Synchronous Variable) 마이크로 사이클 타임

- 각 마이크로 오퍼레이션에 따라서 마이크로 사이클 타임을 다르게 정의하는 방식이다.
- 각 마이크로 오퍼레이션의 사이클 타임이 현저한 차이를 나타낼 경우 유리하다.
- 수행 시간이 유사한 마이크로 오퍼레이션끼리 모아 집합을 이루고 각 집합에 대해서 서로 다른 마이크로 사이클 타임을 정한다.
- 제어장치의 구현이 비교적 복잡하지만, CPU의 시간을 효율적으로 이용할 수 있다.

유사문제

2회 ▶ 07-2, 03-4

1. 동기 가변식(Synchronous Variable) 동작에 대한 설명 중 옳지 않은 것은?

- ① 각 마이크로 오퍼레이션의 사이클 타임이 현저한 차이를 나타낼 때 사용한다.
- ② 모든 마이크로 오퍼레이션의 수행 시간이 유사한 경우에 사용된다.
- ③ 중앙처리장치의 시간을 효율적으로 이용할 수 있다.
- ④ 마이크로 오퍼레이션에 대하여 서로 다른 사이클을 정의 할 수 있다.

1회 ▶ 산 14-3

2. 동기 가변식 마이크로 사이클에 관한 설명으로 틀린 것은?

- ① CPU의 시간을 효율적으로 이용할 수 있다.
- ② 마이크로 오퍼레이션 수행 시간이 현저한 차이를 나타낼 때 사용한다.
- ③ 제어기의 구현이 단순하다.
- ④ 그룹화된 각 마이크로 오퍼레이션들에 대하여 서로 다른 사이클 시간을 정의한다.

2회 ▶ 10-2, 08-4

3. 동기가변식 마이크로 오퍼레이션 사이클 타임을 정의하는 방식은 수행 시간이 유사한 마이크로 오퍼레이션들끼리 모아 집합을 이루고 각 집합에 대해서 서로 다른 마이크로 오퍼레이션 사이클 타임을 정한다. 이 때 각 집합 간의 마이크로 사이클 타임을 정수배가 되도록 하는 이유는?

- ① 각 집합 간 서로 다른 사이클 타임의 동기를 맞추기 위하여
- ② 각 집합 간의 사이클 타임을 동기식과 비동기식으로 하기 위하여
- ③ 각 집합 간의 사이클 타임을 모두 다르게 정의하기 위하여
- ④ 사이클 타임을 비동기식으로 변환하기 위하여

2회 ▶ 산 12-2, 09-2

4. 마이크로 사이클 중 동기 고정식에 비교하여 동기 가변식에 관한 설명으로 옳지 않은 것은?

- ① CPU의 시간을 효율적으로 이용
- ② 마이크로 오퍼레이션 수행 시간의 차이가 클 경우 사용
- ③ 마이크로 오퍼레이션의 수행 시간이 유사한 경우 사용
- ④ 그룹화된 각 마이크로 오퍼레이션들에 대하여 서로 다른 사이클을 정의

1회 ▶ 산 08-2

5. 마이크로 오퍼레이션들 중 수행 시간이 유사한 마이크로오퍼레이션들끼리 모아 집합을 이루고 각 집합에 대해서 서로 다른 마이크로 오퍼레이션 사이클 타임을 정의하며 그 시간을 중앙처리장치의 클럭 주기로 정하는 방식으로 제어는 복잡하지만 중앙처리장치 효율을 높일 수 있는 마이크로오퍼레이션 사이클 타임 방식은?

- ① 동기 고정식                      ② 동기 가변식
- ③ 비동기 고정식                ④ 비동기 가변식

1회 ▶ 산 00-1

6. CPU 클럭 중 동기가변식에 관한 설명이 아닌 것은?

- ① 마이크로 오퍼레이션 수행 시간의 차이가 현저할 때 사용한다.
- ② 중앙처리장치의 시간을 효율적으로 이용할 수 있다.
- ③ 모든 마이크로 오퍼레이션의 수행 시간이 유사한 경우에 사용된다.
- ④ 모든 마이크로 오퍼레이션에 대하여 서로 다른 사이클을 정의할 수 있다.

1회 ▶ 산 05-1추

7. CPU 클럭 중 동기 가변식에 관한 설명이 아닌 것은?

- ① 마이크로 오퍼레이션 수행 시간의 차이가 현저할 때 사용된다.
- ② 중앙처리장치의 시간을 효율적으로 이용할 수 있다.
- ③ 수행 시간이 가장 긴 마이크로 오퍼레이션의 사이클 타임을 클럭 주기로 정한다.
- ④ 모든 마이크로 오퍼레이션에 대하여 서로 다른 사이클을 정의할 수 있다.

2회 ▶ 산 11-3, 10-1

8. 모든 마이크로 오퍼레이션에 대해 서로 다른 마이크로 사이클 시간을 할당하는 방법은?

- ① 비동기식                      ② 동기 고정식
- ③ 동기 가변식                ④ 중앙집중식

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