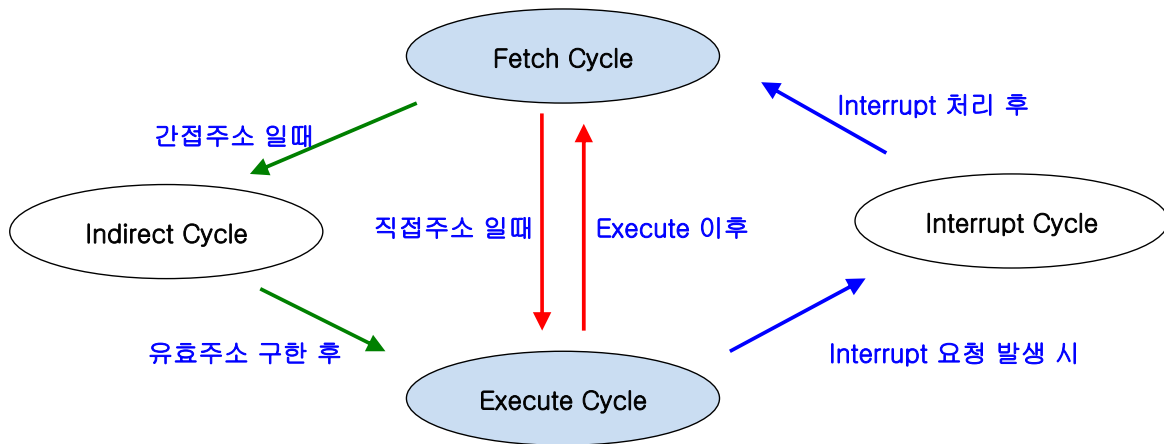


\* CPU의 명령어 수행 순서 : Instruction Fetch → 디코딩 → Operand Fetch → Execute

## 1. Major State 개념 : CPU 의 4가지 동작(상태) ★★★★★



- **Fetch** : 기억장치에서 instruction을 읽어서 CPU로 가져옴 → IR(명령레지스터) 저장 → 해독
- **Indirect** : Fetch 단계에서 해석된 명령의 주소부(operand)가 간접주소인 경우 수행됨
- **Execute** : Fetch 단계에서 인출하여 해석한 명령(연산)을 실행하는 단계
- **Interrupt** : 인터럽트 발생시 복귀주소(PC)를 저장시키고, 제어순서를 인터럽트 처리 프로그램의 첫 번째 명령으로 옮기는 단계

1

## 2. Major State (Micro Operation) ★★★★★☆☆

<b>Fetch</b>  $MAR \leftarrow PC$ $MBR \leftarrow M(MAR), PC \leftarrow PC+1$ $IR \leftarrow MBR$	<b>Indirect</b>  $MAR \leftarrow MBR$ $MBR \leftarrow M(MAR)$
<b>Interrupt</b>  $MBR \leftarrow PC, PC=0$ $MAR \leftarrow PC, PC=PC+1$ $M(MAR) \leftarrow MBR, IEN=0$ goto Fetch	<b>Execute</b>  ADD LDA STA ISZ

MBR :

IR :

MAR :

PC :

★★★★☆☆

\*  $MAR \leftarrow PC$  : 명령을 수행하는 과정에서 가장 먼저 수행되어야 하는 마이크로 오퍼레이션

## [CA 7강]-메이저 스테이트

### 3. Interrupt (개념 설명)

0	
1	
100	
101	

MBR :

MAR :

PC :

MBR ← PC, PC = 0  
MAR ← PC, PC = PC + 1  
M(MAR) ← MBR, IEN = 0  
goto Fetch

\* IEN => 1 : 인터럽트 수행 중  
0 : 인터럽트 수행 완료

\* 메모리 0 번지에 복귀주소 기억

## [CA 7강]-메이저 스테이트

### 4. ADD ★☆☆☆☆

- AC의 내용과 메모리의 내용을 ADD -> 결과 AC

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

### 5. AND

- AC(누산기) 내용과 메모리 내용을 AND연산 -> 결과 AC

MAR ← MBR  
MBR ← M(MAR)  
AC ← AC AND MBR

### 6. LDA ★★☆☆☆

- 메모리 내용을 AC로 가져오는 것(load)

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

## [CA 7강]-메이저 스테이트

### 7. STA (Store AC) ★☆☆☆☆

- AC의 내용을 메모리 저장

MAR ← MBR  
MBR ← AC  
M(MAR) ← MBR

### 8. BUN (Branch unconditionally)

- PC에 특정한 주소를 전송하여 실행명령의 위치를 변경 -> 무조건 분기명령

PC ← MBR[AD]

### 9. ISZ (Increment and Skip if zero)

- 메모리의 값을 읽어, 그 값을 1 증가

MAR ← MBR  
MBR ← M(MAR)  
MBR ← MBR + 1  
M(MAR) ← MBR

## [CA 7강]-메이저 스테이트

1. 전자계산기의 중앙처리장치(CPU)는 4가지 단계를 반복적으로 거치면서 동작을 행한다. 4가지 단계에 속하지 않는 것은?

가. Fetch Cycle      나. Branch Cycle  
다. Interrupt Cycle    라. Execute Cycle

2. CPU가 인스트럭션을 수행하는 순서는?

㉠ 인터럽트 조사      ㉡ 인스트럭션 디코딩  
㉢ 인스트럭션 Fetch    ㉣ Operand Fetch  
㉤ Execution

가. ㉢-㉠-㉡-㉣-㉤      나. ㉢-㉡-㉣-㉤-㉠  
다. ㉡-㉢-㉣-㉤-㉠    라. ㉣-㉢-㉡-㉤-㉠

3. 명령(Instruction)이 실행되기 위해 가장 우선적으로 처리되어야 하는 마이크로 오퍼레이션은?

가. PC → MAR      나. PC → MBR  
다. PC → CPU      라. PC → M

4. 다음일련의 마이크로 오퍼레이션은 어느 사이클인가?

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

가. FETCH CYCLE      나. EXECUTE CYCLE  
다. INDIRECT CYCLE    라. INTERRUPT CYCLE

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

가. 인출(Fetch) 상태      나. 실행(Execute) 상태  
다. 간접(Indirect) 상태    라. 인터럽트(Interrupt) 상태

6. 인스트럭션의 수행 시 유효주소를 구하기 위한 메이저 상태를 무엇이라 하는가?

가. FETCH 메이저 상태      나. EXECUTE 메이저 상태  
다. INDIRECT 메이저 상태    라. INTERRUPT 메이저 상태

[정답] 1.나 2.나 3.가 4.가 5.가 6.다

## [CA 7강]-메이저 스테이트

7. 다음과 같은 마이크로 오퍼레이션이 일어나는 상태는?

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

가. Fetch 나. Indirect 다. Interrupt 라. Execute

8. 한 명령의 Execute Cycle 중에 Interrupt 요청이 있어 Interrupt를 처리한 후 전산기가 맞이하는 다음 사이클은?

가. Fetch Cycle 나. Indirect Cycle  
다. Execute Cycle 라. Direct Cycle

9. 간접(Indirect cycle) 동안에 컴퓨터는 무엇을 하는가?

가. 명령을 읽는다.  
나. 오퍼랜드(Operand)를 읽는다.  
다. 인터럽트(Interrupt)를 처리한다.  
라. 오퍼랜드(Operand)의 어드레스(Address)를 읽는다.

[정답] 7.다 8.가 9.라 10.라 11.나 12.다

10. 다음 마이크로 연산이 나타내는 동작은?

$MAR \leftarrow MBR(AD)$
$MBR \leftarrow M(MAR), AC \leftarrow 0$
$AC \leftarrow AC + MBR$

가. ADD to AC 나. OR to AC  
다. STORE to AC 라. LOAD to AC

11. 다음의 마이크로 오퍼레이션과 관련 있는 것은?

$MAR \leftarrow MBR(ADDR)$
$MBR \leftarrow M(MAR)$
$AC \leftarrow AC + MBR$

가. AND 나. ADD 다. JMP 라. BSA

12. 다음의 예는 실행 주기(Execute Cycle) 중에서 어떤 명령을 나타내는 것인가?

$MAR \leftarrow MBR(AD)$
$MBR \leftarrow M$
$MBR \leftarrow MBR + 1$
$M \leftarrow MBR, \text{if}(MBR = 0) \text{ then } (PC = PC + 1)$

가. JMP 나. AND 다. ISZ 라. BSA

## [CA 7강]-메이저 스테이트

[기-08년3월]

13. 프로그램에 의해 제어되는 동작이 아닌 것은?

가. Input/Output 나. Branch  
다. Status Sense 라. RNI(Fetch)

[기-08년3월]

14. 중앙처리장치가 Fetch 상태인 경우에 제어점을 제어하는 것은?

가. 플래그(Flag) 나. 명령어(Instruction)  
다. 인터럽트 호출 신호 라. 프로그램 카운터

[정답] 13.라 14.나