

13주 2강

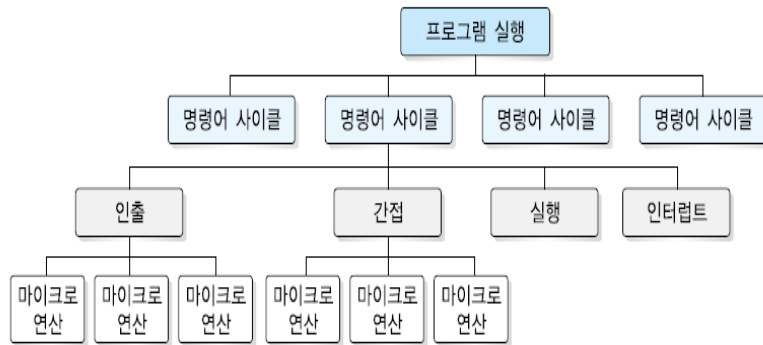
마이크로 연산과 제어장치 모델





완전한 명령어 사이클

- 인출 사이클:
 - 기억장치에서 명령어를 중앙처리장치로 읽어오는 과정
- 간접 사이클:
 - 간접 주소지정 방법을 사용하는 경우에 유효 주소를 찾는 사이클
- 실행 사이클:
 - 여러 종류의 명령어가 해당 명령어에 맞게 동작을 실행하는 단계다.
- 인터럽트 사이클:
 - 인터럽트가 발생시 ISR이 수행 될 수 있도록 해주는 사이클
- 각 부 사이클의 실행은 여러 단계의 마이크로 연산으로 구성된다.



인출 사이클(Fetch cycle)의 마이크로 연산



CPU 클럭	마이크로 연산
t_0	$MAR \leftarrow PC$
t_1	$MBR \leftarrow M(MAR)$ $PC \leftarrow PC + 1$
t_2	$IR \leftarrow MBR$

CPU 클럭	마이크로 연산
t_0	$MAR \leftarrow PC$
t_1	$MBR \leftarrow M(MAR)$
t_2	$IR \leftarrow MBR$ $PC \leftarrow PC + 1$

- t_0 : 다음 명령어의 주소가 저장된 PC 내용을 MAR로 이동
- t_1 : MAR에 저장된 주소번지의 기억장치에 저장되어 있는 명령어를 MBR로 이동
동시에 PC를 증가시켜서 다음 명령어의 주소를 표시
- t_2 : MBR에 저장된 명령어 내용을 실행하기 위해서 IR로 이동
- PC의 증가 단계는 세 번째 클럭에서 수행될 수도 있다.

간접 사이클의 마이크로 연산



CPU 클럭	마이크로 연산
t_0	$MAR \leftarrow IR(addr)$
t_1	$MBR \leftarrow M(MAR) \text{ OR } R(MAR)$
t_2	$IR(addr) \leftarrow MBR(addr)$

- t_0 : 명령어 레지스터에서 주소 부분을 MAR로 이동한다.
- t_1 : 유효 주소가 저장된 기억장치 또는 레지스터의 내용을 MBR에 저장한다.
- t_2 : MBR의 저장되어 있는 유효 주소를 명령어 레지스터의 주소부분으로 이동시킨다.
 - 명령어 레지스터의 명령어 오퍼랜드 부분에 유효 주소가 자리를 잡게 된다. 그리고 실행사이클로 넘어 갈 것이다.

실행 사이클의 마이크로 연산



● ADD 명령어의 어셈블리 언어 표현

ADD R, X ; $R \leftarrow R + \text{Memory}(X)$

- 기억장치 X번지 내용과 레지스터 R이 덧셈하고 그 결과는 R에 저장한다.

● 마이크로 연산 표현

CPU 클럭	마이크로 연산
t_0	$MAR \leftarrow IR(addr)$
t_1	$MBR \leftarrow M(MAR)$
t_2	$R \leftarrow R + MBR$

- t_0 : 명령어 레지스터의 주소 부분만을 MAR로 이동시킨다.
액세스할 데이터가 저장된 기억장치 주소를 얻는 과정이다.
- t_1 : 해당 주소의 기억장치 내용을 MBR로 이동시킨다.
덧셈연산을 위한 데이터가 MBR로 이동하게 되는 것이다.
- t_2 : 기억장치에서부터 이동된 데이터가 저장된 MBR의 내용과 R의 내용을 더하고, 그 결과값을 다시 R에 저장하게 된다.

인터럽트 사이클의 마이크로 연산



- 인터럽트 요청이 있는지 여부를 판단하고, 요청이 있을 경우 인터럽트 서비스 루틴이 시작되도록 해야 한다.
- 인터럽트 사이클의 마이크로 연산

CPU 클럭	마이크로 연산
t_0	$MBR \leftarrow PC$
t_1	$MAR \leftarrow SP$ $PC \leftarrow \text{Routine-address}$
t_2	$M(MAR) \leftarrow MBR$

- t_0 : 복귀주소를 저장하기 위해서 프로그램 카운터 내용을 MBR로 이동
- t_1 : SP를 MAR로 이동시키고 ISR 시작 주소를 PC에 적재한다.
SP는 MBR 내용을 스택에 저장하기 위해서 저장할 위치를 지정
- t_2 : MBR의 내용을 MAR에 근거하여 해당 스택 포인터의 스택에 저장된다



★ 중앙처리장치를 제어하기 위해 필요한 기능적 요구 사항

- 중앙처리장치가 갖추어야 할 필수적인 구성 요소
 - 산술 및 논리 연산장치(ALU)
 - 레지스터 집합
 - 내부 및 외부 데이터 통로
 - 제어장치
- 중앙처리장치가 수행할 마이크로 연산의 정의
 - 레지스터들 간의 데이터 전송
 - 레지스터에서 외부로의 데이터 전송
 - 외부에서 레지스터로의 데이터 전송
 - 산술 및 논리 연산 수행
- 제어장치가 수행할 기능의 정의
 - 순서제어(Sequencing)기능 : 중앙처리장치가 마이크로 연산을 적절한 순서대로 처리하도록 해주는 기능이 필요
 - 실행기능 : 다양한 마이크로 연산에 대한 수행 기능이 필요



기본적인 제어장치의 모델

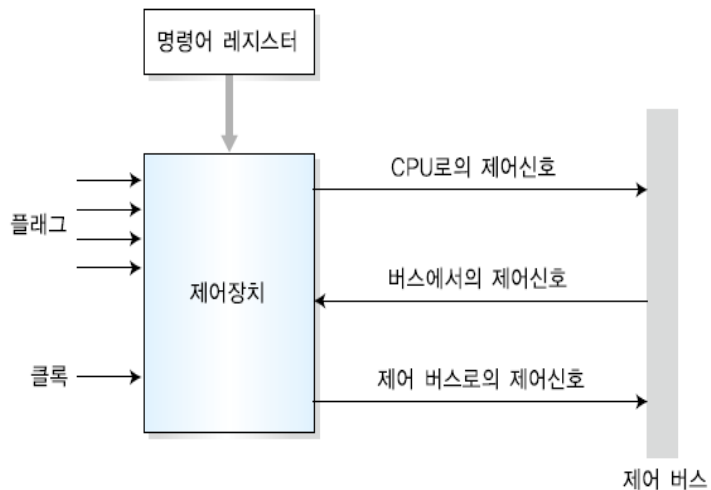


● 제어장치의 입력

- 클럭, 명령어 레지스터에서의 연산코드, 플래그(flag) 신호, 제어 버스를 통해 입력되는 제어 신호

● 제어장치의 출력

- 중앙처리장치로 보내는 제어 신호와 제어 버스를 통해서 외부로 보내는 제어 신호가 존재





★ 제어장치로 들어오는 입력 신호

- 클록(Clock) 신호
 - 제어장치의 동작 타이밍 기준이 되는 신호,
 - 하나의 클록 펄스마다 마이크로 연산이 수행된다.
- 명령어 레지스터
 - 명령어의 연산 코드 필드를 제어장치에 입력하여 수행해야 할 연산을 결정
- 플래그(flag)들
 - 중앙처리장치의 상태, 산술 및 논리 연산장치의 연산 결과를 검사
- 제어 버스에서 입력되는 제어 신호
 - 인터럽트 처리 요구 및 확인 신호, 외부에서 들어오는 제어 신호다.

★ 제어장치에서 내보내는 출력 신호

- 중앙처리장치로 전달되는 제어 신호 : 레지스터 간 데이터를 전송할 때 발생되며, 특정 산술 및 논리 연산의 기능을 활성화할 때 출력 된다.
- 제어 버스를 통해 외부로 출력되는 제어 신호 : 중앙처리장치 밖의 기억장치나 입출력장치 모듈로 보내는 제어 신호가 존재한다.

다음 시간

13주 3강. 마이크로 프로그램을 이용한 제어

