

## Chapter 3. 최상위에서 본 컴퓨터의 기능 및 상호연결

### 복습문제

#### 3.1 폰노이만 구조의 세 가지 핵심개념들을 설명하라

데이터와 명령어는 읽고 쓰기가 가능한 기억장치에 함께 저장된다.

데이터의 형식에 관계없이 주소로 지정한다.

명령어는 한 개씩 순서대로 진행된다.

#### 3.2 I/O 주소 레지스터 및 I/O 버퍼 레지스터 간의 차이점을 설명하라.

I/O 주소 레지스터: I/O 장치의 주소를 지정한다.

I/O 버퍼 레지스터: I/O 모듈과 CPU간 데이터를 일시 저장한다.

#### 3.3 인터럽트를 불가능하게 하는 방식의 단점은 무엇인가?

상대적 우선순위 혹은 시간적 위급성을 고려하지 않는다.

먼저 수신된 데이터는 다음 데이터가 수신될 때까지 처리되지 않으면 소실될 수 있다.

### 연습문제

#### 3.2 그림 3.5에서의 프로그램 실행은 여섯 단계이다. 이를 MAR와 MBR 사용을 포함해 확장하라.

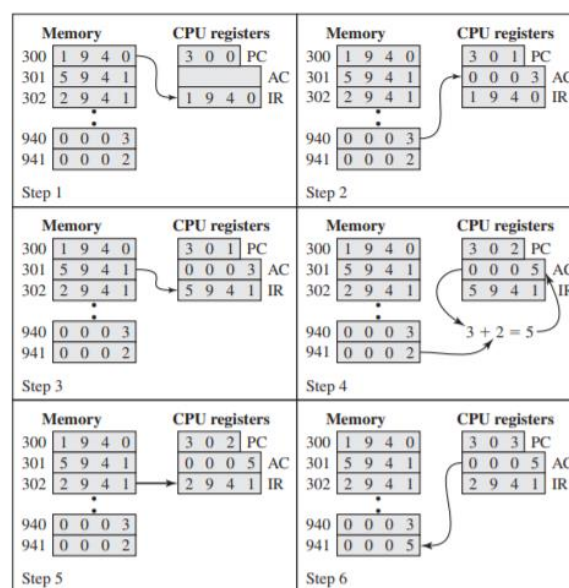


Figure 3.5 Example of Program Execution (contents of memory and registers in hexadecimal)

다음에 실행될 명령어의 opcode가 IR에 적재되고, 주소 부분은 MAR에 적재된다.

그 명령어는 IBR로부터 가져오거나, MBR에 그 단어를 적재한 후 기억장치로부터 읽어온 다음에 IBR, IR 및 MAR로 보내진다.

3.3 두 개의 필드들로 구성된 32비트 명령어를 가진 가상의 32비트 마이크로 프로세서가 있다. 한 바이트로 이루어진 첫째 필드는 opcode를, 나머지는 즉치 오퍼랜드 혹은 오퍼랜드의 주소를 가진다.

a) 직접 주소 지정할 수 있는 기억장치의 용량은?

OPCODE 8bits	OPERAND 24bits
--------------	----------------

$$2^{24} = 16\text{Mbytes}$$

b) 마이크로 프로세서 버스가 다음과 같다면, 시스템 속도에 어떤 영향을 미치겠는가?

1. 32비트 지역주소 버스와 16비트 지역 데이터 버스
2. 16비트 지역주소 버스와 16비트 지역 데이터 버스

1: 주소버스는 32비트여서 모든 메모리 영역 주소를 전송할 수 있으나,

데이터 버스는 32비트 명령어나 오퍼랜드를 읽기 위해 두 번 이상의 메모리 접근이 필요하다.

2: 주소버스는 한 번에 모든 메모리 영역 주소에 접근할 수 없으므로 두 단계에 나눠 주소 연산이 되어야 하며, 데이터 버스에서도 두 번 이상의 메모리 접근이 필요하다.

주소 전달이 늦어지면 데이터 전달도 늦어져, 결과적으로 시스템 속도가 저하된다.

c) 프로그램 카운터와 명령어 레지스터는 각각 몇 비트로 구성되는가?

PC: 24bits 이상. 다음에 인출할 명령어의 주소이므로.

IR: 32bits 이상. 32비트 명령어를 가지므로.

3.4 16비트 주소(프로그램 카운터와 주소 레지스터 폭이 16비트라고 가정)를 발생시키고, 16비트 데이터 버스를 가진 가상의 마이크로 프로세서가 있다고 가정하자.

a) 프로세서가 16비트 기억장치와 접속되어 있다면, 프로세서가 직접 액세스 할 수 있는 최대 기억장치 주소 공간은 얼마인가?

16비트 주소 발생  $\rightarrow 2^{16}$ 개의 주소공간

16비트짜리 기억장치  $\rightarrow 2^{16} \times 16\text{bits}$ 만큼의 주소 공간

$$2^{17} = 128\text{Kbytes}$$

b) 프로세서가 8비트 기억장치와 접속되어 있다면, 프로세서가 직접 액세스 할 수 있는 최대 기억장치 주소 공간은 얼마인가?

16비트 주소 발생  $\rightarrow 2^{16}$ 개의 주소공간

8비트짜리 기억장치  $\rightarrow 2^{16} \times 8\text{bits}$ 만큼의 주소 공간

$$2^{16} = 64\text{Kbytes}$$

c) 이 프로세서가 분리된 I/O 공간을 액세스 할 수 있게 되려면 어떤 구조적 특성을 가져야 하는가?

Isolated I/O: 메모리와 I/O가 별개의 주소공간에 할당되어 CPU 입장에서는 메모리와 I/O를 구분하여 취급해야 한다. 이들을 액세스 하기 위한 제어신호는 r/w 외에 memory request나 I/O request 처럼 독립된 명령어를 가져야 한다.

d) 입출력 명령어가 8비트 I/O 포트 번호를 지정할 수 있다면, 이 마이크로 프로세서는 몇 개의 8비트 I/O 포트들을 지원할 수 있는가? 16비트 I/O 포트 번호라면?

포트번호=설치할 수 있는 I/O 장치의 수. 따라서

8비트 I/O 포트 번호  $\rightarrow 2^8$ 개.

16비트 I/O 포트 번호  $\rightarrow 2^{16}$ 개.