중앙 처리 장치
CPU; Central Process Unit

01 프로세서 구성과 동작

1 컴퓨터 기본 구조와 프로세서

- 컴퓨터의 3가지 핵심 장치 : 프로세서(Processor, CPU), 메모리, 입출력 장치
- 버스(Bus) : 장치 간에 주소, 데이터, 제어 신호를 전송하기 위한 연결 통로(연결선)

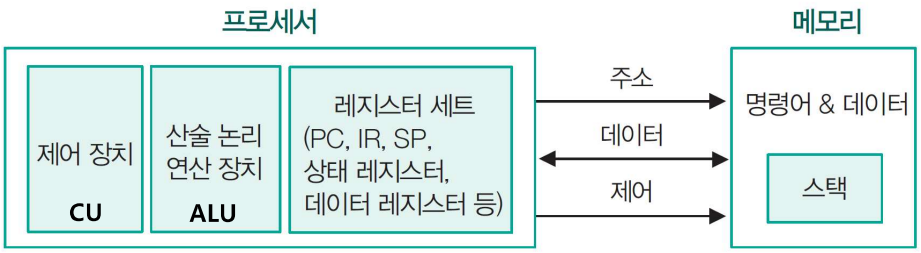
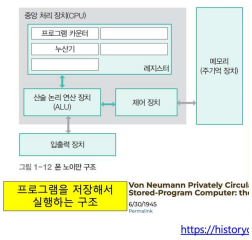


그림 4-1 폰 노이만 컴퓨터의 기본 구조

1945년 (현대) 컴퓨터 구조에 대한 생각

John von Neumann

맨해튼 프로젝트 멤버



https://history

01 프로세서 구성과 동작

- 버스(Bus) : 장치간에 주소, 데이터, 제어 신호를 전송하기 위한 연결 통로(연결선)
  - 내부 버스(internal bus) : 프로세서 내부의 장치 연결
  - 시스템 버스(system bus) : 핵심 장치 및 주변 장치 연결

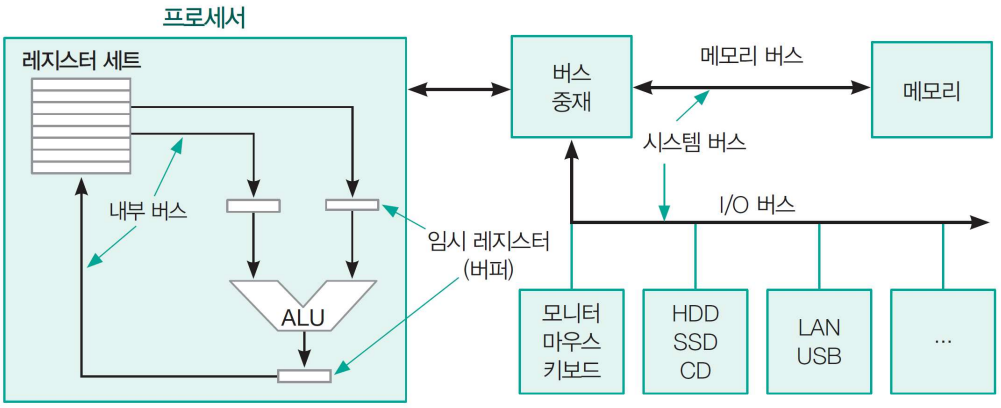
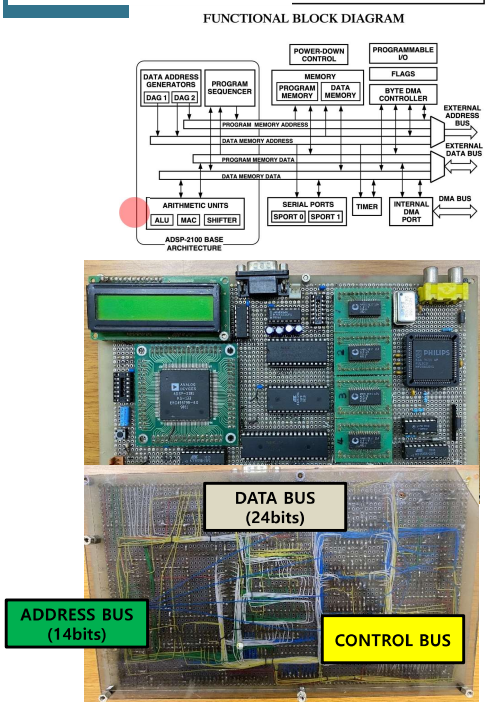


그림 4-2 버스 기반 컴퓨터 구조

DIY 영상처리보드 1997년 ADSP-2181

*시험범위제외



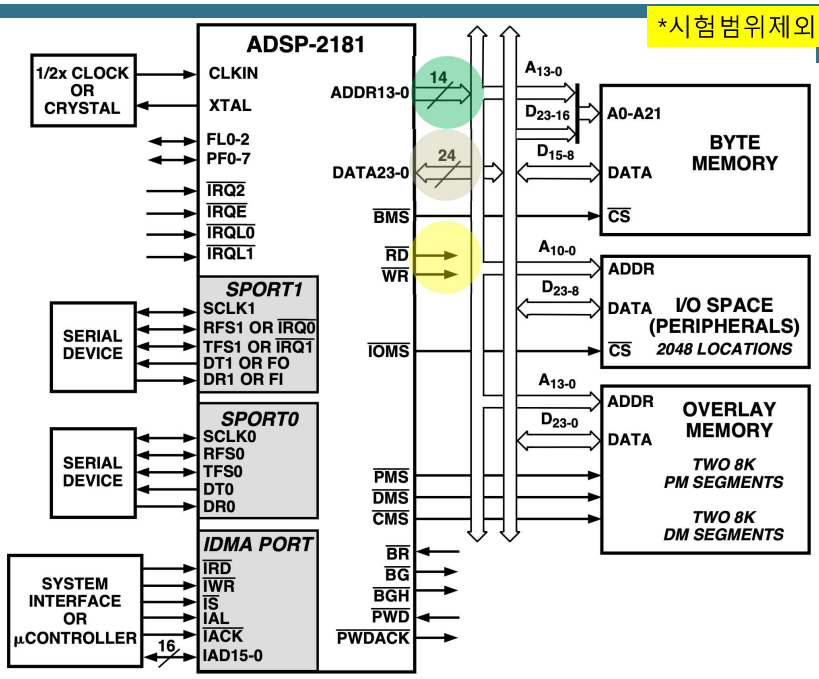
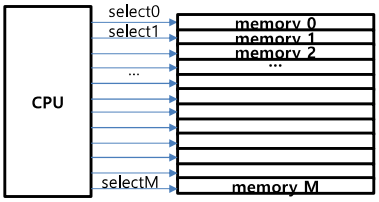


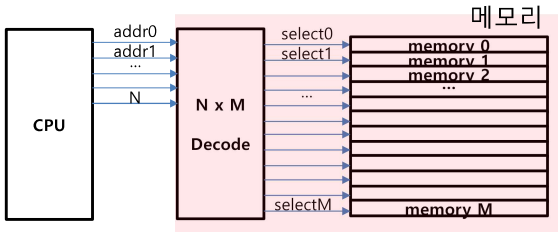
Figure 2. ADSP-2181 Basic System Configuration

Address BUS

- 수학에서 K = 1000, 컴퓨터에서 **K**ilo = **1024**, Mega = 1024 * 1024, Giga = 1024 * 1024 * 1024
- **16bit Address Bus를 갖는 CPU는 직접 지정할 수 있는 메모리 번지가 2^{16}개 있음**
  - 2^{16} = 0 ~ 65535 = **0000**h ~ **FFFF**h(16진수;hex)

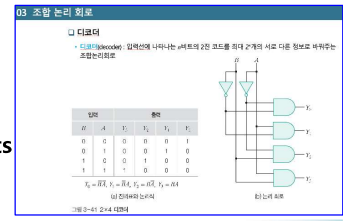


예) 65536개 line이 필요



예) addr 16개, 16 x 65536 Decode 사용

- 16bit Address Bus, 8bit(**1Byte**) Data Bus인 경우, 65536개 x **1Byte** = 64KB
- 16bit Address Bus, 16bit(**2Byte**) Data Bus인 경우, 65536개 x **2Byte** = 128KB
- **M개 용량을 갖는 메모리에 필요한 Address Bus Bit (N): 2^N >= M; N >= $\log_2 M$**
- **예제) 64KB;2^6(=64)*2^{10}(=1024)=2^{16}(=65536) 메모리에 필요한 Address Bus의 크기는 16bits**



01 프로세서 구성과 동작

2 프로세서 구성 요소

❖ **프로세서 3가지 구성 필수 구성요소**

- 산술 논리 연산 장치(Arithmetic Logic Unit, ALU) : 산술 및 논리 연산 등 기본 연산을 수행
- 제어 장치 (Control Unit, CU) : 메모리에서 명령어를 가져와 해독하고 실행에 필요한 장치들을 제어하는 신호를 발생
- 레지스터 세트(register set) : 프로세서 내에 존재하는 용량은 작지만 매우 빠른 메모리, ALU의 연산과 관련된 데이터를 일시 저장하거나 특정 제어 정보 저장

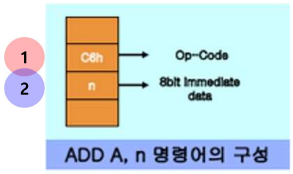
  - 목적에 따라 특수 레지스터와 범용 레지스터로 분류

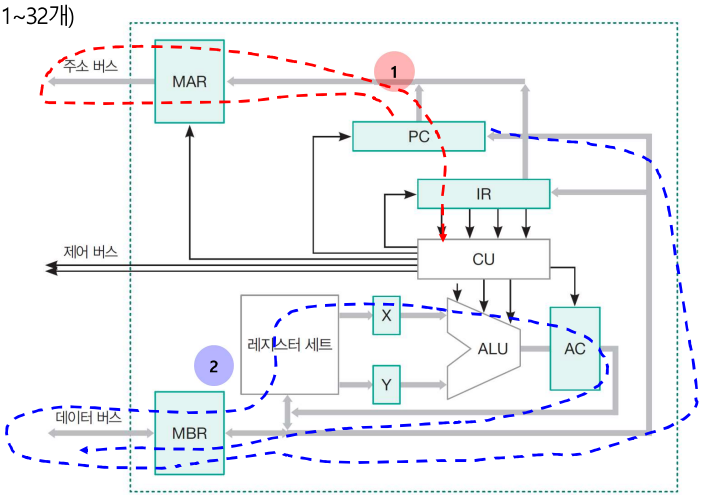
- 현재는 온칩 캐시(on-chip cache), 비디오 컨트롤러(video controller), 실수보조연산 프로세서(FPU) 등 다양한 장치 포함

01 프로세서 구성과 동작

3 프로세서 기본 구조

- 레지스터 세트(일반적으로 1~32개)
- ALU
- CU
- 이들 장치를 연결하는 버스로 구성





MAR Memory Address Register: 메모리 주소 레지스터 PC Program Counter: 프로그램 카운터 CU Control Unit: 제어 장치
AC ACcumulator: 누산기 MBR(MDR) Memory Buffer(Data) Register: 메모리 버퍼(데이터) 레지스터
IR Instruction Register: 명령 레지스터 ALU Arithmetic Logic Unit: 산술 논리 연산 장치

그림 4-3 프로세서 기본 구조

01 프로세서 구성과 동작

❖ ALU

- 덧셈, 뺄셈 등 연산을 수행하고, 그 결과를 누산기(Accumulator, AC)에 저장

❖ 프로세서 명령 분류

레지스터-메모리 명령	• 메모리 워드를 레지스터로 가져올(LOAD) 때 • 레지스터의 데이터를 메모리에 다시 저장(STORE)할 때
레지스터-레지스터 명령	• 레지스터에서 오퍼랜드 2개를 ALU의 입력 레지스터로 가져와 덧셈 또는 논리 AND 같은 몇 가지 연산을 수행하고 • 그 결과를 레지스터 중 하나에 다시 저장

01 프로세서 구성과 동작

4 프로세서 명령 실행 (Instruction Cycle)

- 프로세서는 각 명령을 더 작은 마이크로 명령(microinstruction)들로 나누어 실행

  1단계 다음에 실행할 명령어를 메모리에서 읽어 명령 레지스터(IR)로 가져온다.

  2단계 프로그램 카운터(PC)는 그다음에 읽어올 명령어의 주소로 변경된다.

  3단계 제어 장치는 방금 가져온 명령어를 해독(decode)하고 유형을 결정한다.

  4단계 명령어가 메모리에 있는 데이터를 사용하는 경우 그 위치를 결정한다.

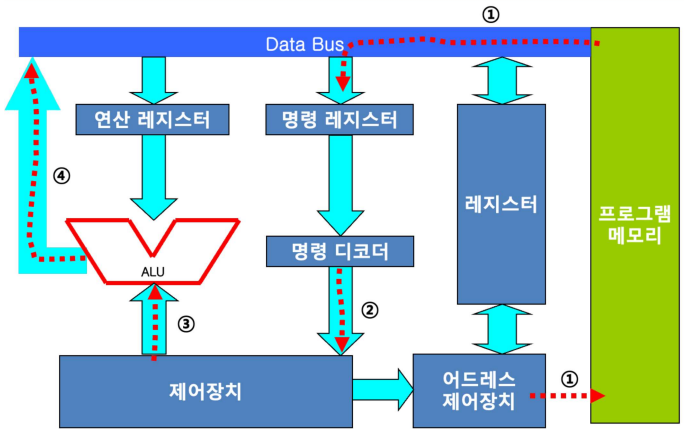
  5단계 필요한 경우 데이터를 메모리에서 레지스터로 가져온다.

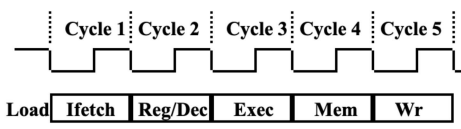
  6단계 명령어를 실행한다.

  7단계 1단계로 이동하여 다음 명령어 실행을 시작한다.

- 이 단계를 요약하면 인출(**fetch**)-해독(**decode**)-실행(**execute**) 사이클로 구성 – 주 사이클(**main cycle**)

Instruction Cycle > Machine Cycle = Fetch + Decode + Execute + Write





- **Ifetch:** Instruction Fetch
- **Reg/Dec:** Registers Fetch and Instruction Decode
- **Exec:** Calculate the memory address
- **Mem:** Read the data from the Data Memory
- **Wr:** Write the data back to the register file

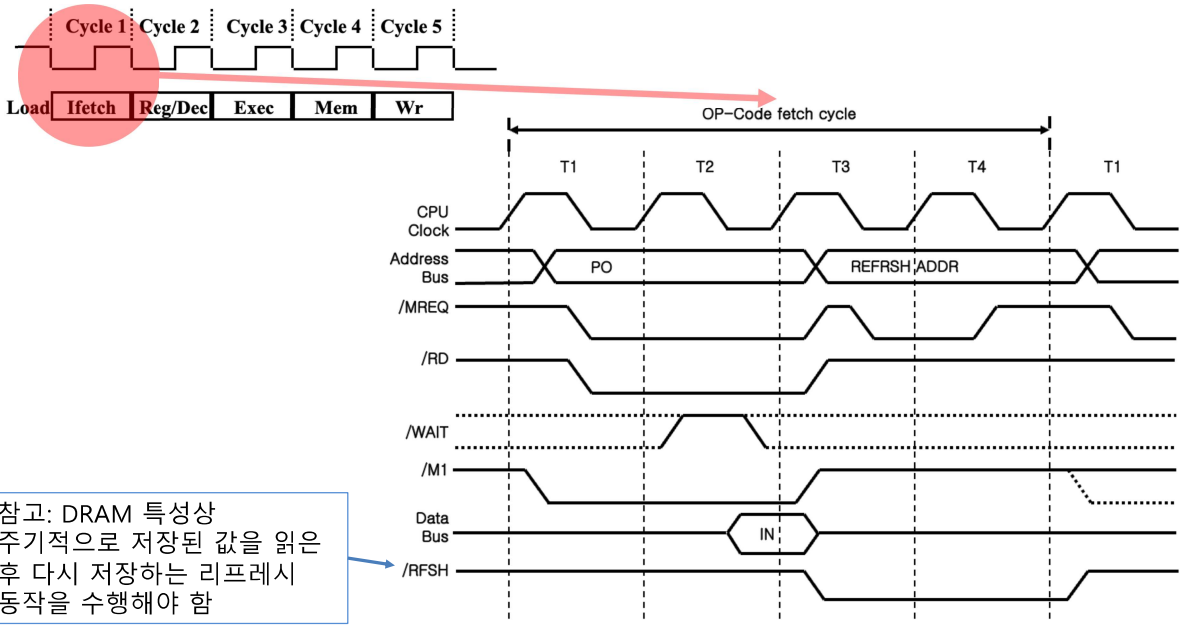
- **CPU 내부의 명령 처리(일반적으로 4단계의 과정)**

  *그림에서 보는 것과 같이, 점선으로 표시된 부분이 명령어 처리과정이다.

  - ① : Instruction **Fetch**
    - 메모리에서 명령어를 읽어가는 과정을 Instruction Fetch라고 하며, Fetch Cycle이라고도 한다.
  - ② : Instruction **Decode**
    - 명령 레지스터에 저장된 명령어를 명령 디코더에서 해독을 하게된다. Decode Cycle이라고 한다.
  - ③ : Instruction **Execute**
    - 해독된 명령어에 따라 제어장치를 이용하여 CPU가 실행을 하게 된다. Execute Cycle이라고도 한다.
  - ④ : Write Back
    - 처리된 결과를 레지스터 또는 메모리에 저장한다.

- **Instruction Cycle**
  - 하나의 명령어를 처리하는 전체 과정을 말하며, 여기서는 ① + ② + ③ + ④ = 하나의 Instruction Cycle이 된다.

Opcode Fetch Cycle Timing Chart



참고: DRAM 특성상
주기적으로 저장된 값을 읽은 후 다시 저장하는 리프레시 동작을 수행해야 함

01 프로세서 구성과 동작

❖ **해독기**(microprogrammed control) : 하드웨어를 소프트웨어로 대체

- 고가의 고성능 컴퓨터는 하드웨어 추가 비용이 크게 부담되지 않아 저가 컴퓨터보다 많은 명령어를 갖게 됨
- 고가인 고성능 컴퓨터의 복잡한 명령어를 저가 컴퓨터에서 실행할 수 있게 하기 위함
- 모리스 윌크스(Maurice Wilkes)가 제안(1951년)
  - 1957년 SDSAC 1.5에 적용
- 1970년대 설계된 거의 모든 컴퓨터가 해독기를 기반
  - Cray-1 같은 매우 고가의 고성능 모델을 제외하고는 1970년대 후반에 해독기를 운영하는 프로세서가 보편적으로 보급
  - 복잡한 명령어에 대한 비용 절감, 훨씬 더 복잡한 명령어 연구
- 제어 기억 장치(control memory)라는 빠른 읽기 전용 메모리