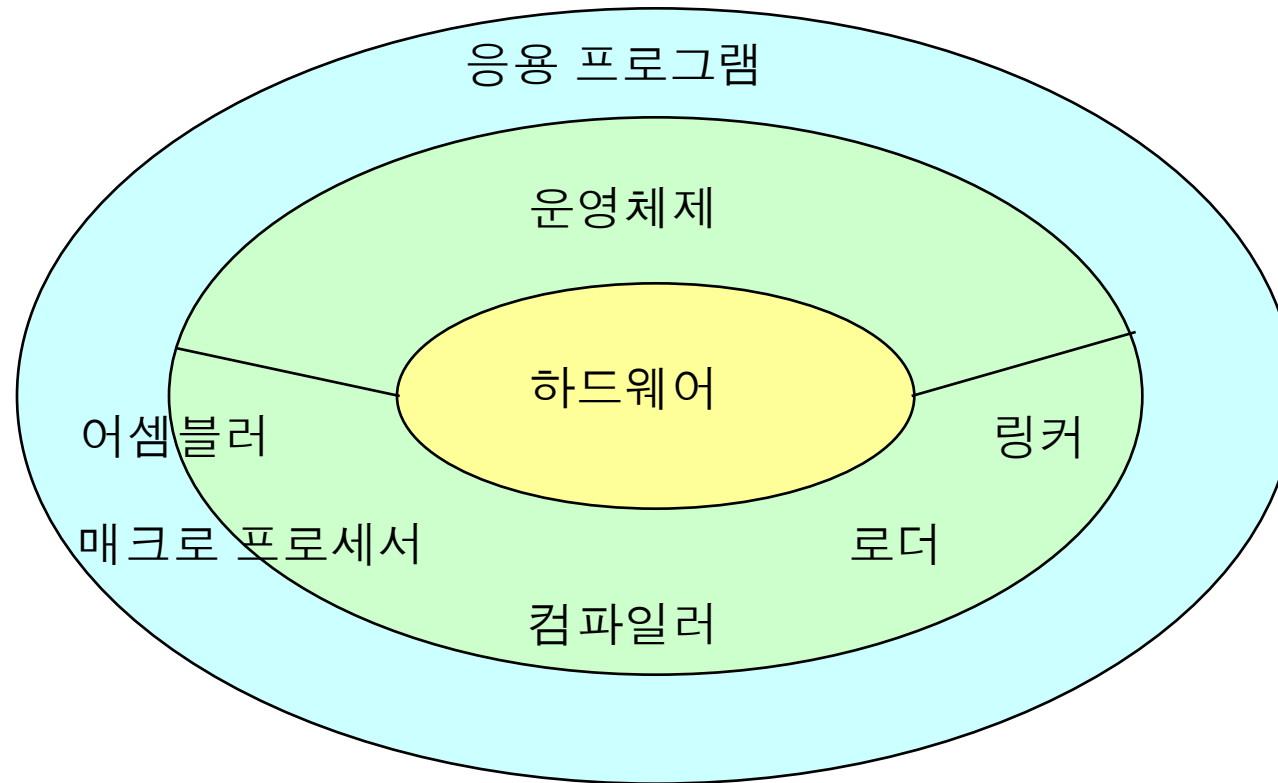


시스템 소프트웨어 개요

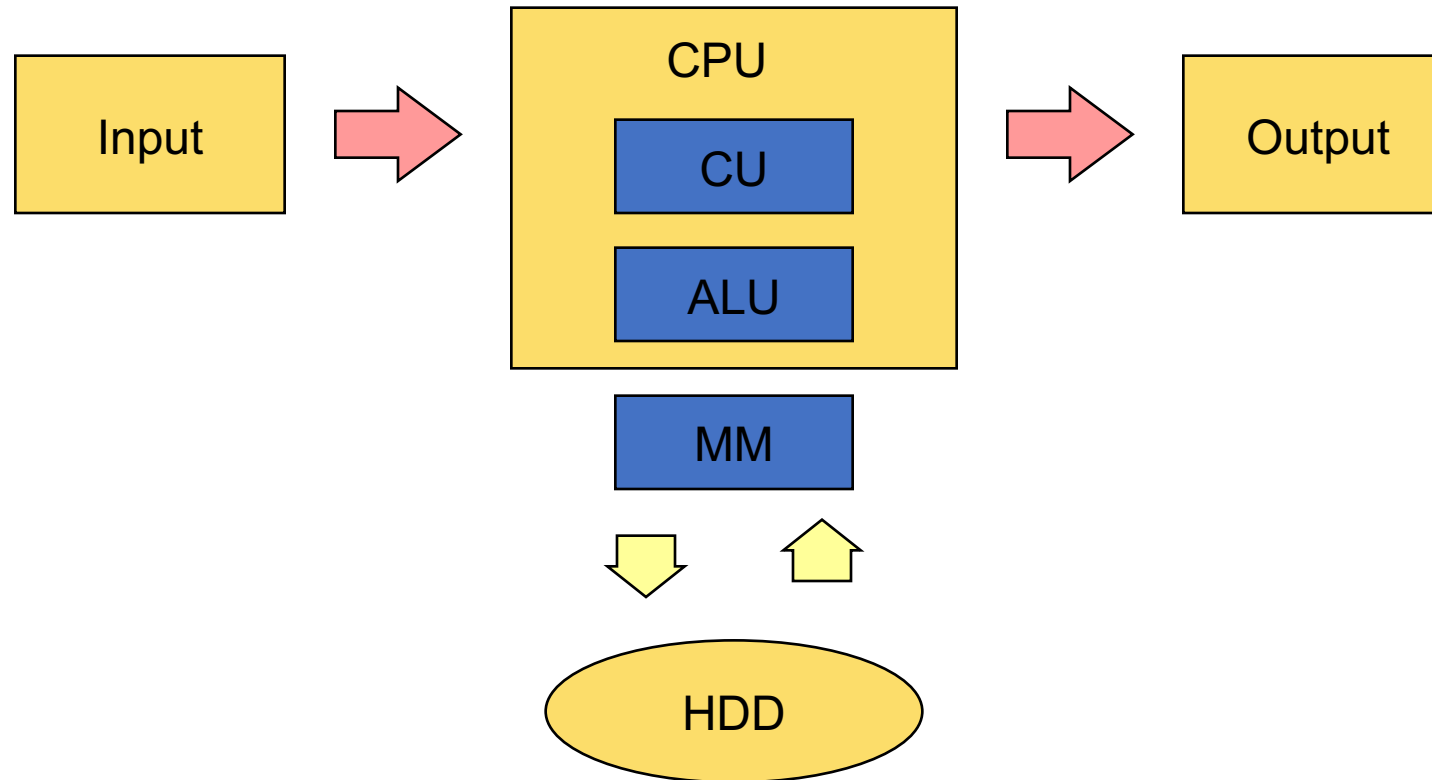
컴퓨터 시스템

- 컴퓨터 시스템의 구성

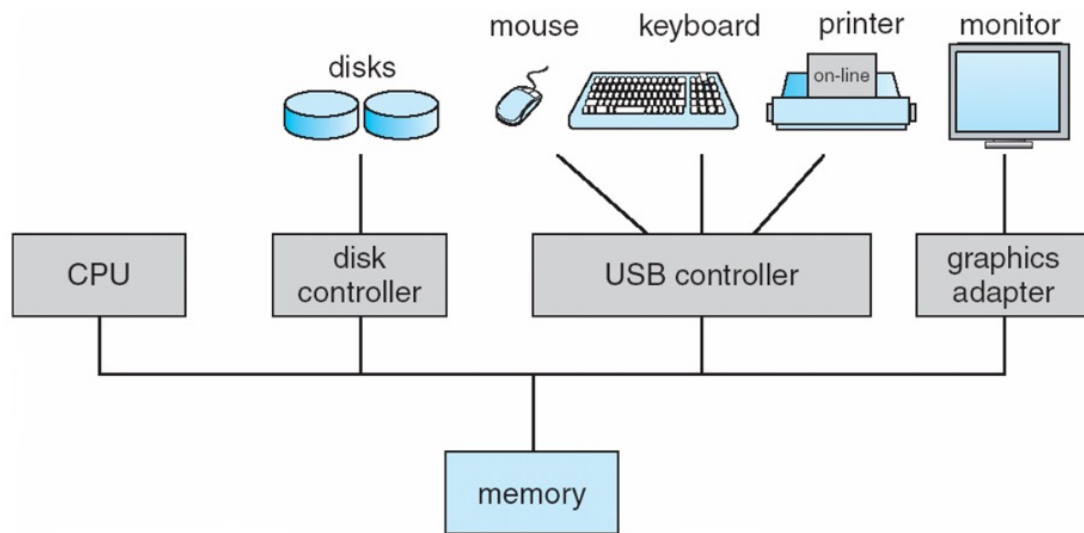


컴퓨터 시스템

- 컴퓨터 하드웨어

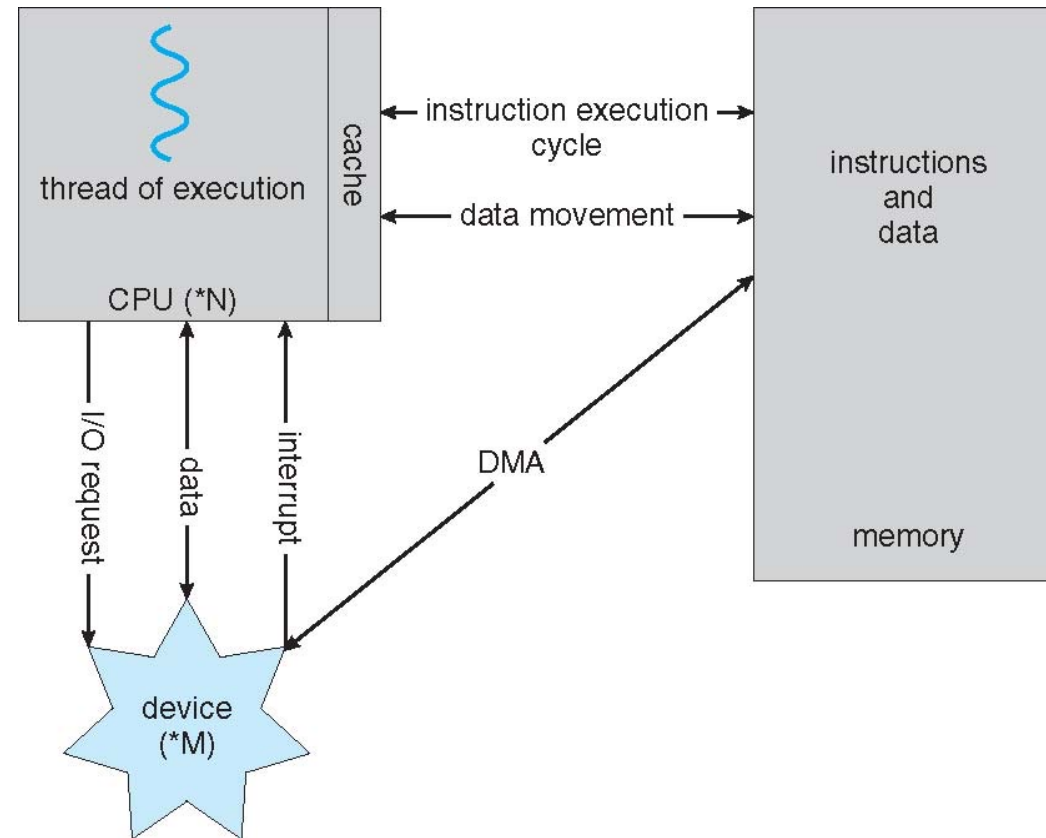


Basic Elements



- Processor (CPU)
- Main Memory
 - volatile
 - referred to as real memory or primary memory
- I/O devices
 - secondary memory devices
 - communications equipment
 - terminals
- System bus
 - communication among processors, memory, and I/O modules

How a Modern Computer Works



컴퓨터 시스템

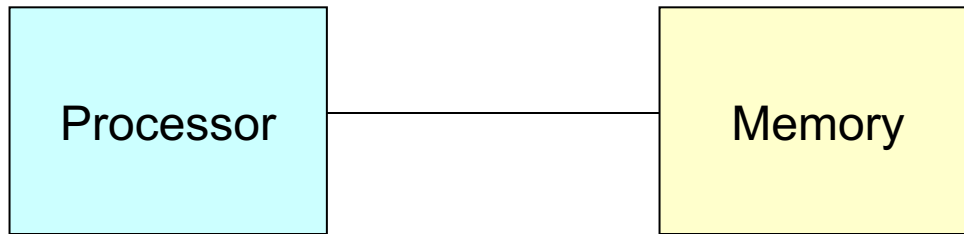
- 컴퓨터 프로그램
 - CPU가 실행해야 할 구체적인 내용을 한 단계씩 순서대로 기술한 명령(instruction)의 연속
- 기계어 (machine language)
 - 0과 1로 표시된 것으로 컴퓨터의 프로세서가 이해하는 유일한 언어
 - 각 기계간에 호환성이 없음

Fetch-Execution Cycle

- ENIAC does not store program
 - 플러그와 점퍼 케이블을 가지고 연결
- Stored program 방식의 컴퓨터
 - von Neumann
 - 현재의 대부분의 컴퓨터
- 계산 -> 프로그램의 수행(running)을 의미
- memory: 프로그램이 저장된 장소
- processor: 계산을 수행하는 기계의 부분
- computer program: list of computer instruction

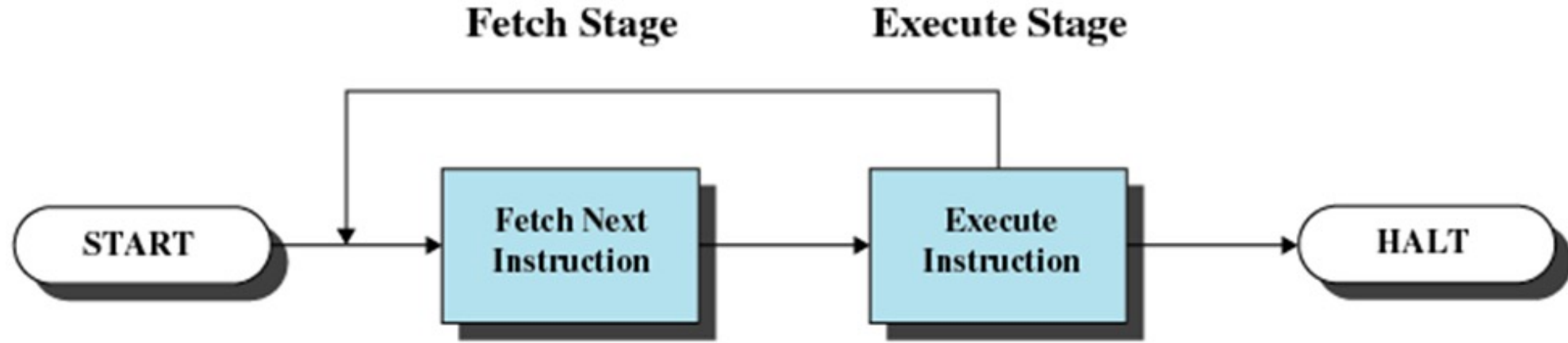
Fetch-Execution Cycle

- Von Neumann Computer



- 프로세서에 의한 계산은 Fetch-Execution Cycle을 사용하여 이루어진다.
 1. The processor fetches an instruction from memory
 2. The processor executes the instruction
 3. The processor cycles back to step “fetch”

Fetch-Execution Cycle



Processor Registers

- User-visible registers
 - Enable programmer to minimize main-memory references by optimizing register use
- Control and status registers
 - Used by processor to control operating of the processor
 - Used by privileged operating-system routines to control the execution of programs

User-Visible Registers

- May be referenced by machine language
- Available to all programs - application programs and system programs
- Types of registers
 - Data
 - Address
 - Index
 - Segment pointer
 - Stack pointer

User-Visible Registers

- Address Registers
 - Index
 - Involves adding an index to a base value to get an address
 - Segment pointer
 - When memory is divided into segments, memory is referenced by a segment and an offset
 - Stack pointer
 - Points to top of stack

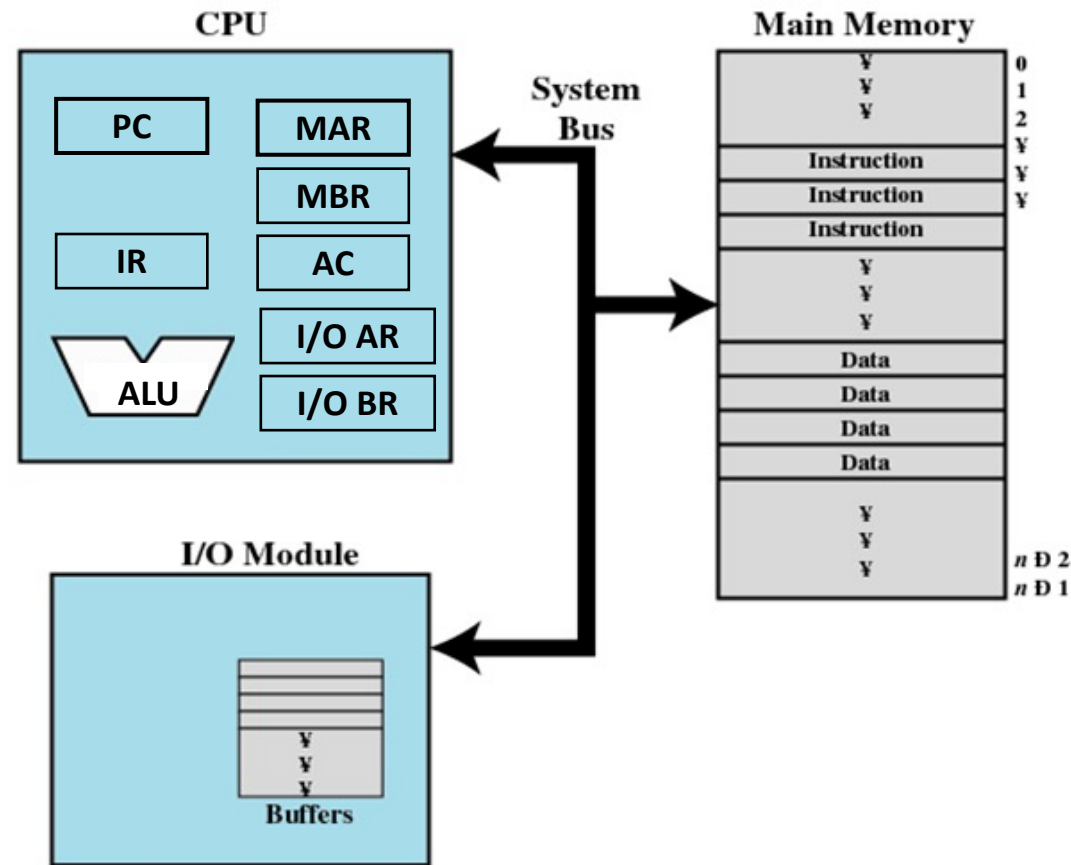
Control and Status Registers

- Program Counter (PC)
 - Contains the address of an instruction to be fetched
- Instruction Register (IR)
 - Contains the instruction most recently fetched
- Program Status Word (PSW)
 - Condition codes
 - Interrupt enable/disable
 - Supervisor/user mode

Control and Status Registers

- Condition Codes or Flags
 - Bits set by the processor hardware as a result of operations
 - Examples
 - Positive / Negative result
 - Zero
 - Overflow

Program Execution of a Hypothetical Machine



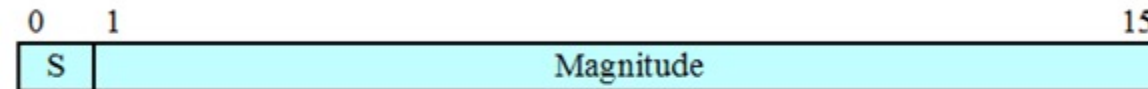
Processor Registers

- Memory address register (MAR)
 - Specifies the address for the next read or write
- Memory buffer register (MBR)
 - Contains data written into memory or receives data read from memory
- Accumulator(AC)
- I/O address register
- I/O buffer register

Characteristics of a Hypothetical Machine



(a) Instruction format



(b) Integer format

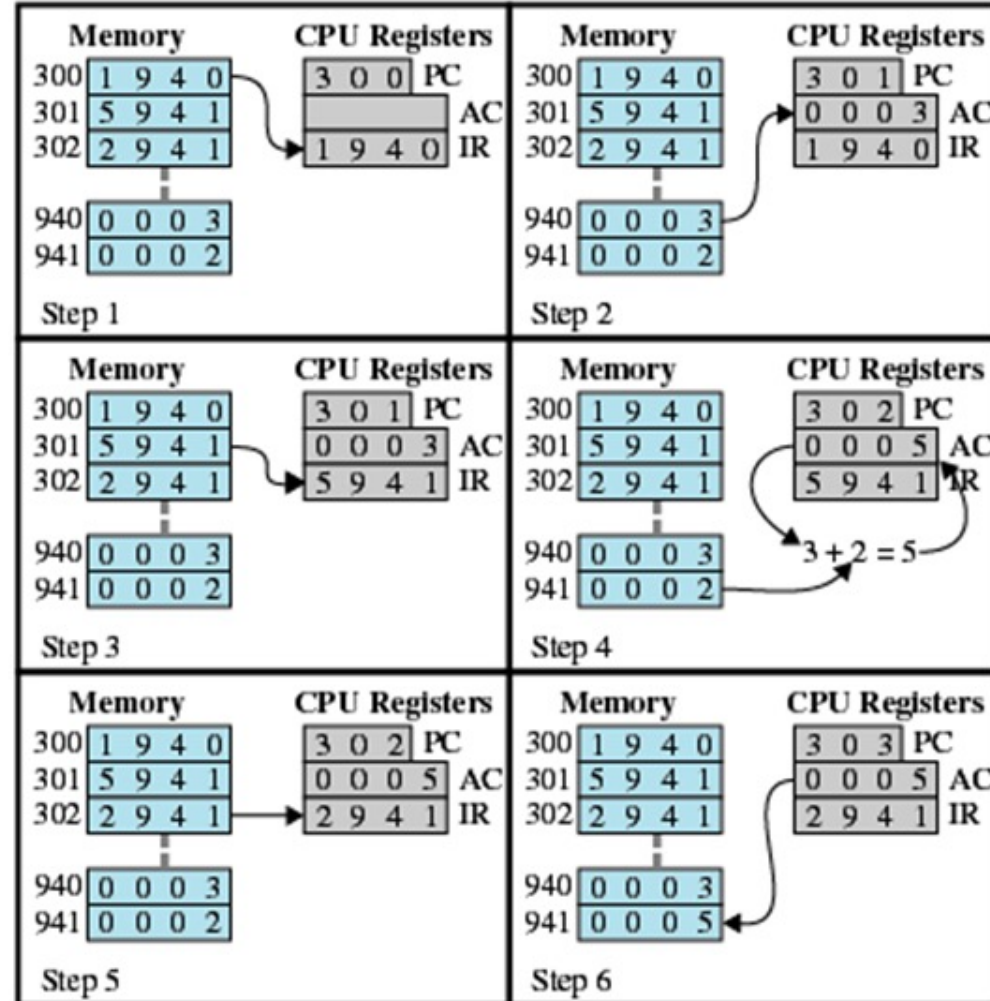
Program Counter (PC) = Address of instruction
Instruction Register (IR) = Instruction being executed
Accumulator (AC) = Temporary storage

(c) Internal CPU registers

0001 = Load AC from Memory
0010 = Store AC to Memory
0101 = Add to AC from Memory

(d) Partial list of opcodes

Example of Program Execution



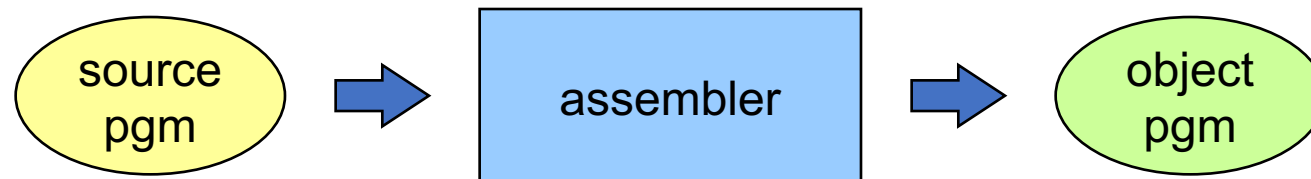
시스템 소프트웨어

- 소프트웨어의 구분
 - 응용 소프트웨어 : 최종 사용자를 목적을 만족시키는 소프트웨어(프로그램)
 - 시스템 소프트웨어 : 다른 프로그램 또는 소프트웨어가 필요한 기능을 제공
 - 시스템 프로그래밍 : 시스템 소프트웨어의 기능을 이용하여 기계를 직접 작동하는 프로그램을 작성하는 일

시스템 소프트웨어

- **Assembler**

- 기계어 vs. 어셈블리어(assembly language)
 - mnemonic instructions
 - symbolic address
- 어셈블리어를 자동적으로 기계어로 번역하는 프로그램



시스템 소프트웨어

- Linker

- 여러 개의 모듈간의 상호 기억 장소 참조를 정리하여 함께 실행될 수 있도록 함

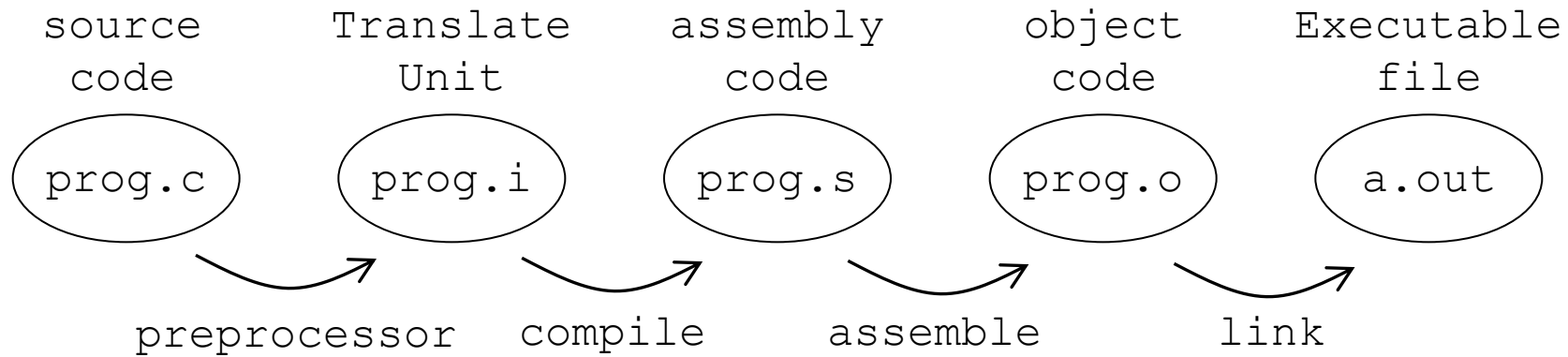
- Loader

- load module을 기억장치에 적재하는 프로그램
- 어셈블러가 로더의 기능까지 포함하는 경우
 - 기억 장소의 낭비
 - 매번 번역하는 번거로움이 있음

시스템 소프트웨어

- Macro Processor
 - 매크로 호출을 매크로에서 정의된 원래 명령어로 대치
- Compiler
 - 고급언어 프로그램을 받아들이며 목적프로그램을 만듦
- Interpreter
 - 고급언어 프로그램의 각 문장을 기계어로 바꿈

GNU C Compiler



`gcc -E` →

`gcc -S` →

`gcc -c` →

`gcc` →

NASM

- Netwide Assembler
 - open source assembler that runs under Linux and DOS
 - debugging에 유리, 많은 기능을 가짐
 - 산업계에서도 많이 사용됨
 - <http://nasm.sourceforge.net/>
 - 설치와 사용은 실습에서 다룸

운영체제

- 운영체제

- 하드웨어를 관리하고, 응용프로그램에 실행 환경을 제공하는 시스템소프트웨어

- 운영체제의 기능

- Process management
 - Memory management
 - CPU scheduling
 - File & I/O subsystems
 - Disk scheduling
 - Utilities

Linux Operating System

- Unix Operating system
 - 하드웨어 제조사와 독립적인 최초 운영체제
 - Bell Labs에서 개발
 - AT&T version
 - BSD(Berkeley Software Divion) version
- Linux
 - non-proprietary version of Unix
 - created by Linus Torvals
 - mostly installed on Intel-based hardware