



CHUNGNAM NATIONAL UNIVERSITY



시스템 프로그래밍

강의 1.2부 컴퓨터시스템 개관

Sep. 2, 2014

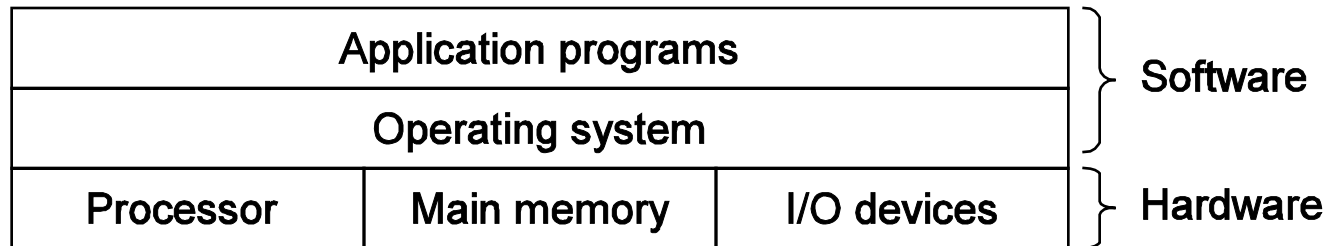
<http://eslab.cnu.ac.kr>



제 1장. 컴퓨터 시스템 개관

■ 컴퓨터 시스템의 본질을 찾아서...

컴퓨터 시스템의 계층도



컴퓨터 하드웨어의 구조

■ 하드웨어 구성요소

- 버스 buses
- 입출력 장치, I/O devices
- 주기억장치, main memory
- 캐쉬, cache memory
- 중앙처리장치, CPU (Central Processing Unit)

■ 버스

- 구성요소간의 정보교환 통로
- 종류 : 입출력버스, 시스템 버스

프로세서

■ 중앙처리장치 Central Processing Unit : CPU

- 산술논리연산장치 : arithmetic and logic unit, ALU
- 제어장치 : Control unit
- 레지스터
 - ▶ 프로그램 카운터 PC
 - ▶ 상태 레지스터
 - ▶ 메모리 주소 레지스터
 - ▶ 메모리 데이터 레지스터
 - ▶ 명령 레지스터, instruction pointer, IP
 - ▶ 범용 레지스터, general-purpose register

CPU

■ 주요기능

- 기억장치로부터 명령/데이터 호출 및 저장
- 명령의 해석 : 제어장치
- 명령의 실행 : ALU
- 입출력 연산 : I/O

■ 명령 주기 Instruction cycle

- 인출 fetch : PC에 의거하여 명령어를 가져옴
- 해독 decode : 명령어를 해석하여 제어 신호 생성
- 실행 execution : 가져온 명령어를 실행
- 결과저장 store : 실행 결과를 저장

운영체제 Operating System

■ 운영체제란?

- 응용프로그램이 하드웨어를 효율적으로 사용할 수 있도록 도와주는 프로그램

■ 목적

- 하드웨어 추상화 Hardware Abstraction
- 시스템 인터페이스 추상화

■ 운영체제의 관리대상

- 프로세스
- 메모리
- 입출력장치
- 소프트웨어 자원
 - ▶ 파일시스템, 라이브러리, 유틸리티

Hello world ?!!

■ 위대한 프로그램 hello.c

```
#include <stdio.h>

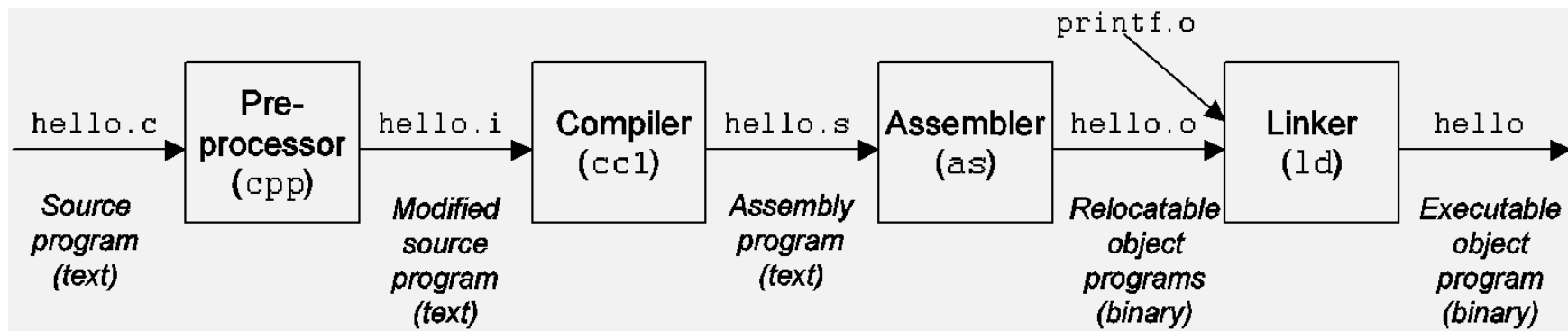
int main()
{
    printf("hello, world\ n");
}
```

■ 본질 ! 컴퓨터의 모든 정보는 비트, 바이트로 저장된다

프로그램은 변신한다

■ hello.c 컴파일 과정

● `unix> gcc -o hello hello.c`



■ **본질 ! 컴파일러의 뒷면에는 이렇게 복잡한 일이 일어나고 있다**

어셈블리 프로그램 ? 실행 목적 프로그램 ? 기계어 프로그램 ?

컴파일 과정을 이해하는 것은 매우 중요하다

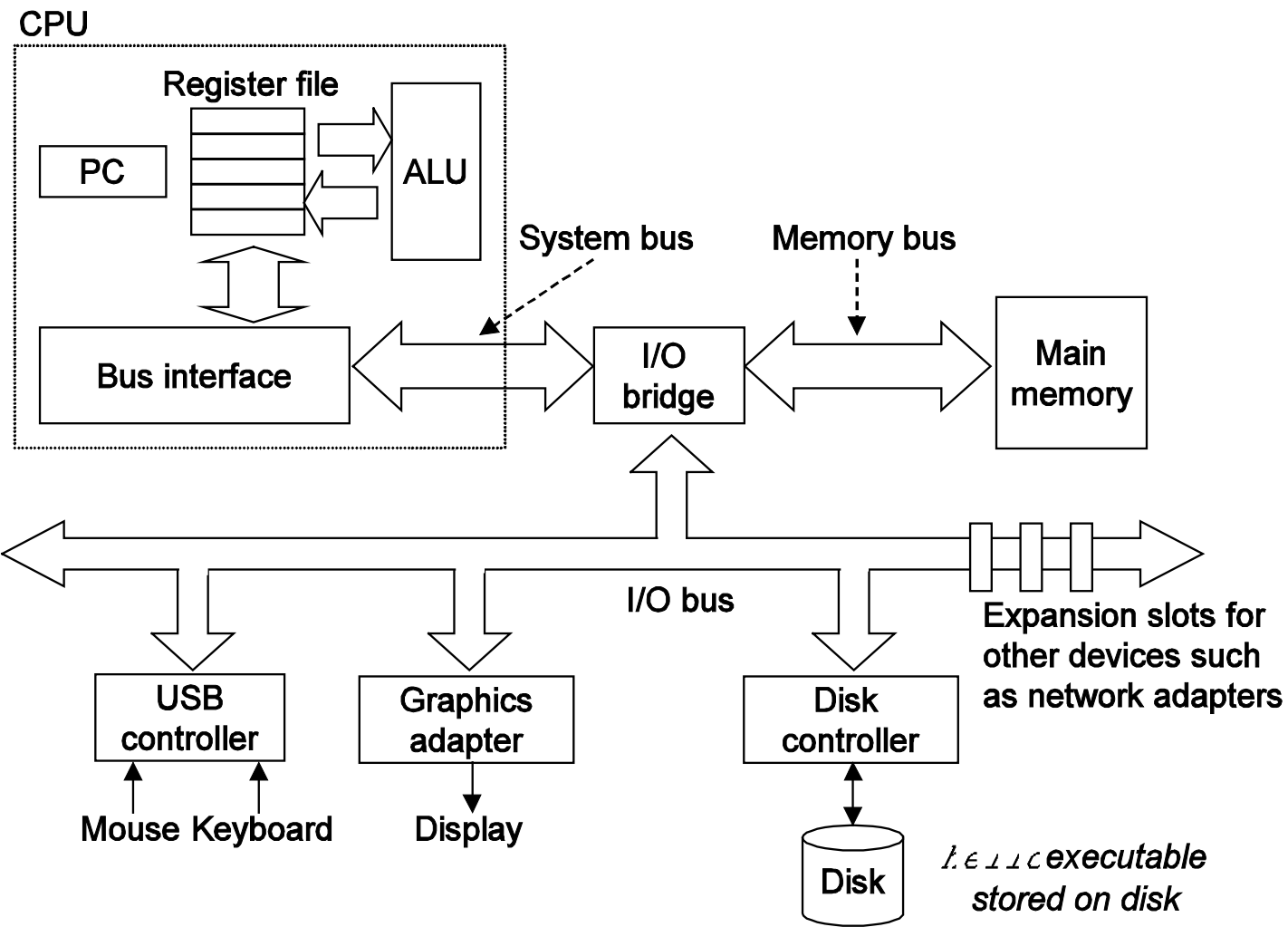
- 프로그램이 컴퓨터 내부에서 어떻게 처리되는지 이해할 수 있다
- 대규모 프로그램 개발 과정을 이해할 수 있다
- 프로그램 성능을 개선할 수 있다
- 링크시에 발생하는 에러를 이해할 수 있다

변환된 프로그램의 실행

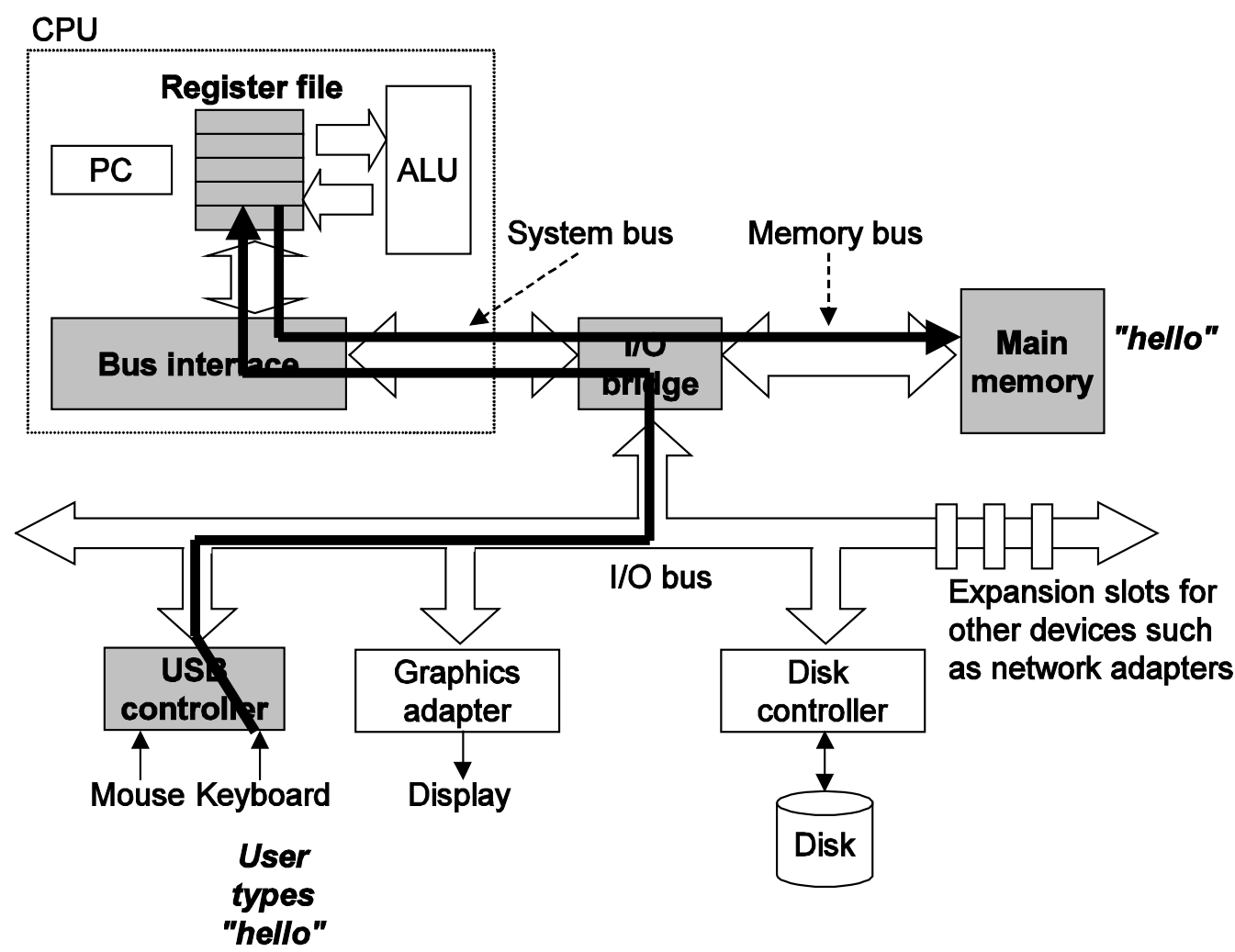
```
cesl-linux> ./hello  
hello, world  
cesl-linux>
```

- 리눅스 셸에서 실행파일 이름을 치면 실행됨
 - 셸이 여러분이 입력한 파일 이름을 찾아서, 그것이 실행파일이면 프로그램을 로드하여 실행함
 - **hello** 프로그램이 실행되면서 결과가 화면에 출력됨.
 - 셸이 다음 명령을 기다리고 있음
- 본질?

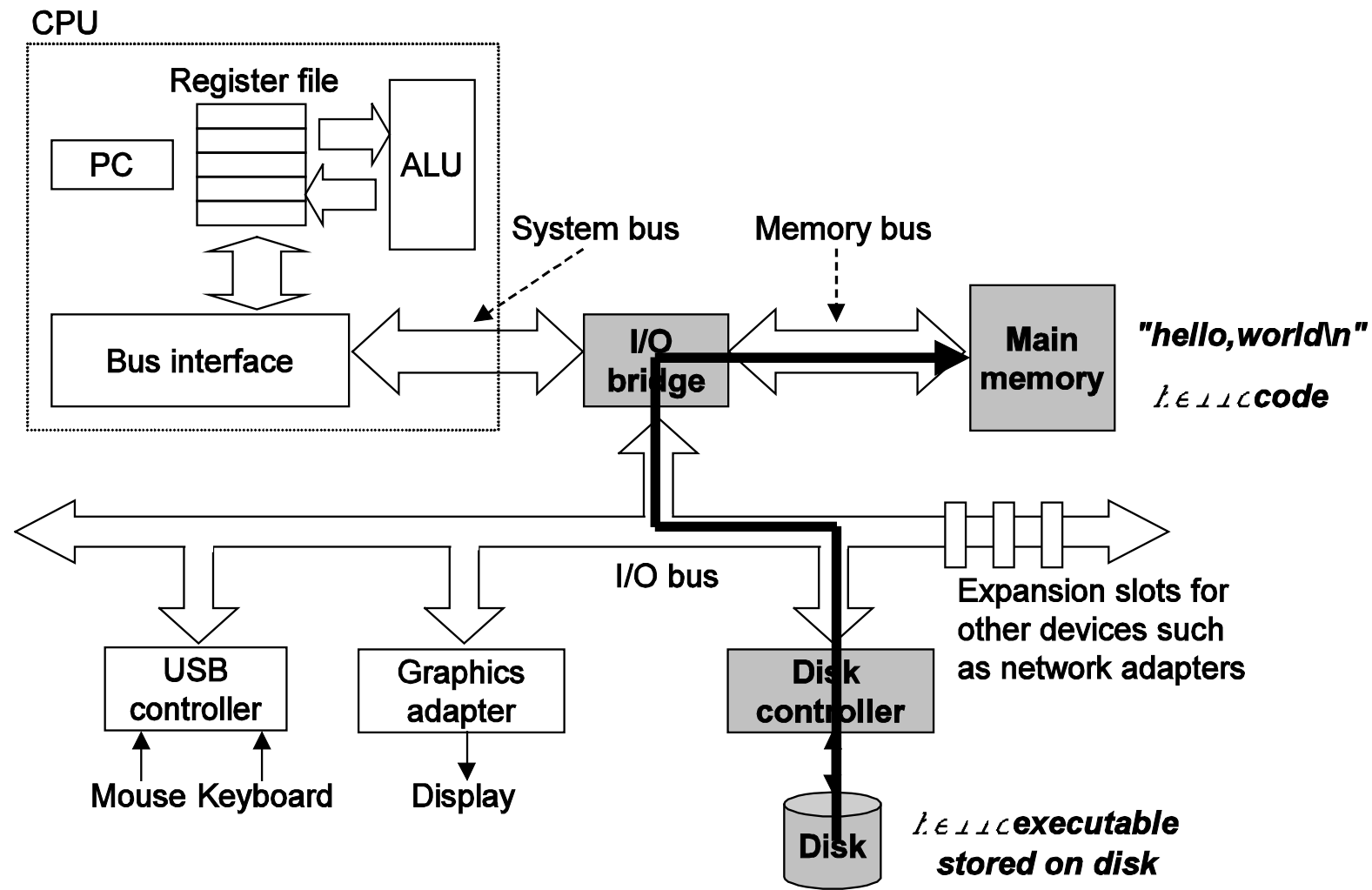
컴퓨터의 구조



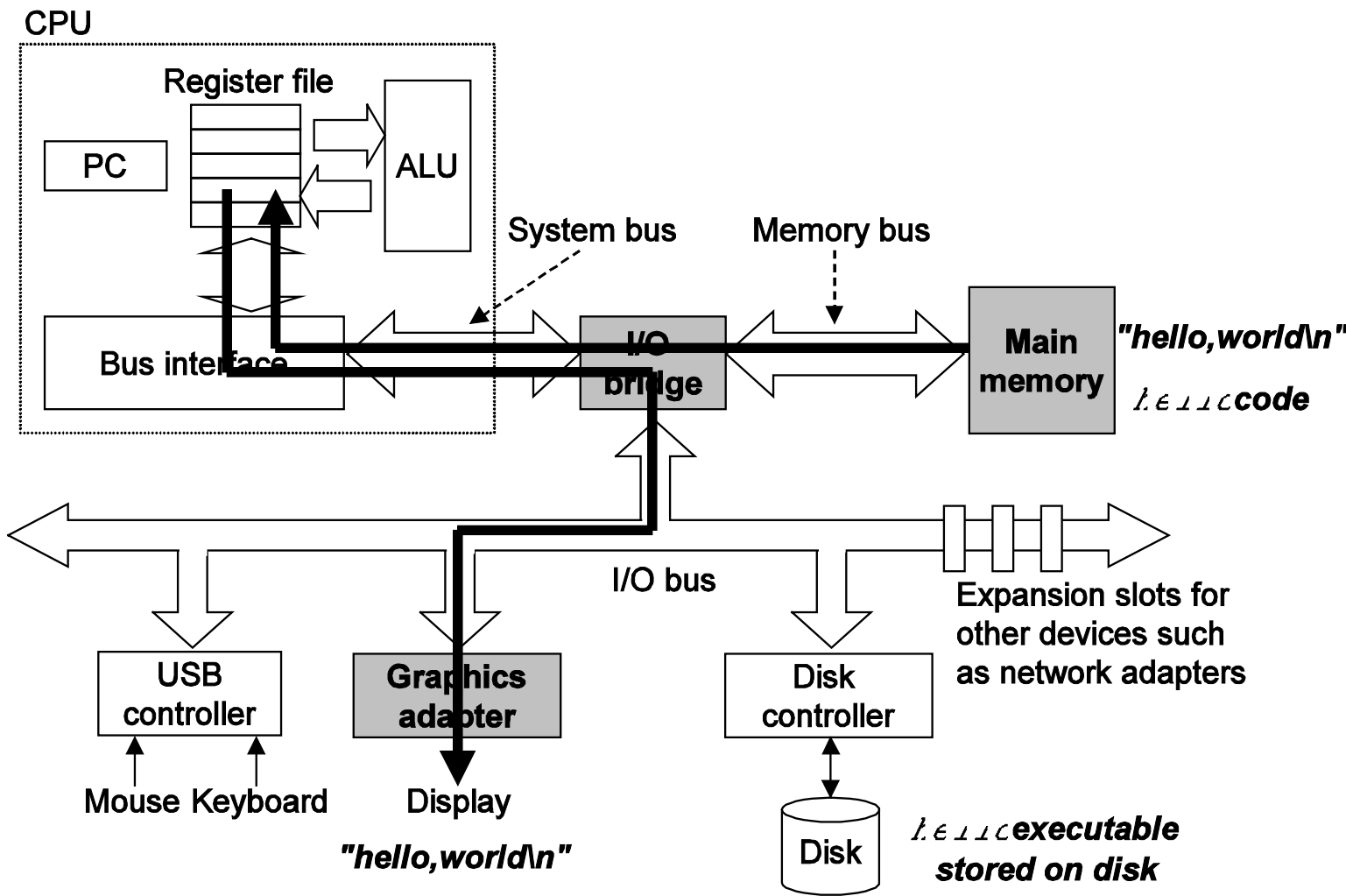
실행의 본질 : hello 명령의 인식



실행의 본질 : hello 프로그램의 로딩



실행의 본질 : hello 프로그램의 실행



요약

- 컴퓨터 시스템은 하드웨어, 시스템 소프트웨어, 응용프로그램으로 구성된다.
- 프로그램은 번역되어 변화한다.
- 프로그램은 시스템 프로그램의 도움으로 하드웨어에서 실행된다.

다음주를 위한 준비

■ 새 학기를 맞는 여러분의 결심

- 강의는 열심히, 생활은 즐겁게
- 전공으로 진입하는 중요한 기회의 학기
- 적절한 수준의 수강신청

■ 예습 숙제 1.

- 교재 2.1.4 Addressing and Byte Ordering(pp.73-80) 읽고 A4 1쪽 이내로 요약
- 다음주 수업시간에 제출

■ 이번주 실습 1

■ 홈페이지 가서 강의자료 내려받기.

C 프로그래밍 능력 진단 퀴즈

- 아래의 `call_swap()` 함수는 `zip1`과 `zip2`에 저장된 정수값을 포인터를 이용해서 위치를 바꿔서 저장해주는 함수 `swap()`를 호출한다. `Swap()` 함수를 구현하시오. `Swap` 함수 호출후에 `zip2=15213`, `zip1=91125`가 저장되어야 한다.

```
int zip1 = 15213;
int zip2 = 91125;

void call_swap()
{
    swap(&zip1, &zip2);
}
```

```
void swap(                )
{

}
}
```