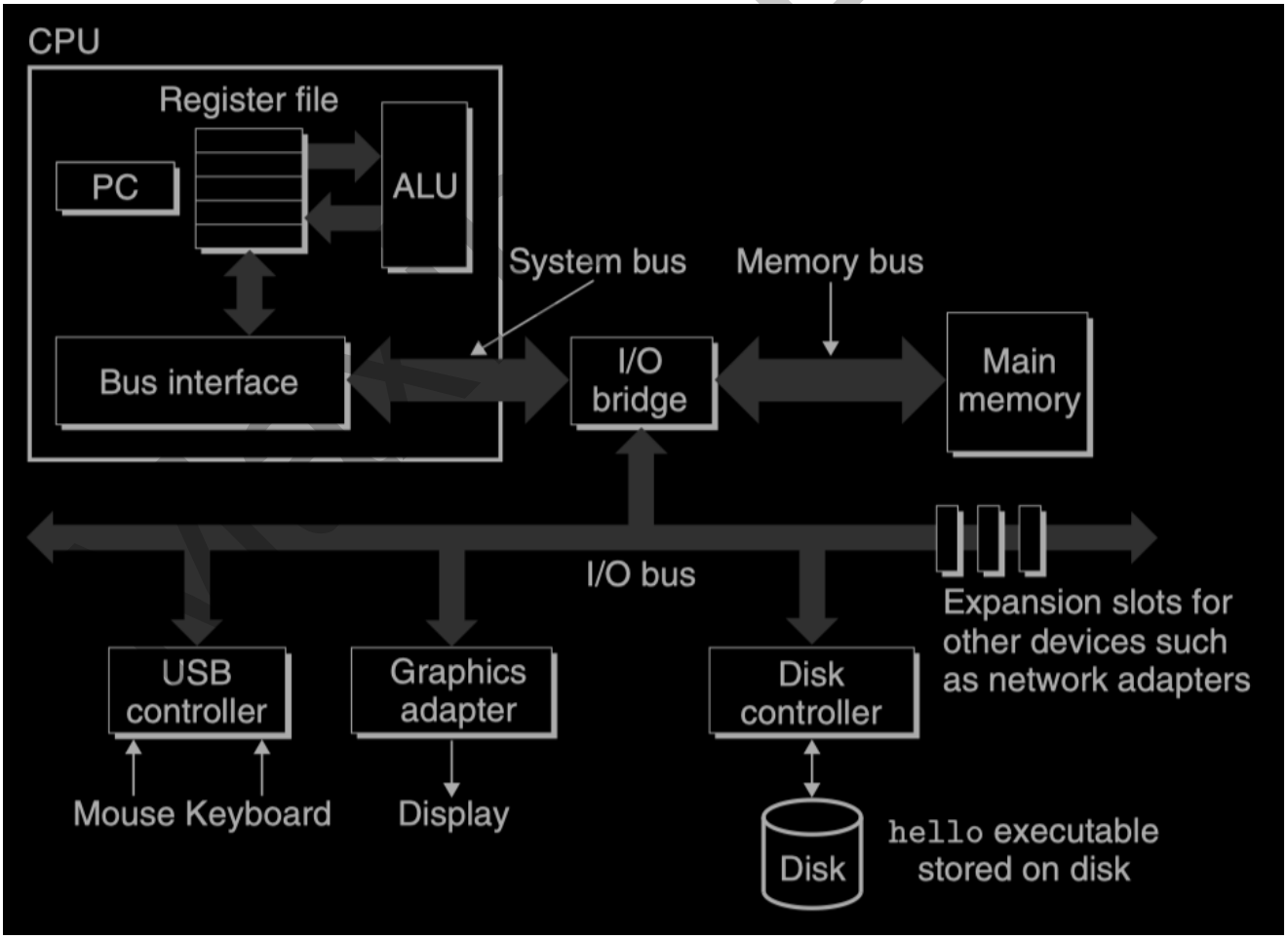
1_4
Chapter 1: A Tour of Computer System
1.4 Processors Read and Interpret Instructions Stored in Memory

- Run the executable object file:

```
1    ./hello
```

  - The shell is a command-line interpreter that prints a prompt(提示符), waits for you to type a command line, and then performs the command.
  - Input our command. If the first word of the command line does not correspond to a built-in shell command, then the shell assumes that it is the name of an executable file that it should load and run.
  - in this case, the shell loads and runs the hello program and then waits for it to terminate.
  - The hello program prints its message to the screen and then terminates.
  - The shell then prints a prompt and waits for the next input command line.

1.4.1 Hardware Organization of a System



- To understand what happens to our hello program when we run it, we need to understand the hardware organization of a typical system as above, including buses, main memory, processor and I/O devices.

Buses – 总线

- Running throughout the system is a collection of electrical conduits (电导管) called buses that **carry bytes of information back and forth between the components.**
- Buses are typically designed to transfer fixed-size chunks of bytes known as words.
- The number of bytes in a word (the word size) is a fundamental system parameter that varies across systems.
  - Most machines today have word sizes of either 4 bytes (32 bits) or 8 bytes (64 bits).

I/O Devices – 输入/输出设备

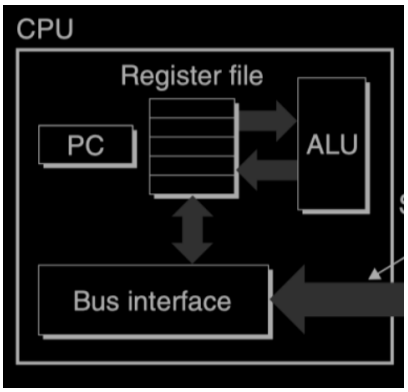
- Input/output (I/O) devices are the system's connection to the external world.
- Initially, the executable hello program resides on the disk.
- Each I/O device is connected to the I/O bus by either a controller(控制器) or an adapter(适配器).
  - Controllers are chip sets in the device itself or on the system's main printed circuit board (often called the motherboard).
  - An adapter is a card that plugs into a slot on the motherboard.
  - The purpose of each is to transfer information back and forth between the I/O bus and an I/O device.

Main Memory – 主内存

- The main memory is a temporary storage device that holds both a program and the data it manipulates while the processor is executing the program.
  - Physically, main memory consists of a collection of dynamic random access memory (DRAM – 动态随机存取存储器) chips.
  - Logically, memory is organized as a linear array of bytes, each with its own unique address (array index) starting at zero.

Processor – 处理器

- The central processing unit (CPU), or simply processor, is the engine that interprets (or executes) instructions stored in main memory.
- At its core is a word-size storage device (or register) called the program counter (PC – 程序计数器). At any point in time, the PC points at (contains the address of) some machine-language instruction in main memory.
  - From the time that power is applied to the system until the time that the power is shut off, **a processor repeatedly executes the instruction pointed at by the program counter and updates the program counter to point to the next instruction.**
  - A processor appears to operate according to a very simple instruction execution model, defined by its instruction set architecture (指令集).
- Processor work flow regarding program counter (PC):
  1. The processor reads the instruction from memory pointed at by the program counter (PC).
  2. The processor interprets the bits in the instruction.
  3. The processor performs some simple operation dictated by the instruction.
  4. The processor updates the PC to point to the next instruction, which may or may not be contiguous in memory to the instruction that was just executed.
- Other Components:

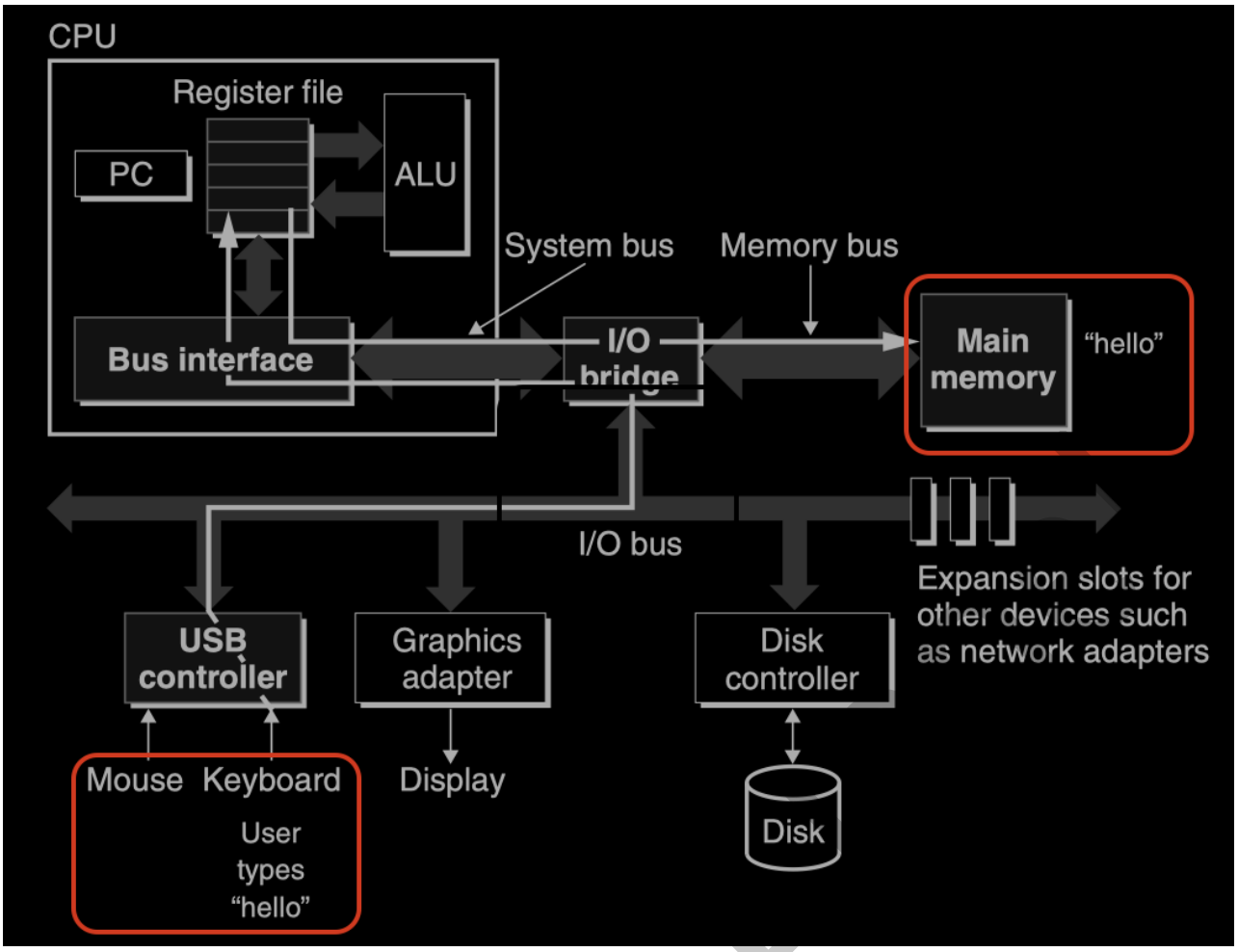


  - register file (寄存器文件) - a small storage device that consists of a collection of word-size registers, each with its own unique name.
  - Arithmetic/logic unit (ALU - 算术逻辑单元) - computes new data and address values.
- 4 Operations that the CPU might carry out at the request of an instruction:
  - Load (加载) - Copy a byte or a word from main memory into a register, overwriting the previous contents of the register.
  - Store (存储) - Copy a byte or a word from a register to a location in main memory, overwriting the previous contents of that location.
  - Operate (操作) -
    1. Copy the contents of two registers to the ALU.
    2. Perform an arithmetic operation on the two words.
    3. Store the result in a register, overwriting the previous contents of that register.
  - Jump (转移) - Extract a word from the instruction itself and copy that word into the program counter (PC), overwriting the previous value of the PC.

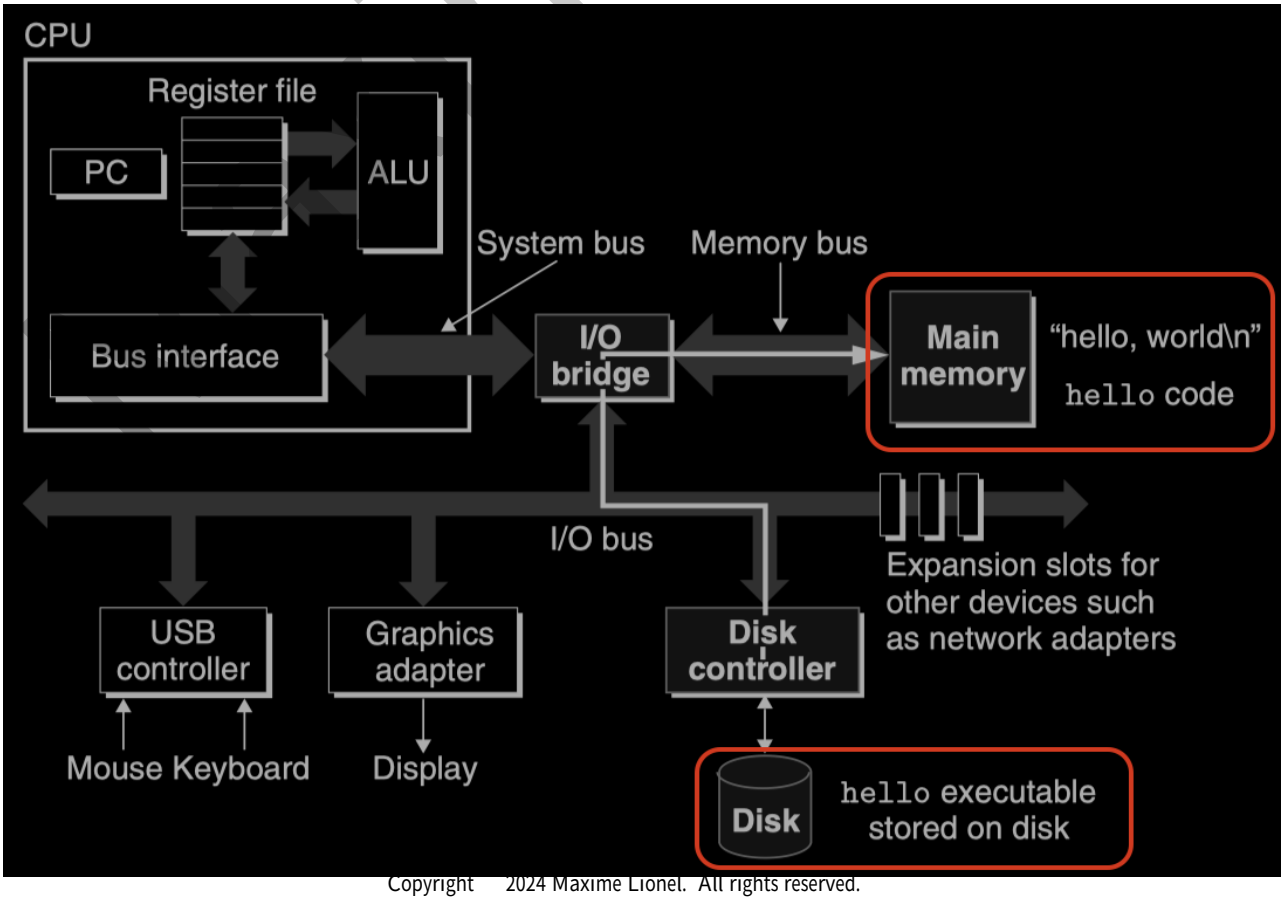
1.4.2 Running the hello Program

- Initially, the shell program is executing its instructions, waiting for us to type a command. As we type the characters ./hello at the keyboard, the shell program reads each one into a register and then stores it in memory.

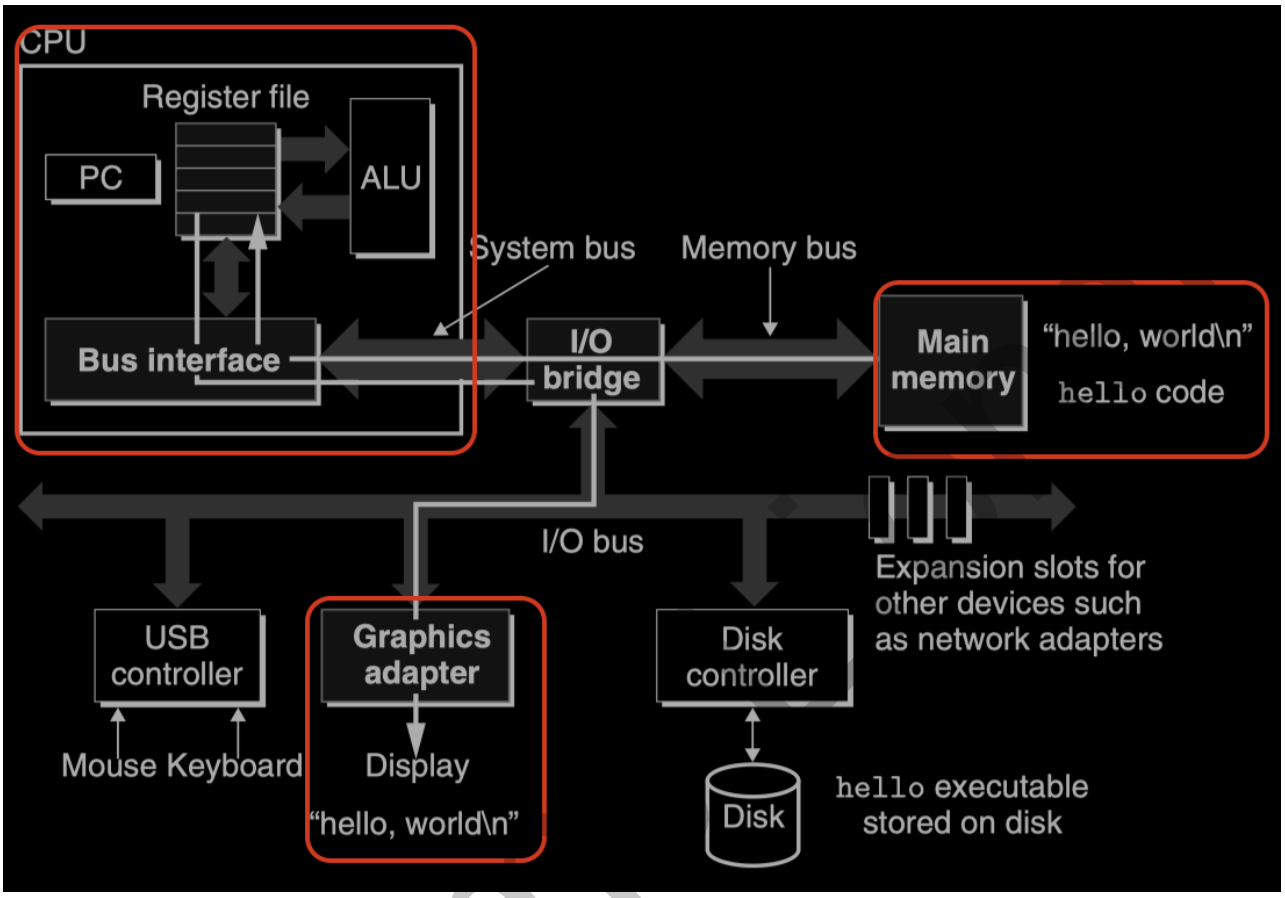
```
maximelionel@Maximes-MBP ~ % ./hello
```



- When we hit the enter key on the keyboard, the shell knows that we have finished typing the command. The shell then loads the executable hello file by executing a sequence of instructions that copies the code and data in the hello object file **from disk to main memory.** The data includes the string of characters 'hello, world\n' that will eventually be printed out.



    - Using a technique known as direct memory access (DMA), the data travel directly from disk to main memory, without passing through the processor.
- Once the code and data in the hello object file are loaded into memory, the processor begins executing the machine-language instructions in the hello program's main routine. These instructions copy the bytes in the hello, world\n string from memory to the register file, and from there to the display device, where they are displayed on the screen.




```
parallels@ubuntu-linux-22-04-desktop:~/csapp/hello$ gcc hello.c -o hello
parallels@ubuntu-linux-22-04-desktop:~/csapp/hello$ ./hello
Hello World!
```