SSD
6

icarnegie

curriculum powered by Carnegie Mellon

Lecture 2

A Tour of Computer Systems

武漢大學

国际软件学院

Lecture 2 A Tour of Computer Systems

班级	助教姓名	电话	邮箱
1～3班	陈硕	15827250026	1575335819@qq.com
	徐乔博	15527814980	714195402@qq.com
4～6班	梁巍巍	15527608252	1242005230@qq.com
	汪俊	15623247505	1914731404@qq.com
国际班	何施灏	15972217477	936196652@qq.com

Lecture 2 A Tour of Computer Systems

- What's information
- Program in various forms
- How compilation systems work
- Processors
- Caches matter
- Hierarchy storage devices
- OS vs. Hardware
- Communication between systems

Lecture 2 What's information

- Information is Bits + Context

```
#include <stdio.h>

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

Lecture 2 What's information

? Information is Bits + Context

#	i	n	c	l	u	d	e	<sp>	<	s	t	d	i	o	.
35	105	110	99	108	117	100	101	32	60	115	116	100	105	111	46
h	>	\n	\n	i	n	t	<sp>	m	a	i	n	(	)	\n	{
104	62	10	10	105	110	1[illegible]	3[illegible]	109	97	105	110	40	41	10	123
\n	<sp>	<sp>	<sp>	<sp>	p	r[illegible]	[illegible]	[illegible]	[illegible]	f	(	"	h	e	l
10	32	32	32	32	112	1[illegible]4	[illegible]	[illegible]	16	102	40	34	104	101	108
l	o	,	<sp>	w	o	r	l	d	\	n	"	)	;	\n	}
108	111	44	32	119	111	114	108	100	92	110	34	41	59	10	125

Bits

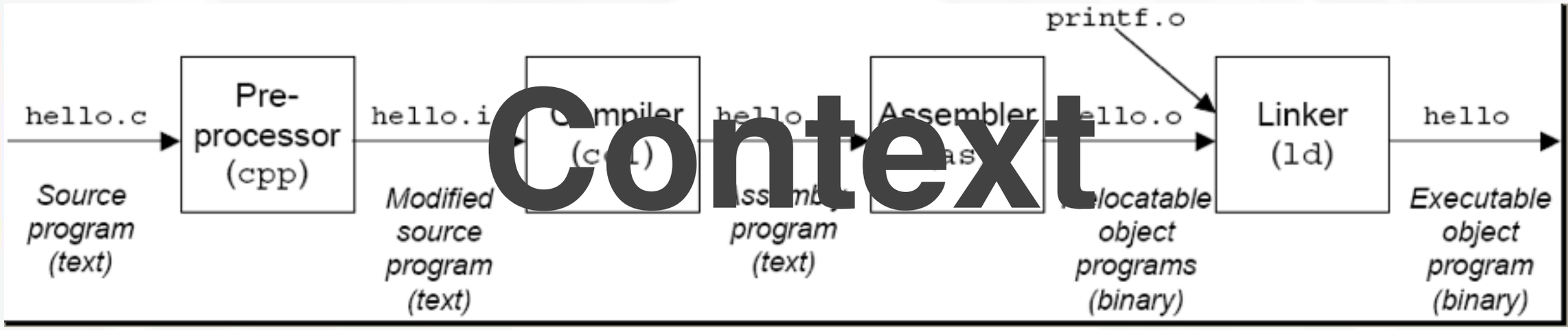
Figure 1.2 **The ASCII text representation of `hello.c`**

```
#include <stdio.h>

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

Lecture 2 Program in various forms

unix> gcc –o hello hello.c



Context

```
#include <stdio.h>

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

Lecture 2 Program in various forms

```
unix> gcc –E hello.c -o hello.i
```

Stop after the preprocessing stage; do not run the compiler proper.

```
unix> gcc –S hello.i -o hello.s
```

Stop after the stage of compilation proper; do not assemble.

```
unix> gcc –c hello.s -o hello.o
```

Compile or assemble the source files, but do not link.

```
unix> gcc hello.o -o hello
```

Compile or assemble the source files, one-shot.

Lecture 2 Program in various forms

Codes in various system

```
int sum (int x, int y)
{
    return x + y;
}
```

Lecture 2 Program in various forms

Codes in various system

```
int sum (int x, int y)
{
    return x + y;
}
```

Linux	55 89 e5 8b 45 0c 03 45 08 89 ec 5d c3
NT	55 89 e5 8b 45 0c 03 45 08 89 ec 5d c3
SUN	81 C3 E0 08 90 02 00 09
Alpha	00 00 30 42 01 80 FA 6B

Lecture 2 How compilation systems work

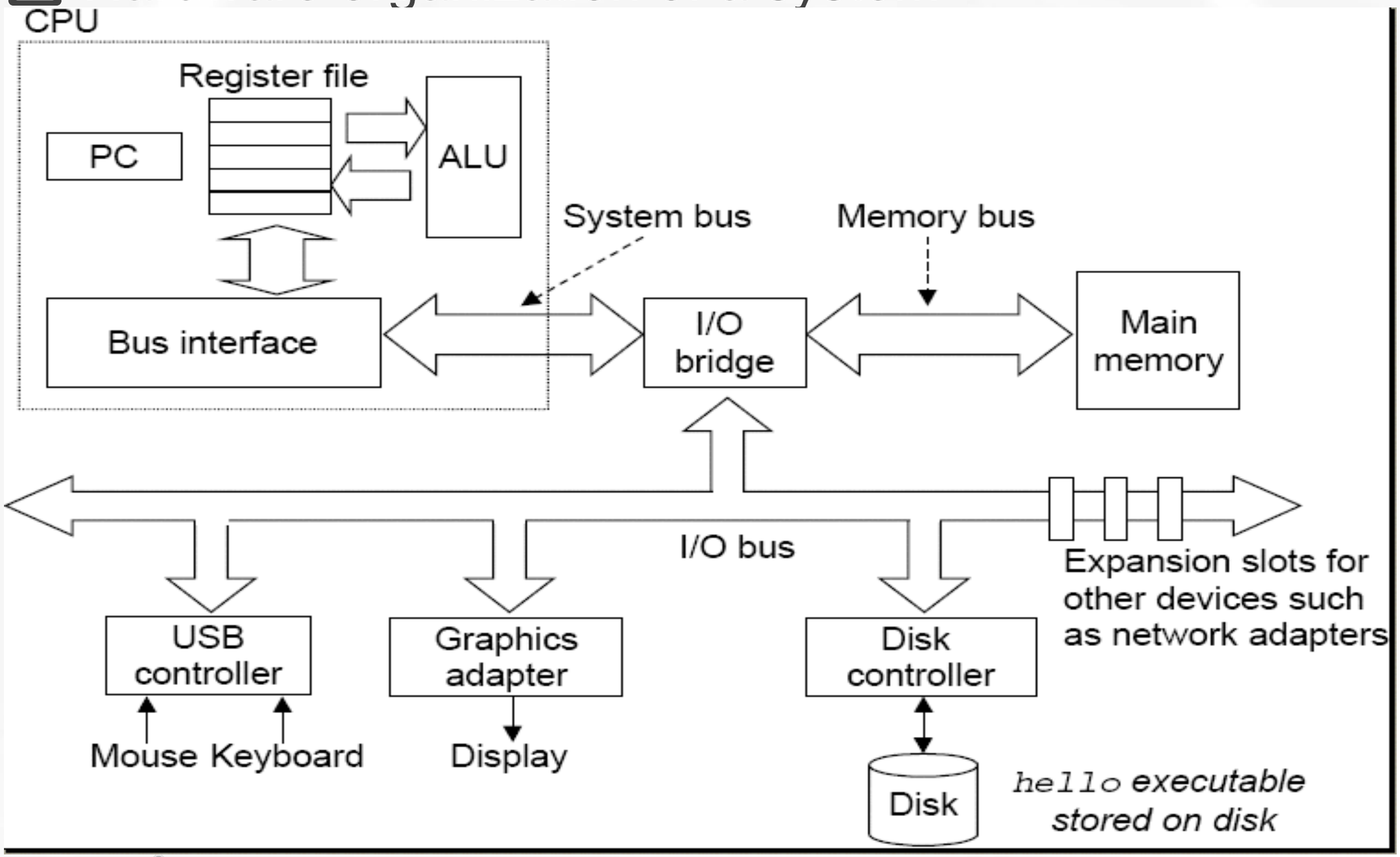
- Optimizing program performance
- Understanding link-time errors
- Avoiding security holes

Lecture 2 Processors read & interpret instructions

- Hardware organization of a system
  - 8086: (1978, 29K transistors)
  - 80286: (1978, 134K transistors)
  - i386: (1985, 275K transistors)
  - i486: (1989, 1.9M transistors)
  - Pentium: (1993, 3.1M transistors)
  - Pentium Pro: (1995, 6.5M transistors)
  - Pentium/MMX: (1997, 4.5M transistors)
  - Pentium II: (1997, 7M transistors)
  - Pentium III: (1999, 8.2M transistors)
  - Pentium IV: (2001, 42M transistors)

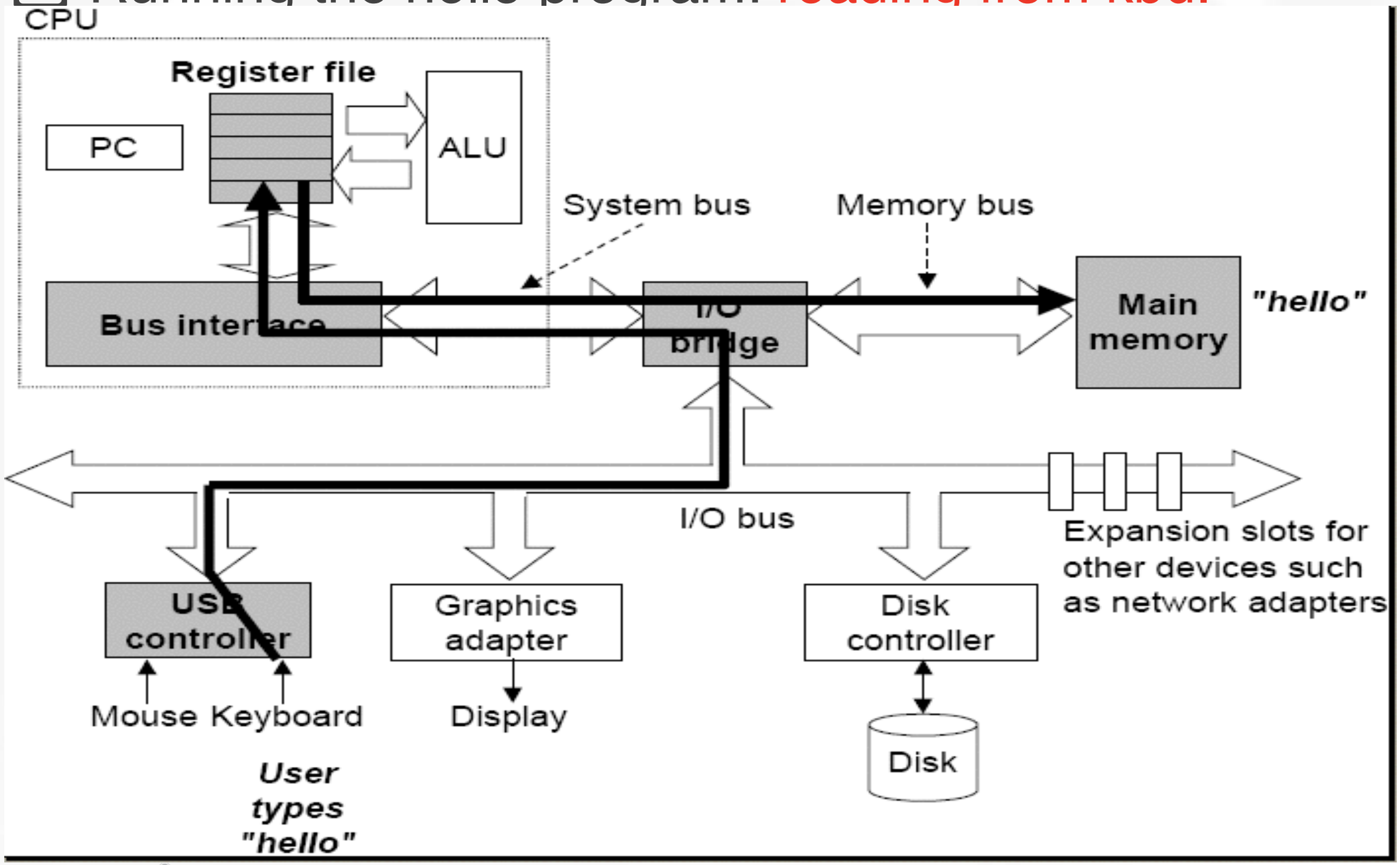
Lecture 2 Processors read & interpret instructions

? Hardware organization of a system



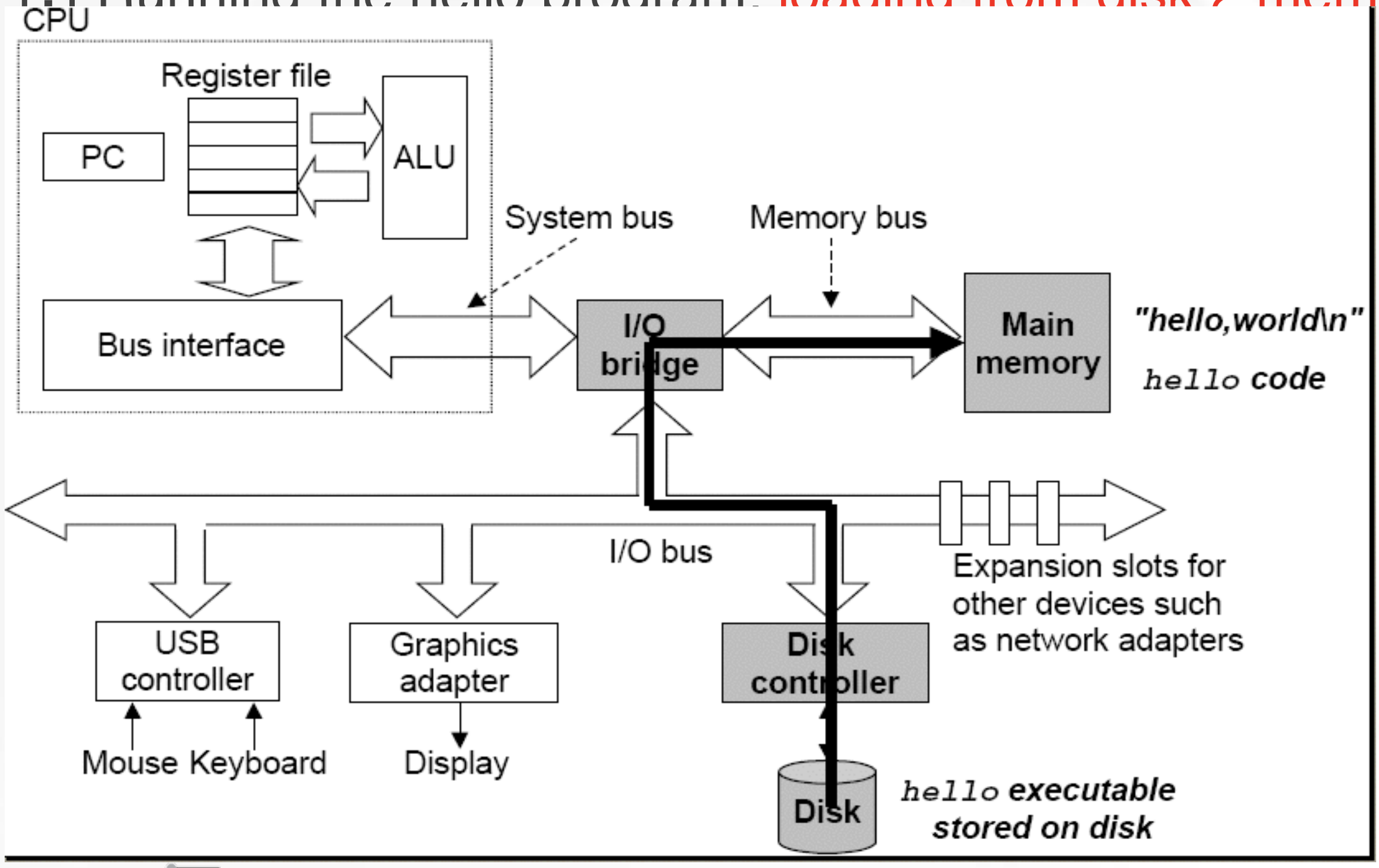
Lecture 2 Processors read & interpret instructions

? Running the hello program: reading from kbd.



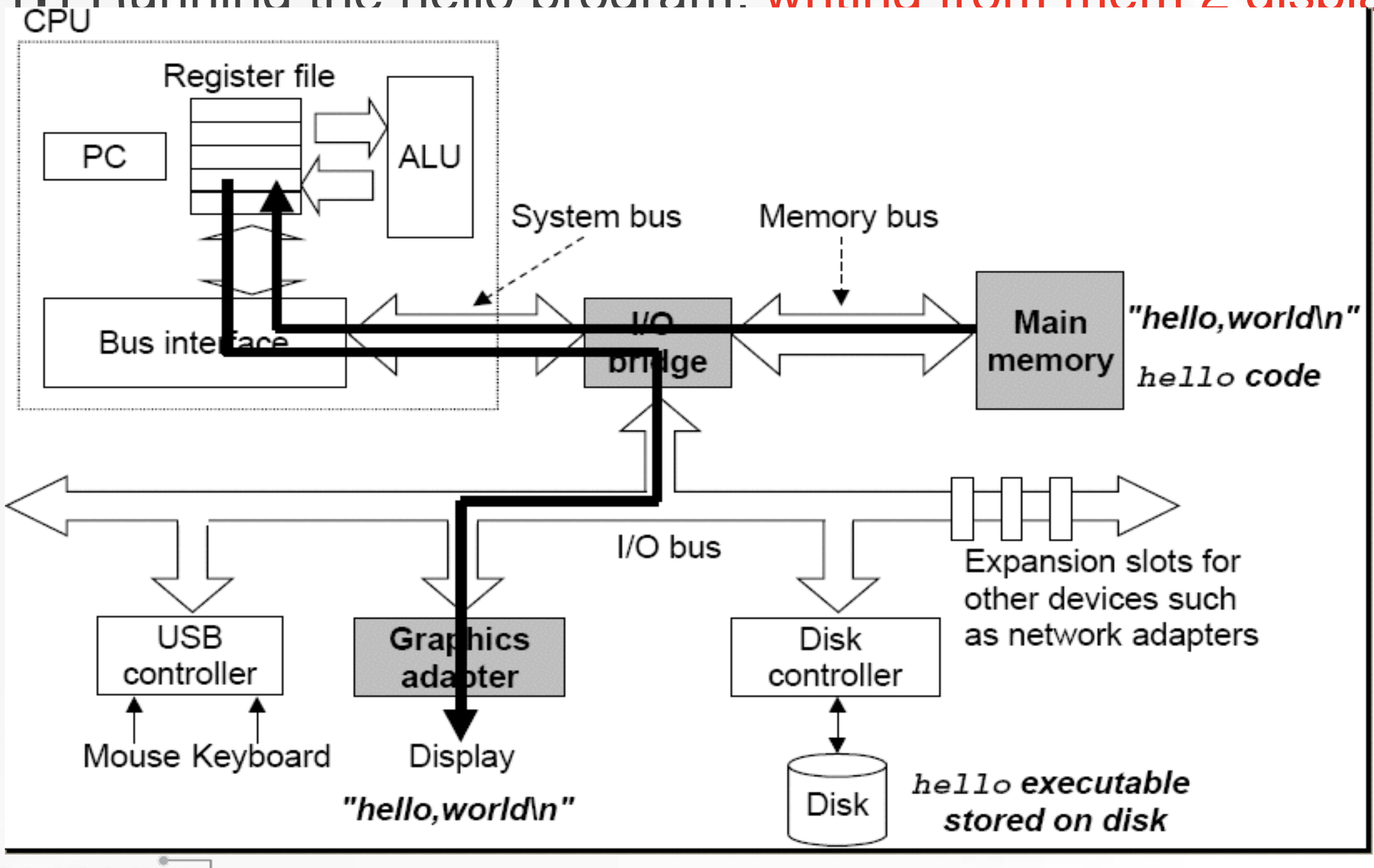
Lecture 2 Processors read & interpret instructions

? Running the hello program: loading from disk 2 mem.



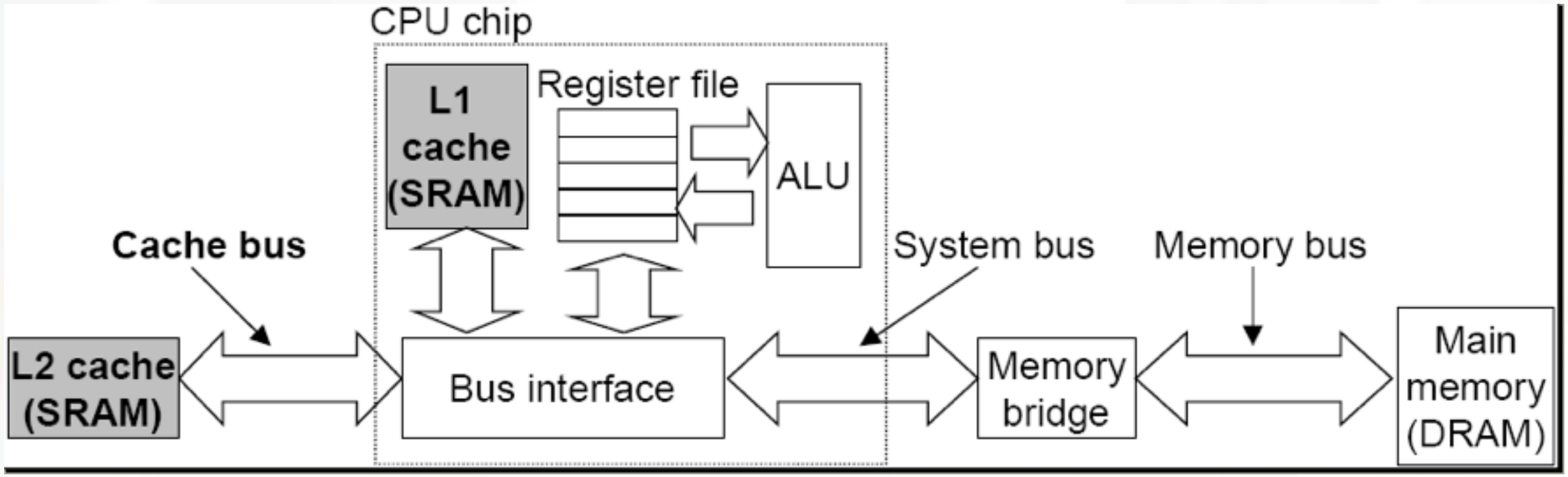
Lecture 2 Processors read & interpret instructions

? Running the hello program: writing from mem 2 display.

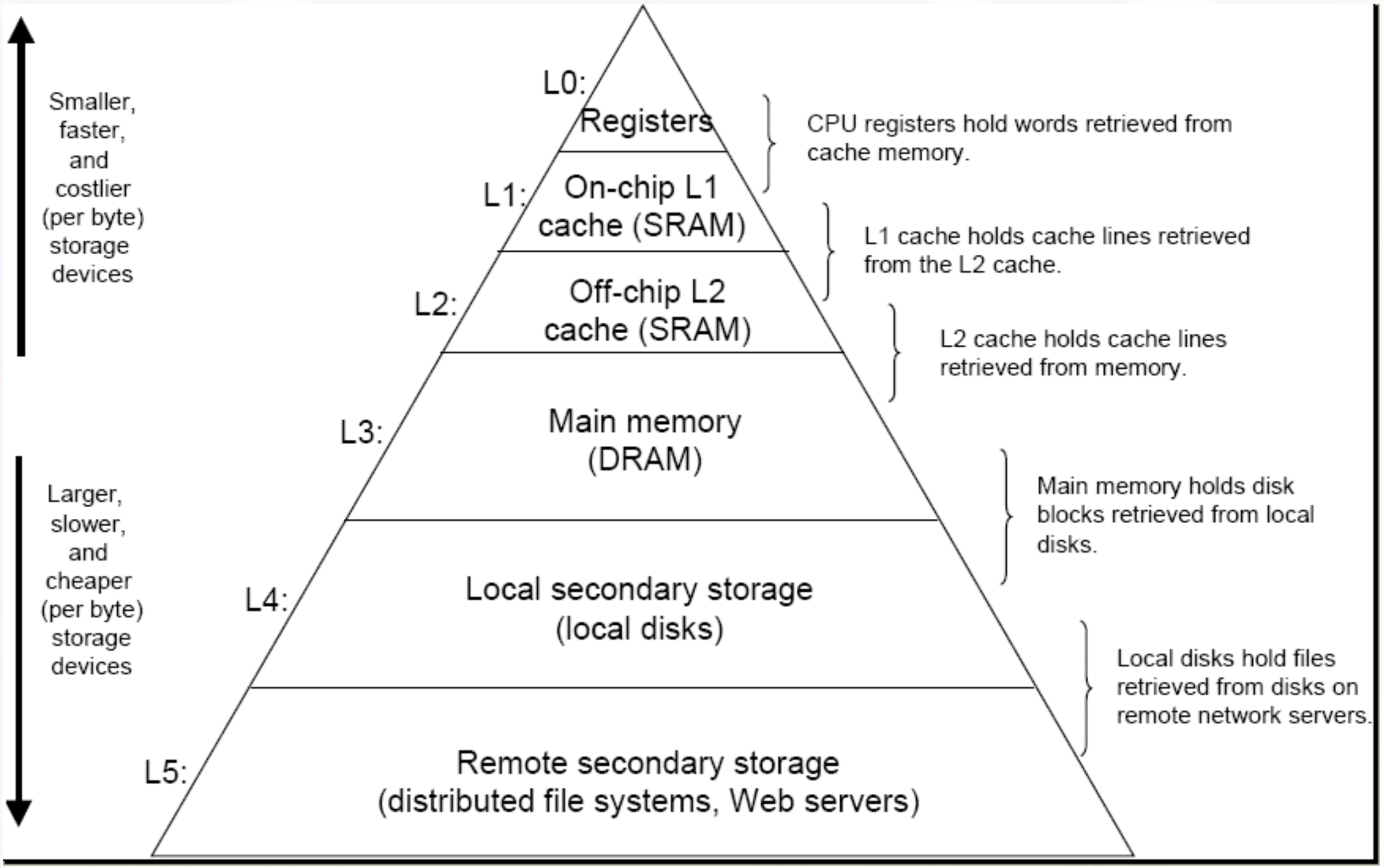


Lecture 2 Caches matter

- Memory size
- Cost
- Speed

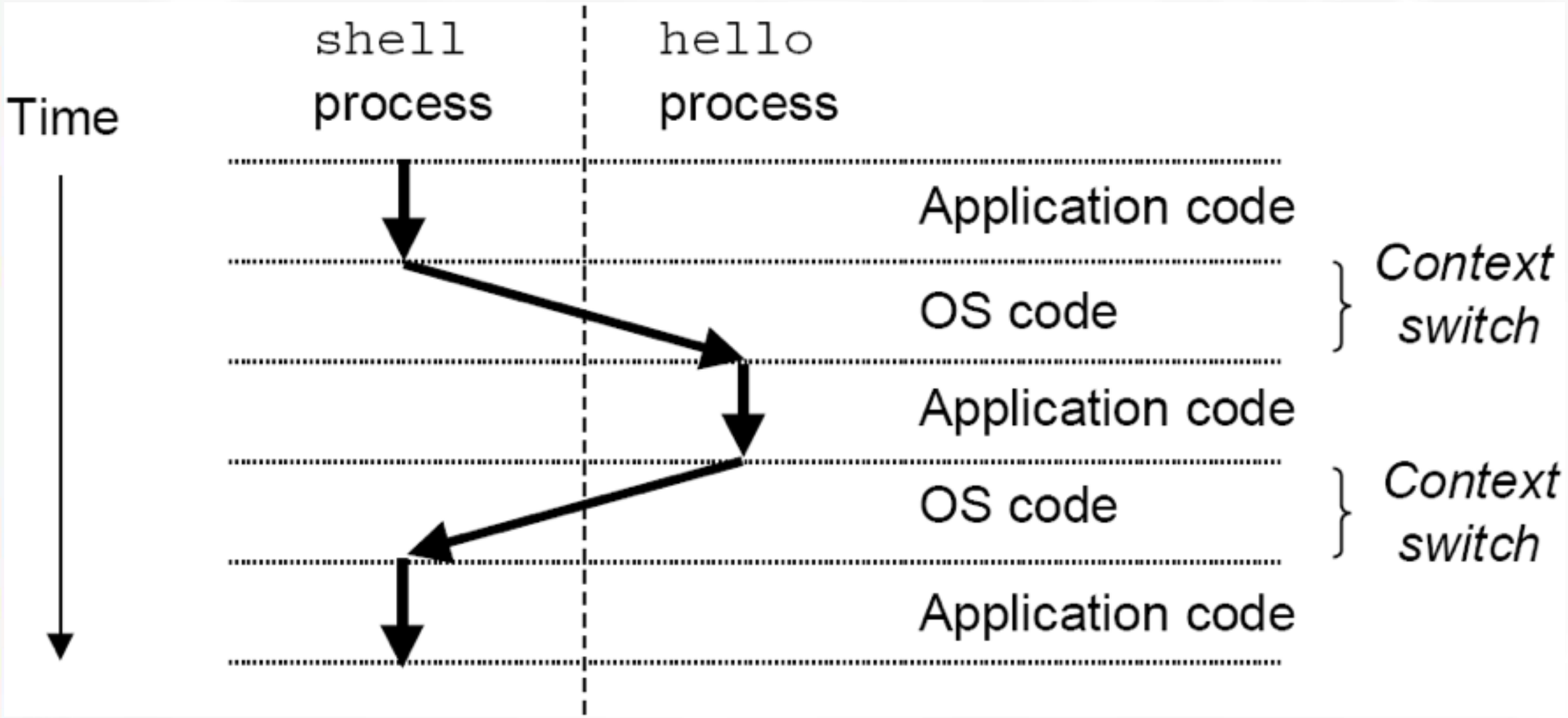


Lecture 2 Hierarchy storage devices



Lecture 2

OS vs. Hardware



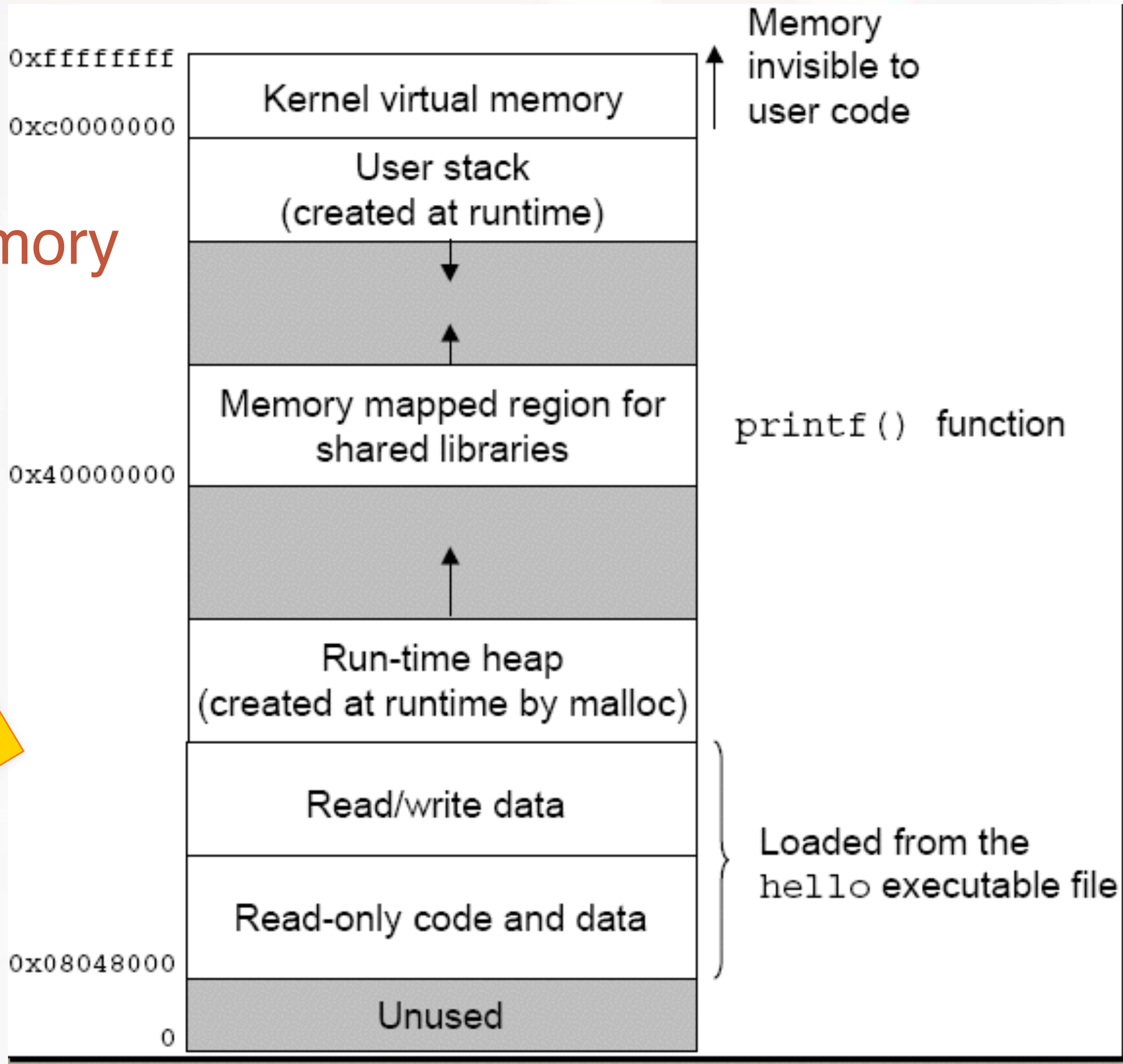
Lecture 2 OS vs. Hardware

- Processes
- Threads
- Virtual memory
- Files

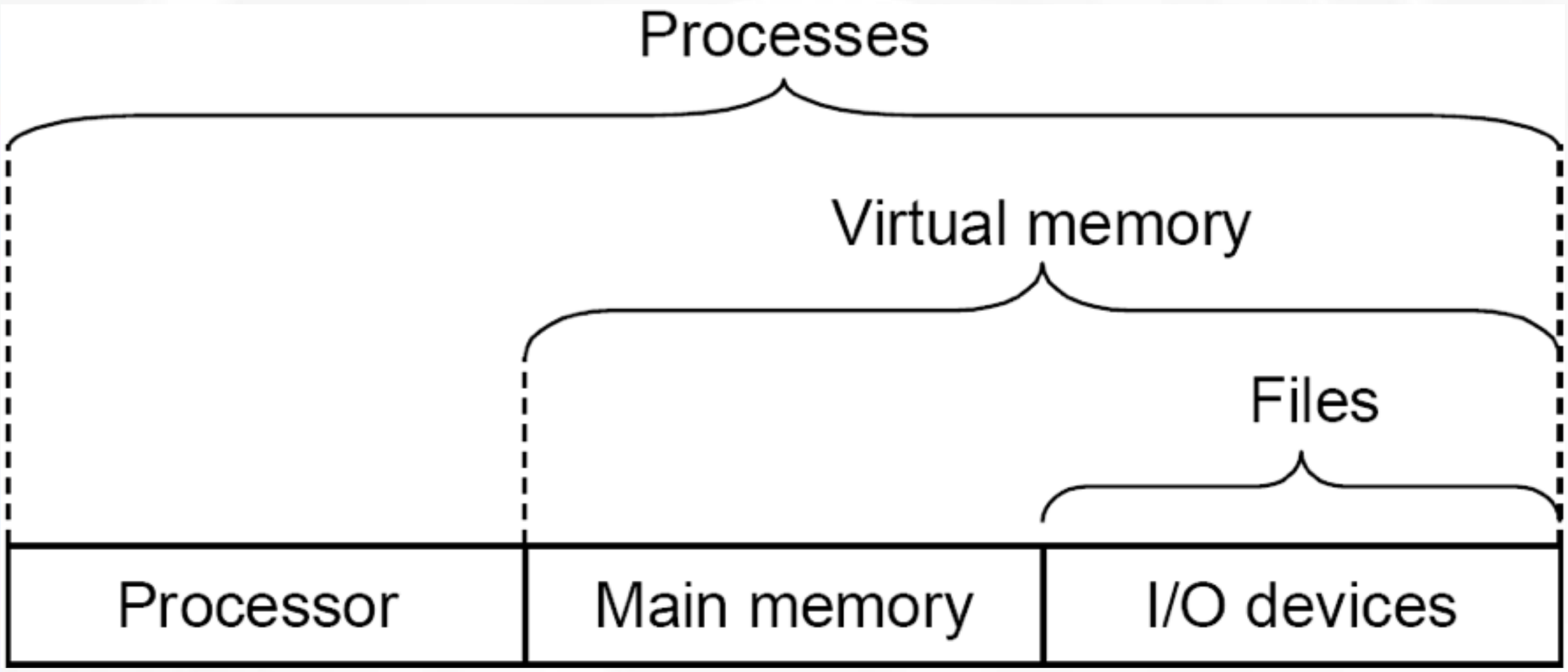
LINUX

icarnegie

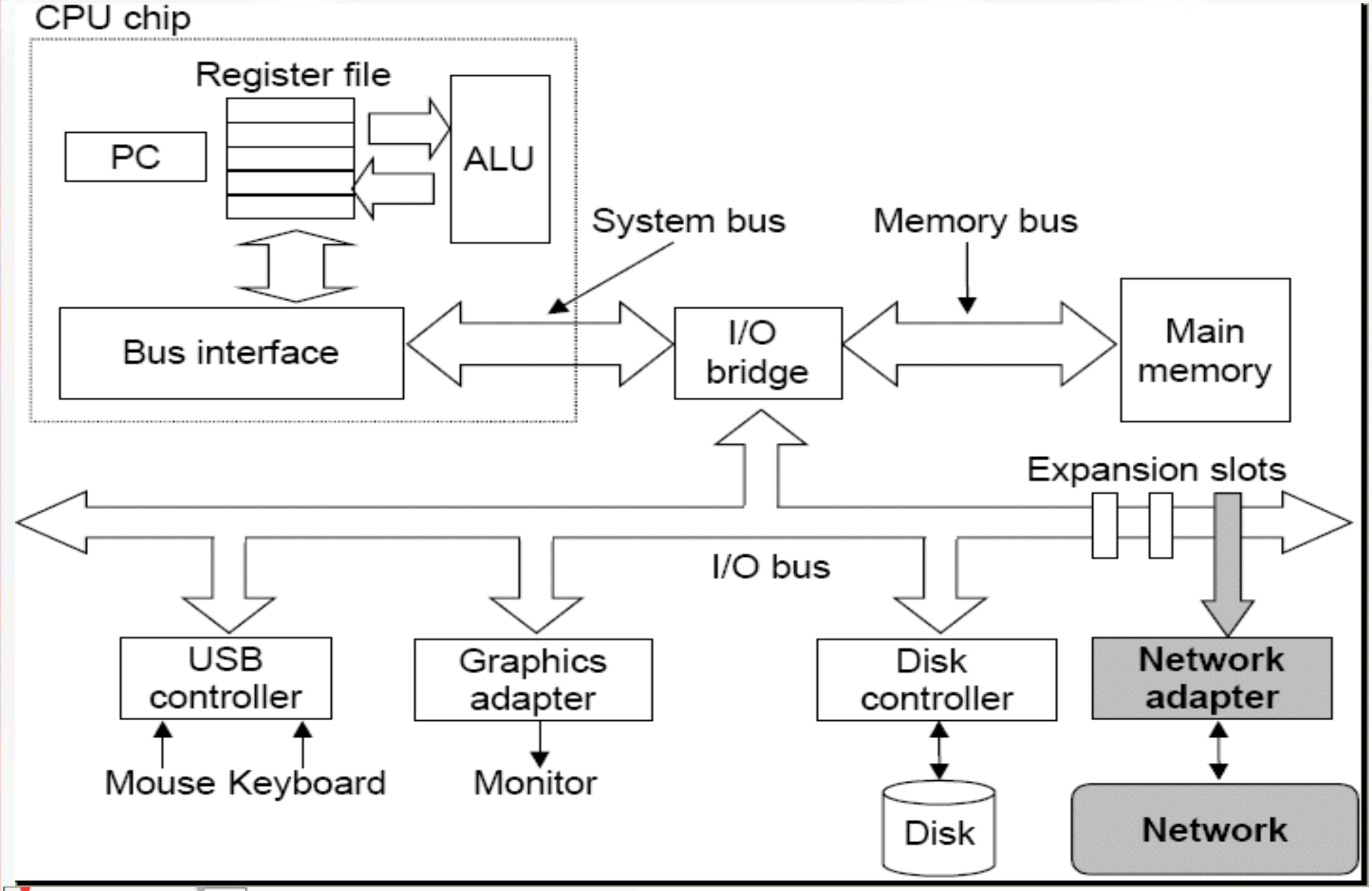
curriculum powered by Carnegie Mellon



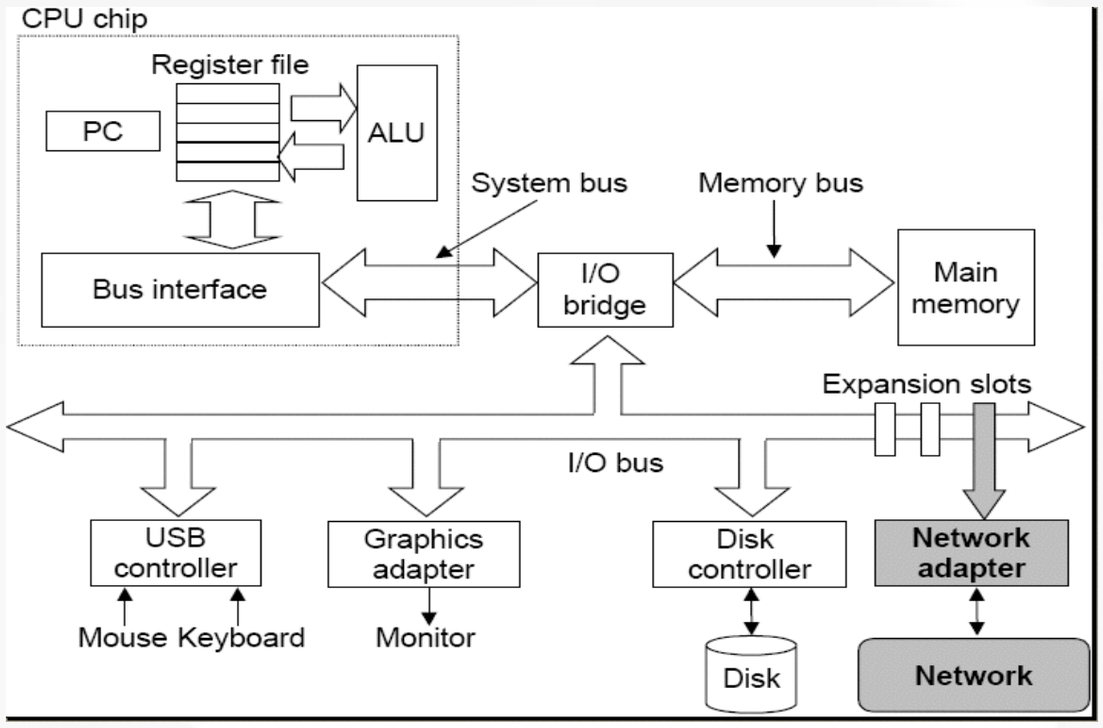
Lecture 2 OS vs. Hardware

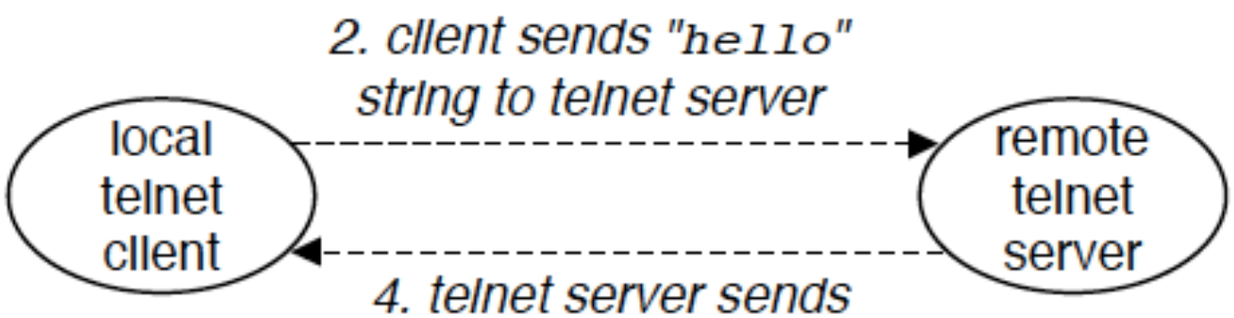


Lecture 2 Communication between systems



Lecture 2 Communication between systems





Lecture 2

Next Time……

Read:

- **Unit 1. C Programming Model**
  - 1.1 The Wonder of Program Execution
  - 1.2 The Visual C++ Debugger
  - 1.3 Variables and Addresses

Do:

- **Multiple-Choice Quiz 1**
- **Multiple-Choice Quiz 2**