

In Lecture 1 ...

- Program: Description of algorithms and data
- Data differs in
 - Basic vs Structured
 - Constant vs Variable
 - Type, e.g., unsigned int vs float vs char ...
 - Lifetime (Static vs Stack vs Heap allocated)
- Representation of data (as groups of bits)
 - Unsigned integer: Binary
 - Signed integer: 2s complement representation

Course Work (MJT)

- Quiz (in class 16/8, 25/8) 10 marks
- Homeworks
- (due 18/8, 23/8, 2/9) 10 marks
- Final exam 5 marks

Real Data

IEEE Floating Point Standard (IEEE 754)

32 bit value with 3 components (s , e , f)

1. s (1 bit sign)
2. e (8 bit exponent)
 - 00000000 - 11111111 (i.e., 0 - 255)
 - Exponent value = $e - 127$ “excess 127 notation”
 - Range of exponent values: -126 to 127
3. f (23 bit fraction)

represents the value

$$(-1)^s \times 1.f \times 2^{e-127}$$

Example: IEEE float

- The real value -23.5_{10} is represented with s: 1, f: 01111000..0, e: 131

-23.5_{10}

-10111.1_2

-1.01111×2^4 when normalized

1 10000011 01111000...00

0xC1BC0000

Special cases (B&O 2.4.2)

- $e=0$: denormalized form

$$(-1)^s \times 0.f \times 2^{-126}$$

- $e=255$

if $f = 0$: +infinity and -infinity

else: Not a Number (NaN)

Rounding (B&O 2.4.4)

- The standard supports 4 rounding modes
 - 1 Round to even
 - 2 Round towards 0
 - 3 Round down (towards $-\infty$)
 - 4 Round up (towards $+\infty$)

Assignment 1 (due 18/8)

- 1 **Calculate** how many times this C loop will iterate.

```
float f=1.0;
```

```
while (f != 0.0) f = f / 2.0;
```

- 2 **Prove** that the C shift operator `<<` (`>>`) can be used to multiply (divide) an unsigned int by powers of 2. Does this idea also work for signed ints? If not, show a counterexample.

Reading

- Read sections in Bryant and O'Hallaron on the topics we have discussed in class
 - Bryant, O'Hallaron. Computer Systems – A Programmer's Perspective, Pearson Education Limited 2016, 3rd Global Edition
- Try to solve some of the problems

Basic Computer Organization

- Main parts of a computer system:
 - Processor: Executes machine instructions
 - Main memory: Holds program and data
 - I/O devices: For communication with outside
- Machine instruction: Primitive operation that processor hardware is able to execute
- Instruction Set: Complete specification of all the kinds of instructions that the processor hardware was built to execute
- CPU registers: Small pieces of memory inside the CPU for temporary storage of data

Kinds of Instructions

1. Arithmetic/logical instructions

- ❑ Add, subtract, multiply, divide, compare (int/fp)
- ❑ Or, and, not, xor
- ❑ Shift (left/right, arithmetic/logical), rotate

2. Data transfer instructions

- ❑ Load (to a CPU register from main memory)
- ❑ Store (to main memory from a CPU register)
- ❑ Move

3. Control transfer instructions

- ❑ Jump, conditional branch, function call, return

4. Other instructions

- ❑ Example: halt

Bitwise and Shift Operators in C

- Bitwise operators $\&$ $|$ \wedge

Bitwise AND, OR, Exclusive-OR

Example: 1010 $\&$ 1100 equals 1000

Truth tables

A	B	A & B	A B	A ^ B
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

- Example:

```
int x, y; ...
```

```
y = x & 1;
```

Bitwise and Shift Operators in C

- Shift operators << >>

<< >> : Shift left, Shift right

Example: 0011 << 2 equals 1100

Observe: Multiplies by power of 2

- Example

```
unsigned int X, Y;
```

```
Y = X << 3;
```

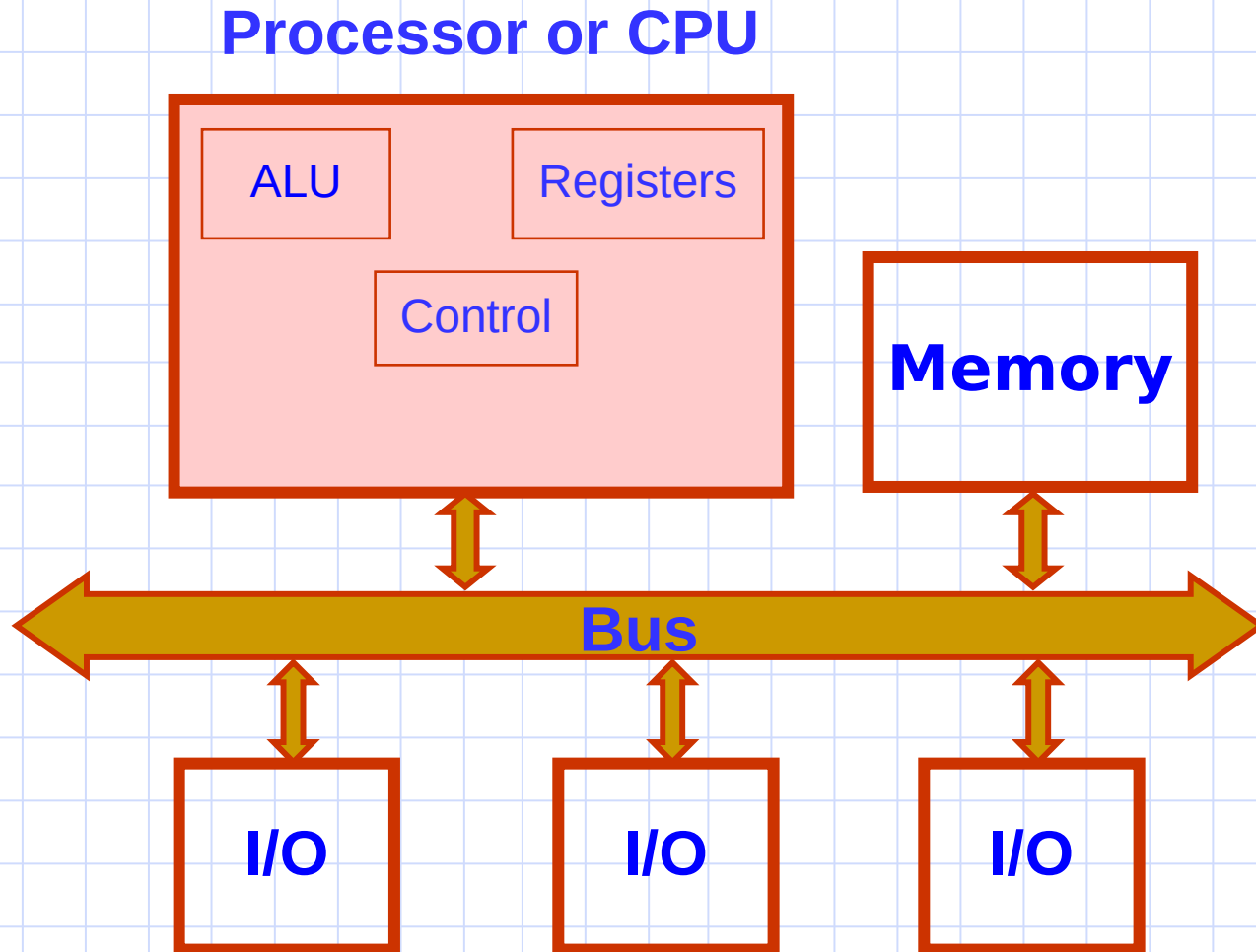
- (Assignment 1 Problem 1) What if X, Y had been signed ints?

- Instruction sets include 2 kinds of shifts

Logical

Arithmetic

Basic Computer Organization

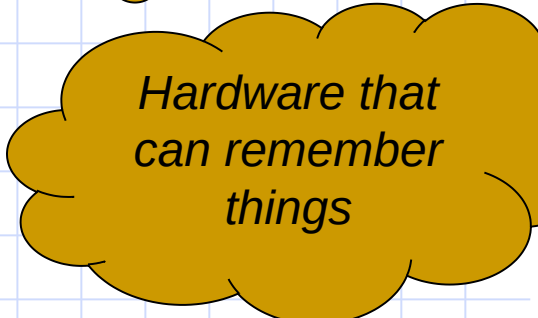


Inside the Processor...

- **Control hardware:** Hardware to manage instruction execution
- **ALU:** Arithmetic and Logical Unit (hardware to do arithmetic, logical operations)

Inside the Processor...

- **Control hardware:** Hardware to manage instruction execution
- **ALU:** Arithmetic and Logical Unit (hardware to do arithmetic, logical operations)
- **Registers:** small units of memory to hold data/instructions temporarily during execution



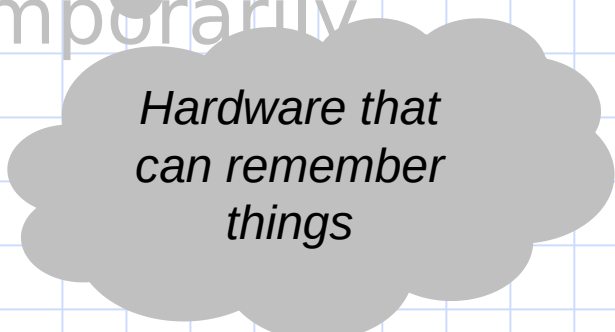
*Hardware that
can remember
things*

Memory

- What is memory?
 - Something that can remember things
- There are different kinds of memory in a computer system
 - Some remember by the state an electrical circuit is in e.g., **SRAM**
 - Others remember by the amount of electrical charge stored in a capacitor e.g., **DRAM**
 - Yet others remember by magnetic or optical properties e.g., **hard disk drive, CD, DVD**
- They can vary substantially in their speed and capacity

Inside the Processor...

- Control hardware: Hardware to manage instruction execution
- ALU: Arithmetic and Logical Unit (hardware to do arithmetic, logical operations)
- Registers: small units of memory to hold data/instructions temporarily during execution
- Two kinds of registers
 1. Special purpose registers
 2. General purpose registers



*Hardware that
can remember
things*