MSACS Kennesaw State University College of Science and Mathematics
Master of Science in *Applied* Computer Science

L02-CS8421-08-25-08

Architecture Overview P2

CS8421
Computing Systems
Dr. Ken Hoganson

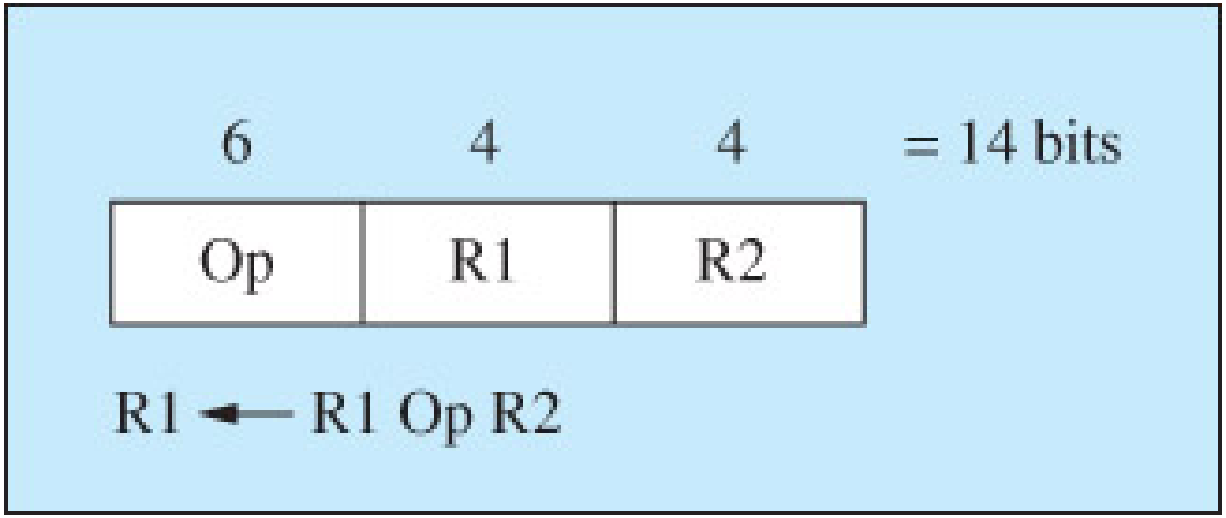
Class

Will

Start

Momentarily…

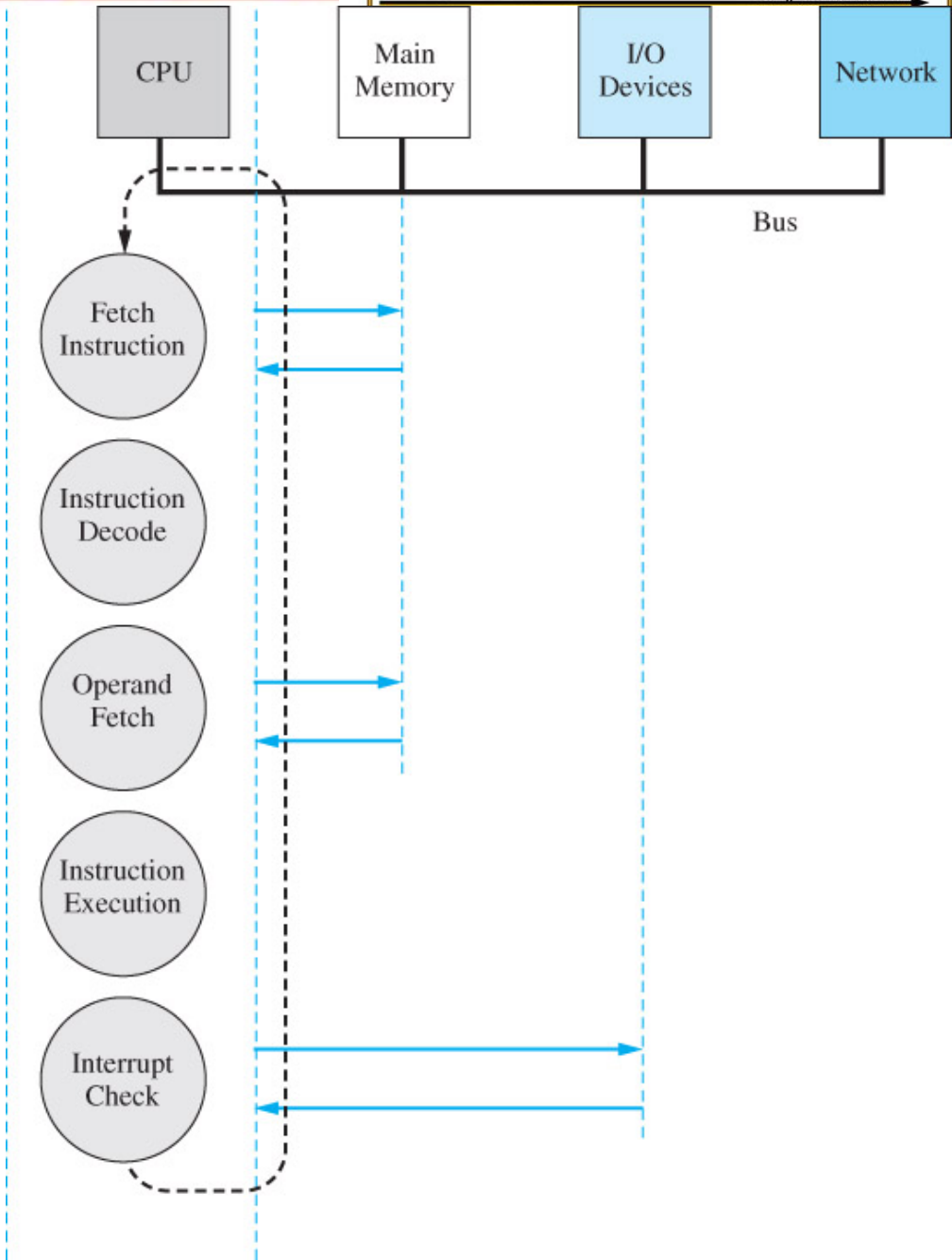
- Instruction Format: machine code and assembly language is organized in an instruction format
- Operation to be performed (OpCode or Op)
- Operands that the instruction will operate on (R1 and R2 are registers).



Fetch Decode Execute Cycle

- CPU repeats the cycle for each instruction
  1. Fetch the Instruction
  2. Decode the instruction (opcode and operands)
  3. Fetch and needed operands
  4. Execute the instruction
  5. Check for Interrupt

Processing Cycle



Simple Computer Program

```
LOAD R1 Num1

LOAD R2 Num2

ADD   R1  R2

STOR  R1  Result
```

Initial State of Computer

- Register 1 empty
- Register 2 empty
- Number 1 = 4
- Number 2 = 5
- Result empty

CPU	
Register	Contents
R1	
R2	

Memory	
Location	Contents
Num1	4
Num2	5
Result	

First Instruction

After executing the first instruction:

LOAD R1 Num1

CPU	
Register	Contents
R1	**4**
R2	

Memory	
Location	Contents
Num1	**4**
Num2	5
Result	

Second Instruction

After executing the second instruction:

LOAD R2 Num2

CPU	
Register	Contents
R1	4
R2	**5**

Memory	
Location	Contents
Num1	4
Num2	5
Result	

Third Instruction

After executing the third instruction:

ADD R1 R2

Note that the sum overwrites the temporary storage in R1.

CPU	
Register	Contents
R1	**9**
R2	**5**

Memory	
Location	Contents
Num1	4
Num2	5
Result	

Fourth Instruction

After executing the fourth instruction:

STOR R1 Result

CPU	
Register	Contents
R1	**9**
R2	5

Memory	
Location	Contents
Num1	4
Num2	5
Result	**9**

High Level Language

```
LOAD R1 Num1
LOAD R2 Num2
ADD    R1  R2
STOR  R1  Result
```

Can be accomplished in a high-level language with a single instruction or programming statement:

```
Result = Num1 + Num2
```

Computer Architecture Layers

1. Transistors and Gates: constructing small devices called Logic Gates from transistors.
2. Simple Devices: building simple devices from gates
3. System Devices: building more complex devices
4. Computer System Architecture: building a system
5. Instruction Set: specifying the CPU internal architecture and capabilities with the instruction set. The instruction set specifies the programmer's interface to the hardware, both machine code and assembly language.
6. Operating Systems
7. Distributed, N-Tier, Client/Server and Parallel systems

Distributed, N-Tier, Client/Server, Parallel and Grid systems.

Operating System: Manages resources and provides user and program interface.

Instruction Set: Internal design of the CPU and the machine code.

Computer System Architecture: building a computer from system level devices

System Devices: more complex system-level devices from simple devices and gates.

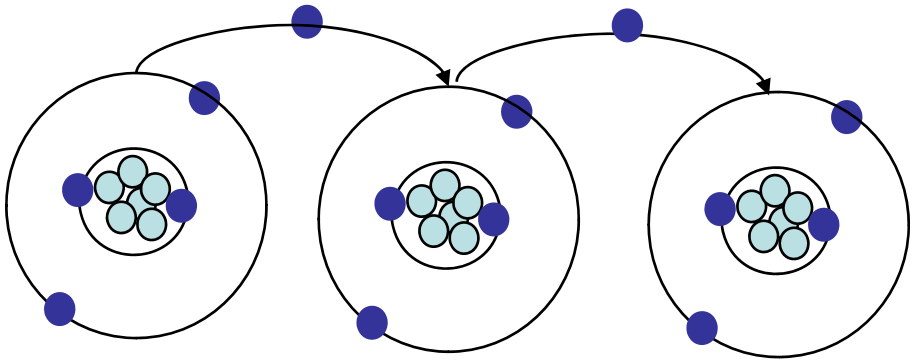
Simple Devices: building simple, elemental devices using gates as the construction components.

Transistors and Gates: constructing small devices called Logic Gates from transistors.

Figure 2.10 Computer System Design Layers

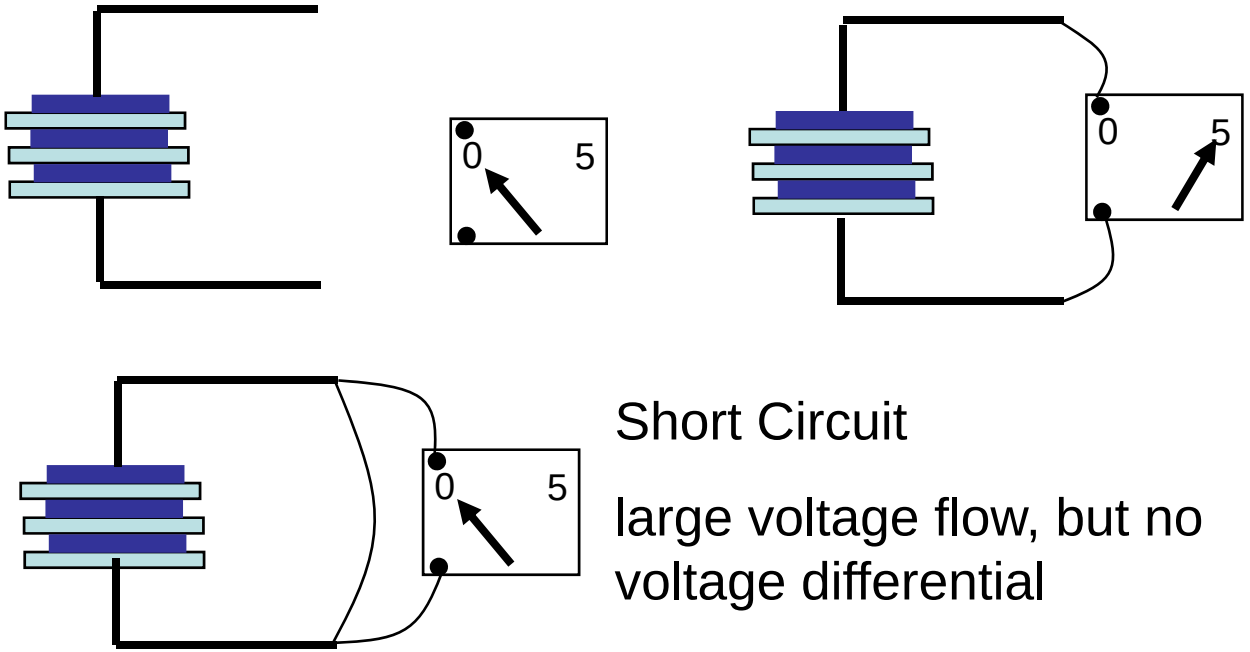
Electricity

- The modern computer is a digital and electronic device.
- Runs on flow of electrons - movement of electrons from outer valence shells on atoms
- Requires force/energy to pull an electron out of a shell - Electro-motive force (EMF) - Voltage
- How tightly the electrons are held determines how much voltage is needed and is quantified as resistance
- Quantity of Electrons flowing is quantified as Amperes



Measuring EMF

- Computers are electronic switching devices
- Voltage source (battery)
- Meter to detect voltage differential

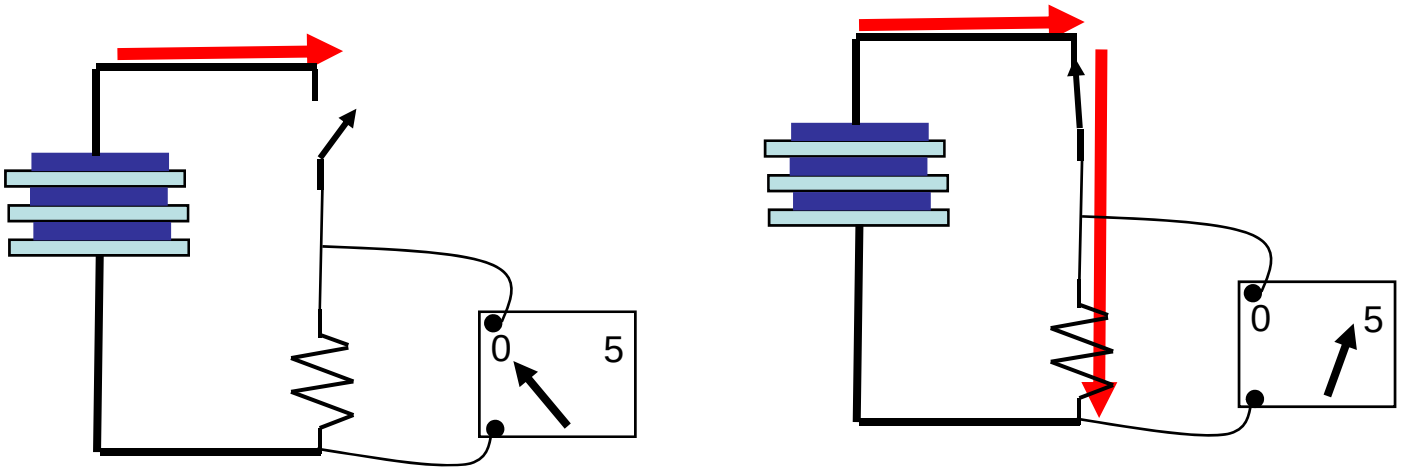


Short Circuit

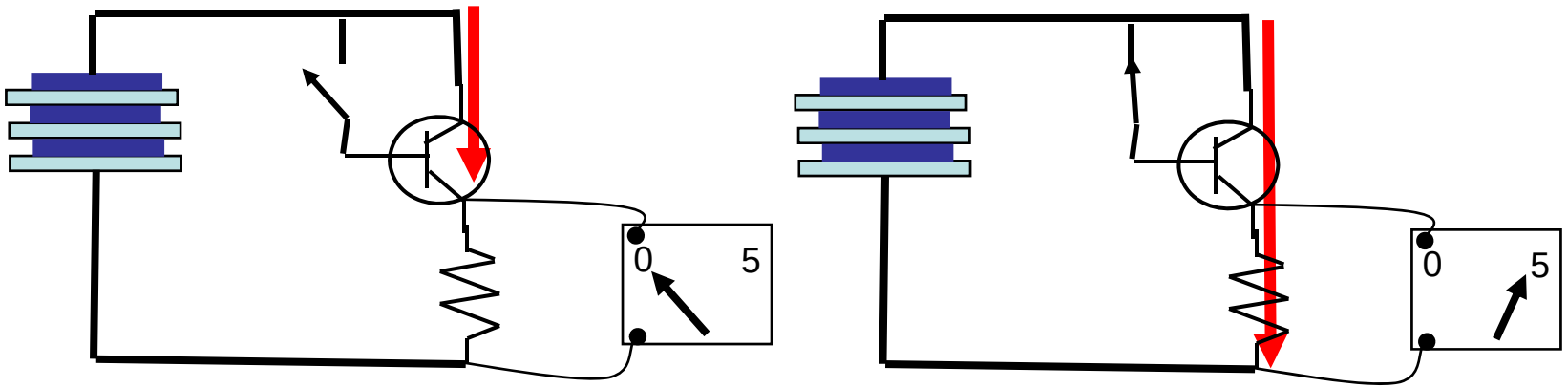
large voltage flow, but no voltage differential

Switching

- Computers are constructed from transistors
- Transistor are used as on-off switches (0-off, 1-on), hence binary

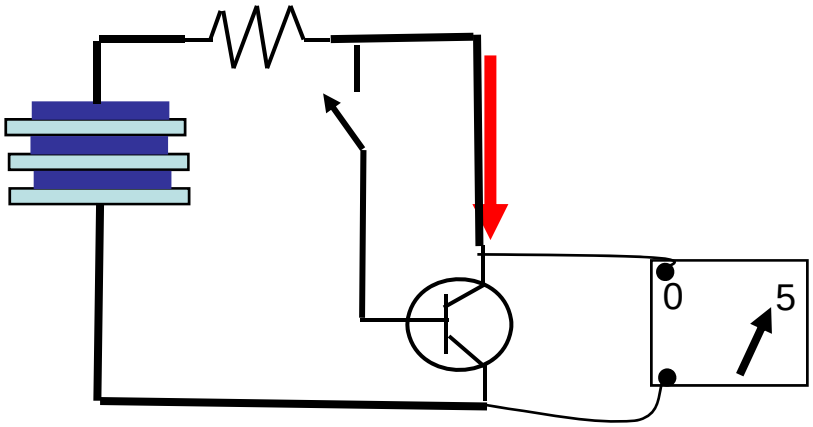


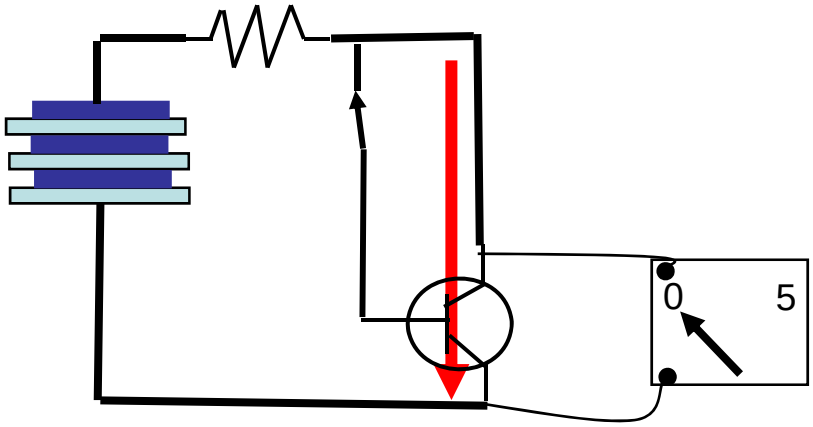
Transistor Switching



Measuring Output before the transistor

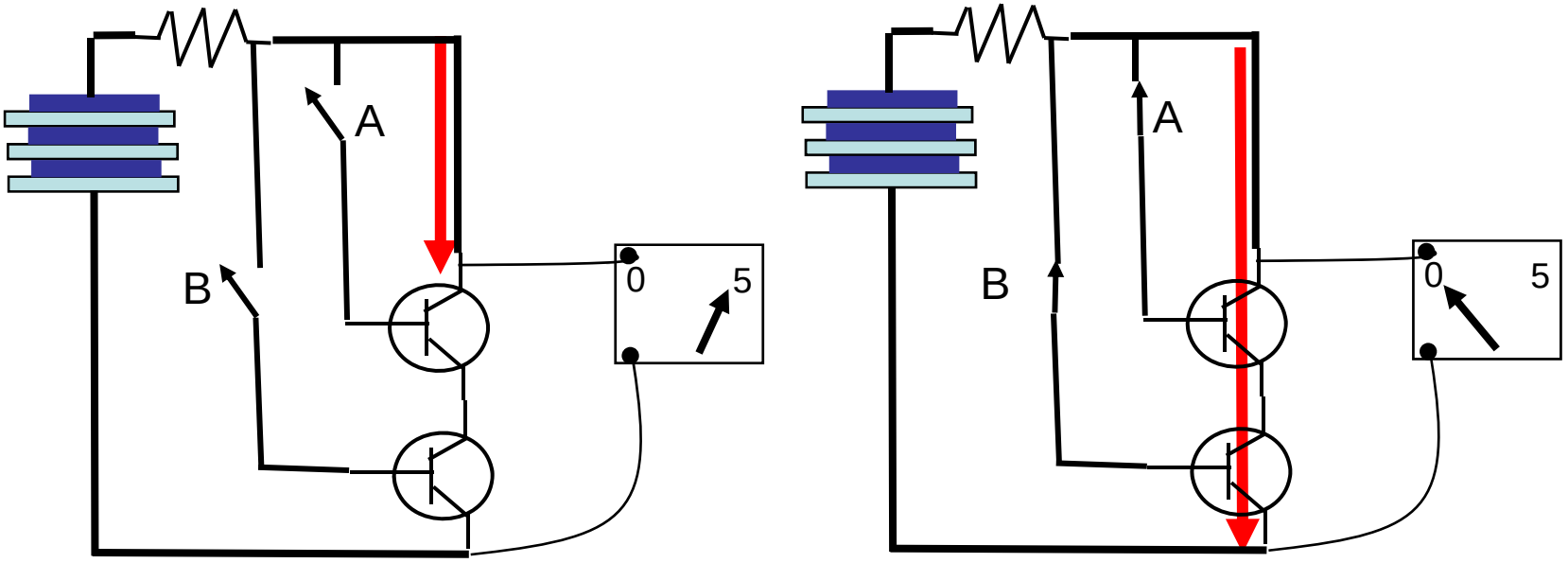
- Acts as an **invertor**
- No voltage on the input causes the meter to read 5 volts
- A voltage on the input causes the meter to read zero





NAND – Not AND

A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0



End of Lecture

End

Of

Today's

Lecture.

This slide intentionally left blank