Gør tanke til handling
VIA University College

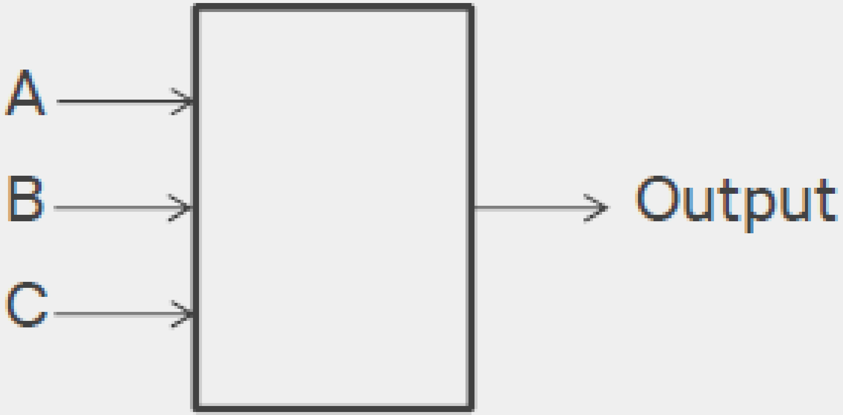
Lecture 2

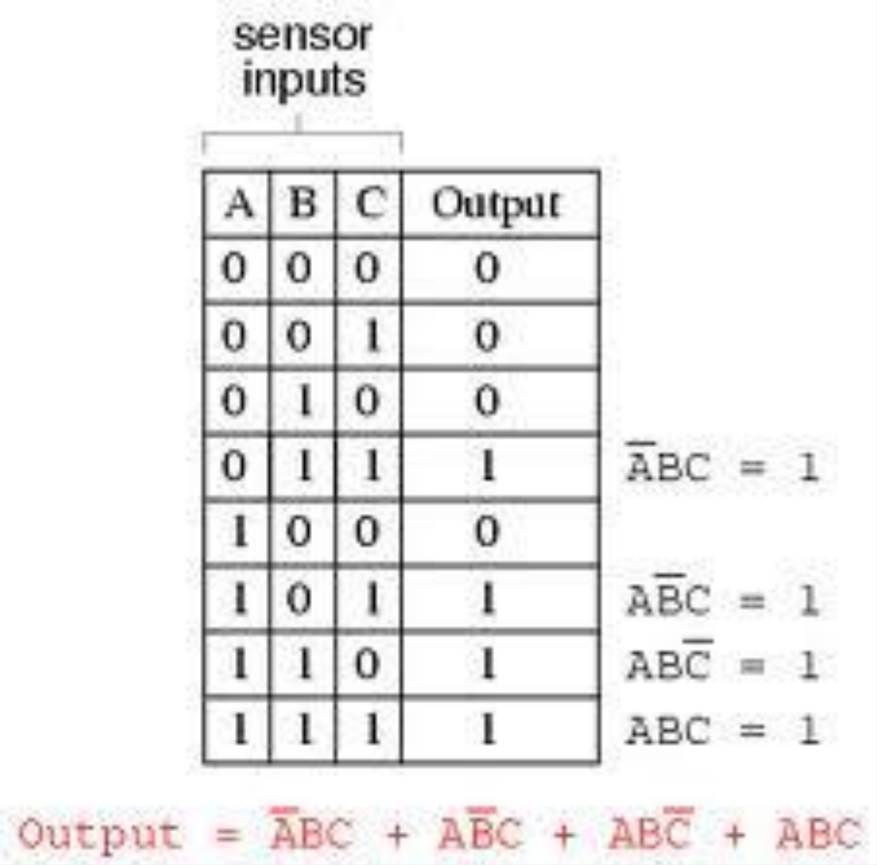
Boolean Arithmetic and ALU

Today u will learn

- Convert truth-table to function and visa versa
- Reducing boolean functions
- Half adder
- Full adder
- Multibit adder
- Multiplexer and demultiplexer
- The Arithmetic Logic Unit (ALU)
- Exercise

Boolean algebra / truth tables

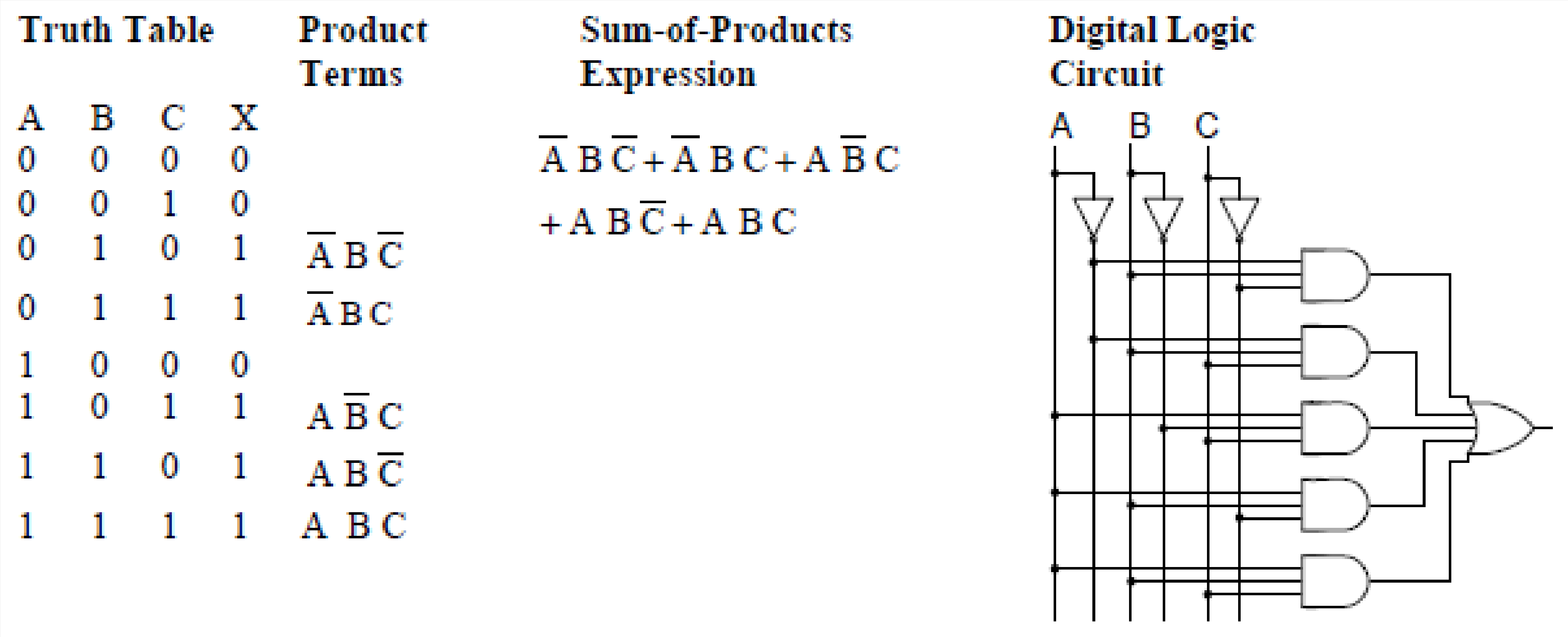



sensor inputs

A	B	C	Output	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{A}BC = 1$
1	0	0	0	
1	0	1	1	$A\overline{B}C = 1$
1	1	0	1	$AB\overline{C} = 1$
1	1	1	1	$ABC = 1$

$$\text{Output} = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

Sum of products



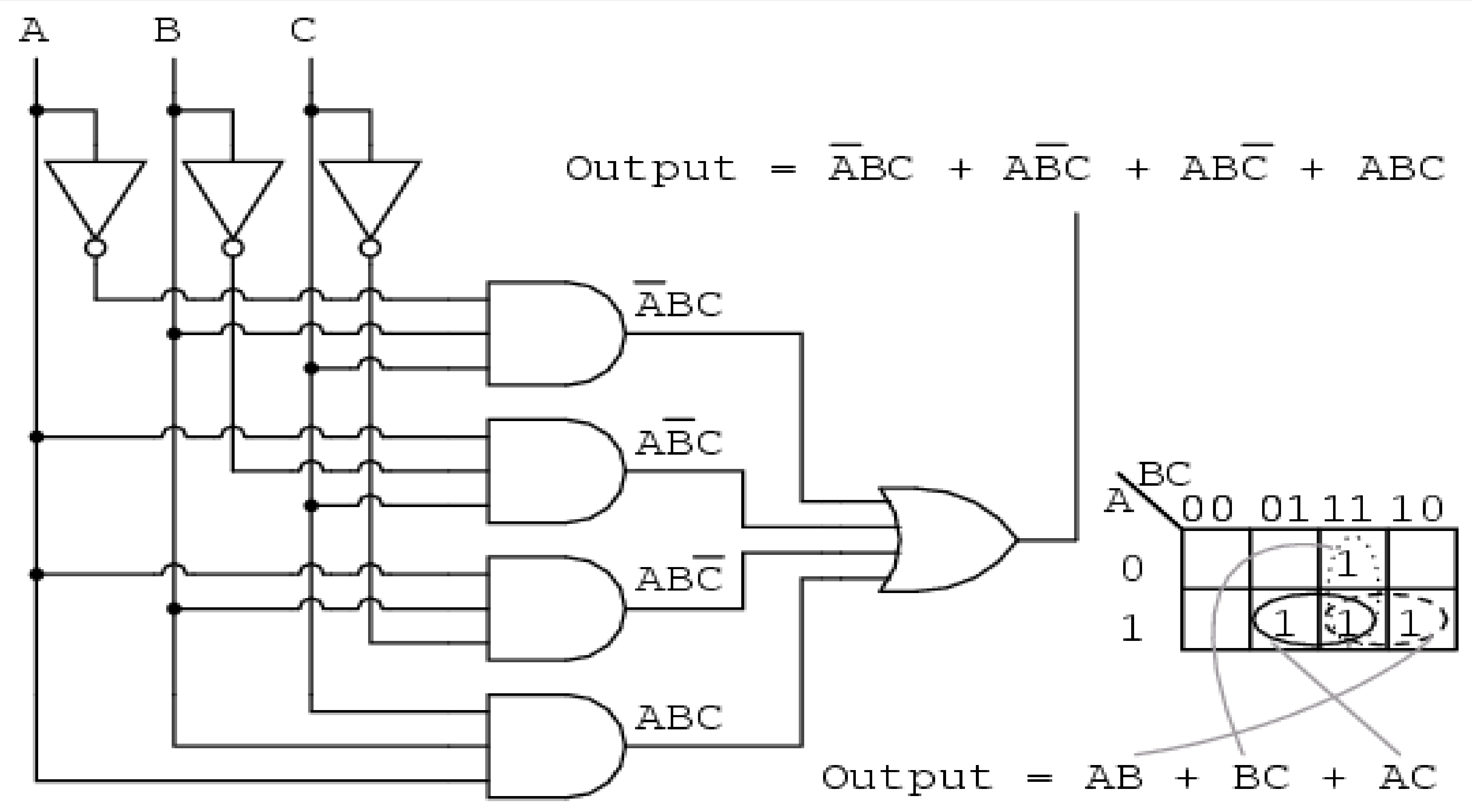
Truth Table				Product Terms
A	B	C	X	
0	0	0	0	
0	0	1	0	
0	1	0	1	$\overline{A}B\overline{C}$
0	1	1	1	$\overline{A}BC$
1	0	0	0	
1	0	1	1	$A\overline{B}C$
1	1	0	1	$AB\overline{C}$
1	1	1	1	ABC

Sum-of-Products Expression

$$\overline{A}B\overline{C}+\overline{A}BC+A\overline{B}C$$
$$+AB\overline{C}+ABC$$

Digital Logic Circuit

KarnaughMaps



Output = $\overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$

A \ BC	00	01	11	10
0			1	
1		1	1	1

Output = AB + BC + AC

Truth table to Function

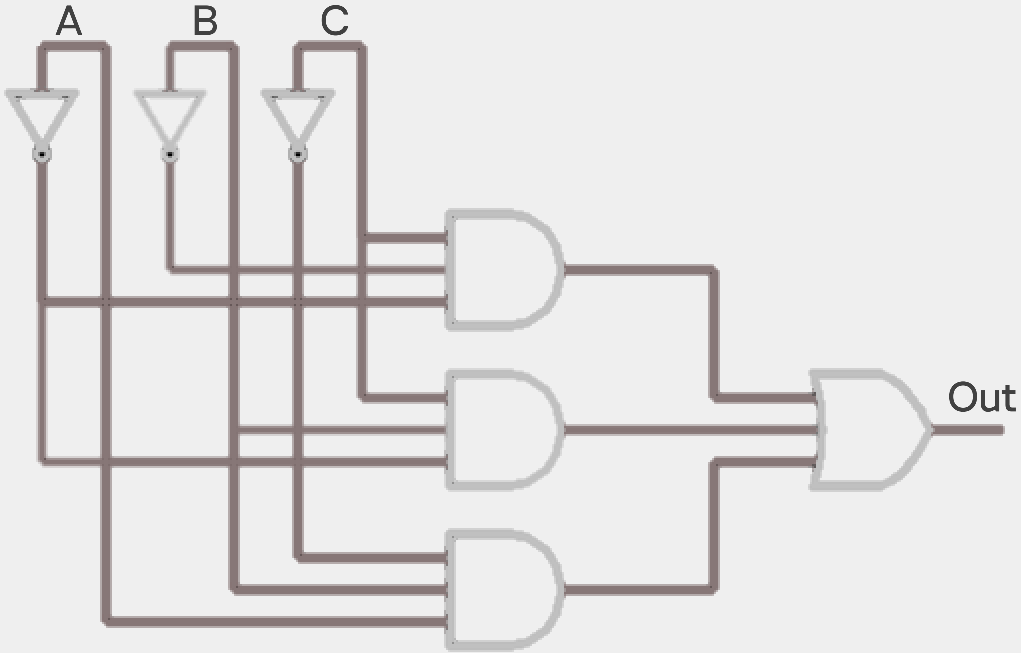
- Describe the 1's
- Simplify the expression

$\overline{AB}C + \overline{A}BC + AB\overline{C}$

$\overline{A}C + \overline{A}C + AB\overline{C}$

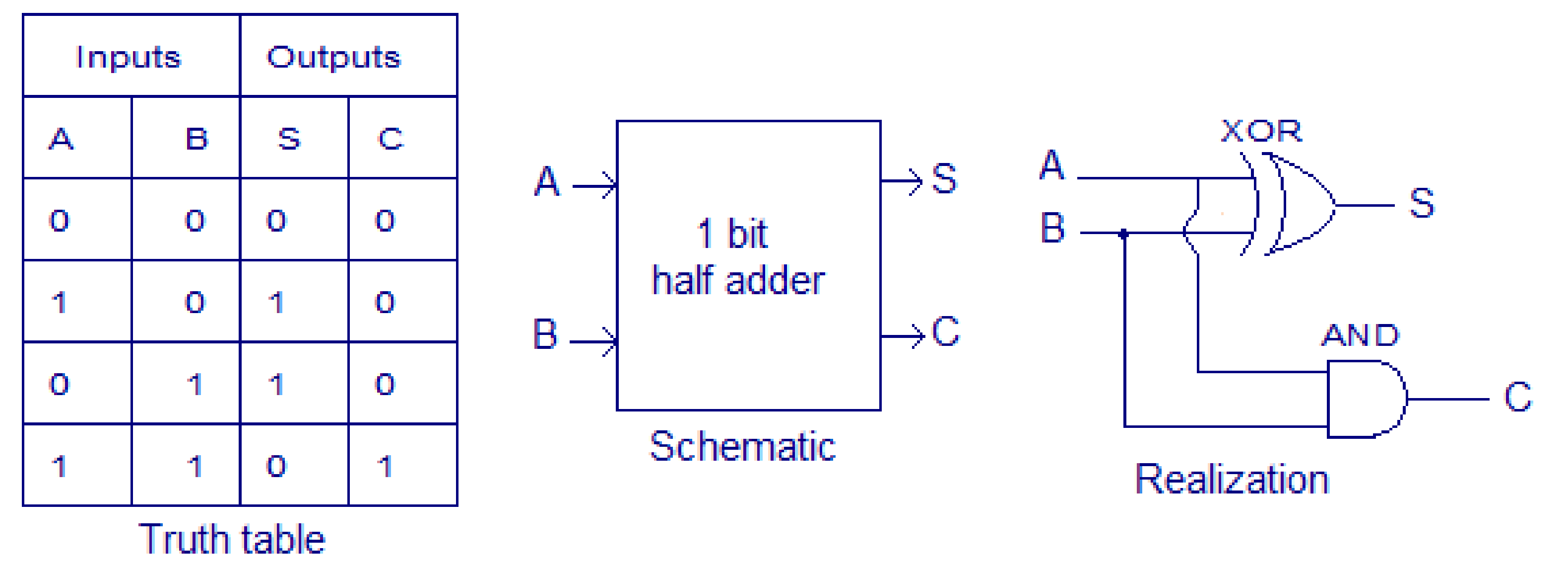
$\overline{A}C + AB\overline{C}$

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



Half- and full-adders

- A half-adder takes two bits and generates sum and carry



Inputs		Outputs	
A	B	S	C
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

Truth table

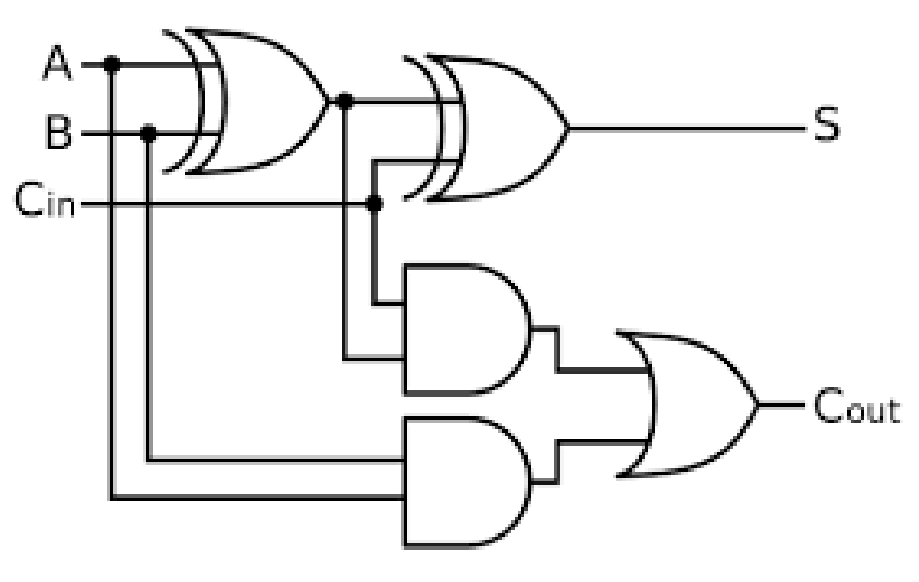
Schematic

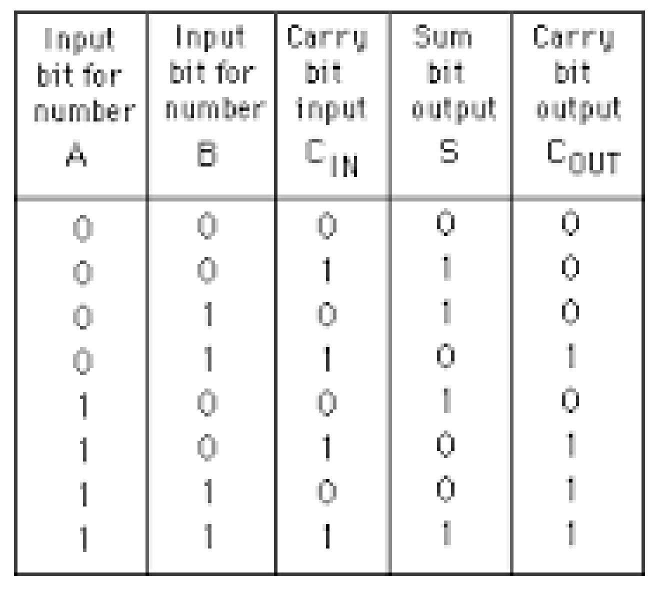
Realization

http://tinyurl.com/ybkt6ffj

Half- and full-adders

- A full-adder takes two bits and the carry from the previous (lower-order) bit and generates sum and carry:

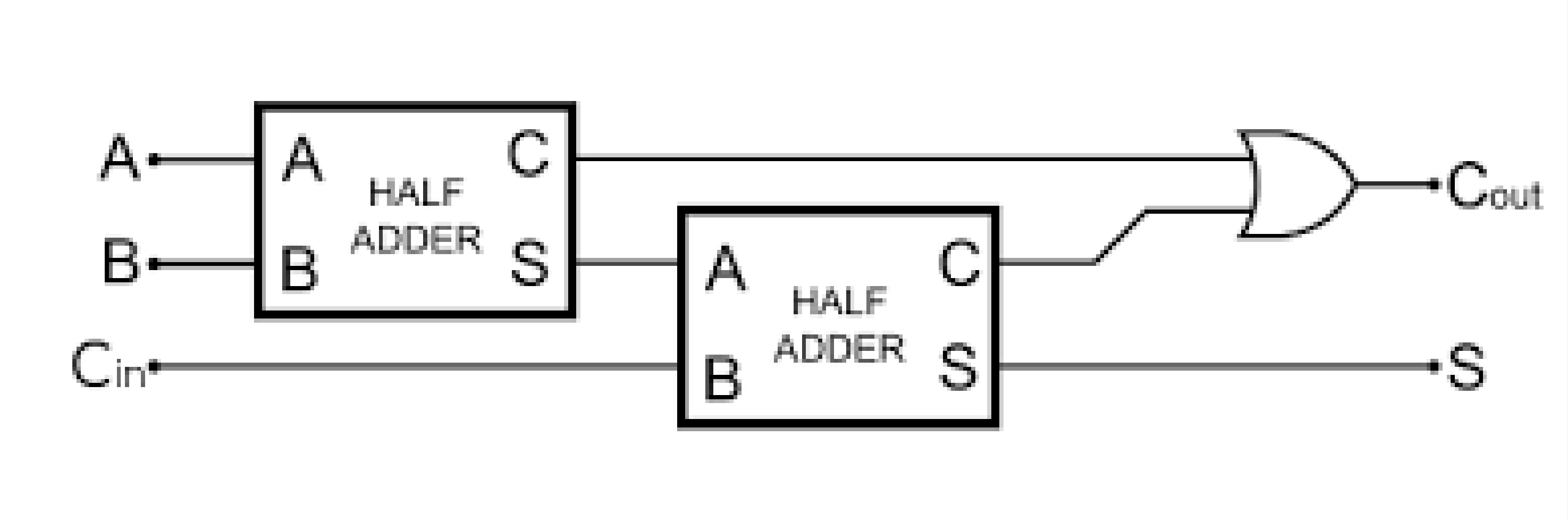




Input bit for number A	Input bit for number B	Carry bit input C_{IN}	Sum bit output S	Carry bit output C_{OUT}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

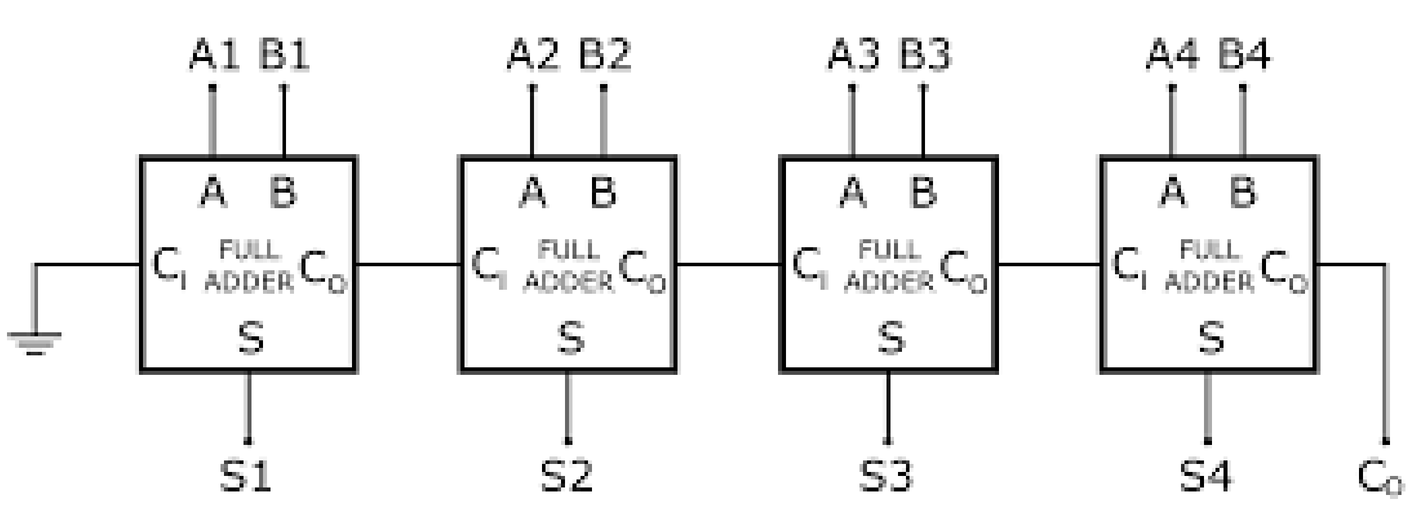
Half- and full-adders

- A full-adder can be built using two half adders



Half- and full-adders

- For more info look at link: http://en.wikibooks.org/wiki/Practical_Electronics/Adders

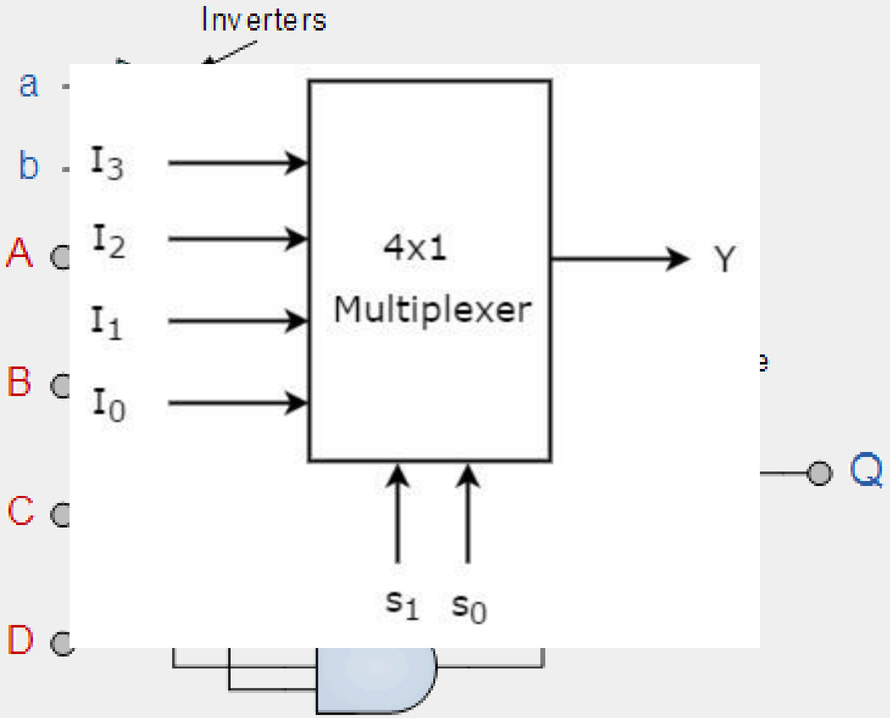


Multibit adder

- A 4bit adder consist of 4 “Full adder”
- The AVR has an 8 bit adder. The carry from the MSB goes in the status register

Multiplexer

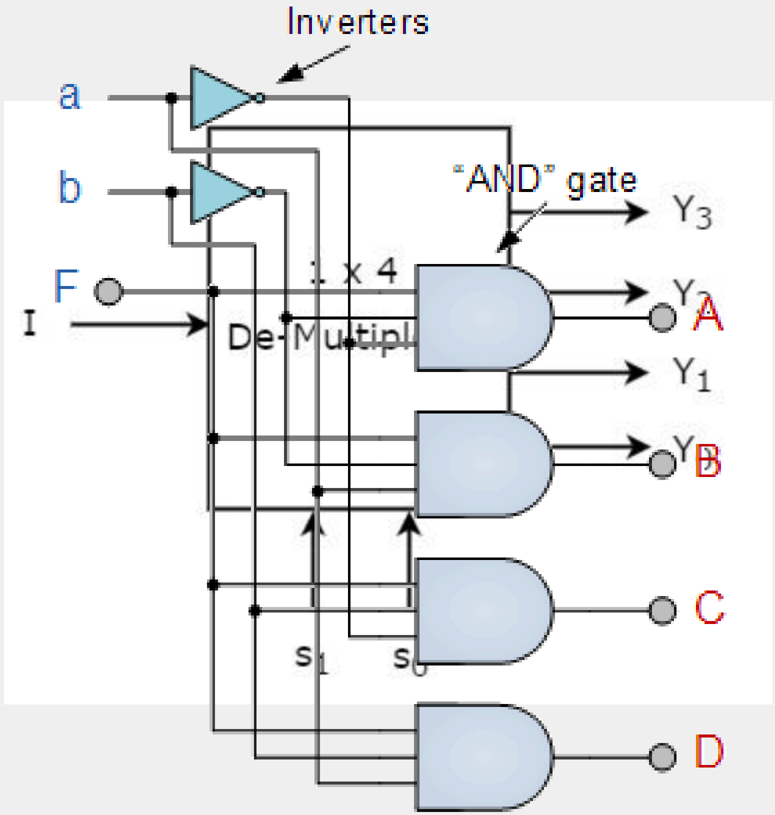
- Used for selecting 1 of multiple inputs
- Has Multiple inputs
- And some select bits



- http://tinyurl.com/yyc4dz36

Demultiplexer

- Use to send 1 input to a specific output
- 1 input
- Some select bit

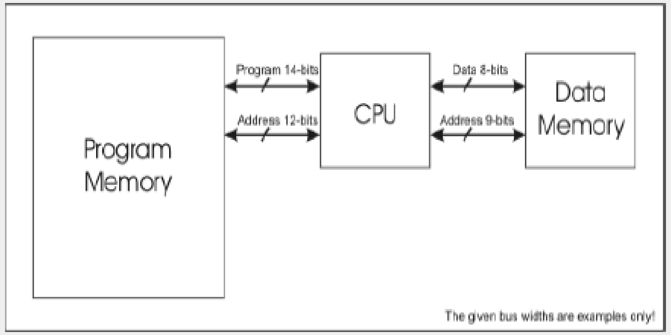


- http://tinyurl.com/y6cx4qx9

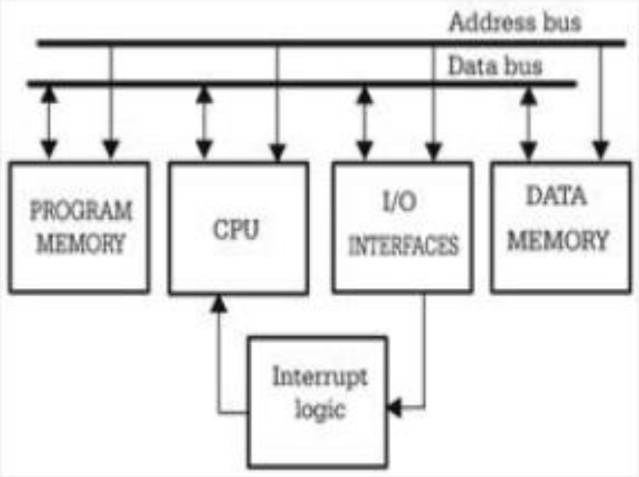
Computer Architecture Types

- –Harvard
    - –One program bus
    - –One data bus
    - –Can work with instructions and data simultaneously

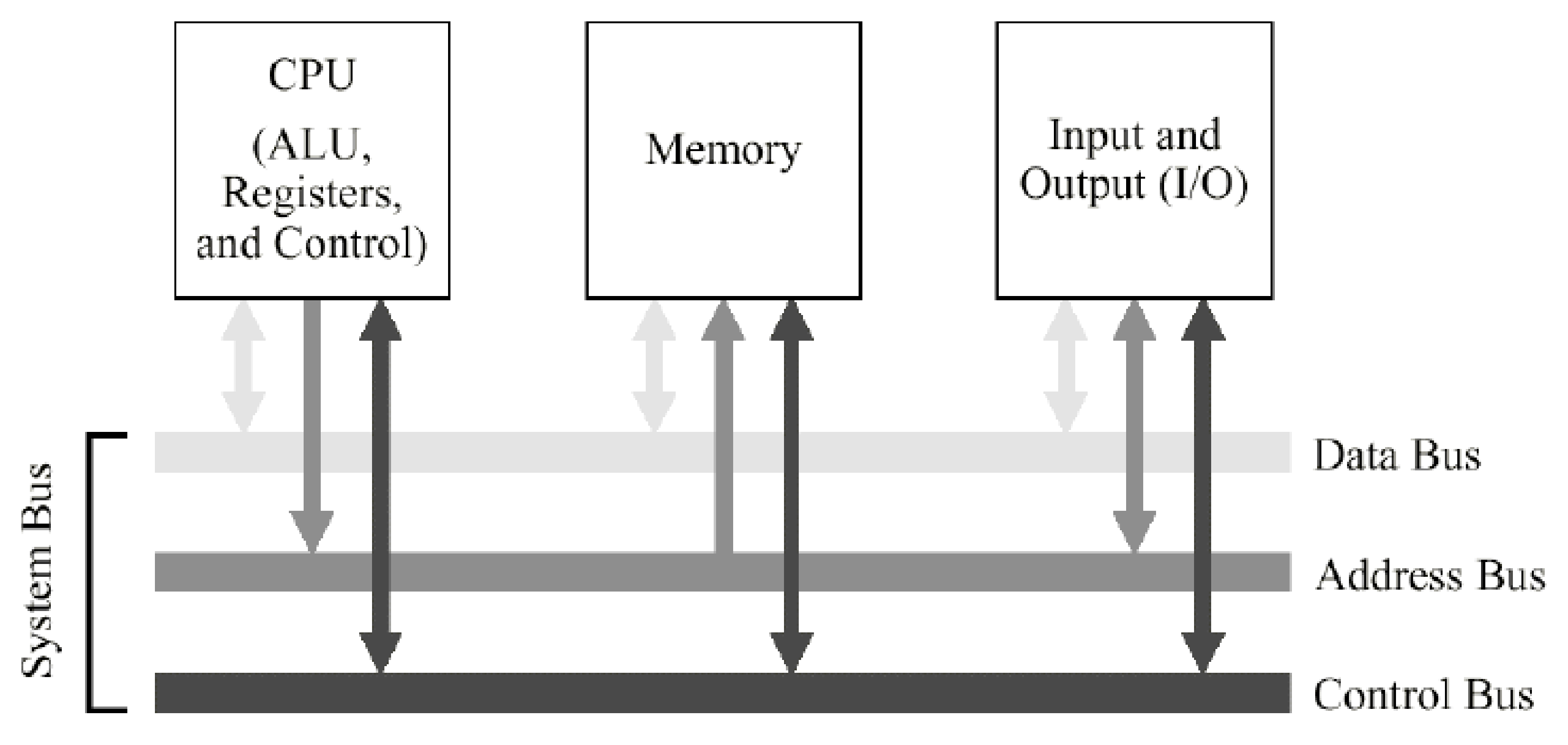
- –Von Neumann
    - –One bus
    - –Bus holds either instruction or data



The given bus widths are examples only!



Computer organisation



A bus is multiple wires. Could be a 8 wire bus, if the the PC is an 8bit processor

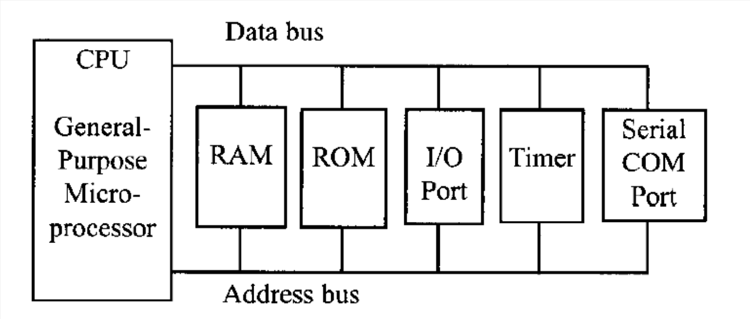
Microprocessor vs. Microcontroller

General purpose computer

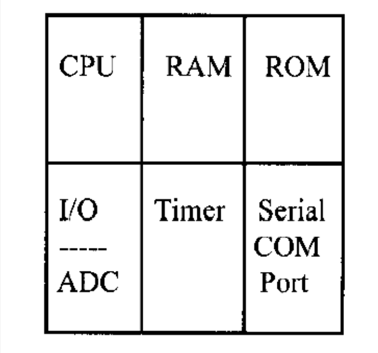
- ·Highly configurable hw
- ·Many applications
- ·Different purposes
- ·Advanced user interface
- ·Human to human communication

Embedded computer

- –Fixed hw configuration
- –Single application
- –One purpose
- –Simple user interface
- –Machine to machine communication



(a) General-Purpose Microprocessor System

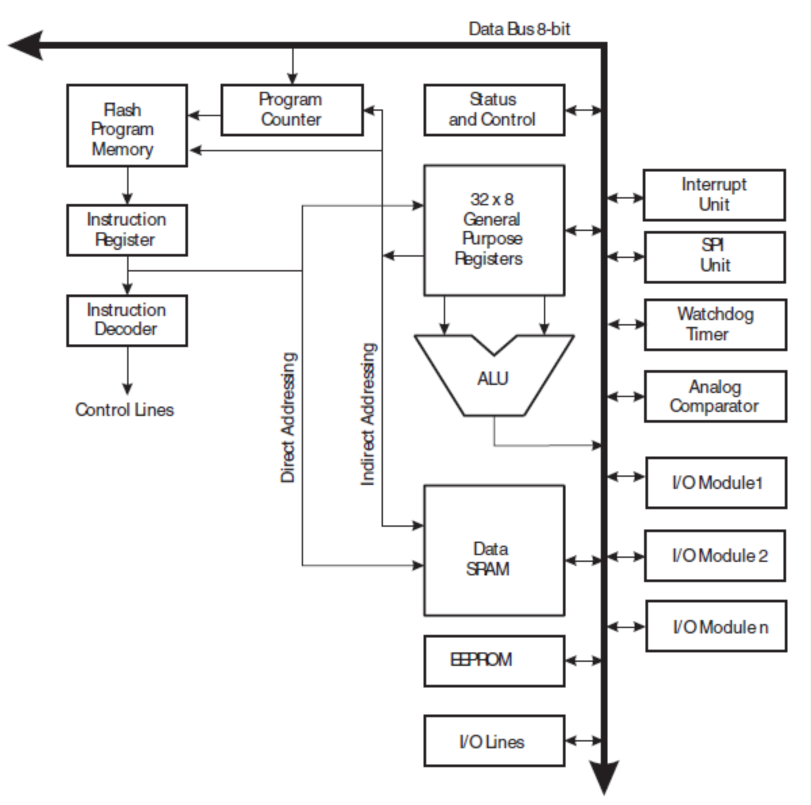


(b) Microcontroller

The Central Processing Unit

The CPU typically consists of

- –ALU
- –Program counter
- –Instruction register
- –Instruction decoder
- –Input/Output devices

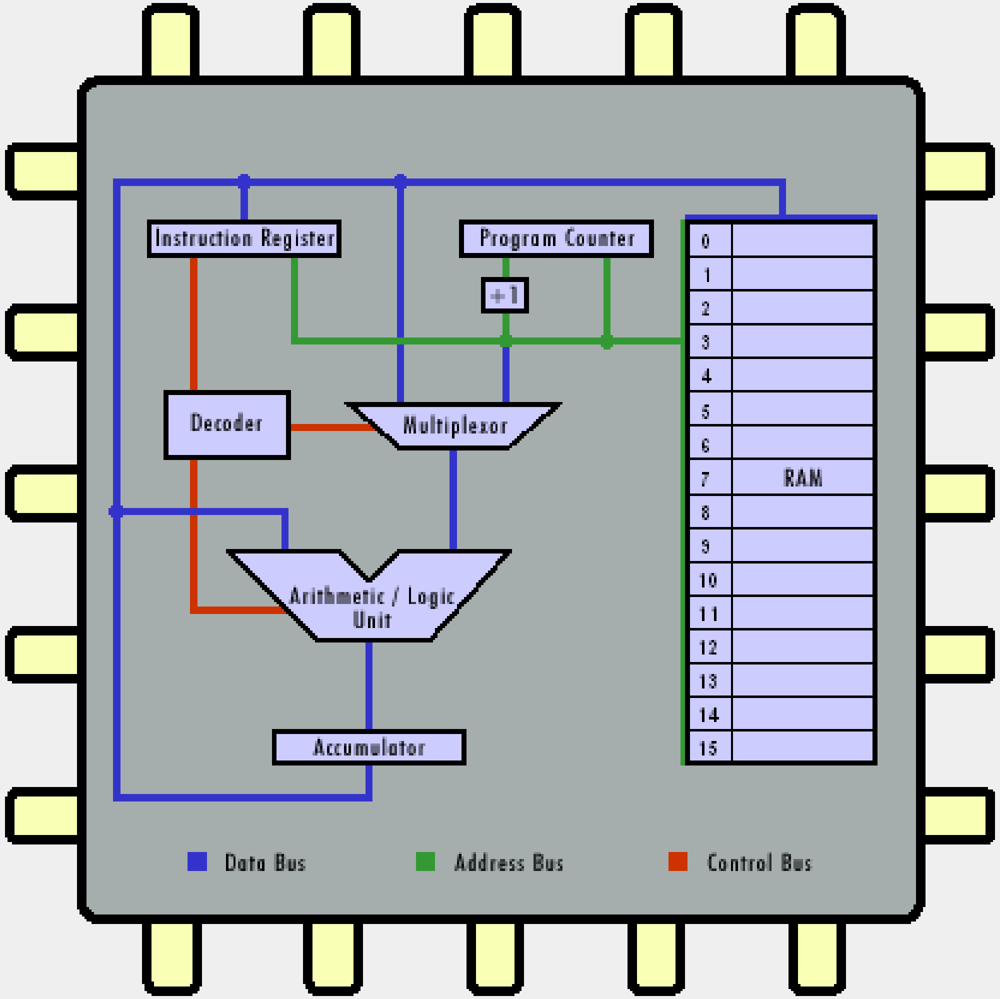


CPU – Animation

The heart of a computer is the central processing unit or CPU. This device contains all the circuitry that the computer needs to manipulate data and execute instructions.

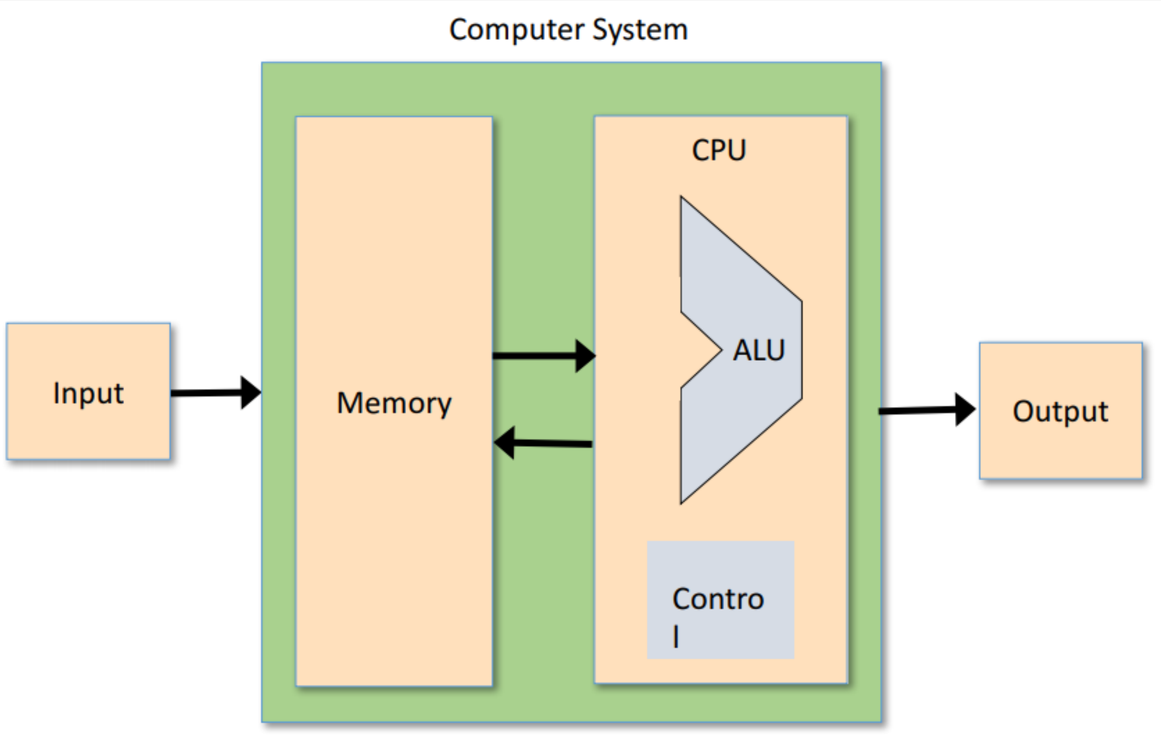
Some of the circuits of a computer are made of gates

http://courses.cs.vt.edu/~csonline/MachineArchitecture/Lessons/CPU/Lesson.html

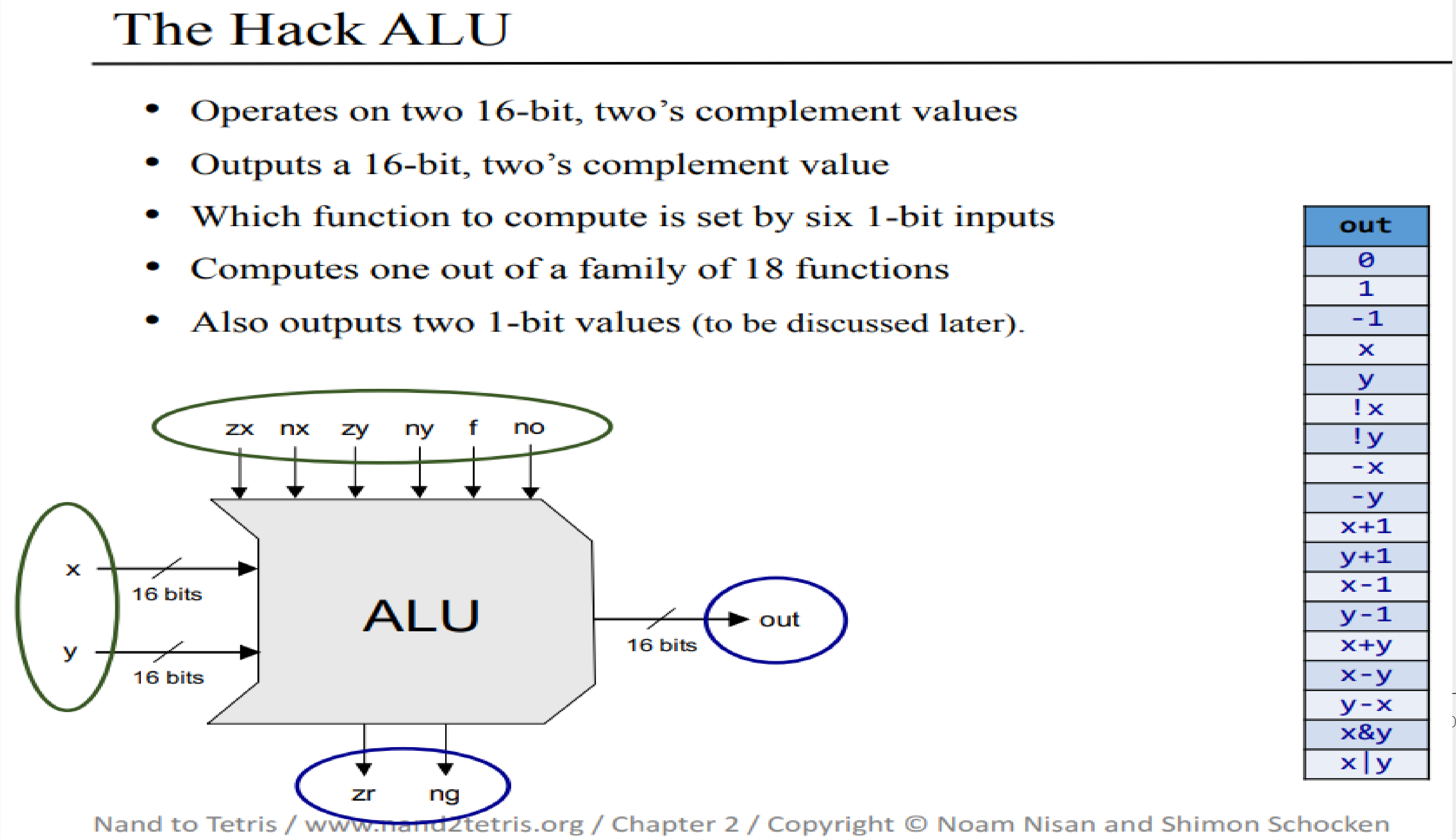


The Arithmetic Logical Unit (ALU)

- Can do very simple math



The Arithmetic Logical Unit (ALU)



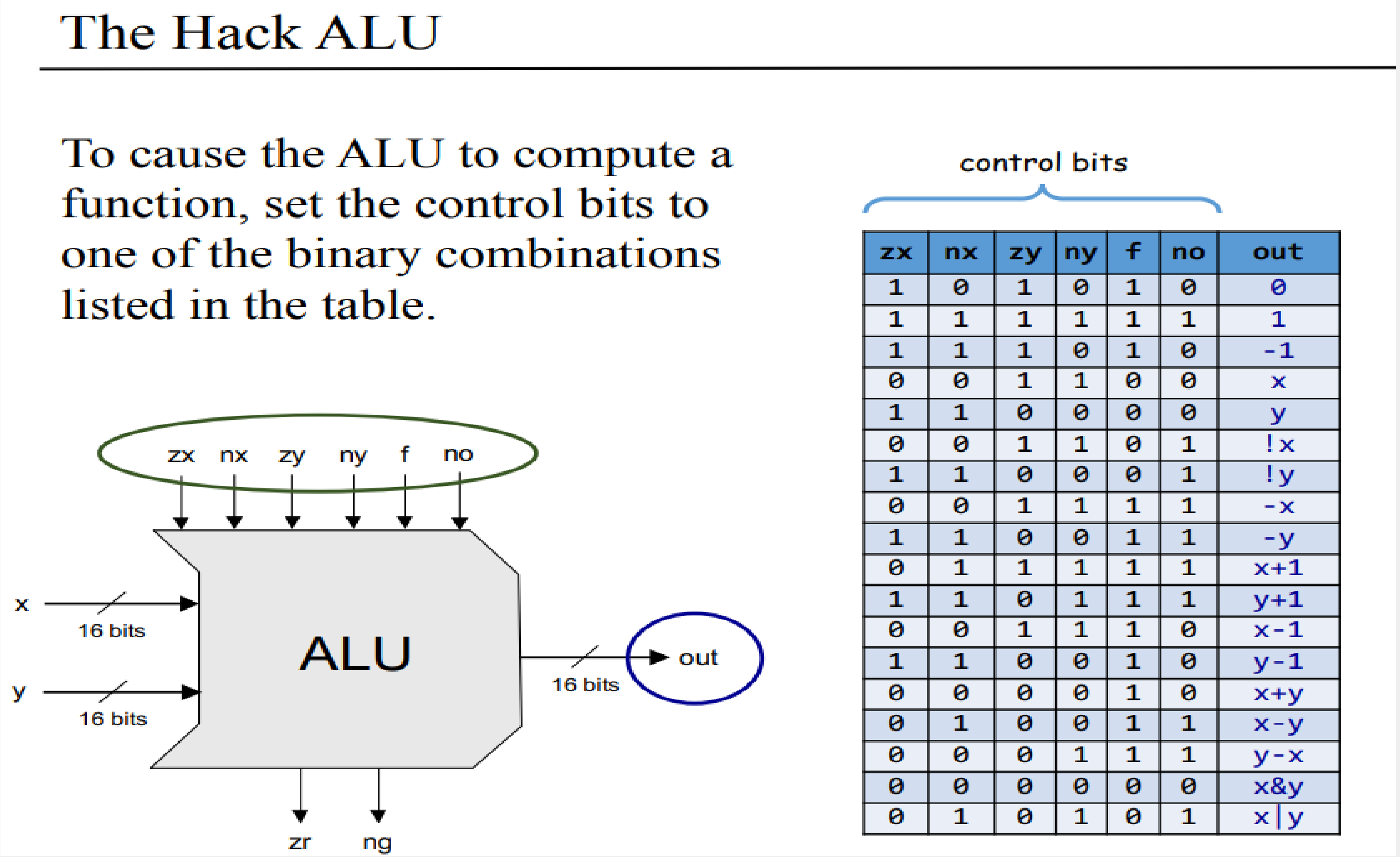
The Hack ALU

- Operates on two 16-bit, two's complement values
- Outputs a 16-bit, two's complement value
- Which function to compute is set by six 1-bit inputs
- Computes one out of a family of 18 functions
- Also outputs two 1-bit values (to be discussed later).

out
0
1
-1
x
y
!x
!y
-x
-y
x+1
y+1
x-1
y-1
x+y
x-y
y-x
x&y
x|y

Nand to Tetris / www.nand2tetris.org / Chapter 2 / Copyright © Noam Nisan and Shimon Schocken

The Arithmetic Logical Unit (ALU)

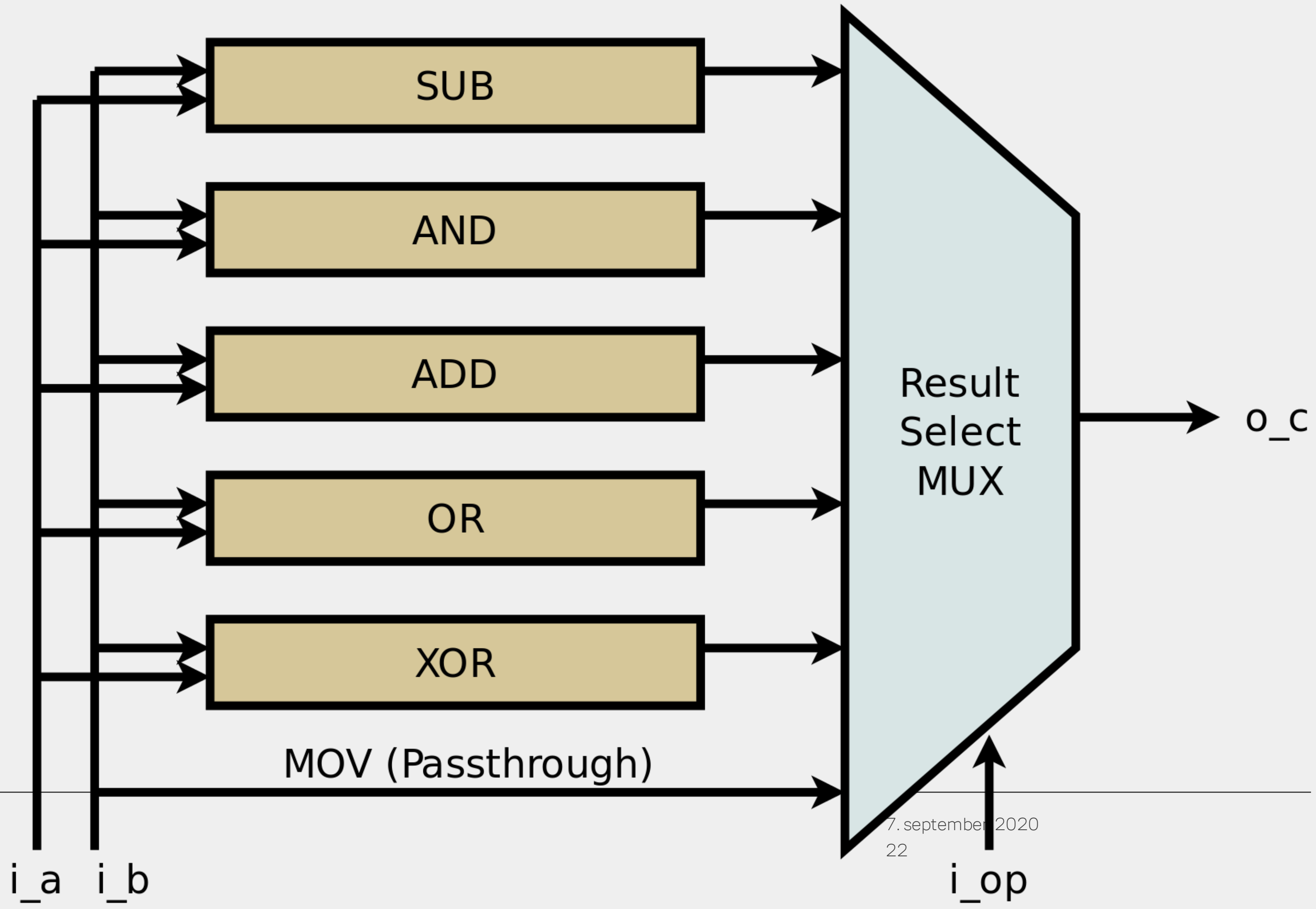


The Hack ALU

To cause the ALU to compute a function, set the control bits to one of the binary combinations listed in the table.

zx	nx	zy	ny	f	no	out
1	0	1	0	1	0	0
1	1	1	1	1	1	1
1	1	1	0	1	0	-1
0	0	1	1	0	0	x
1	1	0	0	0	0	y
0	0	1	1	0	1	!x
1	1	0	0	0	1	!y
0	0	1	1	1	1	-x
1	1	0	0	1	1	-y
0	1	1	1	1	1	x+1
1	1	0	1	1	1	y+1
0	0	1	1	1	0	x-1
1	1	0	0	1	0	y-1
0	0	0	0	1	0	x+y
0	1	0	0	1	1	x-y
0	0	0	1	1	1	y-x
0	0	0	0	0	0	x&y
0	1	0	1	0	1	x|y

The Arithmetic Logical Unit (ALU)



Exercise

- We know how to make OR and XOR from NAND gates. So no reason to do that again.