



# COMPUTER SYSTEMS FUNDAMENTALS ( 4COSC004W )

Lecture: Week 3. Part 1 of 3



# In this video we will cover:

- Binary operations
  - *NOT*
  - *AND*
  - *OR*
  - *Addition*
  - *Shift Left , Shift Right*

# BINARY OPERATIONS

NOT, AND, OR, Addition

# By the end of this section you will:

- Be able to perform the **NOT** binary operation
- Be able to perform the **AND** binary operation
- Be able to perform the **OR** binary operation
- Be able to perform Binary additions

# Binary NOT operation

- Negation
- $\text{NOT}(0) = 1$
- $\text{NOT}(1) = 0$

NOT	NOT
1	0
0	1

NOT							
1	1	1	0	0	1	0	0
0	0	0	1	1	0	1	1

# Binary AND operation

- $a \text{ AND } b$  is True
  - *Iff both  $a$  AND  $b$  are True*

AND	AND	AND	AND
0	1	0	1
0	0	1	1
0	0	0	1

AND							
1	0	1	0	1	0	1	0
1	1	0	0	1	1	0	0
1	0	0	0	1	0	0	0

# Binary OR operation

- $a \text{ OR } b$  is True
  - If  $a$  OR  $b$  are True
  - Or  $a$  AND  $b$  are True

OR	OR	OR	OR
0	1	0	1
0	0	1	1
0	1	1	1

OR							
1	0	1	0	1	0	1	0
1	1	0	0	1	1	0	0
1	1	1	0	1	1	1	0

# Binary addition

addition				
	1	0	1	0
+	0	0	1	1
	1	1	0	1
		1		

$$\begin{array}{r} 10 \\ + 3 \\ \hline 13 \end{array}$$



# Binary addition

addition				
	1	0	1	0
+	0	1	1	1
1	0	0	0	1
1	1	1		

$$\begin{array}{r} 10 \\ + 7 \\ \hline 17 \end{array}$$

# Binary addition

addition				
	1	1	1	0
+	0	1	1	1
1	0	1	0	1
1	1	1		

$$\begin{array}{r} 14 \\ + 7 \\ \hline 21 \end{array}$$

# SHIFT LEFT & SHIFT RIGHT

Arithmetic functions of CPU

# By the end of this section, you will:

- Be able to perform Left Shift operations on Binary values
  - *Appreciate that a single Left Shift is a multiplication by 2*
- Be able to perform Right Shift operations on Binary values
  - *Appreciate that a single Right Shift is a division by 2*
- Understand primitive arithmetic functions of the CPU

# Left Shift

- Shifting a Binary value, one position to the LEFT
- Multiplication by 2

Left Shift					
				1	1
			1	0	2

Left Shift					
			1	0	2
		1	0	0	4

Left Shift					
		1	0	1	5
	1	0	1	0	10

Denary	Binary			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

# Right Shift

- Shifting a Binary value, one position to the Right
- Division by 2

Right Shift						
			1	0		2
				1		1

Right Shift						
		1	1	0		6
			1	1		3

Right Shift						
		1	0	1		5
			1	0	1	2

Denary	Binary			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1



# Left Shift vs. Right Shift

Left Shift – multiply by 2

Binary							
					1	1	3
				1	1	0	6
			1	1	0	0	12
		1	1	0	0	0	24
	1	1	0	0	0	0	48
1	1	0	0	0	0	0	96

Right Shift – divide by 2

Binary							
	1	0	0	1	0	0	36
		1	0	0	1	0	18
			1	0	0	1	9
				1	0	0	4
					1	0	2
						1	1

# Primitive arithmetic functions performed by the CPU (ALU)

- Simple, fast Binary operations:
  - *Addition (last week)*
  - *Multiplication by 2 (Left Shift)*
  - *Division by 2 (Right Shift)*
  - *Subtraction (next unit)*

# Composite arithmetic:

- Multiplication by 5:
  - *Multiply by 2 (Left Shift)*
  - *Multiply by 2 (Left Shift)*
  - *Add original number*
- Multiply by 10:
  - *Multiply by 2 (Left Shift)*
  - *Multiply by 2 (Left Shift)*
  - *Multiply by 2 (Left Shift)*
  - *Add original number*
  - *Add original number*

# In this video we looked at:

- Binary operations
  - *NOT*
  - *AND*
  - *OR*
  - *Addition*
- Shift Left & Shift Right

# Further reading:

- Computer Science Illuminated
  - *Chapter 2*
- Foundation Maths
  - *Chapter 14*

© The University of Westminster (2020)

The right of Noam Weingarten to be identified as author of this work has been asserted by them in accordance with the Copyright, Designs and Patents Act 1988