COMPUTER SYSTEMS FUNDAMENTALS (4COSC004W)

In this lecture 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
- NOT(0) = 1
- NOT(1) = 0

NOT
1
0

NOT
0
1

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

Binary **AND** operation

- *a* AND *b* is True
  - *Iff both a AND b are True*

AND
0
0
0

AND
1
0
0

AND
0
1
0

AND
1
1
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* OR *b* is True
  - *If a OR b are True*
  - *Or a AND b are Ture*

OR
0
0
0

OR
1
0
1

OR
0
1
1

OR
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		

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

Binary addition

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

10
+
7

17

Binary addition

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

14
+
7

21

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 & Right shifting in Denary Burger analogy

- Consider a £10 Burger
  - *Probably a rather pleasant experience*

- Now perform a **Shift Right Denary** operation
- Consider a £1 Burger
  - $\frac{1}{10}$ *the price*
  - *Fast food*
  - *Not that good for your insides*

- Now perform a **Shift Left Denary** operation
- Consider a £100 Burger
  - *10-times the price*
  - *Lamb & mint with a side of Sweet Potato hand-cut chips and bottle of fancy red wine.*
  - *A very fancy experience*

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	1	*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 lecture we looked at:

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

Further reading:

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

Thank you