UNITED STATES AIR FORCE ACADEMY

ECE 281

Lesson 14 Notes

Objectives:

- Demonstrate ability to analyze and implement an ALU using a schematic diagram
- Given a hardware schematic, correctly identify a particular shifter or rotator
- Demonstrate how to multiply a number by 2^N using a LEFT bit shift of N bits
- Demonstrate how to divide a number by 2^N using a RIGHT bit shift of N bits
- Understand the difference between arithmetic and logical shifts and rotators

Arithmetic Logic Unit (ALU): component capable of performing various logic functions, like add, subtract, AND, OR, etc.

What are the engineering design considerations with an ALU vs individual components for the various logic functions?

Design Challenge Problem: build an ALU with the following functionality:

- Must be able to perform AND (both A AND B and A AND B')
- Must be able to perform OR (both A OR B and A OR B')
- Must be able to perform ADDITION (A + B)
- Must be able to perform SUBTRACTION (A – B)
- Must be able to perform Magnitude Comparison (Test if A < B)

Required Components: given the above requirements, what components (i.e. building blocks) will be required?

Control Signals			
F_2	F_1	F_0	Function
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Shifting: is a process used to reposition bits of data, either to the left or to the right. We will focus on logical and arithmetic shifts in this class.

Logical Shift:

Left Logical Shift:

Right Logical Shift:

Arithmetic Shift:

Left Arithmetic Shift:

Right Arithmetic Shift:

What is the significance of the Right Arithmetic Shift?

Rotator: similar to shifter, but the bit that falls off one end attaches to the other end.

Notation: 10110001 in 2's complement is:

Function	Input		Bits	Output	Decimal
Left Logical Shift	**10110001**	**<<**	**1**		
Left Arithmetic Shift	**10110001**	**<<<**	**1**		
Left Rotator	**10110001**	**ROL**	**1**		
Right Logical Shift	**10110001**	**>>**	**1**		
Right Arithmetic Shift	**10110001**	**>>>**	**1**		
Right Rotator	**10110001**	**ROR**	**1**		

Function	Input		Bits	Output	Decimal
Left Logical Shift	**10110001**	**<<**	**2**		
Left Arithmetic Shift	**10110001**	**<<<**	**2**		
Left Rotator	**10110001**	**ROL**	**2**		
Right Logical Shift	**10110001**	**>>**	**2**		
Right Arithmetic Shift	**10110001**	**>>>**	**2**		
Right Rotator	**10110001**	**ROR**	**2**		

Function	Input		Bits	Output	Decimal
Left Logical Shift	**10110001**	**<<**	**3**		
Left Arithmetic Shift	**10110001**	**<<<**	**3**		
Left Rotator	**10110001**	**ROL**	**3**		
Right Logical Shift	**10110001**	**>>**	**3**		
Right Arithmetic Shift	**10110001**	**>>>**	**3**		
Right Rotator	**10110001**	**ROR**	**3**		