Lab #5 - Arithmetic Logic Unit – ALU

Isai Eliseo Mercado Oliveros

Oct 8, 2014

Procedure

Full Adder and 4:1 Mux Verilog codes

```
//
//////////////////////////////////////
module FullAdder(A,B,Cin,Cout,Sum);
   input A,B,Cin;
   output Cout,Sum;
   wire c0,c1,c2;

   xor(Sum, A,B,Cin);
   and(c0,A,B);
   and(c1,A,Cin);
   and(c2,B,Cin);
   or(Cout,c0,c1,c2);

endmodule
```

```
////////////////////////////////////////////////////////
module Mux4to1(Out,A,B,C,D,Sel1,Sel2);
   input A,B,C,D,Sel1,Sel2;
   output Out;

   assign Out = (Sel1 == 0 && Sel2 == 0)? A :
                (Sel1 == 0 && Sel2 == 1)? B :
                (Sel1 == 1 && Sel2 == 0)? C :
                (Sel1 == 1 && Sel2 == 1)? D : Out;

endmodule
```

Full Adder and 4:1 Mux TCL files

FullAdder.tcl - Notepad

File Edit Format View Help

```
#add all signals to the waveform viewer
wave add / -radix hex



#define how the data input signals will behave when you run the simulation
#the command "isim force add (signal) 0 -time 0 -value 1 -time 20ns -repeat 40ns" means that (signal)
#will have a value of 0 from time 0, then change to 1 at time 20ns, and then repeat that cycle every 40ns
isim force add B 0 -time 0 -value 1 -time 10ns -repeat 20ns
isim force add A 0 -time 0 -value 1 -time 20ns -repeat 40ns
isim force add Cin 0 -time 0 -value 1 -time 40ns -repeat 80ns

#Nothing will change in the waveform viewer until you run the simulation for some period of time.
run 80ns
```

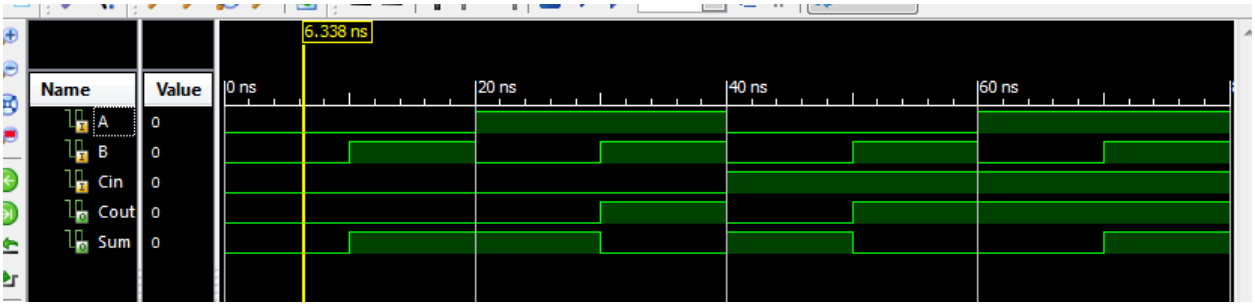
Mux4to1.tcl - Notepad

File Edit Format View Help

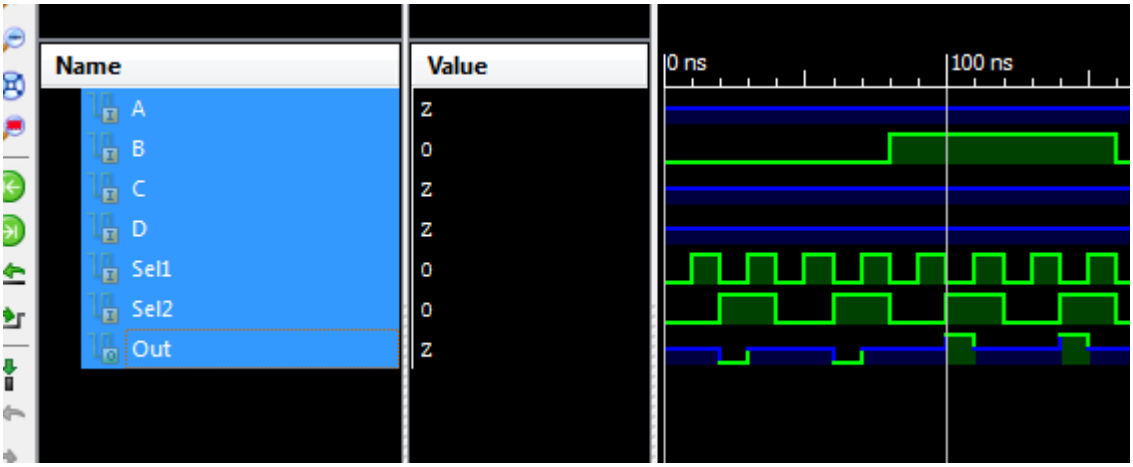
```
 #add all signals to the waveform viewer
wave add / -radix hex



#define how the data input signals will behave when you run the simulation
#the command "isim force add (signal) 0 -time 0 -value 1 -time 20ns -repeat 40ns" means that (signal)
#will have a value of 0 from time 0, then change to 1 at time 20ns, and then repeat that cycle every 40ns
isim force add Sel1 0 -time 0 -value 1 -time 10ns -repeat 20ns
isim force add Sel2 0 -time 0 -value 1 -time 20ns -repeat 40ns
isim force add A 0 -time 0 -value 1 -time 40ns -repeat 80ns
isim force add B 0 -time 0 -value 1 -time 80ns -repeat 160ns
isim force add C 0 -time 0 -value 1 -time 40ns -repeat 320ns
isim force add D 0 -time 0 -value 1 -time 80ns -repeat 640ns

#Nothing will change in the waveform viewer until you run the simulation for some period of time.
run 640ns
```

Full Adder and 4:1 Mux simulation waveforms

Adder



Mux



1 bit ALU Verilog code

```
20  //////////////////////////////////////////////////////////////////////////////////
21  module ALU1bit(Result,Cout,A,B,Cin,Sel1,Sel2);
22     input A,B,Cin,Sel1,Sel2;
23     output Result,Cout;
24     wire Sum,Aout,AandB,notA,CarryOut;
25
26     CarryControl CC(Sel1,Sel2,CarryOut,Cout); //module CarryControl(Sel1,Sel2,Cin,Cout);
27     FullAdder FA(A,B,Cin,CarryOut,Sum); //module FullAdder(A,B,Cin,Cout,Sum);
28     OtherLogic OL(Aout,AandB,notA,A,B); //module OtherLogic(Aout,AandB,notA,A,B);
29     Mux4to1 FO(Result,Aout,Sum,AandB,notA,Sel1,Sel2); //module Mux4to1(Out,A,B,C,D,Sel1,Sel2);
30  endmodule
31
```

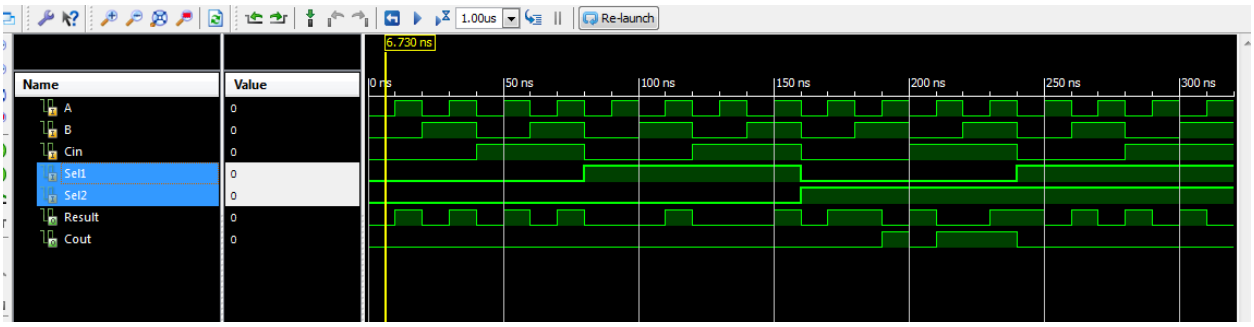
1 bit ALU TCL file

ALU1bit.tcl - Notepad

File Edit Format View Help

```
#add all signals to the waveform viewer
wave add / -radix hex

#define how the data input signals will behave when you run the simulation
#the command "isim force add (signal) 0 -time 0 -value 1 -time 20ns -repeat 40ns" means that (signal)
#will have a value of 0 from time 0, then change to 1 at time 20ns, and then repeat that cycle every 40ns
isim force add A 0 -time 0 -value 1 -time 10ns -repeat 20ns
isim force add B 0 -time 0 -value 1 -time 20ns -repeat 40ns
isim force add Cin 0 -time 0 -value 1 -time 40ns -repeat 80ns
isim force add Sel1 0 -time 0 -value 1 -time 80ns -repeat 160ns
isim force add Sel2 0 -time 0 -value 1 -time 160ns -repeat 320ns

#Nothing will change in the waveform viewer until you run the simulation for some period of time.
run 320ns
```

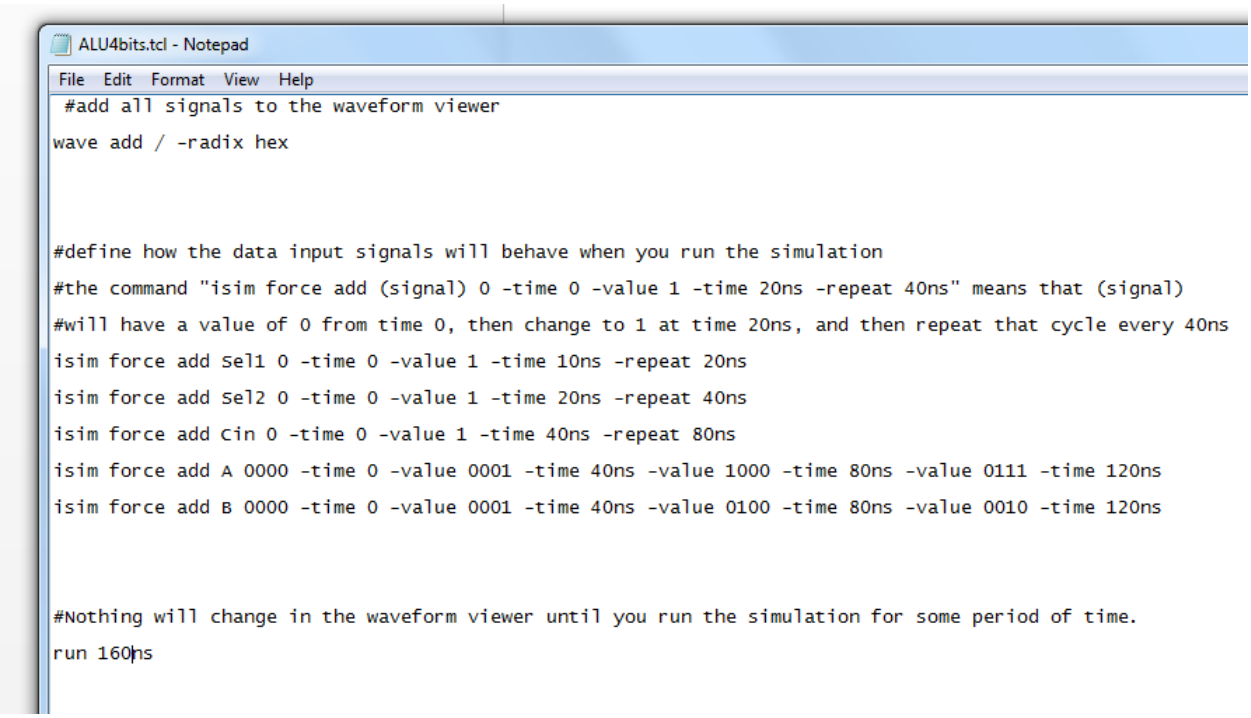
1 bit ALU simulation waveform



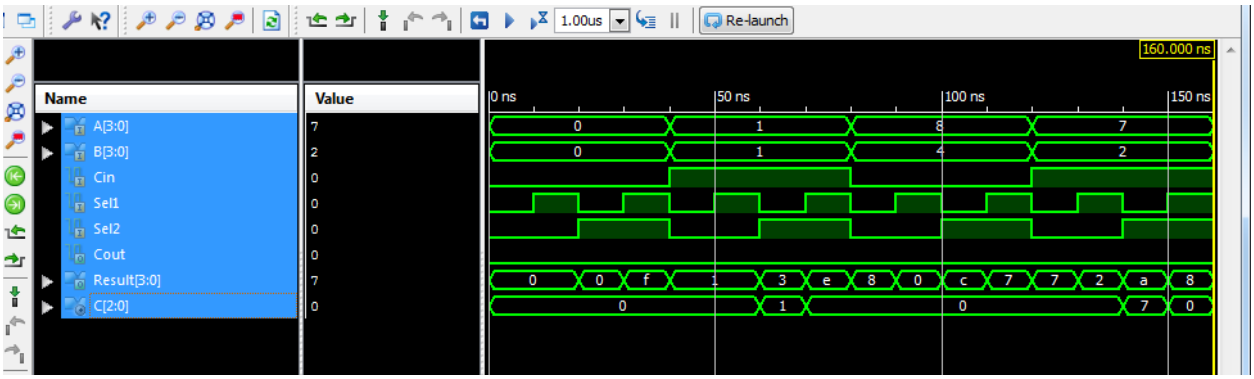
4 bit ALU Verilog code

```
19  //
20  //////////////////////////////////////////////////////////////////////////////////
21  module ALU4bits(A,B,Cin,Cout,Result,Sel1,Sel2);
22     input[3:0] A;
23     input[3:0] B;
24     input Cin,Sel1,Sel2;
25     output Cout;
26     output[3:0] Result;
27     wire[2:0] C;
28     ALU1bit a1(Result[0],C[0],A[0],B[0],Cin,Sel1,Sel2);//module ALU1bit(Result,Cout,A,B,Cin,Sel1,Sel2);
29     ALU1bit a2(Result[1],C[1],A[1],B[1],C[0],Sel1,Sel2);
30     ALU1bit a3(Result[2],C[2],A[2],B[2],C[1],Sel1,Sel2);
31     ALU1bit a4(Result[3],Cout,A[3],B[3],C[2],Sel1,Sel2);
32  endmodule
33
```

4 bit ALU TCL file



4 bit ALU simulation waveform



4 bit ALU UCF file

Anomalies (bugs, problems, and suggestion)

I did not have problems and the TAs were really helpful.