Name: **Gabriel Vega, Anthony Lawlor**

EECE.2650 Logic Design

Assignment 5

Design of Arithmetic Logic Unit

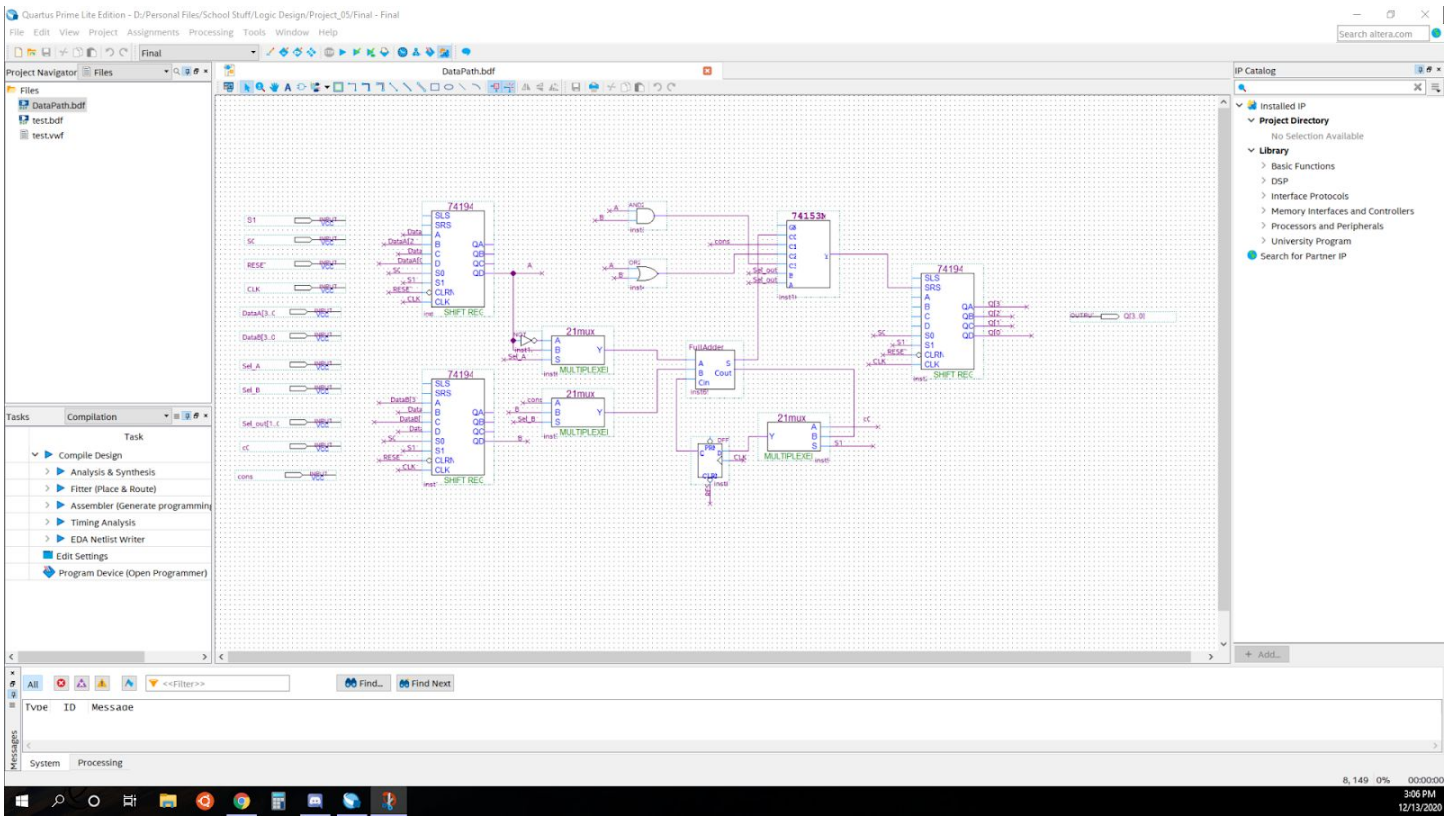
Datapath Module

Data Path Table: Use this table to determine which signals go where for each instruction. You can then use this table to wire your datapath so that the muxes route the correct signals

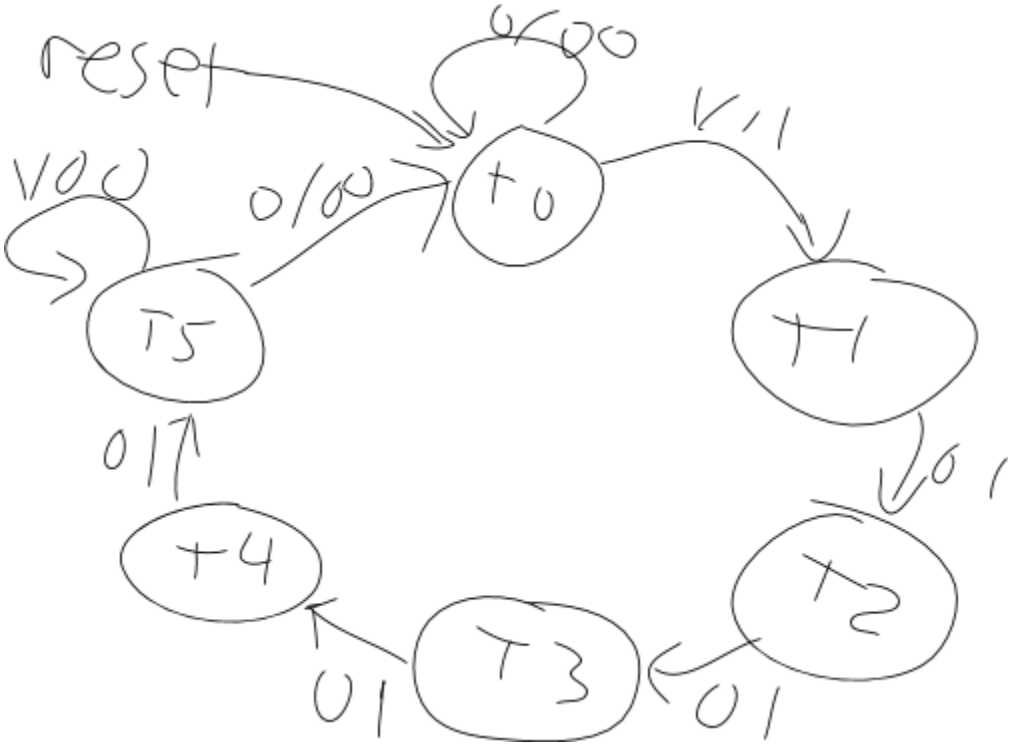
The first instruction is filled out for you. A + B uses the full adder for addition (c0 = 0) the inputs to the full adder xi and yi get Ai and Bi respectively.

OpCode	Instruction	xi	yi	c0	Output Selector
0 0 0	Add A + B	Ai	Bi	0	Full Adder
0 0 1	Increment A	Ai	0	1	Full Adder
0 1 0	Invert A	~Ai	0	0	Full Adder
0 1 1	Subtract B - A	~Ai	Bi	1	Full Adder
1 0 0	Decrement A	Ai	1	0	Full Adder
1 0 1	A | B	Ai	Bi	D	OR Gate
1 1 0	A & B	Ai	Bi	D	AND Gate
1 1 1	Negative A	~Ai	0	1	Full Adder

Completed Datapath:



State Diagram for the State Generator:



Transition Table for the State Generator:

Present State	START	Next State	s1 s0
T0	0	T0	00
T0	1	T1	11
T1	d	T2	01
T2	d	T3	01
T3	d	T4	01
T4	d	T5	01
T5	0	T0	00
T5	1	T5	00

Using the table above, find the Next State Equations for T0-T5 and equations for s1 and s0:

T0+ = (T0 * !start) + (T5 * !start)

T1+ = (T0 * start)

T2+ = T1

T3+ = T2

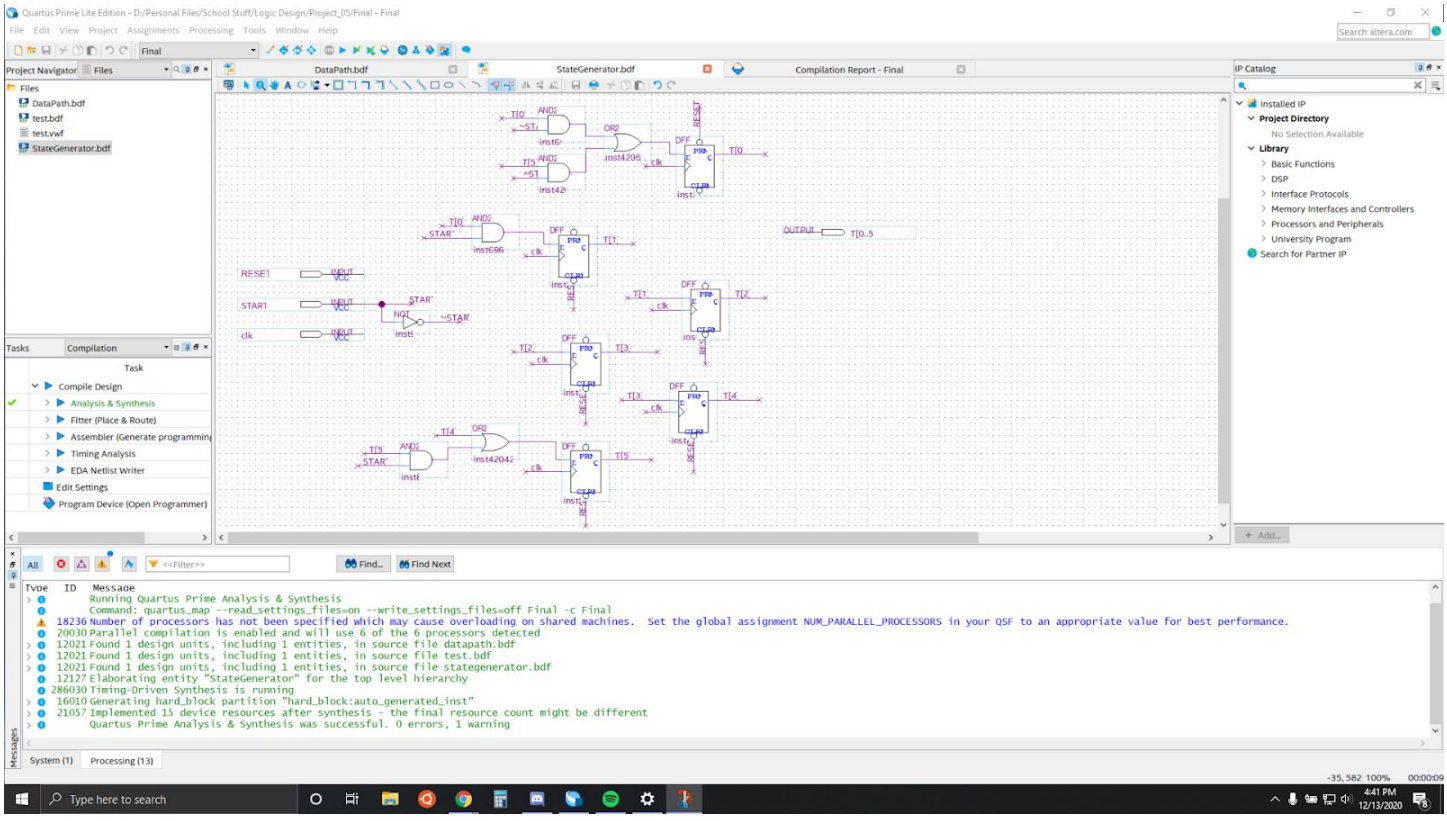
T4+ = T3

T5+ = T4 + (T5 * start)

S1 = T0 * start

S0 = !T5 * !(T0 * !start)

Block Diagram for state generator:



Control Circuit Truth Table:

OpCode	Instruction	Sel_A	Sel_B	Sel_out[1..0]	c0	Const
0 0 0	Add A + B	0	0	00	0	d
0 0 1	Increment A	0	1	00	1	0
0 1 0	Invert A	1	1	00	0	0
0 1 1	Subtract B - A	1	0	00	1	d
1 0 0	Decrement A	0	1	00	0	1
1 0 1	A | B	d	d	10	D	d
1 1 0	A & B	d	d	11	D	d
1 1 1	Negative A	1	1	00	1	0

Equations for each control signal: (You may choose to use a KMAP for each signal)

OP1 = A, OP2 = B, OP3 = C

Sel_A

OP1/ OP2,3	00	01	11	10
0	0	0	1	1
1	0	0	1	0

Sel_B

OP1/ OP2,3	00	01	11	10
0	0	1	0	1
1	1	0	1	0

c0

OP1/ OP2,3	00	01	11	10
0	0	1	1	0
1	0	0	1	0

Sel_A = B

Sel_B = A ⊕ B ⊕ C

c0 = C

const

OP1/ OP2,3	00	01	11	10
0	d	0	d	0
1	1	d	0	d

Sel_out[0]

OP1/ OP2,3	00	01	11	10
0	0	0	0	0
1	0	0	0	1

Sel_out[1]

OP1/ OP2,3	00	01	11	10
0	0	0	0	0
1	0	1	0	1

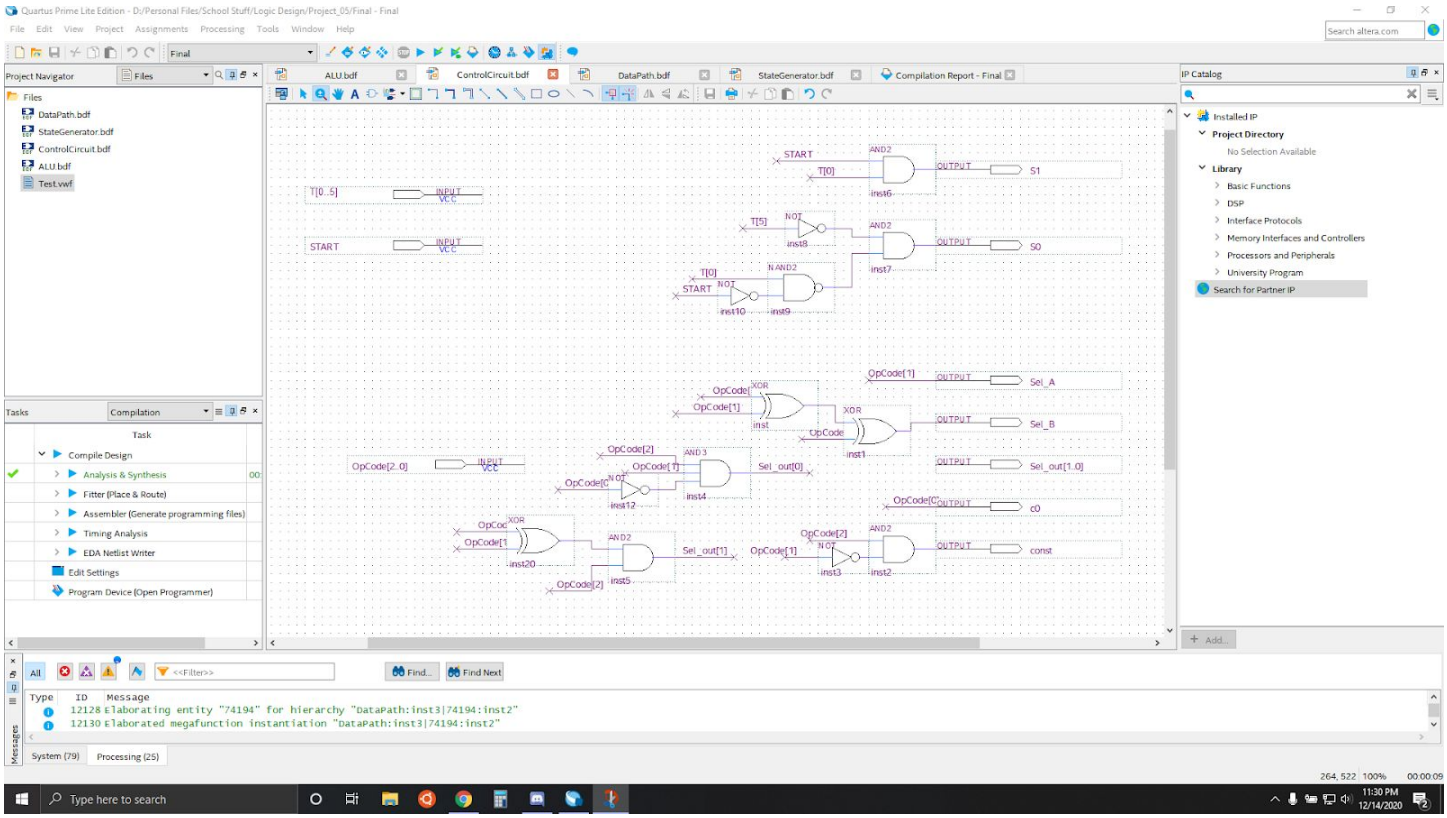
const = A!B

Sel_out[0] = AB!C

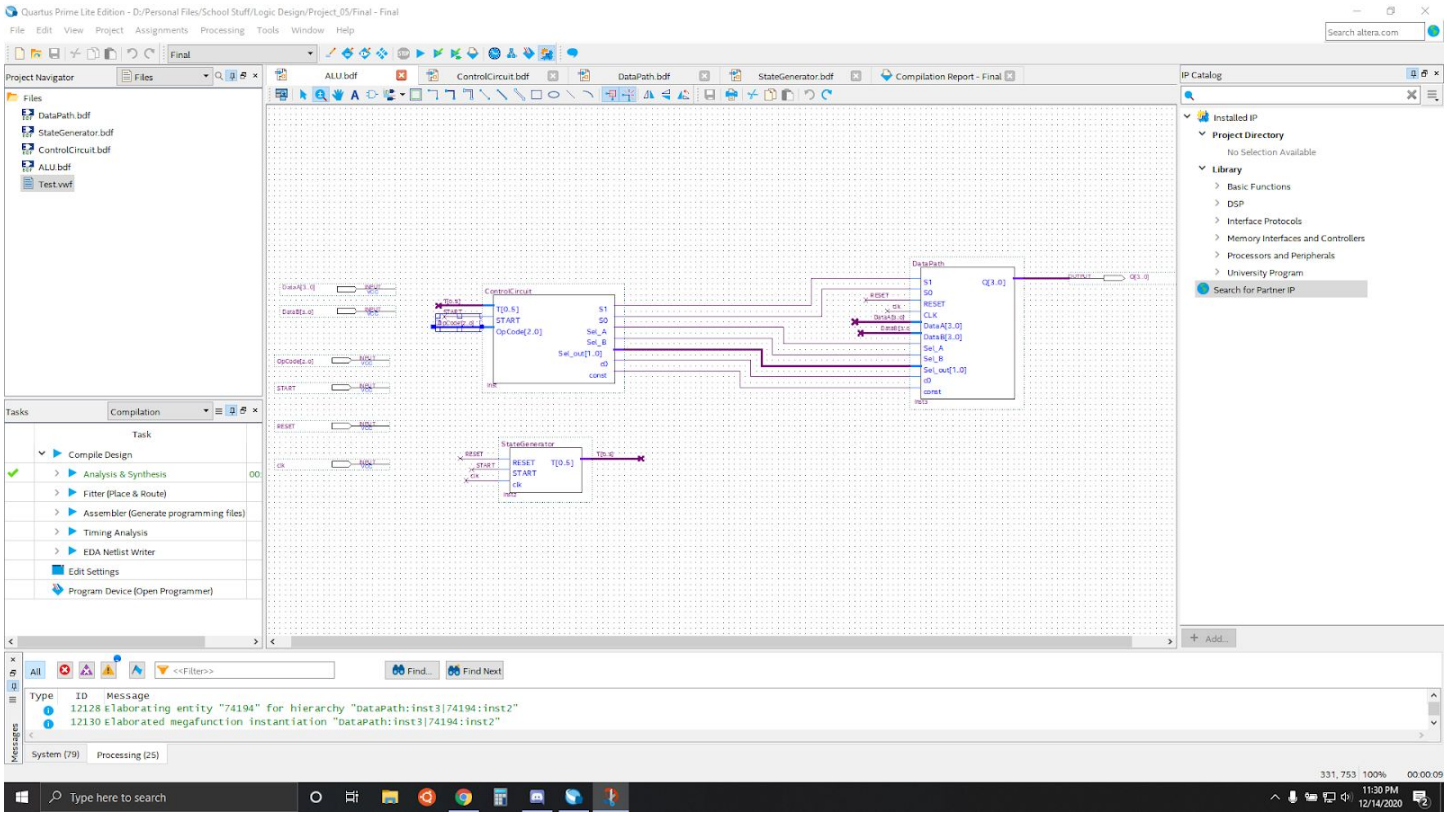
Sel_out[1] = A!BC + AB!C

A(B⊕C)

Block Diagram for Control Circuit:



Block Diagram for Completed Circuit:

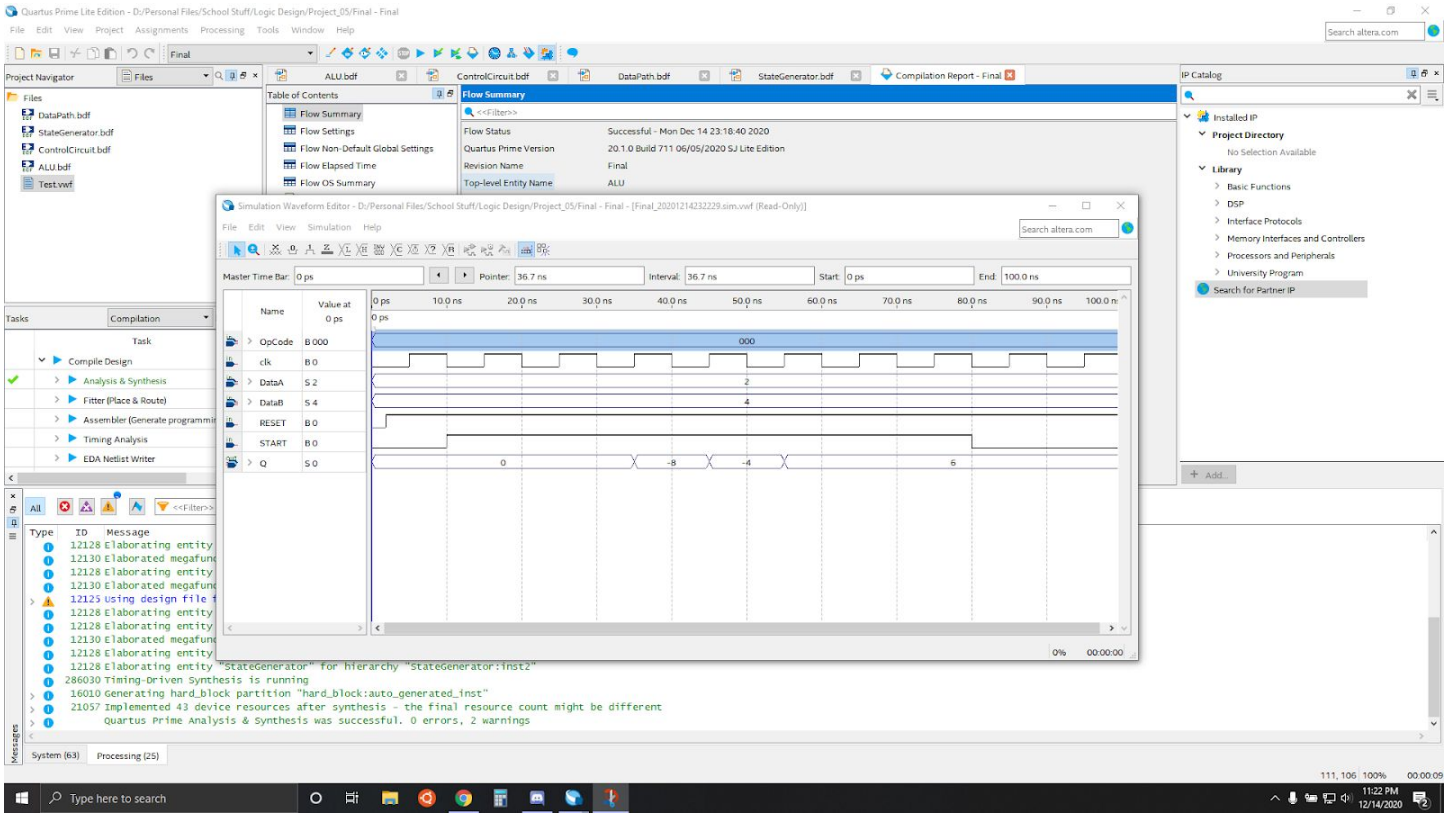


Testing of ALU:

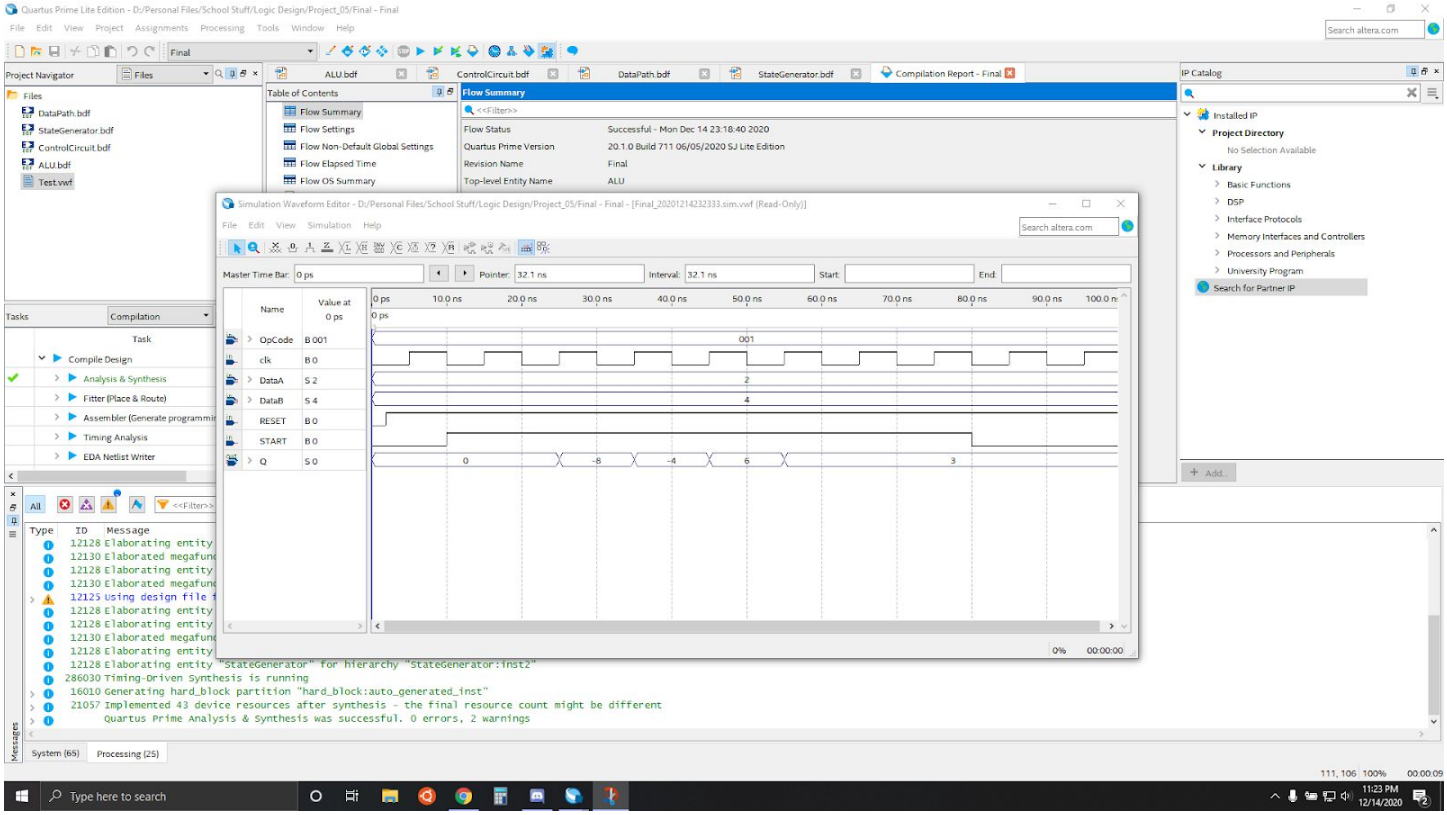
Paste images showing each OpCode being performed successfully below. Be sure to label each image with the Opcode being used and its instruction.

Example:

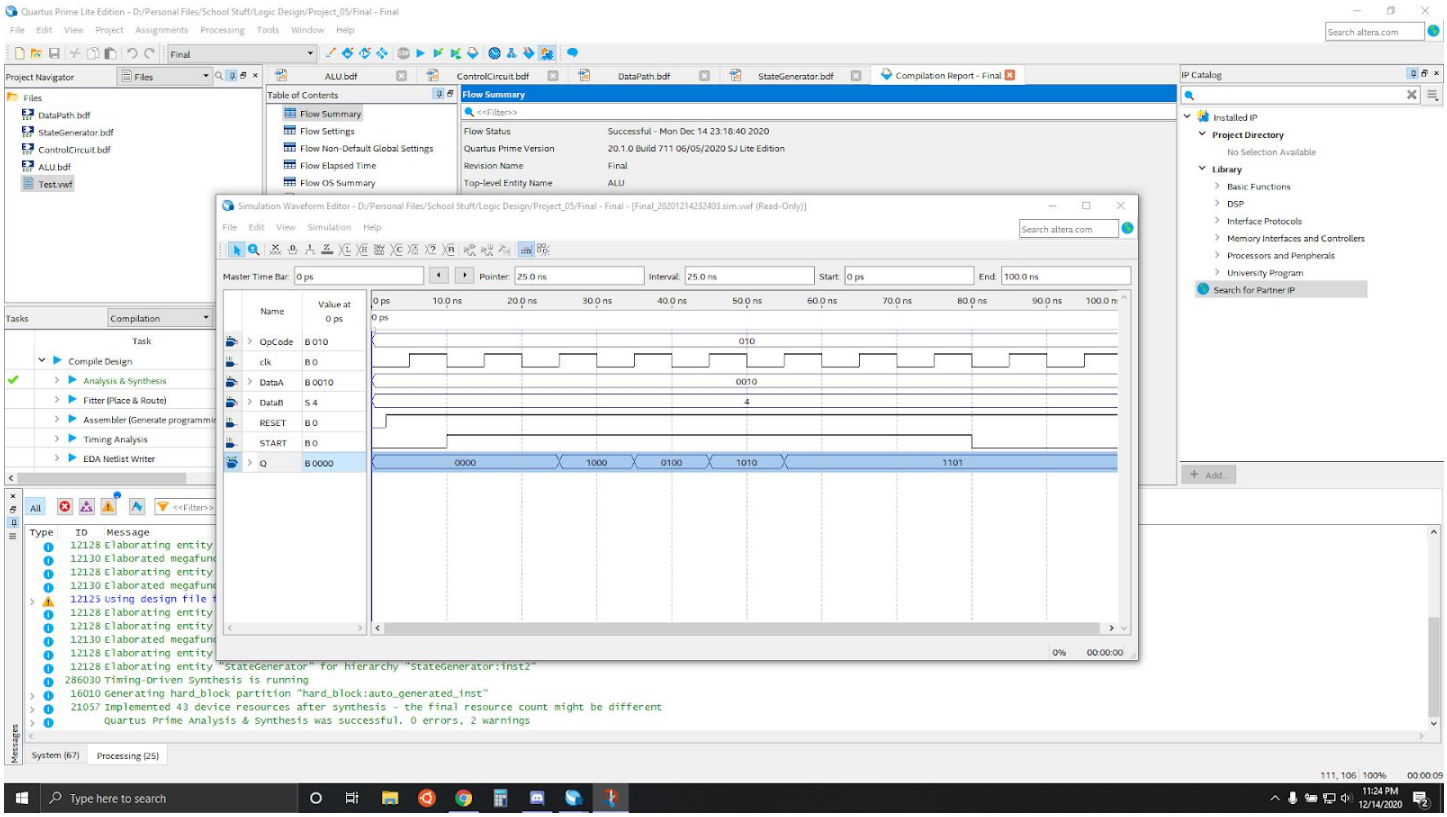
Simulation for OpCode 000, A+B:



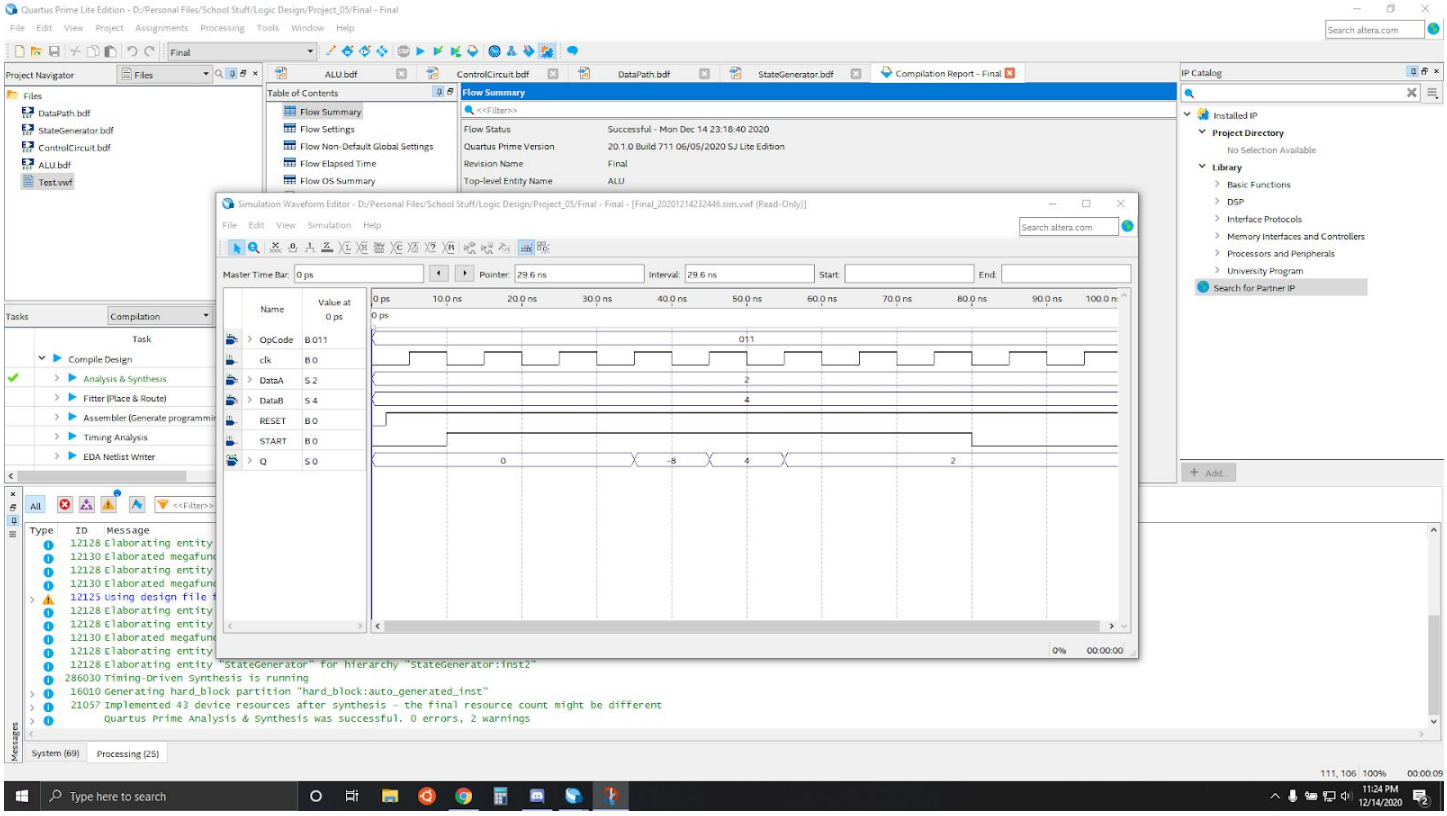
Simulation for OpCode 001, Increment A:



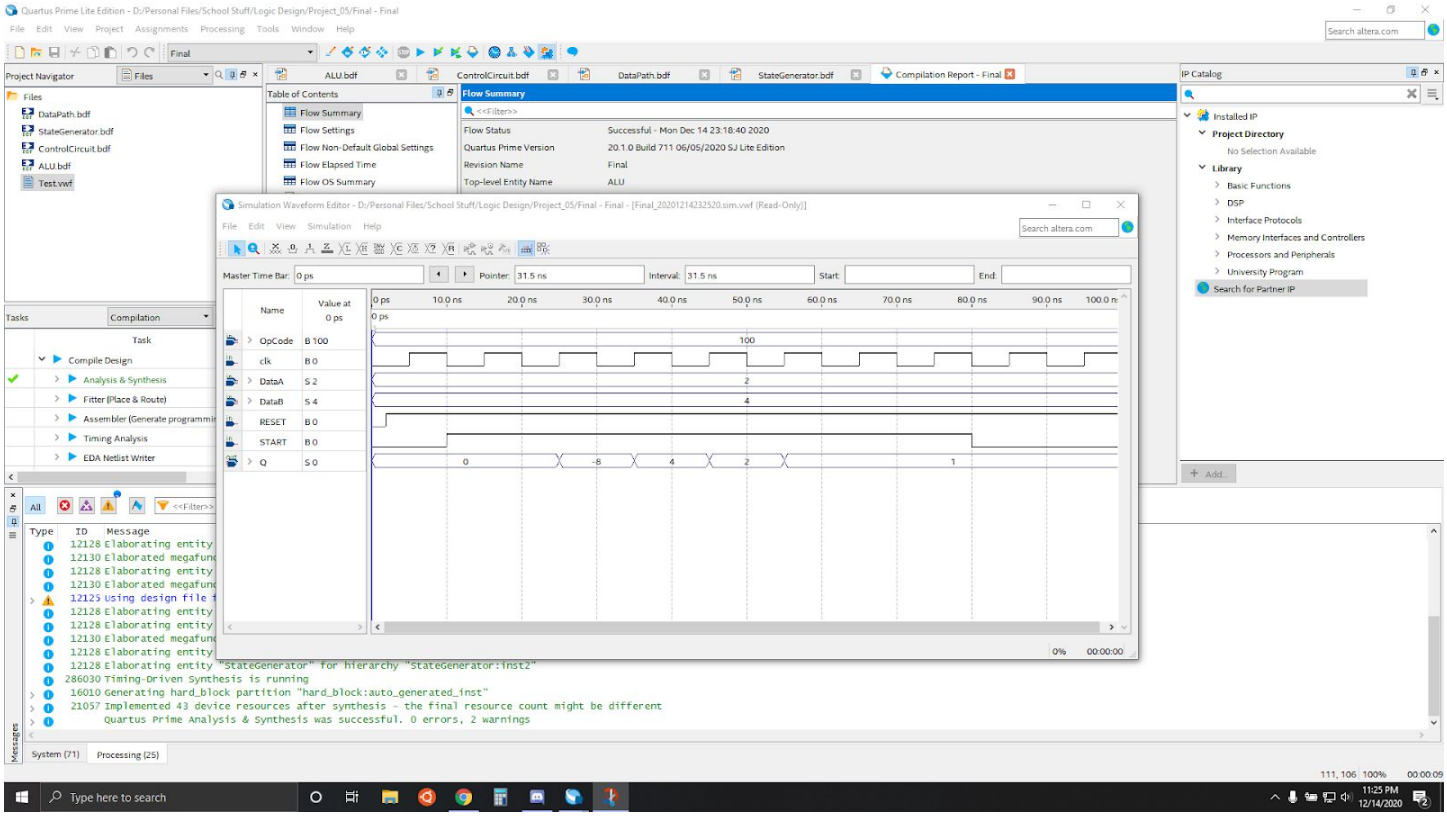
Simulation for OpCode 010, Invert A:



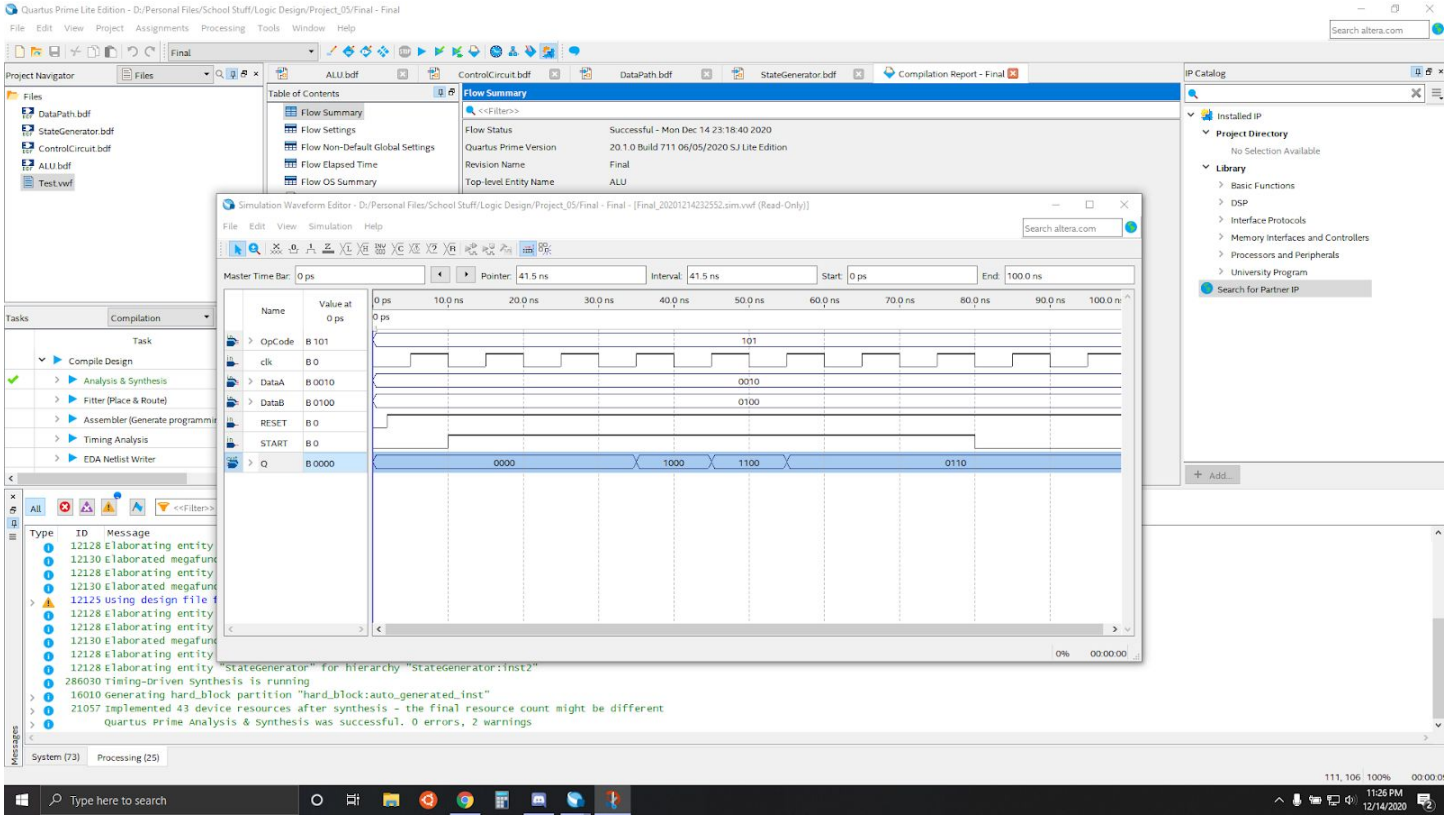
Simulation for OpCode 011, Subtract B - A:



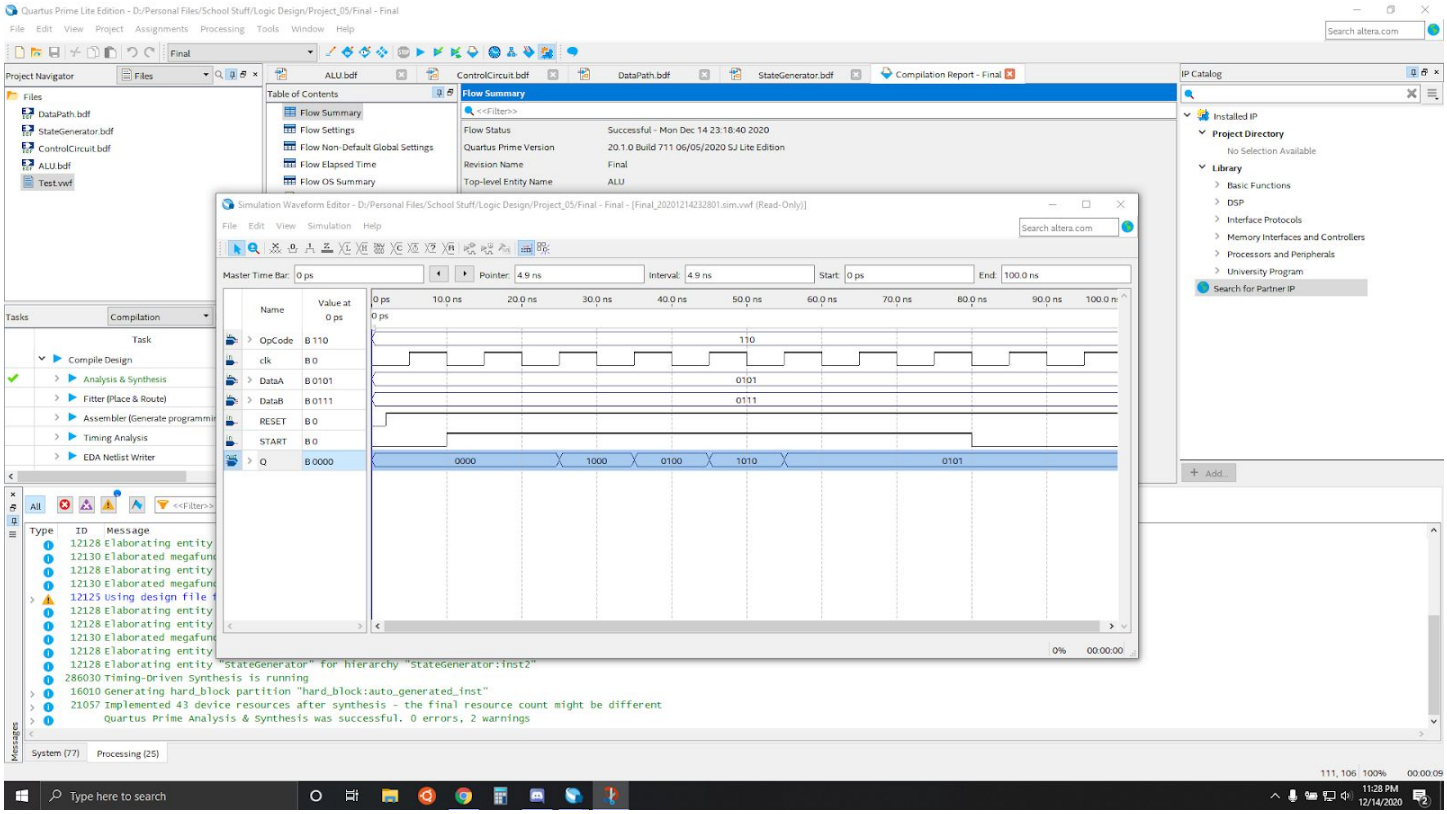
Simulation for OpCode 100, Decrement A:



Simulation for OpCode 101, A | B:



Simulation for OpCode 110, A & B:



Simulation for OpCode 111, Negative A: