Logic Design

Project Report

Mohamed Adel Mohamed Hamzawi-1190500

Mahmoud NejmEldin Aly Yassin-1190326

Mohamed Hussien Ragab Hussien-1190566

Abdelrahman Hamdy Imam-1190345

Work Distribution:

Mohamed Adel Hamzawi:

1)Move 2)And 3)Not 4)Shift right

Mohamed hussien:

1)Or 2)Xor 3)Shift left

Mahmoud NajmEldien:

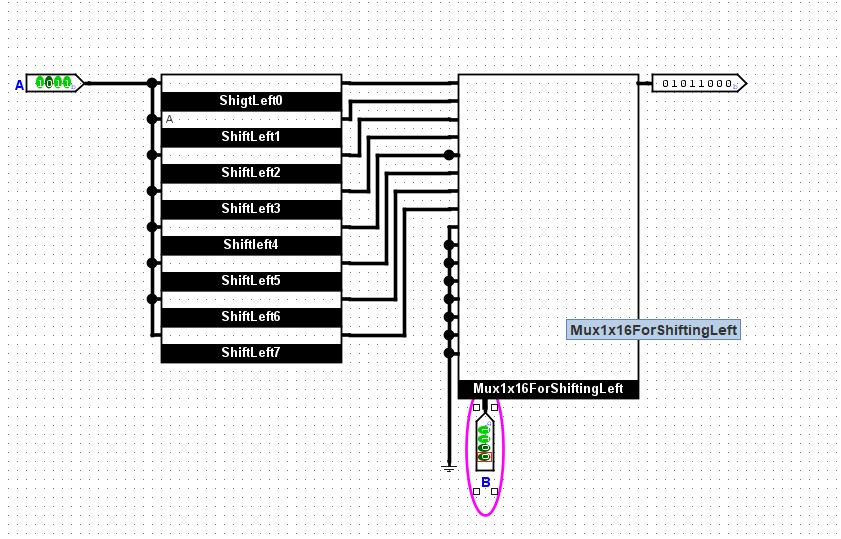
1)Negative 2)Multiplyer

Abdelrahman Hamdy:

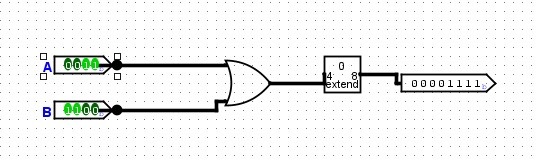
1)Adder 2)Subtracter

The rest of the efforts is shared between group members equally

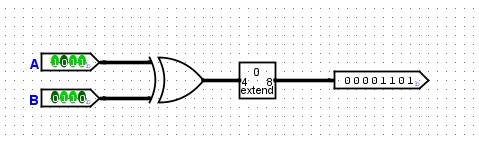
Shift left



Or

****

Xor

****

**Number Of Gates :**

1 – Shifting Left : 0 Gate

2- Or : 4 Gates Or

3- Xor : 4 Gates Xor

Multiplixer 16x1 : 30 And Gates and Gates 15 Or Gates Totally **45 Gates**. (For Mux 8x1 (14 And Gates and 7 Or Gates ) 21 Gates –for Mux 4x1 (6 And Gates and 3 Or gates )9 Gates –for Mux 2X1(2 And Gates and 1 Or Gate ) 3 Gates) .

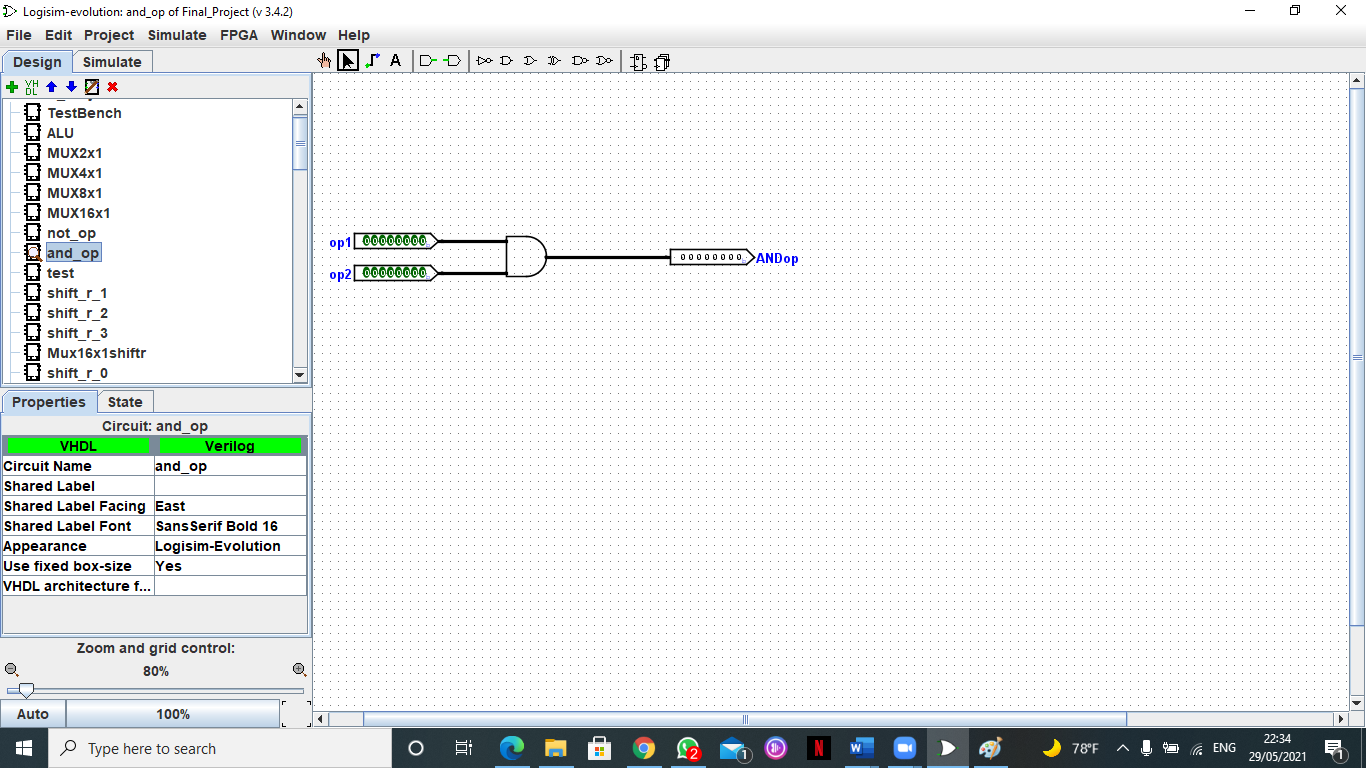
**Brief Description :**

1- Shifting Left : For every Shift , I used two Splitters and Grounds one splitter to split the input and the the second one to collect the input with the Grounds Expect (the 8th Shiftting I only used 8 bits Ground).

2- OR : I used only 1 (4-bits, 2 inputs) **OR** Gate with a **Bit Extender**.

3-XOR : I used only 1 (4-bits, 2 inputs) X**OR** Gate with a **Bit Extender.**

And gate:



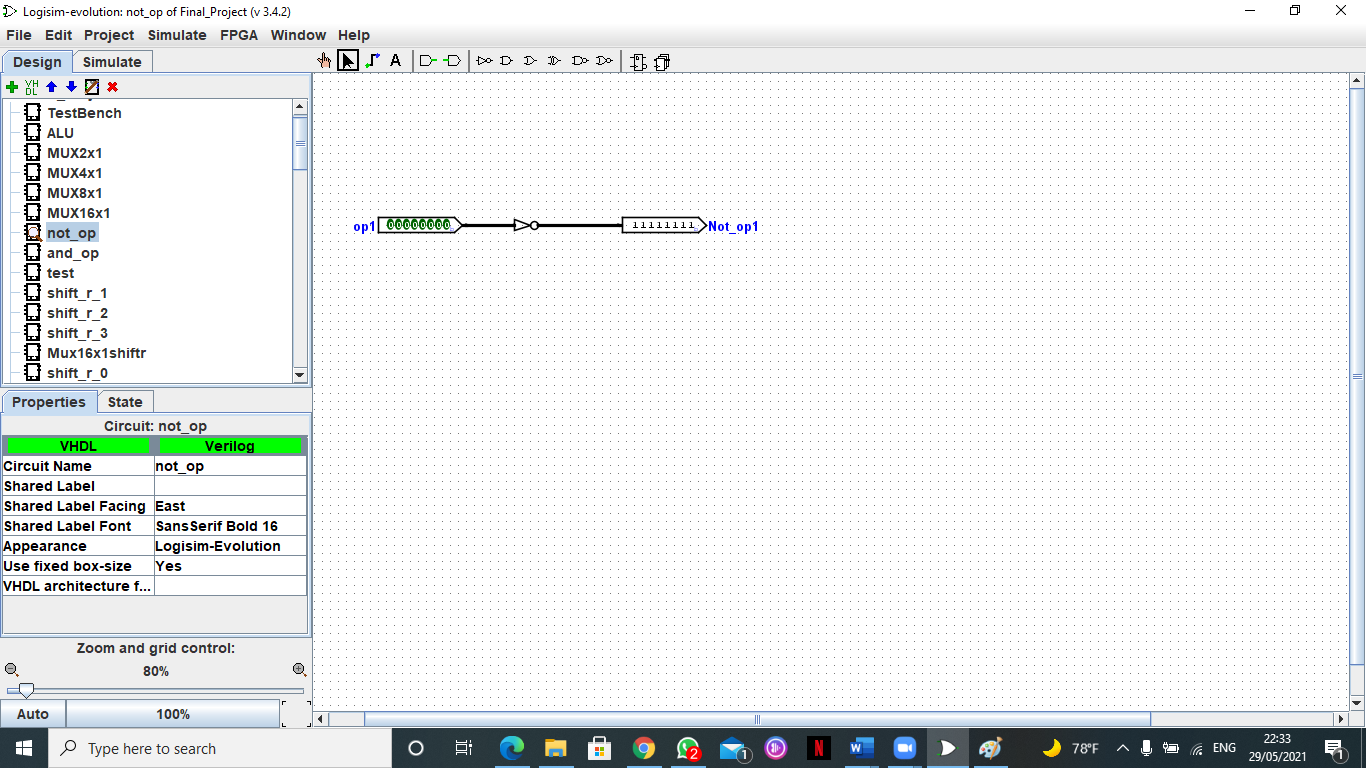
Gates:

8 AND gate

Description:

2 inputs 8 bits each are input to an AND gate

Not:



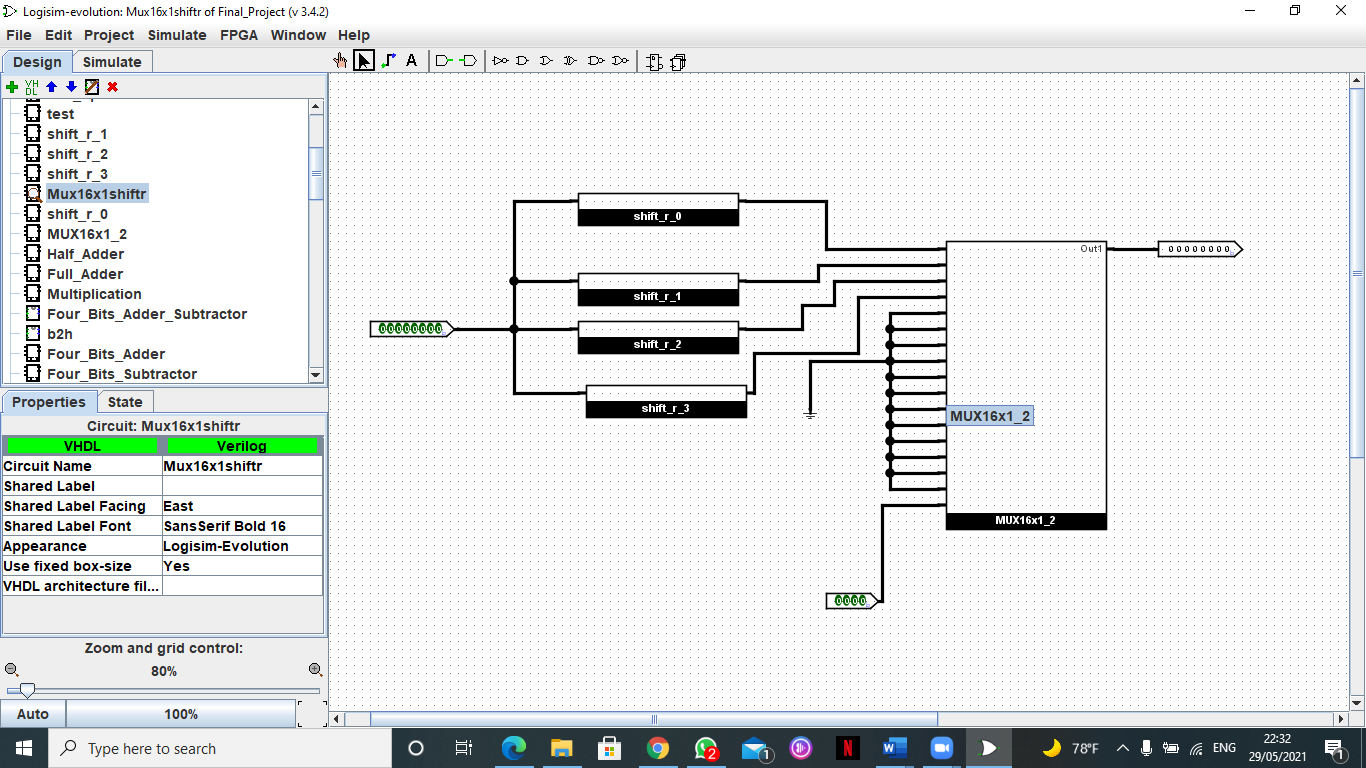
Gates:

8 NOT gates

Description:

1 8-bits input enters a not gate

Shift Right:



Gates:

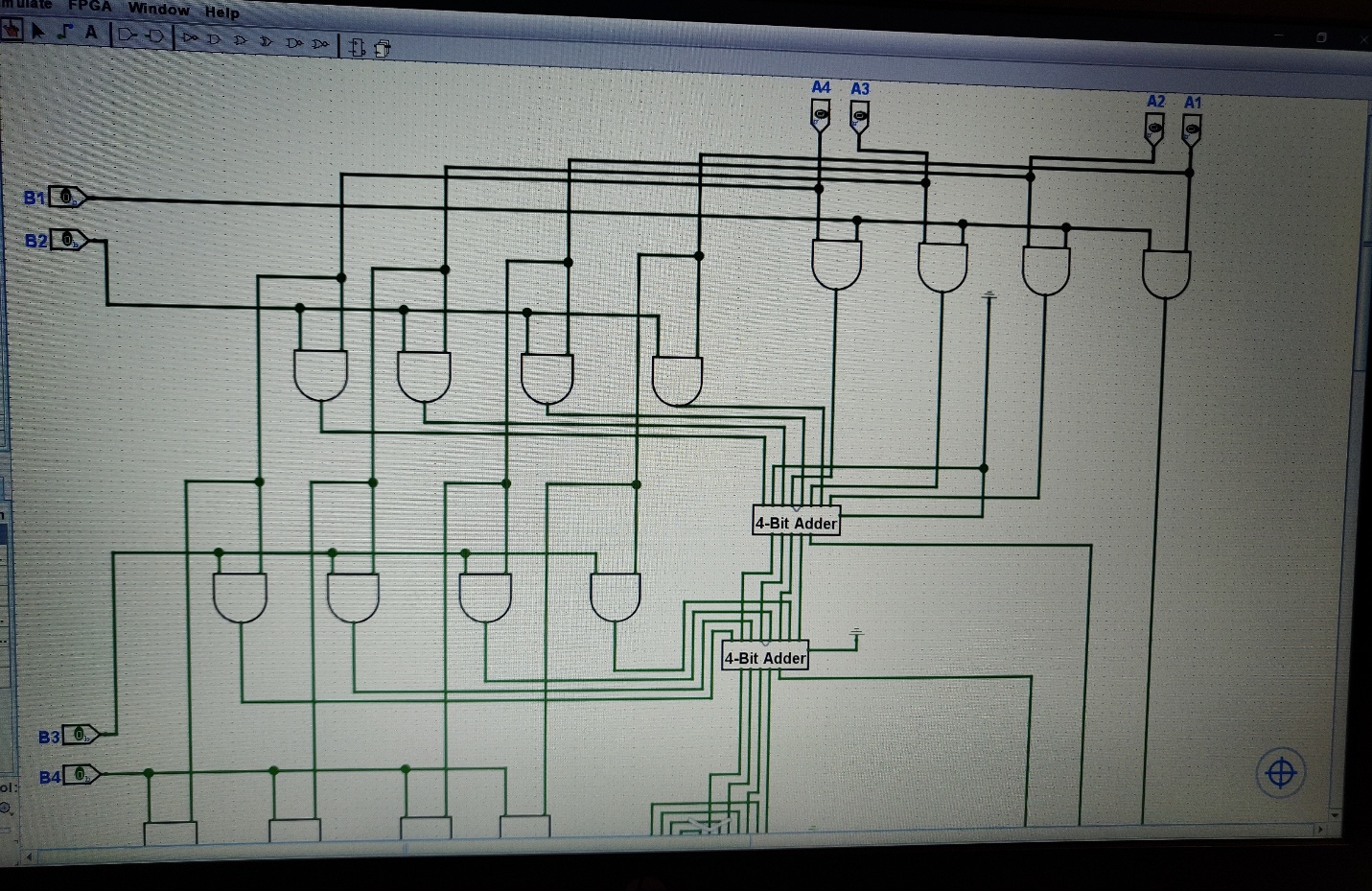
Total no. of gates = No. of gates in Multiplexer 16\*1

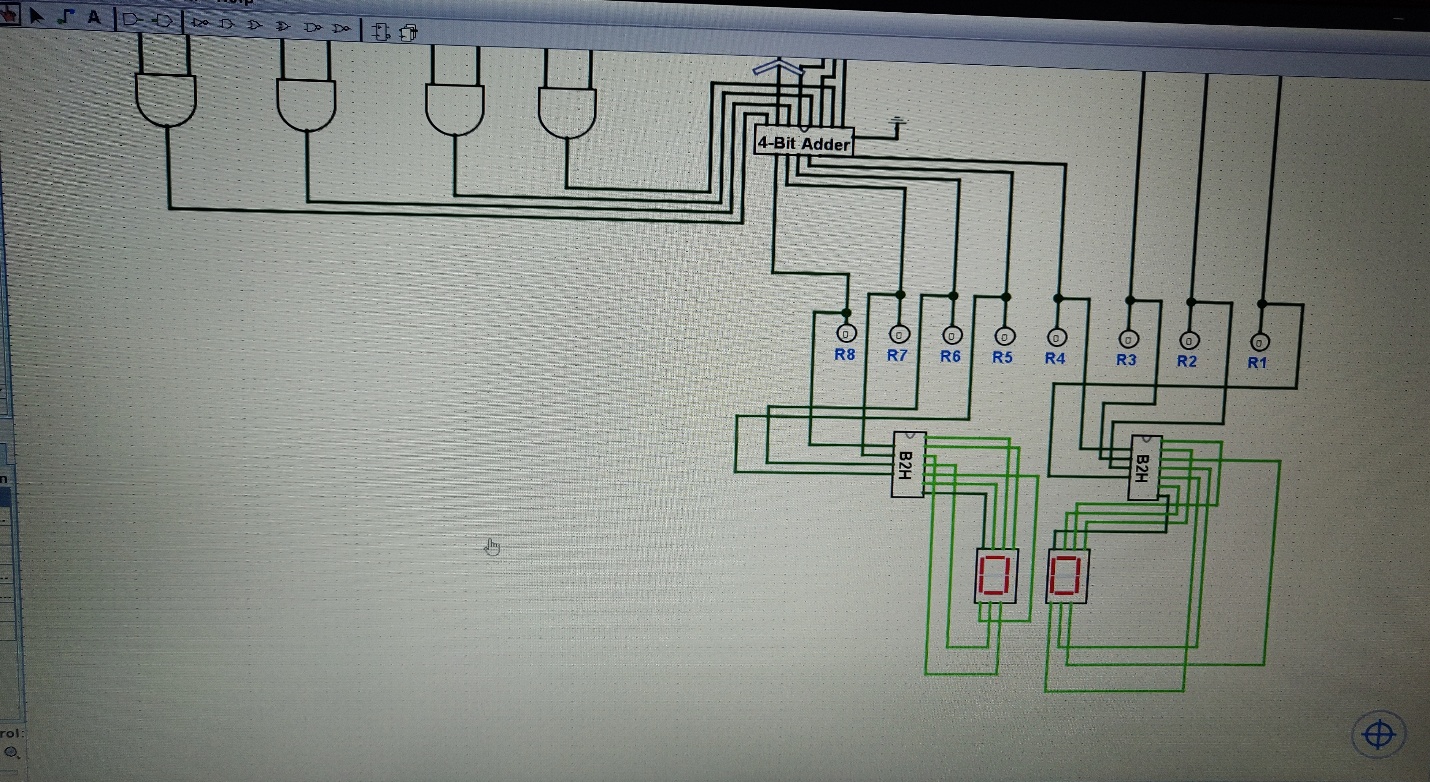
=30\*8 AND gates+ 15\*8 OR gates = 240 AND gate + 120 OR gate

Description:

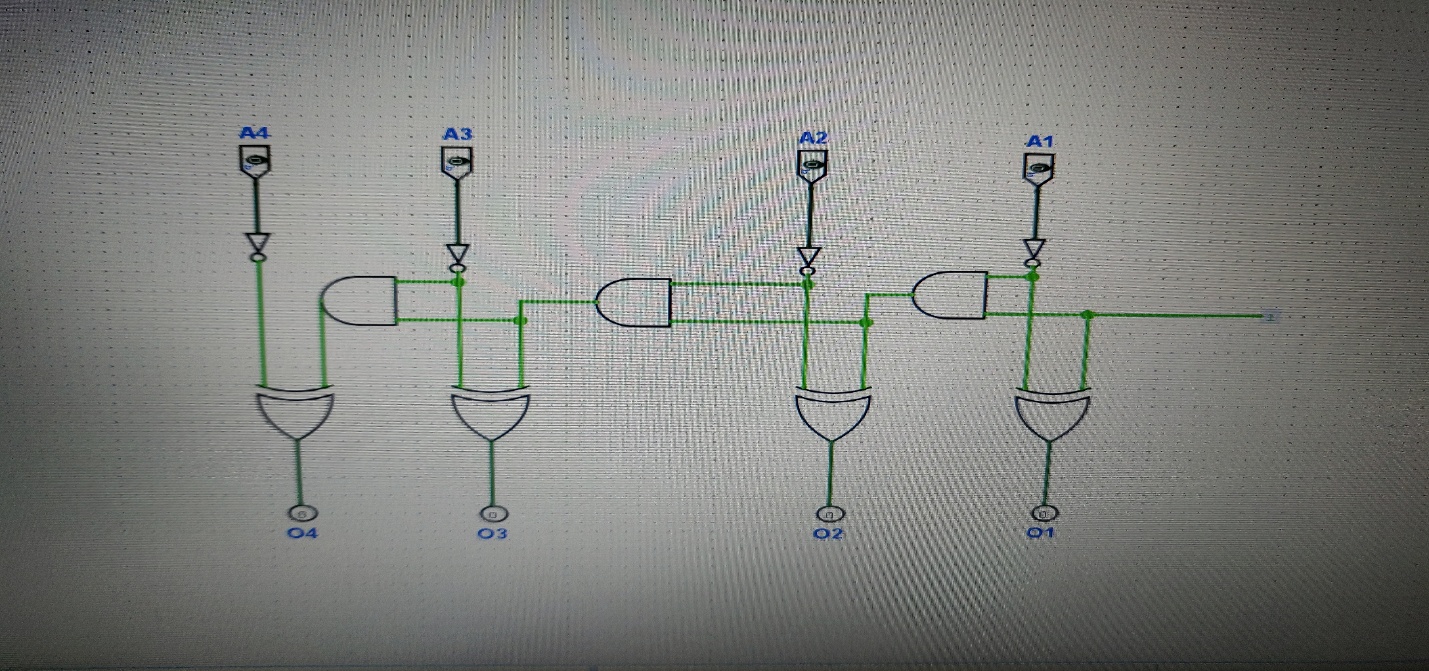
4 circuits are designed for right shift by 0,1,2 and 3 with 1 input using splitter

The 4 circuits are combined together using a 16\*1 Multiplexer with 2 inputs ( The operand(8bits) and the selector(4bits)), the operand is input into each circuit and their output is input in the multiplexer.

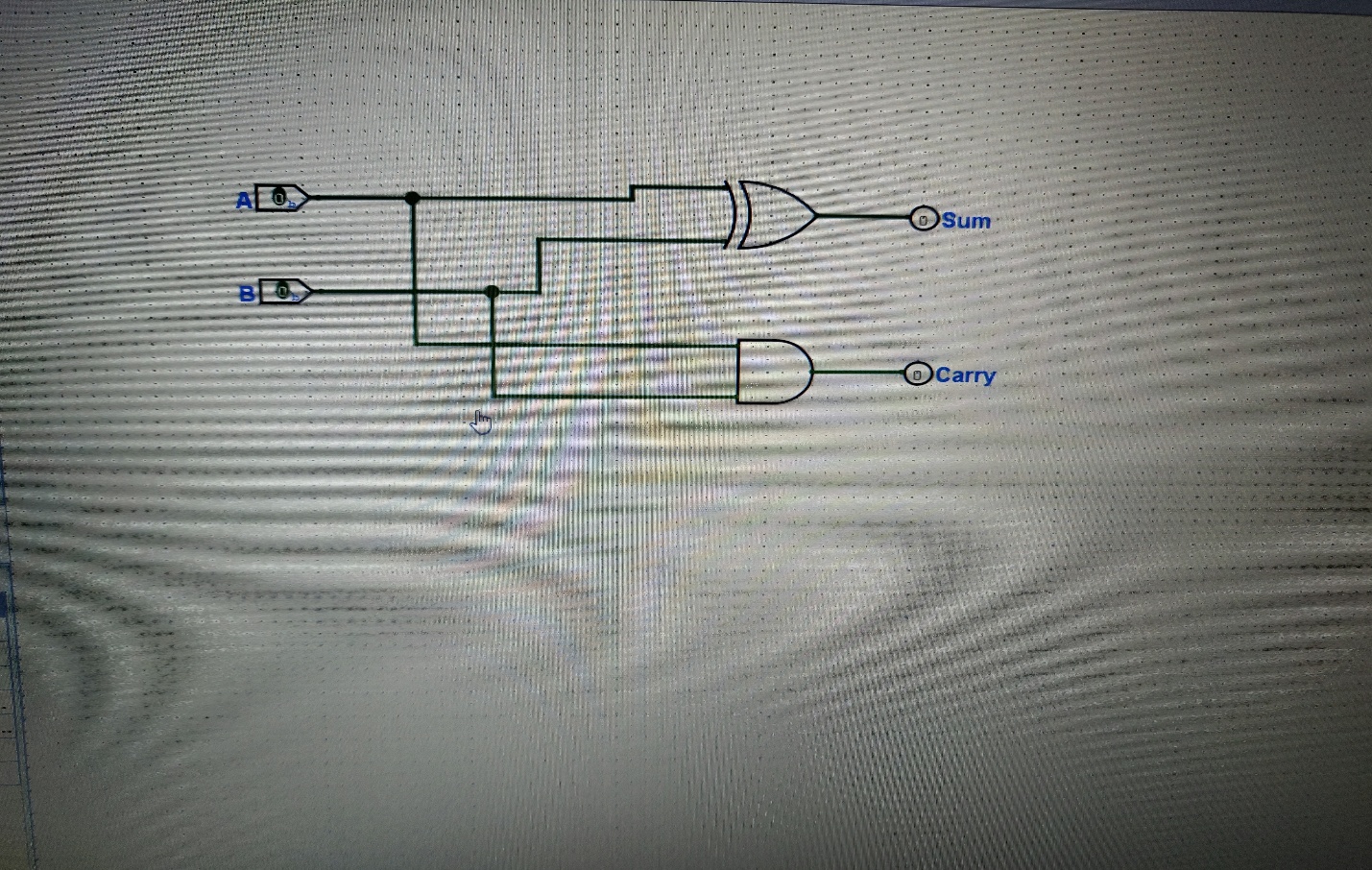
Multiplication

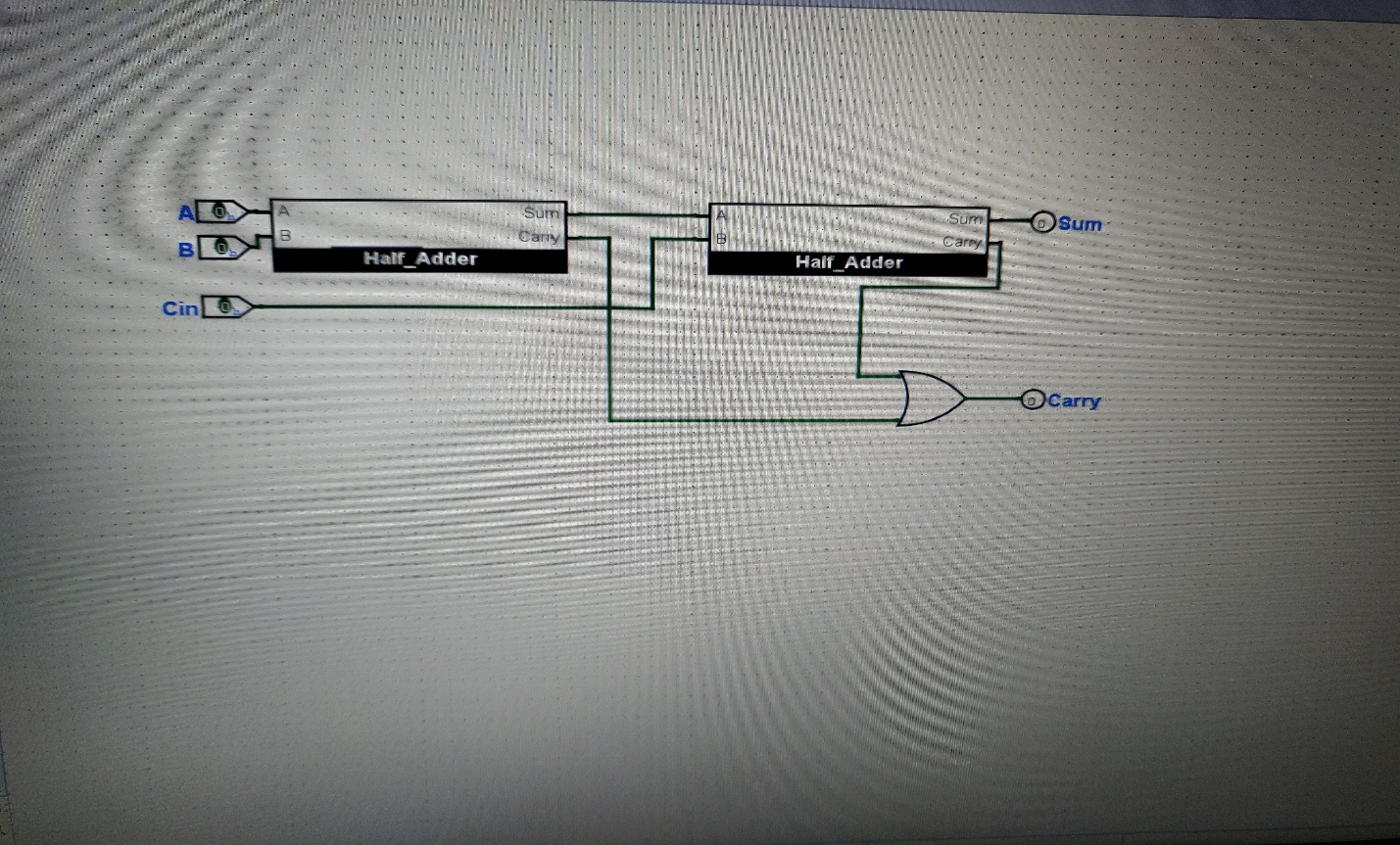


Negative (2’s Complement)



Half Adder



Full Adder

Half Adder:

2 inputs / 2 outputs

1 XOR Gate / 1 AND Gate

Full Adder:

3 inputs / 2 outputs

2 Half Adders + 1 OR Gate = 2 XOR Gates + 2 AND Gates + 1 OR Gate

Negative (2s Complement):

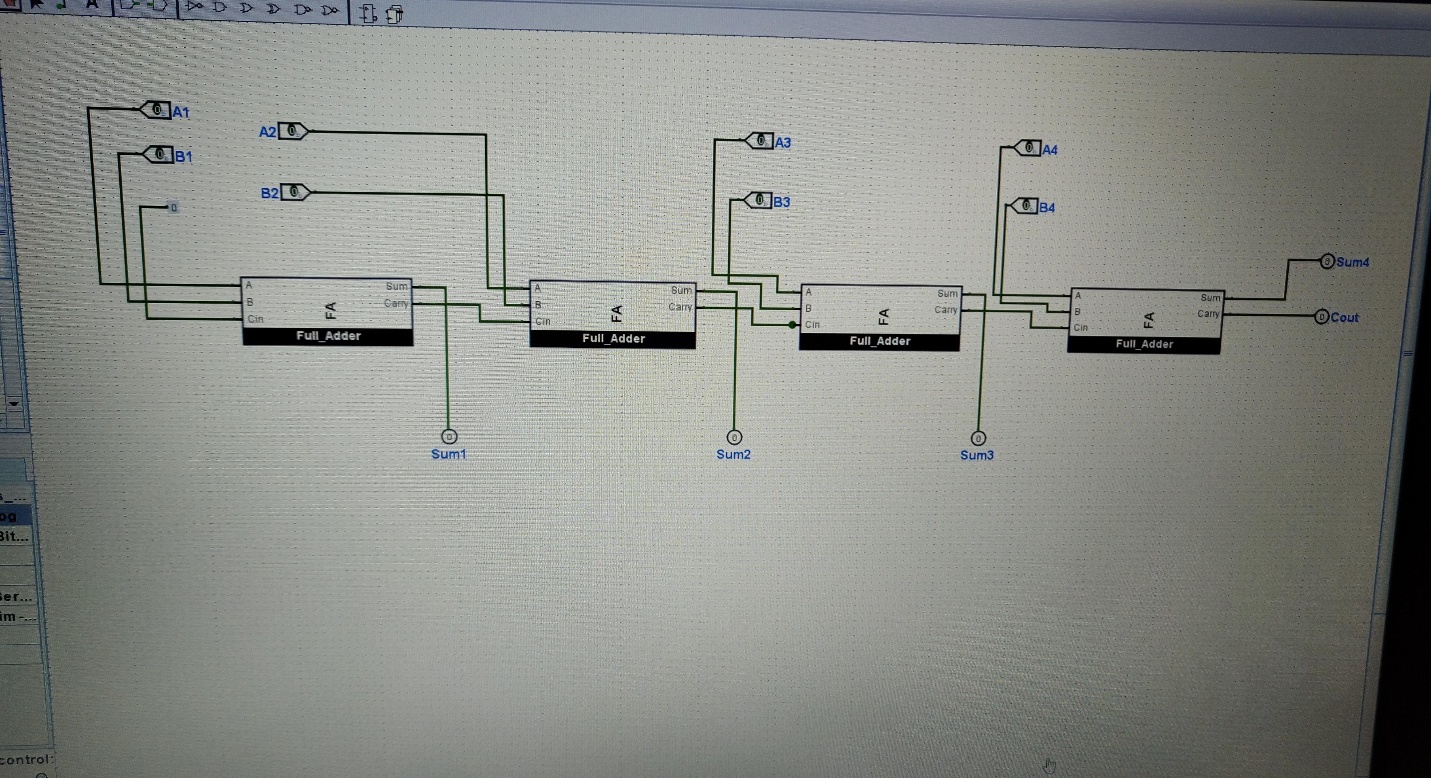
4 inputs and 1 “CONSTANT”/ 4 outputs

4 NOT Gates + 4 AND Gates + 4 XOR GATES

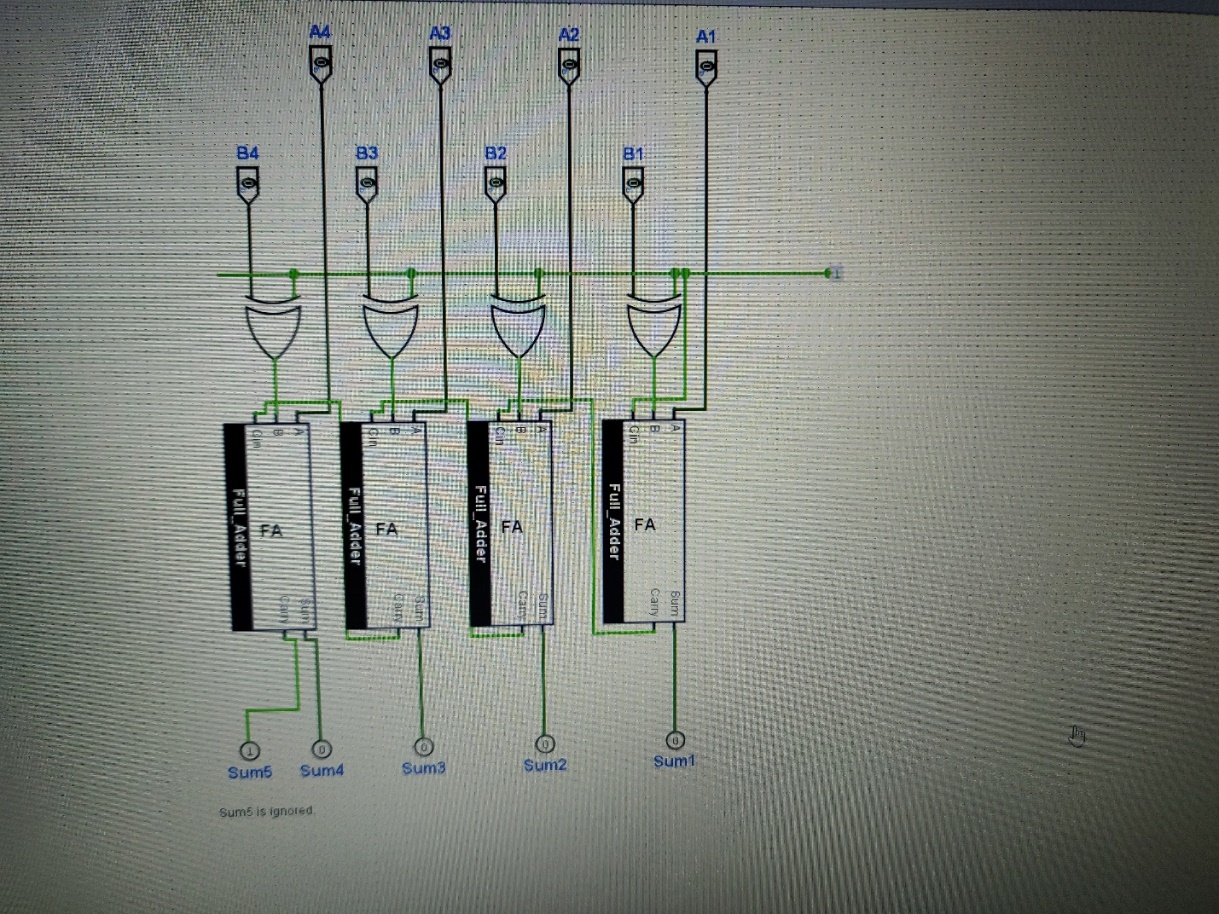
MULTIPLICATION:

16 AND Gates + 3 (4-Bit Adders) = 106 XOR Gates + 40 AND Gates + 24 OR Gates + 8 NOT Gates

B2h: 2(35 AND + 6 OR + 4 NOT)

4-Bit Adder

4-Bit Subtractor



4-Bit Adder:

8 inputs & “0 Constant” / 5 outputs

4 Full Adders

4-Bit Subtractor:

8 inputs & “1 Constant” / 5 outputs

4 Full Adders + 4 XOR