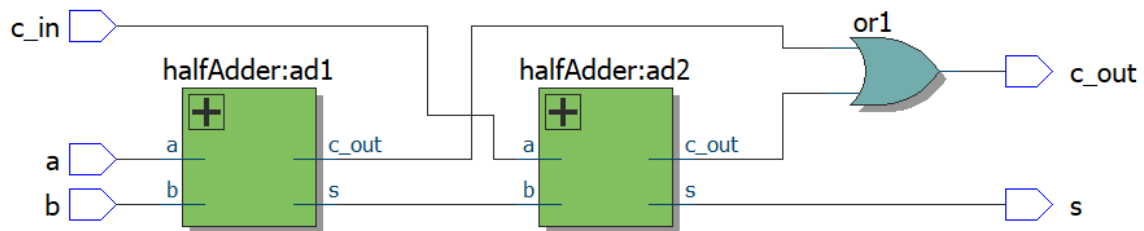


CSE 331 HW2

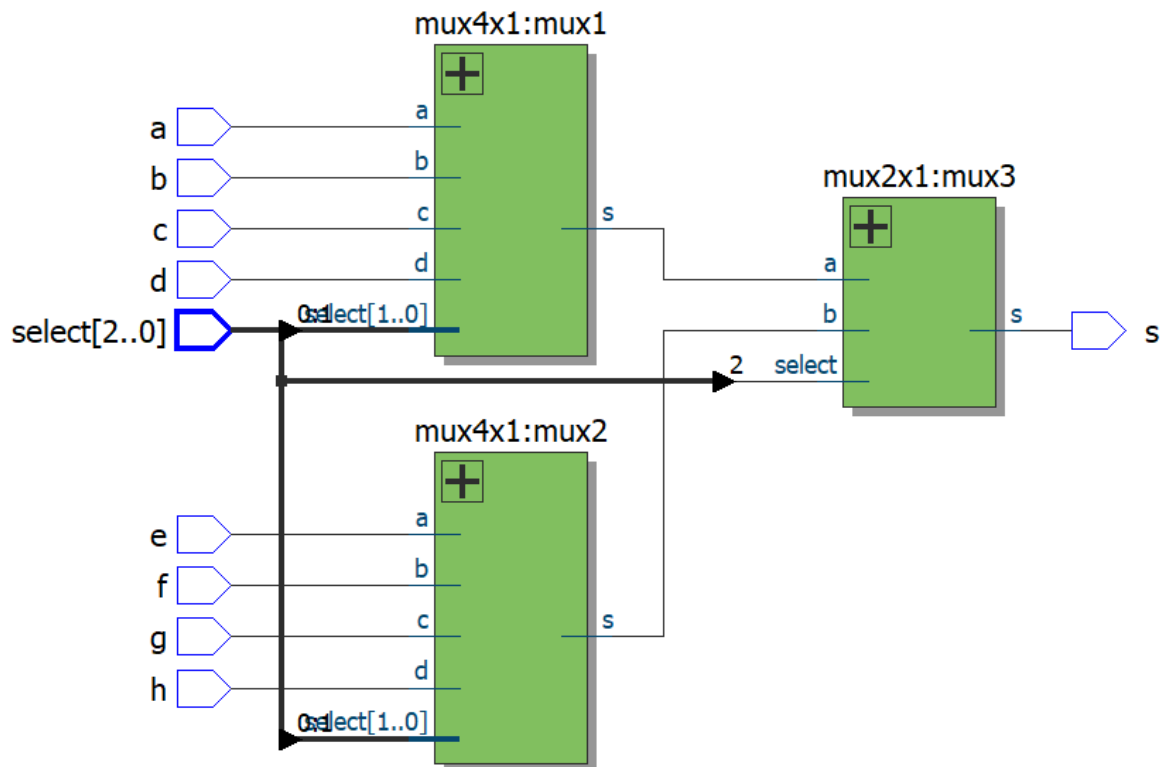
EXPLANATION OF VERILOG FILES

- fullAdder.v



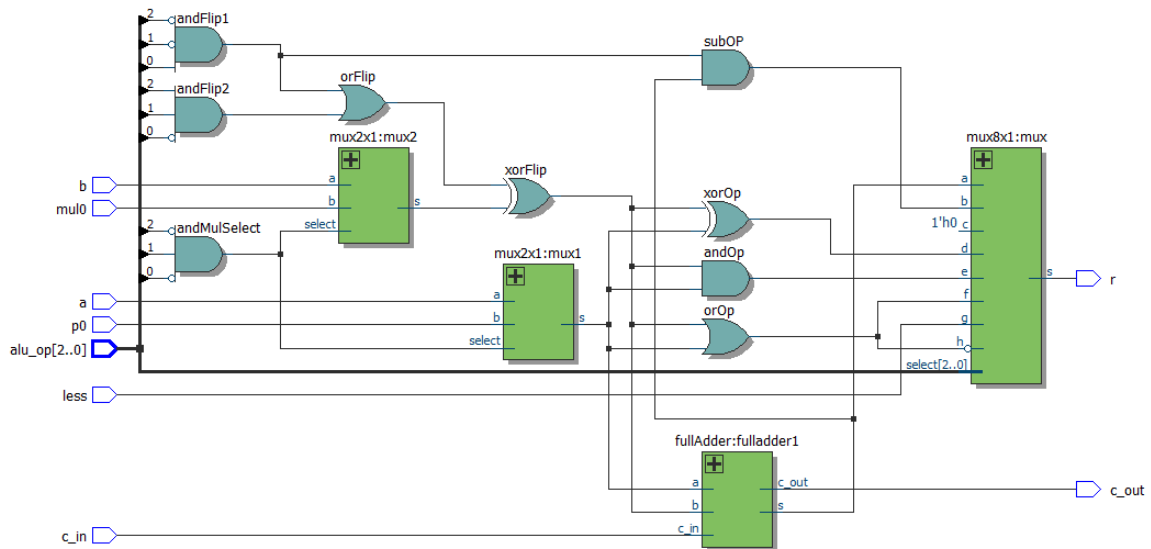
fullAdder module consist of 2 halfAdder module.

- mux8x1.v



The mux8x1 module is consist of 2x mux4x1 module and their output is connected to mux2x1 module to obtain single bit output.

- alu1Bit.v



The alu1Bit module consist of ;

bit fullAdder

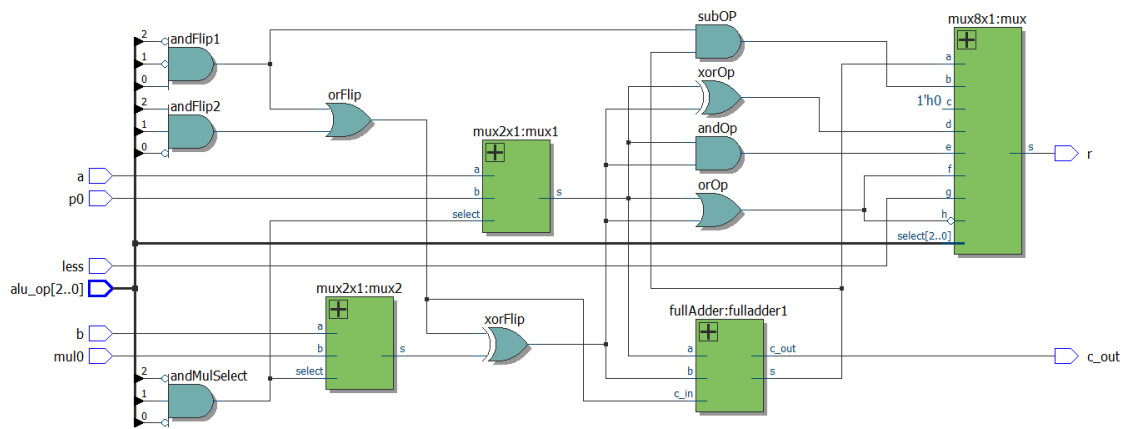
2x mux2x1

mux8x1

The module is similar to alu representation on the lecture. If the subtraction operation is held , then the “b” bit is inverted. SUB and SLT opcode uses subtraction operation behind. If these opcode is called then the “b” bit is inverted.

The “less” is input is 0.

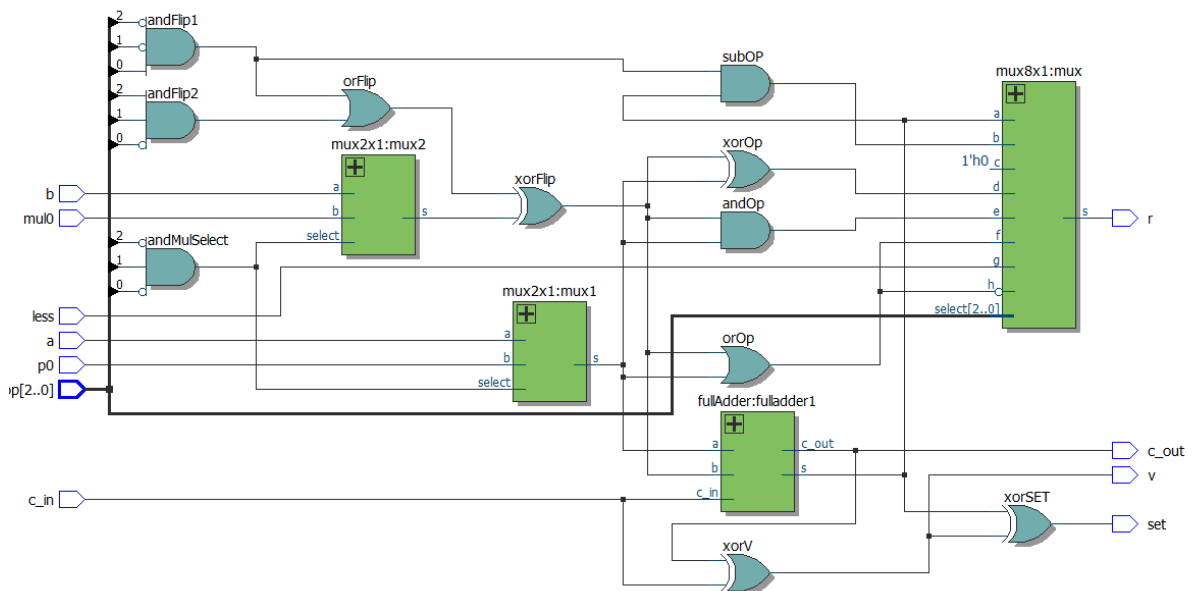
- alu1BitLsb



Essentially, it is same as the `alu1Bit`. But its carry in is came from according to `aluOp`. If the `aluOp` does the subtraction in operation then the carry in becomes 1 else 0.

Also `less` input came from `alu1BitMsb`'s "set" output.

- alu1BitMsb.v



The alu1BitMsb module consist of ;

fullAdder

2x mux2x1

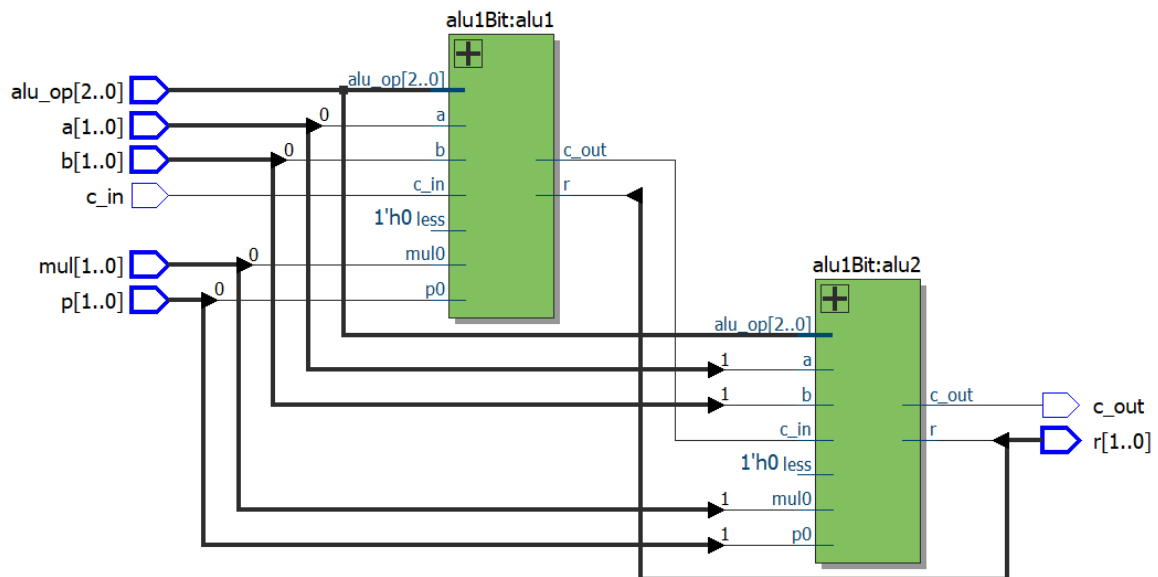
mux8x1

xorV and xorSet gate.

It is almost similar to alu1Bit , but it is using xorV and xorSet gate to detect overflow and indicate the sign of the result.

The “less” input is always 0. Also the value of “set” is directed to least significant bit alu’s carry in input.

- alu2Bit.v



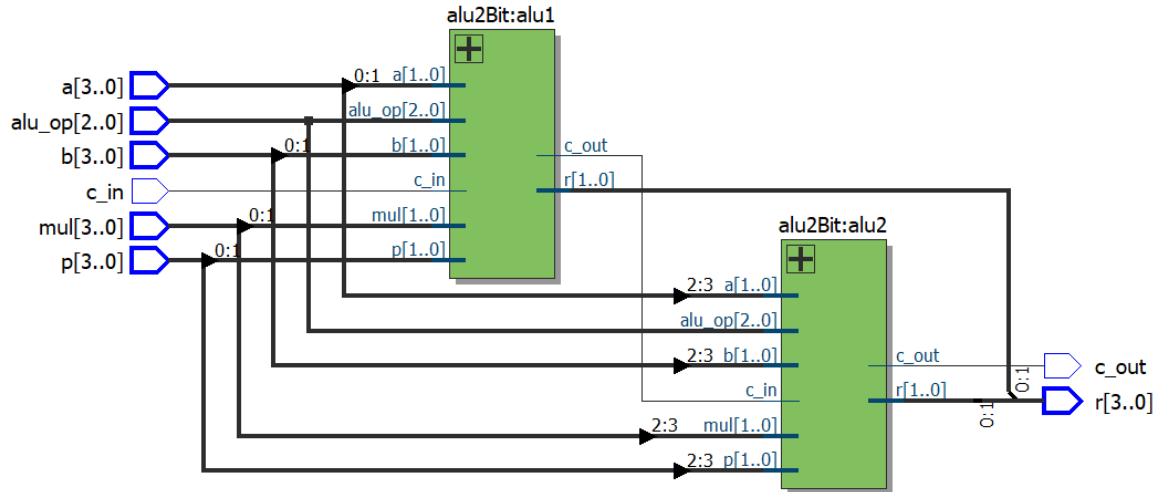
The alu2Bit module consist of ;

2x alu1Bit module.

It is used for create 32xAlu.

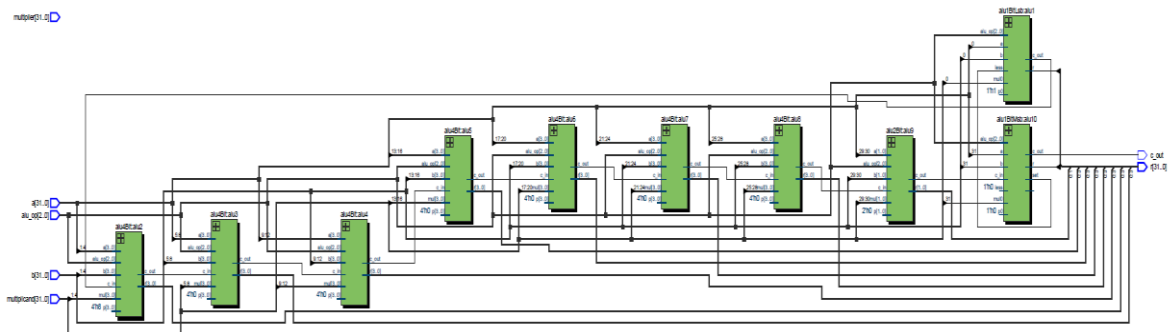
- Alu4Bit.v

The alu4Bit module consist of ;
2x alu2Bit module.
It is used for create 32xAlu.



- alu32.v

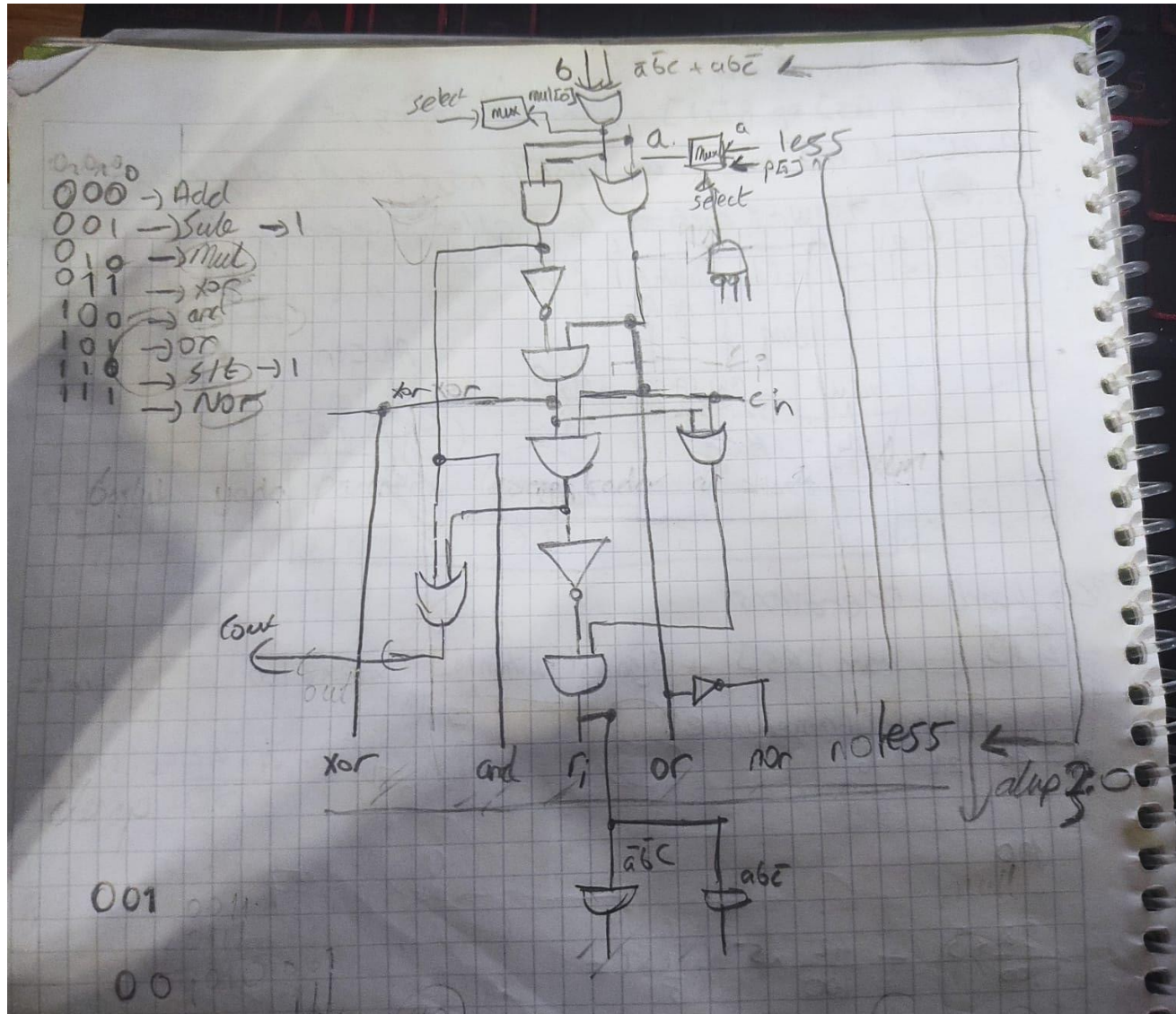
The alu32 module consist of ;
alu1BitLsb,
7x alu4Bit,
alu2Bit,
alu1bitMsb



HANDWORKS:

- 1 BIT ALU with MSB:

It contains 2x mux2x1 for multiplication operations but I have not combined it for 32 bit alu.



- ADD OPERATION ->> OPCODE ->> 000

	Msgs				
+ /addTestbench/a	-268432364	20	2624	-268432364	
+ /addTestbench/b	3347	19	3859	3347	
+ /addTestbench/s	-268429017	39	6483	-268429017	
+ /addTestbench/alu_op	000	000			
+ /addTestbench/c_out	St0				

- SUB OPERATION ->> OPCODE ->> 001

	Msgs				
+ /subTestbench/a	-No Data-	20	2624	-268432364	
+ /subTestbench/b	-No Data-	19	3859	3347	
+ /subTestbench/s	-No Data-	1	-1235	-268435711	
+ /subTestbench/alu_op	-No Data-	001			
+ /subTestbench/c_out	-No Data-				

- MUL OPERATION ->> OPCODE ->> 010



- XOR OPERATION ->> OPCODE ->> 011

	Msgs				
+ /xorTestbench/a	-No Data-	00000000000000000000000010100	00000000000000000000101001000000	11110000000000000000110000010100	
+ /xorTestbench/b	-No Data-	00000000000000000000000010011	00000000000000000000111100010011	00000000000000000000110100010011	
+ /xorTestbench/s	-No Data-	000000000000000000000000111	00000000000000000000101010011	111100000000000000000000010000111	
+ /xorTestbench/alu_op	-No Data-	011			
+ /xorTestbench/c_out	-No Data-				

- AND OPERATION ->> OPCODE ->> 100

	Msgs				
+ /andTestbench/a	-No Data-	00000000000000000000000010100	00000000000000000000101001000000	11110000000000000000110000010100	
+ /andTestbench/b	-No Data-	00000000000000000000000010011	00000000000000000000111100010011	00000000000000000000110100010011	
+ /andTestbench/s	-No Data-	00000000000000000000000010000	00000000000000000000101000000000	00000000000000000000110000010000	
+ /andTestbench/alu_op	-No Data-	100			
+ /andTestbench/c_out	-No Data-				

- OR OPERATION ->> OPCODE ->> 101

	Msgs	
/orTestbench/a	111100000000...	00000000000000000000000010100 0000000000000000000010100100000 11110000000000000000110000010100
/orTestbench/b	000000000000...	00000000000000000000000010011 00000000000000000000111100010011 00000000000000000000110100010011
/orTestbench/s	111100000000...	00000000000000000000000010111 00000000000000000000111101010011 11110000000000000000110100010111
/orTestbench/alu_op	101	101
/orTestbench/c_out	St0	

- SLT OPERATION ->> OPCODE ->> 110

	Msgs	
/sltTestbench/a	-No Data-	20 -268432364
/sltTestbench/b	-No Data-	19 3859 3347
/sltTestbench/s	-No Data-	00000000000000000000000000000000 00000000000000000000000000000001 00000000000000000000000000000001
/sltTestbench/alu_op	-No Data-	110
/sltTestbench/c_out	-No Data-	

- NOR OPERATION ->> OPCODE ->> 111

	Msgs	
/norTestbench/a	-No Data-	00000000000000000000000010100 0000000000000000000010100100000 11110000000000000000110000010100
/norTestbench/b	-No Data-	00000000000000000000000010011 00000000000000000000111100010011 00000000000000000000110100010011
/norTestbench/s	-No Data-	11111111111111111111111101000 1111111111111111111111000010101100 00001111111111111111110101101000
/norTestbench/alu_op	-No Data-	111
/norTestbench/c_out	-No Data-	

- ALL 32 BIT ALU OPERATIONS

	Msgs	
/alu32Testbench/a	000000000000...	00000000000000000000000010100
/alu32Testbench/b	000000000000...	00000000000000000000000010011
/alu32Testbench/alu...	111	000 001
/alu32Testbench/c_...	St0	
/alu32Testbench/s	111111111111...	000000000000000000000000100111 00000000000000000000000000000001

	Msgs	
/alu32Testbench/a	000000000000...	00000000000000000000000010100
/alu32Testbench/b	000000000000...	00000000000000000000000010011
/alu32Testbench/alu...	111	011 100
/alu32Testbench/c_...	St0	
/alu32Testbench/s	111111111111...	000000000000000000000000001111 00000000000000000000000000000000

	Msgs	
/alu32Testbench/a	00000000000000...	00000000000000000000000010100
/alu32Testbench/b	00000000000000...	00000000000000000000000010011
/alu32Testbench/al...	110	101
/alu32Testbench/c_...	St1	110
/alu32Testbench/s	00000000000000...	00000000000000000000000010111

	Msgs	
/alu32Testbench/a	00000000000000...	00000000000000000000000010100
/alu32Testbench/b	00000000000000...	00000000000000000000000010011
/alu32Testbench/al...	110	111
/alu32Testbench/c_...	St1	
/alu32Testbench/s	00000000000000...	111111111111111111111111101000

NOTE: ALL THE OPERATIONS USES ONLY ONE ADDER.

Muhammed Sinan
Pehlivanoğlu
1901042664