

Arithmetic and Logical Unit Design

ECE 212





Arithmetic and Logical Unit Design Project

ECE 212: Digital Circuits

Prepared by: Team 1

Spring 2021

S-CHEP

Team members:

Name	ID
Omar Mohamed Diaaeldin Ibrahim	1802932
Ahmed Magdy Fahmy Mohamed	1805862
Ahmed Osama Mohamed Elgerby Mohamed Elsawy	1803400
Ahmed Mohamed Reda Mahmoud Hassan	1801772
Ahmed Ayman Abd-alaziz Sharf	1806171
Ahmed Magdi Mostafa Hosni	1808714
Abdalrahman Amr Mohammed Effat	1809835
Abd Elrahman Mostafa Rifaat	1805551
Abdelrahman Mohamed Yousif Abdulla	18T7239
Omar wael saad eldwakhly	1804197
Mahmoud Taher rashad younis	1809769

Under supervision of: Dr. Sameh A. Ibrahim

Contents

1.	VHDL	3
➤	ALU Code.....	3
➤	Test Bench	5
➤	Test Bench Output	6
2.	Cadence (Transistor level design)	6
➤	Design the reference inverter.....	6
➤	2 Input NAND.....	8
➤	2 Input NOR.....	9
➤	3 Input NOR.....	10
➤	4 Input NOR.....	11
3.	Cadence (Block diagram)	12
➤	SEL “selections”	12
➤	A00 & B00.....	13
➤	A01 to A03 & B01 to B03	14
➤	B000 to B003	15
➤	C0.....	16
➤	C1 & C2	17
➤	C3.....	18
➤	F0	19
➤	F01 to F03	20
➤	F1 to F3	21
4.	Cadence (All Blocks connected together)	22
5.	Cadence (ALU output waveforms).....	23
6.	Cadence (delay & power calculations)	32
➤	Delay	32
➤	Consumed Power	33

1. VHDL

➤ ALU Code

```
-- 4-bit ALU
-- sel Operation Unit
-- 0000 increment a
-- 0001 decrement a
-- 0010 transfer b
-- 0011 increment b Aritmetic
-- 0100 decrement b
-- 0101 transfer a
-- 0110 add a and b
-- 0111 multiply a by 2
-- 1000 complement a (1s complement)
-- 1001 complement b
-- 1010 AND
-- 1011 OR
-- 1100 XOR Logic
-- 1101 XNOR
-- 1110 NAND
-- 1111 NOR
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_signed.all;
entity alu is
port(
a,b,sel: IN STD_LOGIC_VECTOR(3 downto 0); --Defining a,b,sel
result : out STD_LOGIC_VECTOR(4 downto 0) --Defining result
);
end alu;
architecture dataflow of alu is
begin
process(sel)
```

```

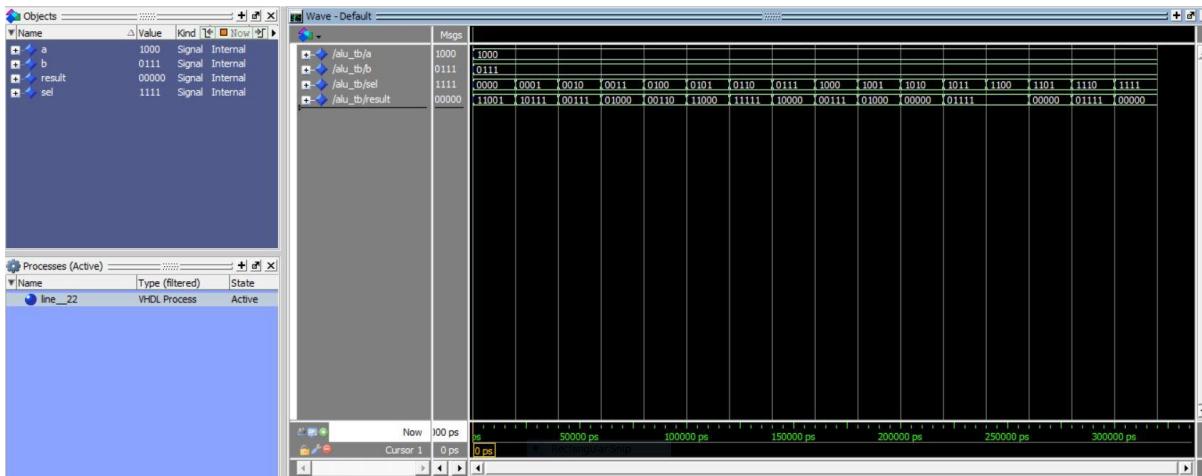
begin
-- adding zeroes to the left to put it in 5 bits where each operand must be in
5 bit format
-- Arithmetic Operations
if sel = "0000" then result <= ("00000"+ a + "00001"); -- increment a
elsif sel = "0001" then result <= ("00000"+ a + "11111"); -- decrement a
elsif sel = "0010" then result <= ( "00000" + b); -- transfer b
elsif sel = "0011" then result <= ("00000"+ b + "00001"); -- increment b
elsif sel = "0100" then result <= ( "00000"+ b + "11111"); -- decrement b
elsif sel = "0101" then result <= ("00000" + a ); -- transfer a
elsif sel = "0110" then result <= ("00000" + a + b ); -- Add a and b
elsif sel = "0111" then result <= ("00000" +(a & "0") );
-- multiply (a) by 2 or shift left 1 bit
-- Logical Operations
elsif sel = "1000" then result(3 downto 0) <= not a; --
1's complement of a (invert a)
elsif sel = "1001" then result(3 downto 0) <= not b ; --
1's complement of b (invert b)
elsif sel = "1010" then result(3 downto 0) <= (a and b); -- a AND b
elsif sel = "1011" then result(3 downto 0) <= (a or b); -- a OR b
elsif sel = "1100" then result(3 downto 0) <= (a xor b); -- a XOR b
elsif sel = "1101" then result(3 downto 0) <= (a xnor b); -- a XNOR b
elsif sel = "1110" then result(3 downto 0) <= (a nand b); -- a NAND b
elsif sel = "1111" then result(3 downto 0) <= (a nor b); -- a NOR b
else result <= "ZZZZZ"; --
Setting result to High impendence(ZZZZZ) if no other condition
is satisfied
end if ;
--Setting MSB bit in Logical Operations to Zero
if sel(3) = '1' then result(4)<= '0'; end if ;
end process;
end dataflow;

```

➤ Test Bench

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_signed.all;
use std.env.stop;
entity alu_tb is
end alu_tb;
architecture tb_arch of alu_tb is
component alu
port(
a,b,sel : IN STD_LOGIC_VECTOR(3 downto 0); --Defining a,b,sel
result : out STD_LOGIC_VECTOR(4 downto 0) --Defining result
);
end component;
SIGNAL a,b,sel : std_logic_VECTOR(3 downto 0);
SIGNAL result : std_logic_VECTOR(4 downto 0);
begin
UUT : alu port map (a => a, b => b, sel => sel,result => result);
process
begin
a <= "1000";
b<= "0111";
sel<="0000";
forloop: FOR i IN 1 TO 16 loop
wait for 20 ns;
sel <= sel + 1 ; -- increment sel by one
END loop;
stop; -- stop infinite loop
end process;
end tb_arch;
```

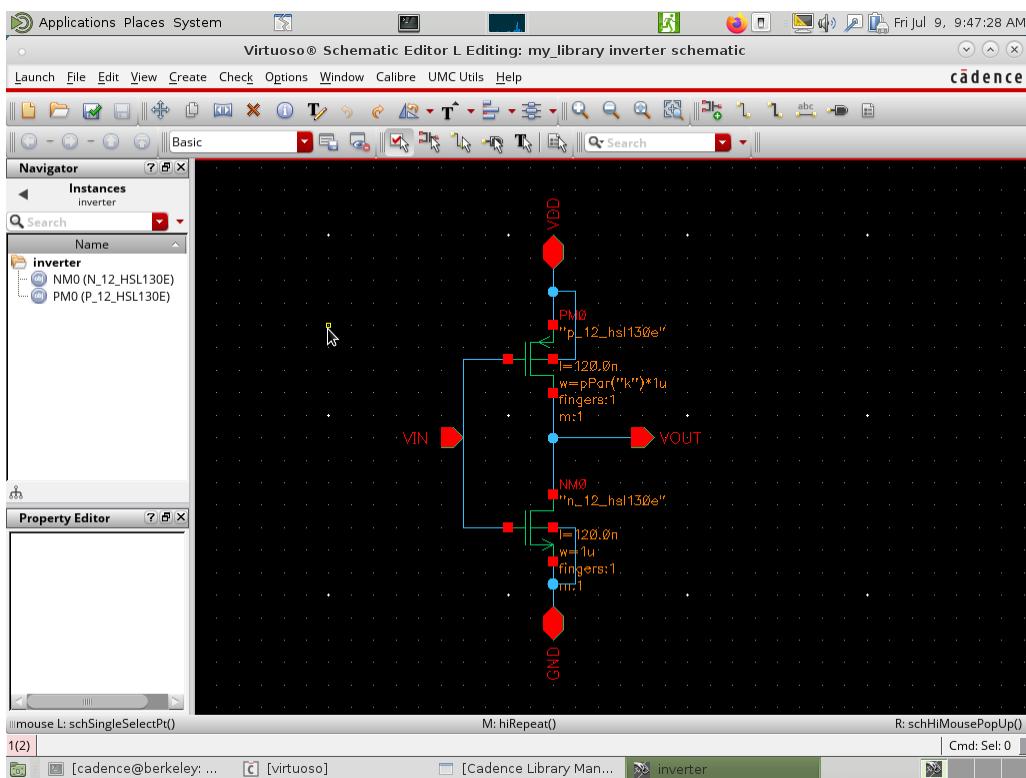
➤ Test Bench Output



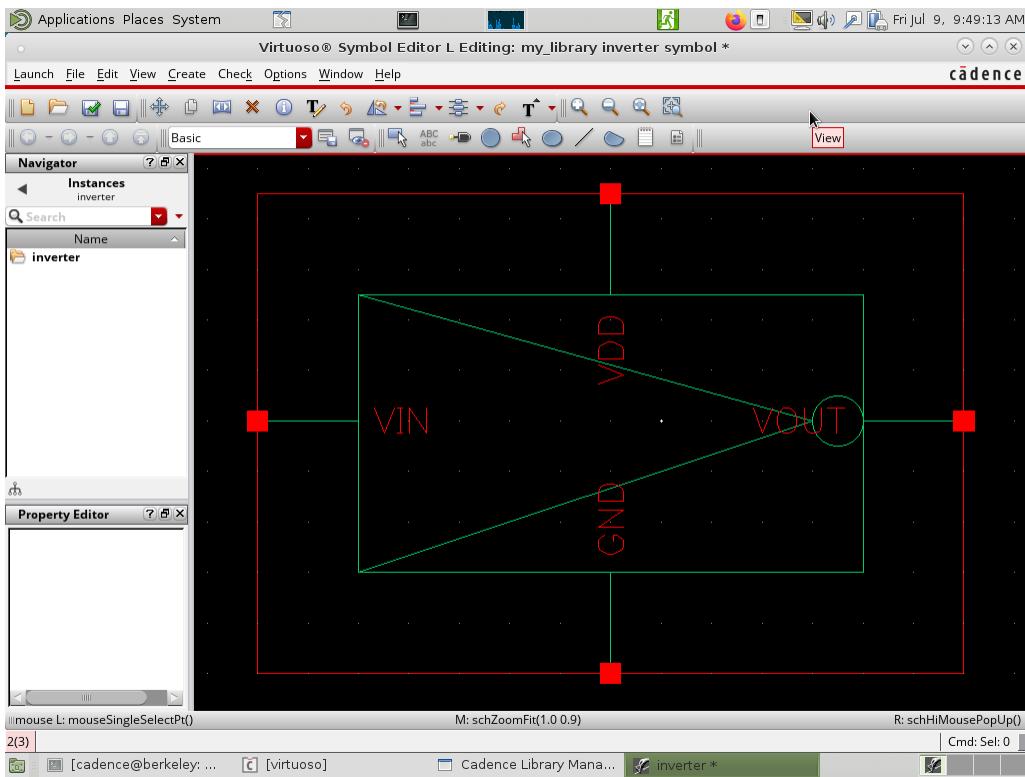
2. Cadence (Transistor level design)

➤ Design the reference inverter

- The inverter schematic

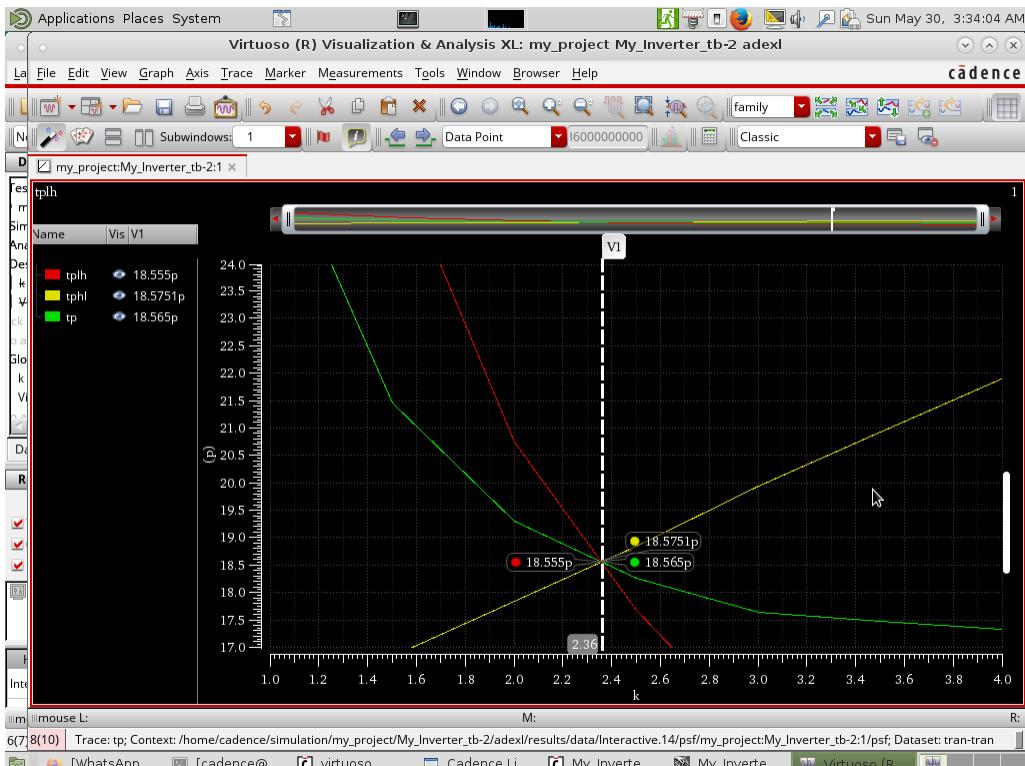


- The inverter symbol



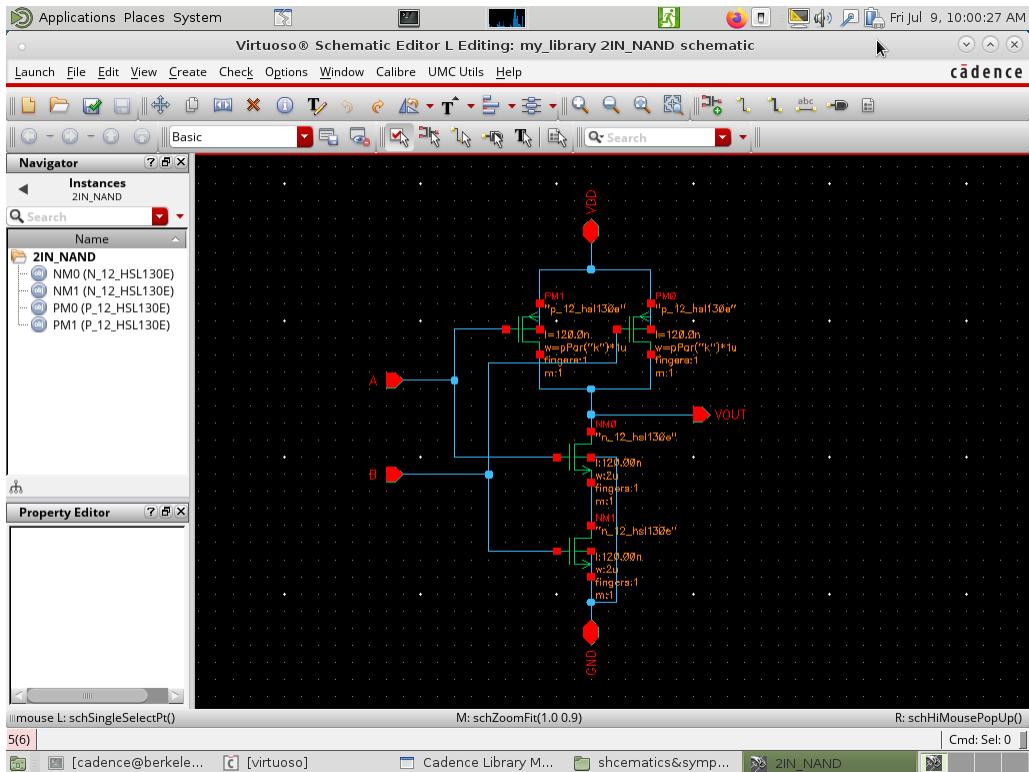
- Choosing the value of “k” to make the inverter symmetric

$$k = 2.36$$

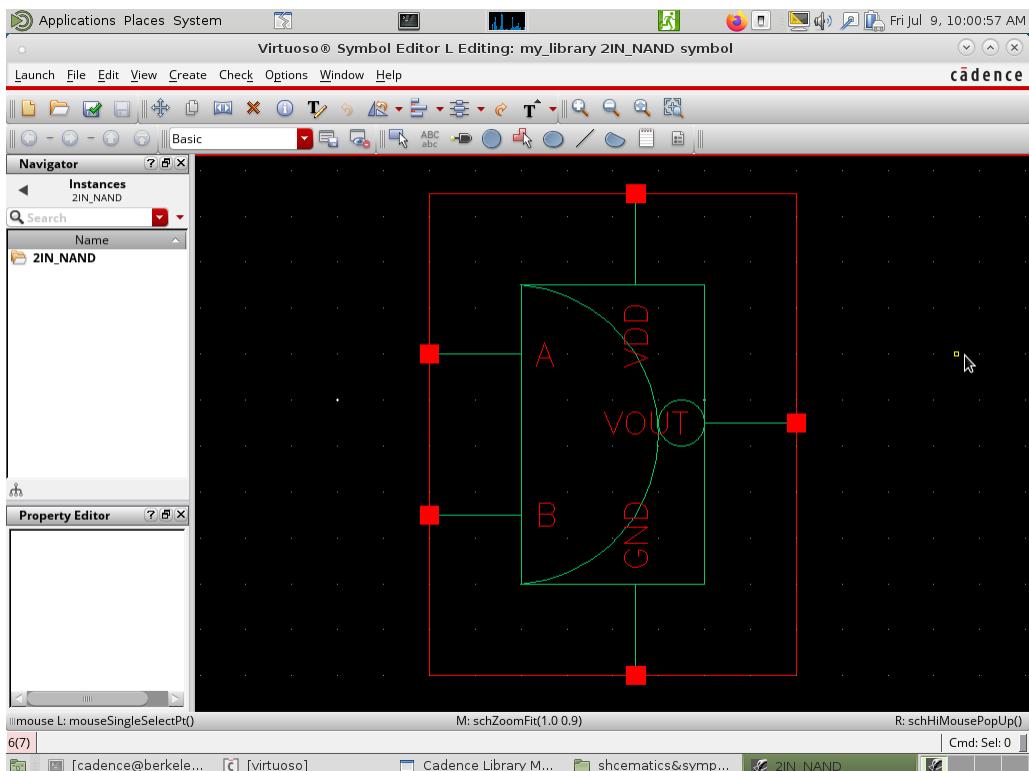


➤ 2 Input NAND

- Schematic

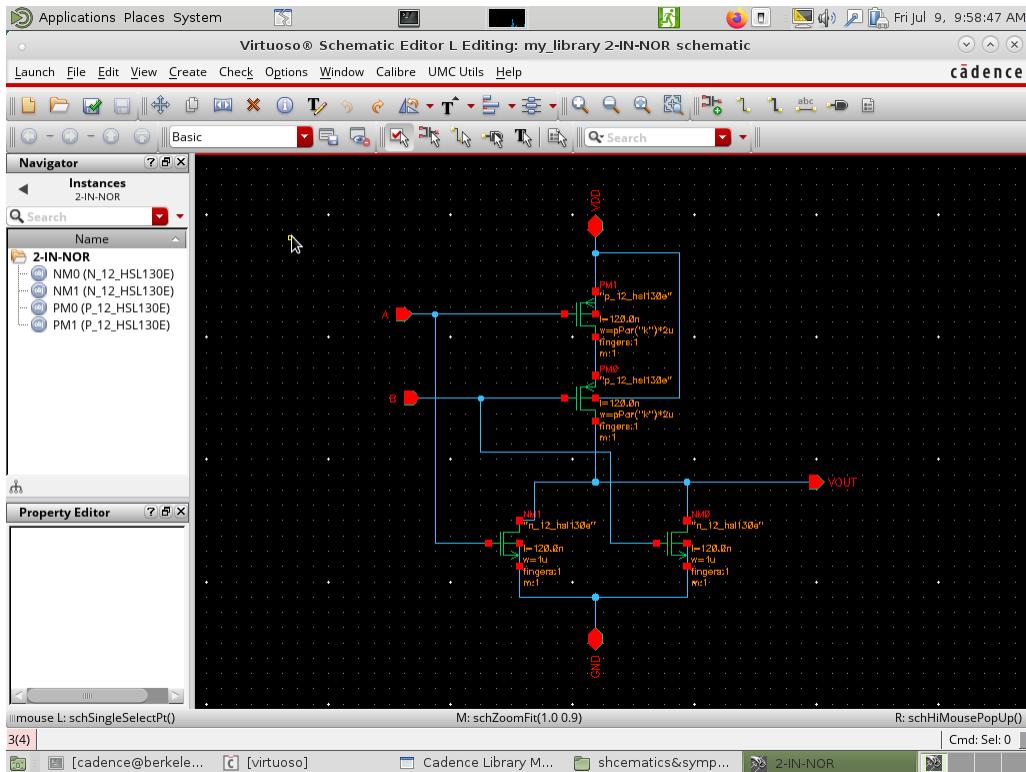


- Symbol

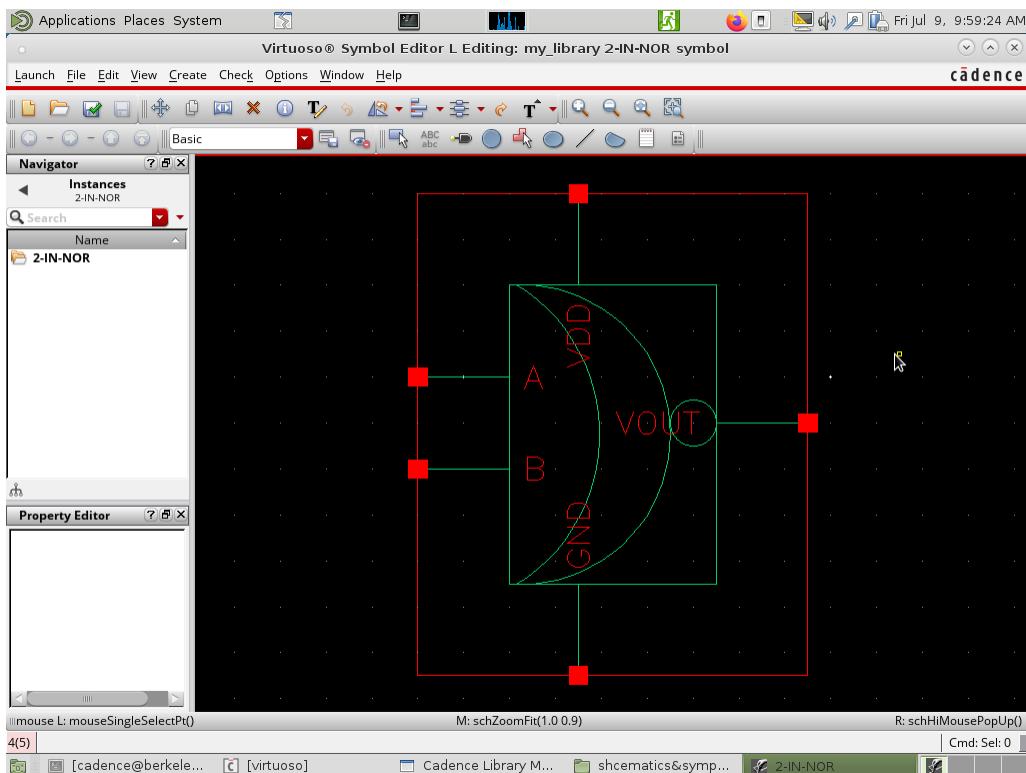


➤ 2 Input NOR

- Schematic

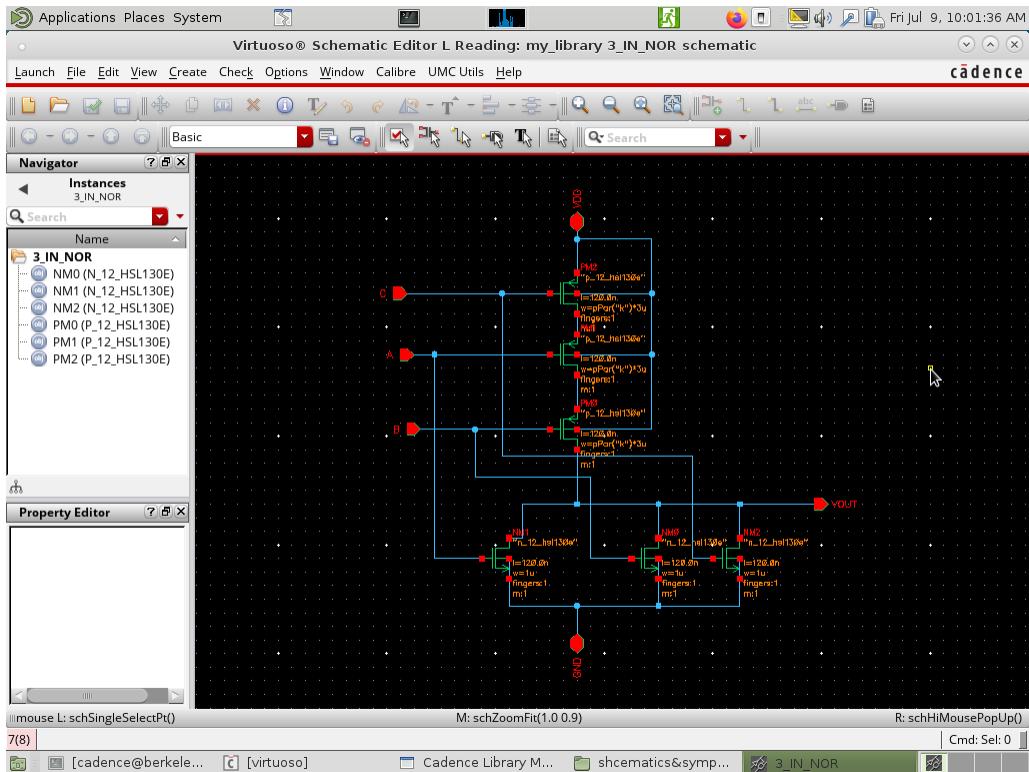


- Symbol

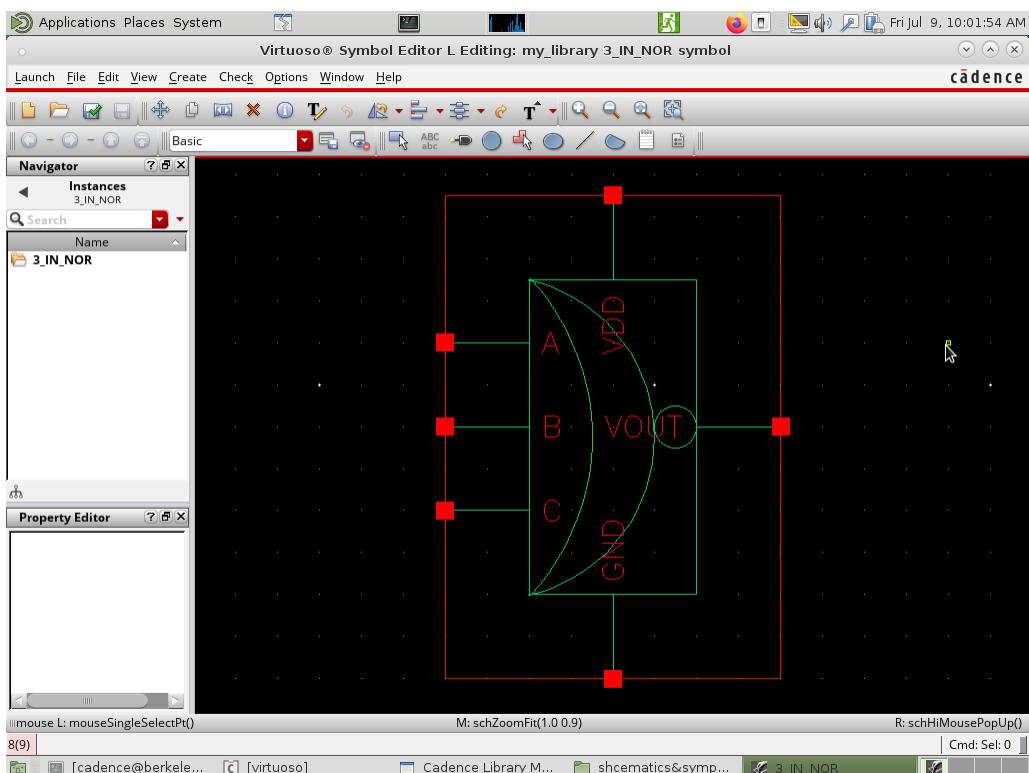


➤ 3 Input NOR

- Schematic

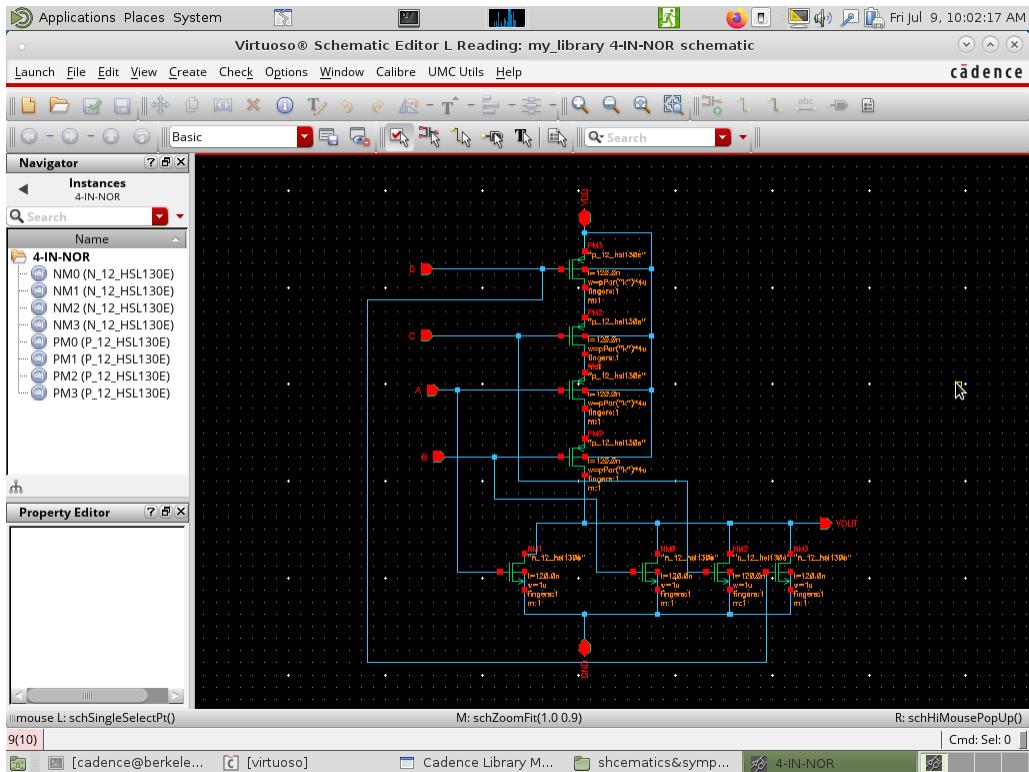


- Symbol

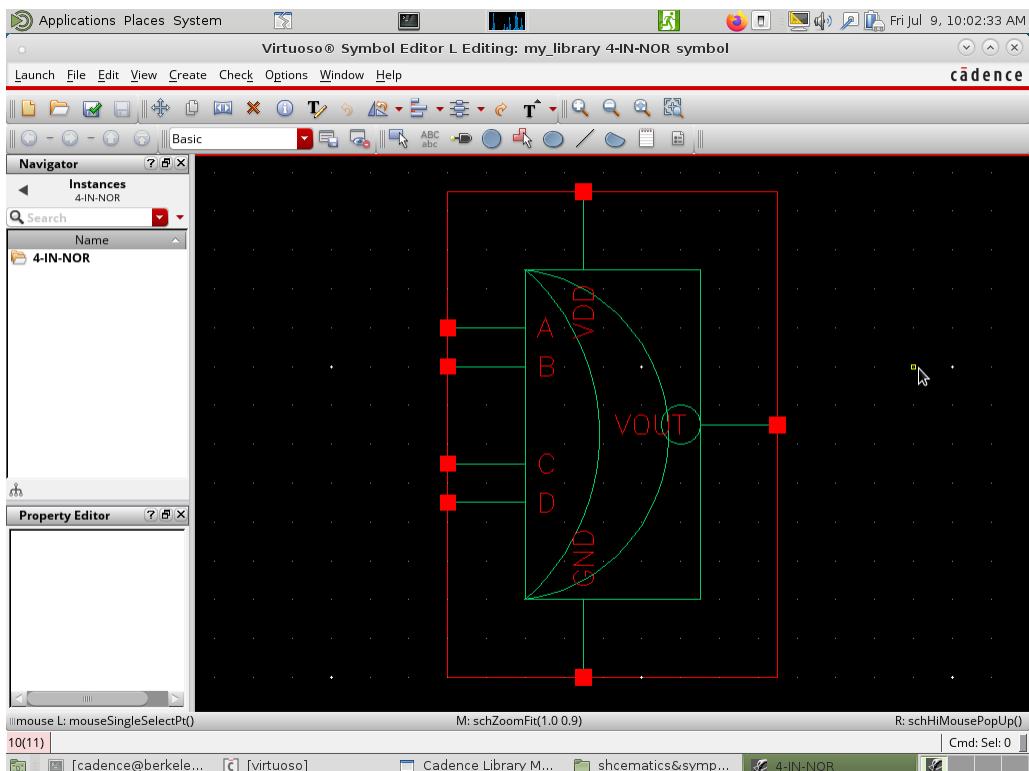


➤ 4 Input NOR

- Schematic



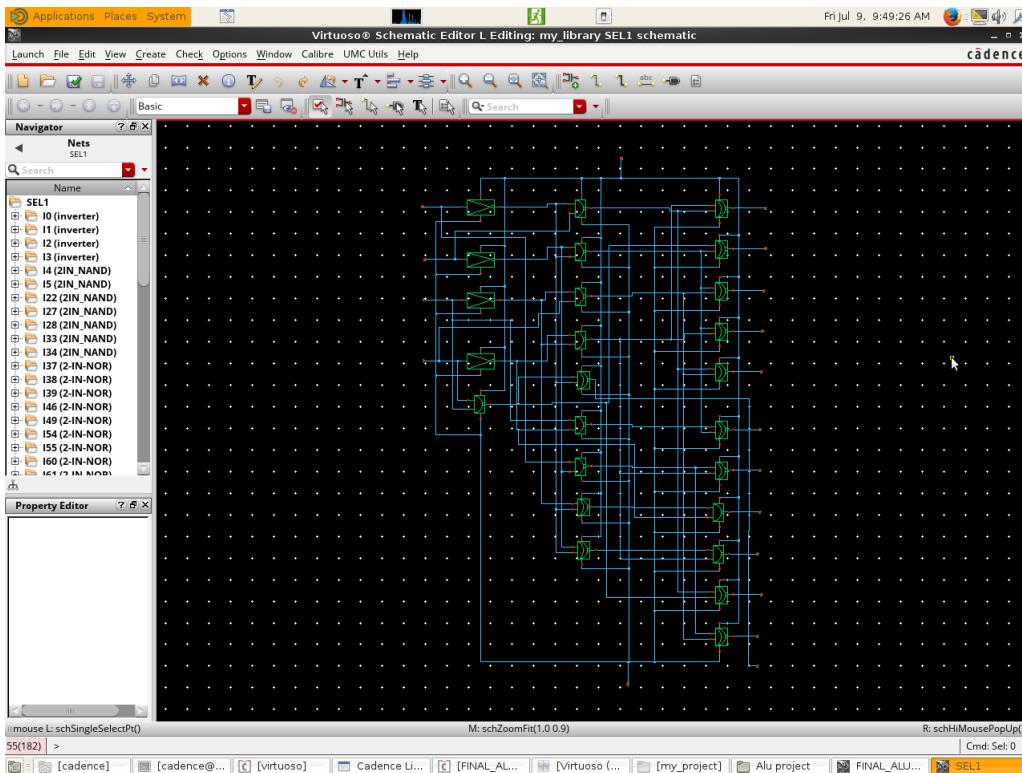
- Symbol



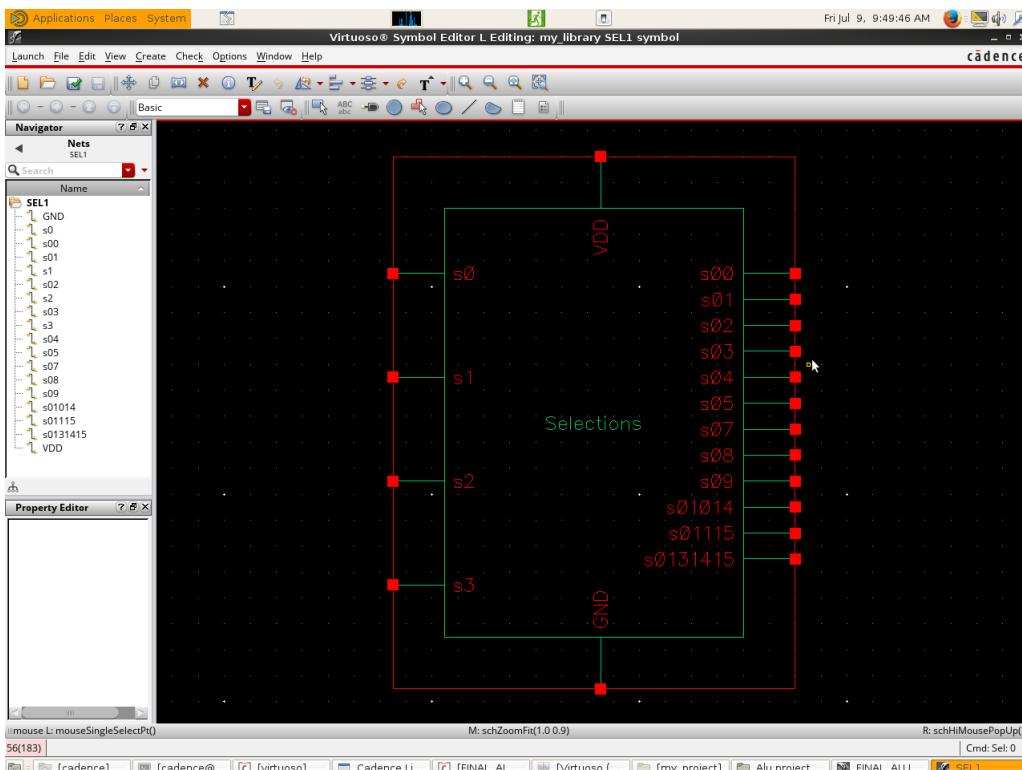
3. Cadence (Block diagram)

➤ SEL “selections”

- Schematic



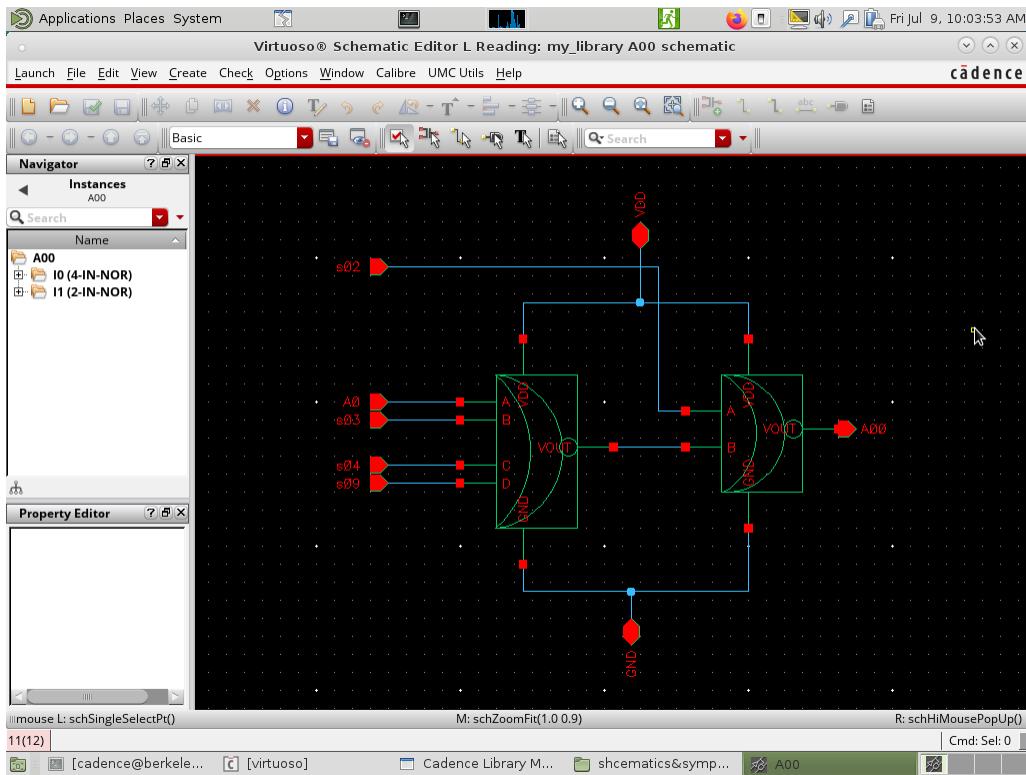
- Block



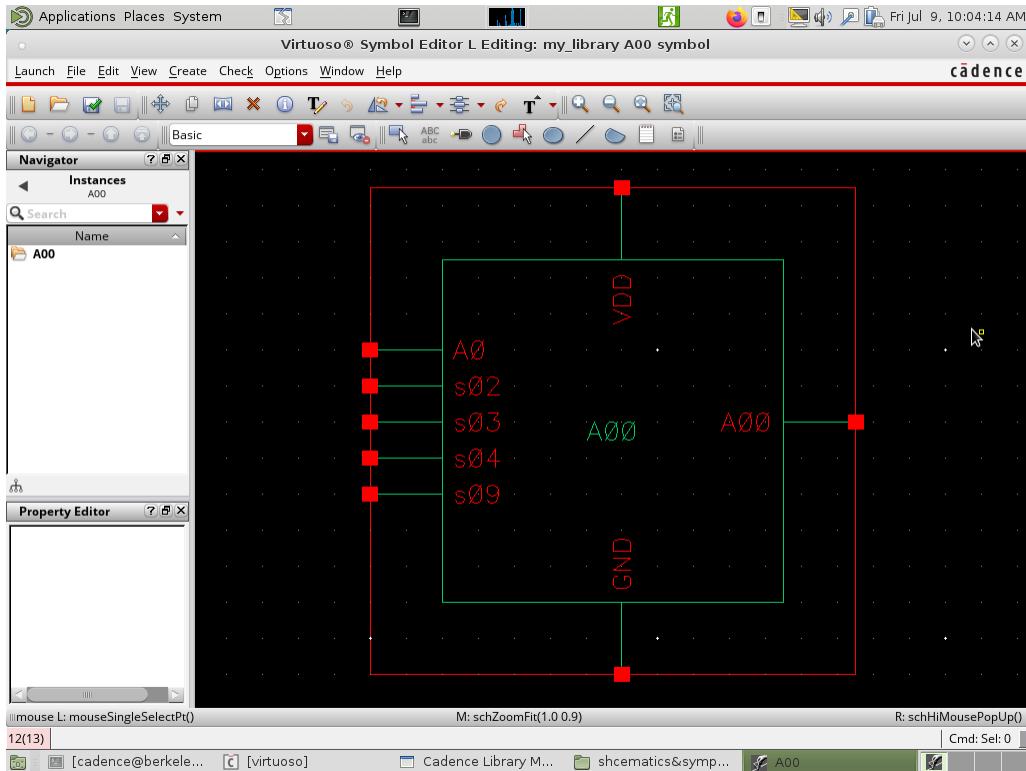
➤ A00 & B00

The circuits are identical but the difference is in the inputs and therefore the output

- Schematic



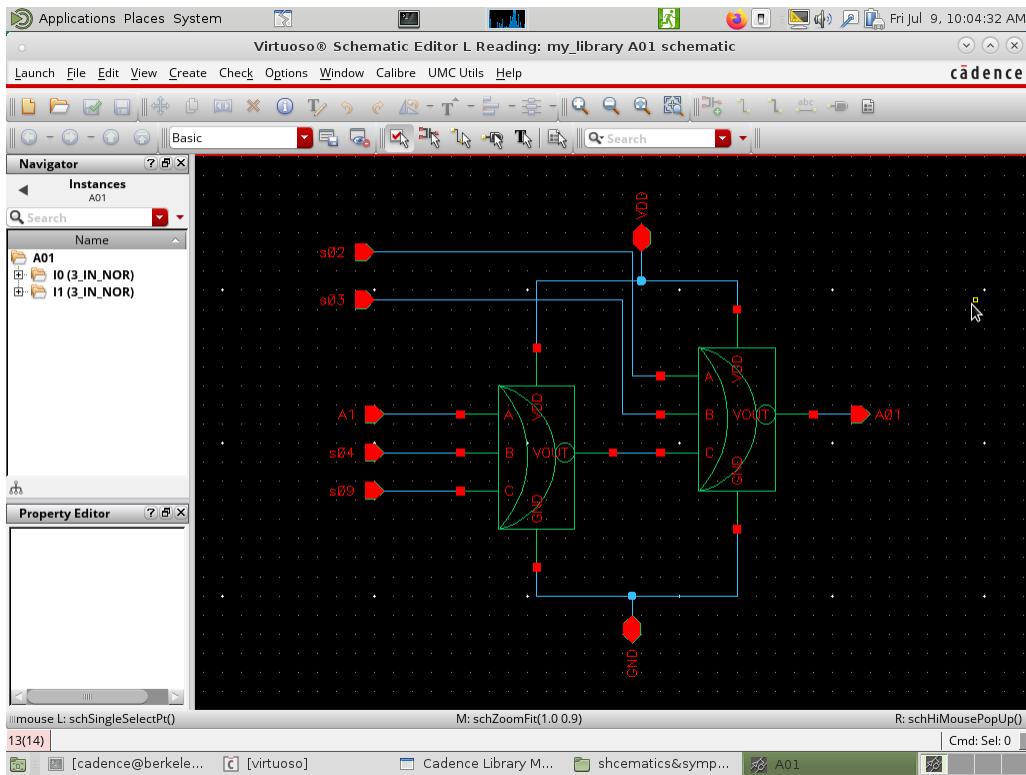
- Block



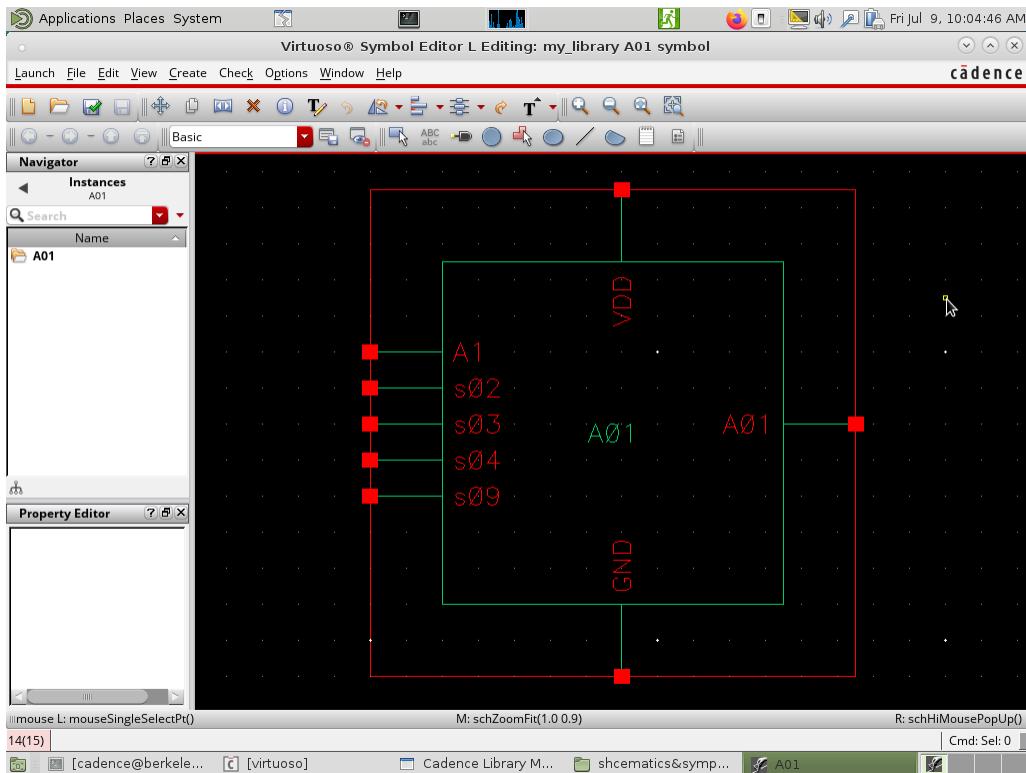
➤ A01 to A03 & B01 to B03

The circuits are identical but the difference is in the inputs and therefore the output

- Schematic



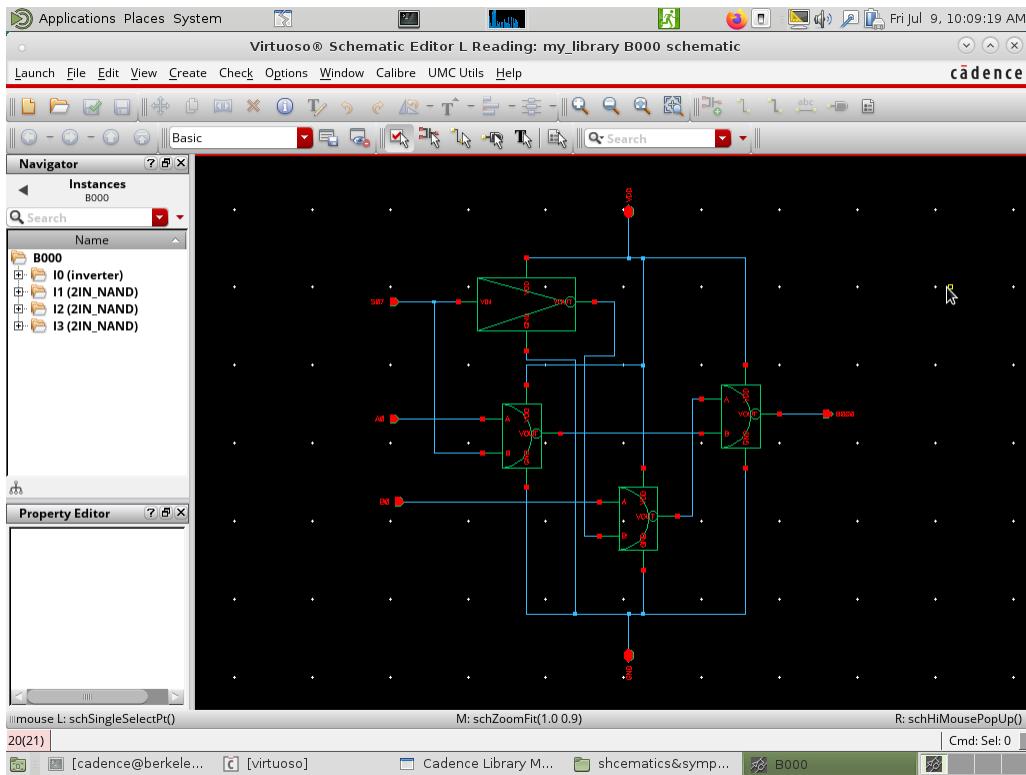
- Block



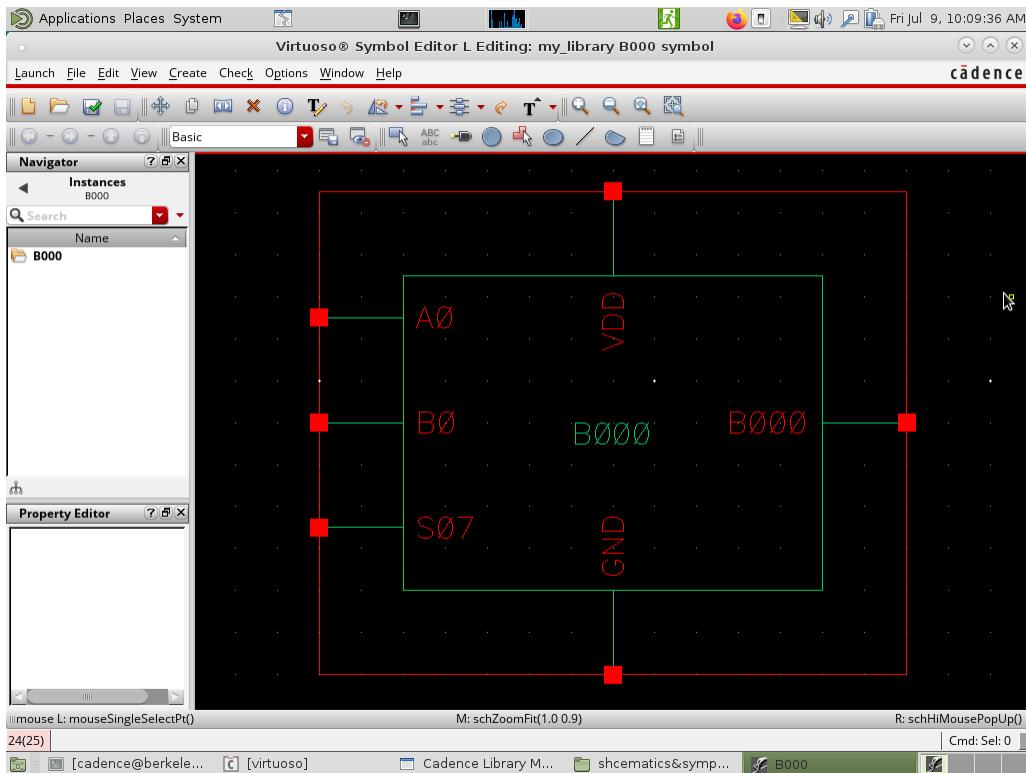
➤ B000 to B003

The circuits are identical but the difference is in the inputs and therefore the output

- Schematic

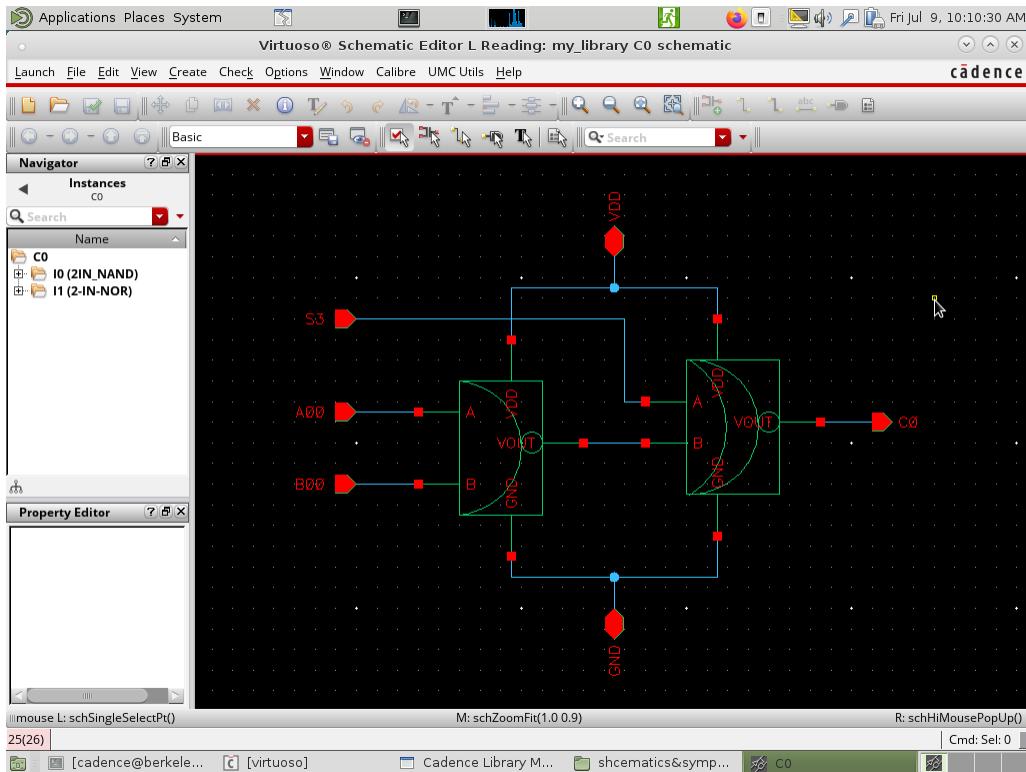


- Block

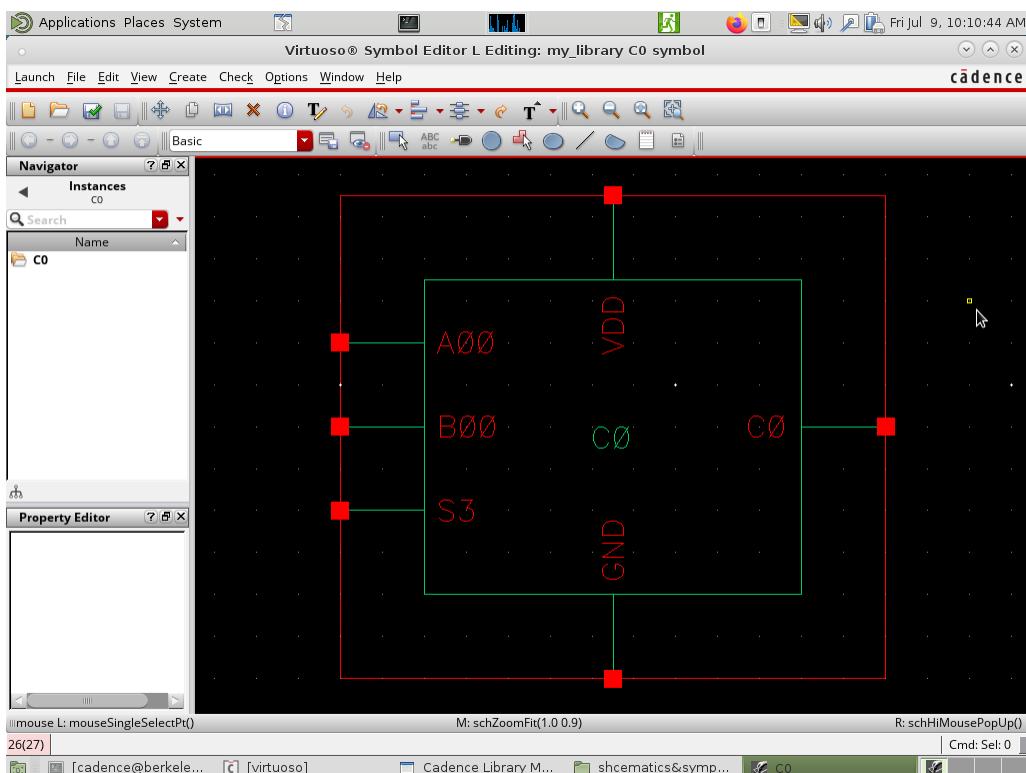


➤ C0

- Schematic



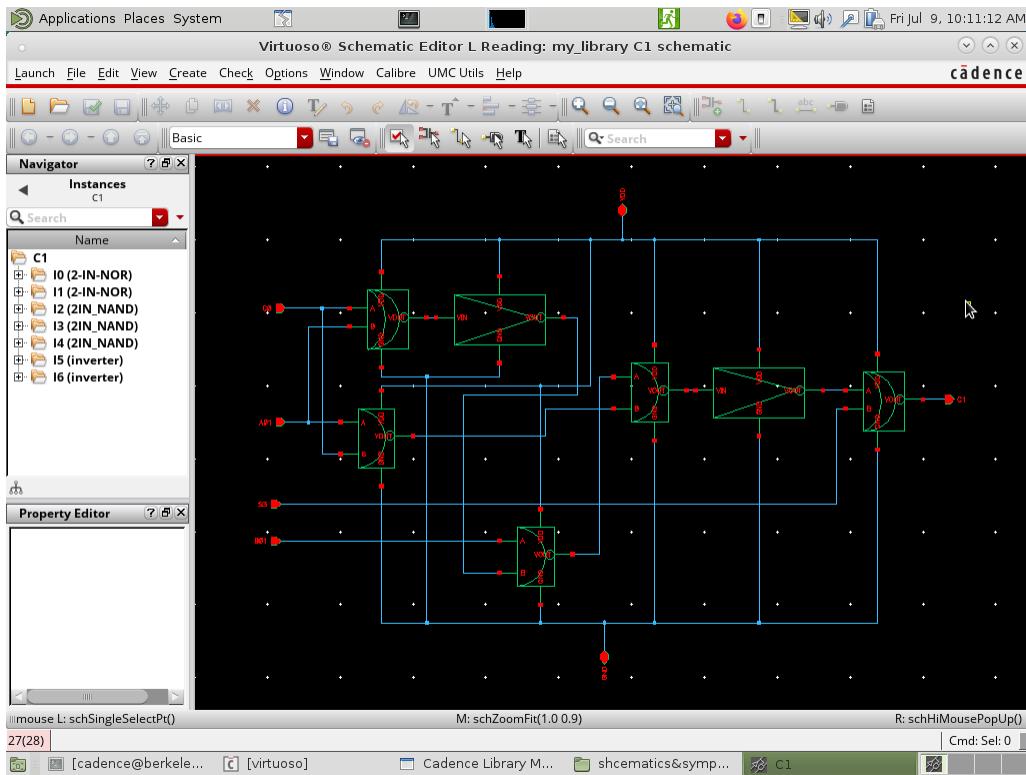
- Block



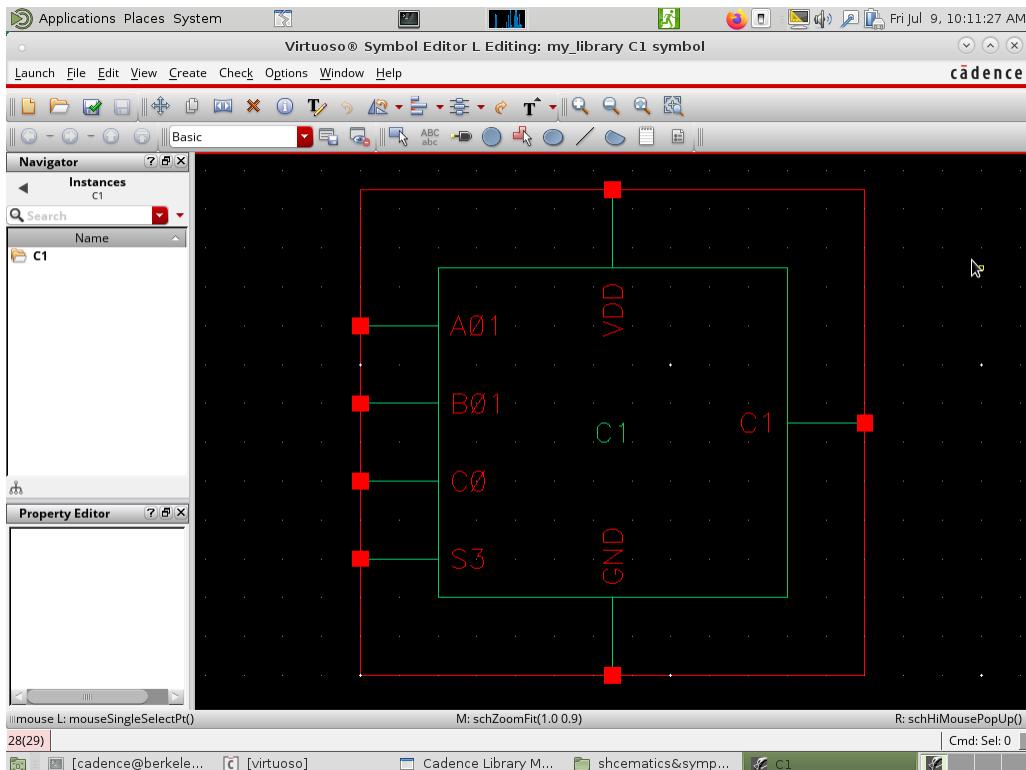
➤ C1 & C2

The circuits are identical but the difference is in the inputs and therefore the output

- Schematic

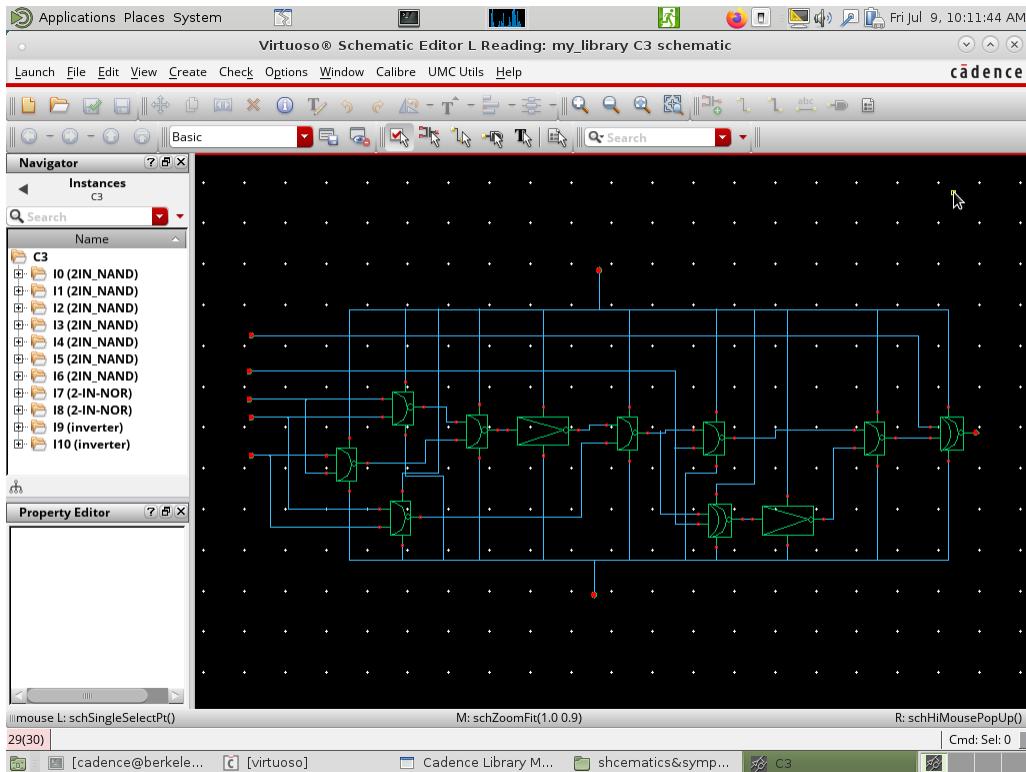


- Block

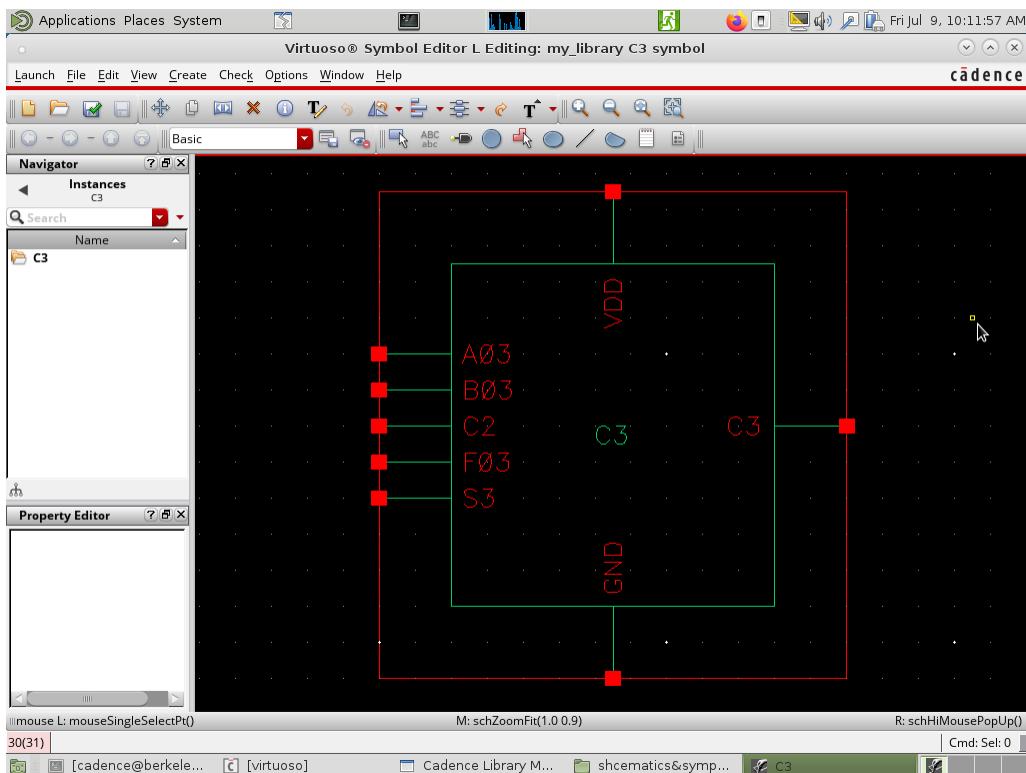


➤ C3

- Schematic

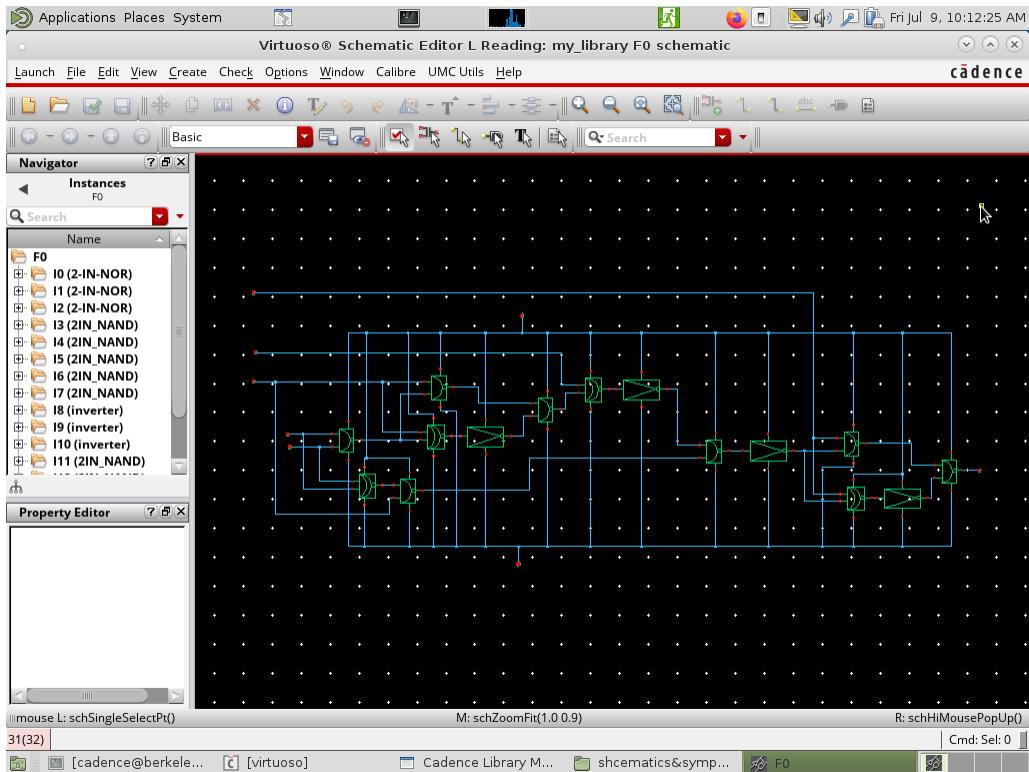


- Block

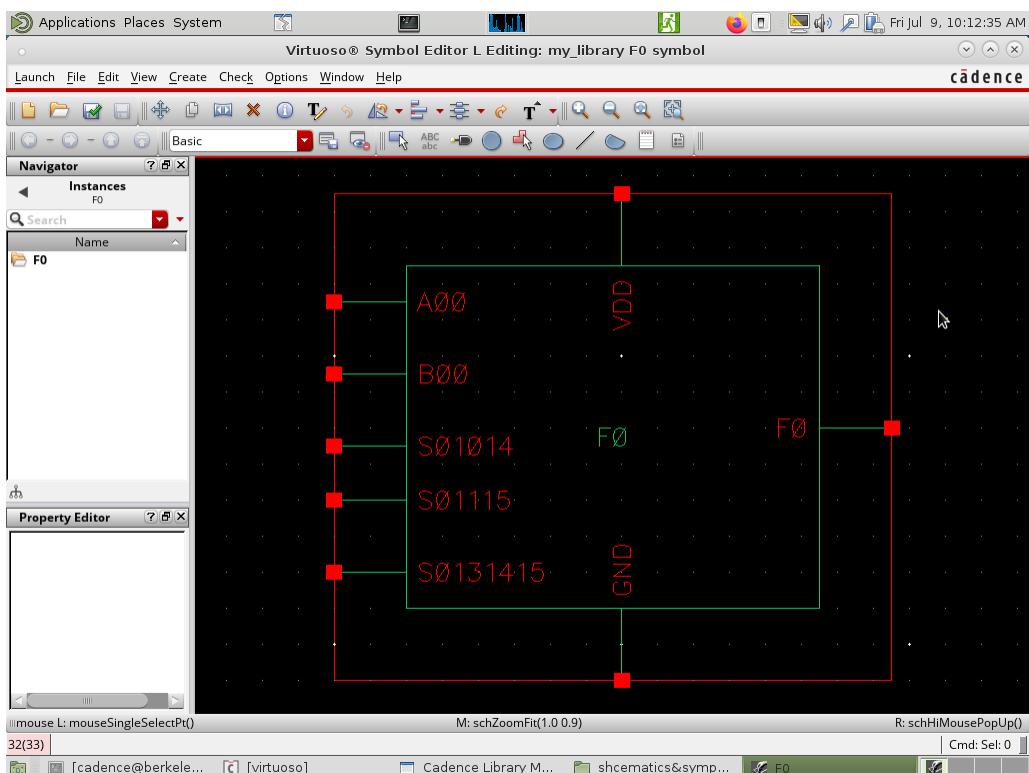


➤ F0

- Schematic



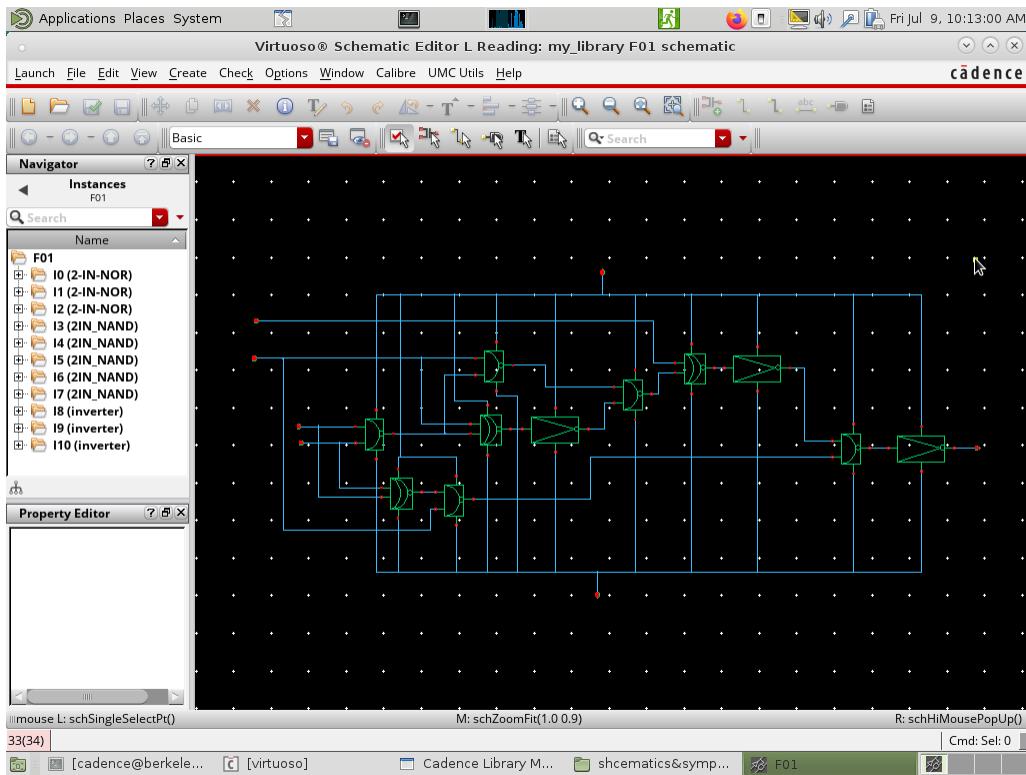
- Block



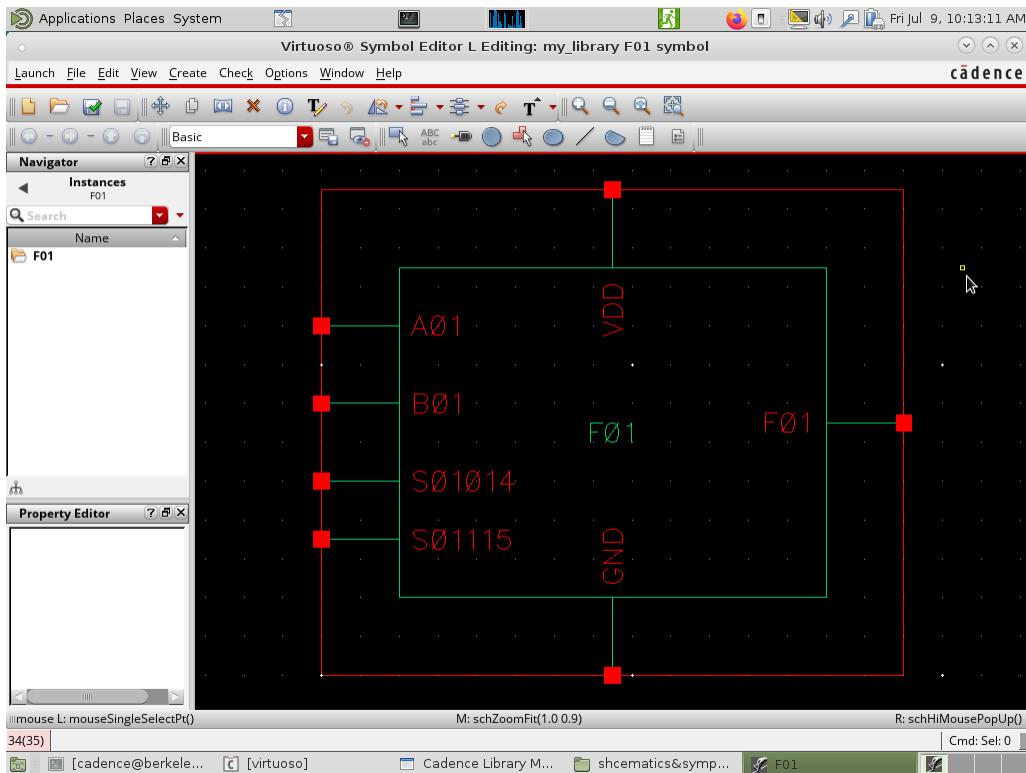
➤ F01 to F03

The circuits are identical but the difference is in the inputs and therefore the output

- Schematic



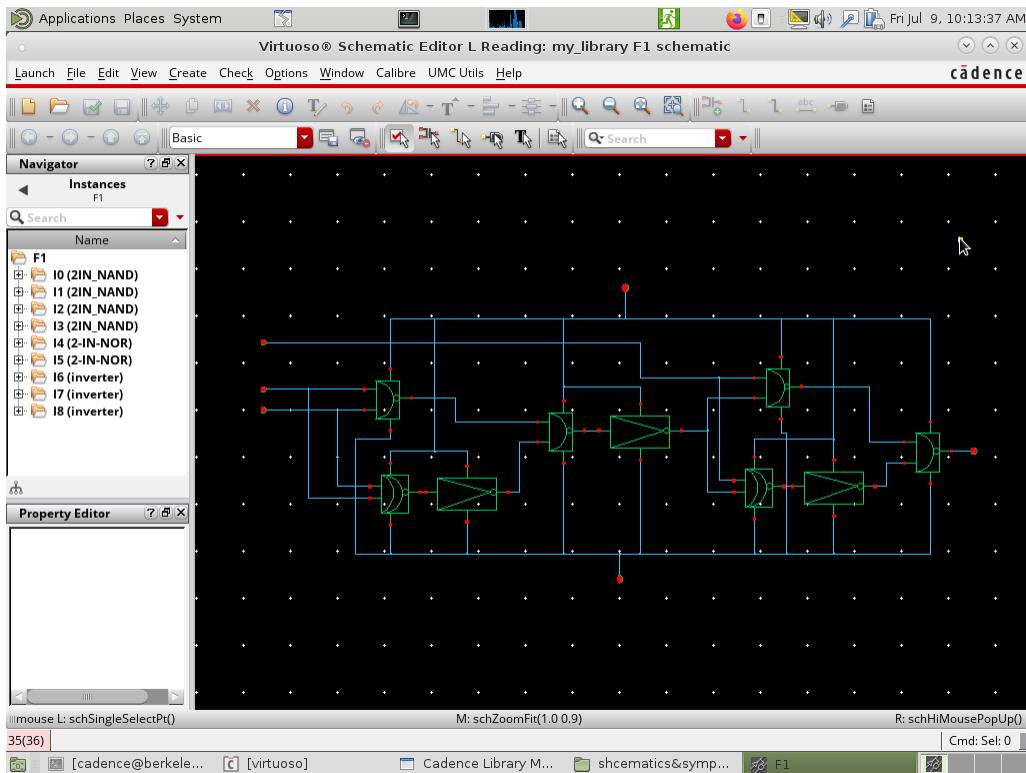
- Block



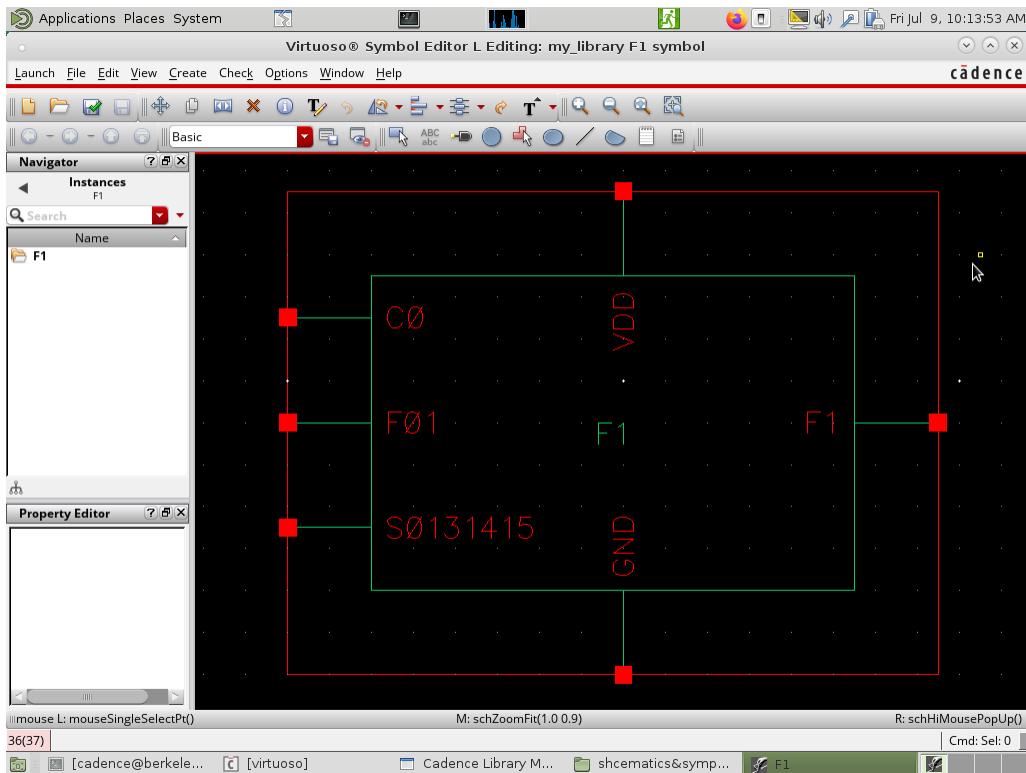
➤ F1 to F3

The circuits are identical but the difference is in the inputs and therefore the output

- Schematic

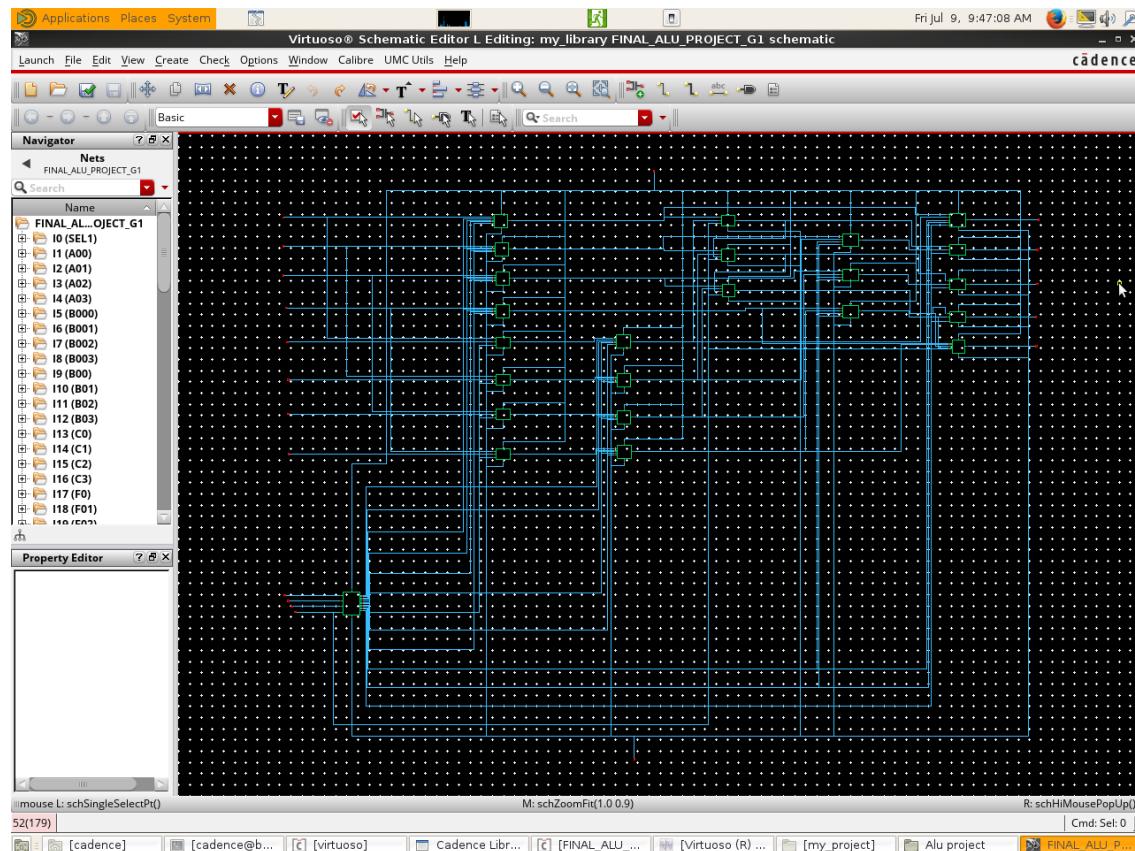


- Block

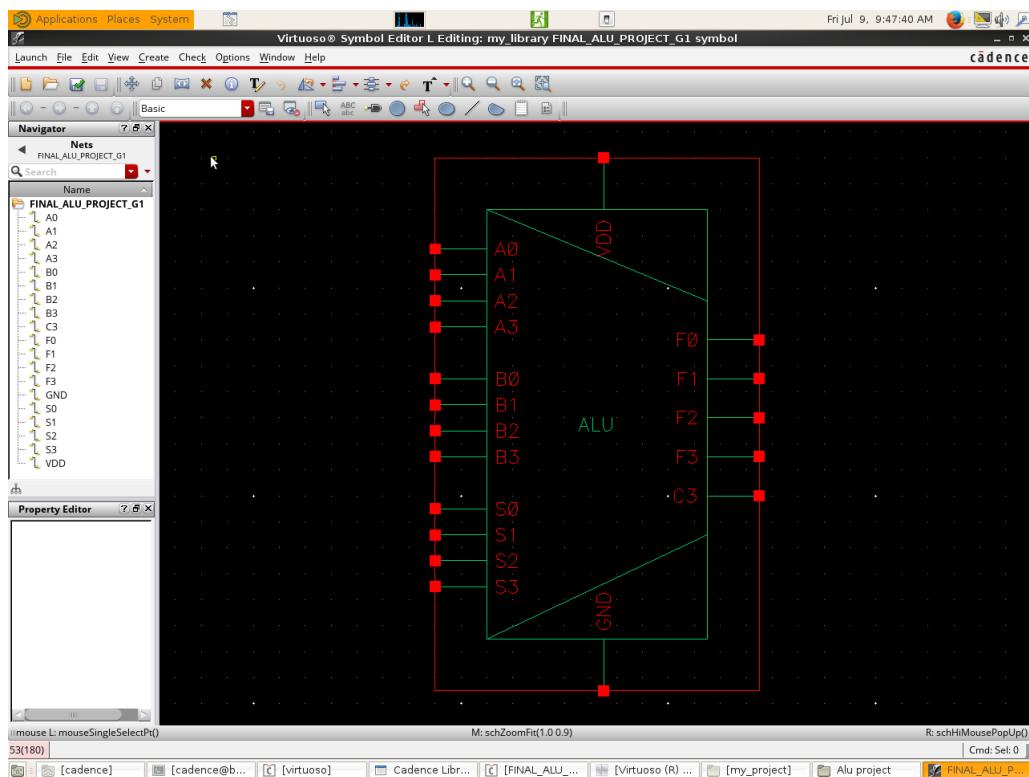


4. Cadence (All Blocks connected together)

- Schematic



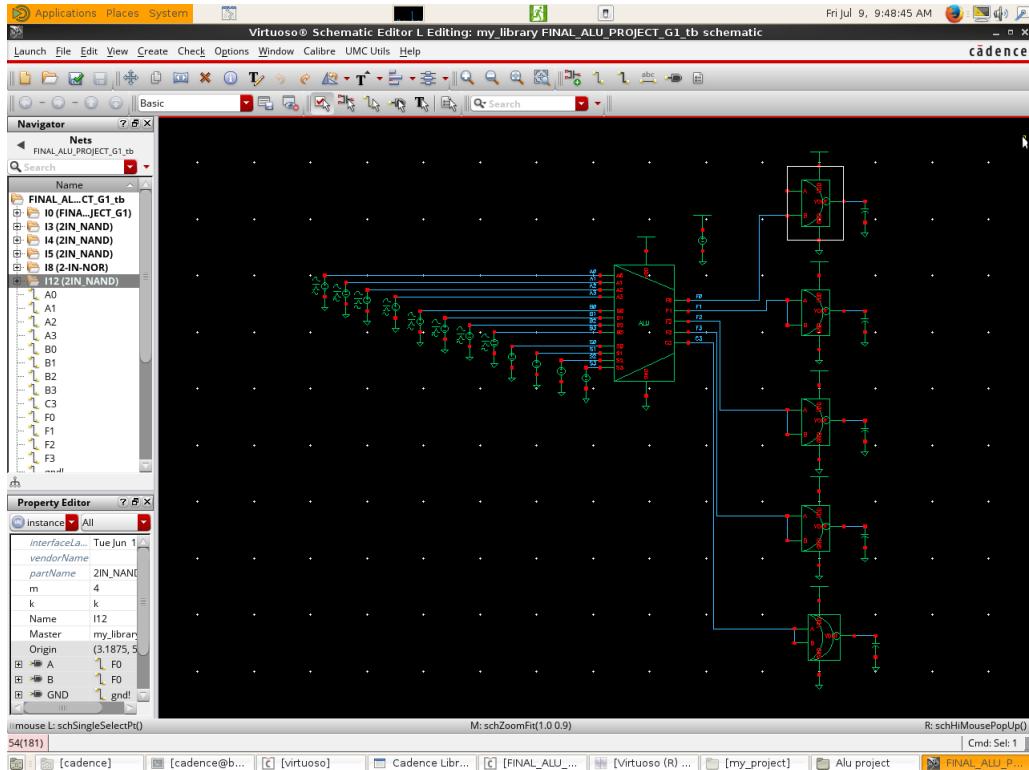
- Block



- Test Bench

We take a copy of the last stage and make it to be (Fan-out of 4) and took the outputs before it.

Note: you can see the value of ($m = 4$) in the picture.



5. Cadence (ALU output waveforms)

All possible outputs at every selection input (from 0000 to 1111)

Note: you can see the selection input in the pictures.

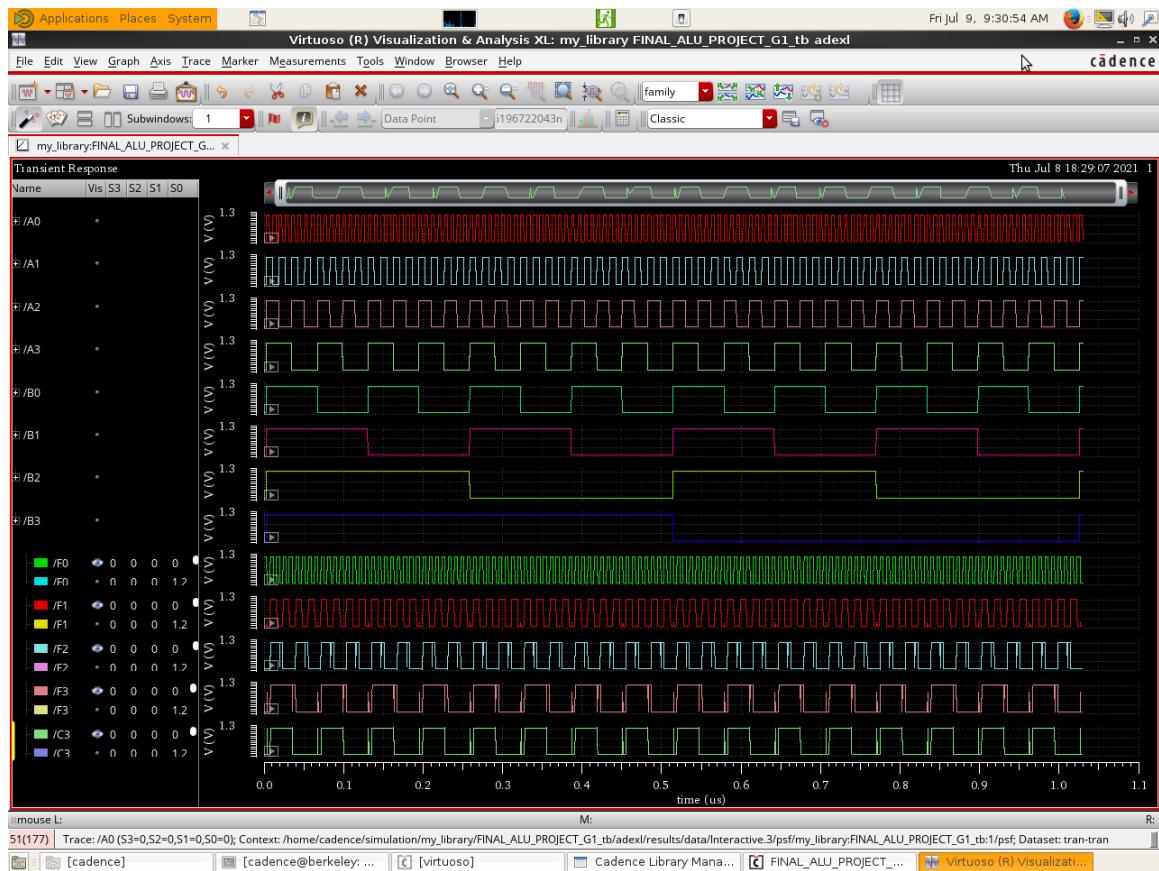
The input pulse duration:

$A_0 = 8 \text{ ns}$, $A_1 = 16 \text{ ns}$, $A_2 = 32 \text{ ns}$, $A_3 = 64 \text{ ns}$, $B_0 = 128 \text{ ns}$, $B_1 = 256 \text{ ns}$,

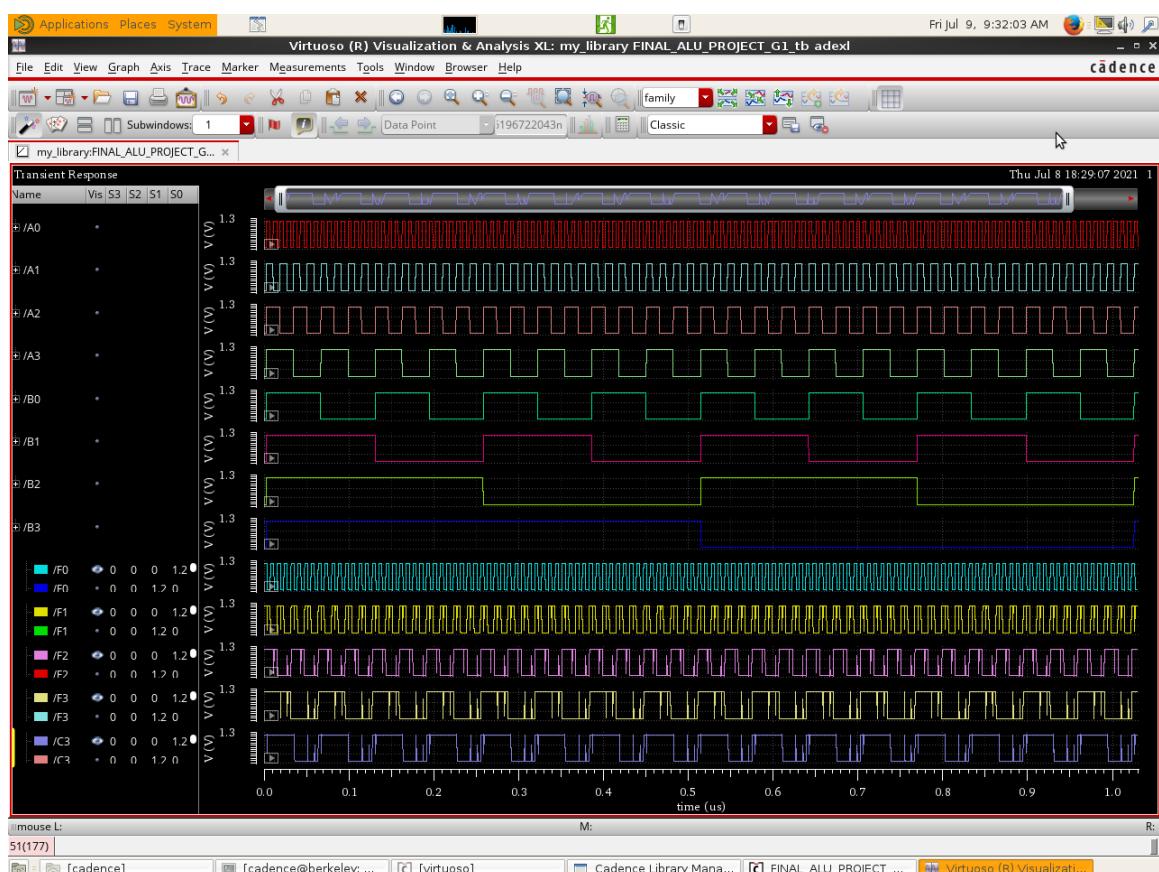
$B_2 = 512 \text{ ns}$, $B_3 = 1024 \text{ ns}$

We make this to draw all of the possible outputs.

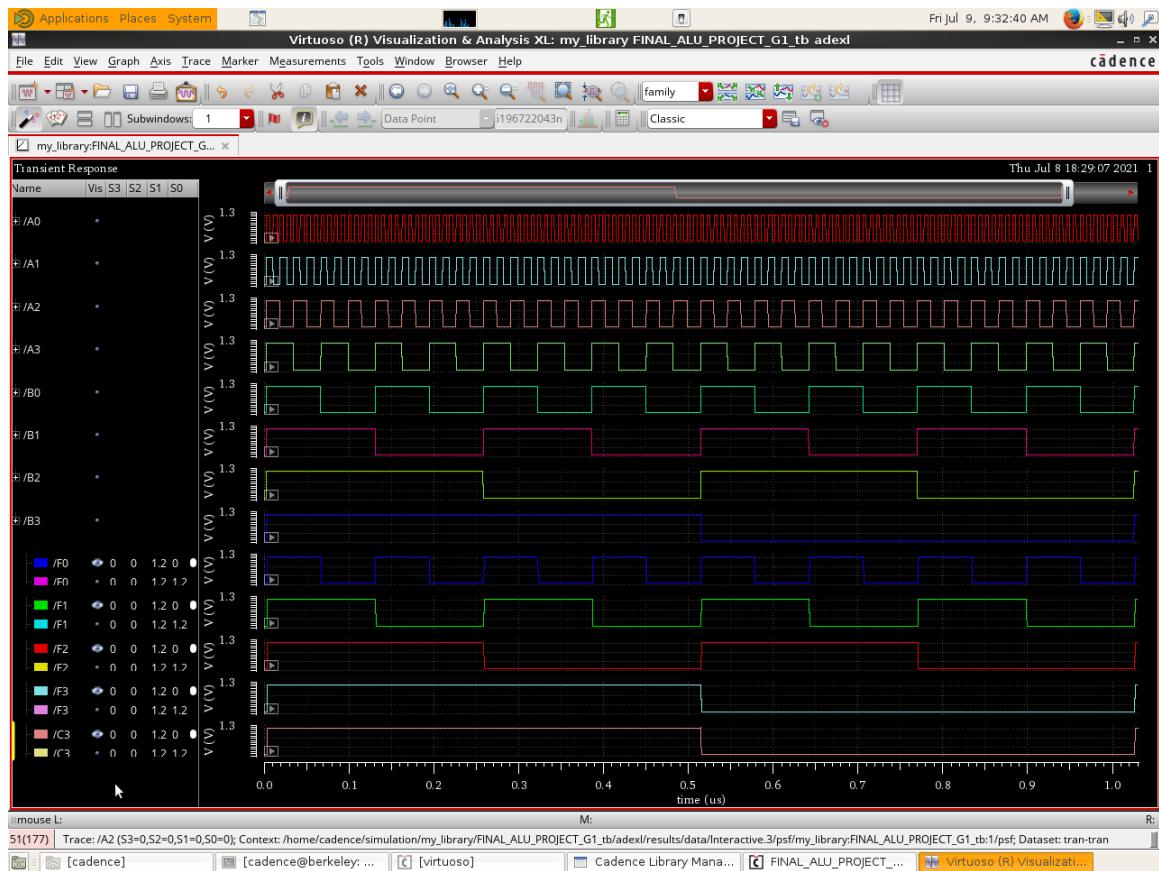
- There're acceptable glitches.



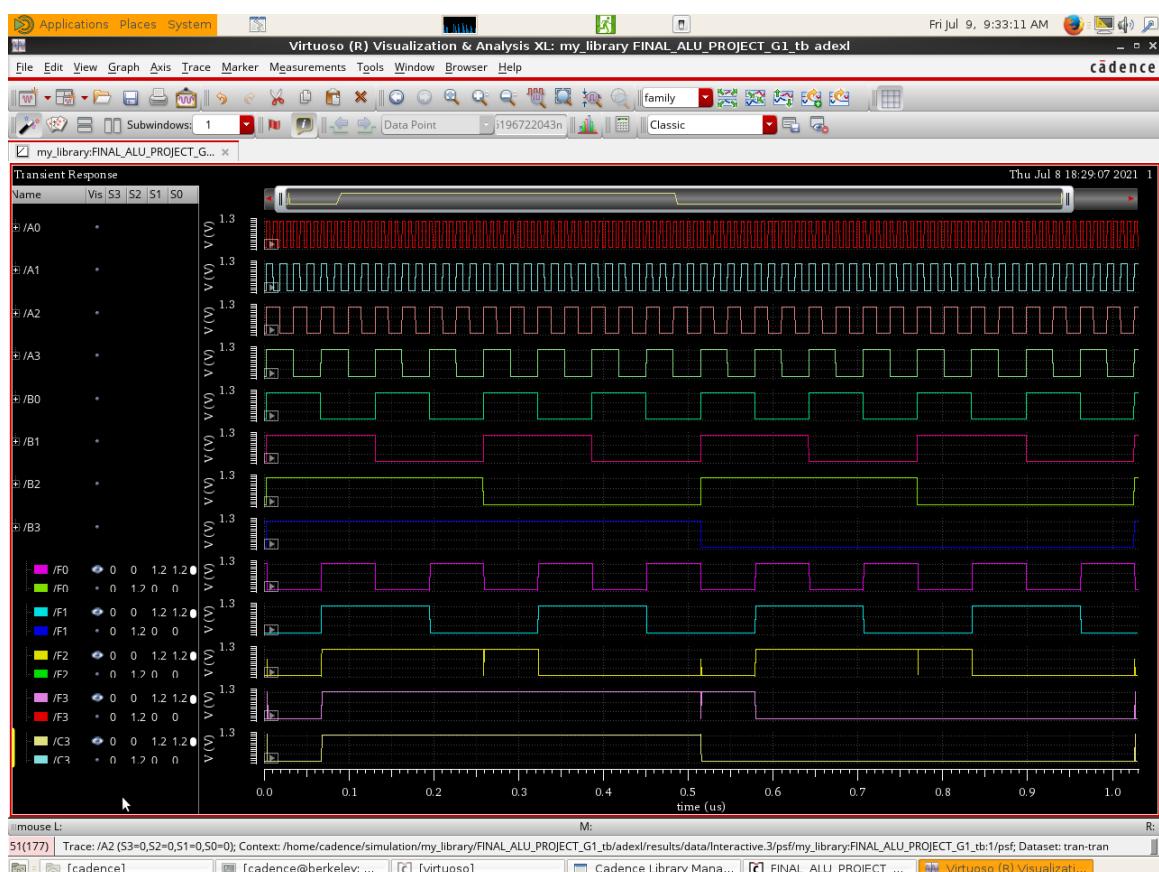
0000



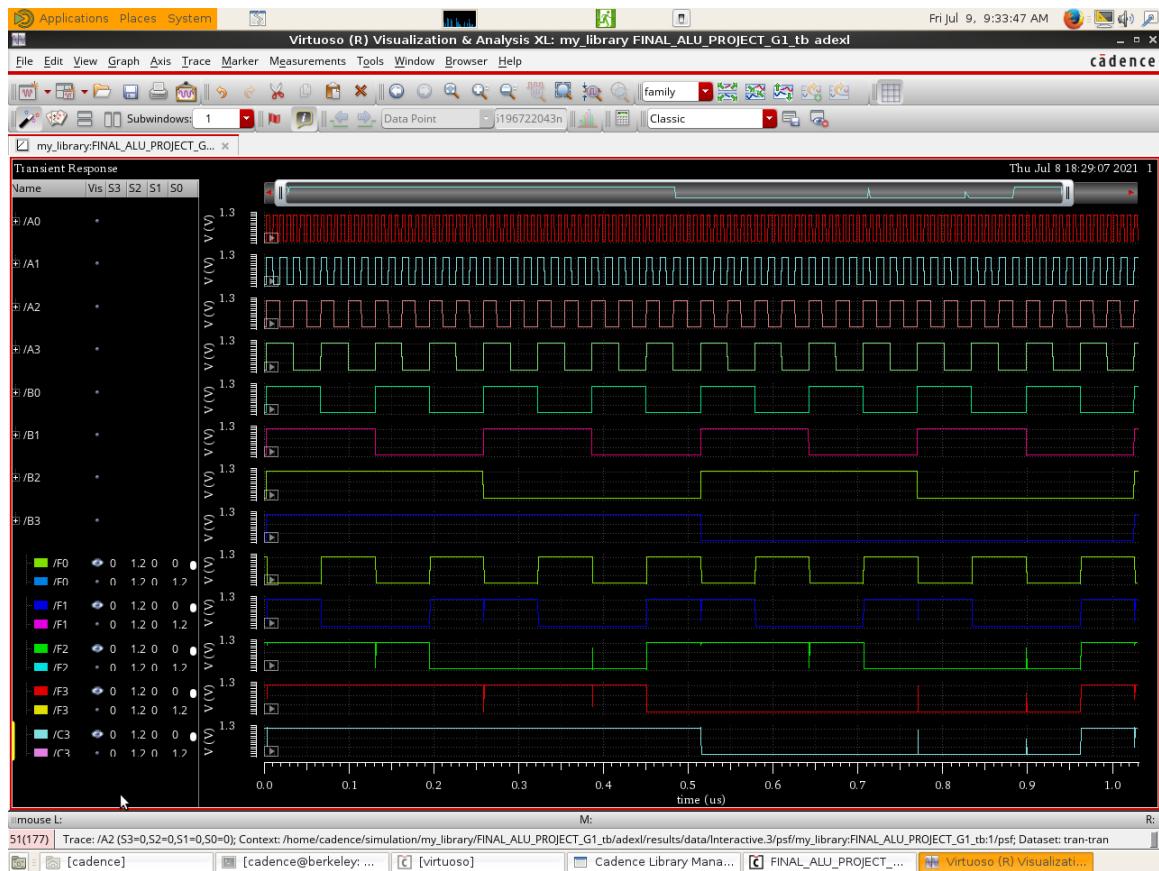
0001



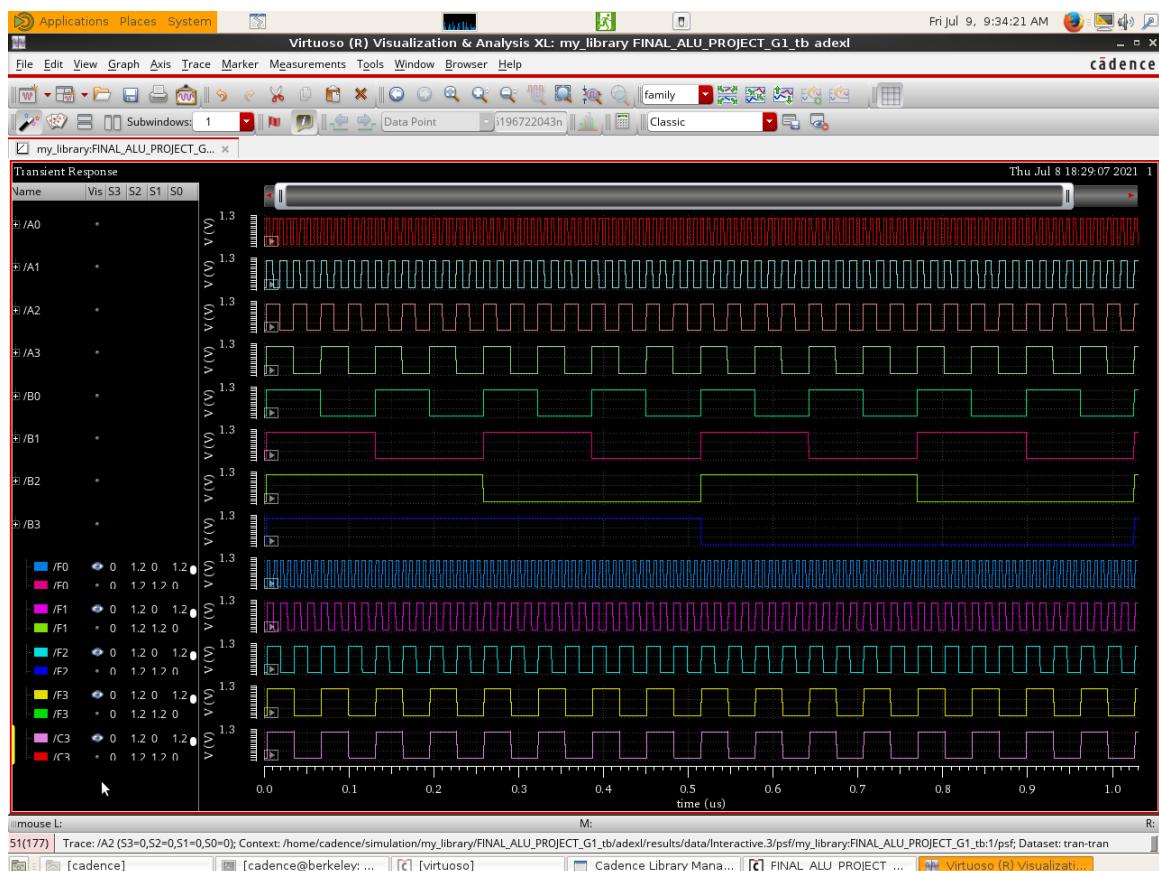
0010



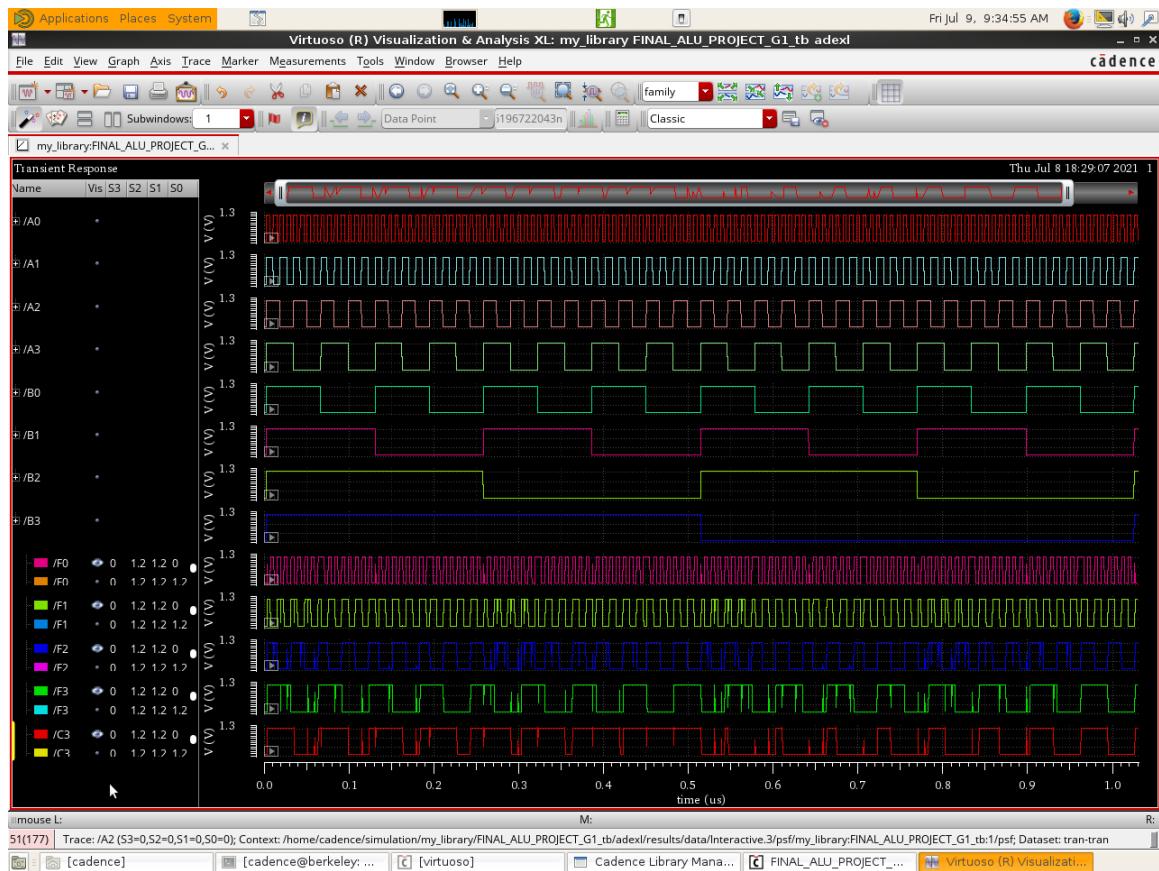
0011



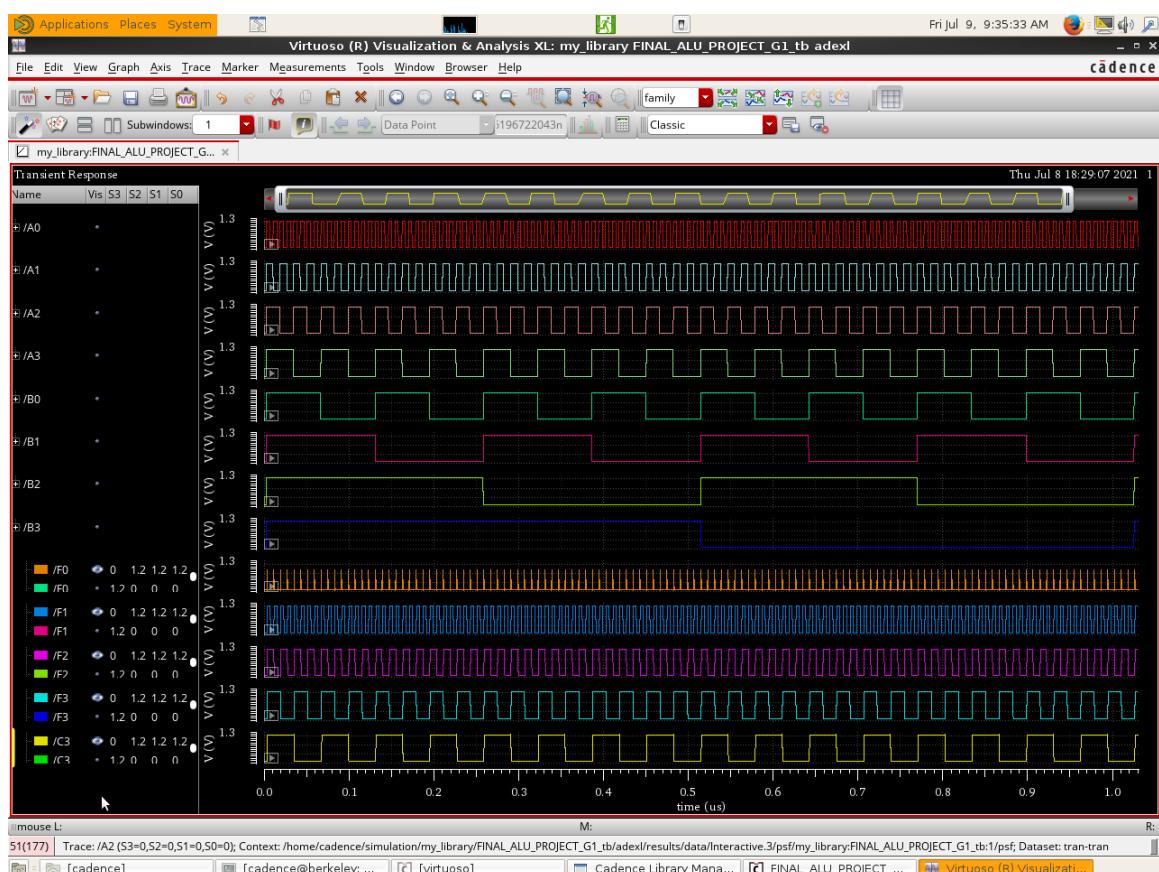
0100



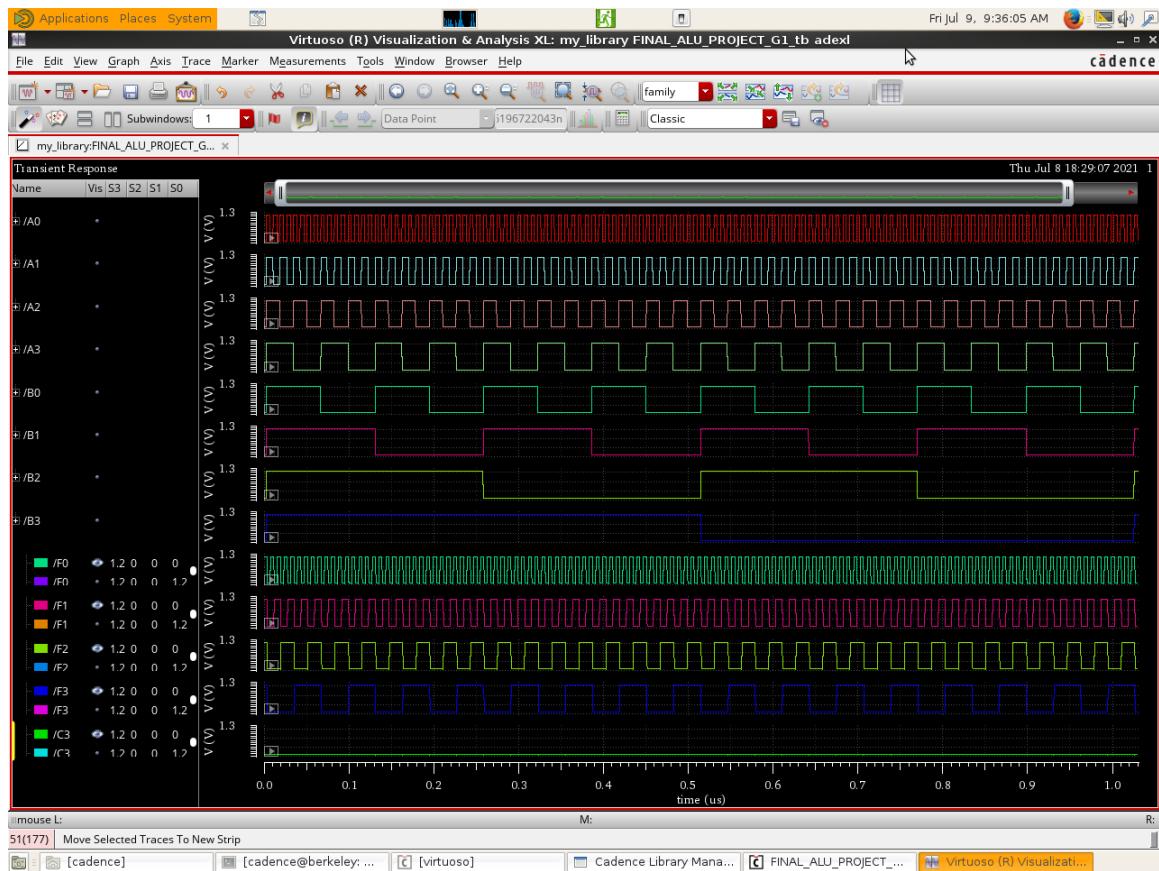
0101



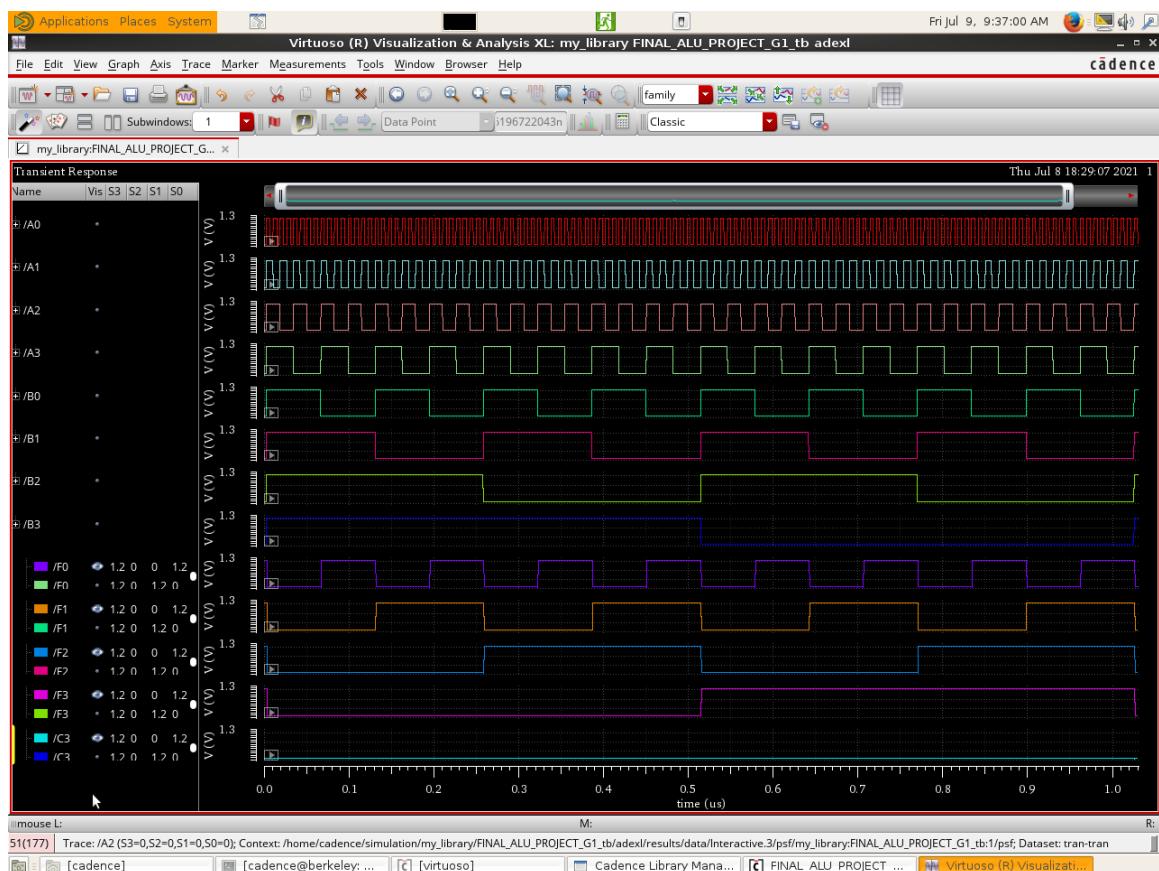
0110



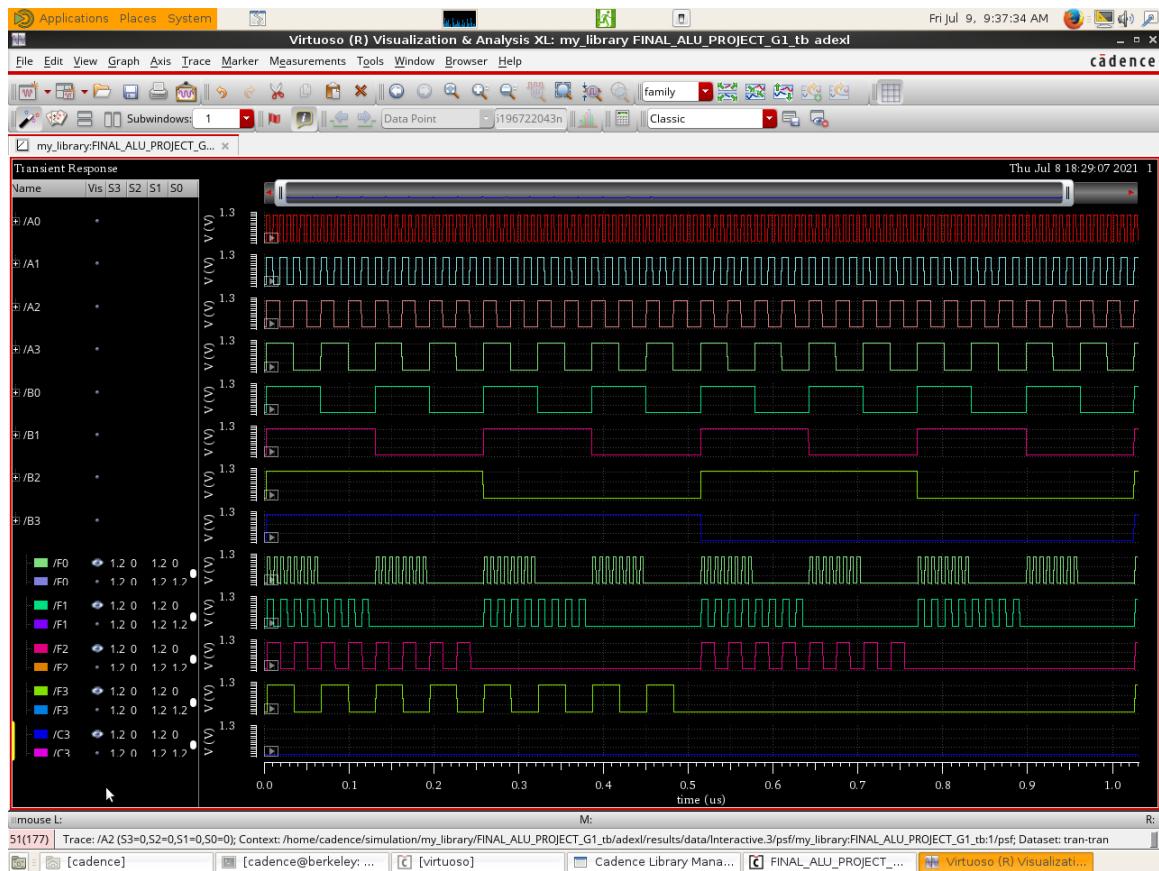
0111



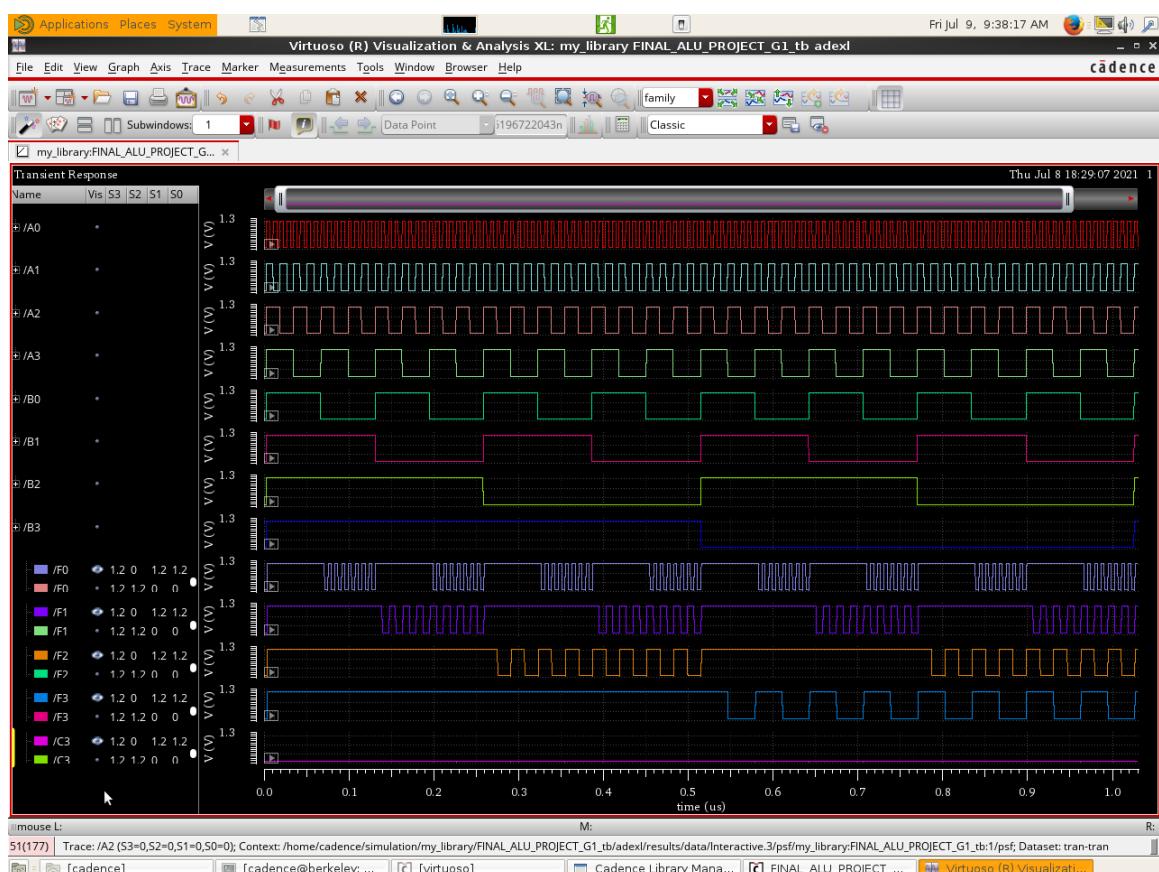
1000



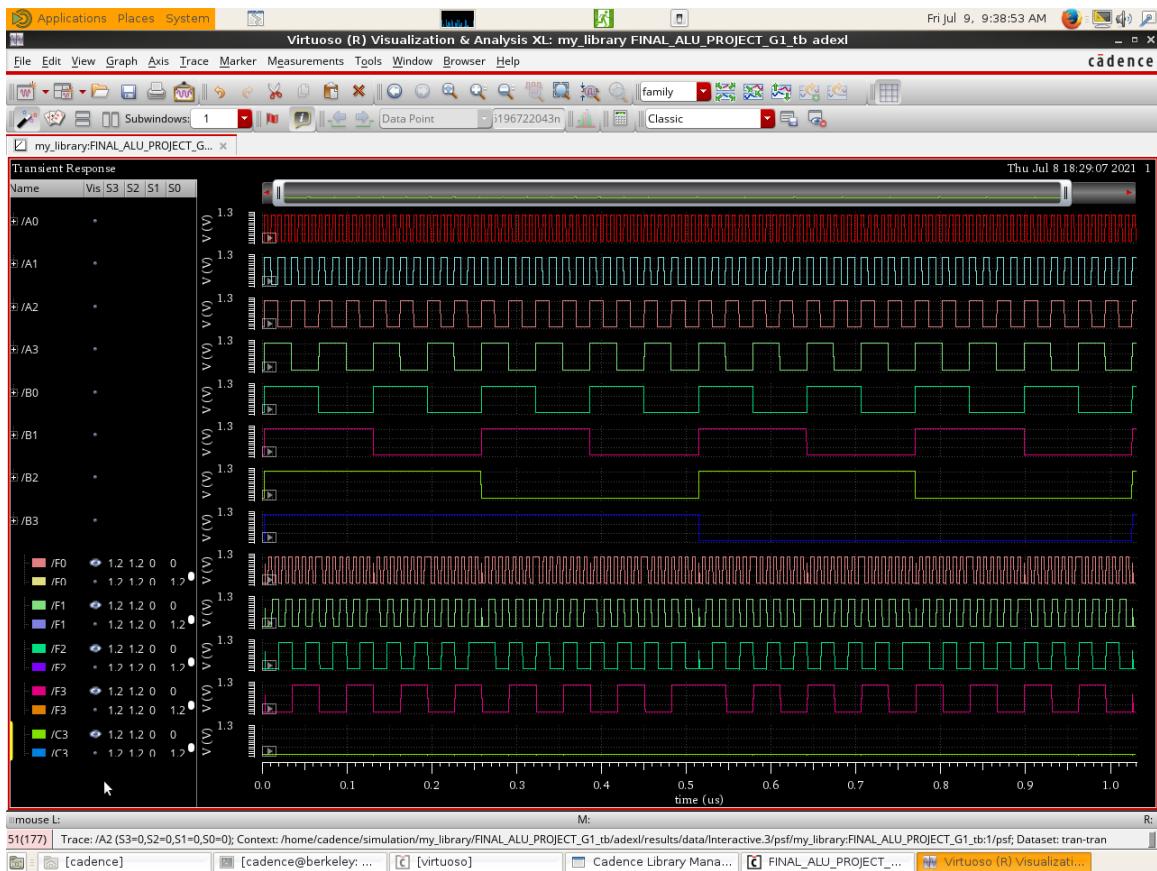
1001



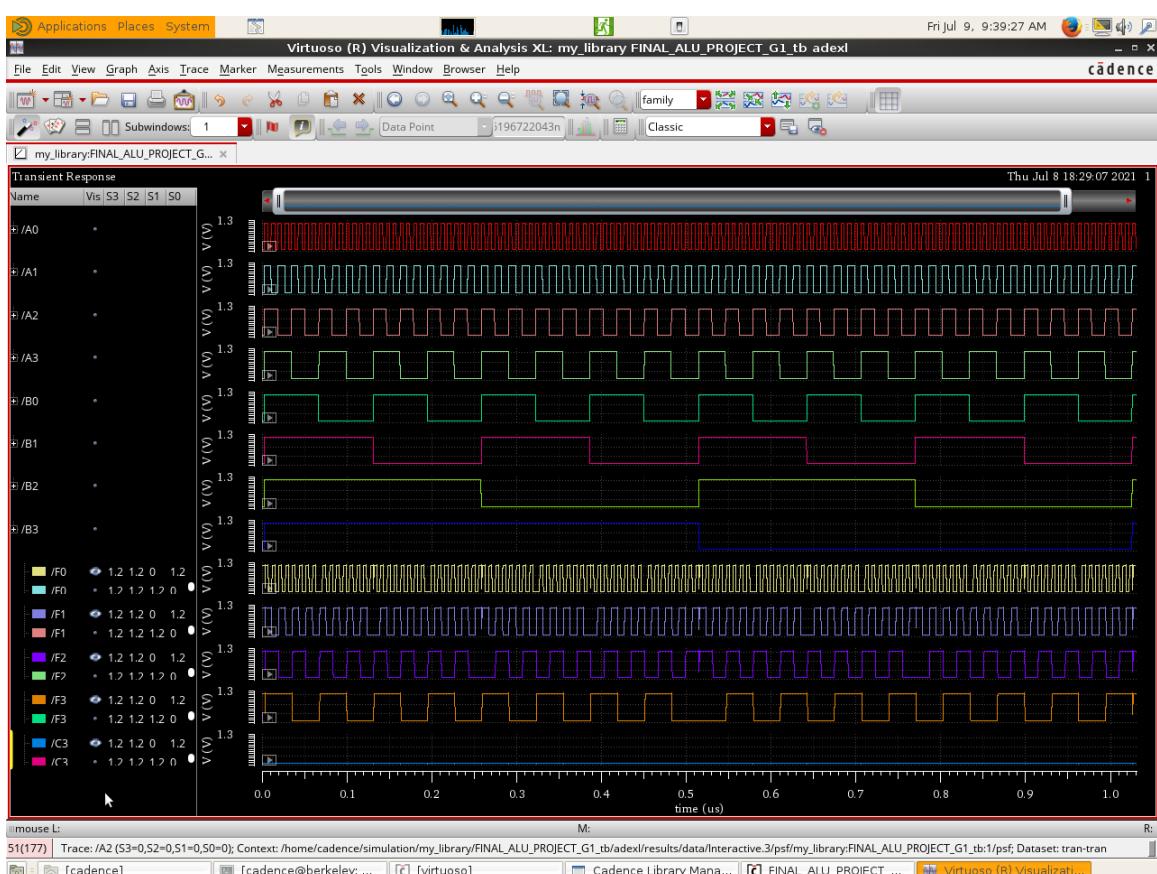
1010



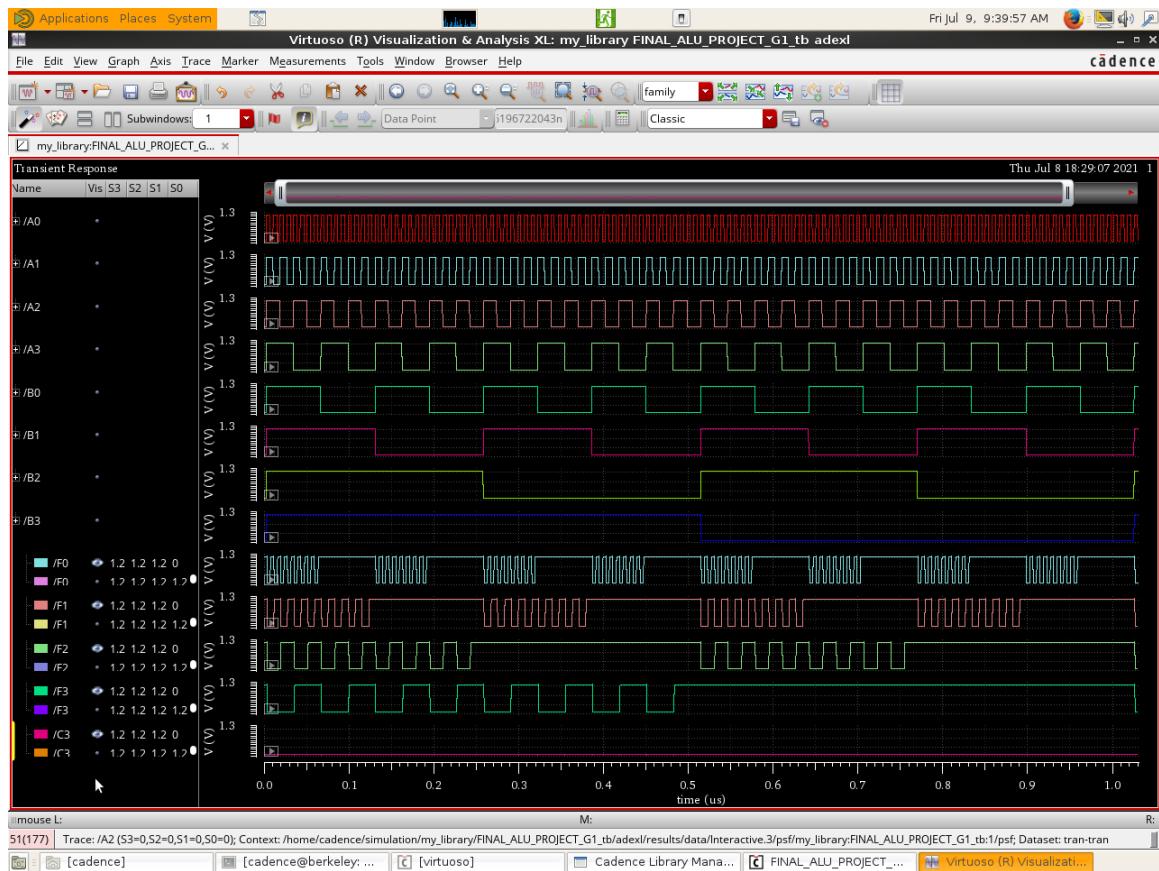
1011



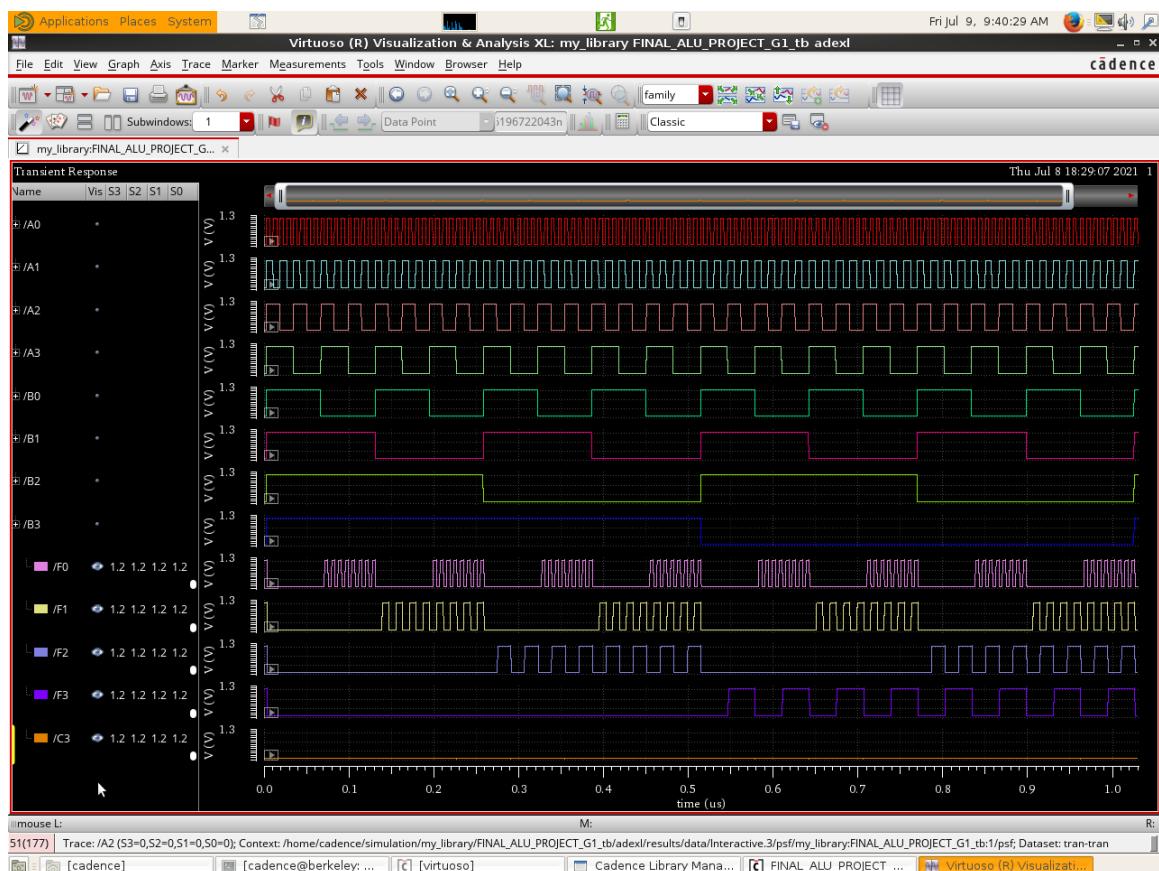
1100



1101



1110



1111

6. Cadence (delay & power calculations)

➤ Delay

The maximum $T_{plh} = 788.2$ ps “occurs when C3 changes because of B0 at selections 0011”

The maximum $T_{phl} = 681.7$ ps “occurs when F3 changes because of A0 at selections 0001”

The minimum $T_{phl} = 261.6$ ps “occurs when F0 changes because of A0 at selections 0101”

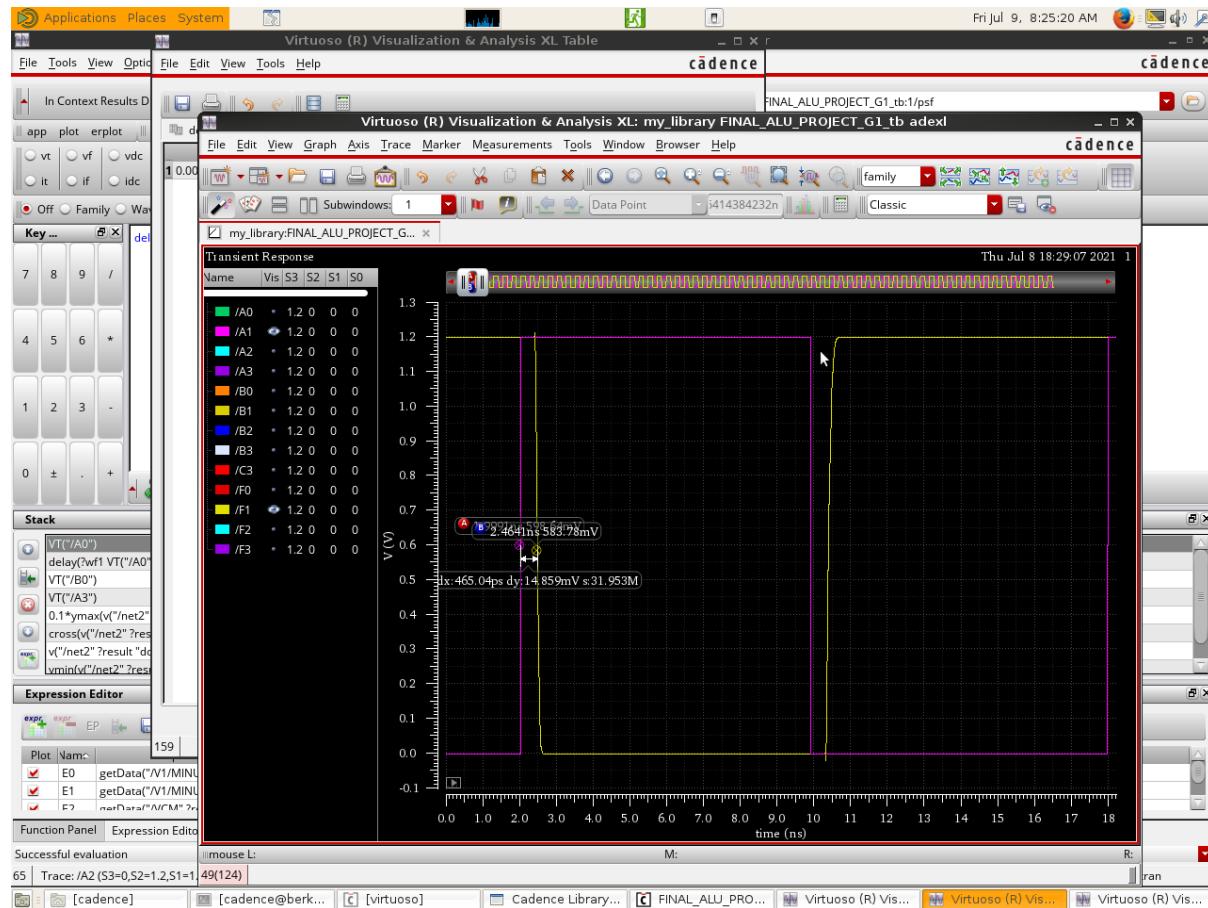
The minimum $T_{plh} = 273.4$ ps “occurs when F0 changes because of A0 at selections 0101”

The maximum $T_p = 678.25$ ps “occurs when F3 changes because of A0 at selections 0001”

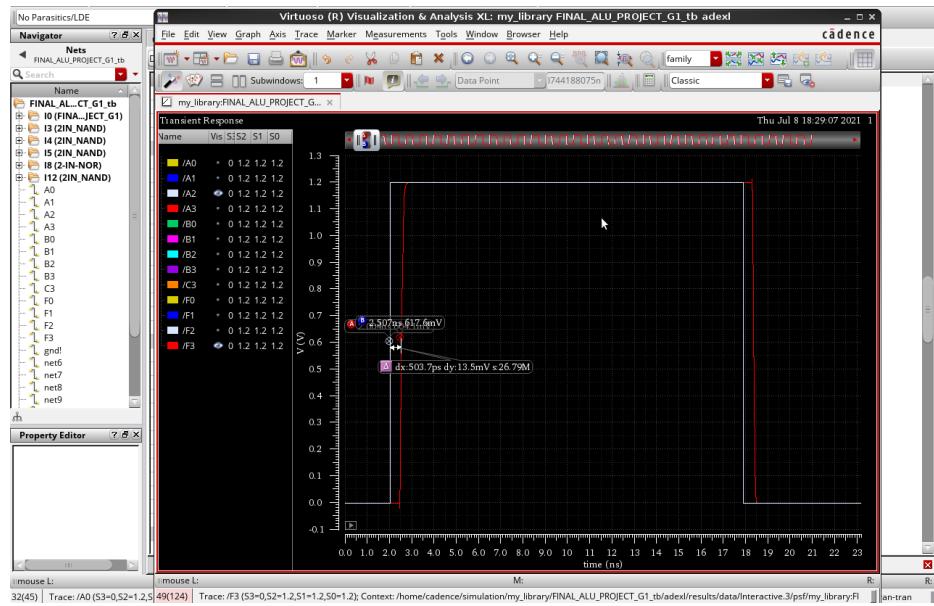
The minimum $T_p = 267.5$ ps “occurs when F0 changes because of A0 at selections 0101”

We take random inputs that make change in the output to calculate the delay and we make an **EXCEL** file contain what we calculate.

- Some Examples:



Manual calculation of the T_{phl} of F1 when A1 changes at selections 1000 = 465 ps



Manual calculation of the Tph of F3 when A2 changes at selections 0111 = 503.7 ps

➤ Consumed Power

We calculate the average power consumed from the calculator of cadence, We use The maximum allowed frequency to calculate the power consumed and this occurs at the maximum delay and our maximum delay was 788.2 ps so we put safety margin and put the max frequency to be $1/(800 \text{ ps}) = 1.25 \text{ GHZ}$.

note: "we put on the all inputs from (A0 to B3) pulse wave with this frequency"

Consumed Power = 3.962 mW

