Project Report

VLSI Design

By Anjali Singh

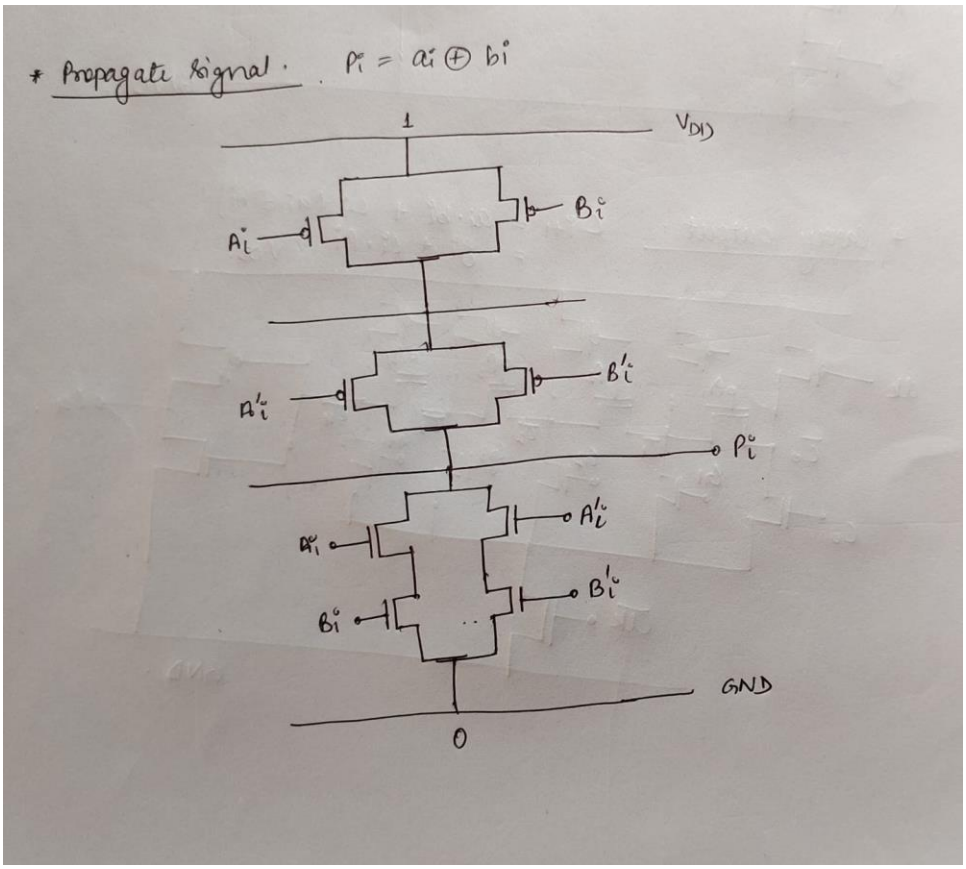
Roll No.: 2020102004

Proposed Structure for the CLA Adder

- I have used the static logic for the propagate and generate block.
- For the carry output and sum output signals, I have used the logic of pass-transistor logic, which is dynamic.
- For the propagate signal, pi = ai XOR bi, used the simple XOR logic for all the four bits.
- Similarly, for the generate signal, gi = ai AND bi, used the simple NAND gate logic and then inverter gate to get the AND output for all the four inputs.
- PTL logic has been used for both carry and sum output signals.
- Carry Output Signals can be given as: -
    - c_in = c0
    - c1 = g0 + c0.p0
    - c2 = g1 + g0.p1 + c0.p0.p1
    - c3 = g2 + g1.p2 + g0.p1.p2 + c0.p0.p1.p2
    - c4 = g3 + g2.p3 + g1.p2.p3 + g0.p1.p2.p3 + c0.p0.p1.p2.p3
- Sum Output Signals: si = ci XOR pi
- For both the sum and carry output signals, there's the use of clock signals as well, which eases in controlling the input and output delay.

Stick Diagrams and NGspice of various Blocks

- **Propagate Signal**
    - **Logic:** pi = ai XOR bi
    - **Stick Diagram**



- A total of 24 NMOS and 24 PMOS transistors are used.
- **NGspice Code**

  CLA bit adder

  .include TSMC_180nm.txt

  .param SUPPLY = 1.8V

  VDD Connections

  vdd 1 0 dc SUPPLY

  Input Initialization

  Va1 2 0 pulse (0 1.8 0 0.1p 0.1p 10n 20n)

  Vb1 3 0 pulse (0 1.8 0 0.1p 0.1p 20n 40n)

Va2 10 0 pulse (0 1.8 0 0.1p 0.1p 10n 20n)

Vb2 11 0 pulse (0 1.8 0 0.1p 0.1p 20n 40n)

Va3 18 0 pulse (0 1.8 0 0.1p 0.1p 10n 20n)

Vb3 19 0 pulse (0 1.8 0 0.1p 0.1p 20n 40n)

Va4 26 0 pulse (0 1.8 0 0.1p 0.1p 10n 20n)

Vb4 27 0 pulse (0 1.8 0 0.1p 0.1p 20n 40n)

(Connections: drain gate source body CMOSN/CMOSP W L)

Propagate Signal For Various Inputs

For Inputs A1 and B1

Inverter Gate

A1 = 2, A1' = 4; B1 = 3, B1' = 5

MP1 4 2 1 1 CMOSP W = 1.8u L = 0.18u

MN1 4 2 0 0 CMOSN W = 0.9u L = 0.18u

MP2 5 3 1 1 CMOSP W = 1.8u L = 0.18u

MN2 5 3 0 0 CMOSN W = 0.9u L = 0.18u

Output: P1 = 7, Inputs: A1 = 2, B1 = 3, Vcc = 1, GND = 0

MP3 6 2 1 1 CMOSP W = 1.8u L = 0.18u

MP4 6 3 1 1 CMOSP W = 1.8u L = 0.18u

MP5 7 4 6 1 CMOSP W = 1.8u L = 0.18u

MP6 7 5 6 1 CMOSP W = 1.8u L = 0.18u

MN3 7 2 8 0 CMOSN W = 0.9u L = 0.18u

MN4 7 4 9 0 CMOSN W = 0.9u L = 0.18u

MN5 8 3 0 0 CMOSN W = 0.9u L = 0.18u

MN6 9 5 0 0 CMOSN W = 0.9u L = 0.18u

C1 7 0 100f

For Inputs A2 and B2

Inverter Gate

A2 = 10, A2' = 12; B2 = 11, B2' = 13

MP7 12 10 1 1 CMOSP W = 1.8u L = 0.18u

MN7 12 10 0 0 CMOSN W = 0.9u L = 0.18u

MP8 13 11 1 1 CMOSP W = 1.8u L = 0.18u

MN8 13 11 0 0 CMOSN W = 0.9u L = 0.18u

Output: P2 = 15, Inputs: A2 = 10, B2 = 11, Vcc = 1, GND = 0

MP9 14 10 1 1 CMOSP W = 1.8u L = 0.18u

MP10 14 11 1 1 CMOSP W = 1.8u L = 0.18u

MP11 15 12 14 1 CMOSP W = 1.8u L = 0.18u

MP12 15 13 14 1 CMOSP W = 1.8u L = 0.18u

MN9 15 10 16 0 CMOSN W = 0.9u L = 0.18u

MN10 15 12 17 0 CMOSN W = 0.9u L = 0.18u

MN11 16 11 0 0 CMOSN W = 0.9u L = 0.18u

MN12 17 13 0 0 CMOSN W = 0.9u L = 0.18u

C2 15 0 100f

For Inputs A3 and B3

Inverter Gate

A3 = 18, A3' = 20; B3 = 19, B3' = 21

MP13 20 18 1 1 CMOSP W = 1.8u L = 0.18u

MN13 20 18 0 0 CMOSN W = 0.9u L = 0.18u

MP14 21 19 1 1 CMOSP W = 1.8u L = 0.18u

MN14 21 19 0 0 CMOSN W = 0.9u L = 0.18u

Output: P3 = 23, Inputs: A3 = 18, B3 = 19, Vcc = 1, GND = 0

MP15 22 18 1 1 CMOSP W = 1.8u L = 0.18u

MP16 22 19 1 1 CMOSP W = 1.8u L = 0.18u

MP17 23 20 22 1 CMOSP W = 1.8u L = 0.18u

MP18 23 21 22 1 CMOSP W = 1.8u L = 0.18u

MN15 23 18 24 0 CMOSN W = 0.9u L = 0.18u

MN16 24 19 0 0 CMOSN W = 0.9u L = 0.18u

MN17 23 20 25 0 CMOSN W = 0.9u L = 0.18u

MN18 25 21 0 0 CMOSN W = 0.9u L = 0.18u

C3 23 0 100f

For Inputs A4 and B4

Inverter Gate

A4 = 26, A4' = 28; B4 = 27, B4' = 29

MP19 28 26 1 1 CMOSP W = 1.8u L = 0.18u

```
MN19 28 26 0 0 CMOSN W = 0.9u L = 0.18u
MP20 29 27 1 1 CMOSP W = 1.8u L = 0.18u
MN20 29 27 0 0 CMOSN W = 0.9u L = 0.18u

*Output: P4 = 31, Inputs: A3 = 26, B3 = 27, Vcc = 1, GND = 0*
MP21 30 26 1 1 CMOSP W = 1.8u L = 0.18u
MP22 30 27 1 1 CMOSP W = 1.8u L = 0.18u
MP23 31 28 1 1 CMOSP W = 1.8u L = 0.18u
MP24 31 29 30 1 CMOSP W = 1.8u L = 0.18u
MN21 31 26 32 0 CMOSN W = 0.9u L = 0.18u
MN22 32 27 0 0 CMOSN W = 0.9u L = 0.18u
MN23 31 28 33 0 CMOSN W = 0.9u L = 0.18u
MN24 33 29 0 0 CMOSN W = 0.9u L = 0.18u
C4 31 0 100f

.tran 0.1u 1u

.control
run
**plot background color**
set color0 = white
**grid lines color**
set color1 = black
**plot 1 color**
```

set color2 = blue

plot 2 color

set color3 = red

plot 3 color

set color4 = brown

plot 4 color

set color5 = green

set xbrushwidth = 3.5

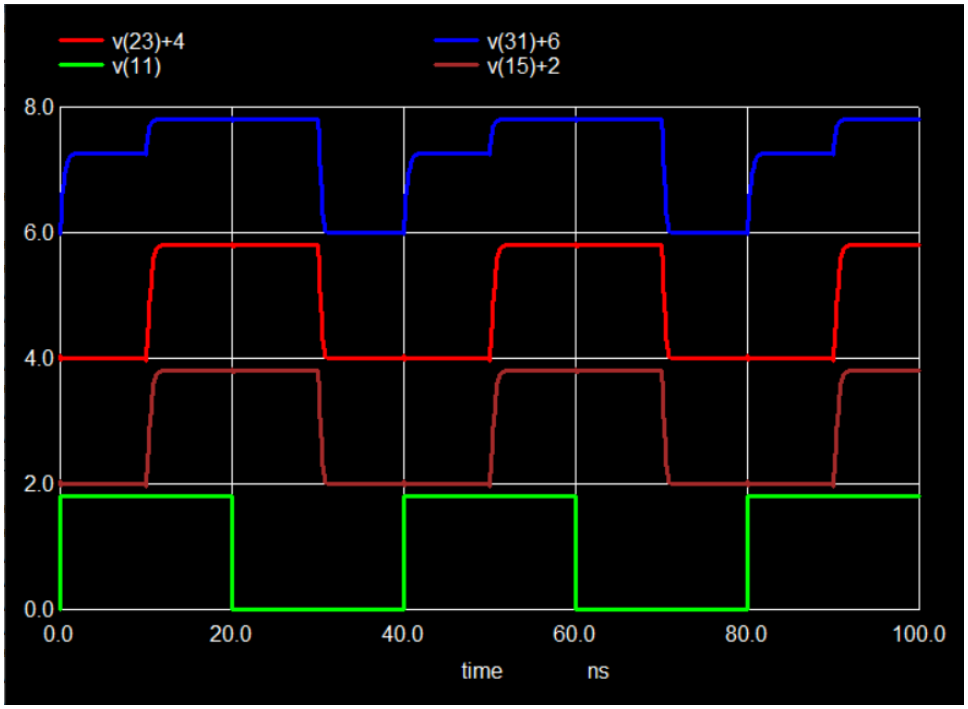
set curplottitle= Anjali-Singh-2020102004-Propagate-Output

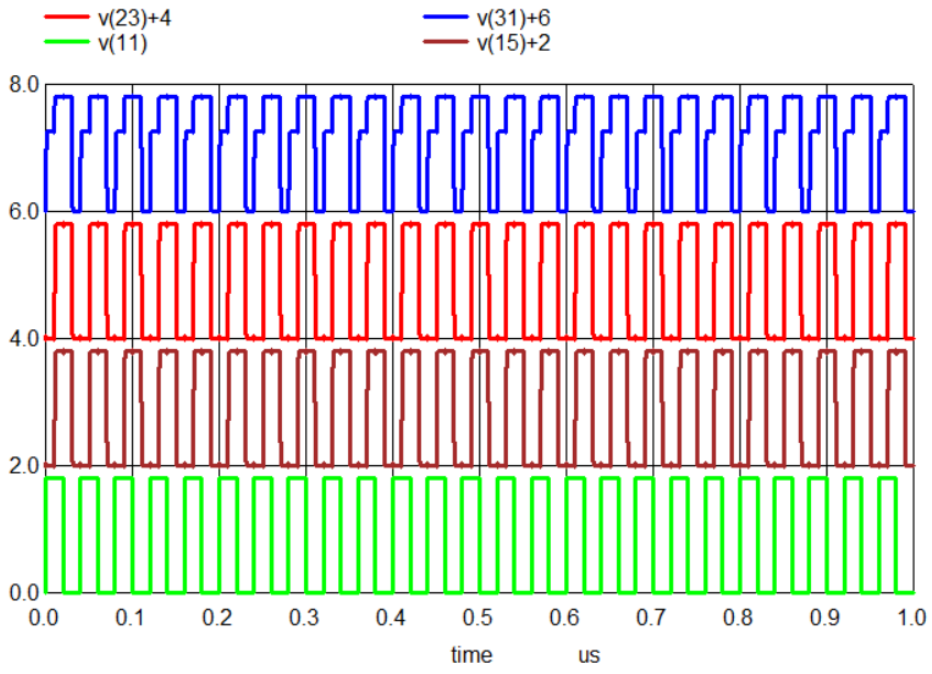
plot v(31)+6 v(23)+4 v(15)+2 v(11)

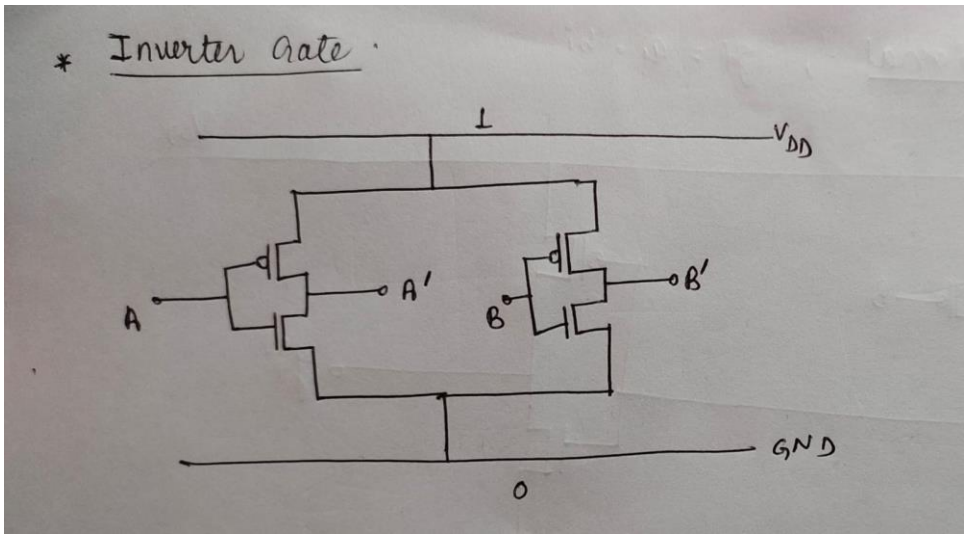
.endc

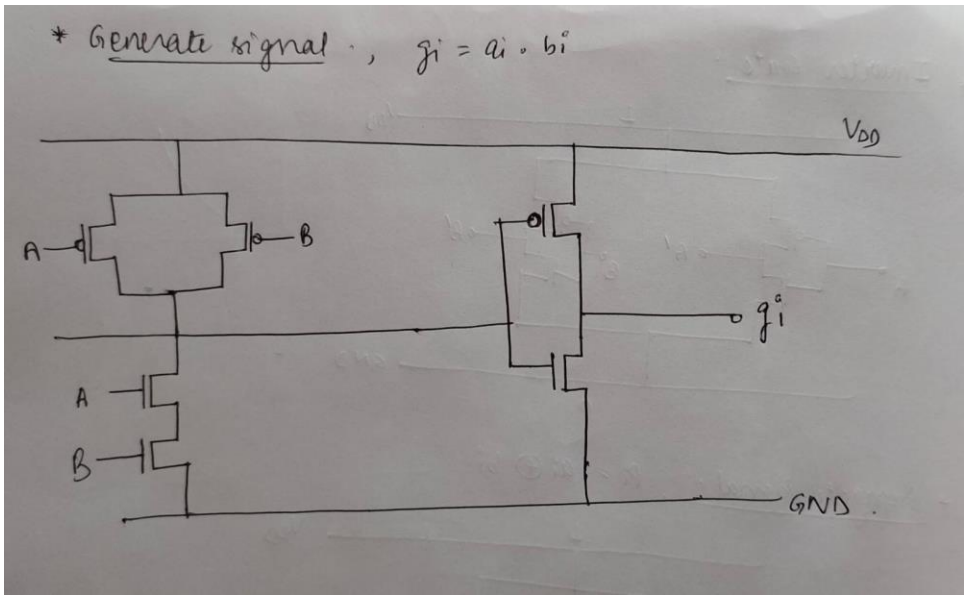
.end

- **Plot Waveform**
- .tran 0.1n 100n
-



- .tran 0.1u 1u
- 

- **Generate Signal**
  - **Logic:** gi = ai AND bi
  - **Stick Diagram**
  - 

- 

- A total of 36 NMOS and 36 PMOS have been used till now (including propagate signal as well)

- **NGspice Code**

  Generate Signal

  .include TSMC_180nm.txt

  .include propagate.cir

  .option TEMP=27C

  (Connections: drain gate source body CMOSN/CMOSP W L)

  Generate Signal for Various Inputs

  For Inputs A1 and B1

  Output: G1; Inputs: A1 = 2, B1 = 3

  MP25 34 2 1 1 CMOSP W = 1.8u L = 0.18u

MP26 34 3 1 1 CMOSP W = 1.8u L = 0.18u

MN25 34 2 35 0 CMOSN W = 0.9u L = 0.18u

MN26 35 3 0 0 CMOSN W = 0.9u L = 0.18u

Inverter Part

MP27 G1 34 1 1 CMOSP W = 1.8u L = 0.18u

MN27 G1 34 0 0 CMOSN W = 0.9u L = 0.18u

C5 G1 0 100f

For Inputs A2 and B2

Output: G2; Inputs: A2 = 10, B2 = 11

MP28 36 10 1 1 CMOSP W = 1.8u L = 0.18u

MP29 36 11 1 1 CMOSP W = 1.8u L = 0.18u

MN28 36 10 37 0 CMOSN W = 0.9u L = 0.18u

MN29 37 11 0 0 CMOSN W = 0.9u L = 0.18u

Inverter Part

MP30 G2 36 1 1 CMOSP W = 1.8u L = 0.18u

MN30 G2 36 0 0 CMOSN W = 0.9u L = 0.18u

C6 G2 0 100f

For Inputs A3 and B3

Output: G3; Inputs: A3 = 18, B3 = 19

MP31 38 18 1 1 CMOSP W = 1.8u L = 0.18u

MP32 38 19 1 1 CMOSP W = 1.8u L = 0.18u

MN31 38 18 39 0 CMOSN W = 0.9u L = 0.18u

MN32 39 19 0 0 CMOSN W = 0.9u L = 0.18u

Inverter Part

MP33 G3 38 1 1 CMOSP W = 1.8u L = 0.18u

MN33 G3 38 0 0 CMOSN W = 0.9u L = 0.18u

C7 G3 0 100f

For Inputs A4 and B4

Output: G4; Inputs: A4 = 18, B4 = 19

MP34 40 26 1 1 CMOSP W = 1.8u L = 0.18u

MP35 40 27 1 1 CMOSP W = 1.8u L = 0.18u

MN34 40 26 41 0 CMOSN W = 0.9u L = 0.18u

MN35 41 27 0 0 CMOSN W = 0.9u L = 0.18u

Inverter Part

MP36 G4 40 1 1 CMOSP W = 1.8u L = 0.18u

MN36 G4 40 0 0 CMOSN W = 0.9u L = 0.18u

C8 G4 0 100f

36 PMOS, 36 NMOS Used till now

.tran 0.1u 1u

.control

run

set color0=white

set color1=black

set color2=blue

set color3=red

set color4=brown

set xbrushwidth=3.5

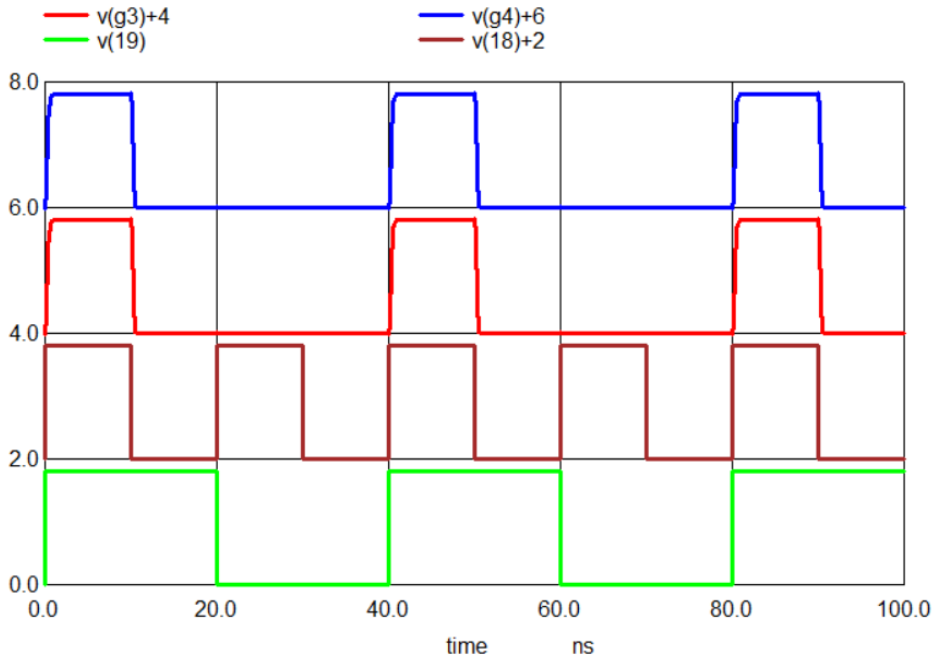
set curplottitle= Anjali-Singh-2020102004-Generate-Output

plot v(G4)+6 v(G3)+4 v(18)+2 v(19)

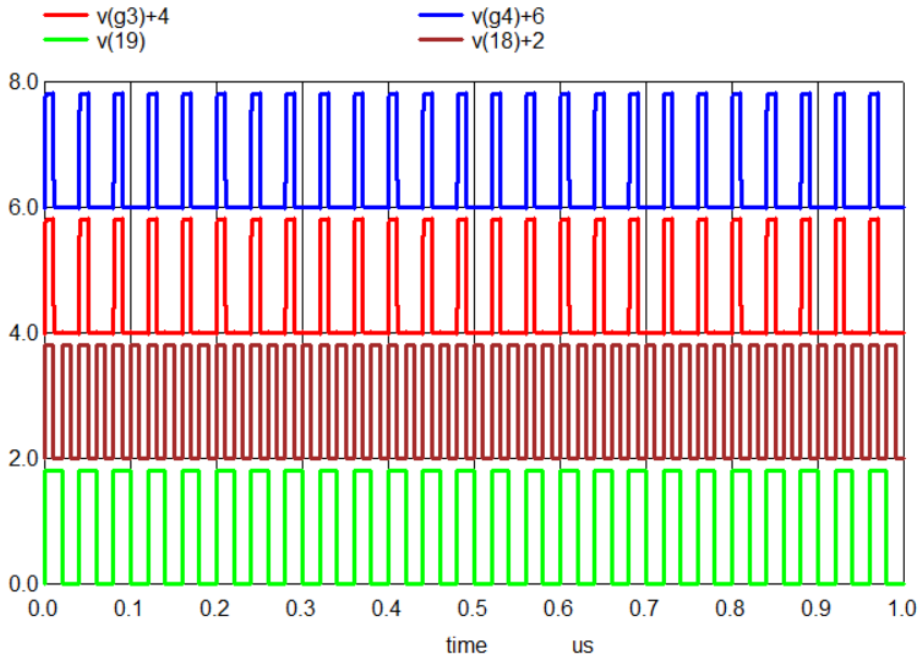
.endc

.end

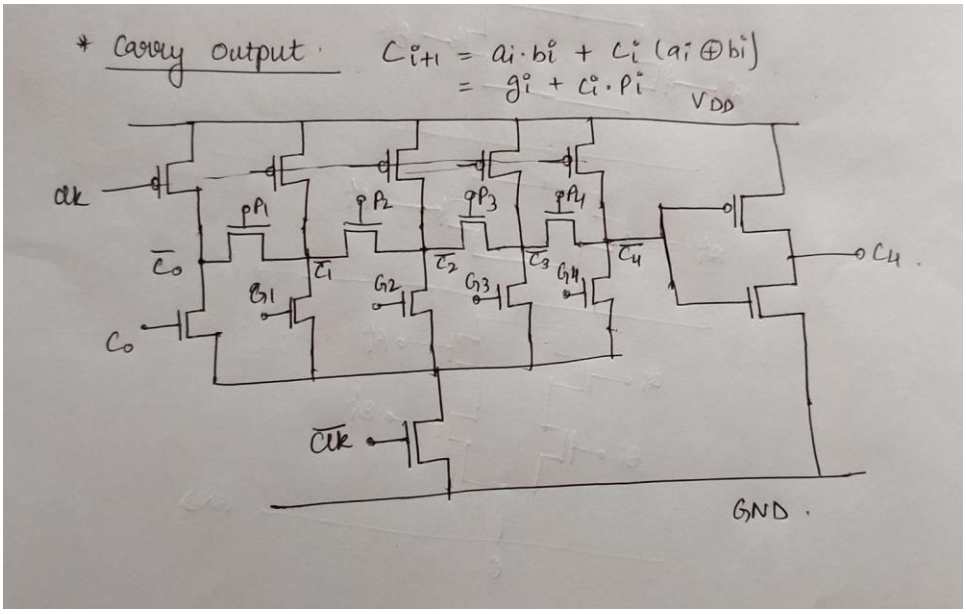
- **Plot Waveform**
- .tran 0.1n 100n
- Input pins: 18, 19; Output pins: g3, g4;



- .tran 0.1u 1u



- **Carry Output**
  - **Logic:** c_{i+1} = ai.bi + ci.(ai XOR bi) = gi + ci.pi
  - **Stick Diagram**



  - A total of 47 PMOS and 52 NMOS transistors have been used till now, including everything from above.

- **NGspice Code**

.include generate.cir

Input Initialization

vcin 49 0 pulse (0 1.8 0 0.1n 0.1n 10n 20n)

vc 52 0 pulse (0 1.8 0 0.1n 0.1n 10n 20n)

(Connections: drain gate source body CMOSN/CMOSP W L)

Inverter Gate

For CLOCK (CLK)

Output: CLK' = 53; Input: CLK = 52

MP37 53 52 1 1 CMOSP W = 1.8u L = 0.18u

MN37 53 52 0 0 CMOSN W = 0.9u L = 0.18u

For Cin

Output: Cin = 49; Input: Cin' = 42

MP38 42 49 1 1 CMOSP W = 1.8u L = 0.18u

MN38 42 49 0 0 CMOSN W = 0.9u L = 0.18u

Main Circuit

Outputs: C1' = 43, C2' = 44, C3' = 45, C4 = 48

Inputs: CLK = 52, Cin = 49, CLK' = 53, G1, G2, G3, G4, P1 = 7, P2 = 15, P3 = 23, P4 = 31

MP39 42 52 1 1 CMOSP W = 1.8u L = 0.18u

MP40 43 52 1 1 CMOSP W = 1.8u L = 0.18u

MP41 44 52 1 1 CMOSP W = 1.8u L = 0.18u

MP42 45 52 1 1 CMOSP W = 1.8u L = 0.18u

MP43 46 52 1 1 CMOSP W = 1.8u L = 0.18u

Using Inputs of Propagate Signal

MN39 43 7 42 0 CMOSN W = 0.9u L = 0.18u

MN40 44 15 43 0 CMOSN W = 0.9u L = 0.18u

MN41 45 23 44 0 CMOSN W = 0.9u L = 0.18u

MN42 46 31 45 0 CMOSN W = 0.9u L = 0.18u

Using Inputs of Generate Signal

MN43 42 49 47 0 CMOSN W = 0.9u L = 0.18u

MN44 43 G1 47 0 CMOSN W = 0.9u L = 0.18u

MN45 44 G2 47 0 CMOSN W = 0.9u L = 0.18u

MN46 45 G3 47 0 CMOSN W = 0.9u L = 0.18u

MN47 46 G4 47 0 CMOSN W = 0.9u L = 0.18u

Inverting the clock signal

MN48 47 53 0 0 CMOSN W = 0.9u L = 0.18u

Inverting the Outputs C1', C2', C3', C4'

For C4, Output: C4 = 48; Input: C4' = 46

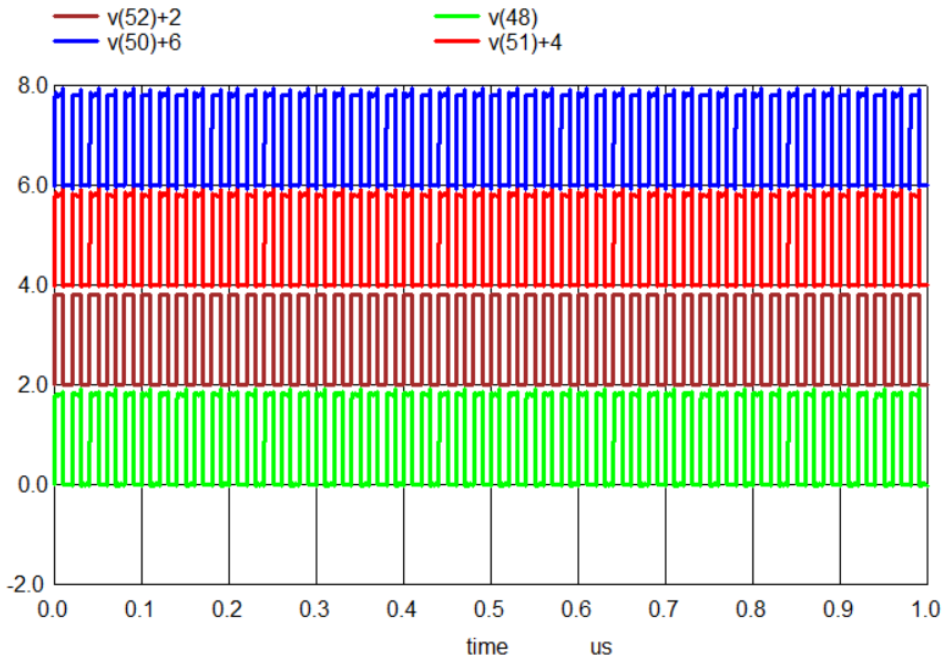
MP44 48 46 1 1 CMOSP W = 1.8u L = 0.18u

```
MN49 48 46 0 0 CMOSN W = 0.9u L = 0.18u
*For C1, Output: C1 = 50; Input: C1' = 43*
MP45 50 43 1 1 CMOSP W = 1.8u L = 0.18u
MN50 50 43 0 0 CMOSN W = 0.9u L = 0.18u
*For C2, Output: C2 = 51; Input: C2' = 44*
MP46 51 44 1 1 CMOSP W = 1.8u L = 0.18u
MN51 51 44 0 0 CMOSN W = 0.9u L = 0.18u
*For C3, Output: C3 = 52; Input: C3' = 45*
MP47 52 45 1 1 CMOSP W = 1.8u L = 0.18u
MN52 52 45 0 0 CMOSN W = 0.9u L = 0.18u

**No. of PMOS = 47, NMOS = 52**

.tran 0.1u 1u
.control
run
set color0=white
set color1=black
set color2=blue
set color3=red
set color4=brown
set xbrushwidth=3.5
set curplottitle= Anjali-Singh-2020102004-Carry-Output
plot v(50)+6 v(51)+4 v(52)+2 v(48)
```

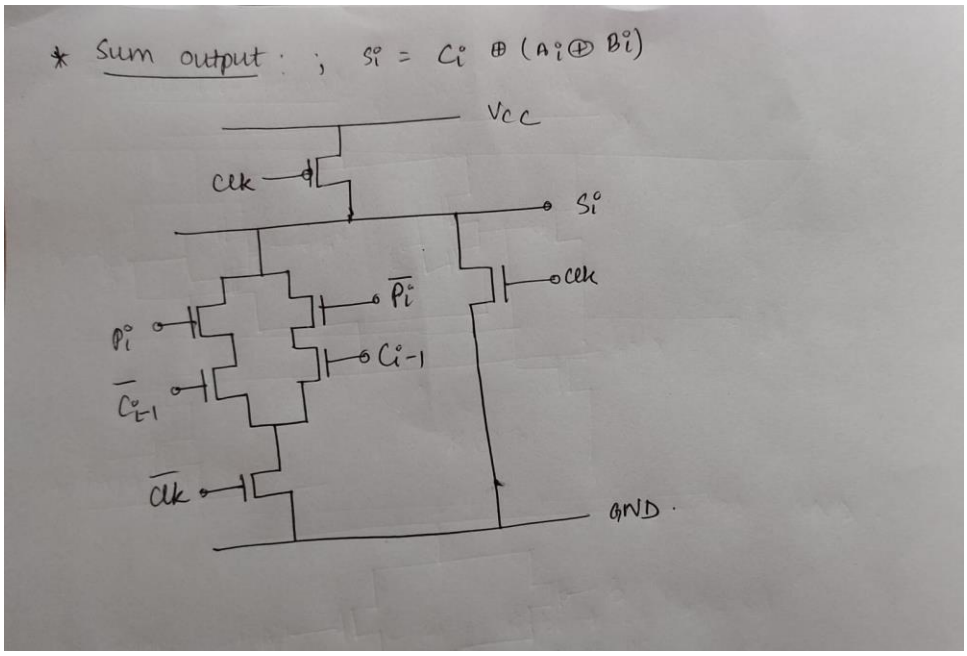
.endc

.end

- **Plot Waveform**
- .tran 0.1u 1u
- Output pins: C4 = 48, C1 = 50, C2 = 51, C3 = 52
-



- **Sum Output**
  - **Logic:** si = ci XOR pi
  - **Stick Diagram**

- 

- A total of 55 PMOS and 80 transistors have been used in total for the entire CLA adder.

- **NGspice Code**

  Sum Output

  .include carry.cir

  Input Initialization

  vc2 55 0 pulse (0 1.8 0 0.1n 0.1n 10n 20n)

  (Connections: drain gate source body CMOSN/CMOSP W L)

  **Inverter Gates For Pi*

  For P1

  Output: P1' = 54, P1 = 7

MP48 54 7 1 1 CMOSP W = 1.8u L = 0.18u

MN53 54 7 0 0 CMOSN W = 0.9u L = 0.18u

For P2

Output: P2' = 61, P2 = 15

MP49 61 15 1 1 CMOSP W = 1.8u L = 0.18u

MN54 61 15 0 0 CMOSN W = 0.9u L = 0.18u

For P3

Output: P3' = 65, P3 = 23

MP50 65 23 1 1 CMOSP W = 1.8u L = 0.18u

MN55 65 23 0 0 CMOSN W = 0.9u L = 0.18u

For P4

Output: P4' = 66, P4 = 31

MP51 66 31 1 1 CMOSP W = 1.8u L = 0.18u

MN56 66 31 0 0 CMOSN W = 0.9u L = 0.18u

Sum Output using carry output pins Ci-1 and propagate output pins Pi

For S1

Output: S1 = 56; Inputs: P1 = 7, P1' = 54, Cin = 49, Cin' = 42

MP52 56 52 1 1 CMOSP W = 1.8u L = 0.18u

MN57 56 7 57 0 CMOSN W = 0.9u L = 0.18u

MN58 56 54 58 0 CMOSN W = 0.9u L = 0.18u

MN59 57 42 59 0 CMOSN W = 0.9u L = 0.18u

MN60 58 49 59 0 CMOSN W = 0.9u L = 0.18u

MN61 59 53 0 0 CMOSN W = 0.9u L = 0.18u

For Second Clock, connecting the Sum Output line and GND

MN62 56 55 0 0 CMOSN W = 0.9u L = 0.18u

For S2

Output: S2 = 60; Inputs: P2 = 15, P2' = 61, C1 = 50, C1' = 43

MP53 60 52 1 1 CMOSP W = 1.8u L = 0.18u

MN63 60 15 62 0 CMOSN W = 0.9u L = 0.18u

MN64 60 61 63 0 CMOSN W = 0.9u L = 0.18u

MN65 62 43 64 0 CMOSN W = 0.9u L = 0.18u

MN66 63 50 64 0 CMOSN W = 0.9u L = 0.18u

MN67 64 53 0 0 CMOSN W = 0.9u L = 0.18u

For Second Clock, connecting the Sum Output line and GND

MN68 60 55 0 0 CMOSN W = 0.9u L = 0.18u

For S3

Output: S3 = 67; Inputs: P3 = 23, P3' = 65, C2 = 51, C2' = 44

MP54 67 52 1 1 CMOSP W = 1.8u L = 0.18u

MN69 67 65 69 0 CMOSN W = 0.9u L = 0.18u

MN70 67 23 68 0 CMOSN W = 0.9u L = 0.18u

MN71 68 44 70 0 CMOSN W = 0.9u L = 0.18u

MN72 69 51 70 0 CMOSN W = 0.9u L = 0.18u

MN73 70 53 0 0 CMOSN W = 0.9u L = 0.18u

For Second Clock, connecting the Sum Output line and GND

MN74 67 55 0 0 CMOSN W = 0.9u L = 0.18u

For S4

Output: S4 = 71; Inputs: P4 = 31, P4' = 66, C3 = 52, C3' = 45

MP55 71 52 1 1 CMOSP W = 1.8u L = 0.18u

MN75 71 31 72 0 CMOSN W = 0.9u L = 0.18u

MN76 71 66 73 0 CMOSN W = 0.9u L = 0.18u

MN77 72 45 74 0 CMOSN W = 0.9u L = 0.18u

MN78 73 52 74 0 CMOSN W = 0.9u L = 0.18u

MN79 74 53 0 0 CMOSN W = 0.9u L = 0.18u

For Second Clock, connecting the Sum Output line and GND

MN80 71 55 0 0 CMOSN W = 0.9u L = 0.18u

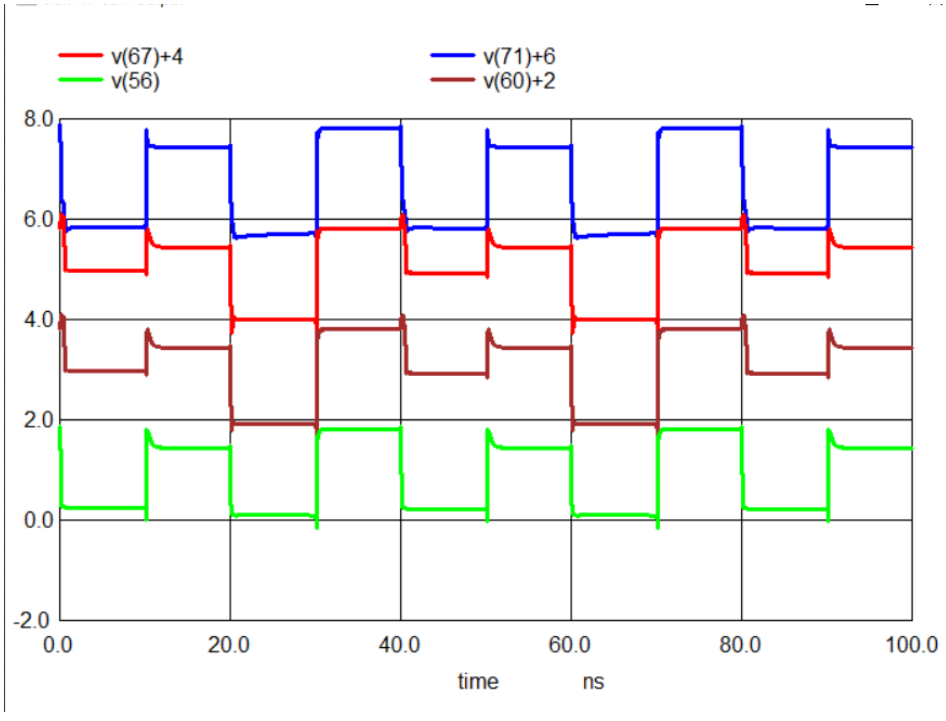
No. of PMOS = 55, NMOS = 80

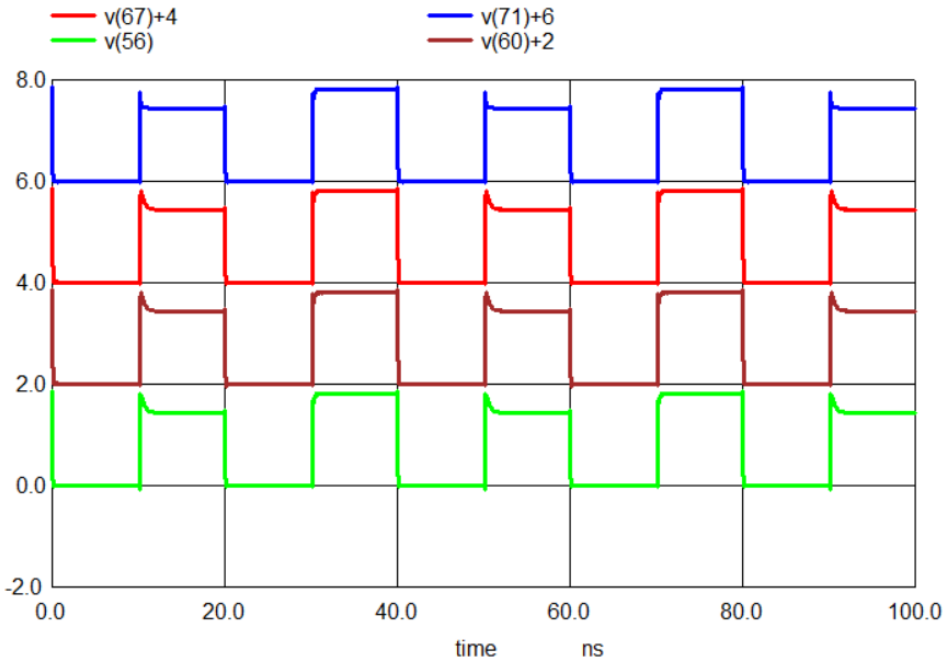
.tran 0.1u 1u

.control

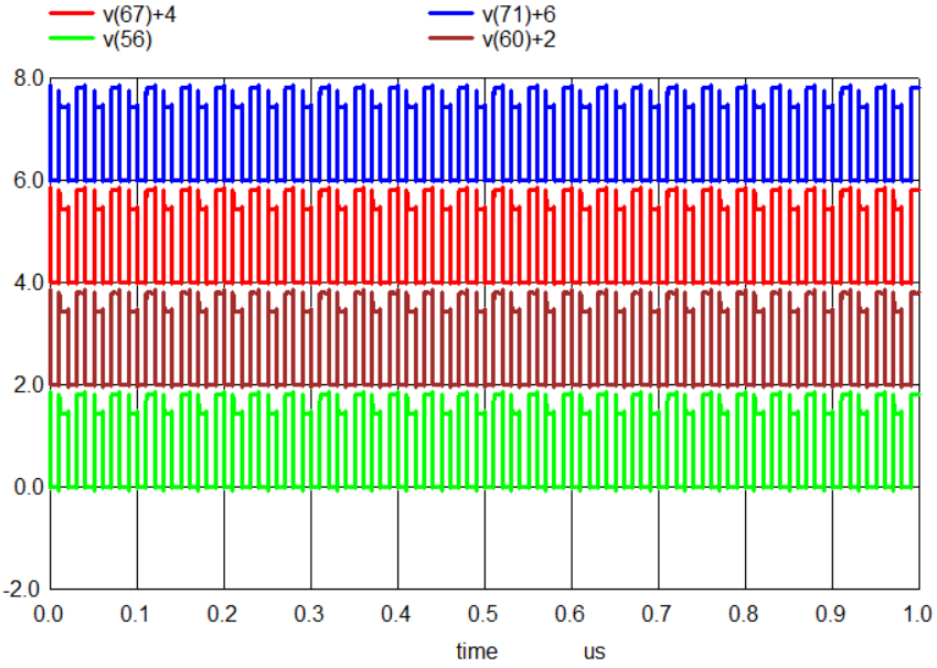
```
run
set color0=white
set color1=black
set color2=blue
set color3=red
set color4=brown
set xbrushwidth=3.5
set curplottitle= Anjali-Singh-2020102004-Sum-Output
plot v(71)+6 v(67)+4 v(60)+2 v(56)
.endc
.end
```

- **Plot Waveform**
- .tran 0.1n 100n
- Output pins: S1 = 56, S2 = 60, S3 = 67, S4 = 71;
-





- .tran 0.1u 1u



Verilog Codes

- CLA Adder Testbench

```
// Testbench for CLA adder code
  module CLAadder_tb;
  reg c_in;
  reg [3:0] a;
  reg [3:0] b;
  integer i, j;
  wire c_out;
  wire [3:0] sum;

  initial
    begin
      $dumpfile("dump.vcd");
      $dumpvars(0, CLAadder_tb);
      //Input initialization
      a = 0;
      b = 0;
      c_in = 0;
      for(i = 0; i < 16; i = i + 1)
        begin
            for (j = 0; j < 16; j = j + 1)
                begin
                    a = i;
                    b = j;
                    #10;
                end
        end
    end
endmodule
```

- CLA Adder Module

```
module cla(sum, c_out, a, b, c_in);
//Initializing Inputs and Outputs
input [3:0] a, b;
input c_in;
output [3:0] sum;
output c_out;


//Internal wires
wire o1, o2, o3, o4, o5, o6, o7, o8, o9, o10, o11, o12, o13, o14, o15;
wire p0, p1, p2, p3, g0, g1, g2, g3;
wire c1, c2, c3, c4;


// Propagate Signal
// p_i = a_i XOR b_i
xor XOR1(p0, a[0], b[0]); // p0 = a[0] XOR b[0]
xor XOR2(p1, a[1], b[1]);
xor XOR3(p2, a[2], b[2]);
xor XOR4(p3, a[3], b[3]);


// Generate Signal
// g_i = a_i AND b_i
and AND1(g0, a[0], b[0]); // g0 = a[0] AND b[0]
and AND2(g1, a[1], b[1]);
and AND3(g2, a[2], b[2]);
and AND4(g3, a[3], b[3]);
```

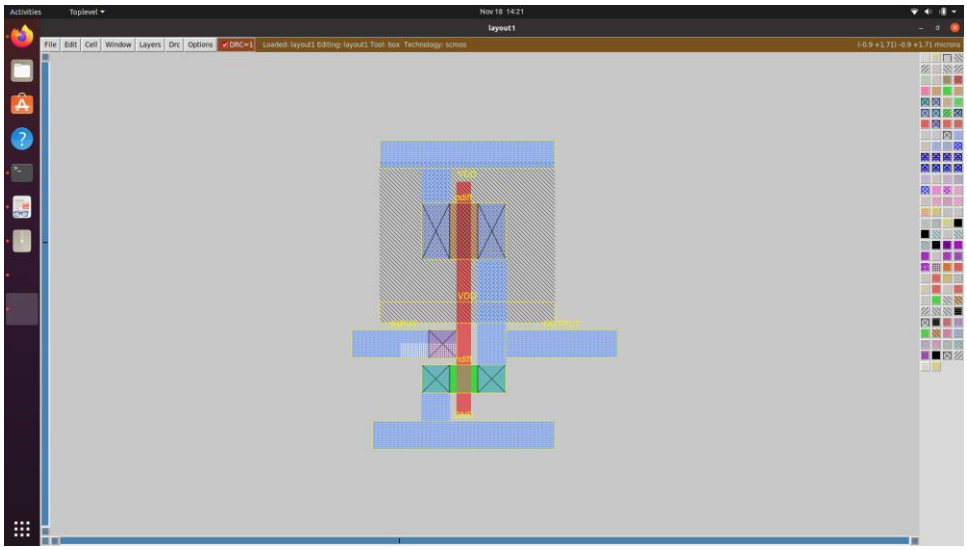
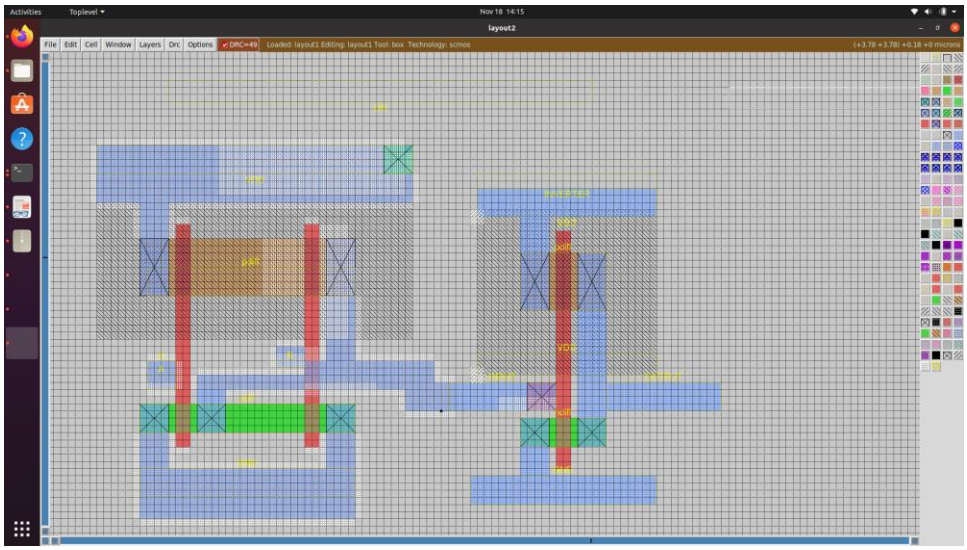
```
// Carry Output
// c_in = c0, c_out = c4
// for c1
and AND5(o1, p0, c_in); // o1 = p0 AND c_in
or OR1(c1, g0, o1); // c1 = g0 OR o1
// for c2
and AND6(o2, o1, p1); // o2 = p1 AND o1
and AND7(o3, g0, p1); // o3 = g0 AND p1
or OR2(o4, o2, o3); // o4 = o2 OR o3
or OR3(c2, g1, o4); // c2 = g1 OR o4
// for c3
and AND8(o5, o2, p2); // o5 = o2 AND p2, o2 = c_in & p0 & p1
and AND9(o6, o3, p2); // o6 = o3 AND p2, o3 = g0 AND p1
and AND10(o7, g1, p2); // o7 = g1 AND p2
or OR4(o8, o5, o6); // o8 = o5 OR o6
or OR5(o9, o7, o8);
or OR6(c3, g2, o9); // c3 = g2 OR o9
// for c4
and AND11(o10, o5, p3); // o10 = o5 AND p3, o5 = c_in & p0 & p1 & p2
and AND12(o11, o7, p3); // o11 = o7 AND p3, o7 = g1 & p2
and AND13(o12, g2, p3); // o12 = g2 AND p3
and AND14(o13, o6, p3); // o13 = g0 & p1 & p2 & p3
or OR7(o14, o10, o11); // o14 = (c_in & p0 & p1 & p2 & p3) | (g1 & p2 &
p3)
or OR8(o15, o14, o13); // o15 = (g1 & p2 & p3) | (g0 & p1 & p2 & p3) |
(c_in & p0 & p1 & p2 & p3)
or OR9(c4, g3, o15);
```

```
assign c_out = c4;


// Sum Output
xor XOR5(sum[0], c_in, p0);
xor XOR6(sum[1], c1, p1);
xor XOR7(sum[2], c2, p2);
xor XOR8(sum[3], c3, p3);


endmodule
```

Design Details:

- **AND Gate**
  - Area: width × length
  - Nwell: 11(lambda) × 40(lambda)
  - Pdiff: 7 (lambda) × 36 (lambda)
  - Ndiff: 7 (lambda) × 36 (lambda)
  - Pdiffusion Contact: 7 (lambda) × 6 (lambda) [metal 1]
  - Polysilicon Contact: 34 (lambda) × 4 (lambda) [poly silicon]
  - Contacts between Vdd and Pdiff: metal 1
  - Contacts between GND and Ndiff: metal 1
  - Common drain of PMOS to be shorted with drain of NMOS 2 [metal 1]
  - Ndiffusion Contact: 7 (lambda) × 6 (lambda)
  - Since we have nwell below Vdd, we will place nwell contact on Vdd
  - Nwell Contact: 6 (lambda) × 7 (lambda)
  - Then inverter gate is attached to get the AND output

MAGIC Layouts

- **Inverter Gate**
  - **Layout**
  - 
- **AND Gate**
  - **Layout**
  - 
  - **Spice Netlist**

    * SPICE3 file created from and.ext - technology: scmos

    .include TSMC_180nm.txt

    .option scale=0.09u

```
M1000 a_n18_15# a_n20_n6# VDD B pfet w=8 l=2
+  ad=80 pd=36 as=115 ps=76
M1001 VDD a_n8_n6# a_n18_15# B pfet w=8 l=2
+  ad=0 pd=0 as=0 ps=0
M1002 OUTPUT a_11_n6# VDD B pfet w=7 l=2
+  ad=35 pd=24 as=0 ps=0
M1003 a_n6_n1# a_n8_n6# a_n18_n1# Gnd nfet w=4 l=2
+  ad=20 pd=18 as=40 ps=28
M1004 a_n18_n1# a_n20_n6# GND Gnd nfet w=4 l=2
+  ad=0 pd=0 as=24 ps=28
M1005 OUTPUT a_11_n6# GND Gnd nfet w=4 l=2
+  ad=20 pd=18 as=0 ps=0
C0 B Gnd 1.33fF

.tran 0.1n 100n

.control
run
**plot background color**
set color0 = black
**grid lines color**
set color1 = white
**plot 1 color**
```

set color2 = blue

plot 2 color

set color3 = red

plot 3 color

set color4 = brown

set xbrushwidth = 1.5

plot v(6)+4 v(2)+2 v(4)

.endc

.end

- **OR Gate**
  - **Layout**
  -