INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY

HYDERABAD

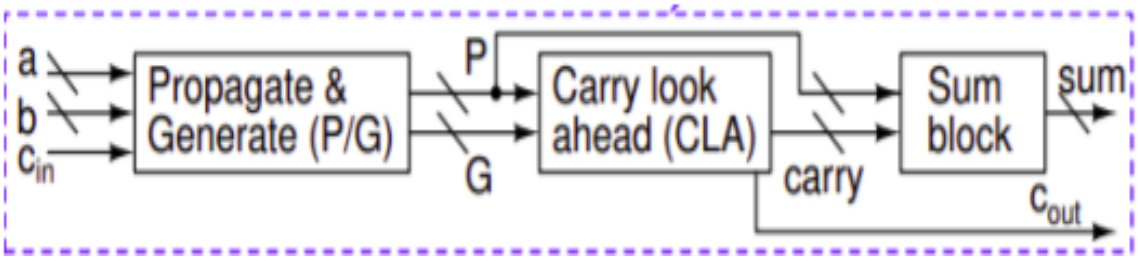
VLSI PROJECT

Carry look ahead adder

Srujana Vanka
2020102005

Question 1

To design a 4-bit carry look ahead (CLA) adder:



Proposed structure for the adder:

- To design a carry look ahead adder, I have used the static logic for all my blocks which have been implemented using basic gates.

<u>The circuit can be divided mainly into 3 blocks:</u>

1) **Propagate and Generate block**
   Propagate and generate for the ith bit is given by:

$$\text{Propagate} = p_i = a_i \oplus b_i$$

$$\text{Generate} = g_i = a_i \,.b_i$$

   Propagate can be designed using 2 input XOR gates, whereas for Generate, we require 2 input AND gates.

2) **Carry look ahead block (CLA)**
   Carry out (c_i+1) of the ith bit position can be written as (assuming $c_0 = 0$) follows:

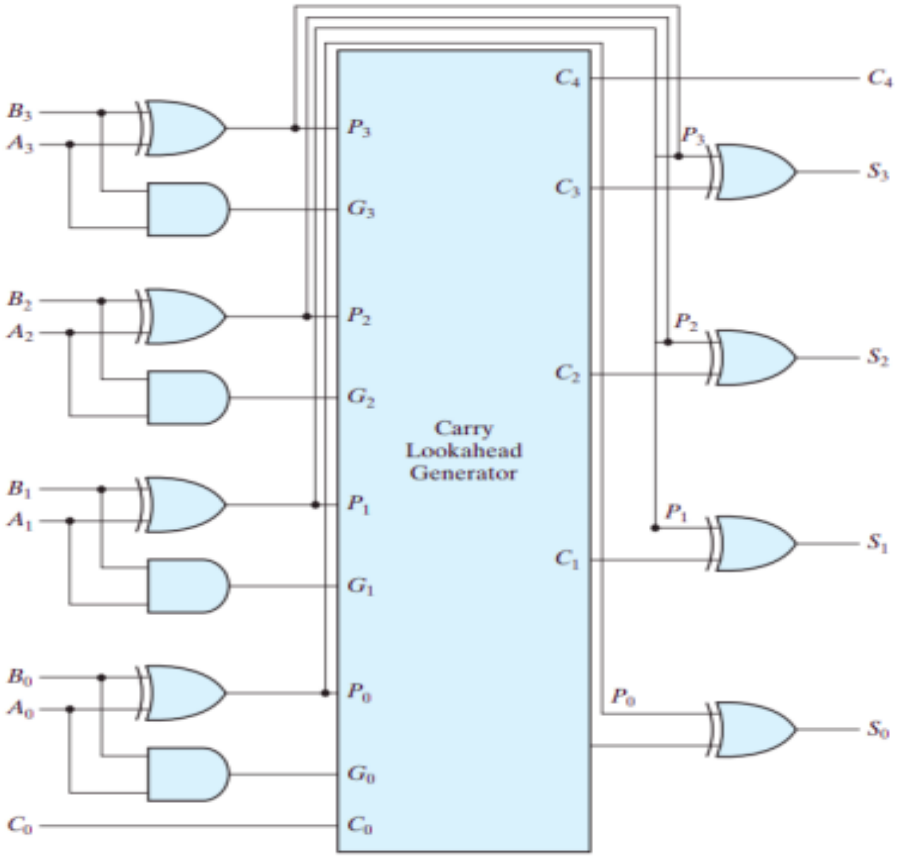
$$\text{Carry} = c_i{+}1 = (p_i \,.c_i) + g_i \text{ , for } i = 1, 2, 3, 4$$

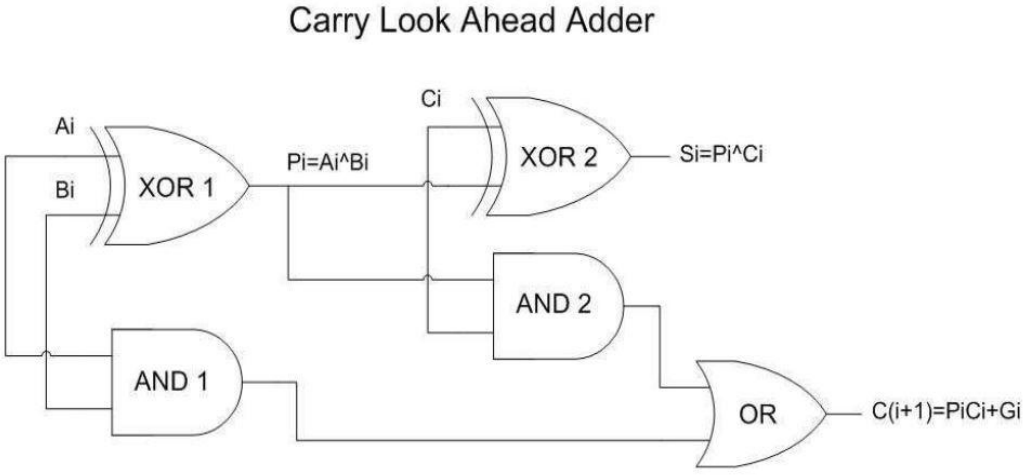
   Carry out can be designed using 2,3,4 input AND and OR gates.

3) **Sum block**
   Sum out (s_i) for the ith bit is given by:
   Sum can be designed using 2 input XOR gates

$$\text{Sum} = s_i = p_i \oplus c_i$$

Circuit diagram for proposed structure:





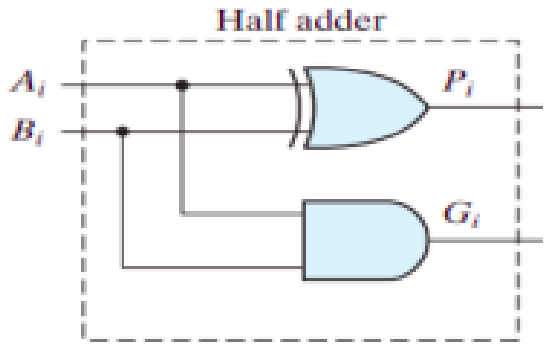
Question 2

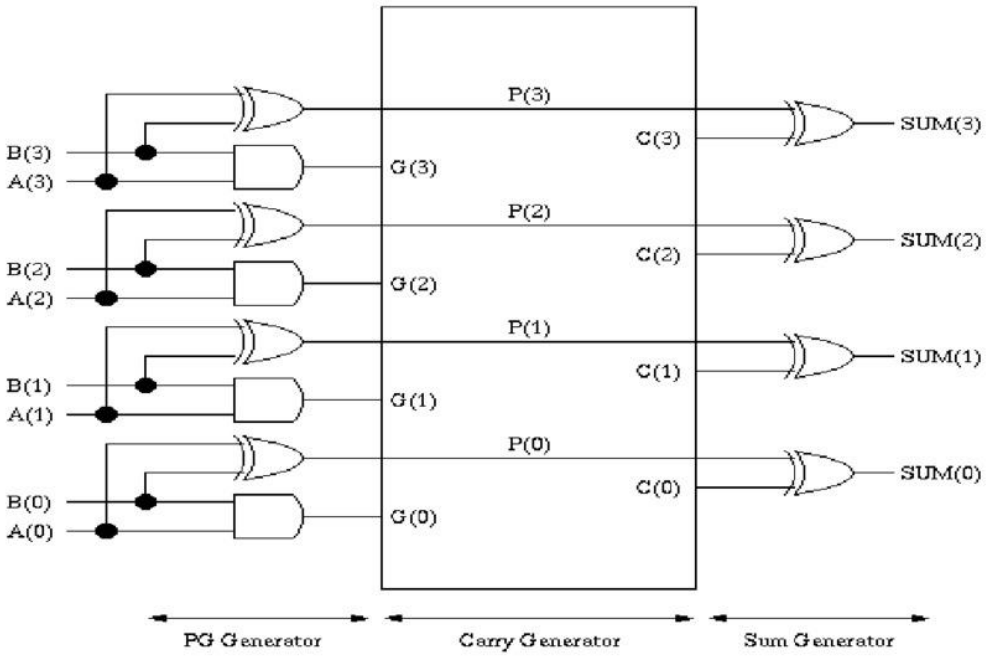
Design details of each block

1) PROPAGATE AND GENERATE BLOCK

The propagate and generate for the ith bits is given by:

Propagate = $p_i = a_i \oplus b_i$
Generate = $g_i = a_i . b_i$





\

2) CARRY LOOK AHEAD BLOCK

$$\text{Carry} = c_i{+}1 = (p_i \,.c_i) + g_i \text{ , for } i = 1, 2, 3, 4$$

Using the propagate and generate bits, we get the following:

For $c_{in} = 0$,

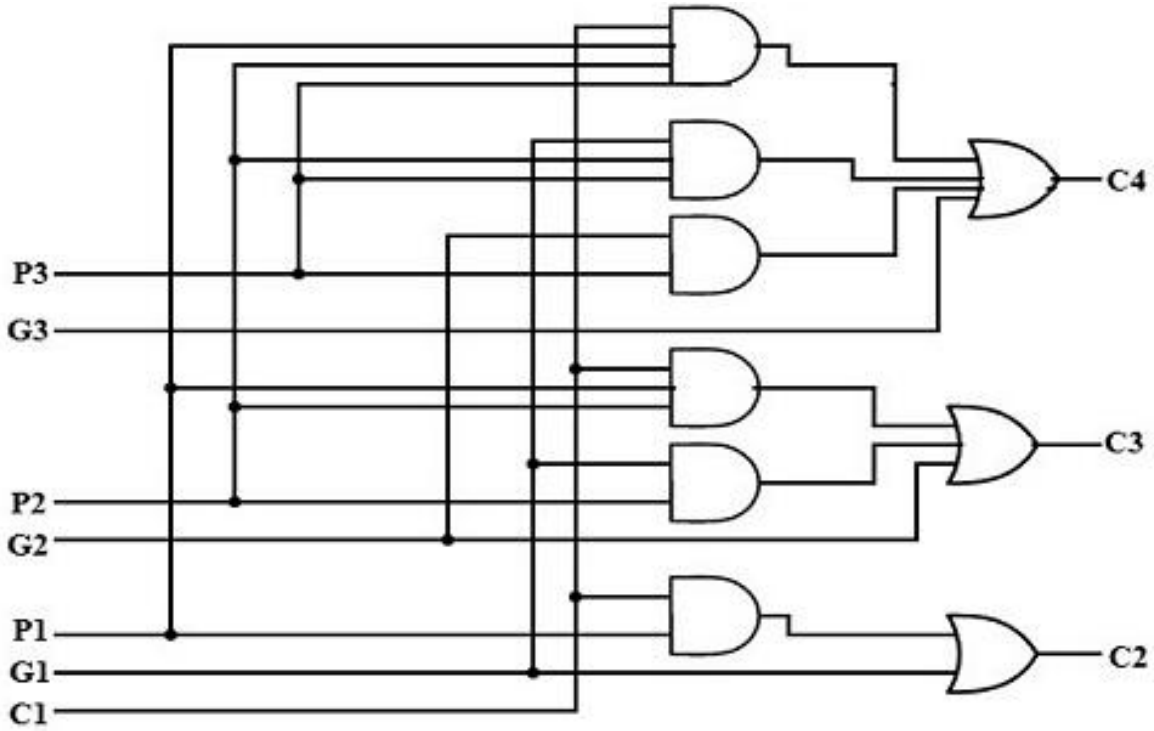
$c_0 = 0$ (given)

$c_1 = g_1$

$c_2 = p_2g_1 + g_2$

$c_3 = p_3p_2g_1 + p_3g_2 + g_3$

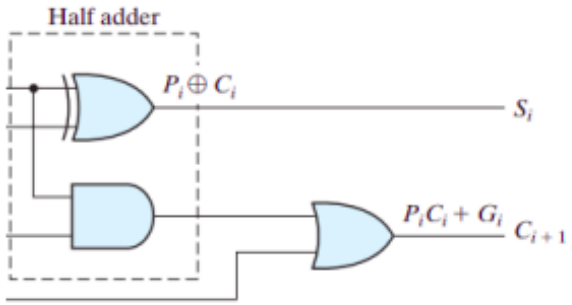
$c_{out} = p_4p_3p_2g_1 + p_4\, p_3g_2 + p_4g_3 + g_4$



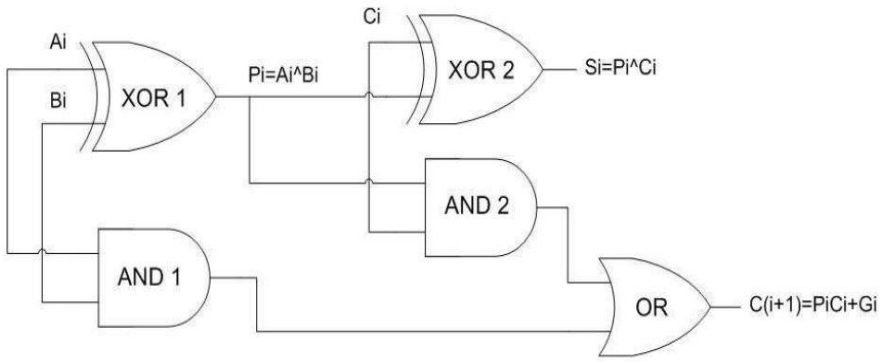
3) SUM BLOCK

The sum of the ith bit is given by:

$s_i = = p_i \oplus c_i$

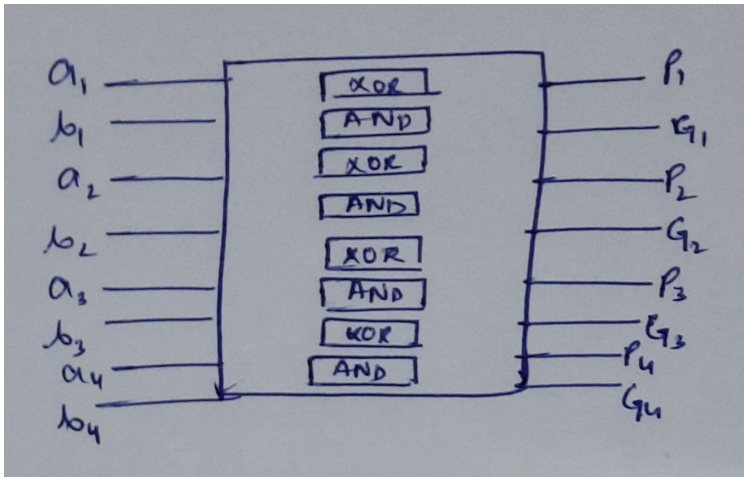


Carry Look Ahead Adder

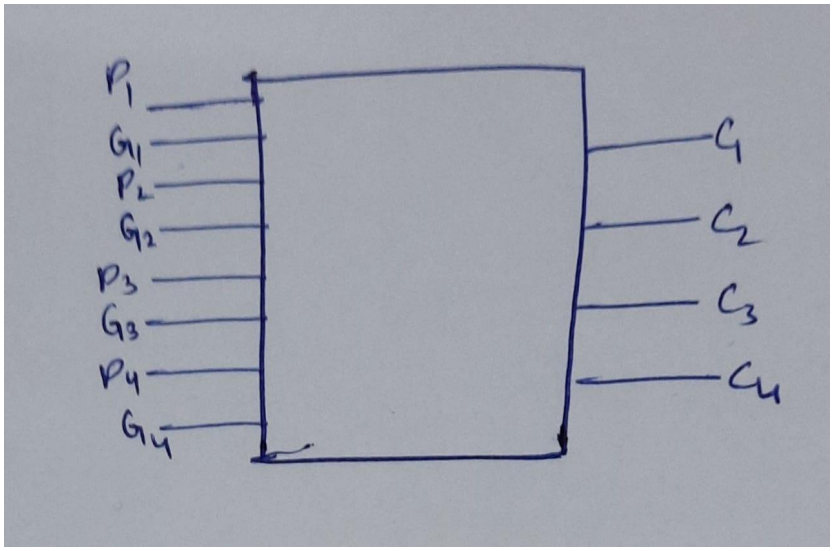


Adder modules

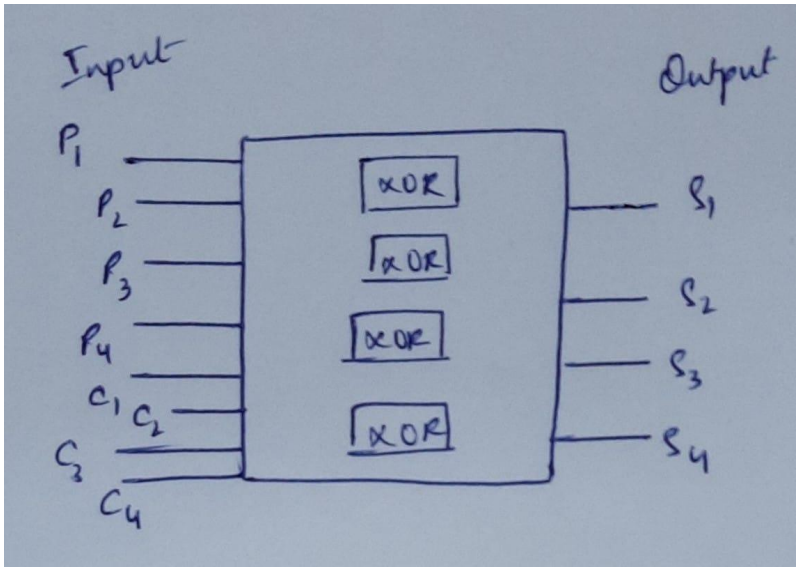
1) Propagate and generate block



2) Carry block



3) Sum block



- The lambda value is taken as 0.09u whereas W = 10*lambda

Question 3

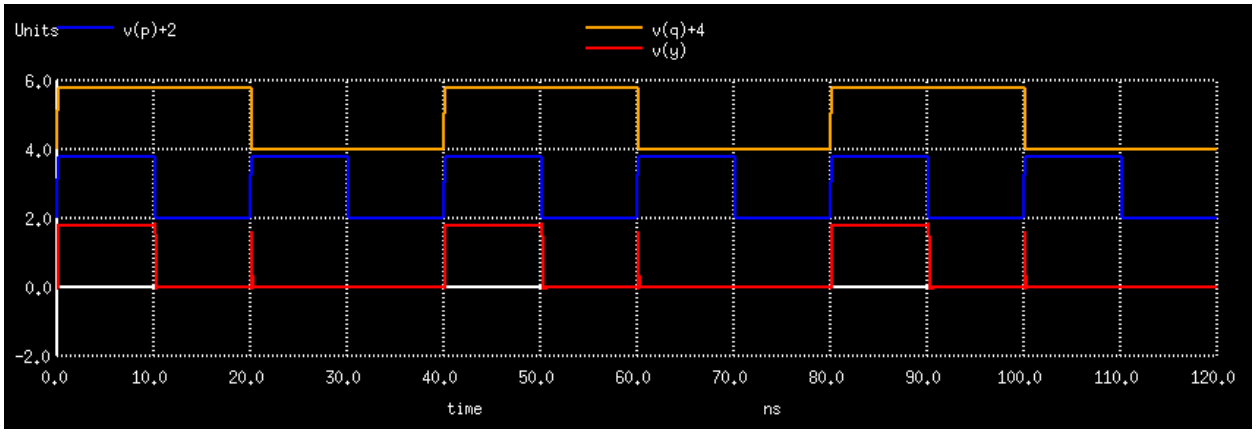
NGSPICE simulations

- **2 input AND gate**

```
*CMOS AND gate*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* inputs
vin1 p gnd pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin2 q gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 r q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 r q s s CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M4 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 y r vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M6 y r gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}
```

```
* delays
.tran 1n 120n

.measure tran trise
+ TRIG v(p) VAL = 0.9V RISE=1
+ TARG v(y) VAL = 0.9V RISE=1

.measure tran tfall
+ TRIG v(p) VAL = 0.9V FALL=1
+ TARG v(y) VAL = 0.9V FALL=1

.measure tran tpd param = '(trise + tfall)/2' goal=0

* outputs
.control
tran 1n 80n
run
plot v(y) v(p)+2 v(q)+4
.endc
.end
```

Output

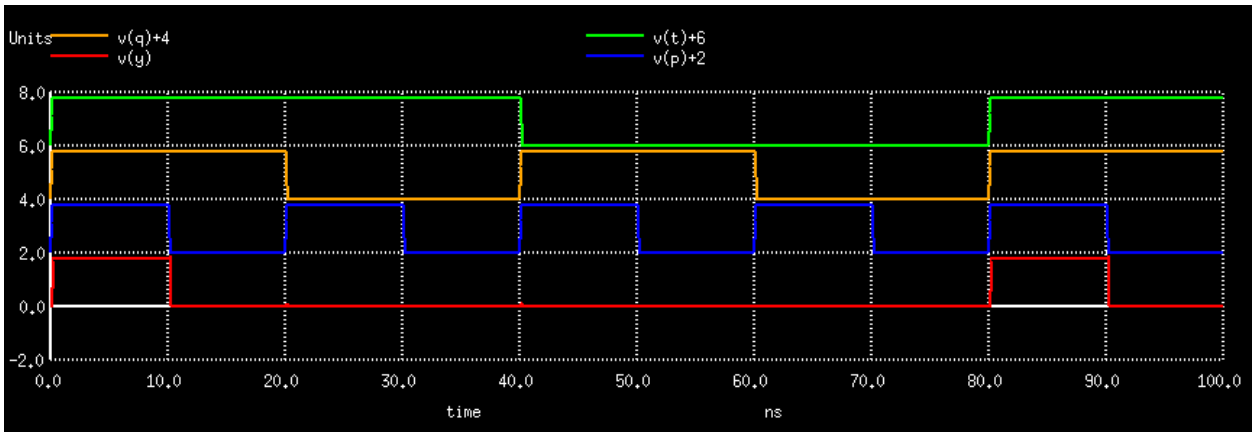


- **3 input AND gate**

```
*CMOS AND gate*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* inputs
vin1 p 0 pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin2 q 0 pulse (0 1.8 0 0.1n 0.1n 20n 40n)
vin3 t 0 pulse (0 1.8 0 0.1n 0.1n 40n 80n)

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 r q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 r t vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 r q s s CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 s t m m CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M6 m p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M7 y r vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}
```

```
M8 y r gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

* outputs
.control
tran 1n 100n
run
plot v(y) v(p)+2 v(q)+4 v(t)+6
.endc
.end
```

Output



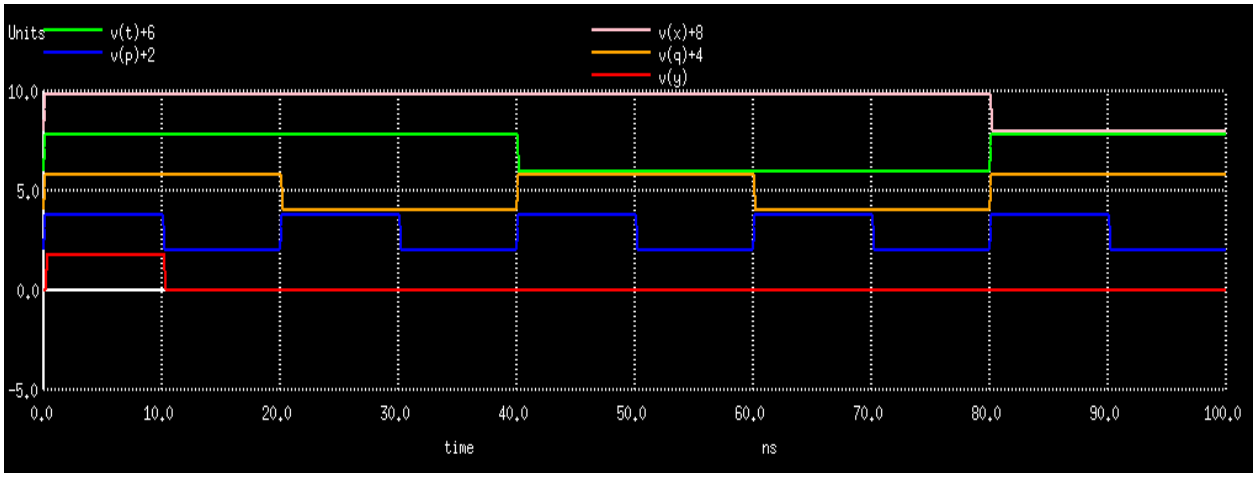
- **4 input AND gate**

```
*CMOS AND gate*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* inputs
```

```
vin1 p 0 pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin2 q 0 pulse (0 1.8 0 0.1n 0.1n 20n 40n)
vin3 t 0 pulse (0 1.8 0 0.1n 0.1n 40n 80n)
vin4 x 0 pulse (0 1.8 0 0.1n 0.1n 80n 160n)

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 r q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 r t vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 r x vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M5 r q s s CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M6 s t m m CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M7 m x n n CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M8 n p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M9 y r vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M10 y r gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

* outputs
.control
tran 1n 100n
run
plot v(y) v(p)+2 v(q)+4 v(t)+6 v(x)+8
```

```
.endc
.end
```

Output

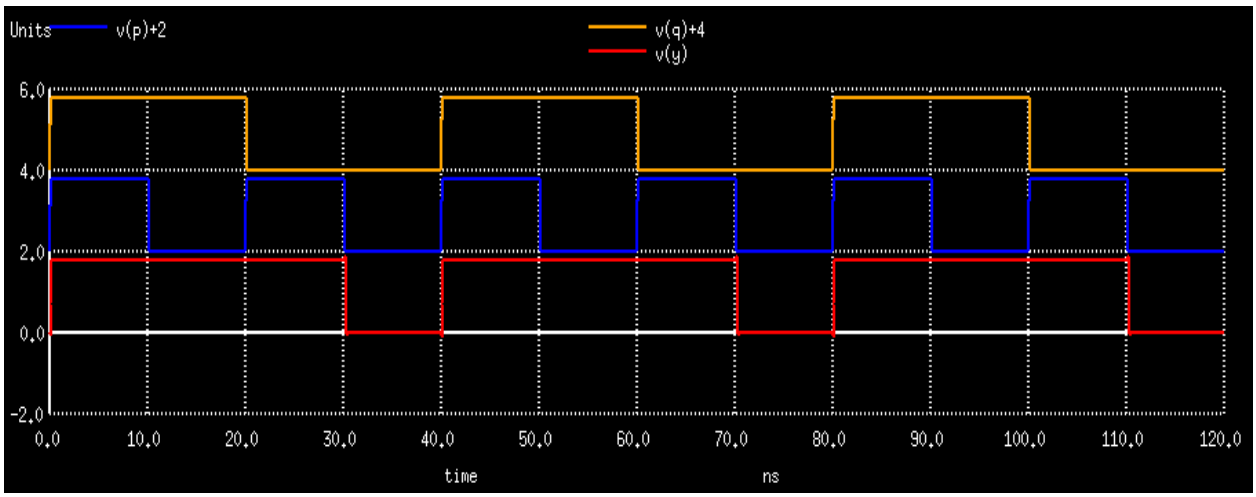


- **2 input OR gate**

```
*CMOS OR gate*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* inputs
vin1 p gnd pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin2 q gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 s q r r CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
```

```
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M4 s q gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 y s vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M6 y s gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

* outputs
.control
tran 1n 80n
run
plot v(y) v(p)+2 v(q)+4
.endc
.end
```

Output

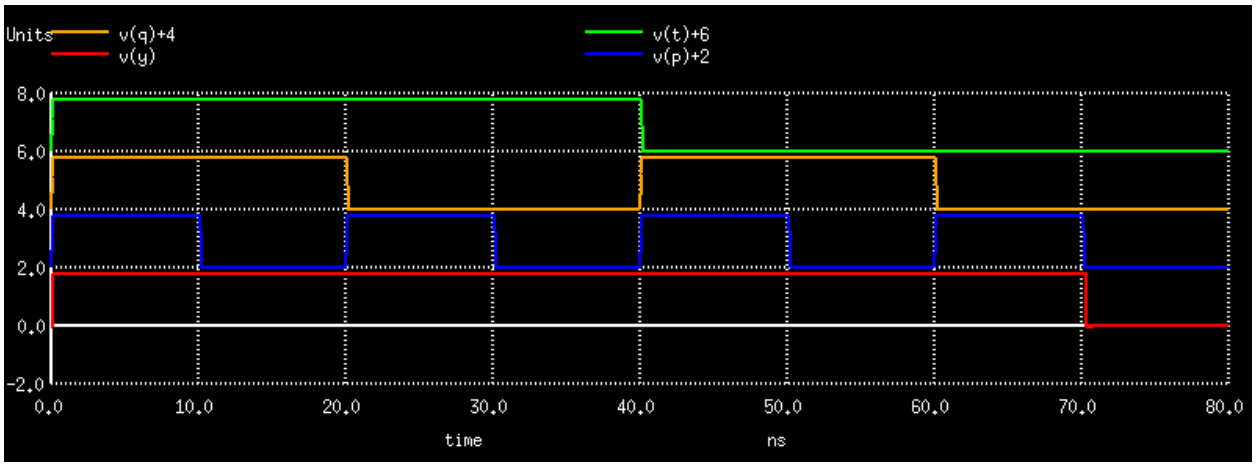


- **3 input OR gate**

```
*CMOS OR gate*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* inputs
vin1 p gnd pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin2 q gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)
vin3 t 0 pulse (0 1.8 0 0.1n 0.1n 40n 80n)

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 m q r r CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 s t m m CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 s q gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 s t gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M7 y s vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M8 y s gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
```

```
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

* outputs
.control
tran 1n 80n
run
plot v(y) v(p)+2 v(q)+4 v(t)+6
.endc
.end
```

Output



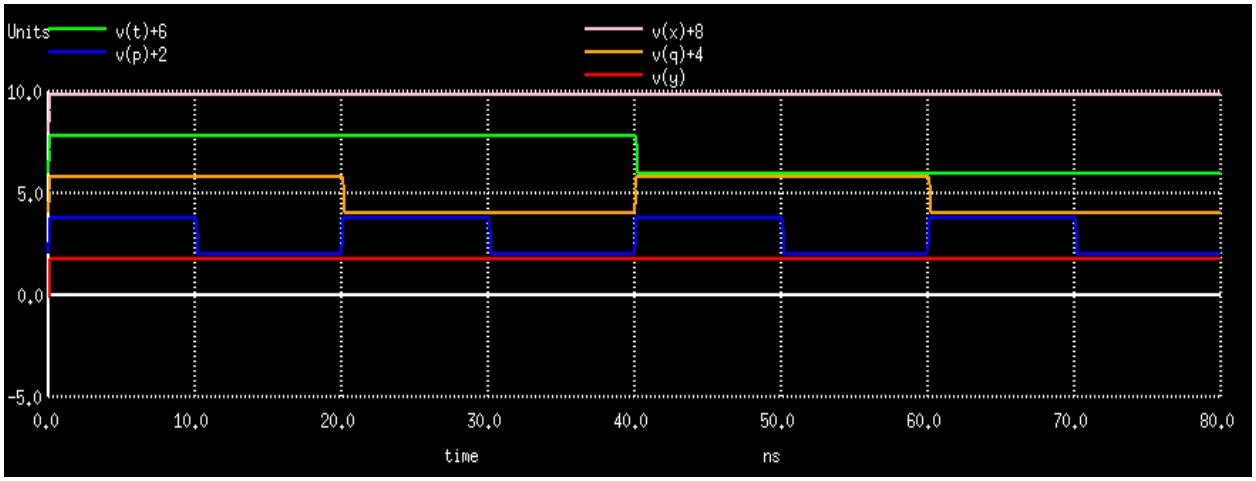
- **4 input OR gate**

```
*CMOS OR gate*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* inputs
```

```
vin1 p gnd pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin2 q gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)
vin3 t 0 pulse (0 1.8 0 0.1n 0.1n 40n 80n)
vin4 x 0 pulse (0 1.8 0 0.1n 0.1n 80n 160n)

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 m q r r CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 n t m m CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 s x n n CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M5 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M6 s q gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M7 s t gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M8 s x gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M9 y s vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M10 y s gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

* outputs
.control
tran 1n 80n
run
plot v(y) v(p)+2 v(q)+4 v(t)+6 v(x)+8
```

```
.endc
.end
```

Output



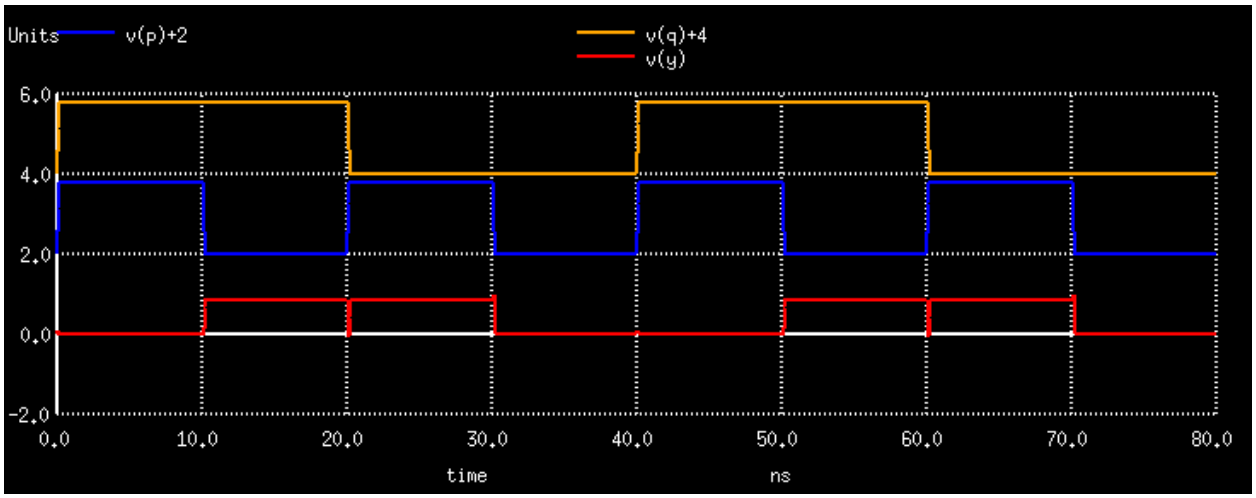
- **XOR gate**

```
*CMOS EXOR gate*

.include TSMC_180nm.txt
.include inverter.sub

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* inputs
vin1 p gnd pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin2 q gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)

.subckt xor_subckt p q y vdd gnd

* inverter
```

```
M1 a p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 a p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M3 b q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 b q gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

* xor gate
M5 r b vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M6 y p r r CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M7 s q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M8 y a s s CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M9 m q y y CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M10 gnd p m m CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M11 n b y y CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M12 gnd a n n CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

.ends xor_subckt

xx1 p q y vdd gnd xor_subckt
```

```
* output
.control
tran 1n 80n
run
plot v(y) v(p)+2 v(q)+4

.endc
.end
```

Output



CLA BLOCK

- **Propagate and generate block**

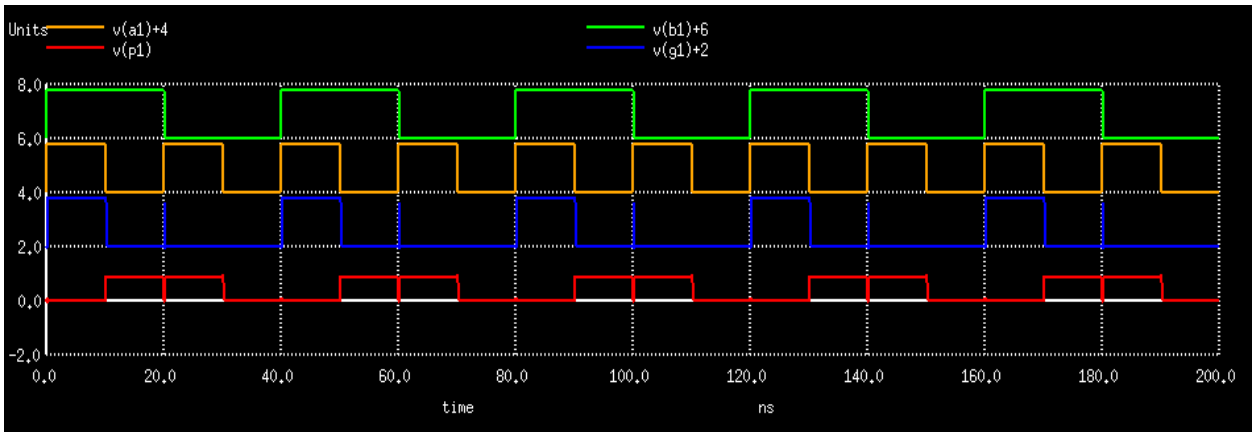
```
*Propagate and generate block*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd
```

```
vdd vdd gnd 'SUPPLY'

* Inputs
vin cin gnd pulse (0 1.8 0 0.1n 0.1n 160n 320n)
* Input A
vin_a1 a1 gnd pulse (0 1.8 0 0.1n 0.1n 10n 20n)
* vin_a2 a2 gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)
* vin_a3 a3 gnd pulse (0 1.8 0 0.1n 0.1n 40n 80n)
* vin_a4 a4 gnd pulse (0 1.8 0 0.1n 0.1n 80n 160n)

* Input B
vin_b1 b1 gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)
* vin_b2 b2 gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)
* vin_b3 b3 gnd pulse (0 1.8 0 0.1n 0.1n 40n 80n)
* vin_b4 b4 gnd pulse (0 1.8 0 0.1n 0.1n 80n 160n)


* XOR gate
.subckt xor_subckt p q y vdd gnd

* inverter
M1 a p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 a p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M3 b q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 b q gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

* xor gate
M5 r b vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M6 y p r r CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}
```

```
M7 s q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M8 y a s s CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M9 m q y y CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M10 gnd p m m CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M11 n b y y CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M12 gnd a n n CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

.ends xor_subckt

* AND gate
.subckt and_subckt p q y vdd gnd

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 r q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 r q s s CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M4 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 y r vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M6 y r gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}
```

```
.ends

* Propagate and generate
xx1 a1 b1 p1 vdd gnd xor_subckt
xa1 a1 b1 g1 vdd gnd and_subckt

* xx2 a2 b2 p2 vdd gnd xor_subckt
* xa2 a2 b2 g2 vdd gnd and_subckt

* xx3 a3 b3 p3 vdd gnd xor_subckt
* xa3 a3 b3 g3 vdd gnd and_subckt

* xx4 a4 b4 p4 vdd gnd xor_subckt
* xa4 a4 b4 g4 vdd gnd and_subckt

* Output
.control
tran 1n 200n
run
plot v(p1) v(g1)+2 v(a1)+4 v(b1)+6
* plot v(p2) v(g2)
* plot v(p3) v(g3)
* plot v(p4) v(g4)
set curplottitle= "Srujana Vanka - 2020102005 - PGblock"
.endc
.end
```

Output



Delay:
Propagate ≈ 0.026ns
Generate ≈ 0.041ns

- **Carry look ahead block**

```
*Carry look ahead block*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param w = {10*LAMBDA}
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* 2 input OR gate
.subckt or2_subckt p q y vdd gnd

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 r q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 r q s s CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M4 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 y r vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M6 y r gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}
```

```
.ends or2_subckt

* 3 input OR gate
.subckt or3_subckt p q t y vdd gnd
M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 m q r r CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 s t m m CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 s q gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 s t gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M7 y s vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M8 y s gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}
.ends or3_subckt

* 4 input OR gate
.subckt or4_subckt p q t x y vdd gnd
M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 m q r r CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 n t m m CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}
```

```
M4 s x n n CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M5 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M6 s q gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M7 s t gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M8 s x gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M9 y s vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M10 y s gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}
.ends or4_subckt

* 2 input AND gate
.subckt and2_subckt p q y vdd gnd

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 r q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 r q s s CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M4 s p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 y r vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}
```

```
M6 y r gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}
.ends and2_subckt

* 3 input AND gate
.subckt and3_subckt p q t y vdd gnd

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 r q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M3 r t vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 r q s s CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M5 s t m m CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M6 m p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M7 y r vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M8 y r gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}
.ends and3_subckt

* 4 input AND gate
.subckt and4_subckt p q t x y vdd gnd

M1 r p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M2 r q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}
```

```
M3 r t vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 r x vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M5 r q s s CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M6 s t m m CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M7 m x n n CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M8 n p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M9 y r vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M10 y r gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}
.ends and4_subckt

*Carry look ahead block*
* C0 = 0

* c1 == g1
xxx p1 cin z__out vdd gnd and_subckt
xxx2 g1 z__out c1 vdd gnd or2_subckt

* C2 = p2g1 + g2
xa5 p2 g1 z2_out1 vdd gnd and_subckt
x01 g2 z2_out1 c2 vdd gnd or2_subckt

* C3 = p3p2g1 + p3g2 + g3
xa6 p3 p2 g1 z3_out2 vdd gnd and3_subckt
xa7 p3 p2 z2_out2 vdd gnd and_subckt
```

```
x02 g3 z3_out2 z2_out2 c3 vdd gnd or3_subckt

* c4 = p4p3p2g1 + p4p3g2 + p4g3 + g4
xa8 p4 p3 p2 g1 z4_out3 vdd gnd and4_subckt
xa9 p4 p3 g2 z3_out3 vdd gnd and3_subckt
xa10 p4 g3 z2_out3 vdd gnd and_subckt
x03 g4 z2_out3 z3_out3 z4_out3 c4 vdd gnd or4_subckt

* Output
.control
tran 1n 200n
run
set curplottitle= "Srujana Vanka - 2020102005 - carry"
.endc
.end
```

The delay of the carry look ahead block is around 0.24ns.

- **Sum block**

```
*Sum block*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param w = {10*LAMBDA}
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* XOR gate
.subckt xor_subckt p q y vdd gnd

* inverter
M1 a p vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}
```

```
M2 a p gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M3 b q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M4 b q gnd gnd CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

* xor gate
M5 r b vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M6 y p r r CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M7 s q vdd vdd CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M8 y a s s CMOSP W={width_P} L={2*LAMBDA} AS={5*width_P*LAMBDA}
PS={10*LAMBDA+2*width_P} AD={5*width_P*LAMBDA} PD={10*LAMBDA+2*width_P}

M9 m q y y CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M10 gnd p m m CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M11 n b y y CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

M12 gnd a n n CMOSN W={width_N} L={2*LAMBDA} AS={5*width_N*LAMBDA}
PS={10*LAMBDA+2*width_N} AD={5*width_N*LAMBDA} PD={10*LAMBDA+2*width_N}

.ends xor_subckt

* Sum
* s1
xx5 p1 c1 s1 vdd gnd xor_subckt
```

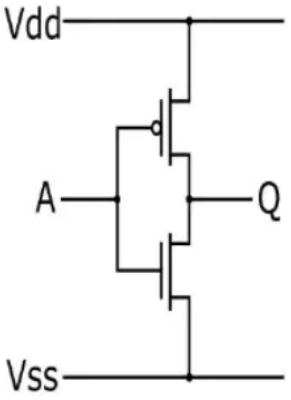
```
* s2
xx6 p2 c2 s2 vdd gnd xor_subckt

* s3
xx7 p3 c3 s3 vdd gnd xor_subckt

* s4
xx8 p4 c4 s4 vdd gnd xor_subckt

* Output
.control
tran 1n 200n
run
plot v(s1) v(s2) v(s3) v(s4)
set curplottitle= "Srujana Vanka - 2020102005 - sum"
.endc
.end
```

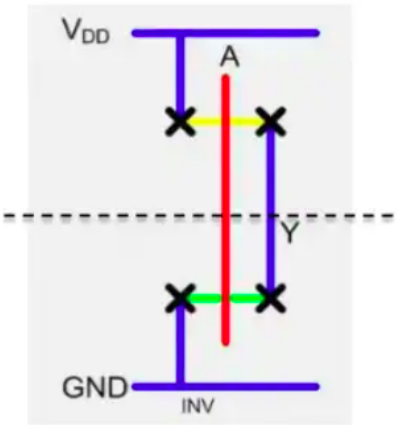
The delay of the sum block is the same as the propagate (since both are XOR gates).

Question 4

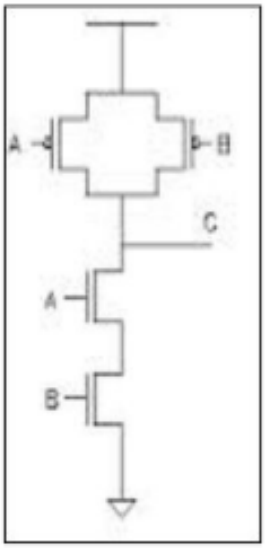
Stick diagrams

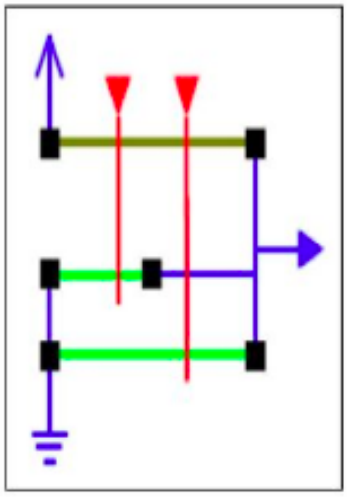
- **NOT gate**



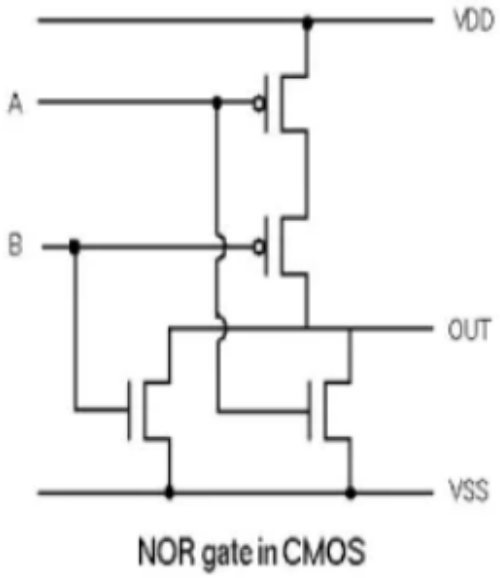


- **NAND gate**

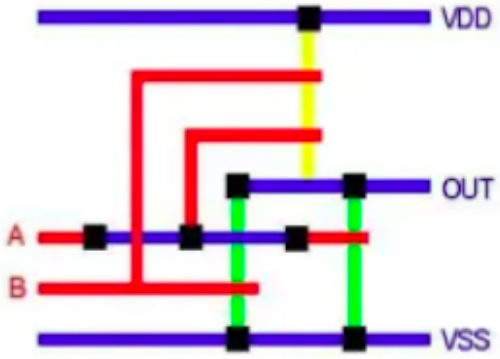




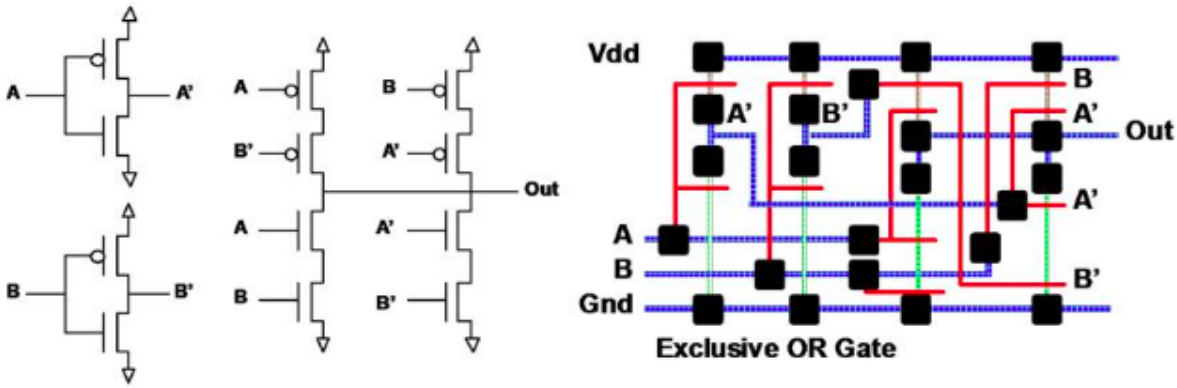
- **NOR gate**



NOR gate in CMOS

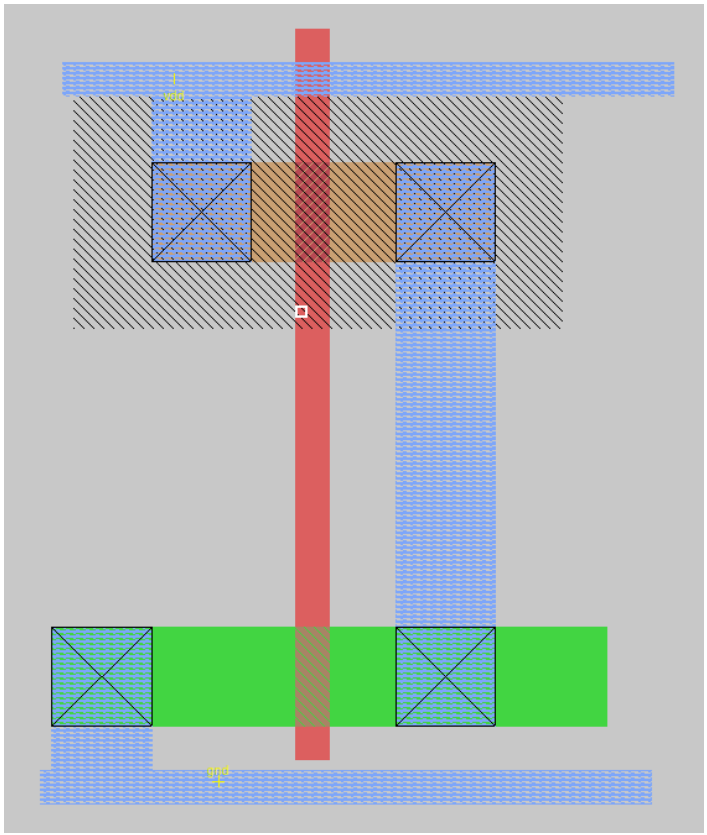


- **XOR gate**

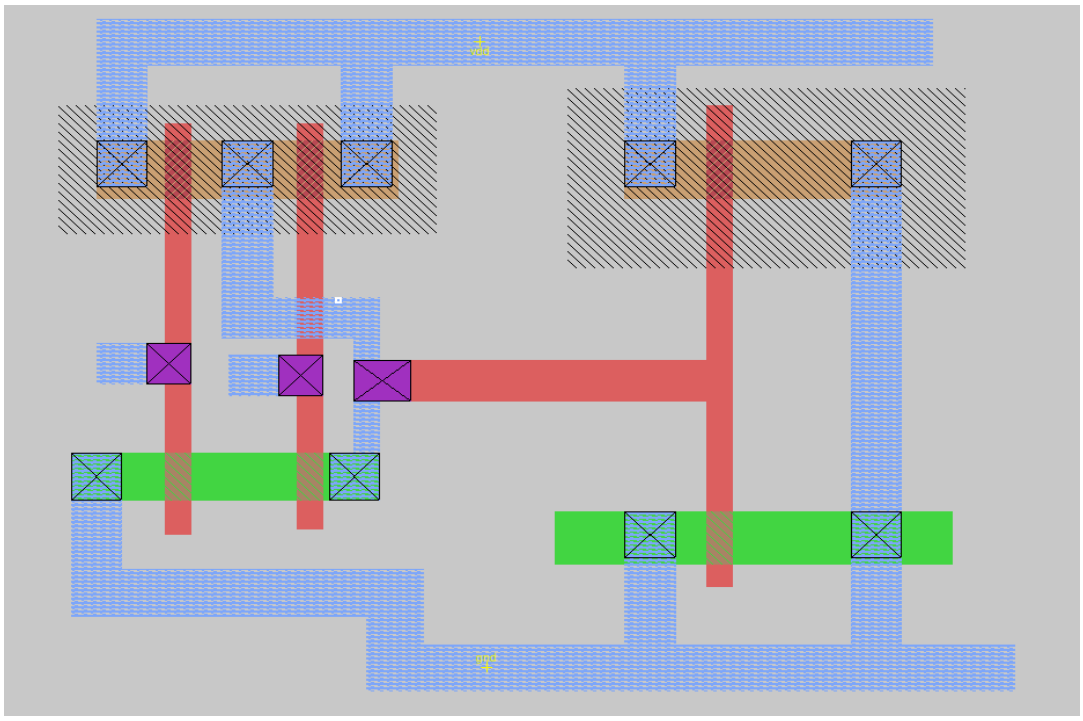


Question 5

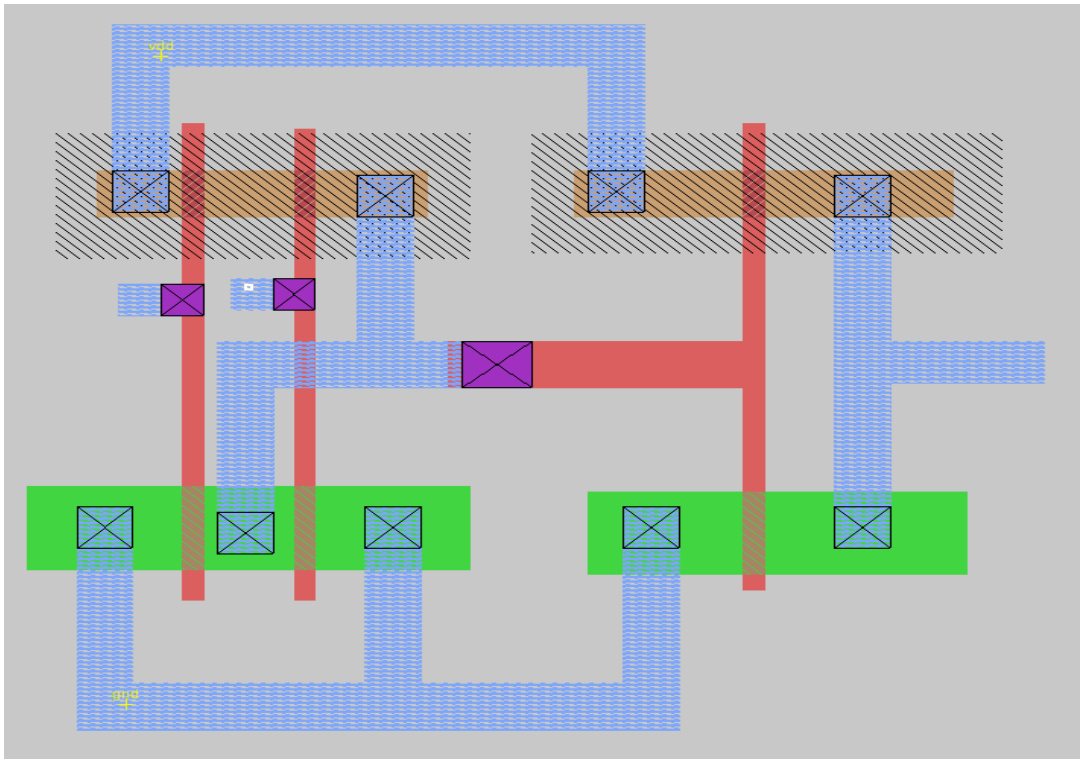
NOT



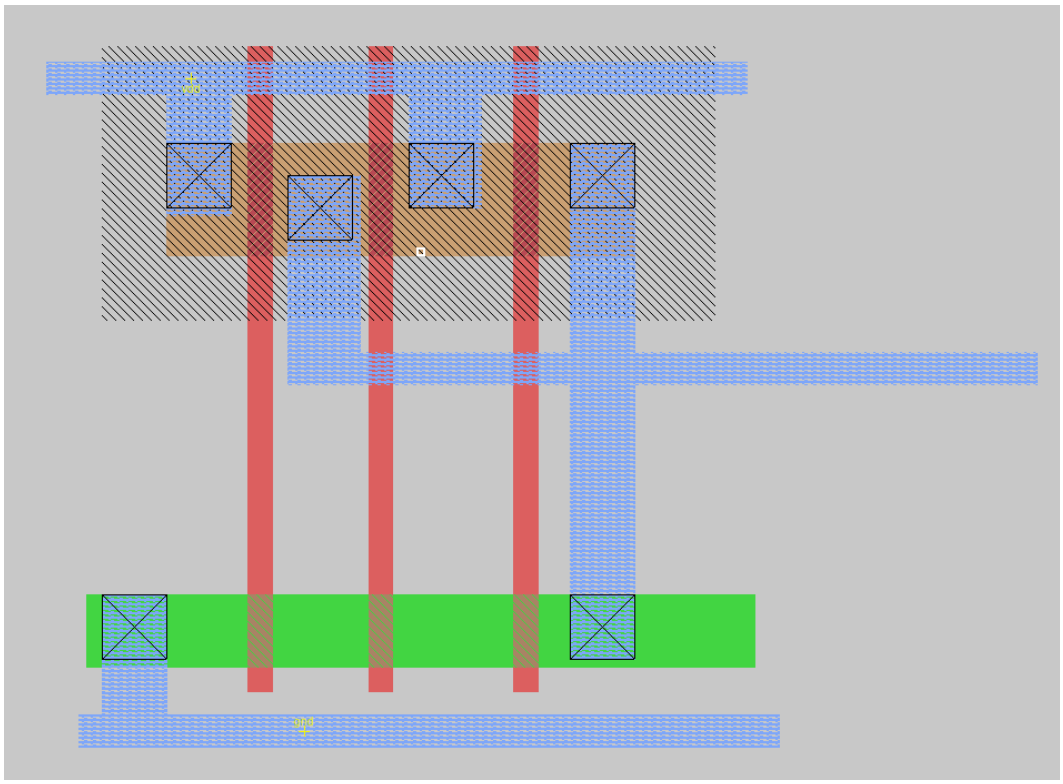
2 INPUT AND



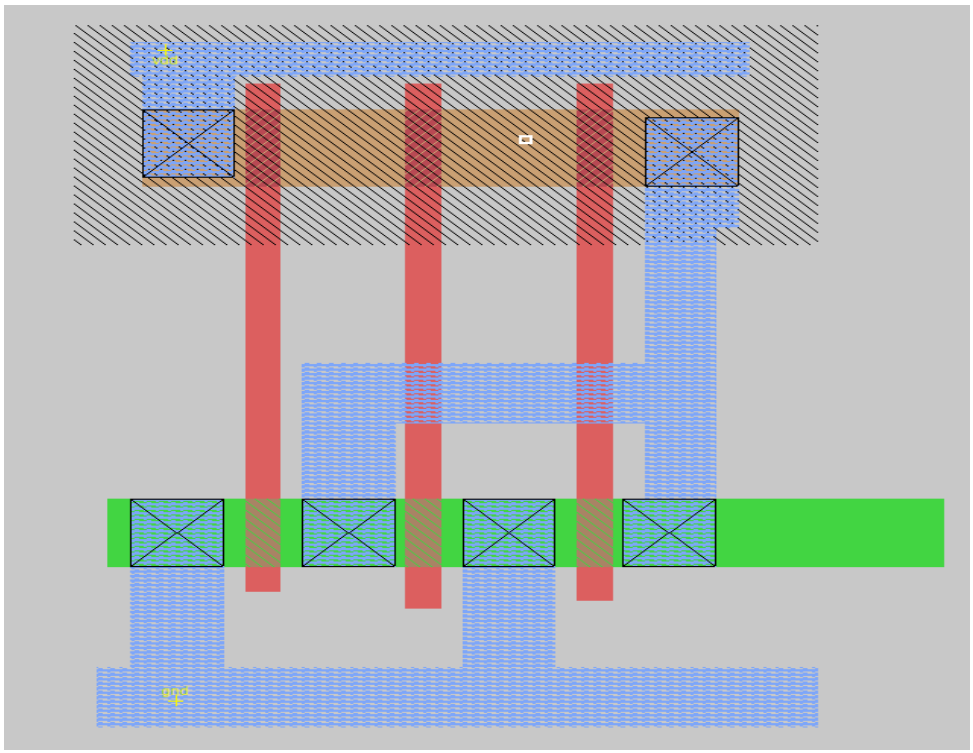
2 INPUT OR



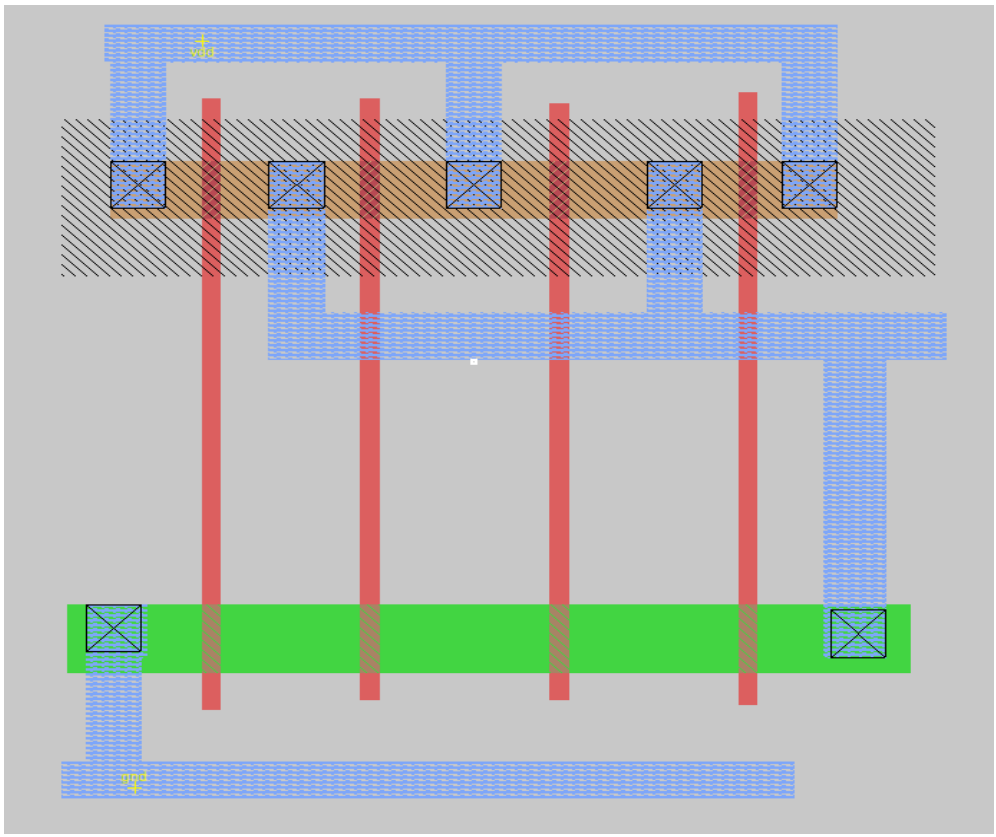
3 INPUT NAND



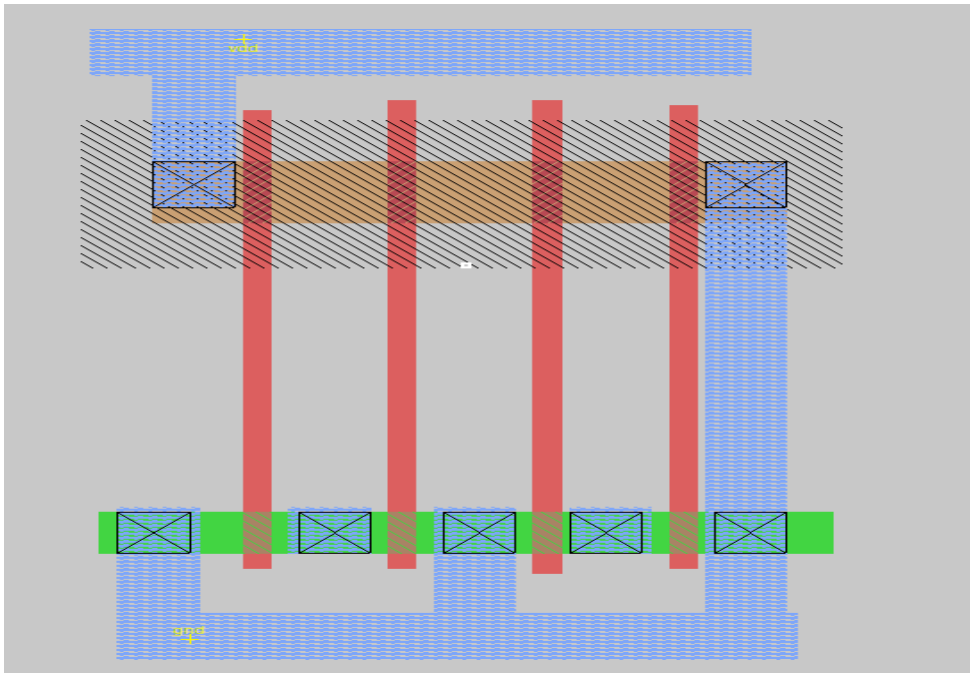
3 INPUT NOR



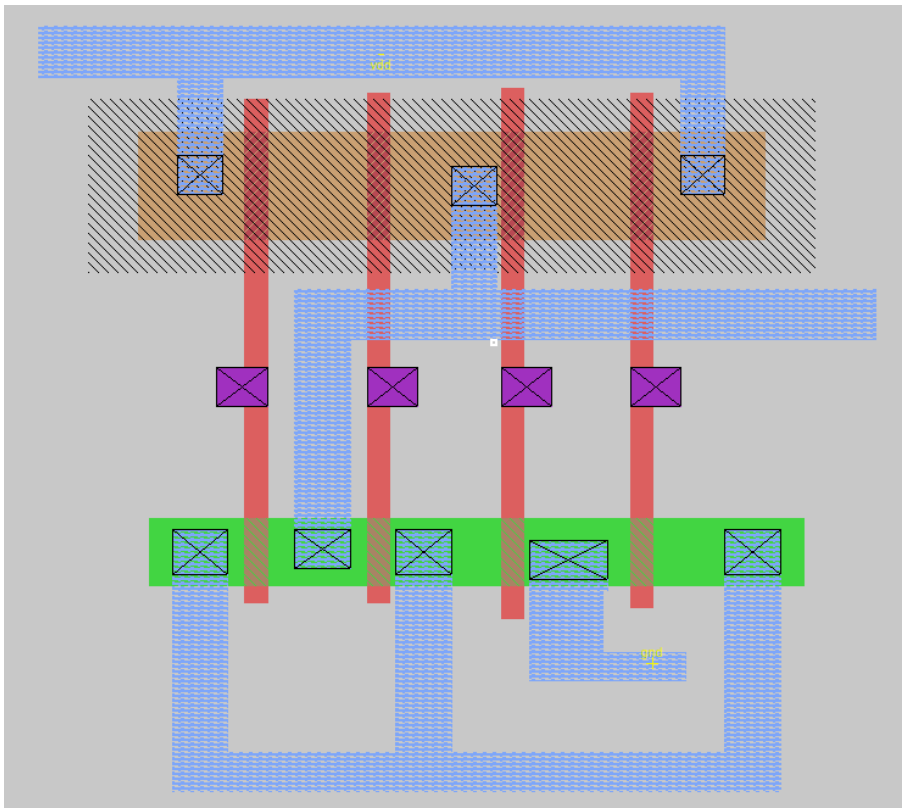
4 INPUT NAND



4 INPUT NOR



XOR



Propagate and generate block

```
* SPICE3 file created from pgblock.ext - technology: scmos
*Propagate and generate block*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

* Inputs
vin cin gnd pulse (0 1.8 0 0.1n 0.1n 160n 320n)
* Input A
vin_a1 a1 gnd pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin_a11 a11 gnd pulse (1.8 0 0 0.1n 0.1n 10n 20n)
```

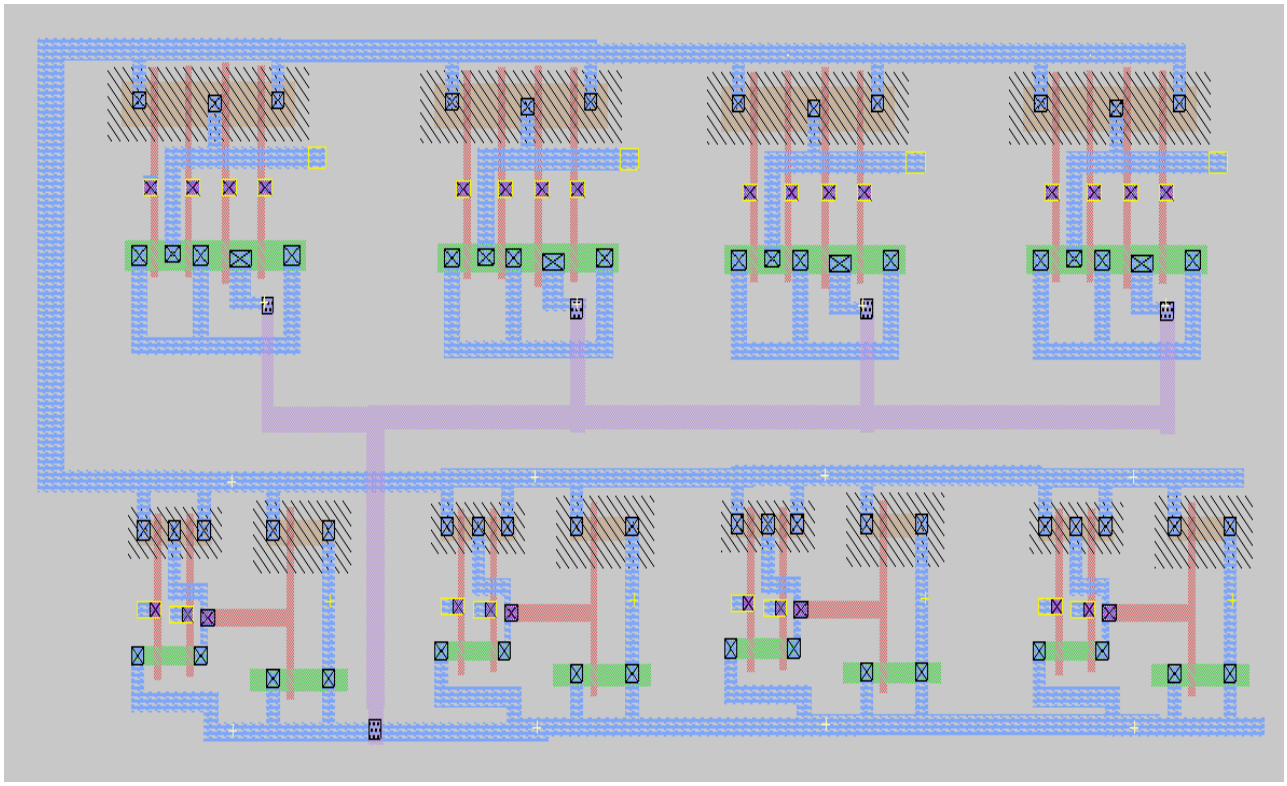
```
vin_a2 a2 gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)
vin_a21 a21 gnd pulse (1.8 0 0 0.1n 0.1n 20n 40n)
vin_a3 a3 gnd pulse (0 1.8 0 0.1n 0.1n 40n 80n)
vin_a31 a31 gnd pulse (1.8 0 0 0.1n 0.1n 40n 80n)
vin_a4 a4 gnd pulse (0 1.8 0 0.1n 0.1n 80n 160n)
vin_a41 a41 gnd pulse (1.8 0 0 0.1n 0.1n 80n 160n)

* Input B
vin_b1 b1 gnd pulse (0 1.8 0 0.1n 0.1n 10n 20n)
vin_b11 b11 gnd pulse (1.8 0 0 0.1n 0.1n 10n 20n)
vin_b2 b2 gnd pulse (0 1.8 0 0.1n 0.1n 20n 40n)
vin_b21 b21 gnd pulse (1.8 0 0 0.1n 0.1n 20n 40n)
vin_b3 b3 gnd pulse (0 1.8 0 0.1n 0.1n 40n 80n)
vin_b31 b31 gnd pulse (1.8 0 0 0.1n 0.1n 40n 80n)
vin_b4 b4 gnd pulse (0 1.8 0 0.1n 0.1n 80n 160n)
vin_b41 b41 gnd pulse (1.8 0 0 0.1n 0.1n 80n 160n)
.option scale=0.09u

M1000 and_cla_0/vdd xor_cla_0/a_25_n47# xor_cla_0/a_6_19#
xor_cla_0/w_n72_13# pfet w=19 l=4
+  ad=4404 pd=1144 as=361 ps=76
M1001 xor_cla_0/a_n61_n43# xor_cla_0/a_25_n47# and_cla_0/gnd Gnd nfet w=12
l=4
+  ad=768 pd=200 as=3516 ps=1048
M1002 xor_cla_0/a_n40_n43# xor_cla_0/a_n22_n46# xor_cla_0/a_n40_19#
xor_cla_0/w_n72_13# pfet w=19 l=4
+  ad=380 pd=78 as=342 ps=74
M1003 xor_cla_0/a_n40_19# xor_cla_0/a_n49_n11# and_cla_0/vdd
xor_cla_0/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1004 xor_cla_0/a_6_19# xor_cla_0/a_2_n49# xor_cla_0/a_n40_n43#
xor_cla_0/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1005 xor_cla_0/a_n61_n43# xor_cla_0/a_n22_n46# xor_cla_0/a_n40_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=216 ps=60
M1006 and_cla_0/gnd xor_cla_0/a_2_n49# xor_cla_0/a_n61_n43# Gnd nfet w=12
l=4
+  ad=0 pd=0 as=0 ps=0
M1007 xor_cla_0/a_n40_n43# xor_cla_0/a_n49_n11# xor_cla_0/a_n61_n43# Gnd
```

```
nfet w=12 l=4
+  ad=0 pd=0 as=0 ps=0
M1008 and_cla_0/vdd xor_cla_1/a_25_n47# xor_cla_1/a_6_19#
xor_cla_1/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=361 ps=76
M1009 xor_cla_1/a_n61_n43# xor_cla_1/a_25_n47# and_cla_0/gnd Gnd nfet w=12
l=4
+  ad=768 pd=200 as=0 ps=0
M1010 xor_cla_1/a_n40_n43# xor_cla_1/a_n22_n46# xor_cla_1/a_n40_19#
xor_cla_1/w_n72_13# pfet w=19 l=4
+  ad=380 pd=78 as=342 ps=74
M1011 xor_cla_1/a_n40_19# xor_cla_1/a_n49_n11# and_cla_0/vdd
xor_cla_1/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1012 xor_cla_1/a_6_19# xor_cla_1/a_2_n49# xor_cla_1/a_n40_n43#
xor_cla_1/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1013 xor_cla_1/a_n61_n43# xor_cla_1/a_n22_n46# xor_cla_1/a_n40_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=216 ps=60
M1014 and_cla_0/gnd xor_cla_1/a_2_n49# xor_cla_1/a_n61_n43# Gnd nfet w=12
l=4
+  ad=0 pd=0 as=0 ps=0
M1015 xor_cla_1/a_n40_n43# xor_cla_1/a_n49_n11# xor_cla_1/a_n61_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=0 ps=0
M1016 and_cla_0/vdd xor_cla_2/a_25_n47# xor_cla_2/a_6_19#
xor_cla_2/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=361 ps=76
M1017 xor_cla_2/a_n61_n43# xor_cla_2/a_25_n47# and_cla_0/gnd Gnd nfet w=12
l=4
+  ad=768 pd=200 as=0 ps=0
M1018 xor_cla_2/a_n40_n43# xor_cla_2/a_n22_n46# xor_cla_2/a_n40_19#
xor_cla_2/w_n72_13# pfet w=19 l=4
+  ad=380 pd=78 as=342 ps=74
M1019 xor_cla_2/a_n40_19# xor_cla_2/a_n49_n11# and_cla_0/vdd
xor_cla_2/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1020 xor_cla_2/a_6_19# xor_cla_2/a_2_n49# xor_cla_2/a_n40_n43#
xor_cla_2/w_n72_13# pfet w=19 l=4
```

```
+  ad=0 pd=0 as=0 ps=0
M1021 xor_cla_2/a_n61_n43# xor_cla_2/a_n22_n46# xor_cla_2/a_n40_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=216 ps=60
M1022 and_cla_0/gnd xor_cla_2/a_2_n49# xor_cla_2/a_n61_n43# Gnd nfet w=12
l=4
+  ad=0 pd=0 as=0 ps=0
M1023 xor_cla_2/a_n40_n43# xor_cla_2/a_n49_n11# xor_cla_2/a_n61_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=0 ps=0
M1024 and_cla_0/vdd xor_cla_3/a_25_n47# xor_cla_3/a_6_19#
xor_cla_3/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=361 ps=76
M1025 xor_cla_3/a_n61_n43# xor_cla_3/a_25_n47# and_cla_0/gnd Gnd nfet w=12
l=4
+  ad=768 pd=200 as=0 ps=0
M1026 xor_cla_3/a_n40_n43# xor_cla_3/a_n22_n46# xor_cla_3/a_n40_19#
xor_cla_3/w_n72_13# pfet w=19 l=4
+  ad=380 pd=78 as=342 ps=74
M1027 xor_cla_3/a_n40_19# m1_n356_79# and_cla_0/vdd xor_cla_3/w_n72_13#
pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1028 xor_cla_3/a_6_19# xor_cla_3/a_2_n49# xor_cla_3/a_n40_n43#
xor_cla_3/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1029 xor_cla_3/a_n61_n43# xor_cla_3/a_n22_n46# xor_cla_3/a_n40_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=216 ps=60
M1030 and_cla_0/gnd xor_cla_3/a_2_n49# xor_cla_3/a_n61_n43# Gnd nfet w=12
l=4
+  ad=0 pd=0 as=0 ps=0
M1031 xor_cla_3/a_n40_n43# m1_n356_79# xor_cla_3/a_n61_n43# Gnd nfet w=12
l=4
+  ad=0 pd=0 as=0 ps=0
M1032 and_cla_0/gnd and_cla_0/a_n23_18# and_cla_0/vdd and_cla_0/w_37_6#
pfet w=10 l=4
+  ad=1080 pd=296 as=0 ps=0
M1033 and_cla_0/a_n23_18# and_cla_0/a_n9_n16# and_cla_0/a_n23_n34# Gnd
nfet w=8 l=4
+  ad=72 pd=34 as=136 ps=50
```

```
M1034 and_cla_0/a_n23_18# and_cla_0/a_n30_n14# and_cla_0/vdd
and_cla_0/w_n44_12# pfet w=10 l=4
+  ad=170 pd=54 as=0 ps=0
** SOURCE/DRAIN TIED
M1035 and_cla_0/gnd and_cla_0/a_n23_18# and_cla_0/gnd Gnd nfet w=9 l=4
+  ad=0 pd=0 as=0 ps=0
M1036 and_cla_0/vdd and_cla_0/a_n9_n16# and_cla_0/a_n23_18#
and_cla_0/w_n44_12# pfet w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1037 and_cla_0/a_n23_n34# and_cla_0/a_n30_n14# and_cla_0/gnd Gnd nfet w=8
l=4
+  ad=0 pd=0 as=0 ps=0
M1038 and_cla_0/gnd and_cla_1/a_n23_18# and_cla_0/vdd and_cla_1/w_37_6#
pfet w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1039 and_cla_1/a_n23_18# and_cla_1/a_n9_n16# and_cla_1/a_n23_n34# Gnd
nfet w=8 l=4
+  ad=72 pd=34 as=136 ps=50
M1040 and_cla_1/a_n23_18# and_cla_1/a_n30_n14# and_cla_0/vdd
and_cla_1/w_n44_12# pfet w=10 l=4
+  ad=170 pd=54 as=0 ps=0
** SOURCE/DRAIN TIED
M1041 and_cla_0/gnd and_cla_1/a_n23_18# and_cla_0/gnd Gnd nfet w=9 l=4
+  ad=0 pd=0 as=0 ps=0
M1042 and_cla_0/vdd and_cla_1/a_n9_n16# and_cla_1/a_n23_18#
and_cla_1/w_n44_12# pfet w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1043 and_cla_1/a_n23_n34# and_cla_1/a_n30_n14# and_cla_0/gnd Gnd nfet w=8
l=4
+  ad=0 pd=0 as=0 ps=0
M1044 and_cla_0/gnd and_cla_3/a_n23_18# and_cla_0/vdd and_cla_3/w_37_6#
pfet w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1045 and_cla_3/a_n23_18# and_cla_3/a_n9_n16# and_cla_3/a_n23_n34# Gnd
nfet w=8 l=4
+  ad=72 pd=34 as=136 ps=50
M1046 and_cla_3/a_n23_18# and_cla_3/a_n30_n14# and_cla_0/vdd
and_cla_3/w_n44_12# pfet w=10 l=4
+  ad=170 pd=54 as=0 ps=0
** SOURCE/DRAIN TIED
```

```
M1047 and_cla_0/gnd and_cla_3/a_n23_18# and_cla_0/gnd Gnd nfet w=9 l=4
+  ad=0 pd=0 as=0 ps=0
M1048 and_cla_0/vdd and_cla_3/a_n9_n16# and_cla_3/a_n23_18#
and_cla_3/w_n44_12# pfet w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1049 and_cla_3/a_n23_n34# and_cla_3/a_n30_n14# and_cla_0/gnd Gnd nfet w=8
l=4
+  ad=0 pd=0 as=0 ps=0
M1050 and_cla_0/gnd and_cla_2/a_n23_18# and_cla_0/vdd and_cla_2/w_37_6#
pfet w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1051 and_cla_2/a_n23_18# and_cla_2/a_n9_n16# and_cla_2/a_n23_n34# Gnd
nfet w=8 l=4
+  ad=72 pd=34 as=136 ps=50
M1052 and_cla_2/a_n23_18# and_cla_2/a_n30_n14# and_cla_0/vdd
and_cla_2/w_n44_12# pfet w=10 l=4
+  ad=170 pd=54 as=0 ps=0
** SOURCE/DRAIN TIED
M1053 and_cla_0/gnd and_cla_2/a_n23_18# and_cla_0/gnd Gnd nfet w=9 l=4
+  ad=0 pd=0 as=0 ps=0
M1054 and_cla_0/vdd and_cla_2/a_n9_n16# and_cla_2/a_n23_18#
and_cla_2/w_n44_12# pfet w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1055 and_cla_2/a_n23_n34# and_cla_2/a_n30_n14# and_cla_0/gnd Gnd nfet w=8
l=4
+  ad=0 pd=0 as=0 ps=0

* Output
.control
tran 1n 200n
run
plot v(p1) v(g1)
plot v(p2) v(g2)
plot v(p3) v(g3)
plot v(p4) v(g4)
set curplottitle= "Srujana Vanka - 2020102005 - PGblock"
.endc
.end
```



Carry block

```
* SPICE3 file created from carry.ext - technology: scmos
*Carry look ahead block*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param w = {10*LAMBDA}
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'
.option scale=0.09u

M1000 a_507_77# g1 nand_0/vdd nand_0/w_n39_n8# CMOSP w=14 l=3
+  ad=336 pd=104 as=350 ps=106
M1001 a_507_77# g2 nand_0/vdd nand_0/w_n39_n8# CMOSP w=14 l=3
+  ad=0 pd=0 as=0 ps=0
```

```
M1002 nand_0/vdd p2 a_507_77# nand_0/w_n39_n8# CMOSP w=14 l=3
+  ad=0 pd=0 as=0 ps=0
M1003 nand_0/a_n18_n51# g2 nand_0/gnd Gnd CMOSN w=9 l=3
+  ad=108 pd=42 as=180 ps=58
M1004 nand_0/a_n3_n51# p2 nand_0/a_n18_n51# Gnd CMOSN w=9 l=3
+  ad=135 pd=48 as=0 ps=0
M1005 a_507_77# g1 nand_0/a_n3_n51# Gnd CMOSN w=9 l=3
+  ad=243 pd=72 as=0 ps=0
M1006 nor3_0/not_cla_0/a_3_n37# a_681_n2# nor3_0/not_cla_0/vdd
nor3_0/not_cla_0/w_n20_n1# CMOSP w=9 l=3
+  ad=135 pd=48 as=117 ps=44
M1007 nor3_0/not_cla_0/a_3_n37# a_681_n2# nor3_0/not_cla_0/gnd Gnd CMOSN
w=9 l=3
+  ad=225 pd=68 as=198 ps=62
M1008 a_681_n2# a_591_67# nor3_0/gnd Gnd CMOSN w=8 l=3
+  ad=320 pd=112 as=192 ps=80
M1009 nor3_0/gnd a_605_63# a_681_n2# Gnd CMOSN w=8 l=3
+  ad=0 pd=0 as=0 ps=0
M1010 a_681_n2# and_cla_1/gnd nor3_0/a_n7_n5# nor3_0/w_n39_n12# CMOSP w=9
l=3
+  ad=99 pd=40 as=108 ps=42
M1011 nor3_0/a_n7_n5# a_605_63# nor3_0/a_n21_n5# nor3_0/w_n39_n12# CMOSP
w=9 l=3
+  ad=0 pd=0 as=99 ps=40
M1012 nor3_0/a_n21_n5# a_591_67# nor3_0/vdd nor3_0/w_n39_n12# CMOSP w=9
l=3
+  ad=0 pd=0 as=81 ps=36
M1013 a_681_n2# and_cla_1/gnd nor3_0/gnd Gnd CMOSN w=8 l=3
+  ad=0 pd=0 as=0 ps=0
M1014 p4p3g2 a_312_n209# nor3_1/not_cla_0/vdd nor3_1/not_cla_0/w_n20_n1#
CMOSP w=9 l=3
+  ad=135 pd=48 as=117 ps=44
M1015 p4p3g2 a_312_n209# nor3_1/not_cla_0/gnd Gnd CMOSN w=9 l=3
+  ad=225 pd=68 as=198 ps=62
M1016 a_312_n209# g2 nor3_1/gnd Gnd CMOSN w=8 l=3
+  ad=320 pd=112 as=192 ps=80
M1017 nor3_1/gnd g2 a_312_n209# Gnd CMOSN w=8 l=3
+  ad=0 pd=0 as=0 ps=0
M1018 a_312_n209# a_256_n157# nor3_1/a_n7_n5# nor3_1/w_n39_n12# CMOSP w=9
l=3
```

```
+  ad=99 pd=40 as=108 ps=42
M1019 nor3_1/a_n7_n5# g2 nor3_1/a_n21_n5# nor3_1/w_n39_n12# CMOSP w=9 l=3
+  ad=0 pd=0 as=99 ps=40
M1020 nor3_1/a_n21_n5# g2 nor3_1/vdd nor3_1/w_n39_n12# CMOSP w=9 l=3
+  ad=0 pd=0 as=81 ps=36
M1021 a_312_n209# a_256_n157# nor3_1/gnd Gnd CMOSN w=8 l=3
+  ad=0 pd=0 as=0 ps=0
M1022 p4p3p2g1 a_418_n113# and4_0/not_cla_0/vdd and4_0/not_cla_0/w_n20_n1#
CMOSP w=9 l=3
+  ad=135 pd=48 as=117 ps=44
M1023 p4p3p2g1 a_418_n113# and4_0/not_cla_0/gnd Gnd CMOSN w=9 l=3
+  ad=225 pd=68 as=198 ps=62
M1024 a_418_n113# g1 and4_0/a_16_n59# Gnd CMOSN w=13 l=3
+  ad=325 pd=76 as=364 ps=82
M1025 and4_0/a_n41_n59# g2 and4_0/gnd Gnd CMOSN w=13 l=3
+  ad=299 pd=72 as=286 ps=70
M1026 and4_0/vdd g1 a_418_n113# and4_0/w_n67_17# CMOSP w=11 l=3
+  ad=616 pd=178 as=561 ps=146
M1027 and4_0/a_16_n59# p2 and4_0/a_n15_n59# Gnd CMOSN w=13 l=3
+  ad=0 pd=0 as=364 ps=82
M1028 a_418_n113# g2 and4_0/vdd and4_0/w_n67_17# CMOSP w=11 l=3
+  ad=0 pd=0 as=0 ps=0
M1029 a_418_n113# p2 and4_0/vdd and4_0/w_n67_17# CMOSP w=11 l=3
+  ad=0 pd=0 as=0 ps=0
M1030 and4_0/vdd g2 a_418_n113# and4_0/w_n67_17# CMOSP w=11 l=3
+  ad=0 pd=0 as=0 ps=0
M1031 and4_0/a_n15_n59# g2 and4_0/a_n41_n59# Gnd CMOSN w=13 l=3
+  ad=0 pd=0 as=0 ps=0
M1032 cout or4_0/gnd or4_0/not_cla_0/vdd or4_0/not_cla_0/w_n20_n1# CMOSP
w=9 l=3
+  ad=135 pd=48 as=117 ps=44
M1033 cout or4_0/gnd or4_0/not_cla_0/gnd Gnd CMOSN w=9 l=3
+  ad=225 pd=68 as=198 ps=62
M1034 or4_0/a_n5_9# or4_0/a_n8_n59# or4_0/a_n21_9# or4_0/w_n42_0# CMOSP
w=12 l=3
+  ad=156 pd=50 as=156 ps=50
M1035 or4_0/gnd g4 or4_0/a_11_n56# Gnd CMOSN w=8 l=3
+  ad=352 pd=136 as=96 ps=40
M1036 or4_0/a_11_9# or4_0/a_8_n60# or4_0/a_n5_9# or4_0/w_n42_0# CMOSP w=12
l=3
```

```
+  ad=144 pd=48 as=0 ps=0
M1037 or4_0/gnd or4_0/a_n8_n59# or4_0/a_n21_n56# Gnd CMOSN w=8 l=3
+  ad=0 pd=0 as=104 ps=42
M1038 or4_0/a_n21_n56# or4_0/a_n24_n59# or4_0/gnd Gnd CMOSN w=8 l=3
+  ad=0 pd=0 as=0 ps=0
M1039 or4_0/a_n21_9# or4_0/a_n24_n59# or4_0/vdd or4_0/w_n42_0# CMOSP w=12
l=3
+  ad=0 pd=0 as=120 ps=44
M1040 or4_0/gnd g4 or4_0/a_11_9# or4_0/w_n42_0# CMOSP w=12 l=3
+  ad=120 pd=44 as=0 ps=0
M1041 or4_0/a_11_n56# or4_0/a_8_n60# or4_0/gnd Gnd CMOSN w=8 l=3
+  ad=0 pd=0 as=0 ps=0
M1042 a_591_67# a_507_77# not_cla_0/vdd not_cla_0/w_n20_n1# CMOSP w=9 l=3
+  ad=135 pd=48 as=117 ps=44
M1043 a_591_67# a_507_77# not_cla_0/gnd Gnd CMOSN w=9 l=3
+  ad=225 pd=68 as=198 ps=62
M1044 p2g1 and_cla_0/a_n23_18# and_cla_0/vdd and_cla_0/w_37_6# CMOSP w=10
l=4
+  ad=270 pd=74 as=360 ps=132
M1045 and_cla_0/a_n23_18# m1_n11_79# and_cla_0/a_n23_n34# Gnd CMOSN w=8
l=4
+  ad=72 pd=34 as=136 ps=50
M1046 and_cla_0/a_n23_18# p1 and_cla_0/vdd and_cla_0/w_n44_12# CMOSP w=10
l=4
+  ad=170 pd=54 as=0 ps=0
** SOURCE/DRAIN TIED
M1047 p2g1 and_cla_0/a_n23_18# p2g1 Gnd CMOSN w=9 l=4
+  ad=651 pd=200 as=0 ps=0
M1048 and_cla_0/vdd m1_n11_79# and_cla_0/a_n23_18# and_cla_0/w_n44_12#
CMOSP w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1049 and_cla_0/a_n23_n34# p1 p2g1 Gnd CMOSN w=8 l=4
+  ad=0 pd=0 as=0 ps=0
M1050 and_cla_1/gnd and_cla_1/a_n23_18# and_cla_1/vdd and_cla_1/w_37_6#
CMOSP w=10 l=4
+  ad=270 pd=74 as=360 ps=132
M1051 and_cla_1/a_n23_18# g2 and_cla_1/a_n23_n34# Gnd CMOSN w=8 l=4
+  ad=72 pd=34 as=136 ps=50
M1052 and_cla_1/a_n23_18# g2 and_cla_1/vdd and_cla_1/w_n44_12# CMOSP w=10
l=4
```

```
+  ad=170 pd=54 as=0 ps=0
** SOURCE/DRAIN TIED
M1053 and_cla_1/gnd and_cla_1/a_n23_18# and_cla_1/gnd Gnd CMOSN w=9 l=4
+  ad=651 pd=200 as=0 ps=0
M1054 and_cla_1/vdd g2 and_cla_1/a_n23_18# and_cla_1/w_n44_12# CMOSP w=10
l=4
+  ad=0 pd=0 as=0 ps=0
M1055 and_cla_1/a_n23_n34# g2 and_cla_1/gnd Gnd CMOSN w=8 l=4
+  ad=0 pd=0 as=0 ps=0
M1056 p4g3 and_cla_2/a_n23_18# and_cla_2/vdd and_cla_2/w_37_6# CMOSP w=10
l=4
+  ad=270 pd=74 as=360 ps=132
M1057 and_cla_2/a_n23_18# m1_272_n368# and_cla_2/a_n23_n34# Gnd CMOSN w=8
l=4
+  ad=72 pd=34 as=136 ps=50
M1058 and_cla_2/a_n23_18# m1_250_n366# and_cla_2/vdd and_cla_2/w_n44_12#
CMOSP w=10 l=4
+  ad=170 pd=54 as=0 ps=0
** SOURCE/DRAIN TIED
M1059 p4g3 and_cla_2/a_n23_18# p4g3 Gnd CMOSN w=9 l=4
+  ad=651 pd=200 as=0 ps=0
M1060 and_cla_2/vdd m1_272_n368# and_cla_2/a_n23_18# and_cla_2/w_n44_12#
CMOSP w=10 l=4
+  ad=0 pd=0 as=0 ps=0
M1061 and_cla_2/a_n23_n34# m1_250_n366# p4g3 Gnd CMOSN w=8 l=4
+  ad=0 pd=0 as=0 ps=0
M1062 or_cla_0/a_74_n55# or_cla_0/a_n6_n54# or_cla_0/gnd Gnd CMOSN w=16
l=3
+  ad=464 pd=90 as=1056 ps=228
M1063 or_cla_0/a_n6_n54# m1_154_106# or_cla_0/a_n6_14# or_cla_0/w_n27_6#
CMOSP w=9 l=3
+  ad=144 pd=50 as=117 ps=44
M1064 or_cla_0/a_n6_n54# p2g1 or_cla_0/gnd Gnd CMOSN w=16 l=3
+  ad=208 pd=58 as=0 ps=0
M1065 or_cla_0/gnd m1_154_106# or_cla_0/a_n6_n54# Gnd CMOSN w=16 l=3
+  ad=0 pd=0 as=0 ps=0
M1066 or_cla_0/a_n6_14# p2g1 or_cla_0/vdd or_cla_0/w_n27_6# CMOSP w=9 l=3
+  ad=0 pd=0 as=324 ps=108
M1067 or_cla_0/a_74_n55# or_cla_0/a_n6_n54# or_cla_0/vdd or_cla_0/w_41_7#
CMOSP w=9 l=3
```

```
+  ad=243 pd=72 as=0 ps=0
C0 or4_0/a_11_n56# or4_0/gnd 0.12fF
C1 m1_250_n366# p1 0.03fF
C2 and_cla_1/a_n23_18# and_cla_1/w_n44_12# 0.04fF
C3 nor3_1/vdd nor3_1/w_n39_n12# 0.17fF
C4 m1_272_n368# g1 0.03fF
C5 a_418_n113# g1 0.16fF
C6 and_cla_2/a_n23_18# and_cla_2/w_37_6# 0.16fF
C7 and_cla_2/vdd and_cla_2/w_n44_12# 0.06fF
C8 a_681_n2# nor3_0/not_cla_0/w_n20_n1# 0.09fF
C9 g3 g4 0.36fF
C10 g2 nor3_1/w_n39_n12# 0.25fF
C11 or_cla_0/a_74_n55# g2 0.08fF
C12 a_591_67# nor3_0/w_n39_n12# 0.13fF
C13 or4_0/gnd cout 0.10fF
C14 and4_0/not_cla_0/vdd p4p3p2g1 0.06fF
C15 or_cla_0/a_n6_n54# or_cla_0/w_41_7# 0.11fF
C16 or_cla_0/vdd or_cla_0/w_n27_6# 0.03fF
C17 nor3_1/not_cla_0/vdd a_312_n209# 0.08fF
C18 p4p3g2 nor3_1/not_cla_0/w_n20_n1# 0.05fF
C19 m3_256_n157# p2 0.11fF
C20 p1 g1 4.67fF
C21 or4_0/a_n21_n56# or4_0/gnd 0.05fF
C22 g4 g1 0.34fF
C23 a_256_n157# nor3_1/w_n39_n12# 0.13fF
C24 m1_272_n368# p2 0.03fF
C25 a_418_n113# p2 0.16fF
C26 or4_0/a_n24_n59# or4_0/w_n42_0# 0.13fF
C27 p1 and_cla_0/w_n44_12# 0.09fF
C28 and_cla_2/a_n23_n34# m1_272_n368# 0.01fF
C29 nor3_0/vdd a_681_n2# 0.07fF
C30 and4_0/not_cla_0/gnd p4p3p2g1 0.09fF
C31 or4_0/gnd or4_0/not_cla_0/vdd 0.08fF
C32 g3 m1_250_n366# 0.03fF
C33 or_cla_0/a_n6_n54# or_cla_0/w_n27_6# 0.04fF
C34 m1_154_106# p2g1 0.05fF
C35 nor3_1/not_cla_0/vdd nor3_1/not_cla_0/w_n20_n1# 0.08fF
C36 p1 p2 0.60fF
C37 g4 p2 0.36fF
C38 p4p3p2g1 and4_0/not_cla_0/w_n20_n1# 0.05fF
```

```
C39 g2 and_cla_1/w_n44_12# 0.18fF
C40 or4_0/a_n24_n59# p4p3p2g1 0.16fF
C41 or4_0/a_n8_n59# p4p3g2 0.16fF
C42 a_507_77# nand_0/gnd 0.05fF
C43 m1_272_n368# g2 0.06fF
C44 nand_0/vdd g1 0.13fF
C45 g2 a_418_n113# 0.16fF
C46 m3_256_n157# a_256_n157# 0.08fF
C47 m1_250_n366# g1 0.03fF
C48 not_cla_0/vdd a_591_67# 0.06fF
C49 and_cla_2/a_n23_18# and_cla_2/w_n44_12# 0.04fF
C50 p4 g4 5.22fF
C51 p4g3 m1_272_n368# 0.40fF
C52 and4_0/w_n67_17# a_418_n113# 0.12fF
C53 and_cla_0/a_n23_18# m1_n11_79# 0.44fF
C54 and4_0/not_cla_0/vdd a_418_n113# 0.08fF
C55 p4g3 p4p3g2 0.54fF
C56 g2 p1 1.10fF
C57 a_312_n209# nor3_1/not_cla_0/w_n20_n1# 0.09fF
C58 g2 g4 0.72fF
C59 nand_0/vdd p2 0.13fF
C60 p4g3 or4_0/a_n21_n56# 0.09fF
C61 m1_250_n366# p2 0.03fF
C62 cout or4_0/not_cla_0/w_n20_n1# 0.05fF
C63 and_cla_1/gnd a_681_n2# 0.13fF
C64 a_507_77# a_591_67# 0.03fF
C65 m1_272_n368# and_cla_2/w_n44_12# 0.09fF
C66 a_605_63# a_681_n2# 0.13fF
C67 and4_0/not_cla_0/gnd a_418_n113# 0.05fF
C68 and_cla_0/vdd and_cla_0/w_37_6# 0.05fF
C69 p4 m1_250_n366# 0.20fF
C70 a_418_n113# and4_0/not_cla_0/w_n20_n1# 0.09fF
C71 or_cla_0/a_n6_14# m1_154_106# 0.01fF
C72 nor3_1/vdd a_312_n209# 0.07fF
C73 g2 nand_0/vdd 0.13fF
C74 or4_0/a_n24_n59# p4p3g2 0.16fF
C75 a_507_77# nand_0/vdd 0.11fF
C76 m1_250_n366# g2 1.69fF
C77 g1 p2 0.60fF
C78 m1_154_106# p1 0.02fF
```

```
C79 g3 p4 3.47fF
C80 or4_0/not_cla_0/vdd or4_0/not_cla_0/w_n20_n1# 0.08fF
C81 g2 a_312_n209# 0.13fF
C82 nor3_0/not_cla_0/gnd nor3_0/not_cla_0/a_3_n37# 0.09fF
C83 or4_0/vdd or4_0/w_n42_0# 0.05fF
C84 g3 g2 3.47fF
C85 m1_154_106# or_cla_0/w_n27_6# 0.28fF
C86 a_256_n157# a_312_n209# 0.13fF
C87 m3_256_n157# p4p3p2g1 0.22fF
C88 and_cla_1/a_n23_18# g2 0.46fF
C89 and_cla_1/gnd and_cla_1/w_37_6# 0.07fF
C90 p4g3 or4_0/a_8_n60# 0.10fF
C91 g2 g1 1.21fF
C92 a_507_77# g1 0.09fF
C93 nand_0/vdd nand_0/w_n39_n8# 0.27fF
C94 p4g3 or4_0/gnd 0.19fF
C95 not_cla_0/w_n20_n1# a_591_67# 0.05fF
C96 m1_250_n366# and_cla_2/w_n44_12# 0.09fF
C97 and_cla_1/gnd nor3_0/vdd 0.09fF
C98 and4_0/w_n67_17# g1 0.14fF
C99 nor3_0/w_n39_n12# a_681_n2# 0.04fF
C100 m1_272_n368# p4p3p2g1 0.19fF
C101 g2 and4_0/gnd 0.18fF
C102 p4p3p2g1 a_418_n113# 0.07fF
C103 g4 or4_0/w_n42_0# 0.13fF
C104 or4_0/gnd or4_0/not_cla_0/w_n20_n1# 0.09fF
C105 and_cla_0/a_n23_18# and_cla_0/w_37_6# 0.16fF
C106 and_cla_0/vdd and_cla_0/w_n44_12# 0.06fF
C107 nor3_0/vdd a_605_63# 0.09fF
C108 p4p3p2g1 p4p3g2 1.19fF
C109 or_cla_0/vdd m1_154_106# 0.13fF
C110 or_cla_0/a_74_n55# or_cla_0/w_41_7# 0.03fF
C111 and4_0/gnd a_256_n157# 0.15fF
C112 and_cla_1/vdd and_cla_1/w_37_6# 0.05fF
C113 g2 p2 4.69fF
C114 not_cla_0/vdd a_507_77# 0.08fF
C115 a_507_77# p2 0.09fF
C116 g1 nand_0/w_n39_n8# 0.15fF
C117 and4_0/w_n67_17# p2 0.14fF
C118 and_cla_2/a_n23_18# m1_272_n368# 0.46fF
```

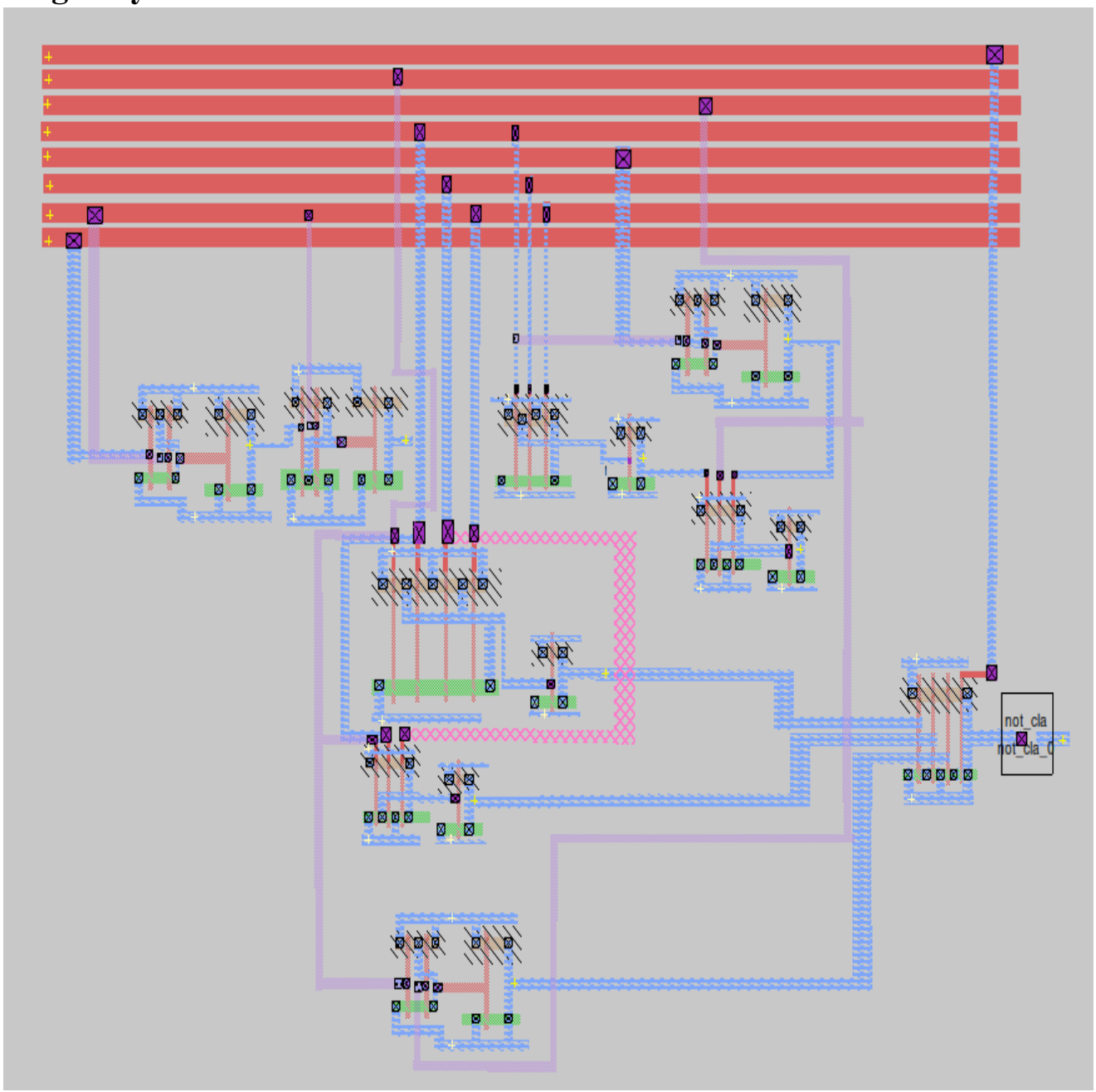
```
C119 p4g3 and_cla_2/w_37_6# 0.07fF
C120 nor3_0/not_cla_0/a_3_n37# nor3_0/not_cla_0/vdd 0.06fF
C121 m1_154_106# g1 0.14fF
C122 g2 nor3_1/vdd 0.44fF
C123 and4_0/vdd g1 0.23fF
C124 m1_n11_79# p1 0.96fF
C125 nor3_0/vdd a_591_67# 0.09fF
C126 p2g1 or_cla_0/w_n27_6# 0.11fF
C127 or_cla_0/a_n6_n54# m1_154_106# 0.28fF
C128 nor3_1/not_cla_0/gnd p4p3g2 0.09fF
C129 nor3_1/vdd a_256_n157# 0.17fF
C130 p4g3 or4_0/a_n8_n59# 0.16fF
C131 p2g1 and_cla_0/w_37_6# 0.07fF
C132 p2 nand_0/w_n39_n8# 0.15fF
C133 m1_272_n368# and_cla_1/gnd 0.16fF
C134 g2 and4_0/w_n67_17# 0.28fF
C135 or4_0/not_cla_0/gnd cout 0.09fF
C136 nor3_0/not_cla_0/a_3_n37# a_681_n2# 0.09fF
C137 m1_272_n368# a_605_63# 0.05fF
C138 or4_0/a_8_n60# or4_0/w_n42_0# 0.13fF
C139 and4_0/vdd p2 0.28fF
C140 and_cla_0/a_n23_18# and_cla_0/w_n44_12# 0.04fF
C141 nor3_0/vdd nor3_0/w_n39_n12# 0.17fF
C142 nor3_0/gnd a_681_n2# 0.16fF
C143 m1_272_n368# p4p3g2 0.19fF
C144 or4_0/gnd or4_0/w_n42_0# 0.06fF
C145 nor3_1/w_n39_n12# a_312_n209# 0.04fF
C146 m1_272_n368# p1 0.03fF
C147 and_cla_1/a_n23_18# and_cla_1/w_37_6# 0.16fF
C148 and_cla_1/vdd and_cla_1/w_n44_12# 0.06fF
C149 g2 nand_0/w_n39_n8# 0.15fF
C150 not_cla_0/vdd not_cla_0/w_n20_n1# 0.08fF
C151 a_507_77# nand_0/w_n39_n8# 0.10fF
C152 not_cla_0/gnd a_591_67# 0.09fF
C153 and_cla_2/vdd and_cla_2/w_37_6# 0.05fF
C154 nor3_0/not_cla_0/vdd a_681_n2# 0.08fF
C155 nor3_0/not_cla_0/a_3_n37# nor3_0/not_cla_0/w_n20_n1# 0.05fF
C156 g2 and4_0/vdd 0.88fF
C157 p1 g4 0.34fF
C158 and4_0/vdd and4_0/w_n67_17# 0.13fF
```

```
C159 p4p3g2 nor3_1/not_cla_0/vdd 0.06fF
C160 m1_n11_79# g1 0.31fF
C161 nor3_1/gnd a_312_n209# 0.16fF
C162 p4g3 or4_0/a_n24_n59# 0.16fF
C163 a_507_77# not_cla_0/w_n20_n1# 0.09fF
C164 and4_0/not_cla_0/vdd and4_0/not_cla_0/w_n20_n1# 0.08fF
C165 cout or4_0/not_cla_0/vdd 0.06fF
C166 and_cla_1/gnd nor3_0/w_n39_n12# 0.13fF
C167 m1_272_n368# m1_250_n366# 0.04fF
C168 nor3_0/not_cla_0/vdd nor3_0/not_cla_0/w_n20_n1# 0.08fF
C169 or4_0/a_n8_n59# or4_0/w_n42_0# 0.13fF
C170 m1_n11_79# and_cla_0/w_n44_12# 0.09fF
C171 a_605_63# nor3_0/w_n39_n12# 0.13fF
C172 g3 m1_272_n368# 0.20fF
C173 p4p3g2 a_312_n209# 0.05fF
C174 or_cla_0/vdd or_cla_0/w_41_7# 0.03fF
C175 m3_256_n157# g1 0.13fF
C176 g3 Gnd 17.85fF **FLOATING
C177 p4 Gnd 17.86fF **FLOATING
C178 or_cla_0/gnd Gnd 0.71fF
C179 or_cla_0/a_74_n55# Gnd 0.31fF
C180 or_cla_0/vdd Gnd 0.46fF
C181 or_cla_0/a_n6_n54# Gnd 1.17fF
C182 p2g1 Gnd 1.79fF
C183 or_cla_0/w_41_7# Gnd 1.55fF
C184 or_cla_0/w_n27_6# Gnd 1.42fF
C185 p4g3 Gnd 1.67fF
C186 and_cla_2/vdd Gnd 0.68fF
C187 and_cla_2/a_n23_18# Gnd 1.14fF
C188 m1_272_n368# Gnd 1.48fF
C189 m1_250_n366# Gnd 0.74fF
C190 and_cla_2/w_37_6# Gnd 1.96fF
C191 and_cla_2/w_n44_12# Gnd 1.33fF
C192 and_cla_1/gnd Gnd 1.68fF
C193 and_cla_1/vdd Gnd 0.68fF
C194 and_cla_1/a_n23_18# Gnd 1.14fF
C195 g2 Gnd 45.57fF
C196 and_cla_1/w_37_6# Gnd 1.96fF
C197 and_cla_1/w_n44_12# Gnd 1.33fF
C198 and_cla_0/vdd Gnd 0.68fF
```

```
C199 and_cla_0/a_n23_18# Gnd 1.14fF
C200 m1_n11_79# Gnd 1.07fF
C201 p1 Gnd 19.76fF
C202 and_cla_0/w_37_6# Gnd 1.96fF
C203 and_cla_0/w_n44_12# Gnd 1.33fF
C204 not_cla_0/gnd Gnd 0.20fF
C205 not_cla_0/vdd Gnd 0.12fF
C206 a_507_77# Gnd 0.53fF
C207 not_cla_0/w_n20_n1# Gnd 0.93fF
C208 or4_0/a_11_n56# Gnd 0.02fF
C209 or4_0/a_n21_n56# Gnd 0.02fF
C210 or4_0/gnd Gnd 1.60fF
C211 or4_0/vdd Gnd 0.41fF
C212 g4 Gnd 20.54fF
C213 or4_0/a_8_n60# Gnd 0.40fF
C214 or4_0/a_n8_n59# Gnd 0.39fF
C215 or4_0/a_n24_n59# Gnd 0.38fF
C216 or4_0/w_n42_0# Gnd 2.45fF
C217 or4_0/not_cla_0/gnd Gnd 0.20fF
C218 cout Gnd 0.18fF
C219 or4_0/not_cla_0/vdd Gnd 0.12fF
C220 or4_0/not_cla_0/w_n20_n1# Gnd 0.93fF
C221 and4_0/gnd Gnd 0.60fF
C222 and4_0/vdd Gnd 0.65fF
C223 and4_0/w_n67_17# Gnd 4.31fF
C224 and4_0/not_cla_0/gnd Gnd 0.20fF
C225 p4p3p2g1 Gnd 0.37fF
C226 and4_0/not_cla_0/vdd Gnd 0.12fF
C227 a_418_n113# Gnd 1.62fF
C228 and4_0/not_cla_0/w_n20_n1# Gnd 0.93fF
C229 nor3_1/gnd Gnd 0.39fF
C230 nor3_1/vdd Gnd 0.00fF
C231 a_256_n157# Gnd 0.34fF
C232 nor3_1/w_n39_n12# Gnd 1.70fF
C233 nor3_1/not_cla_0/gnd Gnd 0.20fF
C234 p4p3g2 Gnd 0.39fF
C235 nor3_1/not_cla_0/vdd Gnd 0.12fF
C236 a_312_n209# Gnd 0.90fF
C237 nor3_1/not_cla_0/w_n20_n1# Gnd 0.93fF
C238 nor3_0/gnd Gnd 0.39fF
```

```
C239 nor3_0/vdd Gnd 0.00fF
C240 a_605_63# Gnd 0.34fF
C241 a_591_67# Gnd 0.57fF
C242 nor3_0/w_n39_n12# Gnd 1.70fF
C243 nor3_0/not_cla_0/gnd Gnd 0.20fF
C244 nor3_0/not_cla_0/a_3_n37# Gnd 0.14fF
C245 nor3_0/not_cla_0/vdd Gnd 0.12fF
C246 a_681_n2# Gnd 0.97fF
C247 nor3_0/not_cla_0/w_n20_n1# Gnd 0.93fF
C248 nand_0/gnd Gnd 0.32fF
C249 nand_0/vdd Gnd 0.04fF
C250 g1 Gnd 20.74fF
C251 p2 Gnd 20.85fF
C252 nand_0/w_n39_n8# Gnd 2.60fF

* Output
.control
tran 1n 200n
run
plot v(g1)
plot v(c2)
plot v(c3)
plot v(c4)
set curplottitle= "Srujana Vanka - 2020102005 - carry"
.endc
.end
```

Magic layout



Sum block
Netlist

```
* SPICE3 file created from sum.ext - technology: scmos
*Sum block*

.include TSMC_180nm.txt

.param SUPPLY=1.8
.param LAMBDA=0.09u
.param w = {10*LAMBDA}
.param width_P={20*LAMBDA}
.param width_N={10*LAMBDA}
.global gnd vdd

vdd vdd gnd 'SUPPLY'

.option scale=0.09u

M1000 xor_cla_0/vdd xor_cla_0/a_25_n47# xor_cla_0/a_6_19#
xor_cla_0/w_n72_13# pfet w=19 l=4
+  ad=741 pd=154 as=361 ps=76
M1001 xor_cla_0/a_n61_n43# xor_cla_0/a_25_n47# xor_cla_0/gnd Gnd nfet w=12
l=4
+  ad=768 pd=200 as=228 ps=62
M1002 xor_cla_0/a_n40_n43# xor_cla_0/a_n22_n46# xor_cla_0/a_n40_19#
xor_cla_0/w_n72_13# pfet w=19 l=4
+  ad=380 pd=78 as=342 ps=74
M1003 xor_cla_0/a_n40_19# xor_cla_0/a_n49_n11# xor_cla_0/vdd
xor_cla_0/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1004 xor_cla_0/a_6_19# xor_cla_0/a_2_n49# xor_cla_0/a_n40_n43#
xor_cla_0/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1005 xor_cla_0/a_n61_n43# xor_cla_0/a_n22_n46# xor_cla_0/a_n40_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=216 ps=60
M1006 xor_cla_0/gnd xor_cla_0/a_2_n49# xor_cla_0/a_n61_n43# Gnd nfet w=12
l=4
```

```
+  ad=0 pd=0 as=0 ps=0
M1007 xor_cla_0/a_n40_n43# xor_cla_0/a_n49_n11# xor_cla_0/a_n61_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=0 ps=0
M1008 xor_cla_1/vdd xor_cla_1/a_25_n47# xor_cla_1/a_6_19#
xor_cla_1/w_n72_13# pfet w=19 l=4
+  ad=741 pd=154 as=361 ps=76
M1009 xor_cla_1/a_n61_n43# xor_cla_1/a_25_n47# xor_cla_1/gnd Gnd nfet w=12
l=4
+  ad=768 pd=200 as=228 ps=62
M1010 xor_cla_1/a_n40_n43# xor_cla_1/a_n22_n46# xor_cla_1/a_n40_19#
xor_cla_1/w_n72_13# pfet w=19 l=4
+  ad=380 pd=78 as=342 ps=74
M1011 xor_cla_1/a_n40_19# xor_cla_1/a_n49_n11# xor_cla_1/vdd
xor_cla_1/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1012 xor_cla_1/a_6_19# xor_cla_1/a_2_n49# xor_cla_1/a_n40_n43#
xor_cla_1/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1013 xor_cla_1/a_n61_n43# xor_cla_1/a_n22_n46# xor_cla_1/a_n40_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=216 ps=60
M1014 xor_cla_1/gnd xor_cla_1/a_2_n49# xor_cla_1/a_n61_n43# Gnd nfet w=12
l=4
+  ad=0 pd=0 as=0 ps=0
M1015 xor_cla_1/a_n40_n43# xor_cla_1/a_n49_n11# xor_cla_1/a_n61_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=0 ps=0
M1016 xor_cla_3/vdd xor_cla_3/a_25_n47# xor_cla_3/a_6_19#
xor_cla_3/w_n72_13# pfet w=19 l=4
+  ad=741 pd=154 as=361 ps=76
M1017 xor_cla_3/a_n61_n43# xor_cla_3/a_25_n47# xor_cla_3/gnd Gnd nfet w=12
l=4
+  ad=768 pd=200 as=228 ps=62
M1018 xor_cla_3/a_n40_n43# xor_cla_3/a_n22_n46# xor_cla_3/a_n40_19#
xor_cla_3/w_n72_13# pfet w=19 l=4
+  ad=380 pd=78 as=342 ps=74
M1019 xor_cla_3/a_n40_19# xor_cla_3/a_n49_n11# xor_cla_3/vdd
xor_cla_3/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
```

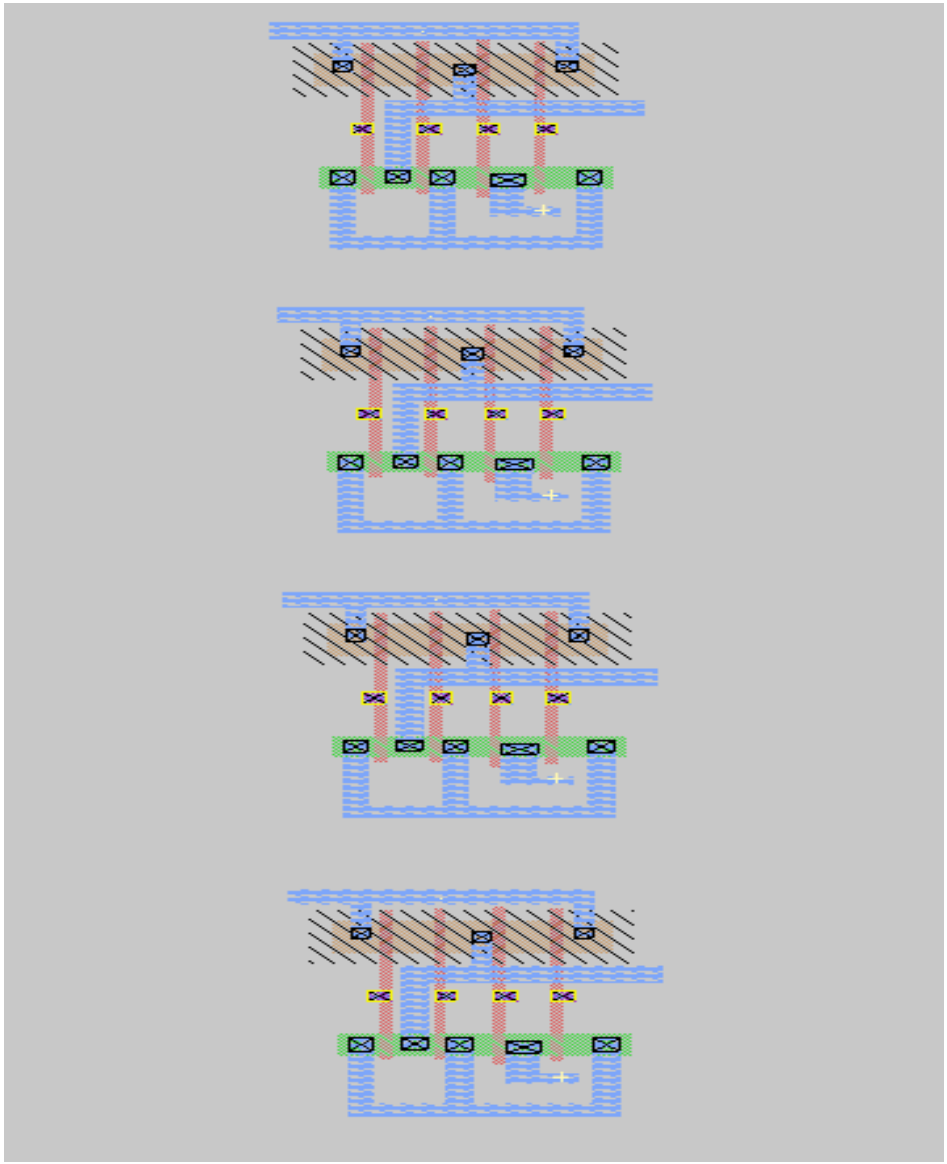
```
M1020 xor_cla_3/a_6_19# xor_cla_3/a_2_n49# xor_cla_3/a_n40_n43#
xor_cla_3/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1021 xor_cla_3/a_n61_n43# xor_cla_3/a_n22_n46# xor_cla_3/a_n40_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=216 ps=60
M1022 xor_cla_3/gnd xor_cla_3/a_2_n49# xor_cla_3/a_n61_n43# Gnd nfet w=12
l=4
+  ad=0 pd=0 as=0 ps=0
M1023 xor_cla_3/a_n40_n43# xor_cla_3/a_n49_n11# xor_cla_3/a_n61_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=0 ps=0
M1024 xor_cla_2/vdd xor_cla_2/a_25_n47# xor_cla_2/a_6_19#
xor_cla_2/w_n72_13# pfet w=19 l=4
+  ad=741 pd=154 as=361 ps=76
M1025 xor_cla_2/a_n61_n43# xor_cla_2/a_25_n47# xor_cla_2/gnd Gnd nfet w=12
l=4
+  ad=768 pd=200 as=228 ps=62
M1026 xor_cla_2/a_n40_n43# xor_cla_2/a_n22_n46# xor_cla_2/a_n40_19#
xor_cla_2/w_n72_13# pfet w=19 l=4
+  ad=380 pd=78 as=342 ps=74
M1027 xor_cla_2/a_n40_19# xor_cla_2/a_n49_n11# xor_cla_2/vdd
xor_cla_2/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1028 xor_cla_2/a_6_19# xor_cla_2/a_2_n49# xor_cla_2/a_n40_n43#
xor_cla_2/w_n72_13# pfet w=19 l=4
+  ad=0 pd=0 as=0 ps=0
M1029 xor_cla_2/a_n61_n43# xor_cla_2/a_n22_n46# xor_cla_2/a_n40_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=216 ps=60
M1030 xor_cla_2/gnd xor_cla_2/a_2_n49# xor_cla_2/a_n61_n43# Gnd nfet w=12
l=4
+  ad=0 pd=0 as=0 ps=0
M1031 xor_cla_2/a_n40_n43# xor_cla_2/a_n49_n11# xor_cla_2/a_n61_n43# Gnd
nfet w=12 l=4
+  ad=0 pd=0 as=0 ps=0
C0 xor_cla_3/a_n49_n11# xor_cla_3/w_n72_13# 0.11fF
C1 xor_cla_3/a_n40_n43# xor_cla_3/a_25_n47# 0.25fF
C2 xor_cla_2/a_n40_n43# xor_cla_2/a_n22_n46# 0.35fF
C3 xor_cla_2/gnd xor_cla_2/a_n61_n43# 0.03fF
```

```
C4 xor_cla_0/w_n72_13# xor_cla_0/a_25_n47# 0.11fF
C5 xor_cla_1/a_n40_n43# xor_cla_1/w_n72_13# 0.03fF
C6 xor_cla_3/a_n40_n43# xor_cla_3/a_2_n49# 0.25fF
C7 xor_cla_2/a_n40_n43# xor_cla_2/a_n49_n11# 0.06fF
C8 xor_cla_0/w_n72_13# xor_cla_0/a_2_n49# 0.11fF
C9 xor_cla_1/a_n40_n43# xor_cla_1/a_25_n47# 0.25fF
C10 xor_cla_3/a_n40_n43# xor_cla_3/a_n22_n46# 0.35fF
C11 xor_cla_3/gnd xor_cla_3/a_n61_n43# 0.03fF
C12 xor_cla_2/a_n40_n43# xor_cla_2/w_n72_13# 0.03fF
C13 xor_cla_0/w_n72_13# xor_cla_0/a_n22_n46# 0.11fF
C14 xor_cla_1/a_n40_n43# xor_cla_1/a_2_n49# 0.25fF
C15 xor_cla_0/a_n40_n43# xor_cla_0/a_n49_n11# 0.06fF
C16 xor_cla_0/w_n72_13# xor_cla_0/vdd 0.05fF
C17 xor_cla_1/vdd xor_cla_1/w_n72_13# 0.05fF
C18 xor_cla_3/a_n40_n43# xor_cla_3/a_n49_n11# 0.06fF
C19 xor_cla_1/a_n40_n43# xor_cla_1/a_n22_n46# 0.35fF
C20 xor_cla_1/gnd xor_cla_1/a_n61_n43# 0.03fF
C21 xor_cla_1/a_25_n47# xor_cla_1/w_n72_13# 0.11fF
C22 xor_cla_3/a_n40_n43# xor_cla_3/w_n72_13# 0.03fF
C23 xor_cla_2/vdd xor_cla_2/w_n72_13# 0.05fF
C24 xor_cla_1/a_n40_n43# xor_cla_1/a_n49_n11# 0.06fF
C25 xor_cla_1/a_2_n49# xor_cla_1/w_n72_13# 0.11fF
C26 xor_cla_2/a_25_n47# xor_cla_2/w_n72_13# 0.11fF
C27 xor_cla_1/a_n22_n46# xor_cla_1/w_n72_13# 0.11fF
C28 xor_cla_3/vdd xor_cla_3/w_n72_13# 0.05fF
C29 xor_cla_0/a_25_n47# xor_cla_0/a_n40_n43# 0.25fF
C30 xor_cla_0/gnd xor_cla_0/a_n61_n43# 0.03fF
C31 xor_cla_2/a_2_n49# xor_cla_2/w_n72_13# 0.11fF
C32 xor_cla_1/a_n49_n11# xor_cla_1/w_n72_13# 0.11fF
C33 xor_cla_3/a_25_n47# xor_cla_3/w_n72_13# 0.11fF
C34 xor_cla_0/a_2_n49# xor_cla_0/a_n40_n43# 0.25fF
C35 xor_cla_2/a_n22_n46# xor_cla_2/w_n72_13# 0.11fF
C36 xor_cla_0/w_n72_13# xor_cla_0/a_n49_n11# 0.11fF
C37 xor_cla_3/a_2_n49# xor_cla_3/w_n72_13# 0.11fF
C38 xor_cla_0/a_n40_n43# xor_cla_0/a_n22_n46# 0.35fF
C39 xor_cla_2/a_n49_n11# xor_cla_2/w_n72_13# 0.11fF
C40 xor_cla_2/a_n40_n43# xor_cla_2/a_25_n47# 0.25fF
C41 xor_cla_3/a_n22_n46# xor_cla_3/w_n72_13# 0.11fF
C42 xor_cla_2/a_n40_n43# xor_cla_2/a_2_n49# 0.25fF
C43 xor_cla_0/w_n72_13# xor_cla_0/a_n40_n43# 0.03fF
```

```
C44 xor_cla_2/gnd Gnd 0.17fF
C45 xor_cla_2/a_n61_n43# Gnd 0.91fF
C46 xor_cla_2/a_n40_n43# Gnd 0.62fF
C47 xor_cla_2/vdd Gnd 0.63fF
C48 xor_cla_2/a_25_n47# Gnd 0.47fF
C49 xor_cla_2/a_2_n49# Gnd 0.49fF
C50 xor_cla_2/a_n22_n46# Gnd 0.47fF
C51 xor_cla_2/a_n49_n11# Gnd 0.46fF
C52 xor_cla_2/w_n72_13# Gnd 4.05fF
C53 xor_cla_3/gnd Gnd 0.17fF
C54 xor_cla_3/a_n61_n43# Gnd 0.91fF
C55 xor_cla_3/a_n40_n43# Gnd 0.62fF
C56 xor_cla_3/vdd Gnd 0.63fF
C57 xor_cla_3/a_25_n47# Gnd 0.47fF
C58 xor_cla_3/a_2_n49# Gnd 0.49fF
C59 xor_cla_3/a_n22_n46# Gnd 0.47fF
C60 xor_cla_3/a_n49_n11# Gnd 0.46fF
C61 xor_cla_3/w_n72_13# Gnd 4.05fF
C62 xor_cla_1/gnd Gnd 0.17fF
C63 xor_cla_1/a_n61_n43# Gnd 0.91fF
C64 xor_cla_1/a_n40_n43# Gnd 0.62fF
C65 xor_cla_1/vdd Gnd 0.63fF
C66 xor_cla_1/a_25_n47# Gnd 0.47fF
C67 xor_cla_1/a_2_n49# Gnd 0.49fF
C68 xor_cla_1/a_n22_n46# Gnd 0.47fF
C69 xor_cla_1/a_n49_n11# Gnd 0.46fF
C70 xor_cla_1/w_n72_13# Gnd 4.05fF
C71 xor_cla_0/gnd Gnd 0.17fF
C72 xor_cla_0/a_n61_n43# Gnd 0.91fF
C73 xor_cla_0/a_n40_n43# Gnd 0.62fF
C74 xor_cla_0/vdd Gnd 0.63fF
C75 xor_cla_0/a_25_n47# Gnd 0.47fF
C76 xor_cla_0/a_2_n49# Gnd 0.49fF
C77 xor_cla_0/a_n22_n46# Gnd 0.47fF
C78 xor_cla_0/a_n49_n11# Gnd 0.46fF
C79 xor_cla_0/w_n72_13# Gnd 4.05fF

* Output
.control
tran 1n 200n
```

```
run
plot v(c1) v(c2) v(c3) v(c4)
set curplottitle= "Srujana Vanka - 2020102005 - PGblock"
.endc
.end
```

Magic layout



Question 9

Delays:

	Prelayout	Post layout
Sum/Propagate	0.026ns	0.029ns
Generate	0.041ns	0.069ns
Carry	0.24ns	0.29ns

	Pre layout	Post layout
ALL BLOCKS	0.61ns	0.93ns

Question 10

Using Verilog HDL to write the structural description

```
module CLA_Adder(cin,a1,a2,a3,a4,b1,b2,b3,b4,s1,s2,s3,s4,cout);
input cin,a1,a2,a3,a4,b1,b2,b3,b4;
output s1,s2,s3,s4,cout;
wire p1,p2,p3,p4,g1,g2,g3,g4,t1,t2,t3,t4,t5,t6;

// propagate and generate
xor #1 (p1,a1,b1);
xor #2 (p2,a2,b2);
xor #3 (p3,a3,b3);
xor #4 (p4,a4,b4);

and #1 (g1,a1,b1);
and #2 (g2,a2,b2);
```

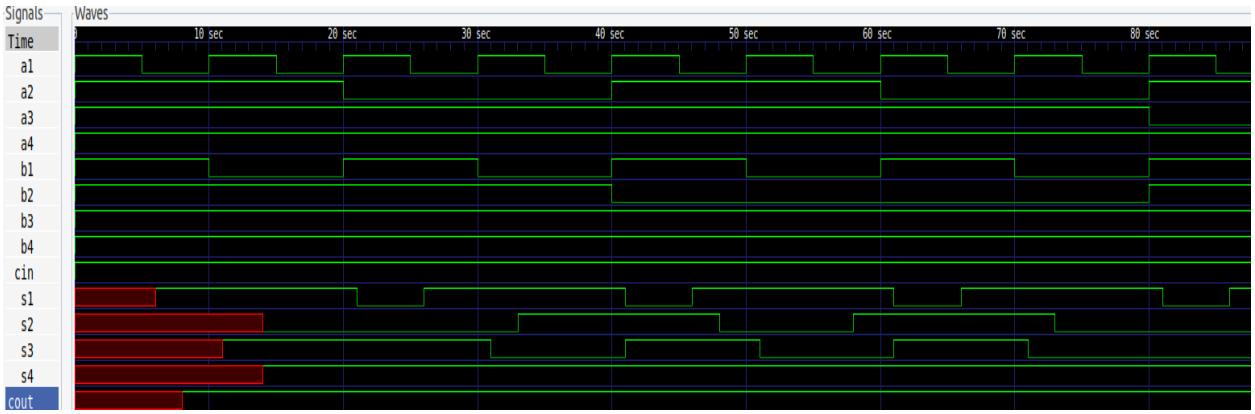
```
and #3 (g3,a3,b3);
and #4 (g4,a4,b4);

// carry
// C1
and #5 (t1, p1, cin);
or #1 (c1, g1, t1);

// C2
and #6 (t2, p1, p2, cin);
and #7 (t3, g1, p2);
or #2 (c2, t2, t3, g1);

// C3
and #8 (t4, p1, p2, p3, cin);
and #9 (t5, g1, p2, p3);
and #10 (t6, g2, p3);
or #3 (c3, g2, t6, t5, t4);

// C4
and #11 (t7, p1, p2, p3, p4, cin);
and #12 (t8, g1, p2, p3, p4);
and #13(t9, g2, p3, p4);
and #14(t10, g3, p4);
or #4 (c4, g3, t10, t9, t8, t7);

// sum
xor #5 (s1,p1,c1);
xor #6 (s2,p2,c2);
xor #7 (s3,p3,c3);
xor #8 (s4,p4,c4);

endmodule
```

Testbench

```
module testbench;
reg cin,a1,a2,a3,a4,b1,b2,b3,b4;
wire s1,s2,s3,s4,c5;
CLA_Adder DUT(cin,a1,a2,a3,a4,b1,b2,b3,b4,s1,s2,s3,s4,cout);
```

```
initial begin
$dumpfile("cla.vcd");
$dumpvars(0,testbench);
//print statement
$monitor($time," A=%b%b%b%b B=%b%b%b%b Cout=%b
S=%b%b%b%b",a4,a3,a2,a1,b4,b3,b2,b1,cout,s4,s3,s2,s1);
   cin = 1'b1;
   a1 = 1'b1;
   b1 = 1'b1;
   a2 = 1'b1;
   b2 = 1'b1;
   a3 = 1'b1;
   b3 = 1'b1;
   a4 = 1'b1;
   b4 = 1'b1;

   #2560 $finish;
end
always #5 a1 = ~a1;
always #10 b1 = ~b1;
always #20 a2 = ~a2;
always #40 b2 = ~b2;
always #80 a3 = ~a3;
always #160 b3 = ~b3;
always #320 a4 = ~a4;
always #640 b4 = ~b4;
always #1280 cin = ~cin;
endmodule
```

Waveform


Signals
Waves
Time
a1
a2
a3
a4
b1
b2
b3
b4
cin
s1
s2
s3
s4
cout
10 sec
20 sec
30 sec
40 sec
50 sec
60 sec
70 sec
80 sec