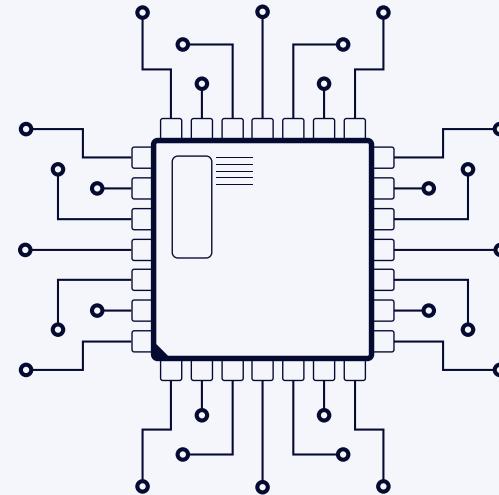


Speed and Energy Optimization of 16-bit Carry Look-Ahead Adder (CLA)

KATYA ALKOBRI

MASA ITMAIZA

RAZAN ODEH



Background & Introduction

We used these gates and modules designed in the project:

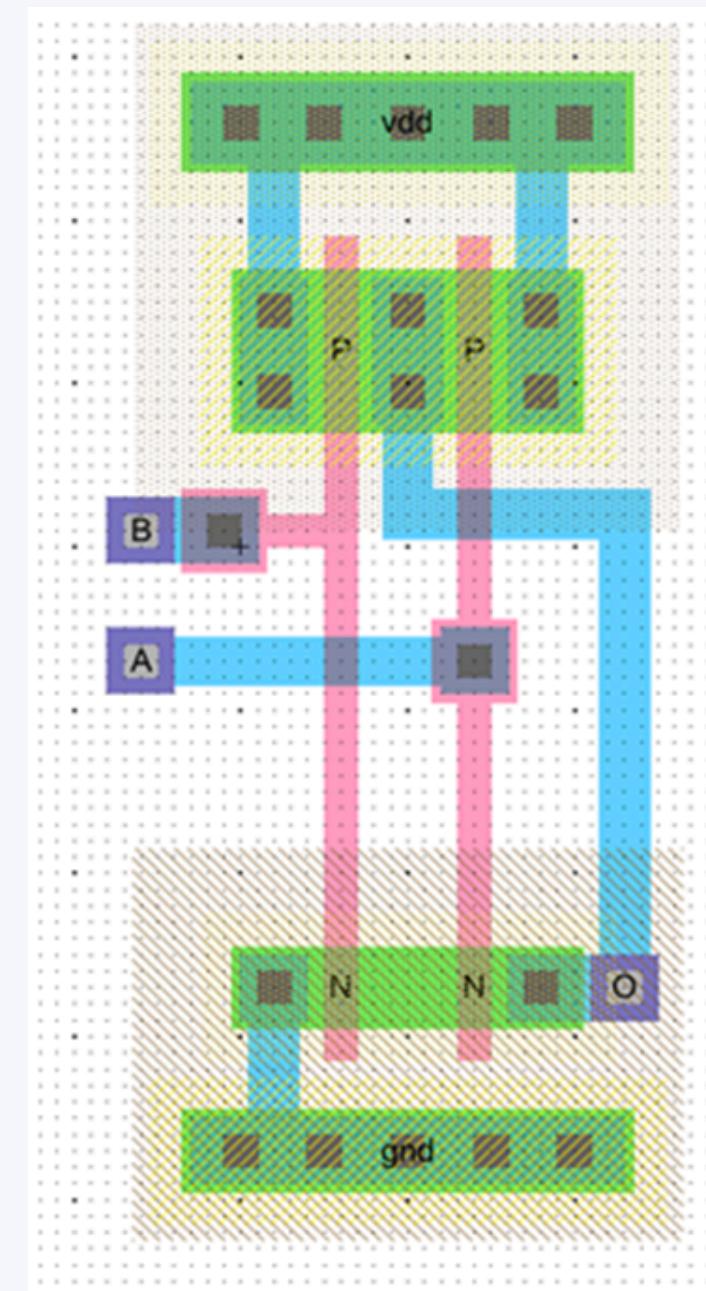
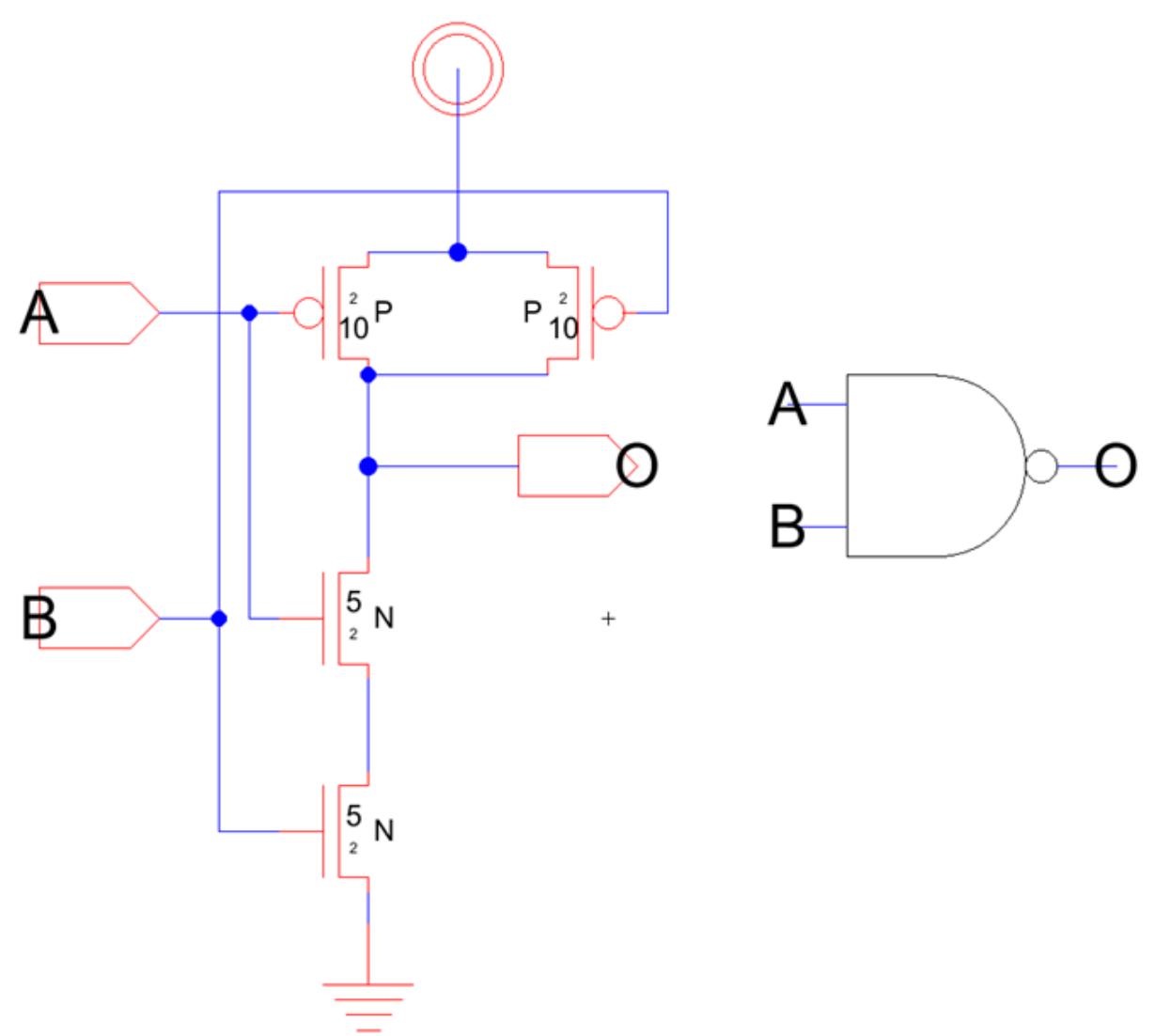
Two-input NAND

CMOS Inverter

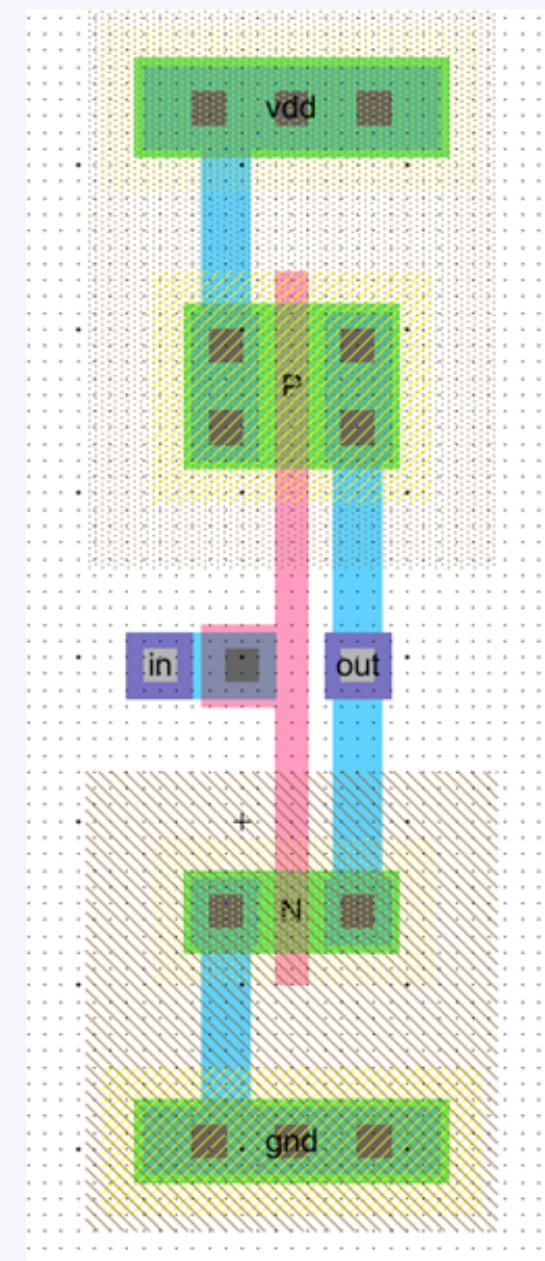
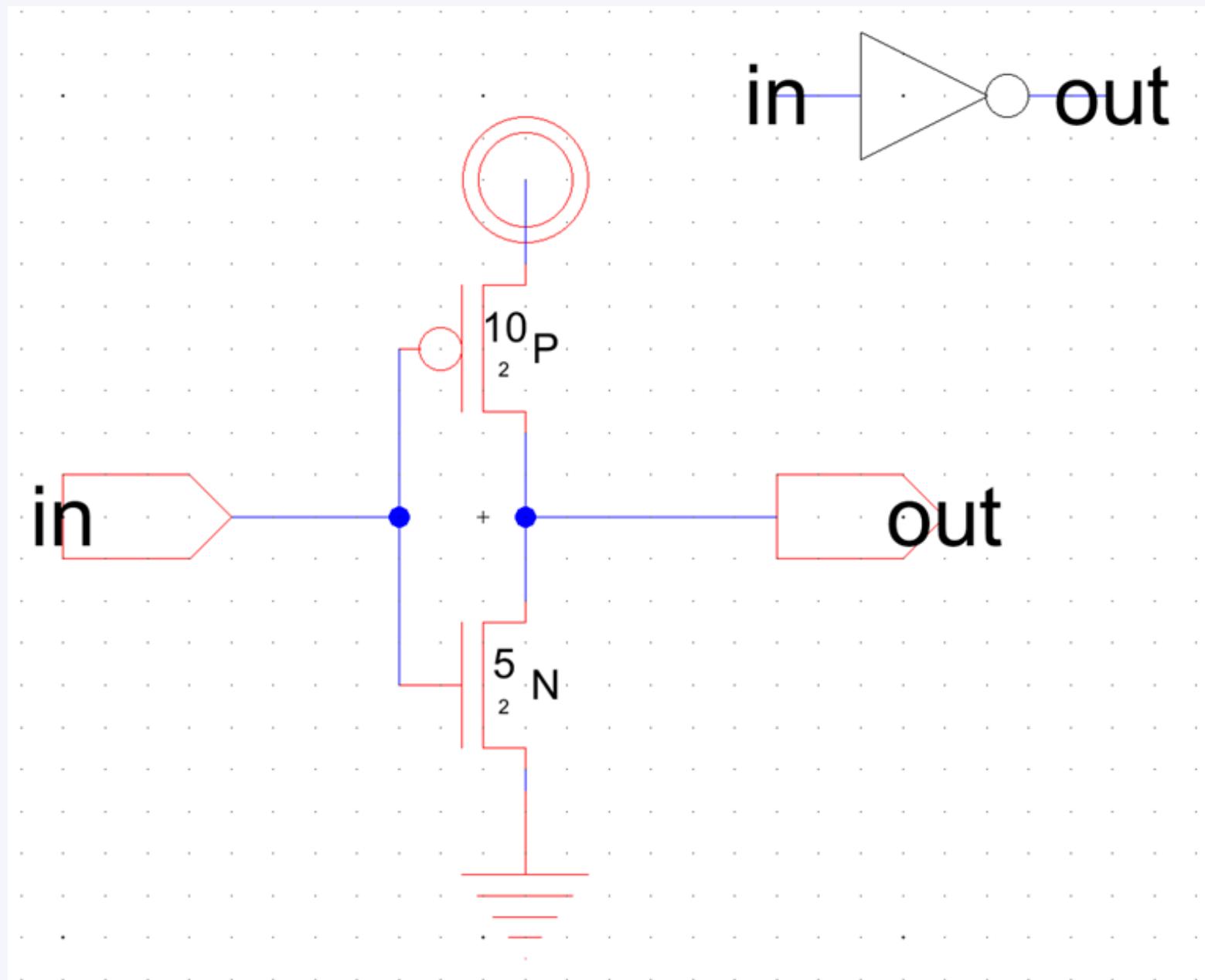
Two-input XOR

Full Adder

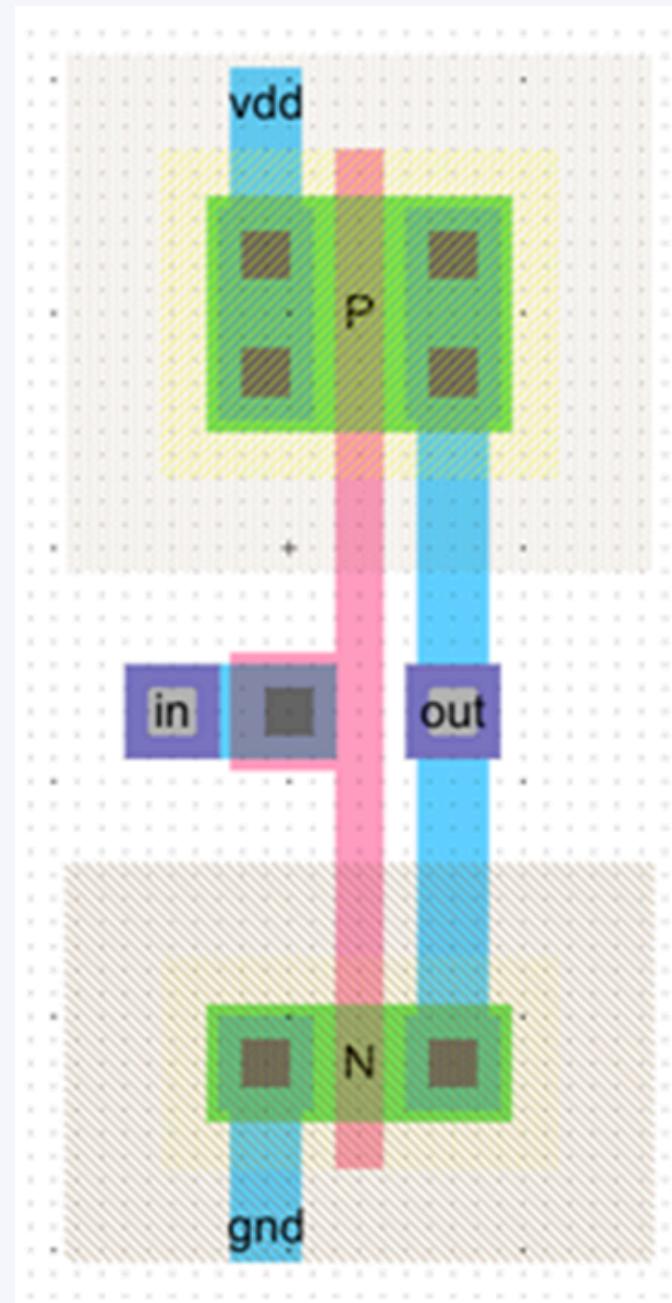
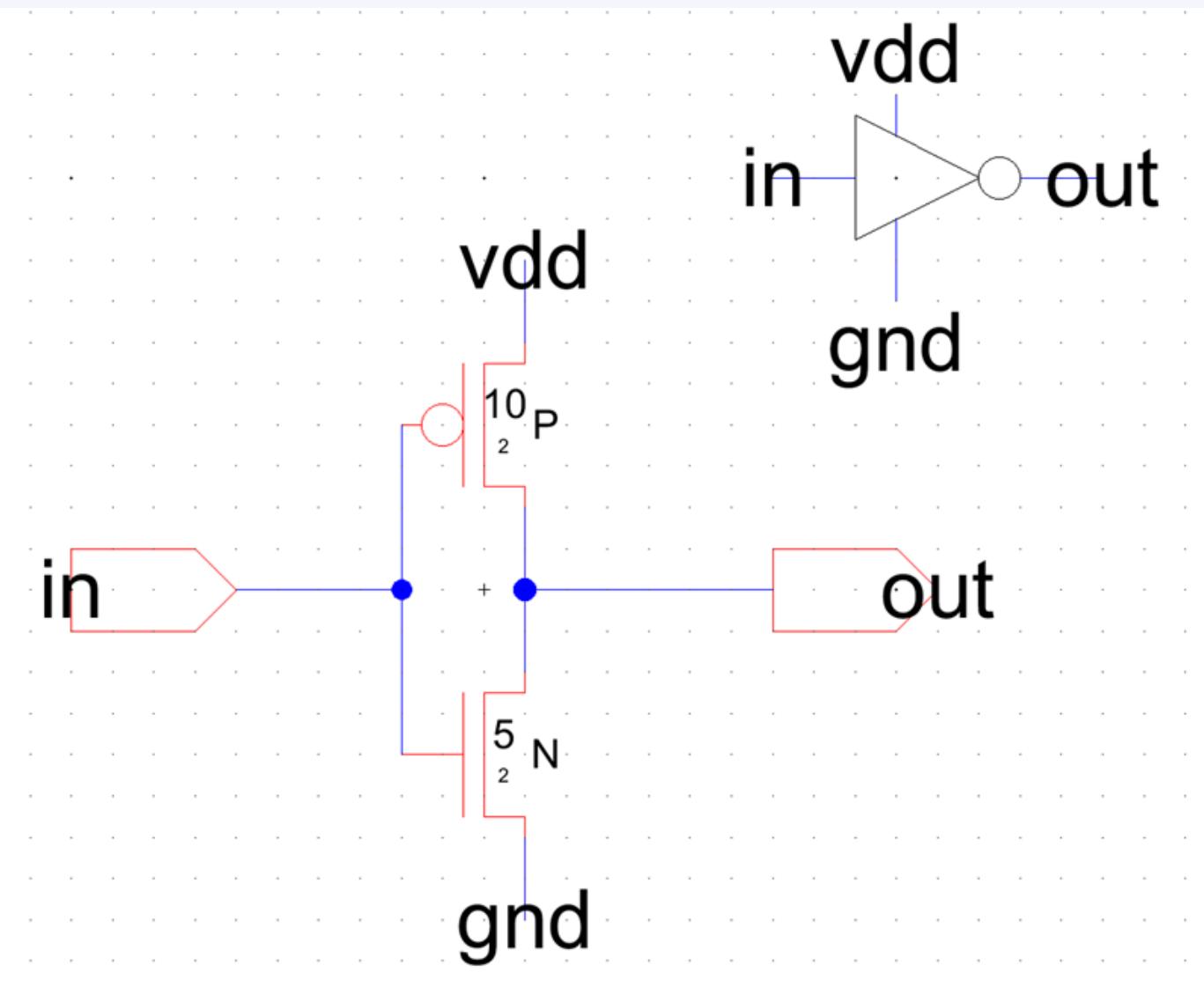
NAND



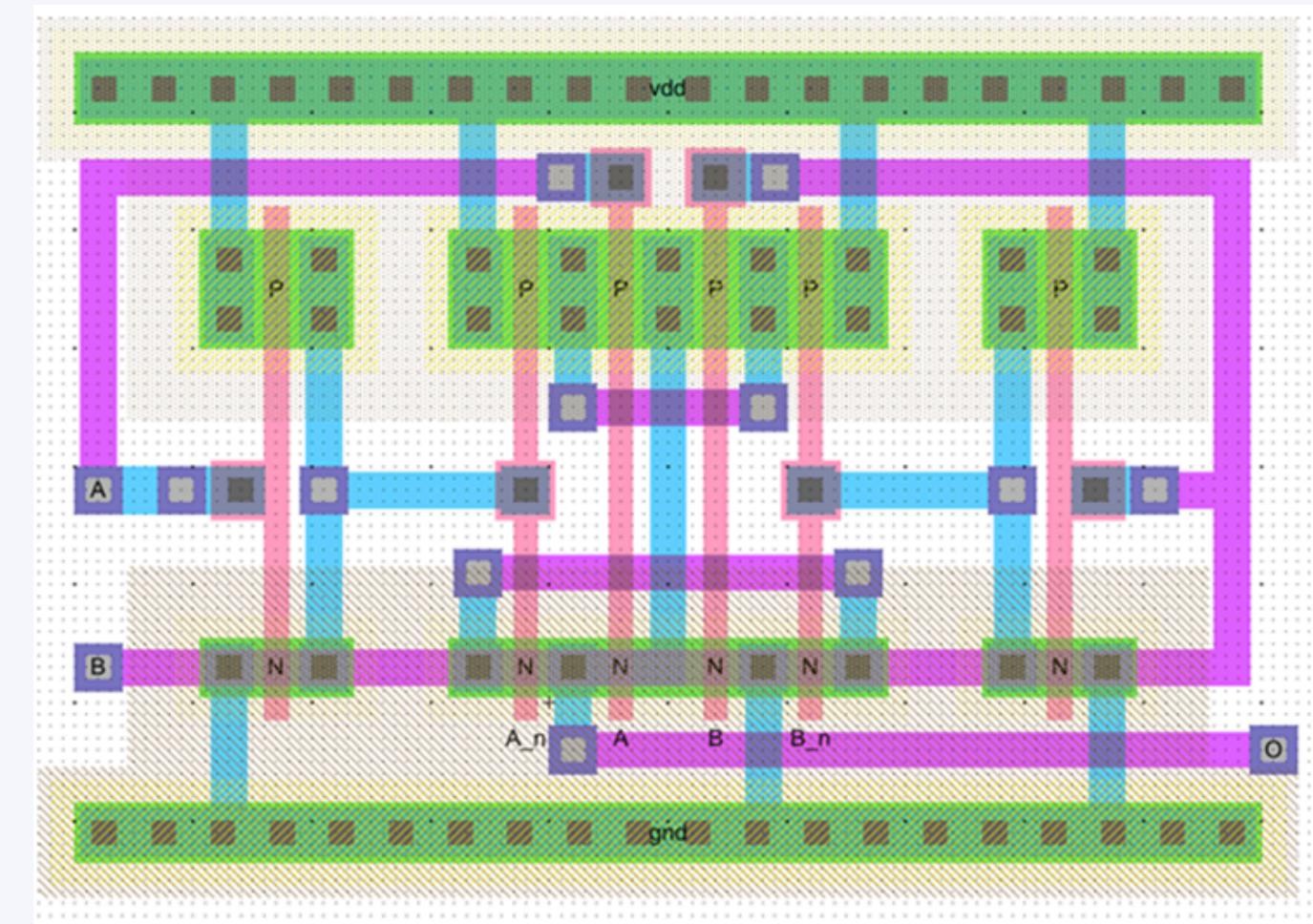
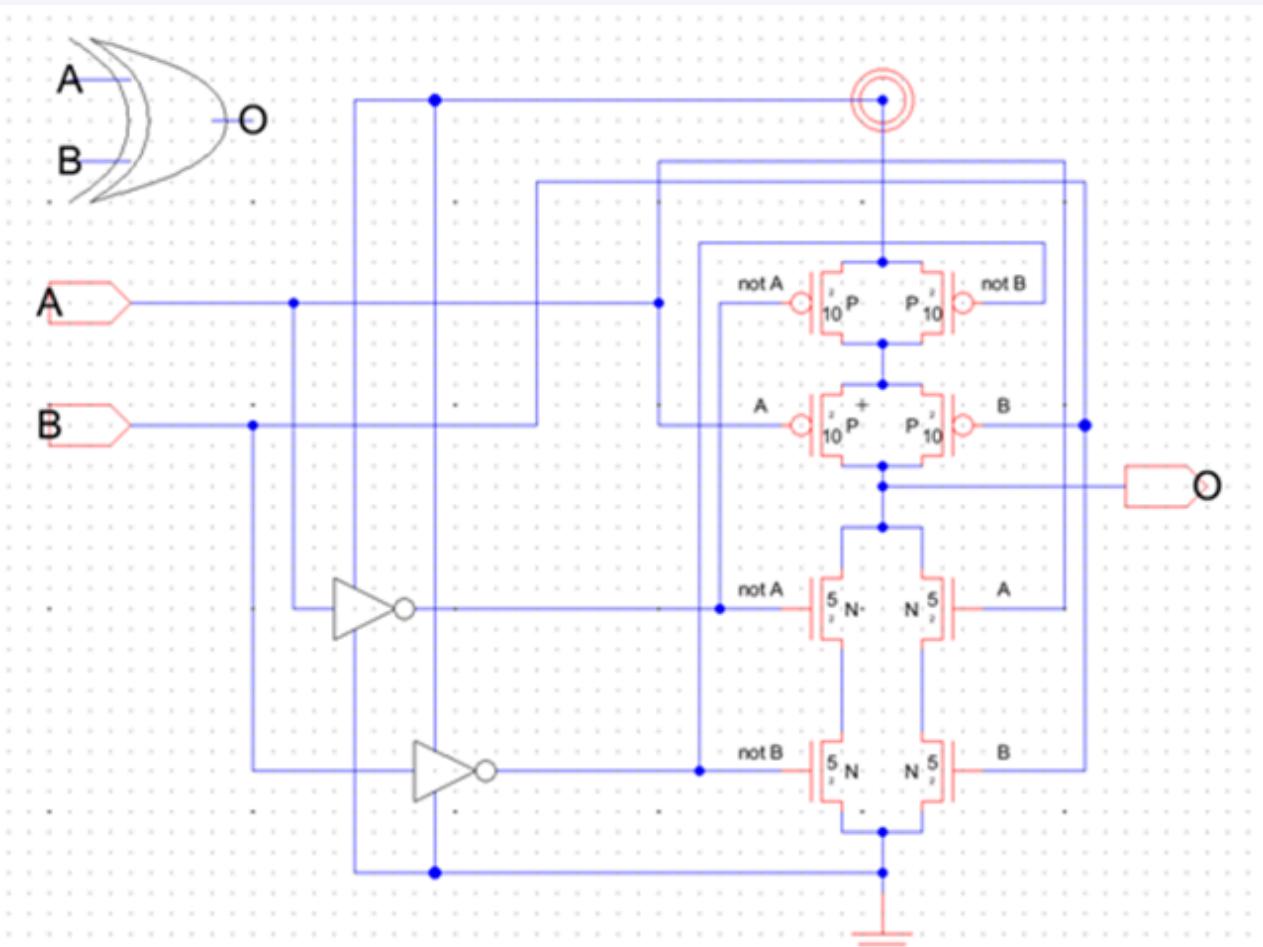
Inverter (with wells)



Inverter (No Wells)



XOR



Background & Introduction

Modified Full Adder:

The carry-out output has been removed, and two new outputs, propagate (P) and generate (G), have been added.

The propagate signal (P_i) is calculated as:

$$P_i = A_i \oplus B_i = \overline{A}_i B_i + A_i \overline{B}_i$$

The generate signal (G_i) is determined using:

$$G_i = A_i \cdot B_i$$

Background & Introduction

LCU Modules – C1, C2, C3

Computes carry-in signal for second, third, and fourth full adder, respectively

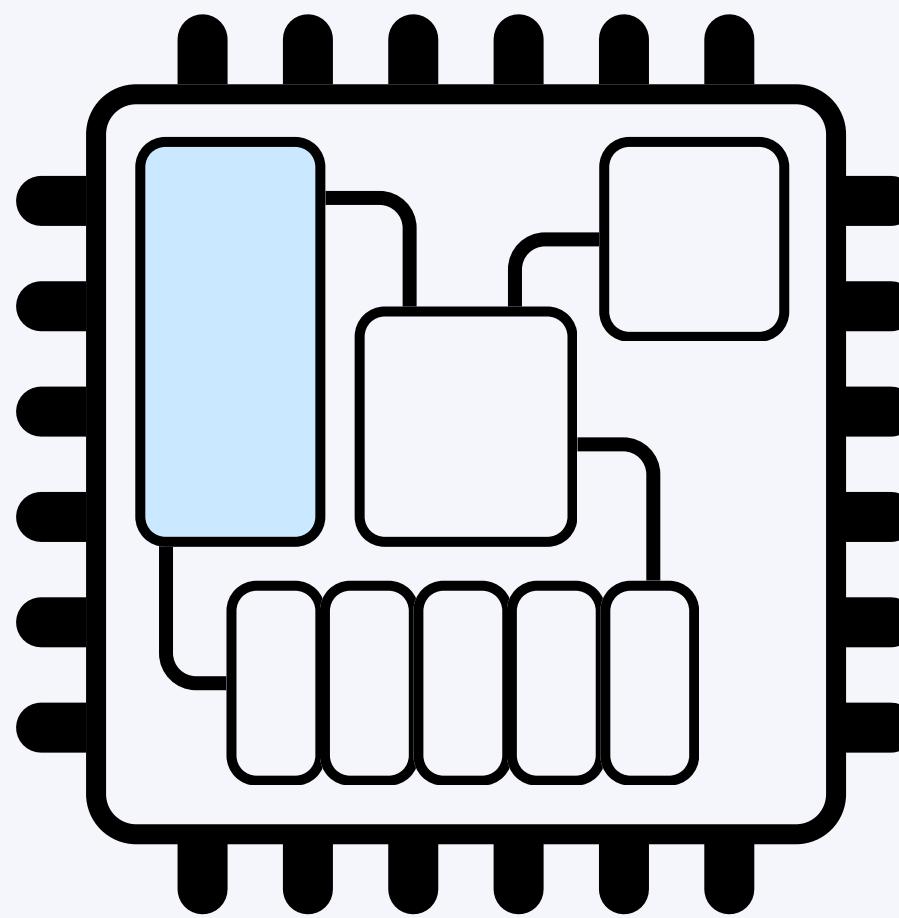
$$C_1 = G_0 + P_0 C_{in}$$

$$C_2 = G_1 + P_1 G_0 + P_1 P_0 C_{in}$$

$$C_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_{in}$$

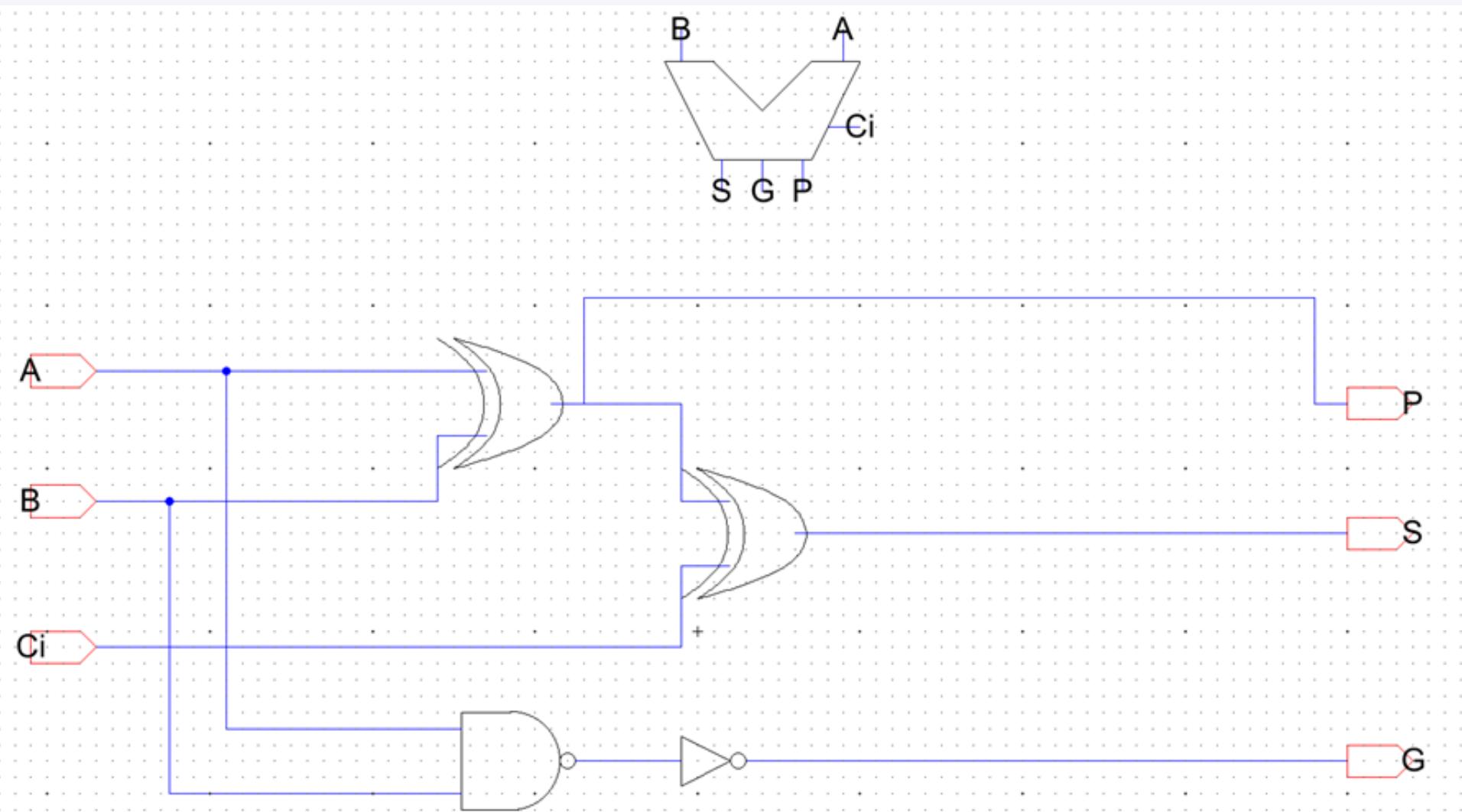
Background & Introduction

Using four full adders modules, we can create a scalable four-bit CLA adder, construct a 16-bit CLA.



Electric Schematic

Modified Full Adder

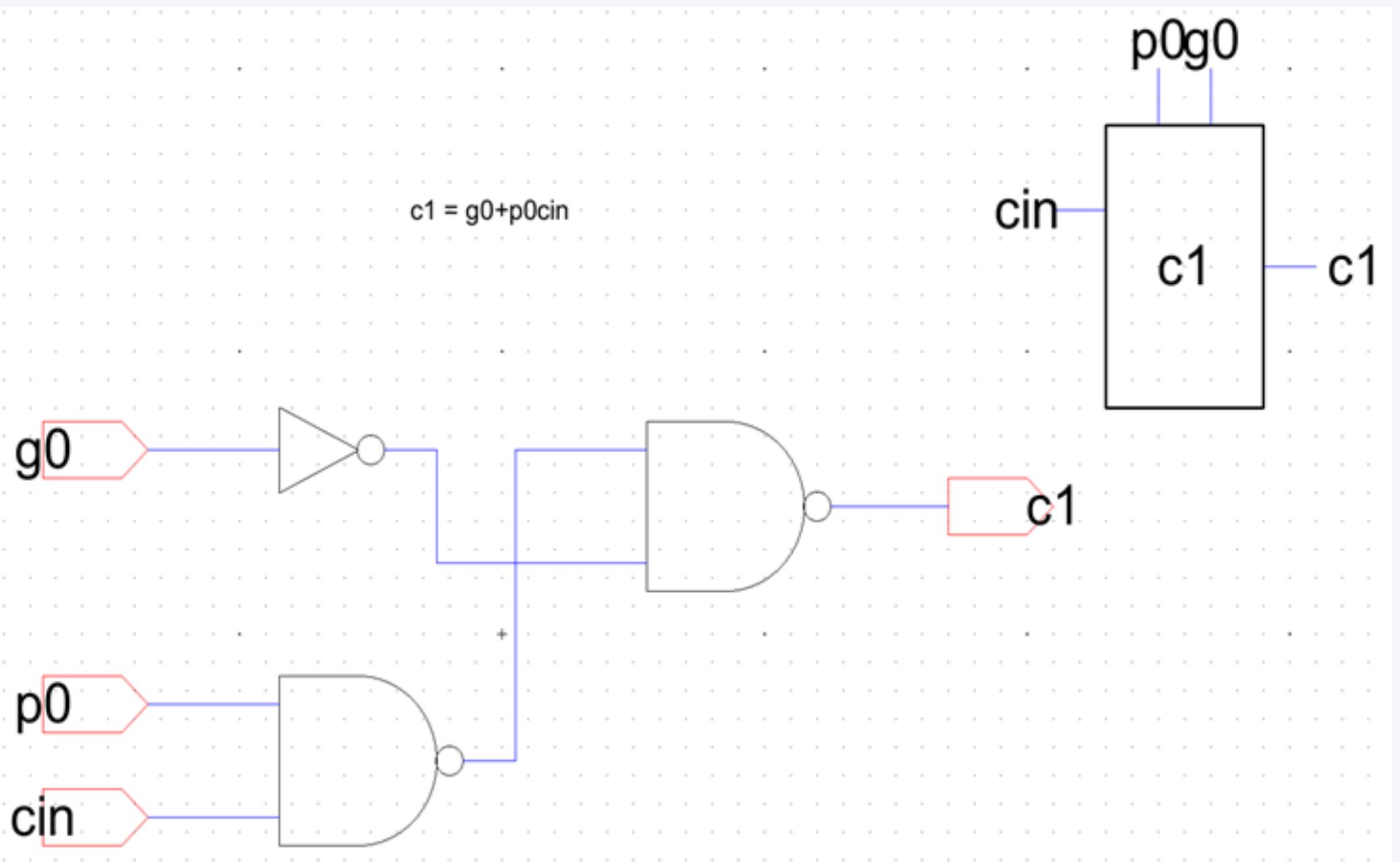


$$S = A \oplus B \oplus C_i$$

$$P = A \oplus B$$

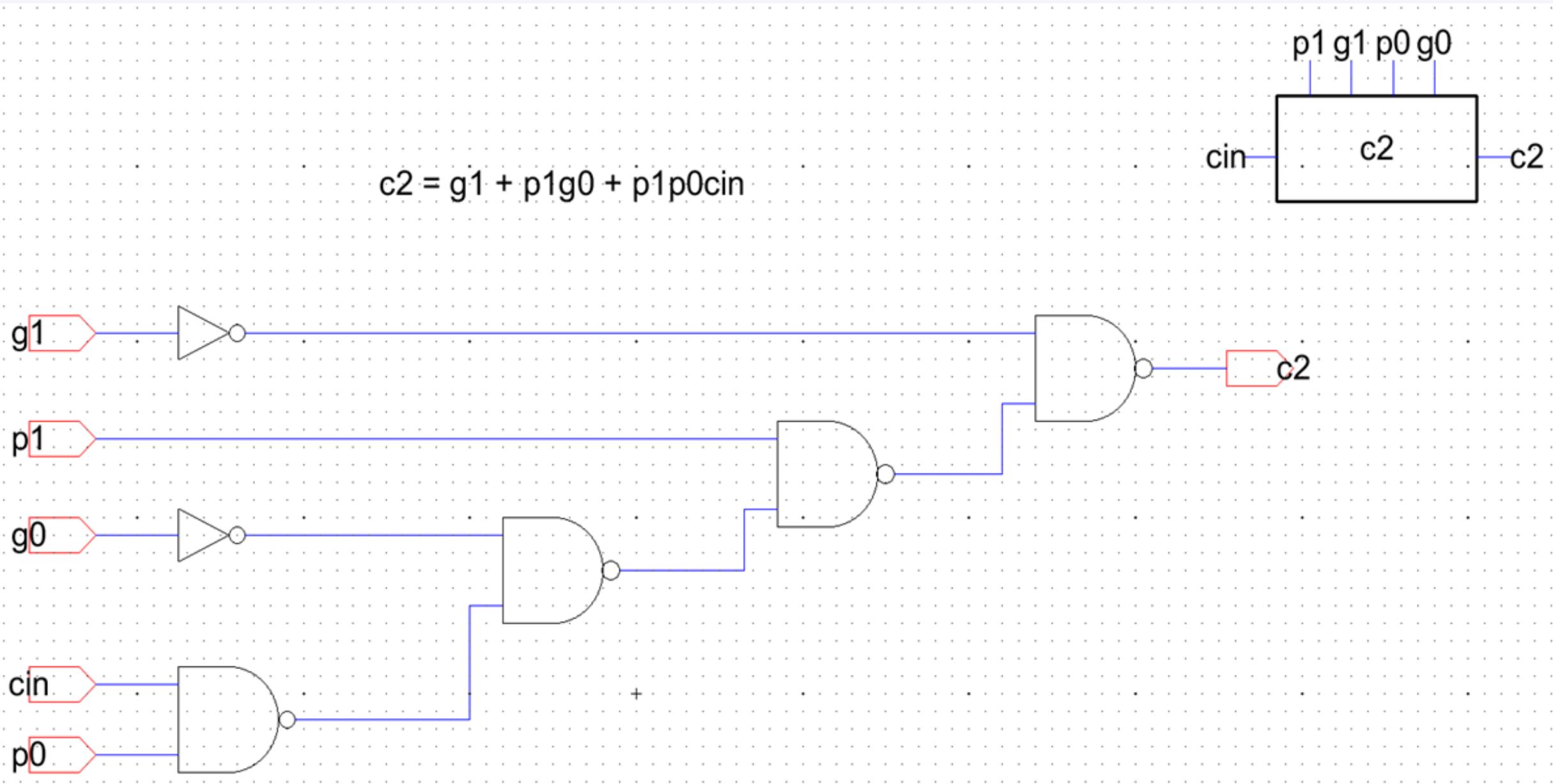
$$G = A \cdot B$$

C1 Module



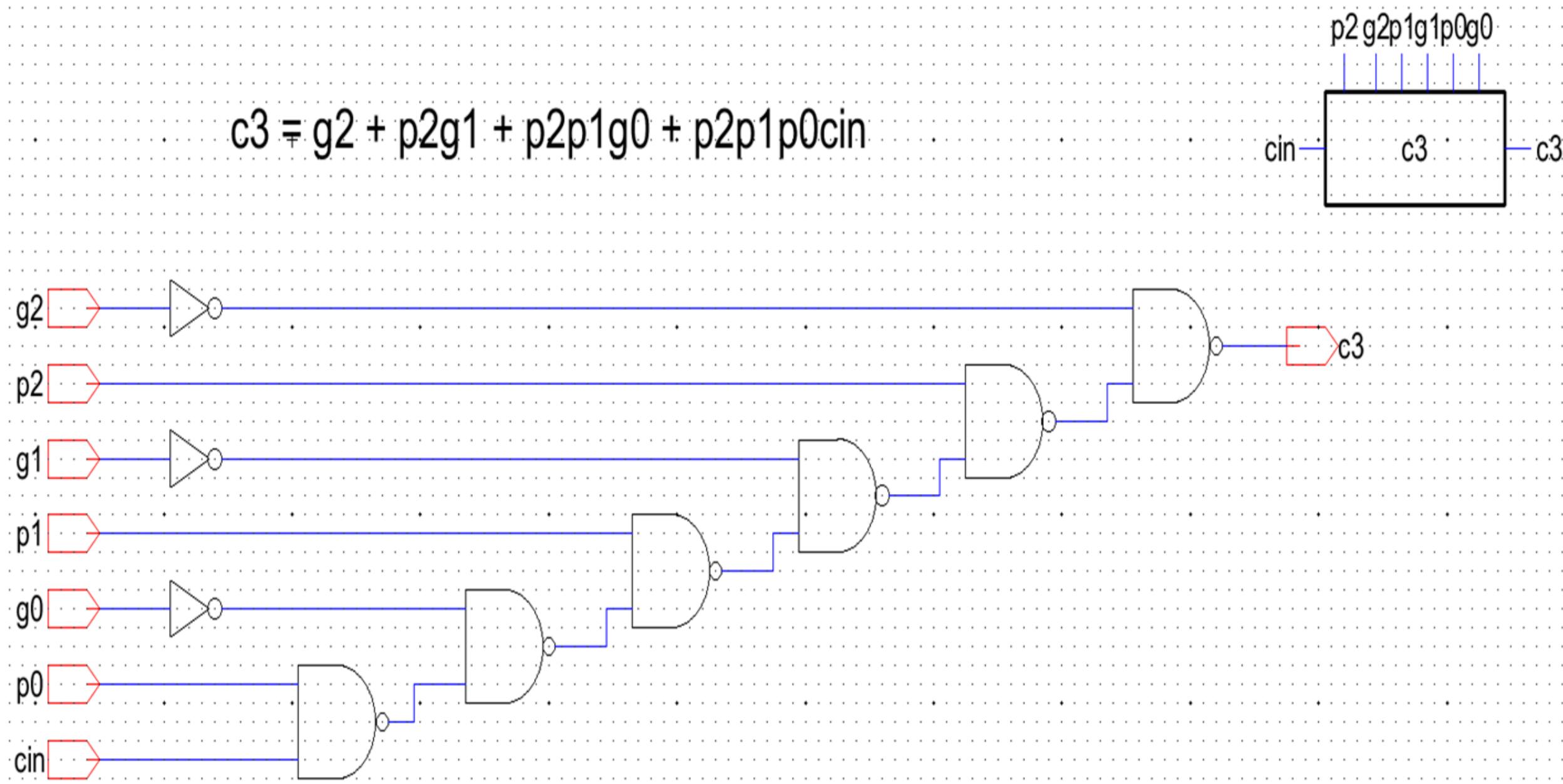
C2 Module

$$c_2 = g_1 + p_1g_0 + p_1p_0c_{in}$$

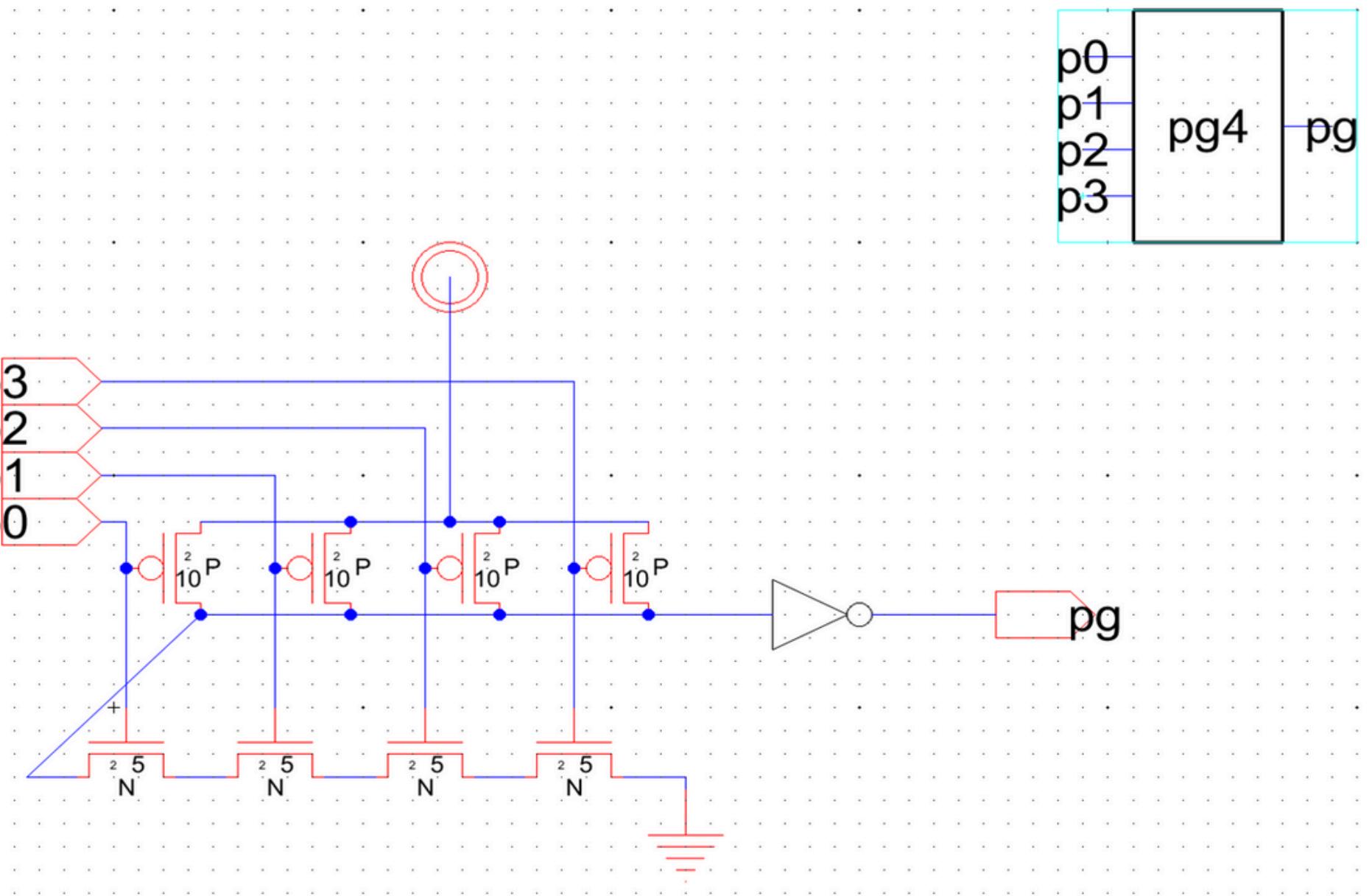


C3 Module

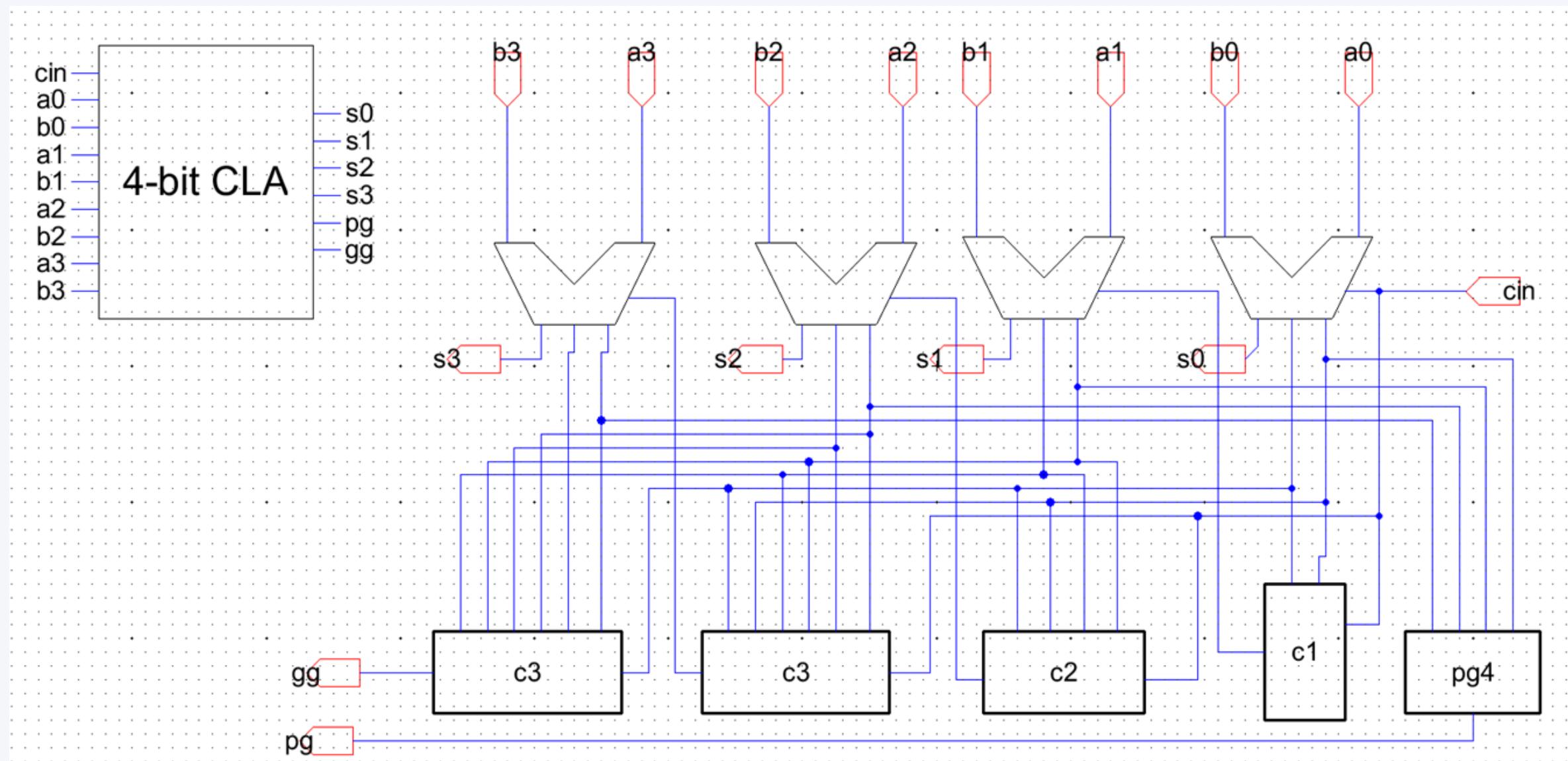
$$c3 = g2 + p2g1 + p2p1g0 + p2p1p0cin$$



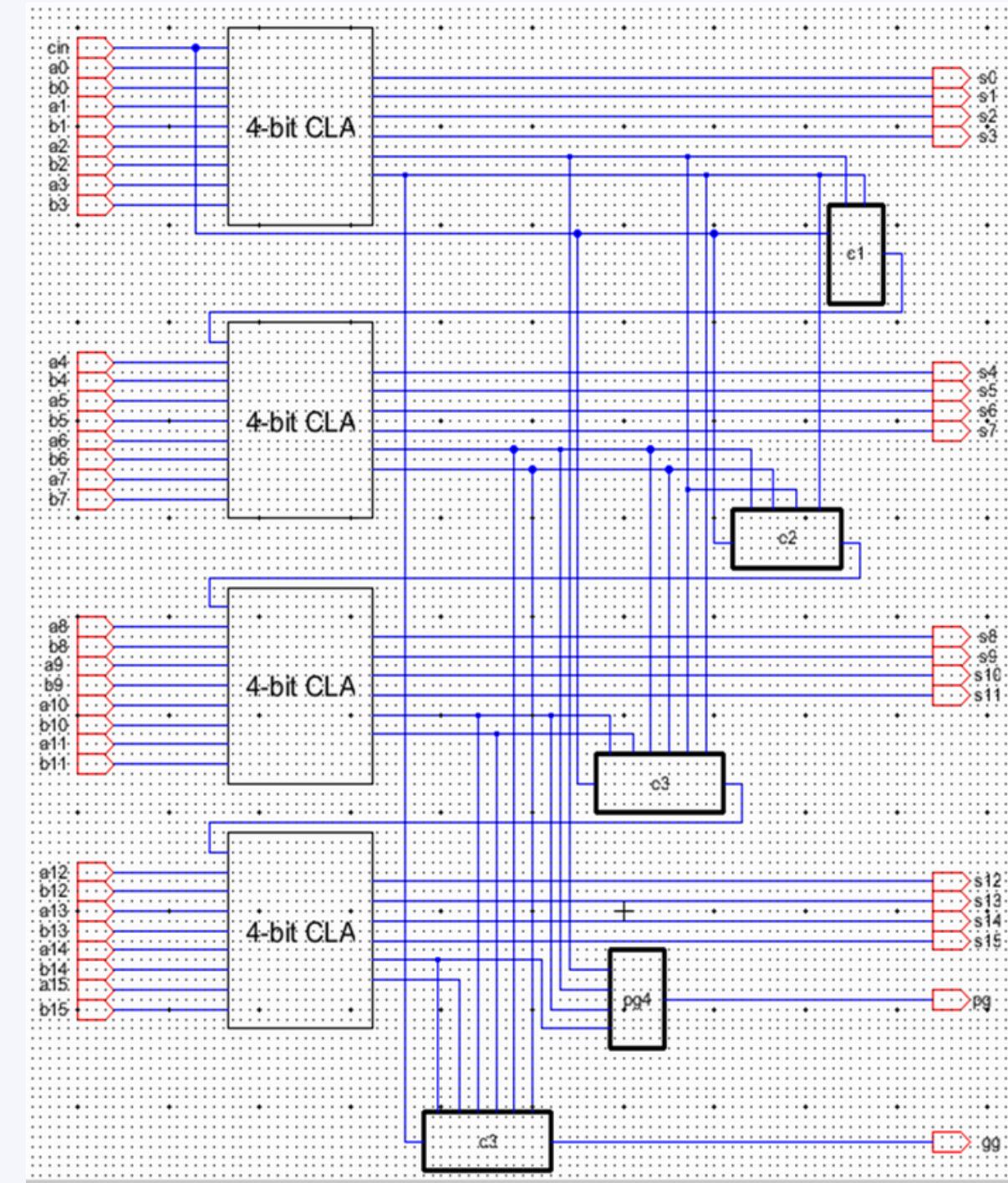
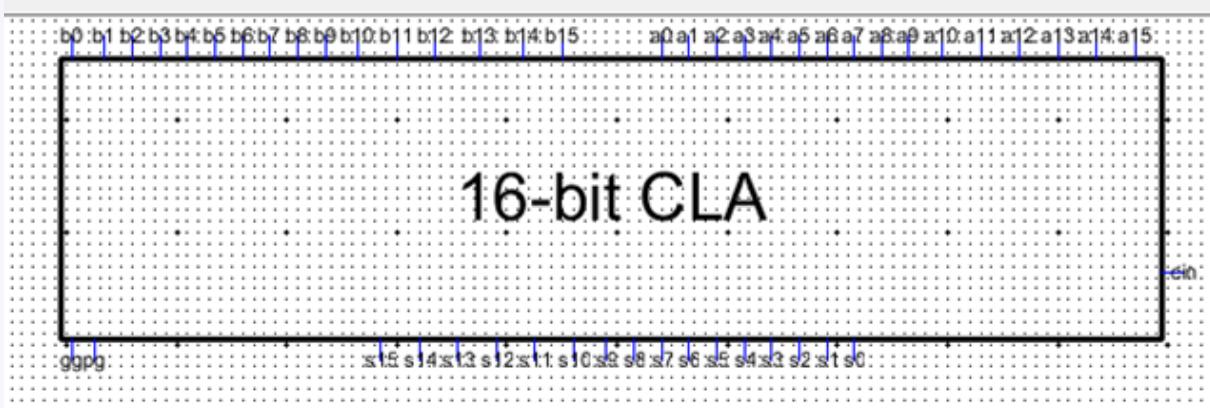
PG4 Module



4-bit CLA

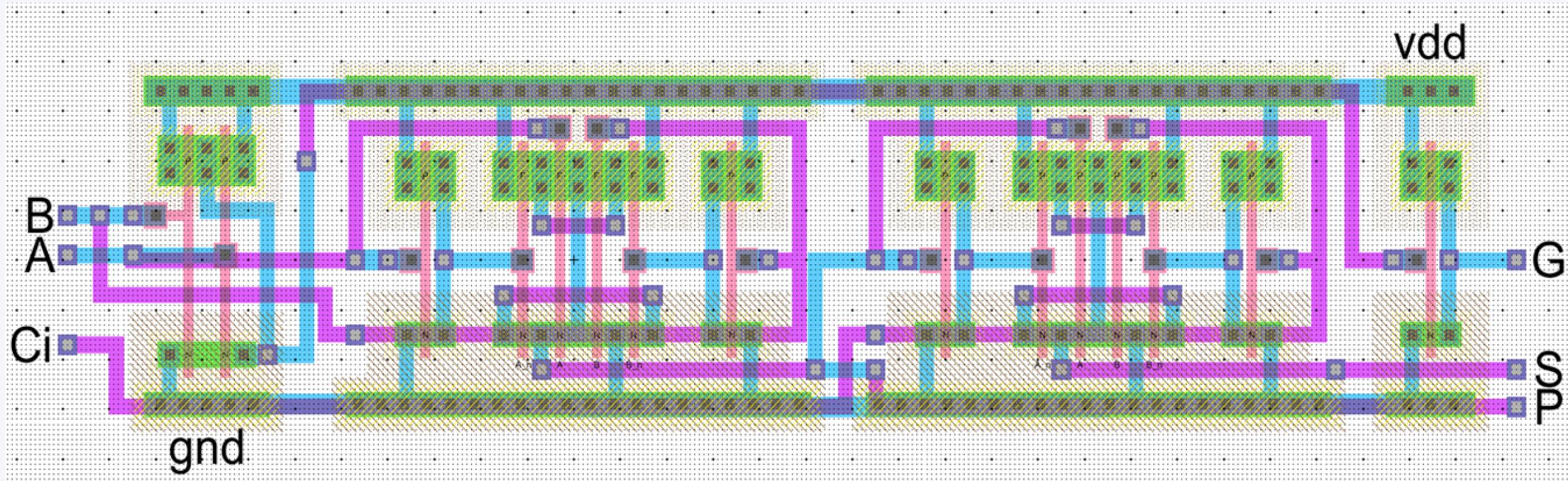


16-bit CLA

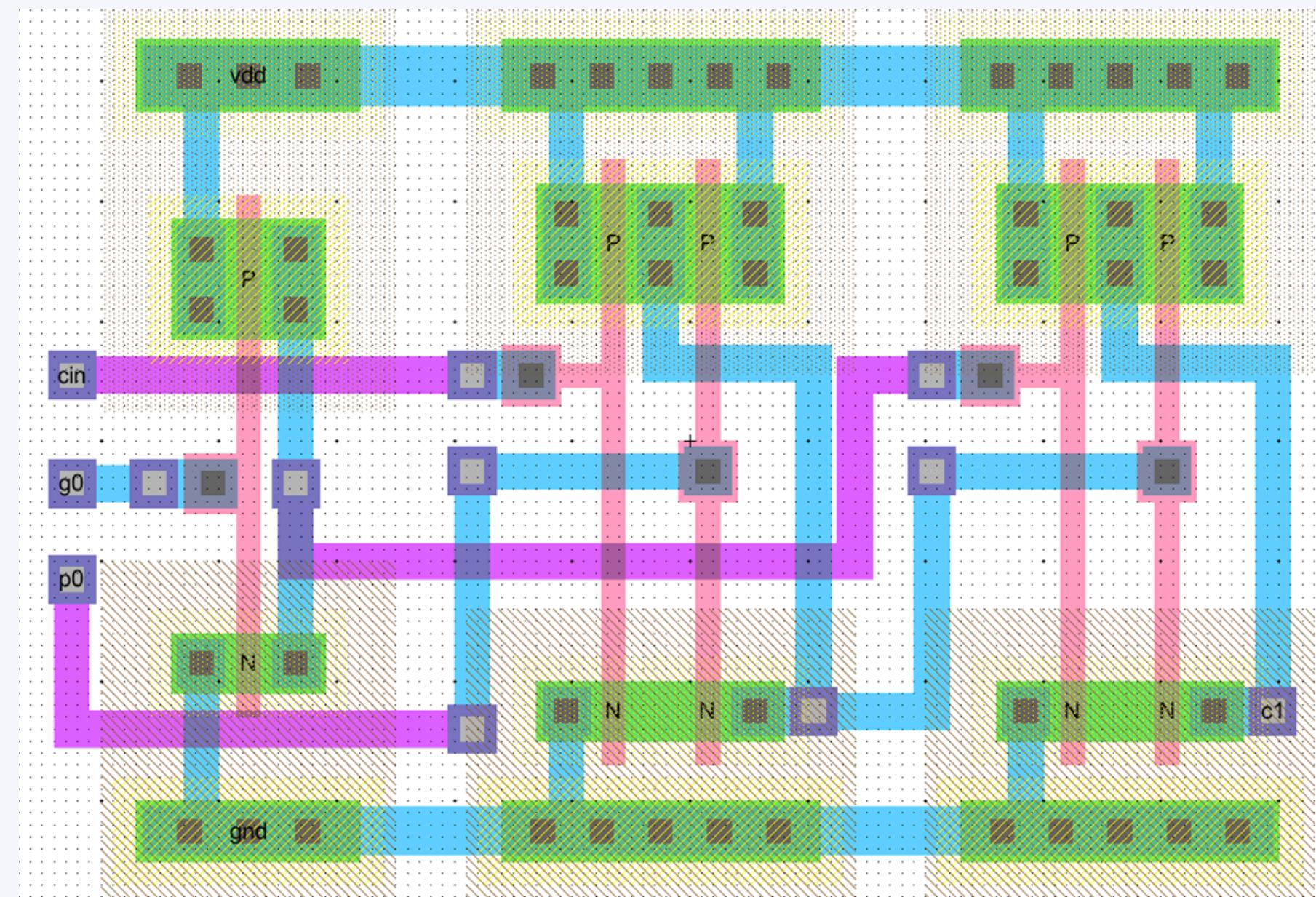


Electric Layout

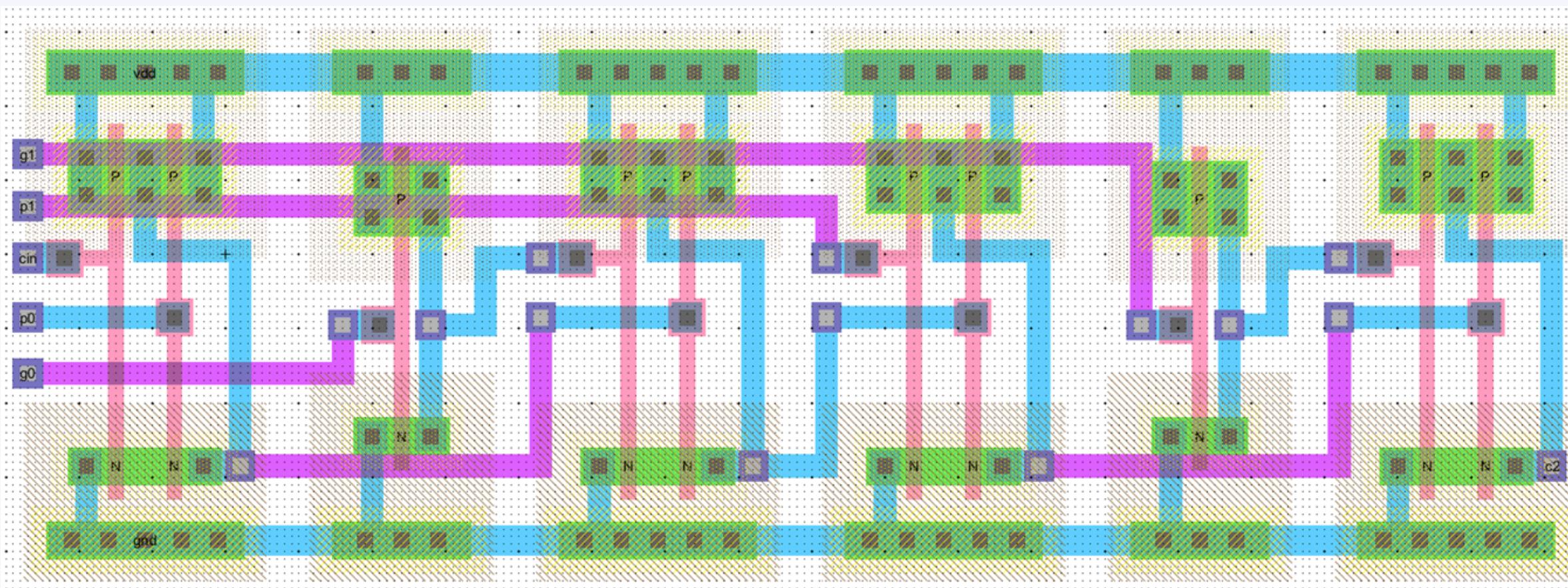
Modified Full Adder



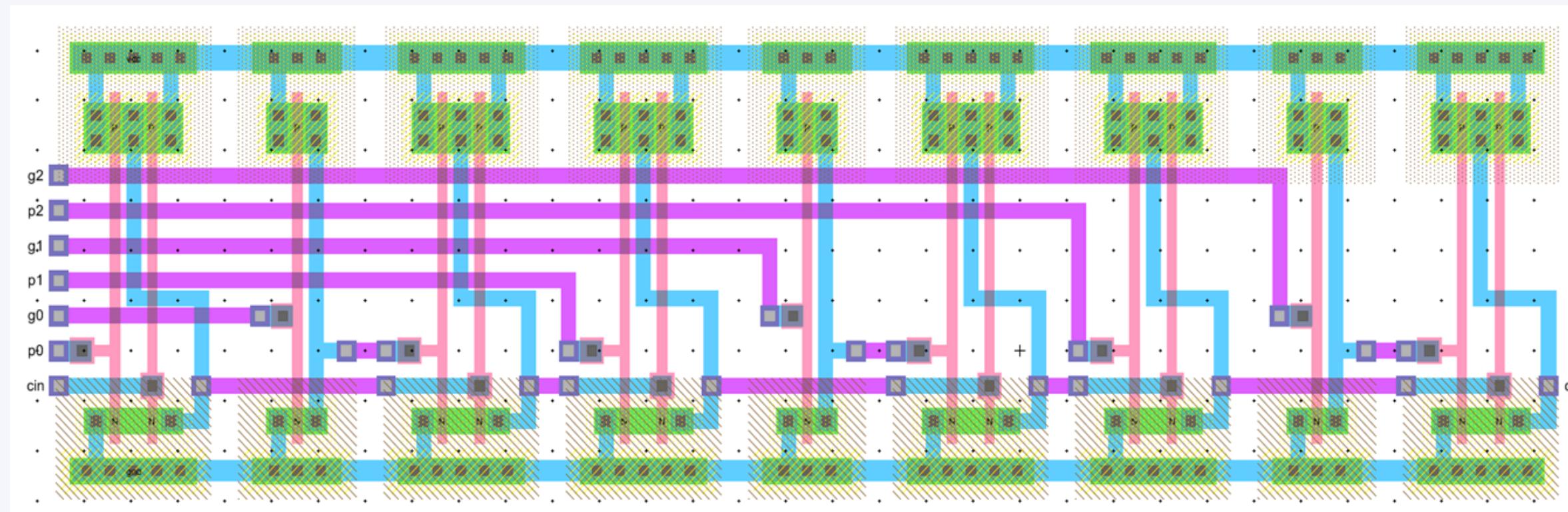
C1 Module



C2 Module

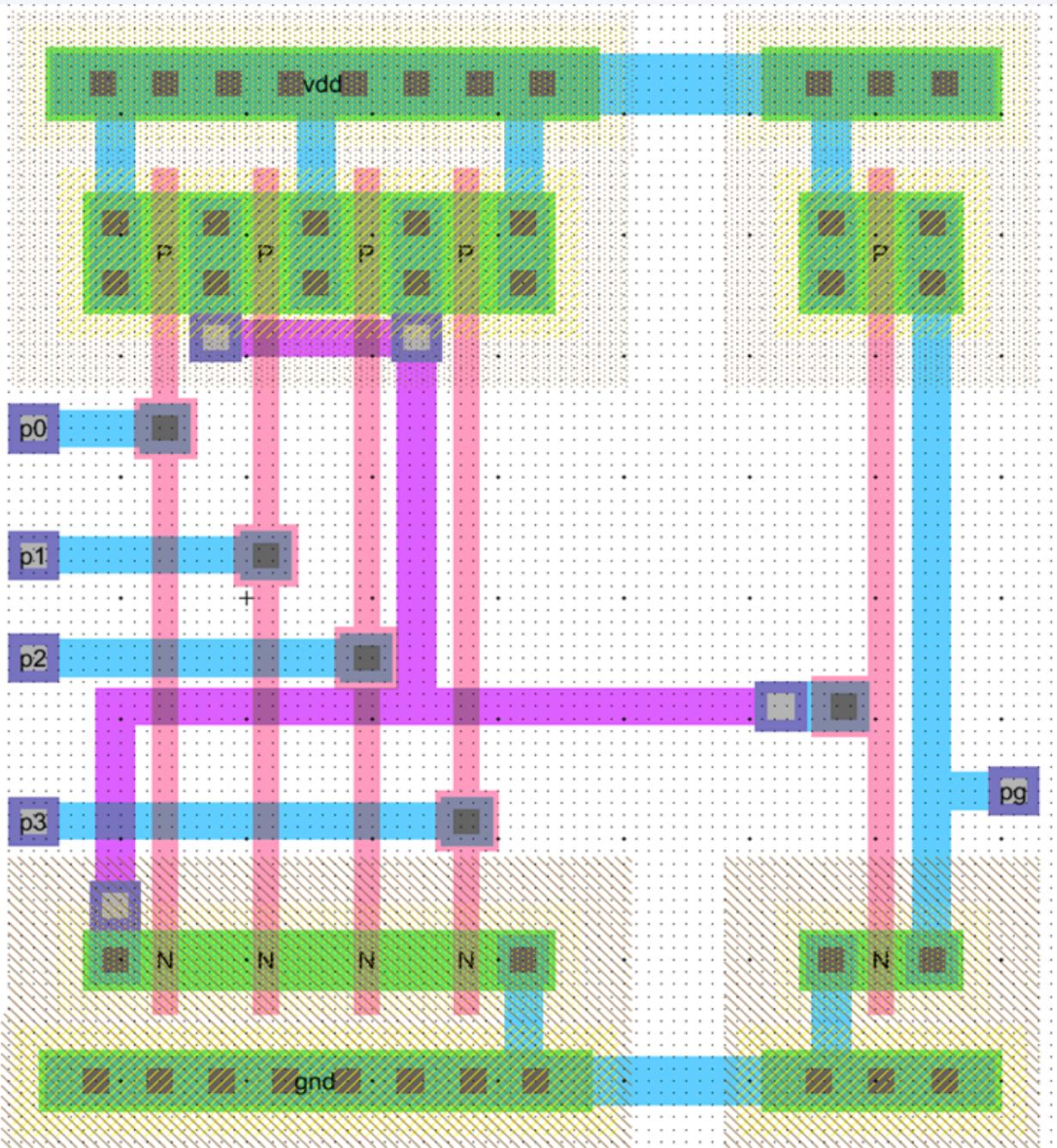


C3 Module

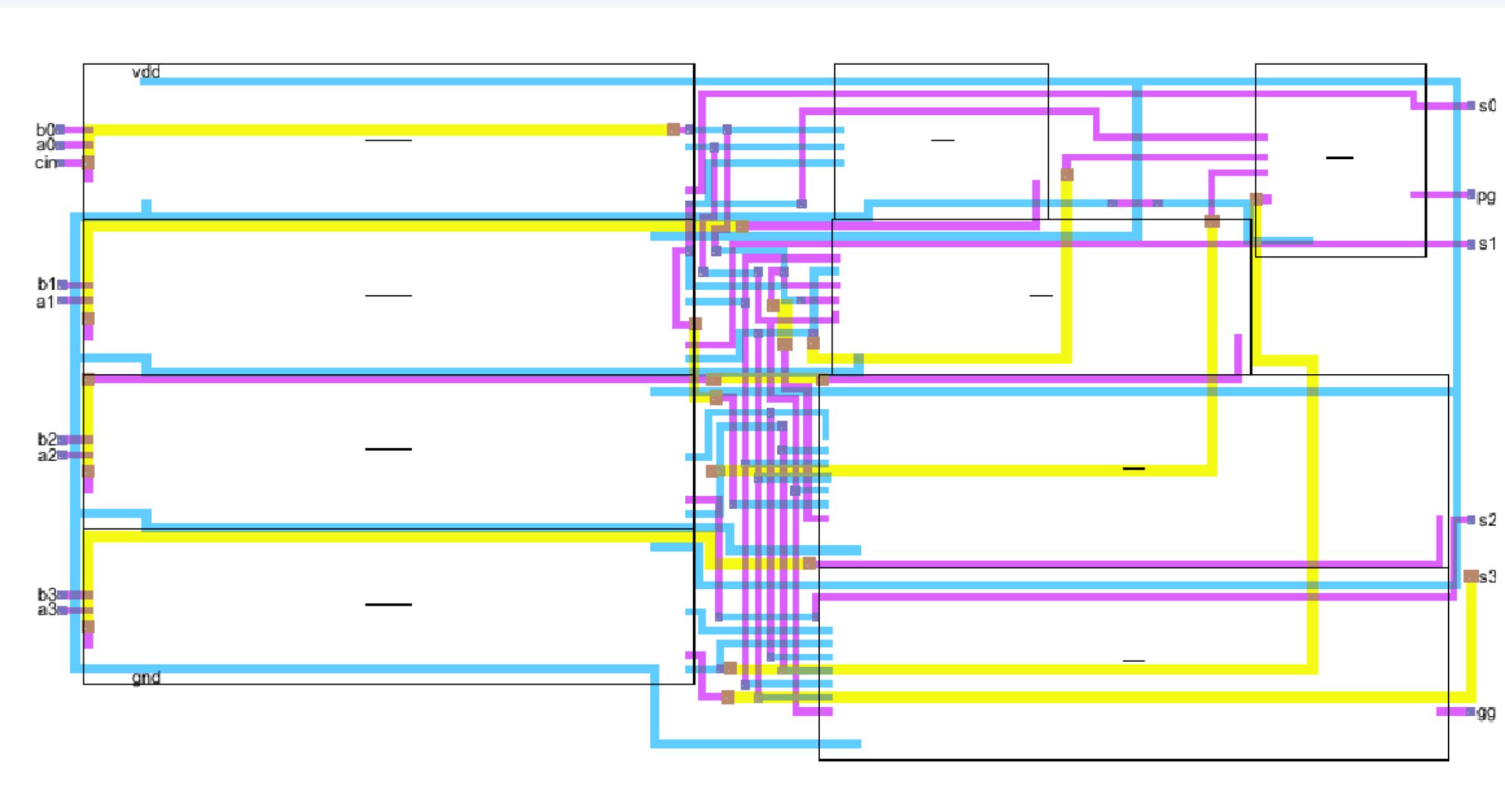


NAND and Inverter layouts used here are slightly taller than in the C1 and C2 modules
Also in the PG4 module, next slide

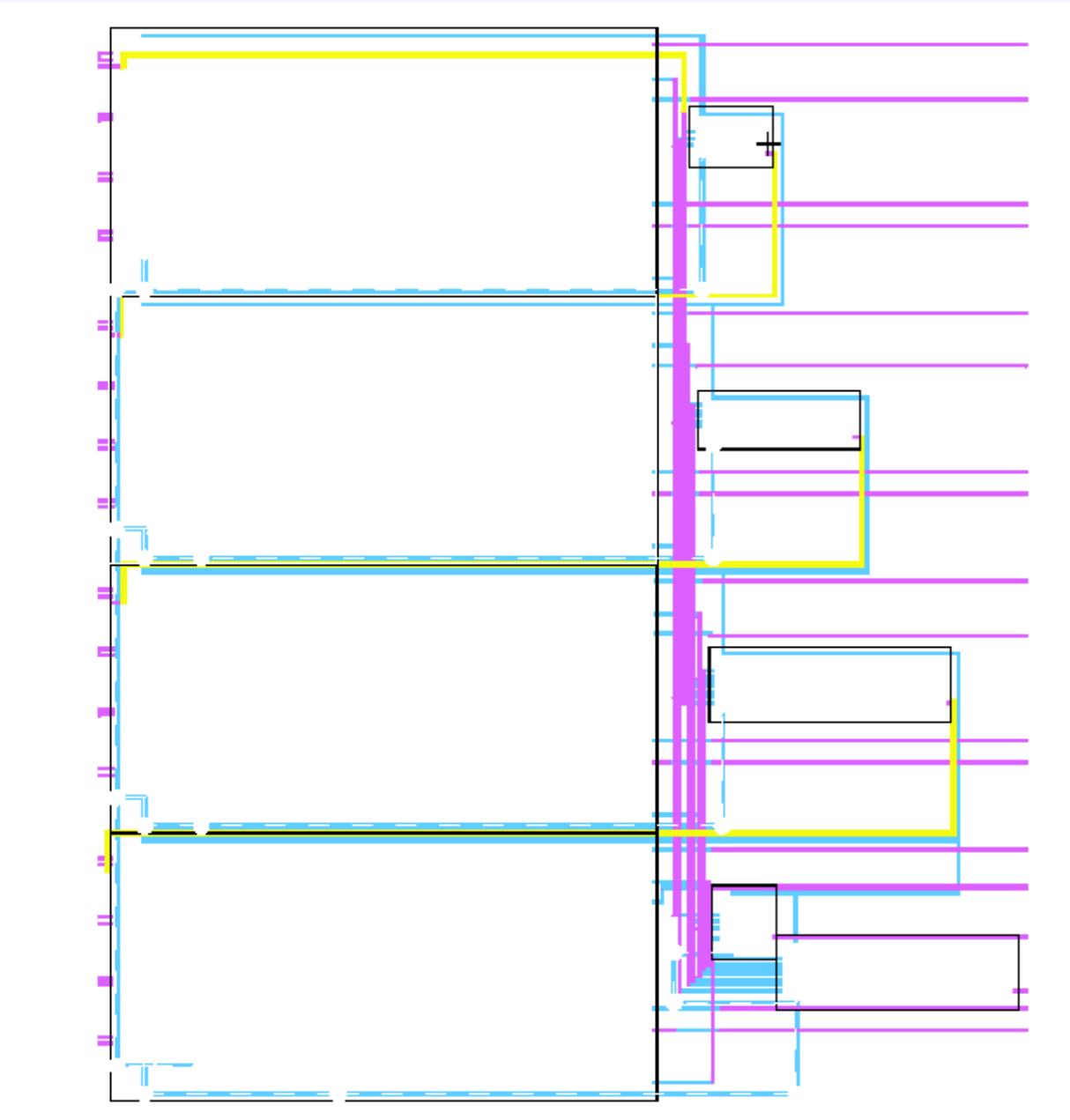
PG4 Module



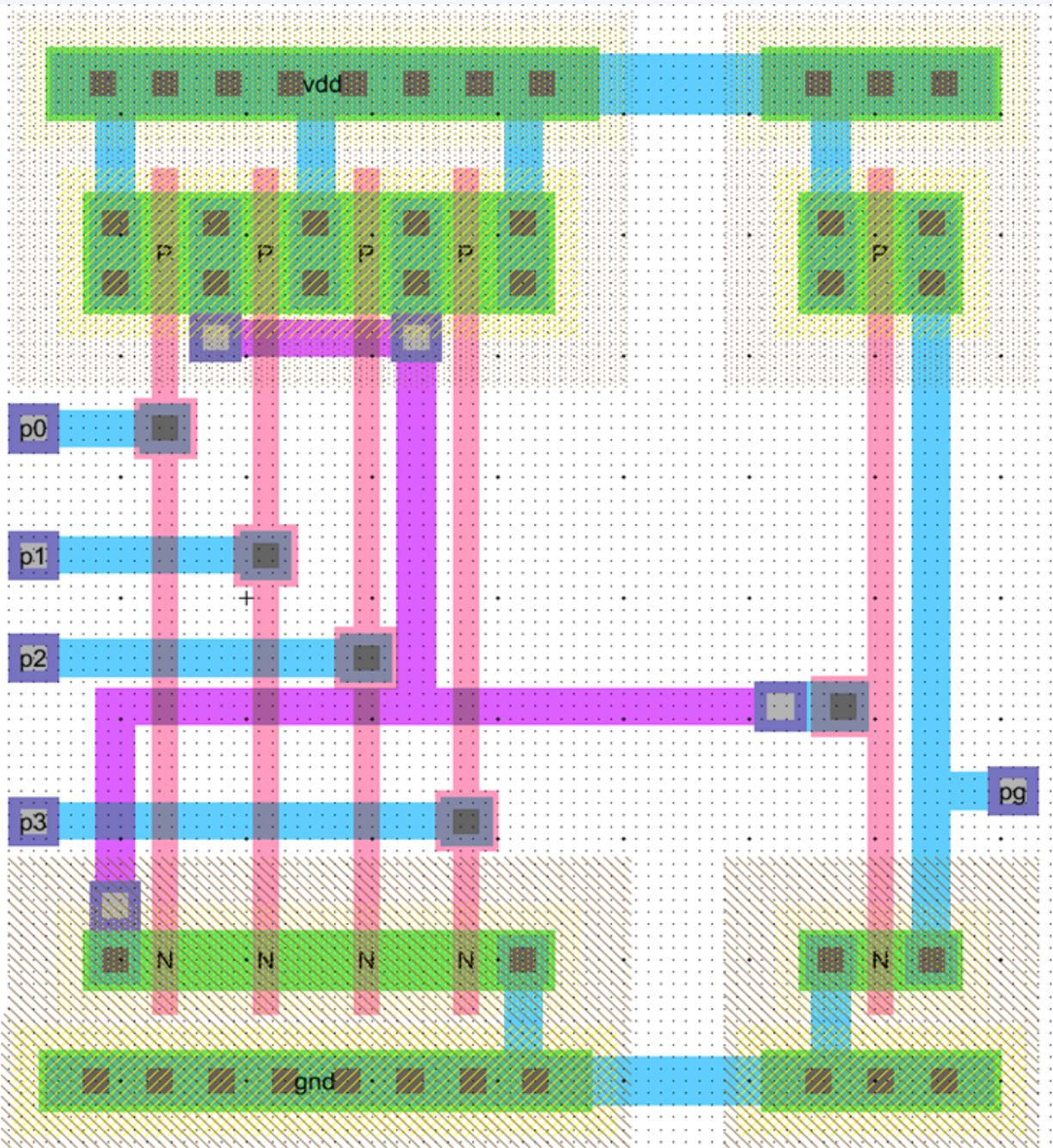
4-bit CLA



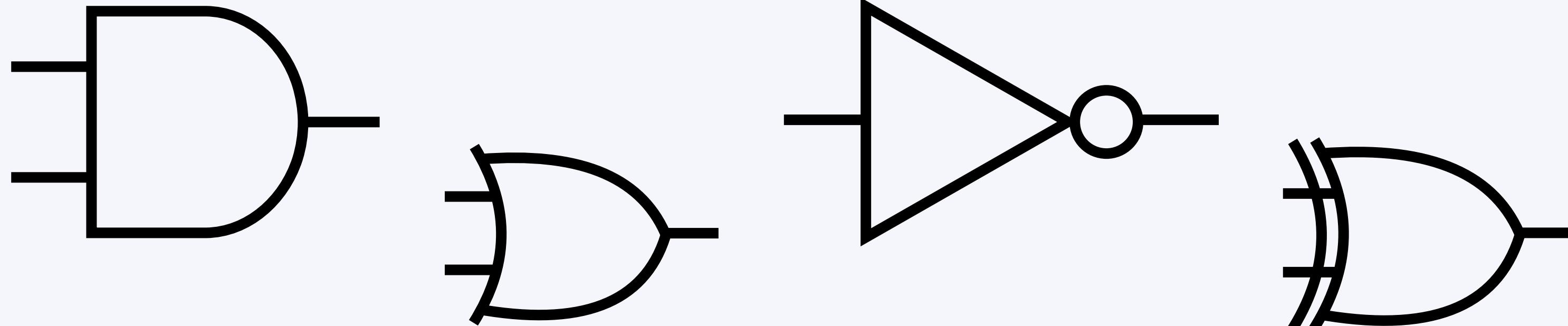
16-bit CLA



PG4 Module



Boolean equations & Truth Tables



1-bit CLA test bench output.

Time	Cin	G0
0	0	0
10	1	0
20	0	1
30	1	1
40	0	0

2-Bit CLA testbench output.

Time	Cin	G0	G1	P0	P1	C2
0	0	0	0	0	0	0
10	1	0	0	0	0	0
20	0	1	0	0	0	0
30	0	0	1	0	0	1
40	1	0	0	1	0	0
50	0	0	1	0	1	1
60	1	1	1	1	1	1

3-Bit Carry Look-Ahead Adder(CLA).

The 3-bit Carry Look-Ahead Adder (CLA) computes the carry-out (c_3) based on the equation:

$$c_3 = g_2 + p_2 * g_1 + p_2 * p_1 * g_0 + p_2 * p_1 * p_0 * c_{in}$$

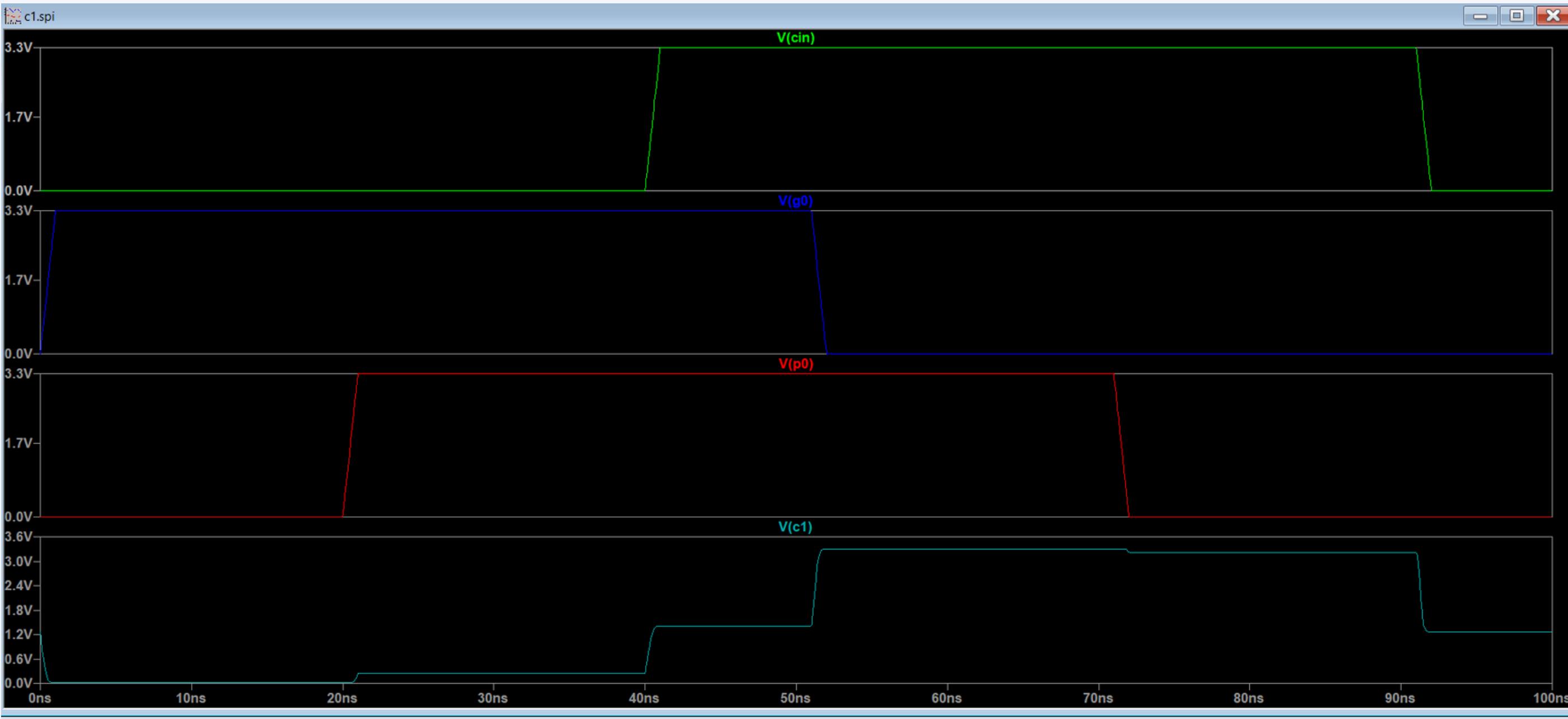
4-Bit CLA testbench output.

16-Bit CLA testbench output.

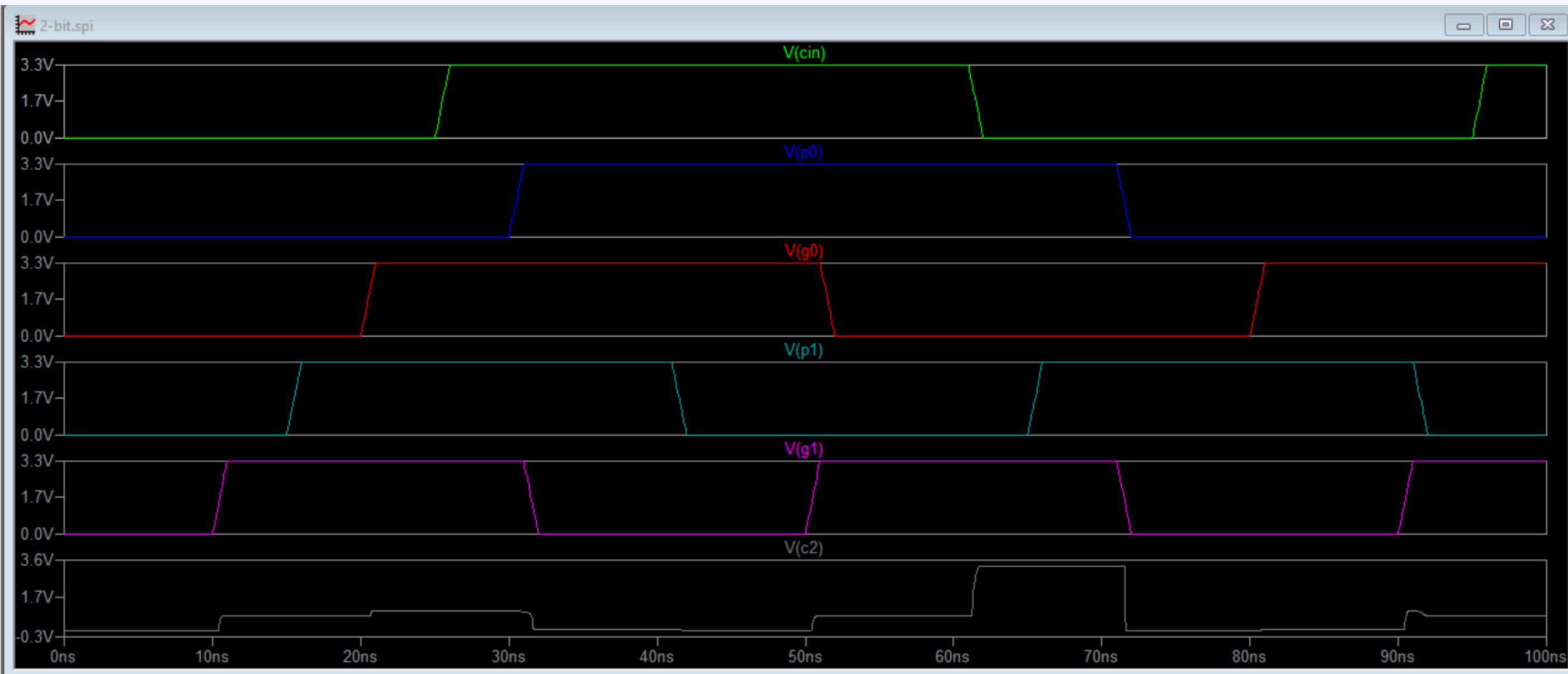
Time (ns)	A	B	Cin	Sum	GG	PG
0	0000000000000000	0000000000000000	0	0000000000000000	0	0
10000	0000000000000001	0000000000000001	0	00000000000000010	0	0
20000	1111111111111111	0000000000000000	1	0000000000000000	1	0
30000	1111111111111111	1111111111111111	1	1111111111111110	1	0
40000	1000000000000000	0111111111111111	1	0000000000000000	0	1
50000	1010101010101010	0101010101010101	1	1111111111111111	0	1
60000	0010100100101010	0101010101010101	1	0110100101011011	0	1
70000	0000000000000000	0000000000000000	1	0000000000000001	0	0

Tspice output

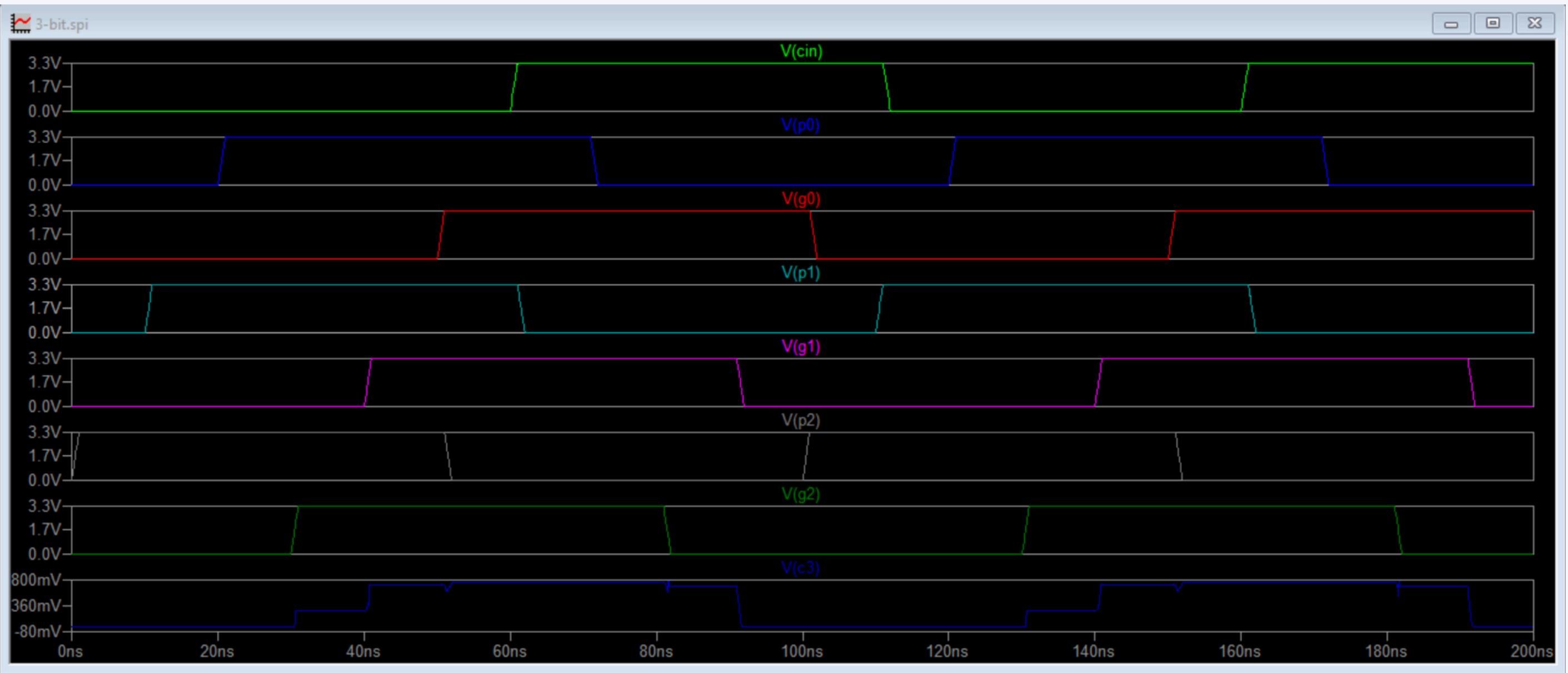
1-Bit CLA LTspice output



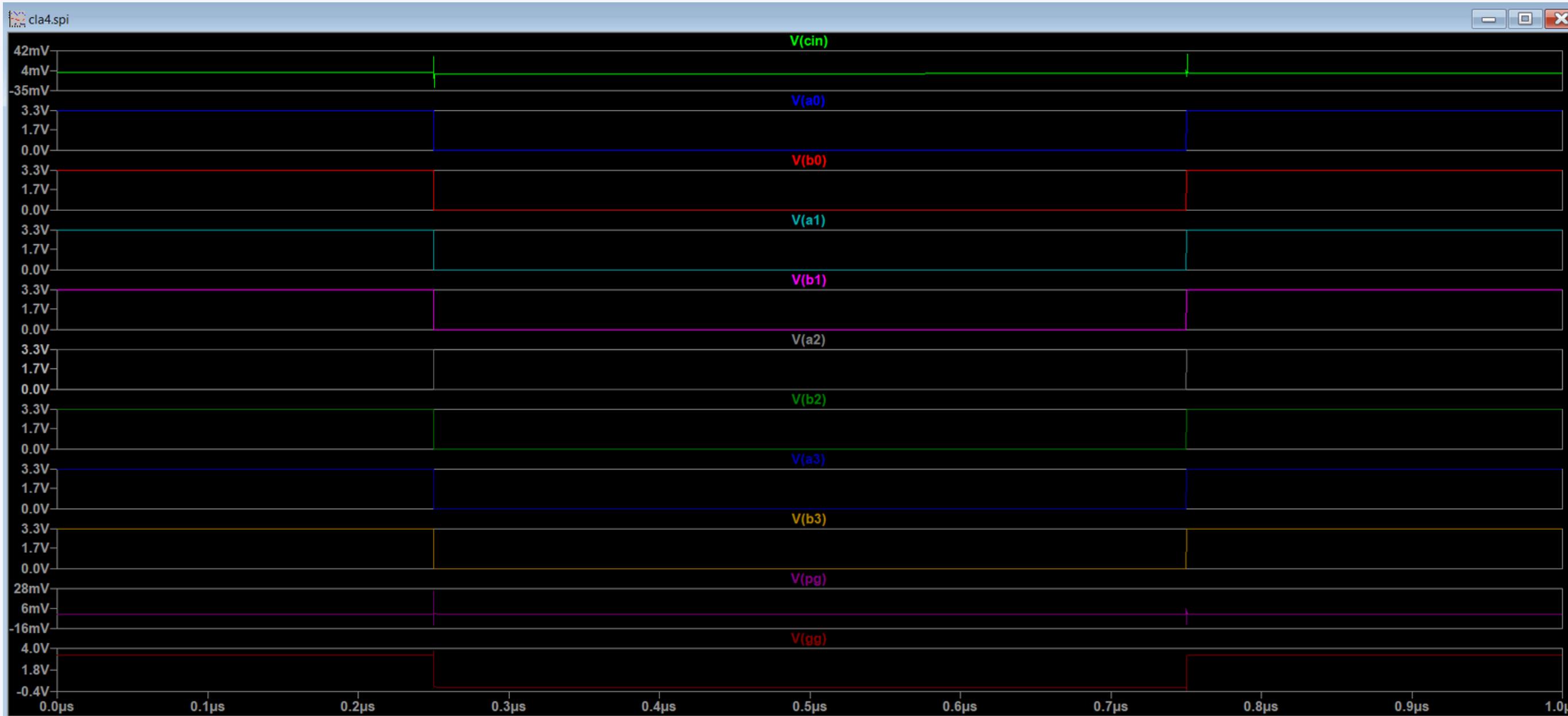
2-Bit CLA LTspice output



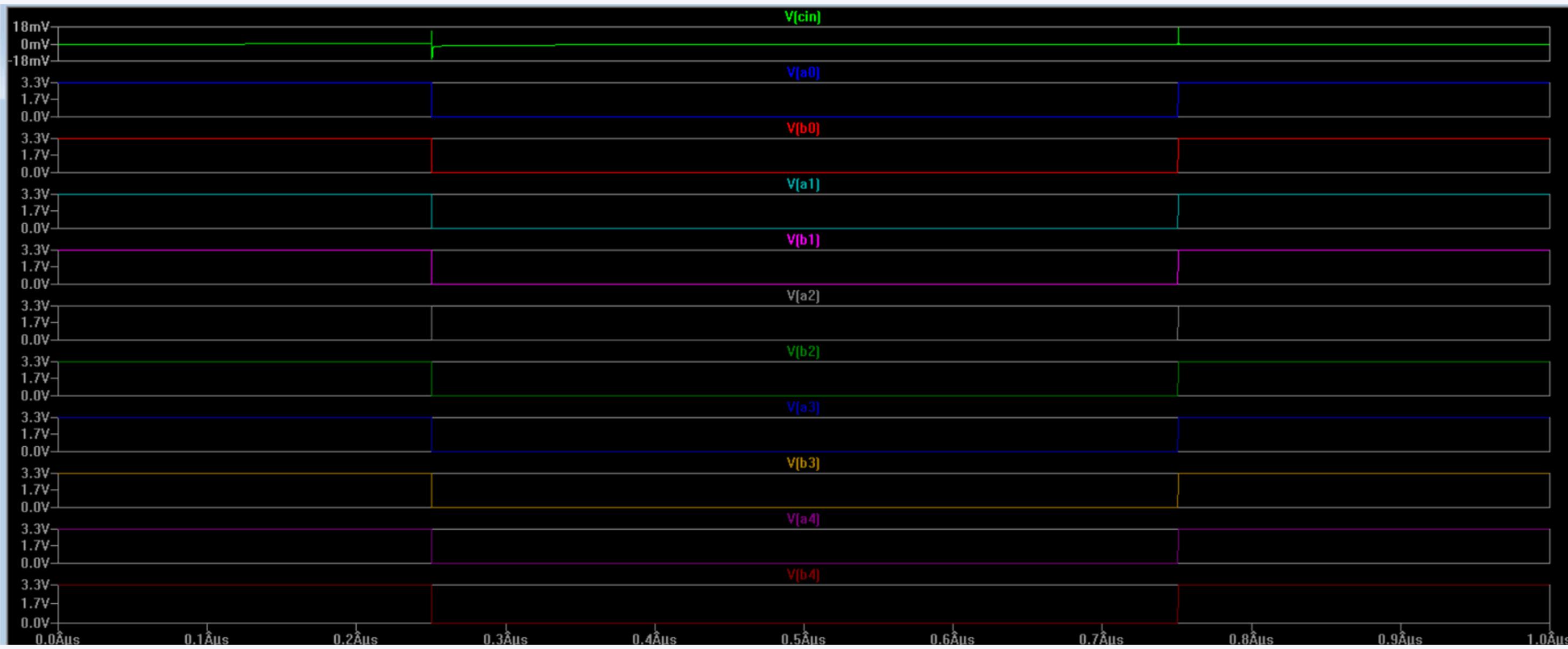
3-Bit CLA LTspice output



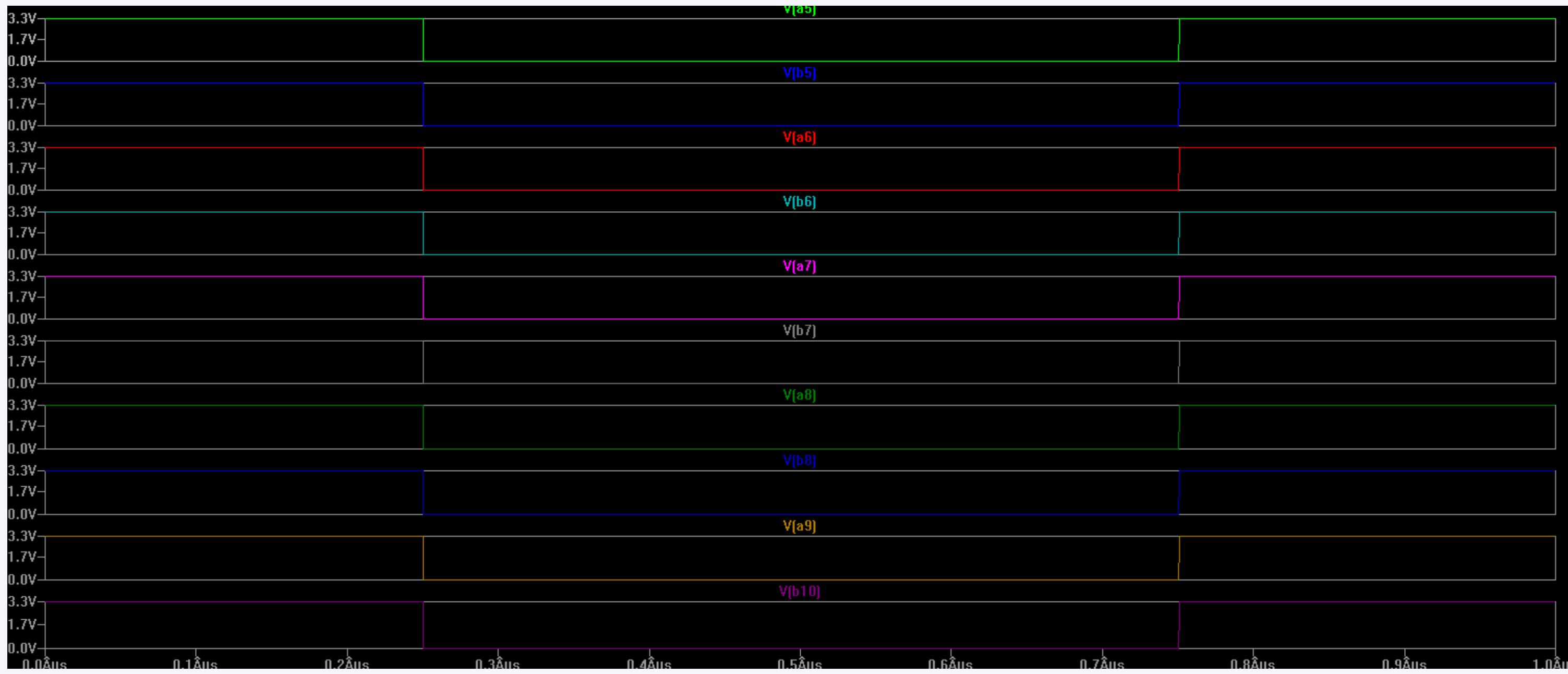
4-Bit CLA LTspice output



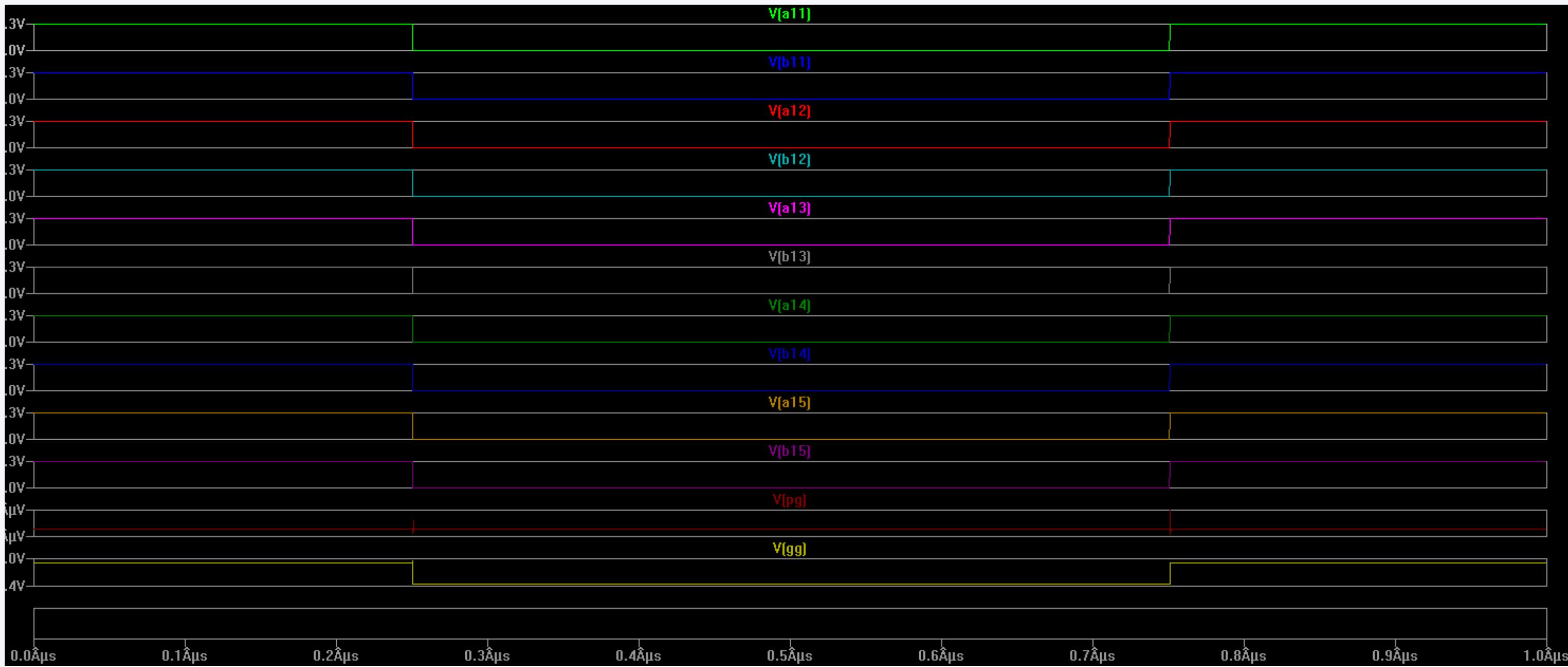
16-Bit CLA LTspice output



16-Bit CLA LTspice output



16-Bit CLA LTspice output



Measurements in LTSPICE

Calculations for All CLA Models.

Model	Power (mW)	Delay (ps)	Area (μm^2)
1-bit CLA	0.95	55	283.18
2-bit CLA	1.52	110	1132.70
3-bit CLA	1.89	75	845.35
4-bit CLA	3.78	150	1237.37
16-bit CLA	15.14	780	4949.49

THANK YOU!

