3-Coefficient FIR Filter Chip

Nick Bodnaruk and Carrie Chung

April 11, 2001

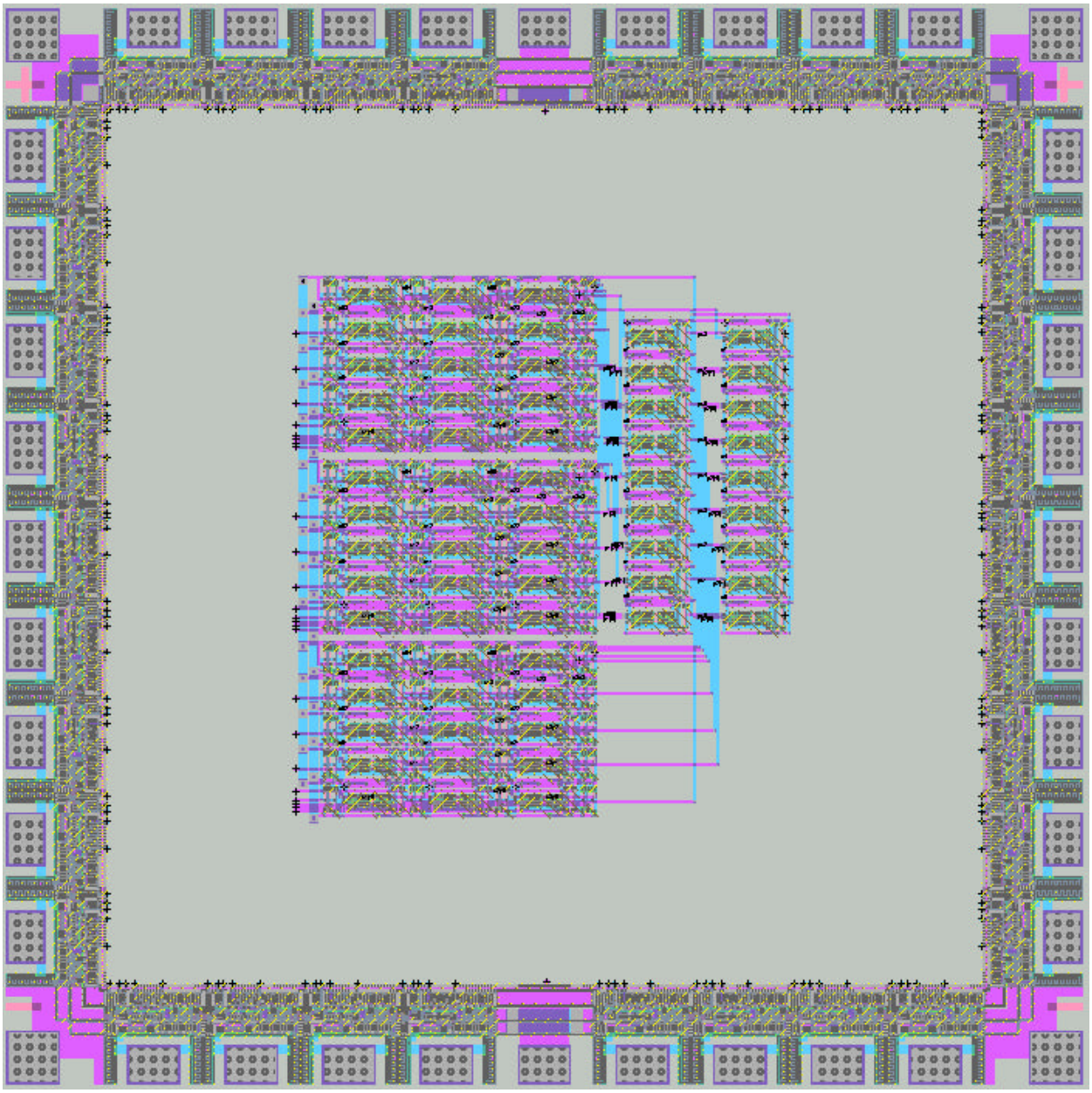


Table of Contents

Functional Overview

This chip contains the logic for a three-coefficient Finite Impulse Response (FIR) filter, described by the equation:

$$y[n] = a_0x[n] + a_1x[n\text{-}1] + a_2x[n\text{-}2],$$

where $y[n]$ is the output signal and $x[n]$ is the input signal at sample time n, and a_k are the coefficients. Each data input and coefficient is a 4-bit number, and the final output is a 10-bit number. These signals are shown in Table 1.

Signals	I/O	Description
d0[3:0]	Input	Current data
d1[3:0]	Input	n-1 data
d2[3:0]	Input	n-2 data
a0[3:0]	Input	Coefficient 0
a1[3:0]	Input	Coefficient 1
a2[3:0]	Input	Coefficient 2
y[9:0]	Output	Output Signal
Total I/O Pins Used		34

Table 1. List of I/O signals.

The major components of the chip are three multipliers, one for each coefficient/data pair, and two adders that are used to sum the products of the multipliers.

Pinout Diagram

VDD	d2[1] IN	d2[2] IN	d2[3] IN	a0[0] IN	GND	a0[1] IN	a0[2] IN	a0[3] IN	a1[0] IN	VDD
d2[0] IN										a1[1] IN
d1[3] IN										a1[2] IN
d1[2] IN										a1[3] IN
d1[1] IN										a2[0] IN
d1[0] IN										a2[1] IN
d0[3] IN										a2[2] IN
d0[2] IN										a2[3] IN
d0[1] IN										y[0] OUT
d0[0] IN										y[1] OUT
GND	y[9] OUT	y[8] OUT	y[7] OUT	y[6] OUT	VDD	y[5] OUT	y[4] OUT	y[3] OUT	y[2] OUT	GND

Chip Floorplan

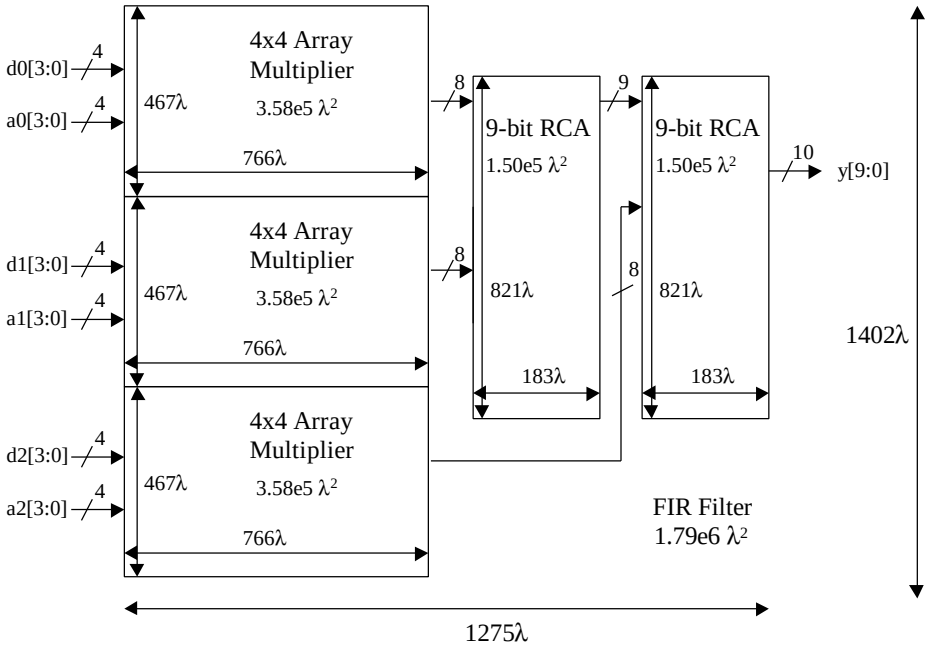


Figure 1. Actual Floorplan with Dimensions.

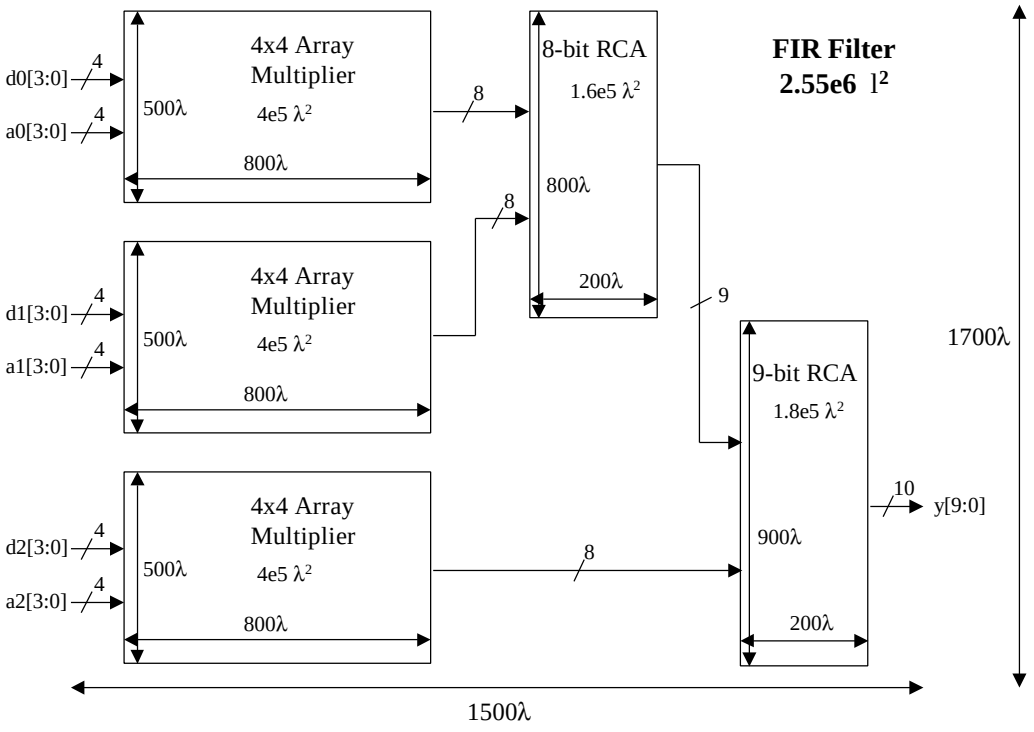


Figure 2. Estimated Floorplan.

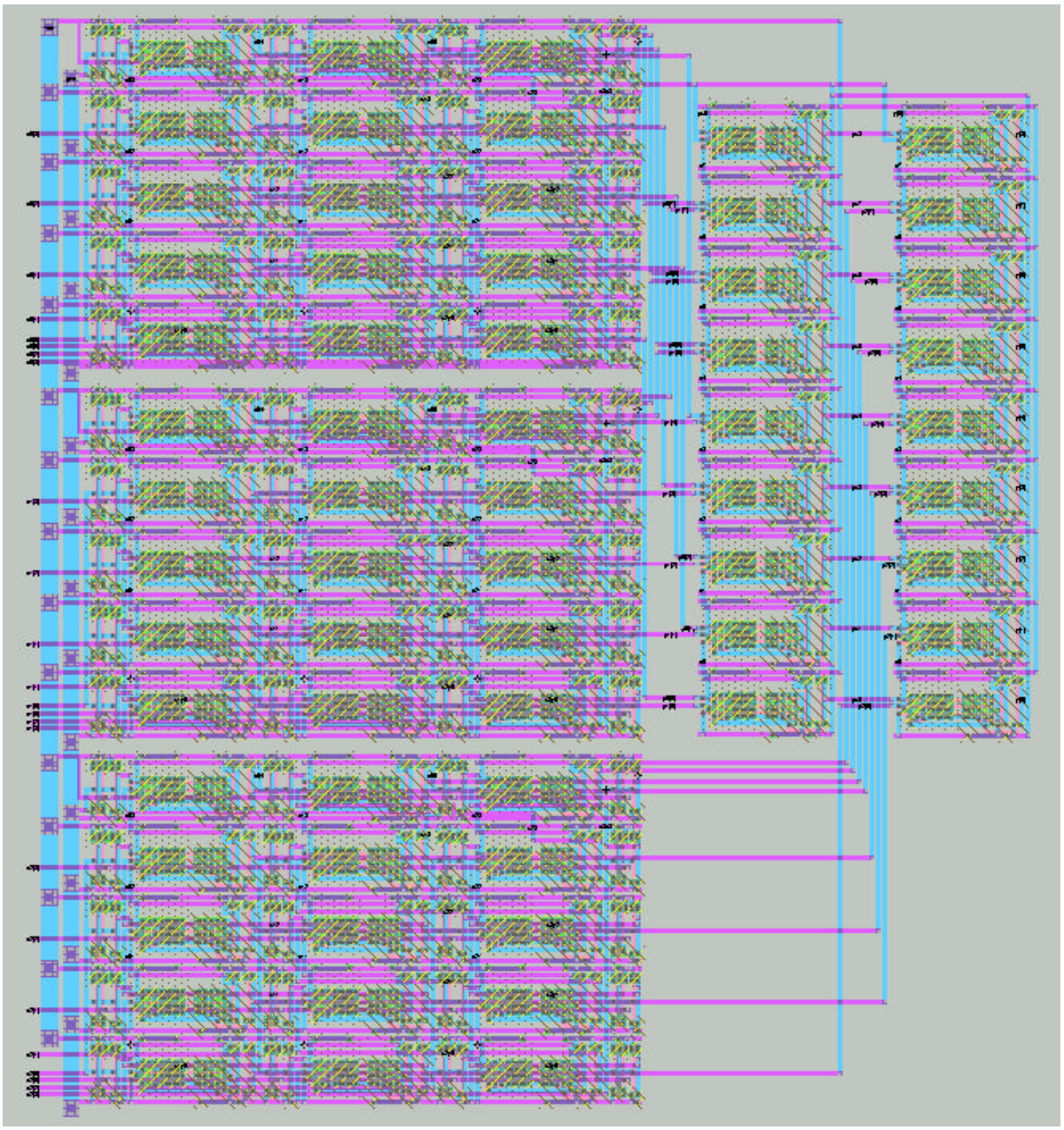


Figure 3. Actual layout of FIR Filter

The arrangement of the facets on the chip is basically the same as our original floorplan estimate. The main differences between the actual and the original floorplans are their areas and the positioning of the facets. The original area estimate was $2.55 \times 10^6\ \lambda^2$, while the actual area occupied by the filter is $1.79 \times 10^6\ \lambda^2$, a difference of $7.6 \times 10^5\ \lambda^2$. We initially allocated 100 λ between each multiplier instance for wiring. In our final design, many of the wires were overlaid on the instances, allowing them to be placed closely to one another. Also, the standard full adder and AND leaf cells were modified such that the n- and p-wells associated with the well contacts do not overlap. This enabled the leaf cells to be stacked closely, resulting in smaller heights for our facets. These improvements along with efficient wiring between the adders led to the reduced overall area in our final design.

Area and Design Time Data

Facet	Function	Inputs	Outputs	Sub-facets	Complexity
AND	logic gate	a,b	y	N/A	1
fulladder	1-bit adder	a, b, cin	sum, cout	N/A	2
9adder	9-bit adder	a[8:0],b[8:0]	s[9:0]	8 fulladders	2
4xbitslice	bitslice for 4slicemul	a,d[3:0],cin[2:0] b[2:0]	ad3, s[2:0] cout[2:0]	4 AND gates 3 fulladders	3
4slicemul	4x4 multiplier	d[3:0] a[3:0]	p[7:0]	5 4xbitslices	4
3cfir	3-coefficient FIR Filter	d0[3:0], a0[3:0] d1[3:0], a1[3:0] d2[3:0], a2[3:0]	y[9:0]	3 4slicemul's 2 9adders	3

Table 2. Facet Descriptions.

Facet	Actual Area ( )^2)	Estimated Area ( )^2)	Time Spent (Hours)				
			Schematic	Icon	Layout	Simulation	Verification
AND	49*97 4.75E+03	50*80 4.00E+03	0.5	0	0.5	0	0.5
fulladder	169*97 1.64E+04	170*80 1.36E+04	0.5	0	1	0	0.5
9adder	821*183 1.50E+05	800*200 1.60E+05	1	0.5	3	6	1
4xbitslice	709*97 6.88E+04	N/A N/A	1	0.5	2	1	1
4slicemul	467*766 3.58E+05	500*800 4.00E+05	8	0.5	20	6	10
3cfir	1402*1275 1.79E+06	1500*1700 2.55E+06	4	0	15	4	10
		Total	15	1.5	41.5	17	23
		Grand Total	98				

Table 3. Area and Design Time.

Schematic

While designing the schematics, most of our time was spent on the 4slicemul facet. This was because our initial design was not optimized for the layout process. It was a direct translation of the array multiplier at the logical level. We redesigned it such that it utilizes the 4xbitslice facet in order to reduce the amount of layout work, at the expense of introducing redundant transistors.

Layout

The multiplier and the filter required many wires for the interconnections. The placement of these wires was the most difficult aspect of the layout, and consequently, took the most amount of time. Additional time was spent towards placing the facets as close together as possible.

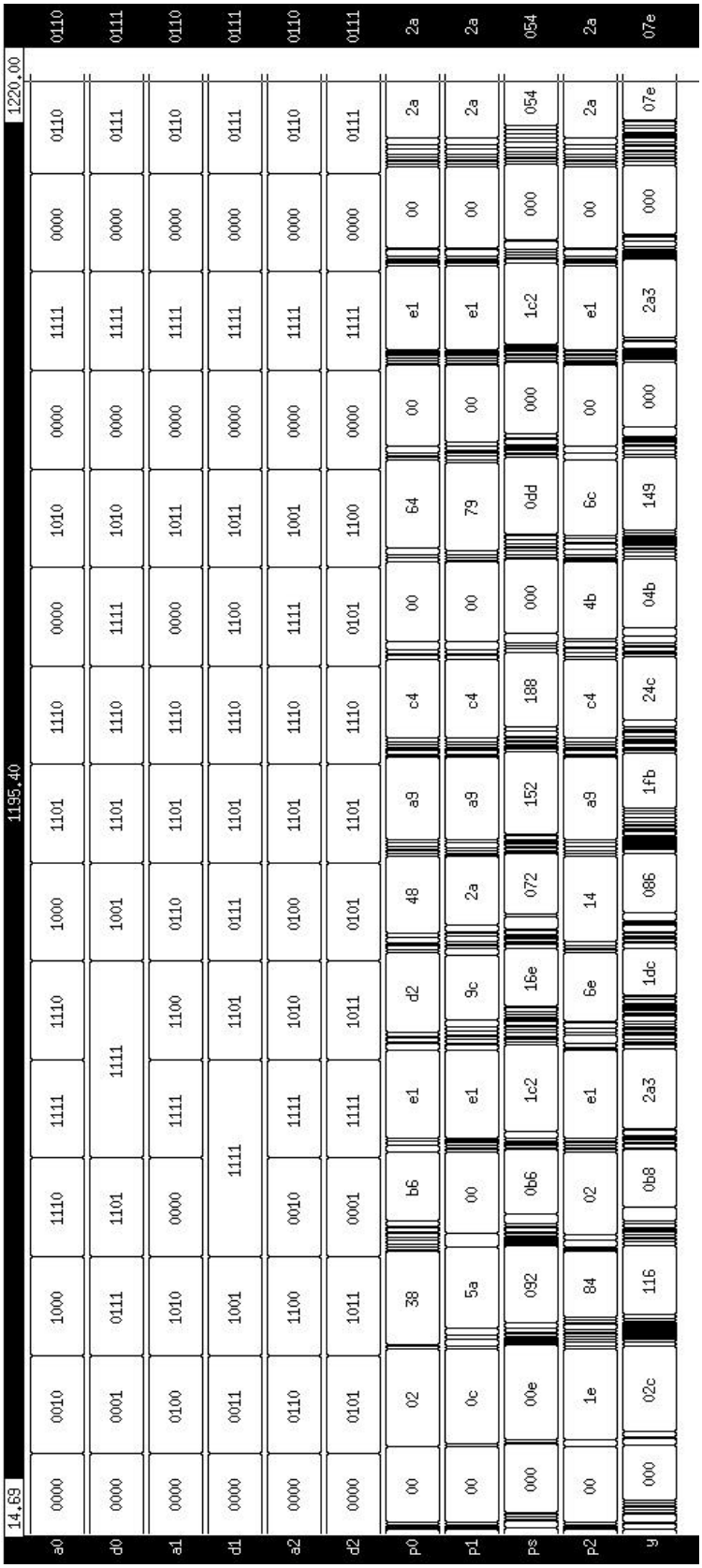
Simulation

IRSIM was used to simulate all of our circuits. Since we were not familiar with this program, several hours were spent learning how to use it. The rest of the simulation time was used to develop test vectors that verify the performance of the designs.

Verification

There was a problem in the Electric NCC routine that causes NCC to fail when the 'merge parallel components' option is selected. Finding this bug was a part of our verification process. We also used Gemini to double check Electric's NCC results and no problems were found. The rest of the verification time was used to debug mistakes we made during the wiring process.

Simulation Results



Expected Results for Simulation

test#	d0	a0	d1	a1	d2	a2	p0	p1	ps	p2	y(hex)
1	0	0	0	0	0	0	0	0	0	0	0
2	1	2	3	4	5	6	2	c	e	1e	2c
3	7	8	9	10	11	12	38	5a	92	84	116
4	13	14	15	0	1	2	b6	0	0b6	2	0b8
5	15	15	15	15	15	15	e1	e1	1c2	e1	2a3
6	15	14	13	12	11	10	d2	9c	16e	6e	1dc
7	9	8	7	6	5	4	48	2a	72	14	86
8	13	13	13	13	13	13	a9	a9	152	a9	1fb
9	14	14	14	14	14	14	c4	c4	188	c4	24c
10	15	0	12	0	5	15	0	0	0	4b	04b
11	10	10	11	11	12	9	64	79	0dd	6c	149
12	0	0	0	0	0	0	0	0	0	0	0
13	15	15	15	15	15	15	e1	e1	1c2	e1	2a3
14	0	0	0	0	0	0	0	0	0	0	0
15	7	6	7	6	7	6	2a	2a	54	2a	07e

Table 4: Inputs and Outputs for Simulation.

Max Delay: 55nsec (for layout simulation)

Verification Results

The top-level design 3cfir and all its subfacets pass DRC, ERC, and NCC. NCC was checked with both Electric and Gemini.

Test Plan

The first step in testing the chip is to examine the power (VDD) and ground (GND) pins before connecting a power supply to the chip. The resistance between VDD and GND pins should be very high (on the order of MΩs). The resistance between two VDD pins should be small, and similarly for GND pins. Also, the resistance between any power pin and signal pin should be high. With these tests complete, hook up a power supply, and make sure a large current is not drawn and that the chip stays at room temperature.

Unfortunately, all the I/O pins on this chip are used, so we were not able to include any test circuits to help in the testing procedure. To test the basic functionality of the chip, make sure all the input pins are grounded (not floating). Connect a0[0] and d0[0] high, this should result in y[0] outputting high. Ground a0[0] and set a0[1] high, which will cause y[0] to fall low, and y[1] to output high. If these two tests succeed, proceed to hook up the chip to a chip tester and use the following IRSIM deck. The desired results are shown in Table 4 above.

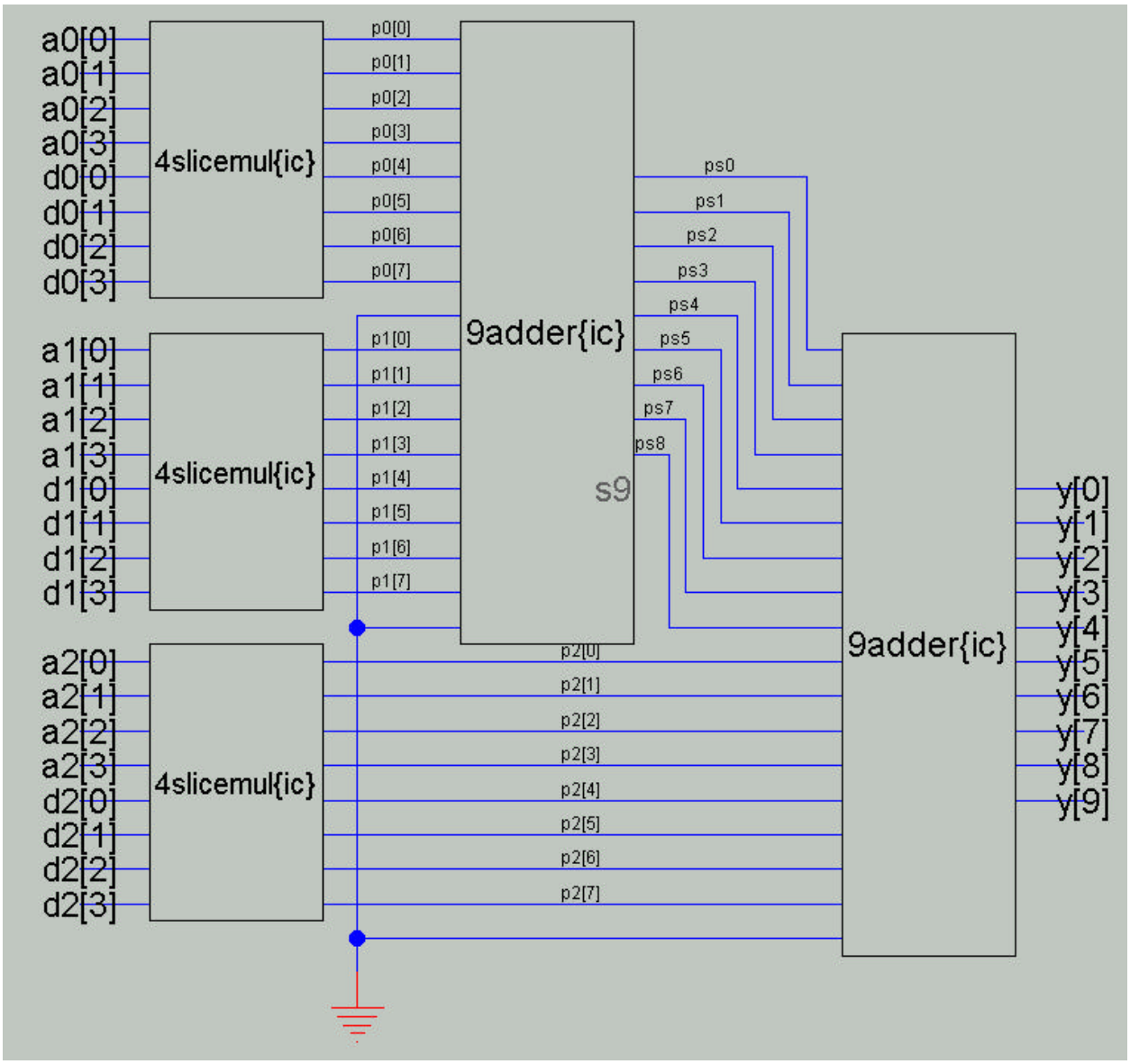
```
vector d0 d0[3] d0[2] d0[1] d0[0]
vector d1 d1[3] d1[2] d1[1] d1[0]
vector d2 d2[3] d2[2] d2[1] d2[0]
vector a0 a0[3] a0[2] a0[1] a0[0]
vector a1 a1[3] a1[2] a1[1] a1[0]
vector a2 a2[3] a2[2] a2[1] a2[0]
vector p0 p0[7] p0[6] p0[5] p0[4] p0[3] p0[2] p0[1] p0[0]
vector p1 p1[7] p1[6] p1[5] p1[4] p1[3] p1[2] p1[1] p1[0]
vector ps ps8 ps7 ps6 ps5 ps4 ps3 ps2 ps1 ps0
vector p2 p2[7] p2[6] p2[5] p2[4] p2[3] p2[2] p2[1] p2[0]
vector y y[9] y[8] y[7] y[6] y[5] y[4] y[3] y[2] y[1] y[0]
ana a0 d0 a1 d1 a2 d2 p0 p1 ps p2 y
set d0 0000
set a0 0000
set d1 0000
set a1 0000
set d2 0000
set a2 0000
s 80
set d0 0001
set a0 0010
set d1 0011
set a1 0100
set d2 0101
set a2 0110
s 80
set d0 0111
set a0 1000
set d1 1001
set a1 1010
set d2 1011
```

```
set a2 1100
s 80
set d0 1101
set a0 1110
set d1 1111
set a1 0000
set d2 0001
set a2 0010
s 80
set d0 1111
set a0 1111
set d1 1111
set a1 1111
set d2 1111
set a2 1111
s 80
set d0 1111
set a0 1110
set d1 1101
set a1 1100
set d2 1011
set a2 1010
s 80
set d0 1001
set a0 1000
set d1 0111
set a1 0110
set d2 0101
set a2 0100
s 80
set d0 1101
set a0 1101
set d1 1101
set a1 1101
set d2 1101
set a2 1101
s 80
set d0 1110
set a0 1110
set d1 1110
set a1 1110
set d2 1110
set a2 1110
s 80
set d0 1111
set a0 0000
set d1 1100
set a1 0000
set d2 0101
set a2 1111
s 80
set d0 1010
set a0 1010
set d1 1011
set a1 1011
set d2 1100
set a2 1001
```

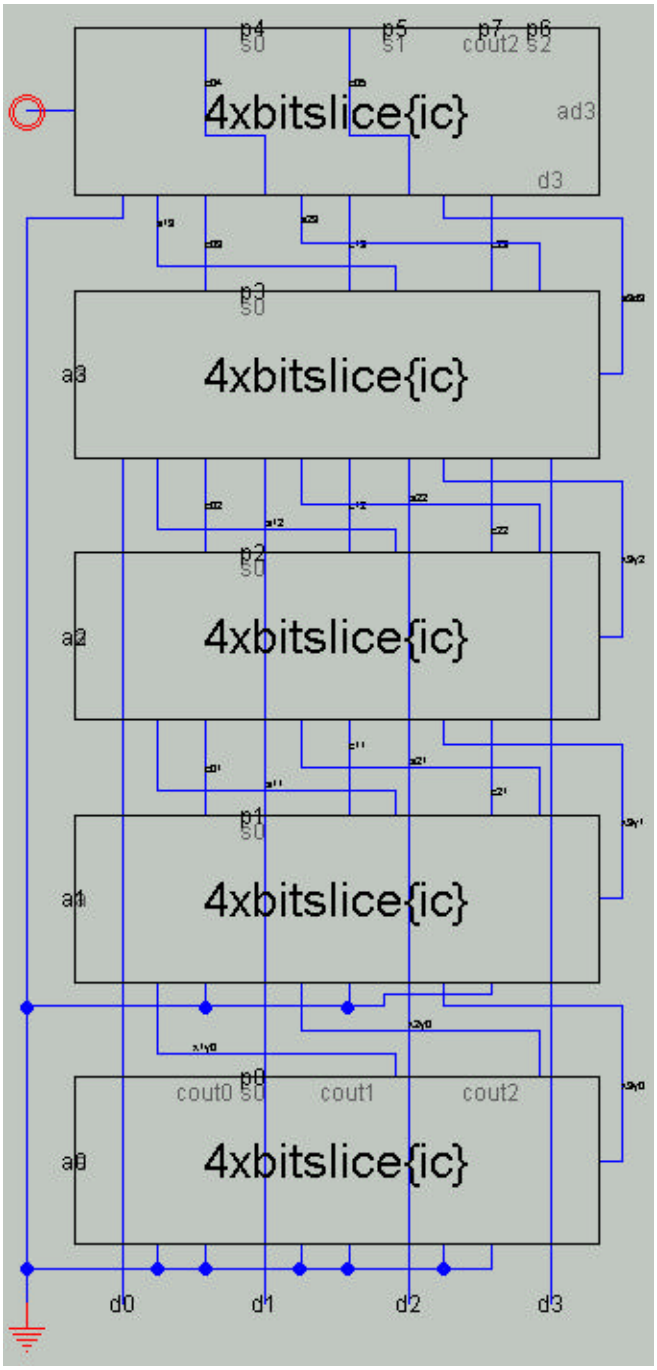
```
s 80

set d0 0000
set a0 0000
set d1 0000
set a1 0000
set d2 0000
set a2 0000
s 80
set d0 1111
set a0 1111
set d1 1111
set a1 1111
set d2 1111
set a2 1111
s 80
set d0 0000
set a0 0000
set d1 0000
set a1 0000
set d2 0000
set a2 0000
s 80
set d0 0111
set a0 0110
set d1 0111
set a1 0110
set d2 0111
set a2 0110
s 80
```

Schematics

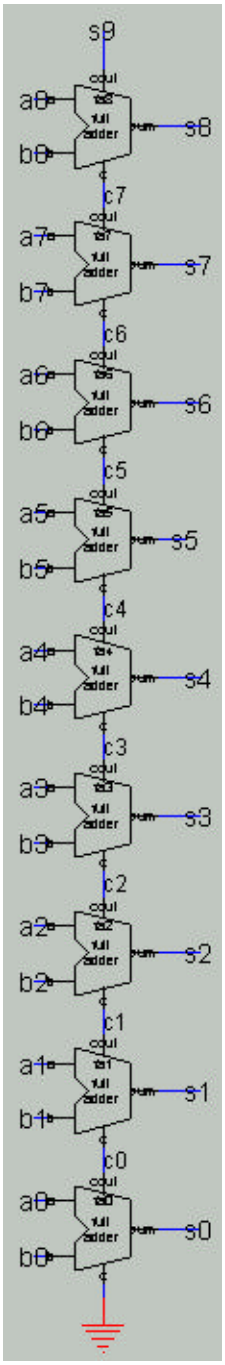
3cfir



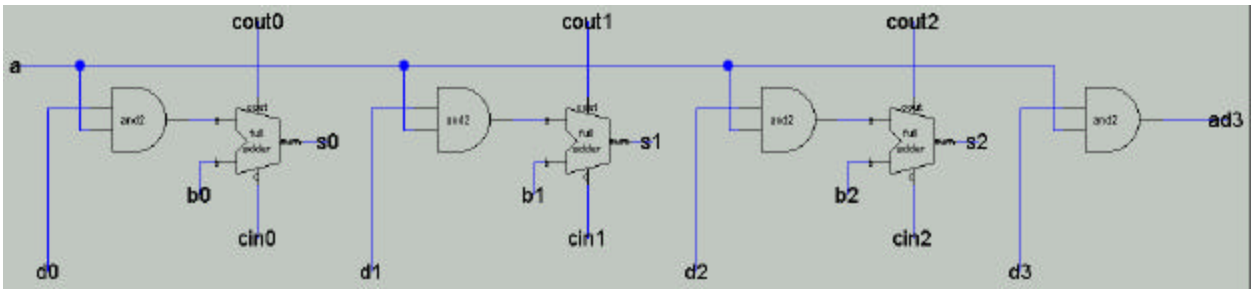
4slicemul



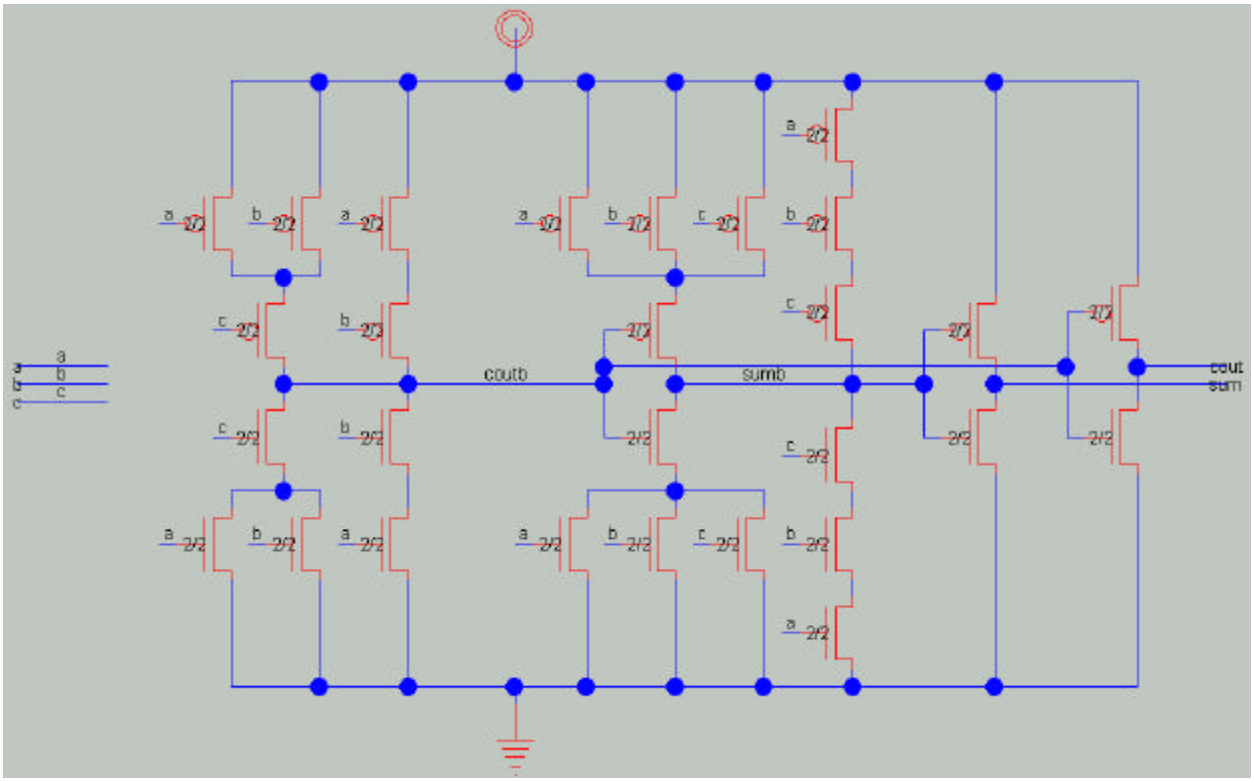
9adder



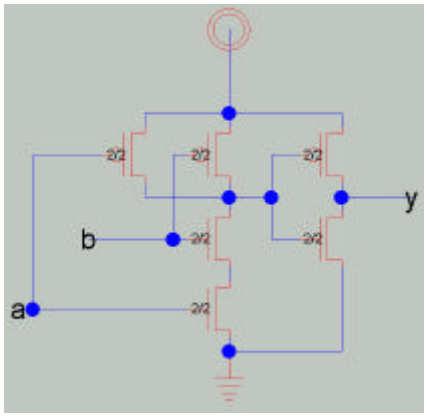
4xbitslice



fulladder

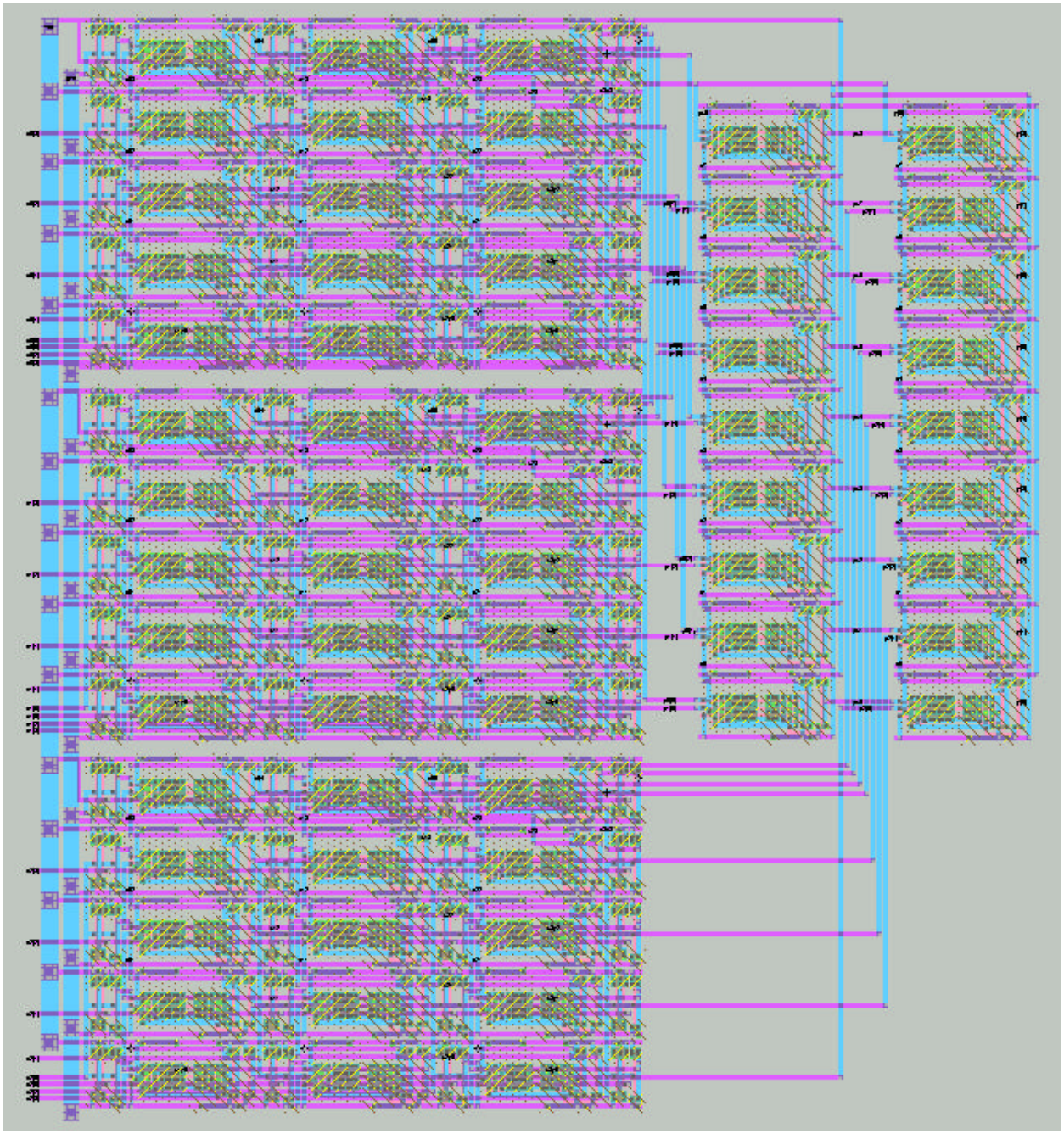


and2

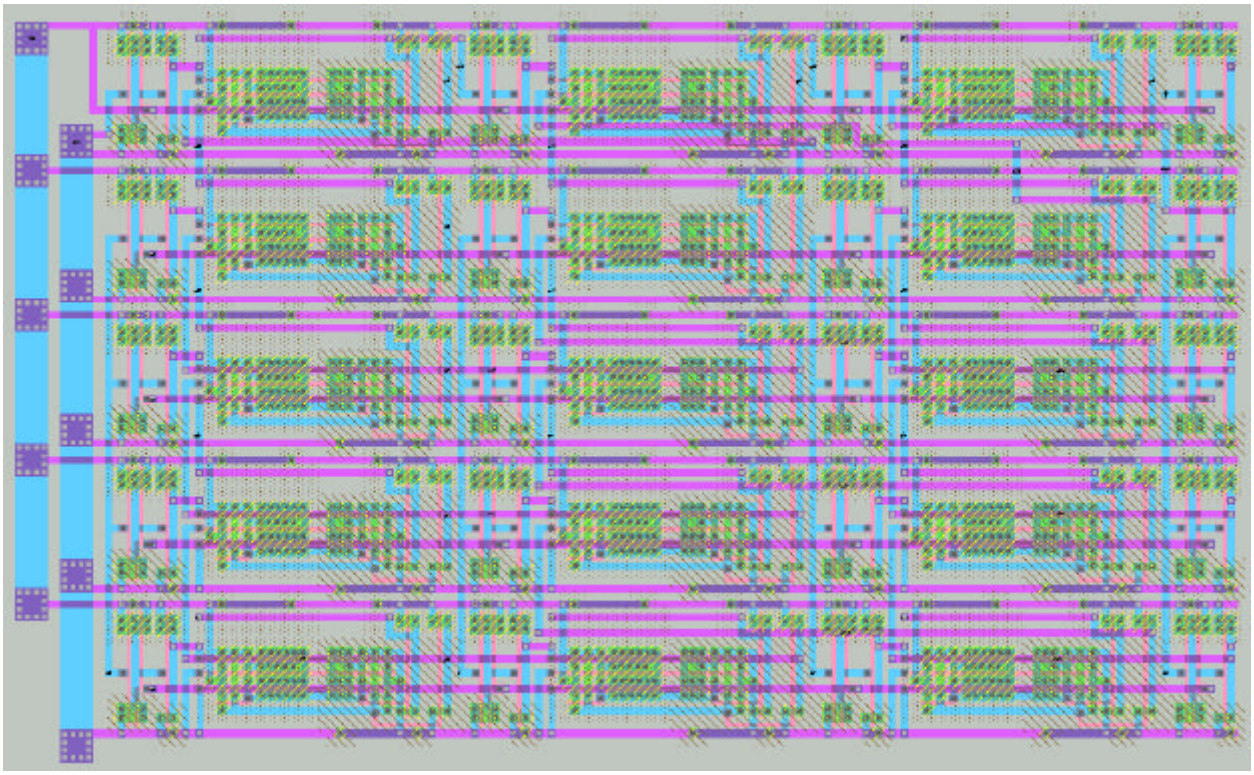


Layouts

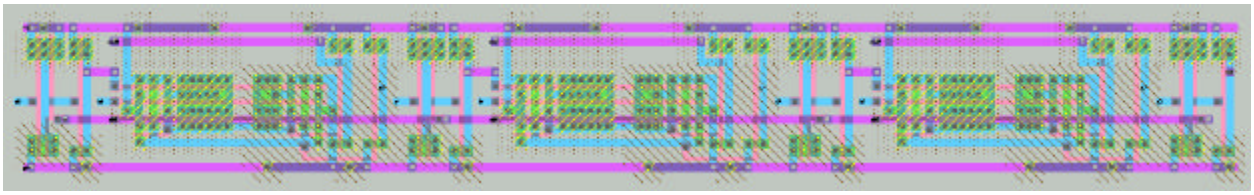
3cfir



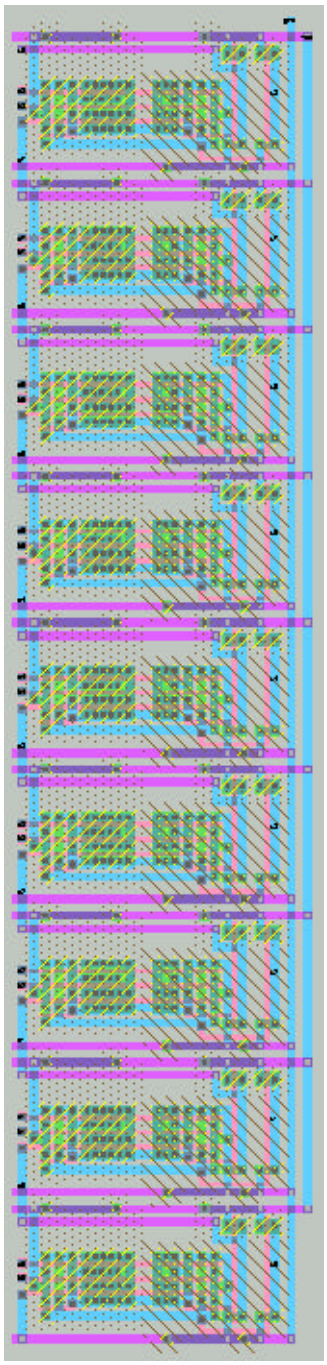
4slicemul



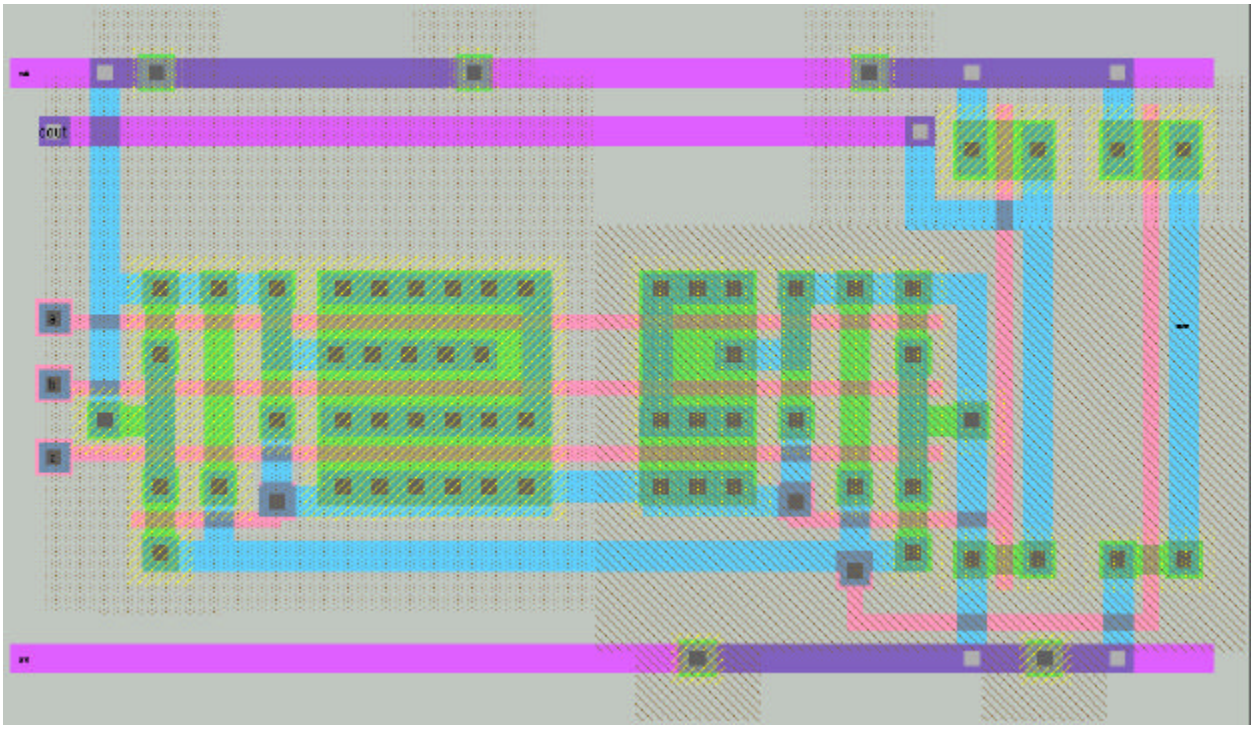
4xbitslice



9adder



fulladder



and2