BOOTH ENCODING MULTIPLIER

Team Member:
Meng Zhang, Tianyu Feng
Ming Gao, Xintong Li

OUTLINE……

1. Principle introduction
2. Circuit Estimation
3. Circuit Design

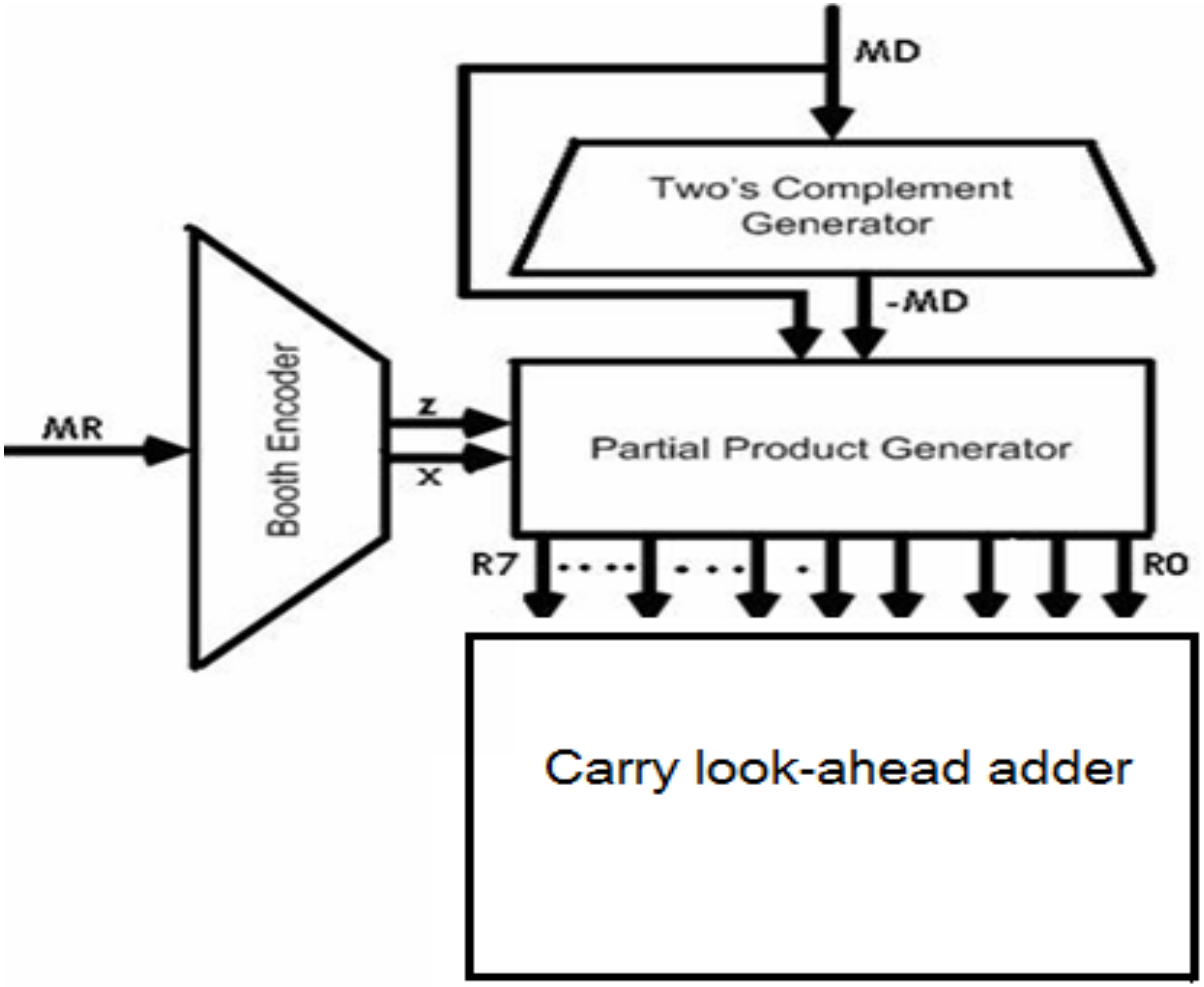
Principle Introduction

PROJECT SUMMARY

- In our project, we are aiming to build up a Booth Encoding Radix-4 8 bits Multiplier. Booth Encoding is an effective method which greatly increase the speed of our algebra.

- We also attempts to reduce the number of partial products generated in a multiplication process by using the modified Booth algorithm

SYSTEM ARCHITECTURE



ARCHITECTURE DESCRIPTION

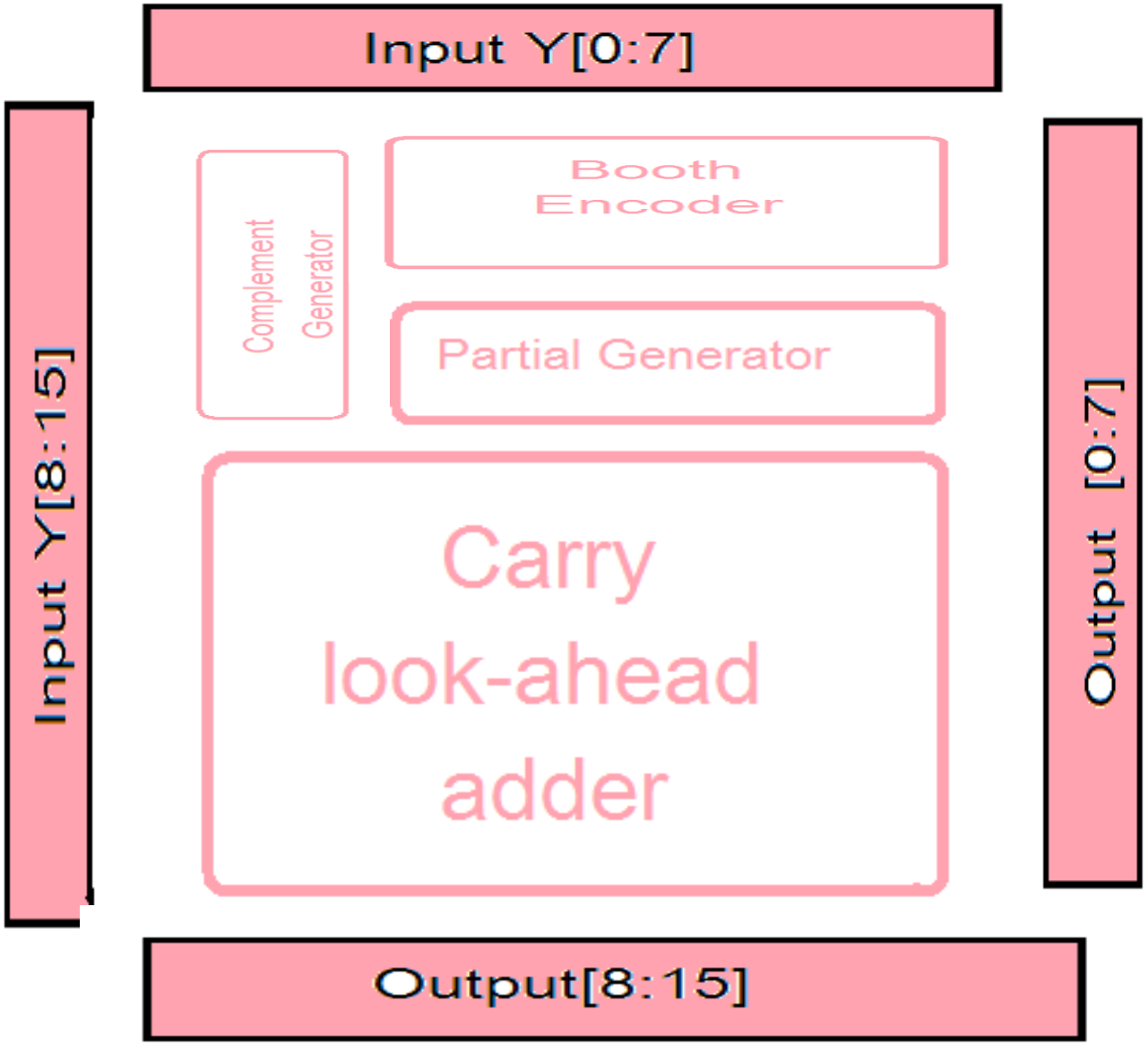
The multiplier takes in 2 8-bits operands: the multiplier(MR) and the multiplicand (MD), then produces 16-bit multiplication result of the two as its output.

The architecture comprises four parts: Complement Generator, Booth Encoder, Partial Product and Carry Look-ahead Adder. We adapt the simplest way to demonstrate the multiplier.

PRINCIPLE INTRODUCTION

Booth encoder	• Booth's algorithm involves repeatedly adding one of two predetermined values to a product P , then performing a rightward arithmetic shift on *P*.
Partial Product Generator	• A product formed by multiplying the multiplicand by one digit of the multiplier when the multiplier has more than one digit. Partial products are used as intermediate steps in calculating larger products.
Carry look-ahead adder	• a type of adder used in digital logic. It can be contrasted with the simpler, but usually slower, *ripple carry adder*

CHIP FLOORPLAN



FULL SYSTEM DESIGN BACKGROUND

- The multiplier has M-bits X and N-bits Y as input and generate M*N-bits output Z.

$$X = \sum_{i=0}^{M-1} X_i 2^i$$

$$Y = \sum_{j=0}^{N-1} Y_j 2^j$$

$$Z = X \times Y = \sum_{i=0}^{M-1} \left(\sum_{j=0}^{N-1} X_i Y_j 2^{i+j} \right)$$

FULL SYSTEM DESIGN BACKGROUND

$$X \times 00111110 = X \times (2^5 + 2^4 + 2^3 + 2^2 + 2^1) = X \times 62,$$

The number of partial product and the number of operations can be reduced to two by rewriting the equation as

$$X \times 00111110 = X \times (2^6 - 2^1) = X \times (64 - 2) = X \times 62$$

When Booth encounters the first digit of a block of ones (0 1), it follows this scheme.

When Booth encounters the end of the block (1 0), it follows a subtraction.

FULL SYSTEM DESIGN BACKGROUND

- Booth Encoding algorithm…
- Do multiplication on both non-negative and negative operand.
- Decrease the number of partial product, which lead to substantially delay and area reduction.

  Shown as below:

Circuit Estimation

ESTIMATION:TRANSISTOR

Gate Type	Transistor #	required #	Total transistor
AND2	6	155	930
XOR2	8	248	1984
INV	2	478	956
NAND2	4	263	1052
NAND3	6	268	1608
NAND4	8	175	600
OR2	6	35	210
Total			8140

ESTIMATION: POWER & AREA

$P(dynamic) = \alpha C V^2 f$
$= 0.1 * 5354 * 4 * 0.05 * 2 * 5 * 5 * 10^{-9}$
$= 0.0053$ mw/MHz

L = 4 * 0.05 * 5354 = 1070.8 um
I = 1070.8 * (0.02/2 + 0.002) = 12.8496 nA
P (static) = 5 * 12.8496 = 0.000064248 mW

Assuming the frequency is 100 MHz

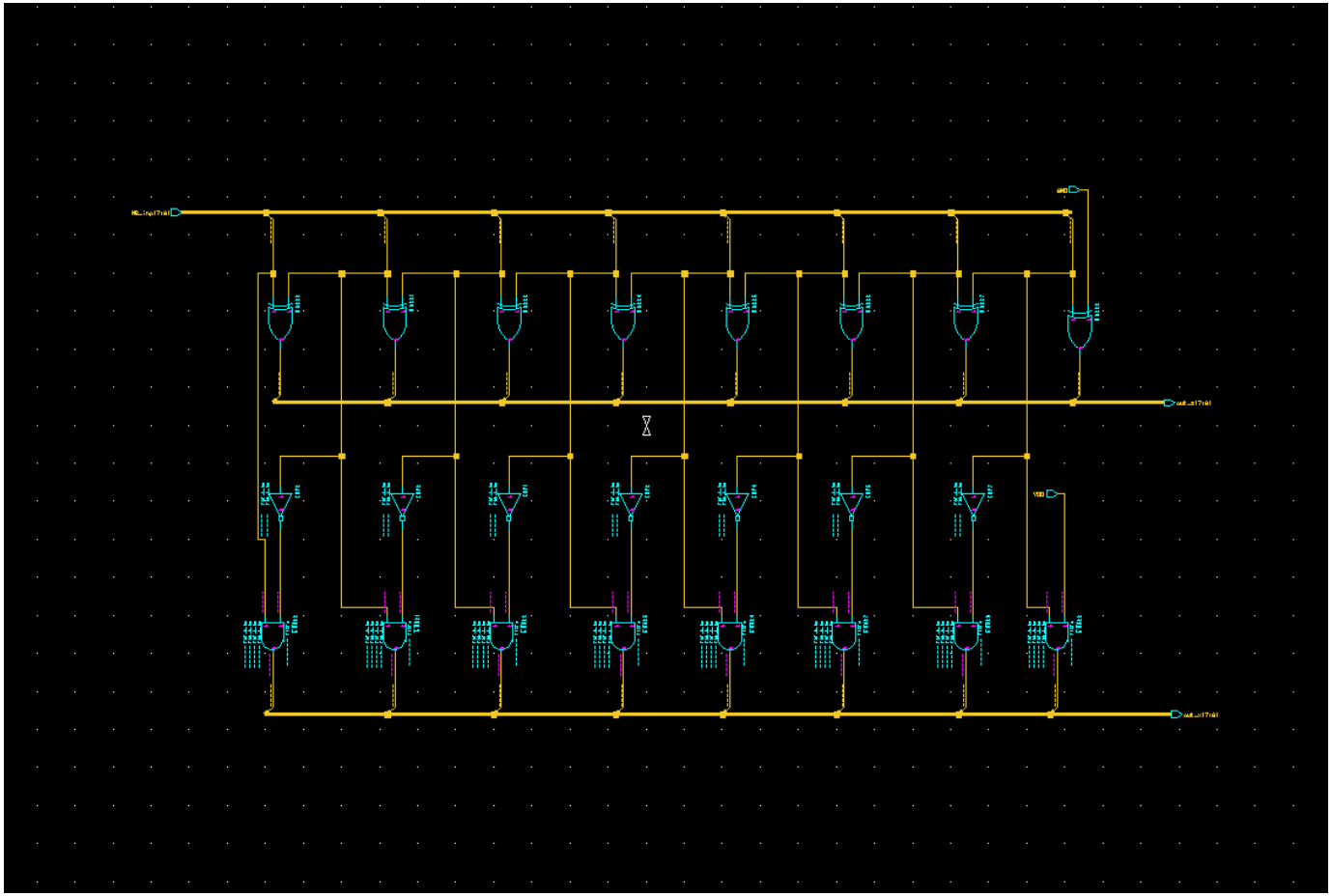
P = 0.53mw

Area=7.76 cm^2

Circuit Design

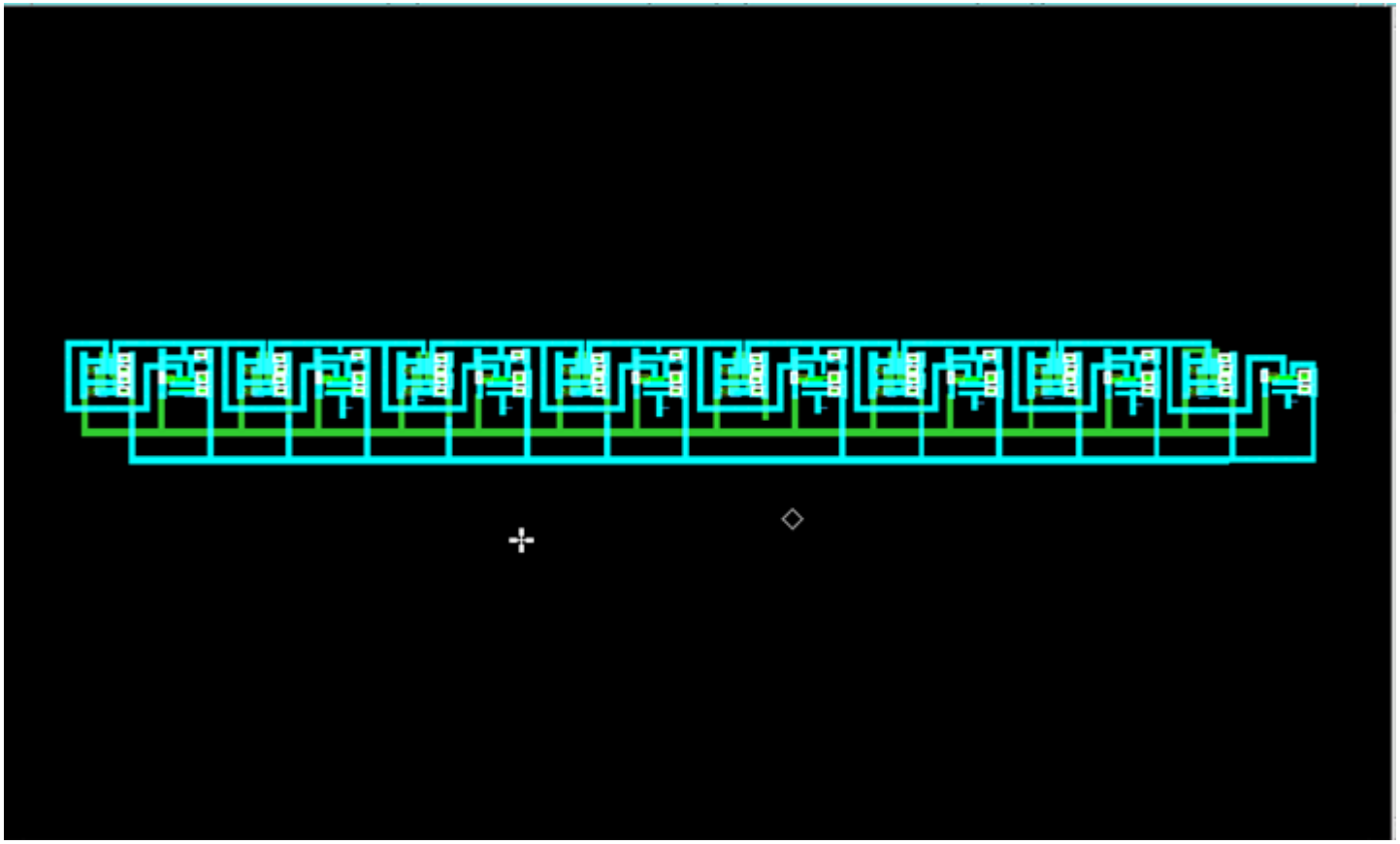
SUB-CIRCUIT DESIGN

- Booth Encoder-Schematic



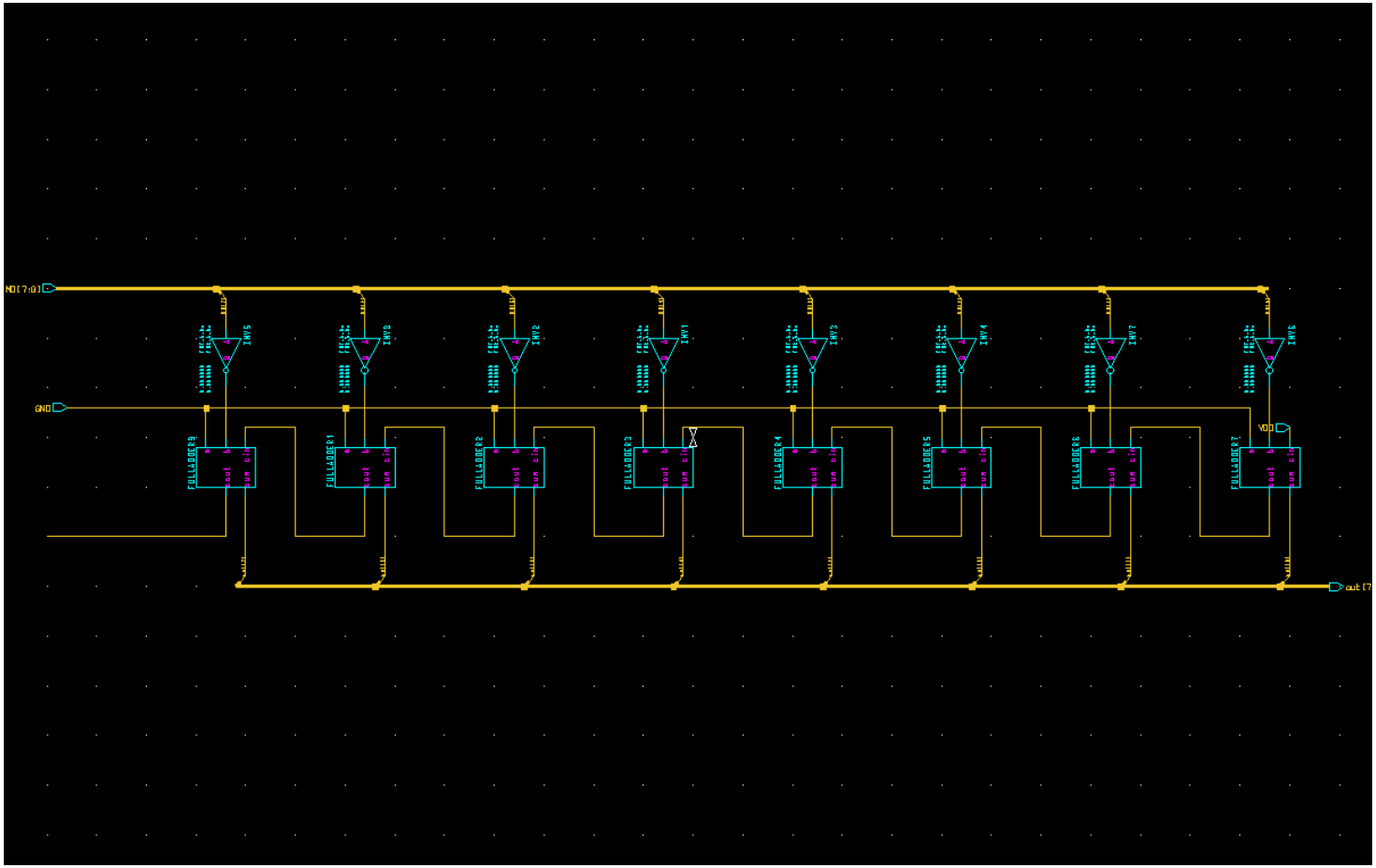
SUB-CIRCUIT DESIGN

- Booth Encoder- Layout



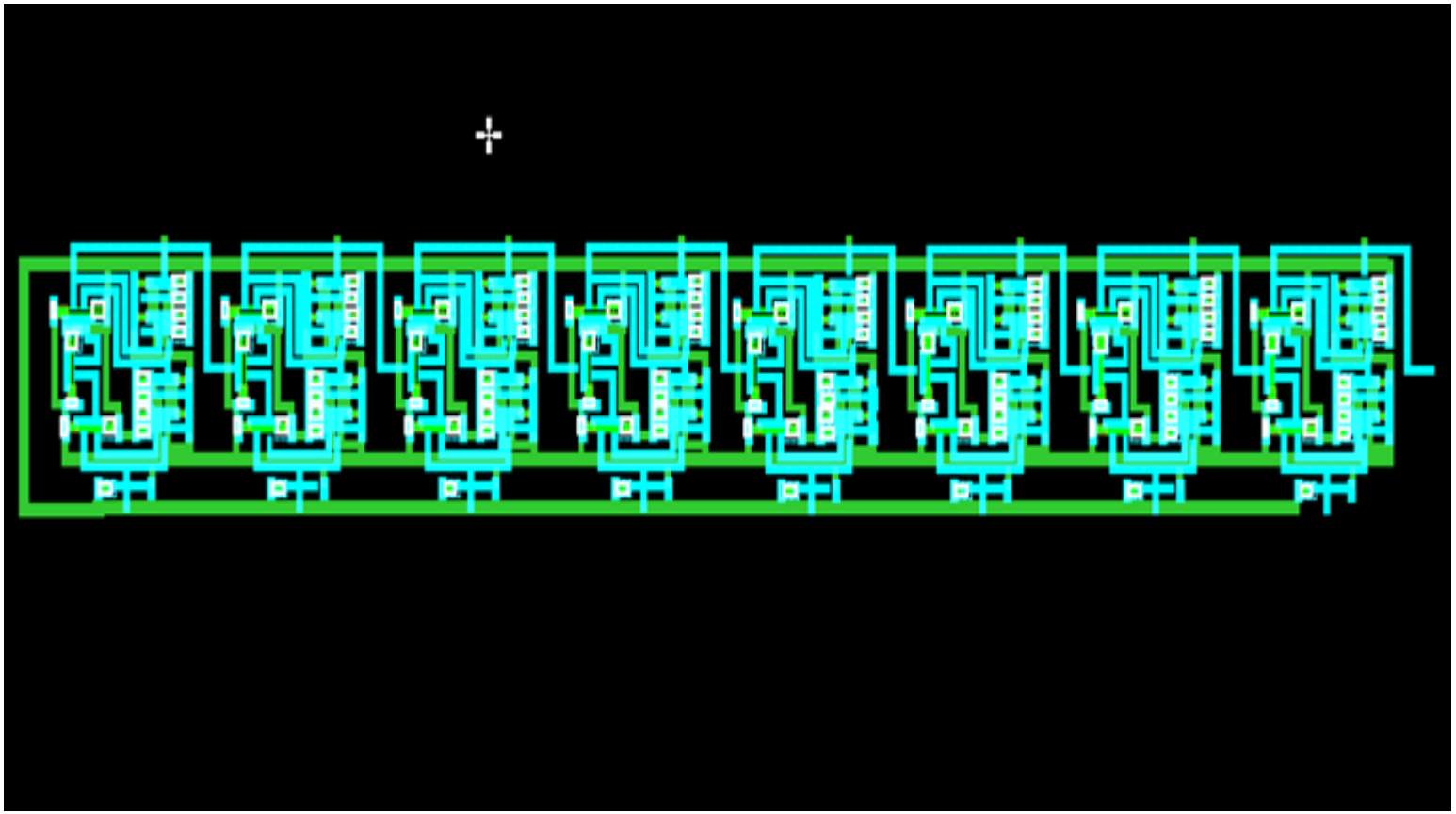
SUB-CIRCUIT DESIGN

- Complement Generator-Schematic



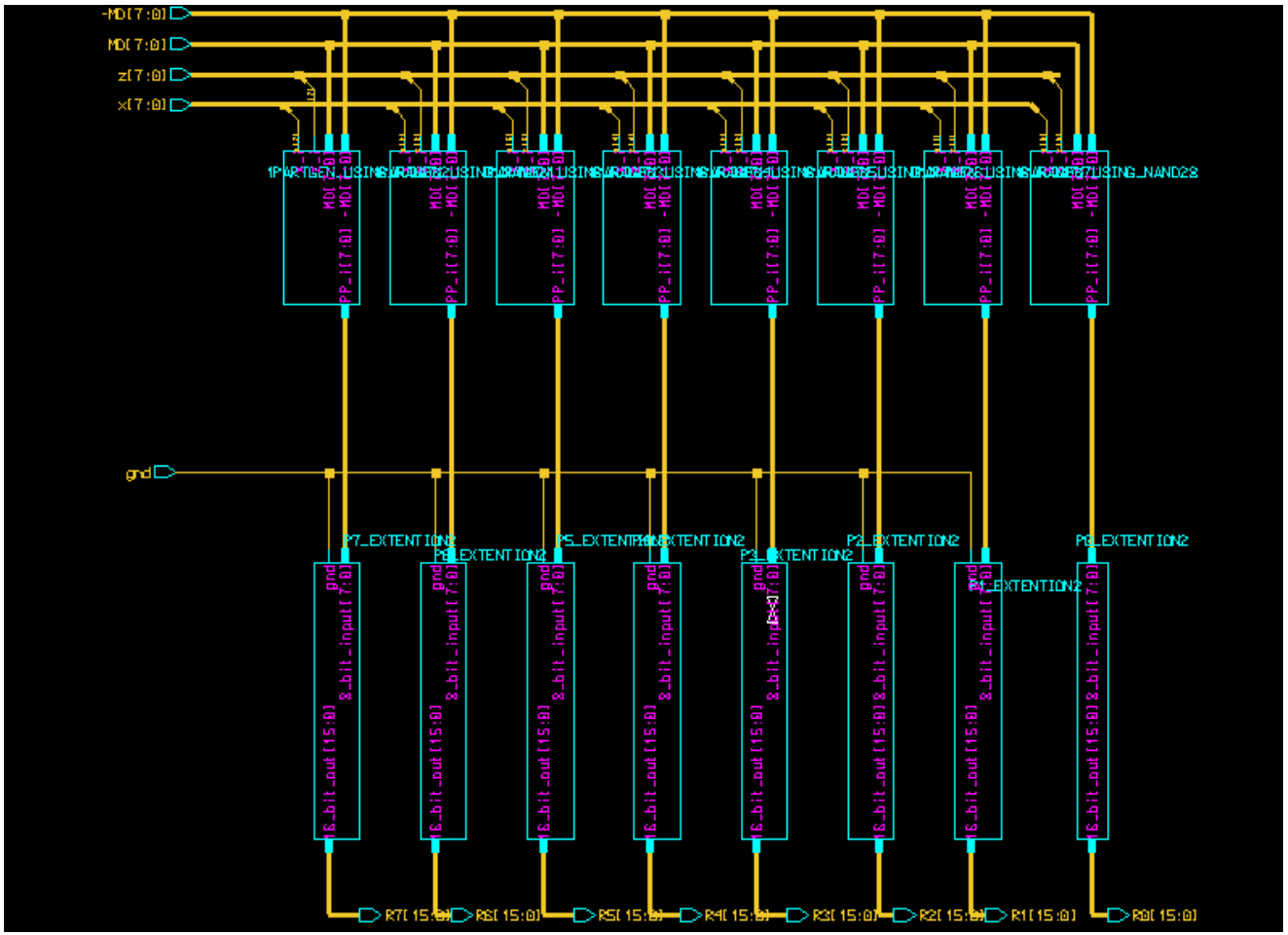
SUB-CIRCUIT DESIGN

- Complement Generator-Layout



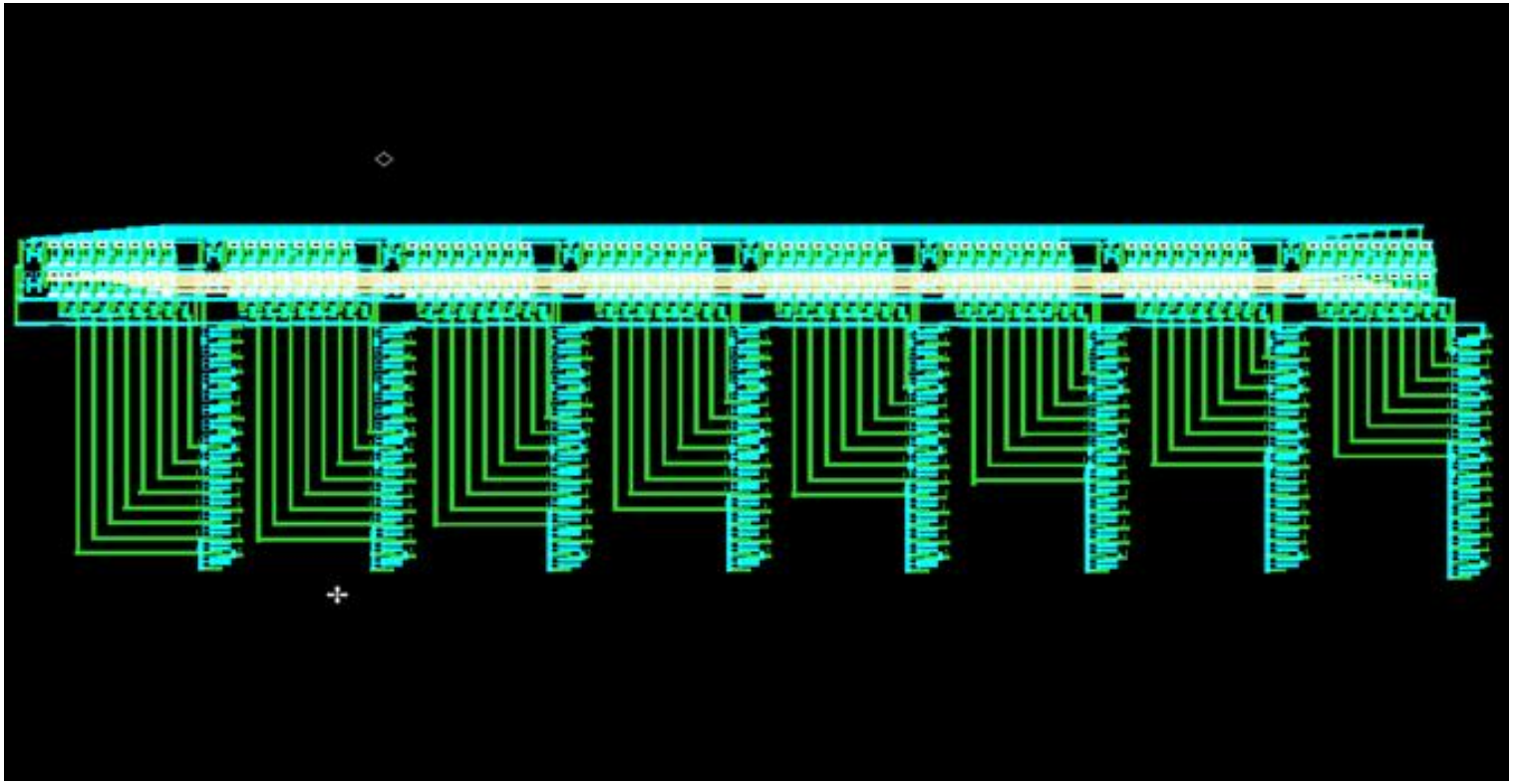
SUB-CIRCUIT DESIGN

- Partial Products Generator-Schematic



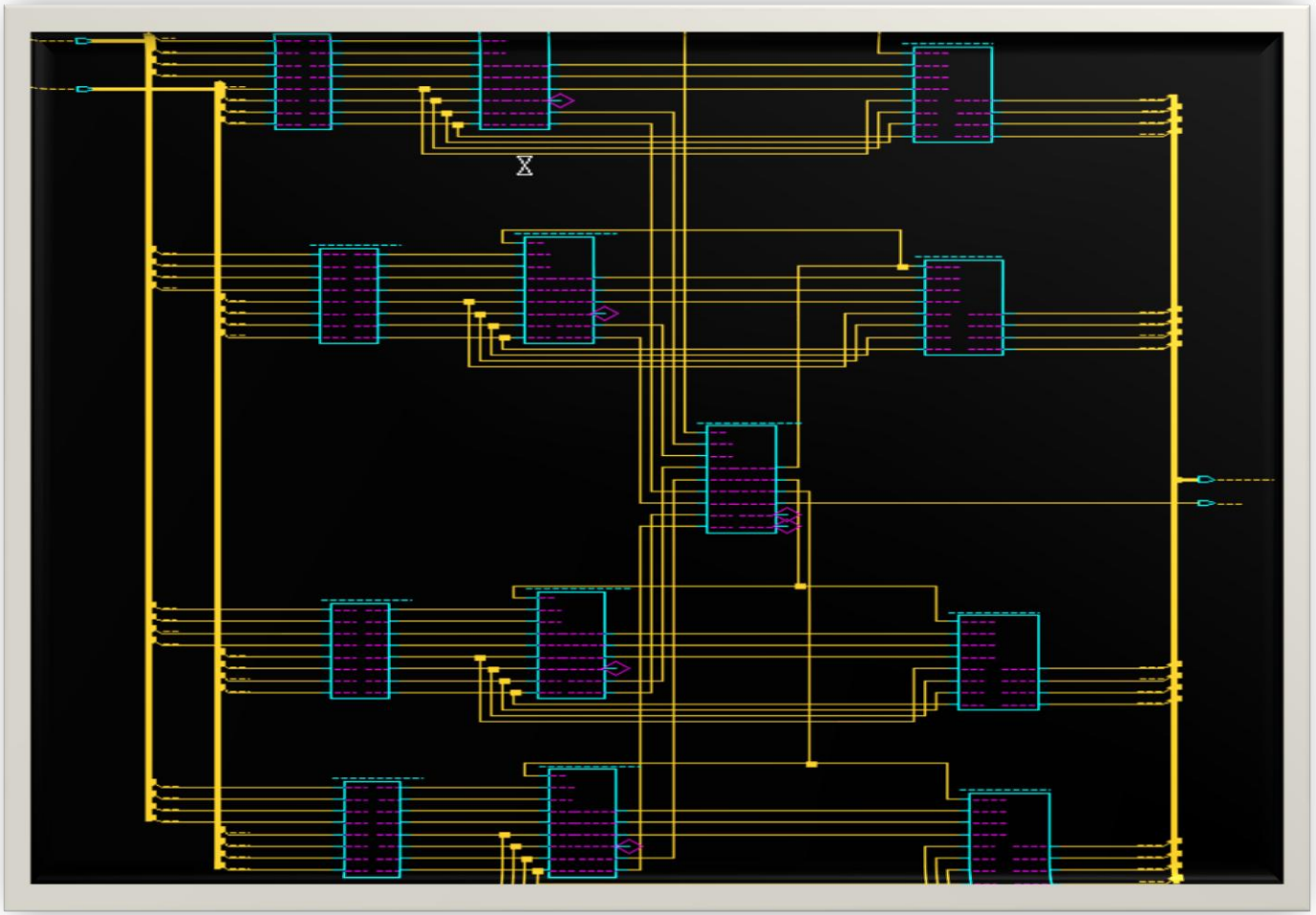
SUB-CIRCUIT DESIGN

- Partial Products Generator-Layout



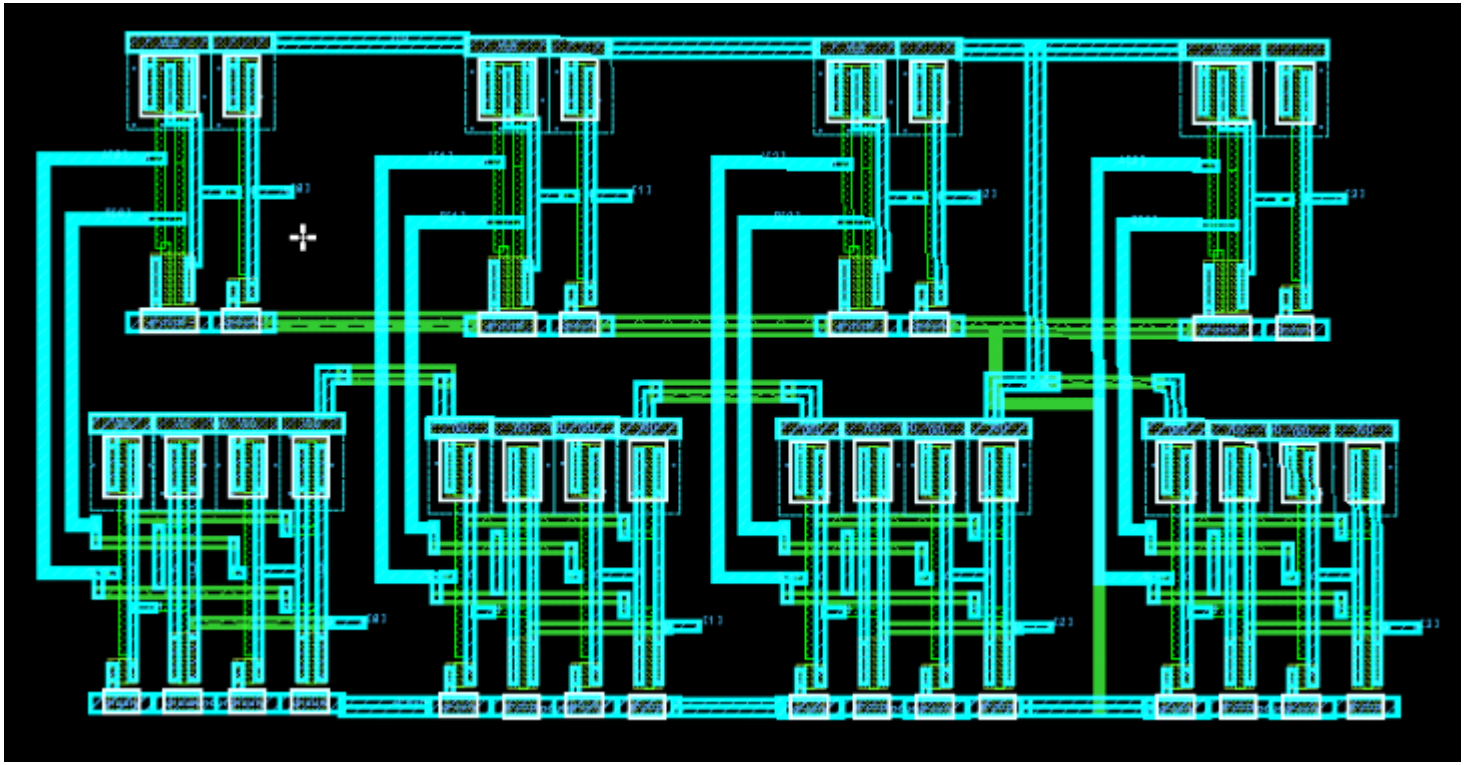
SUB-CIRCUIT DESIGN

- Carry Look-ahead Adders-Schematic



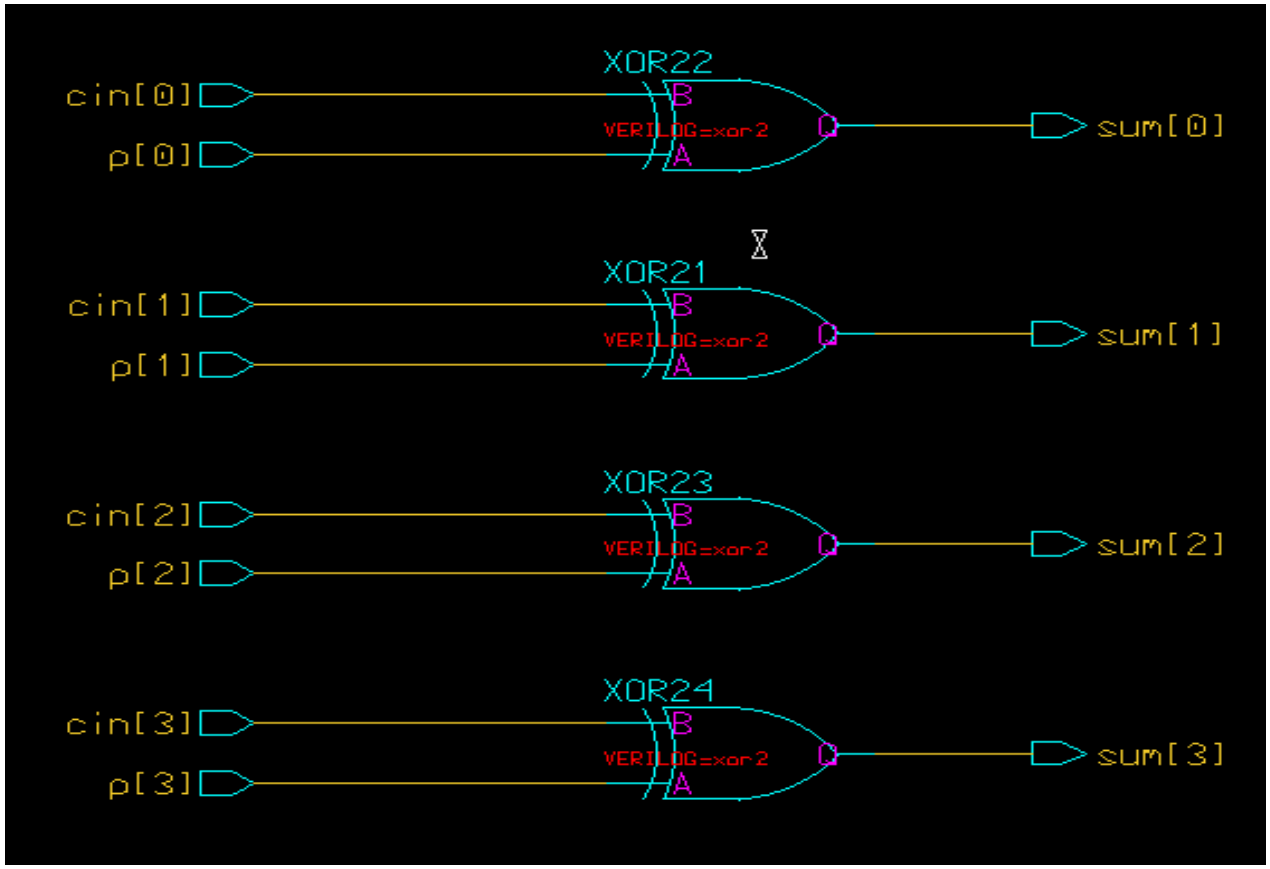
SUB-CIRCUIT DESIGN

- Carry Look-ahead Adders-Layout



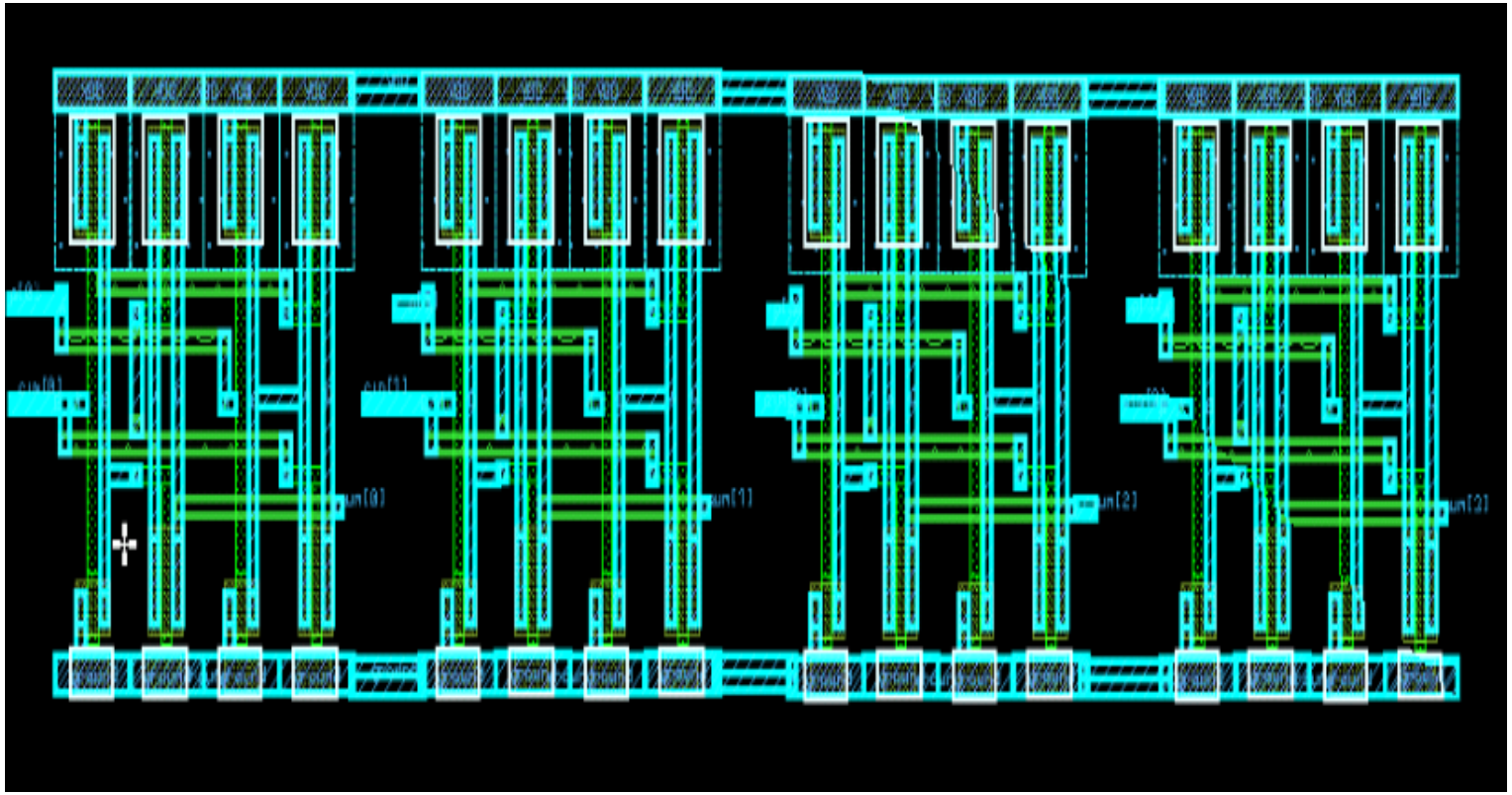
SUB-CIRCUIT DESIGN

- Sum Generator-Schematic



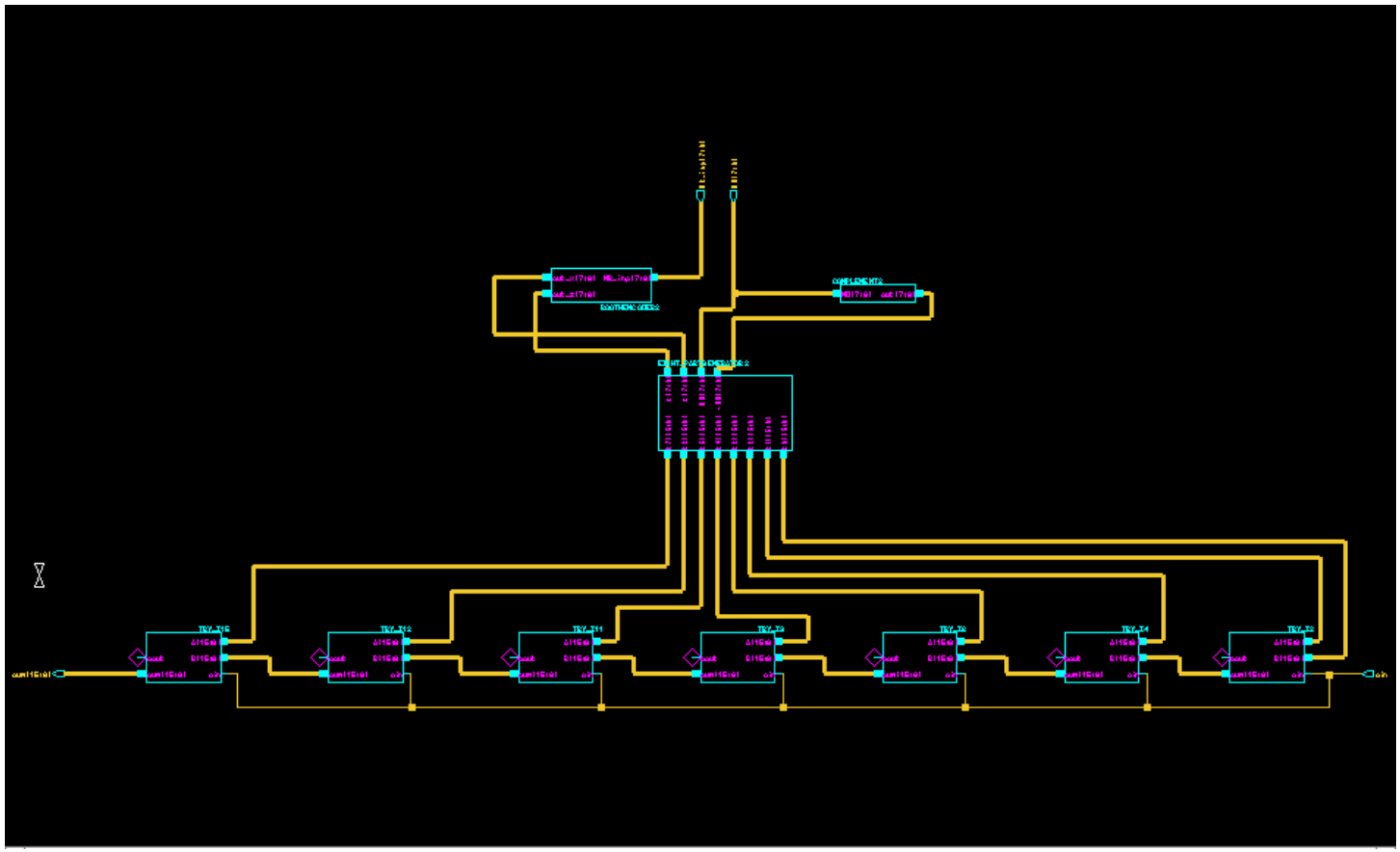
SUB-CIRCUIT DESIGN

- Sum Generator-Layout



FULL SYSTEM DESIGN

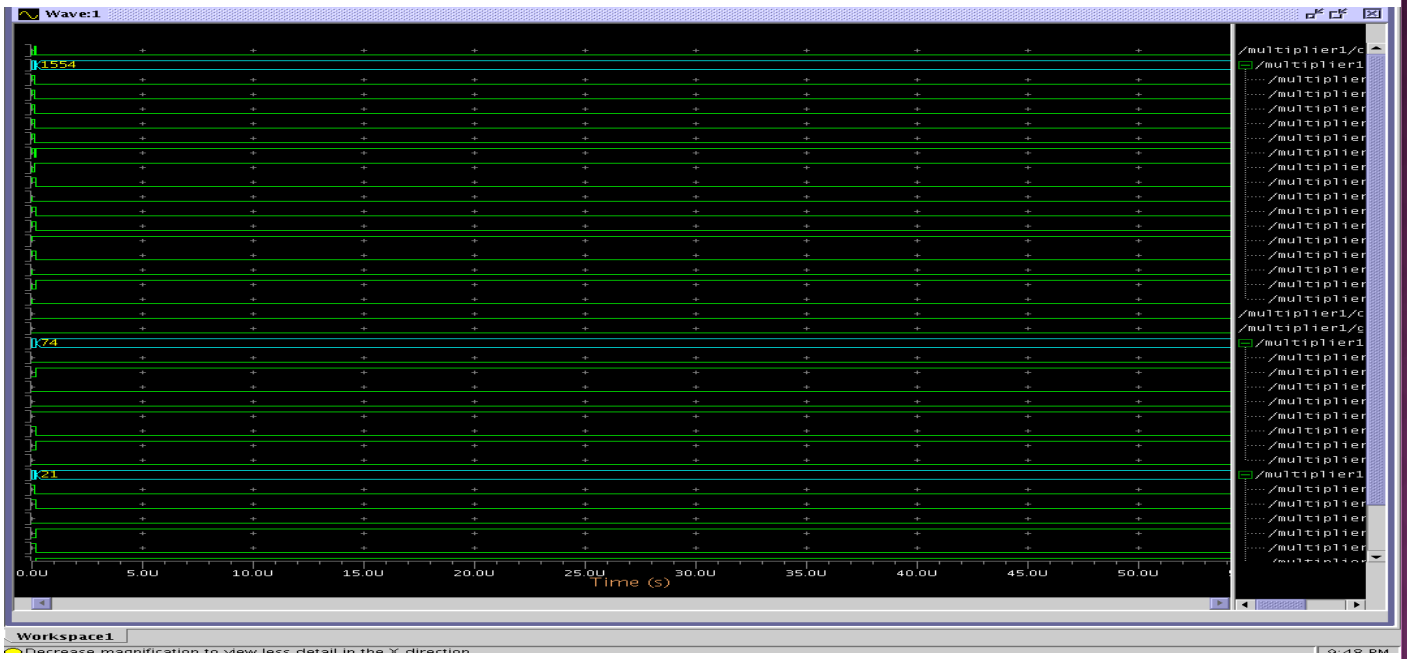
- Full system-schematic



FULL SYSTEM DESIGN

- Digital simulation

Positive

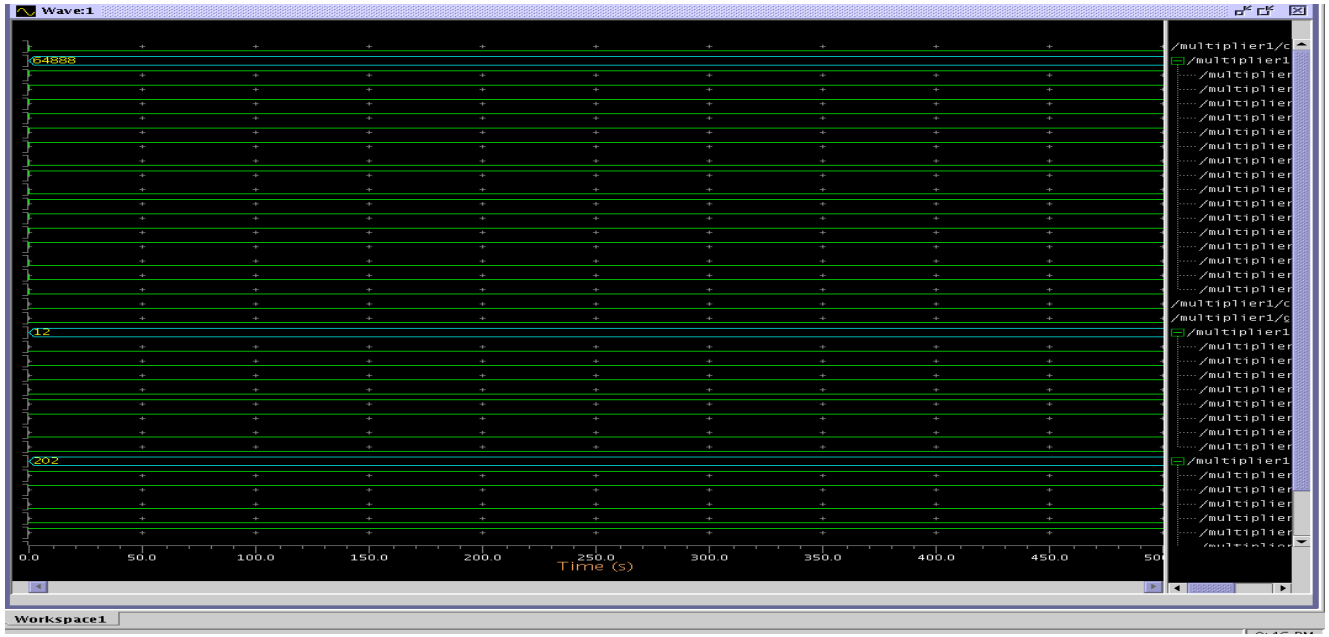


Multiplier:21
Multiplicand:74
Product:1554

FULL SYSTEM DESIGN

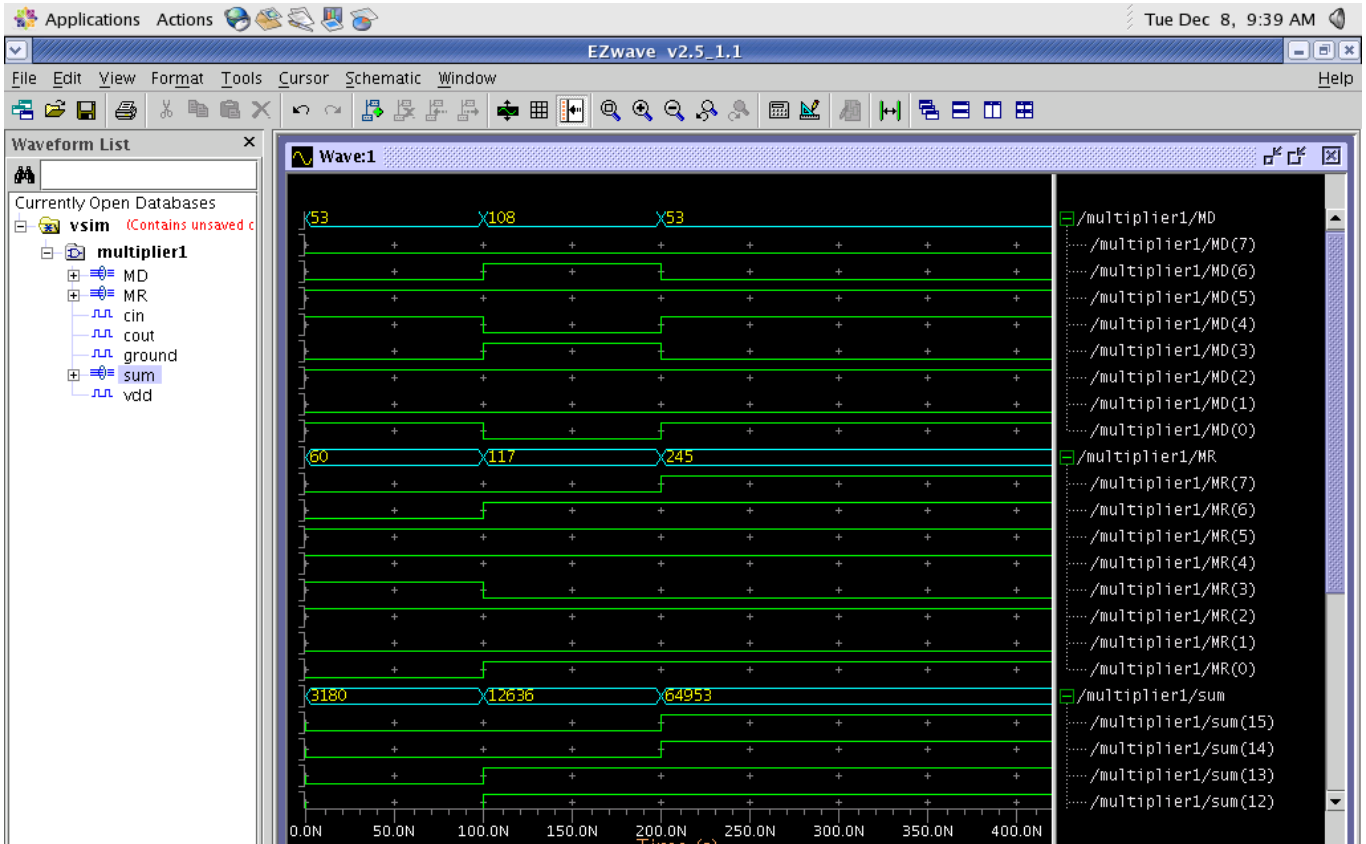
- Digital simulation

Minus



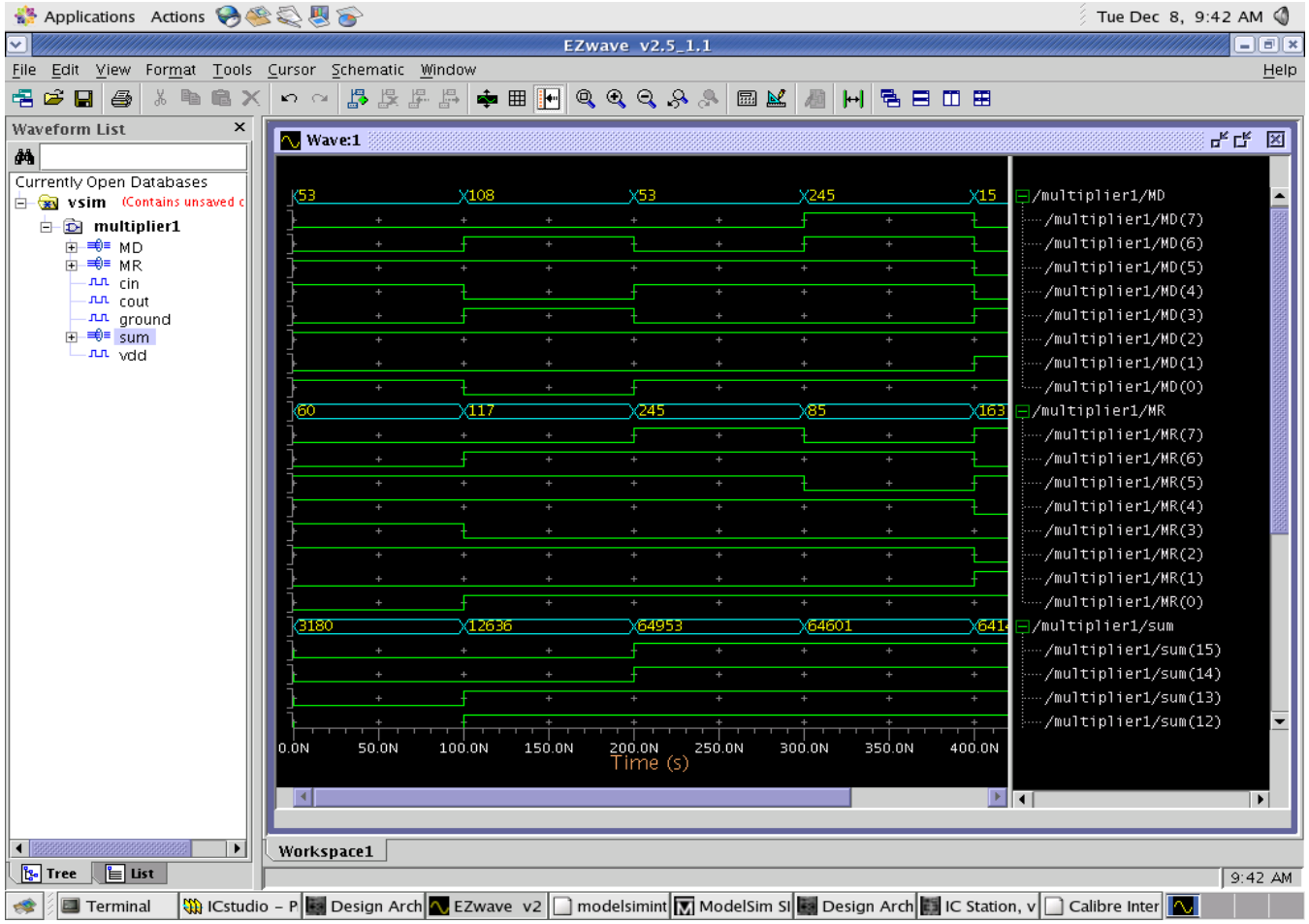
Multiplier:12
Multiplicand: -54(202-2^8=-54)
Product: -648(64888-2^16=-648)

FULL SYSTEM DESIGN



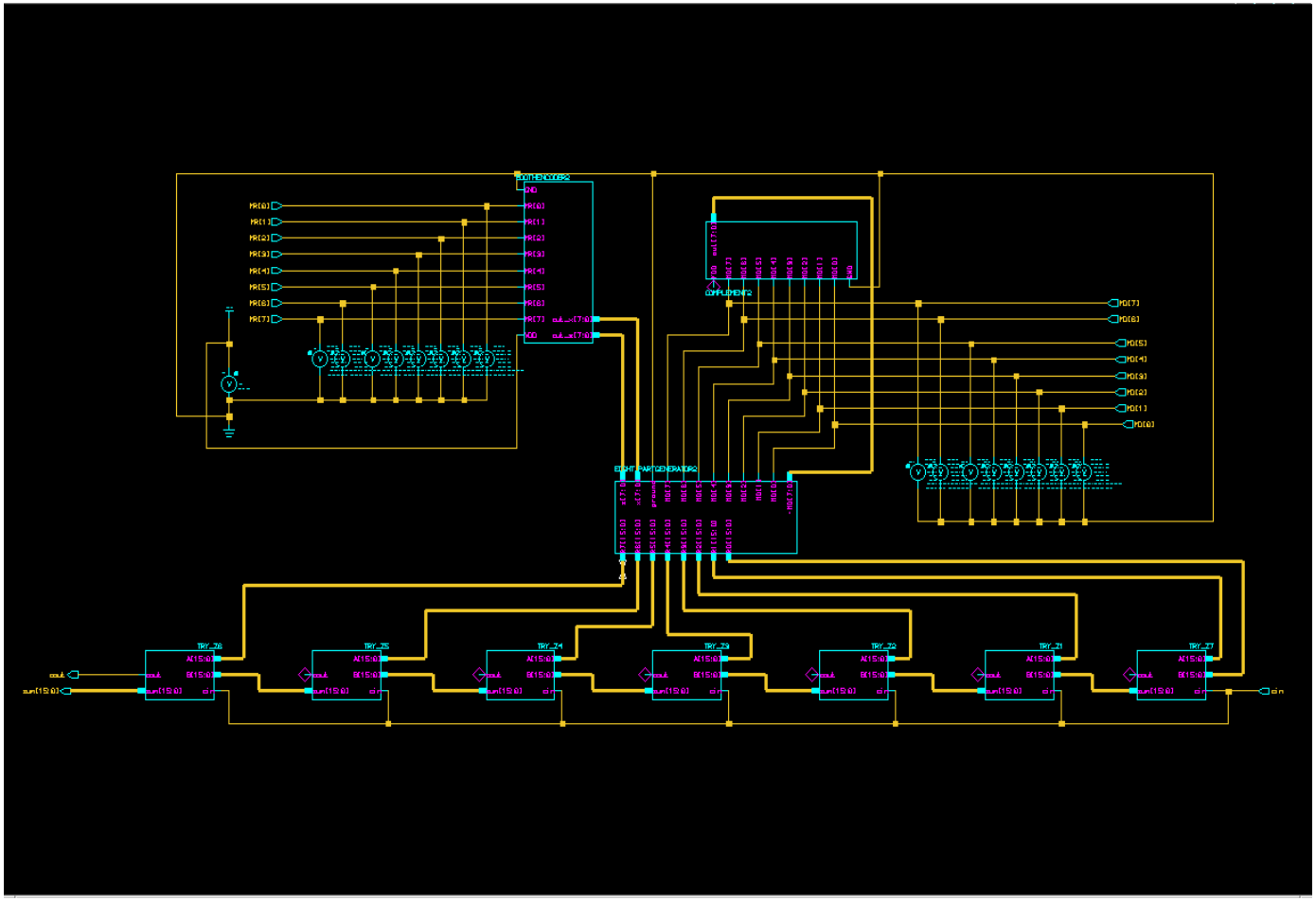
FULL SYSTEM DESIGN

- Digital simulation



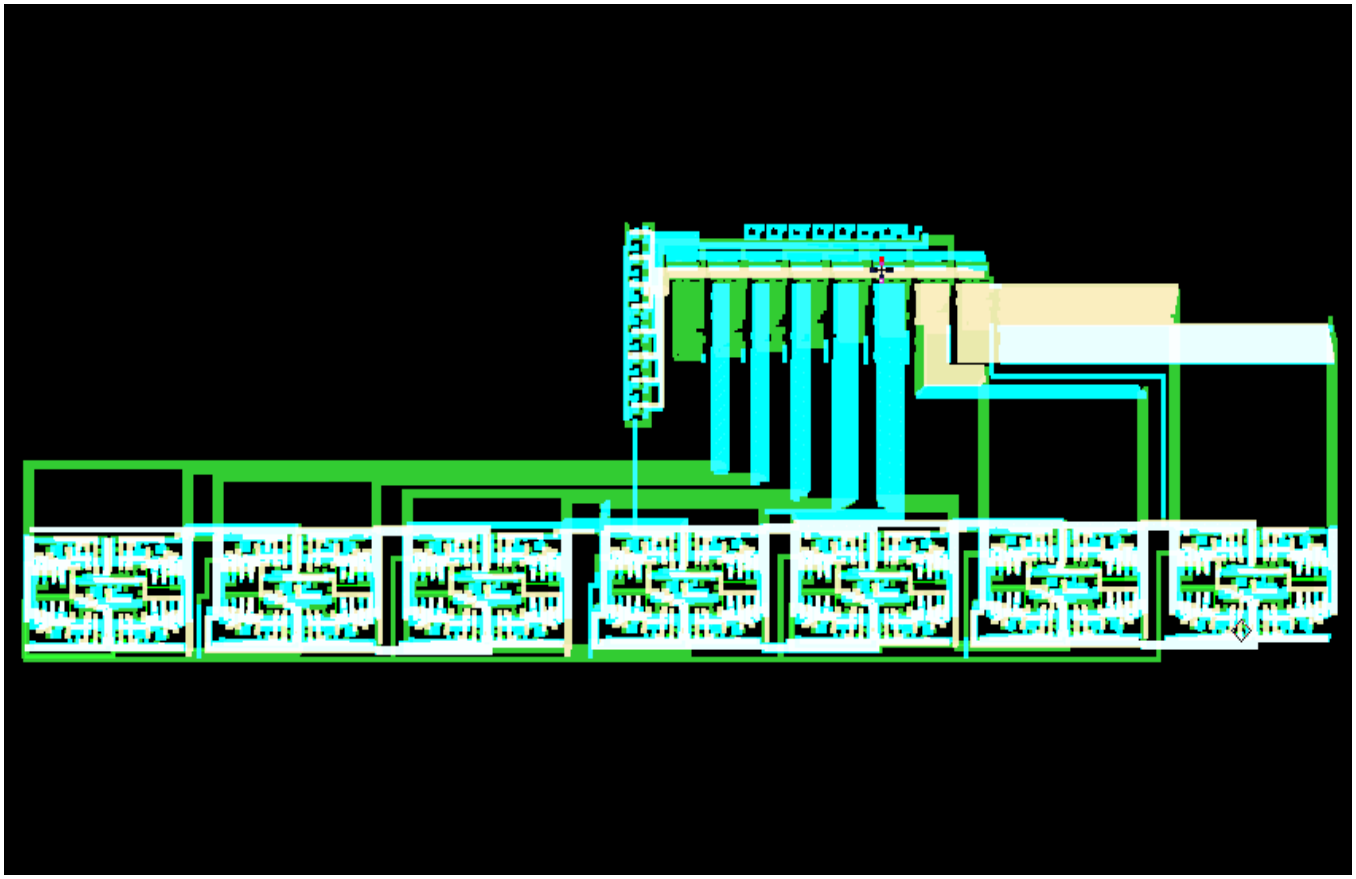
FULL SYSTEM DESIGN

- Analog simulation



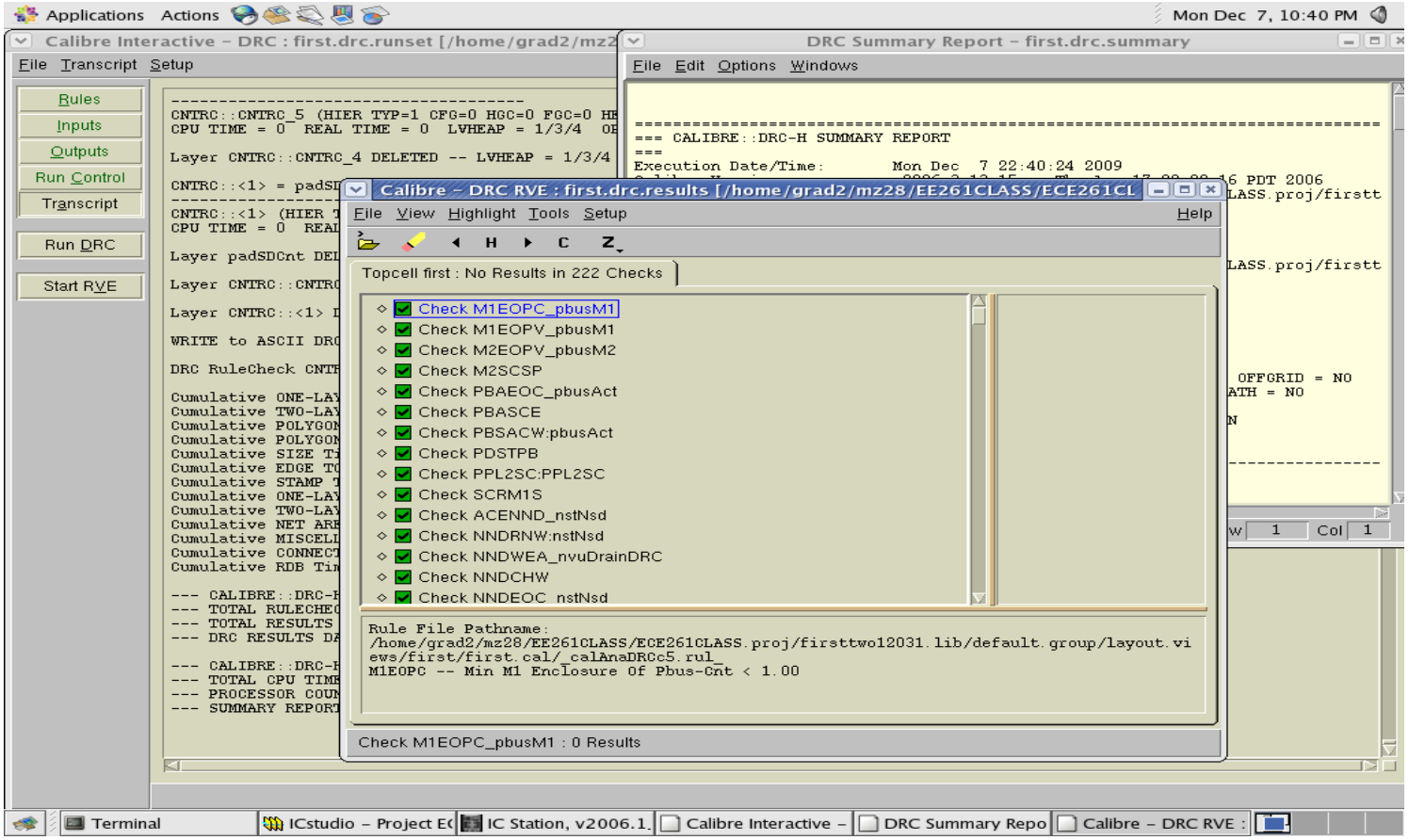
FULL SYSTEM DESIGN

- Full system-layout



FULL SYSTEM DESIGN

- DRC Check



FULL SYSTEM DESIGN

- LVS Check