SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

TIRUCHIRAPPALLI

# FPGA-BASED DESIGN AND IMPLEMENTATION OF AN (8:2) COMPRESSOR FOR OPTIMIZED 8X8 MULTIPLICATION

## [Industry, Innovation and Infrastructure (I3) ]


**Team Members**

Krithika M
(RA2211004050021)
Suba Lakshmi R
(RA2211004050048)




**Guided by,**

Dr. S Aditya
Assistant Professor
ECE Dept

# *ABSTRACT*


- **Efficient Multiplier Design** –Our proposed design is a FPGA-based approximate multiplier using a novel (8:2) compressor architecture for improved speed, power efficiency, and resource utilization.

- **Compressor Architecture –** We are utilizing (4:2) and (4:3) compressors in the first stage and a (5:2) compressor in the second stage to optimize delay, power, and area.

- **FPGA Implementation –** It will be designed, simulated, and verified using Xilinx Vivado, and implemented on the Artix-7 FPGA board.

- **Performance & Impact** –Our design ensures energy-efficient, high-speed computing with scalability and hardware-aware optimization for next-generation FPGA systems.

# *INTRODUCTION*

- The growing demand for high-performance computing in applications like image processing and real-time systems has led to the adoption of approximate computing techniques. These techniques help balance speed, power efficiency, and resource utilization, making them essential for modern digital systems.

- Traditional multipliers struggle with power consumption and delay, making it necessary to explore optimized designs for low-power, high-speed computations.

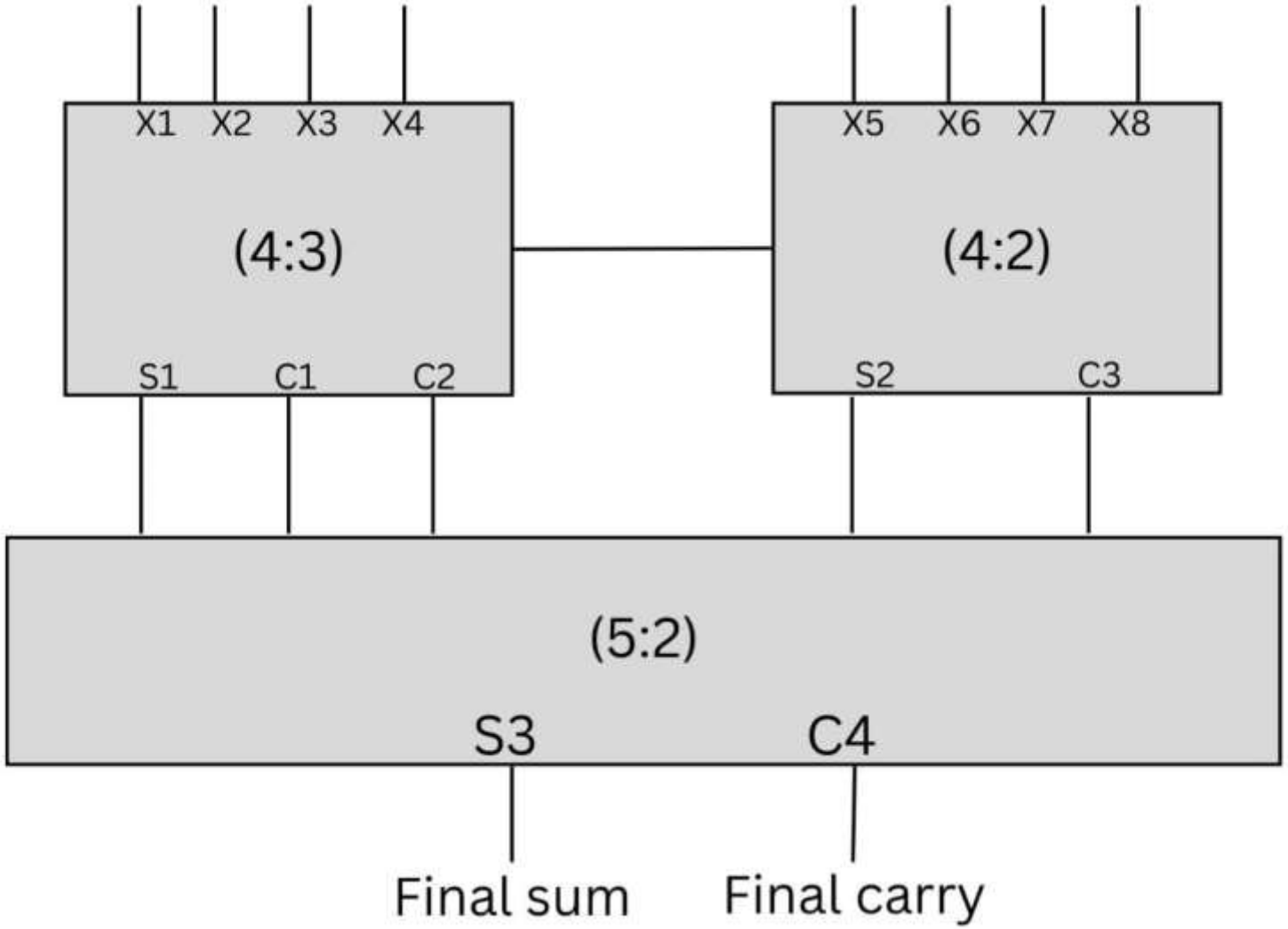
- To improve multiplier efficiency, compressors like (4:2), (4:3), and (5:2) are used to reduce the number of partial products. The (8:2) compressor architecture further enhances performance by minimizing computational complexity while maintaining acceptable accuracy.

- This work presents an (8:2) compressor-based multiplier designed for high-speed and low-power applications. The architecture is implemented and validated on the Artix-7 FPGA, where its performance is evaluated in terms of delay, power consumption, and hardware utilization.

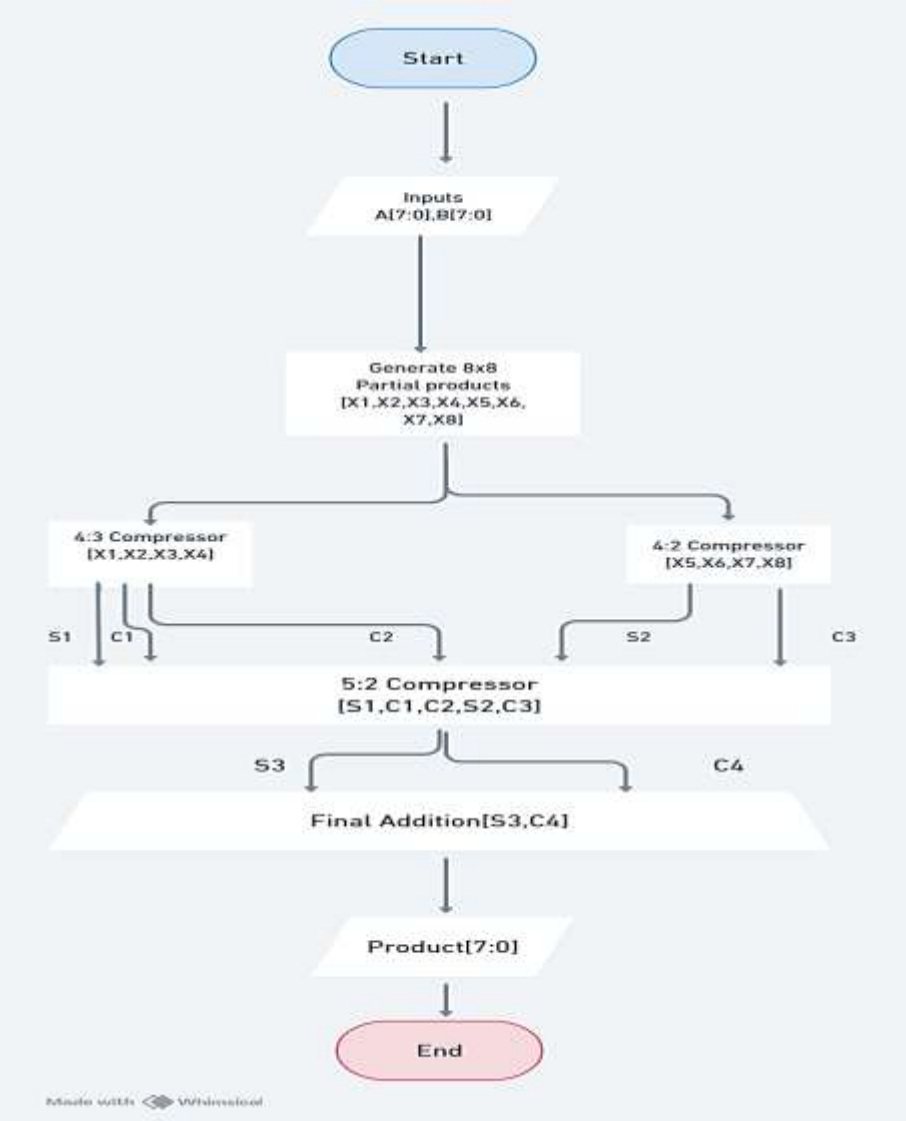
# ALGORITHM

- Partial Product Generation
- Compression using 8:2compressor
- Final Summation
- FPGA Implementation & Verification

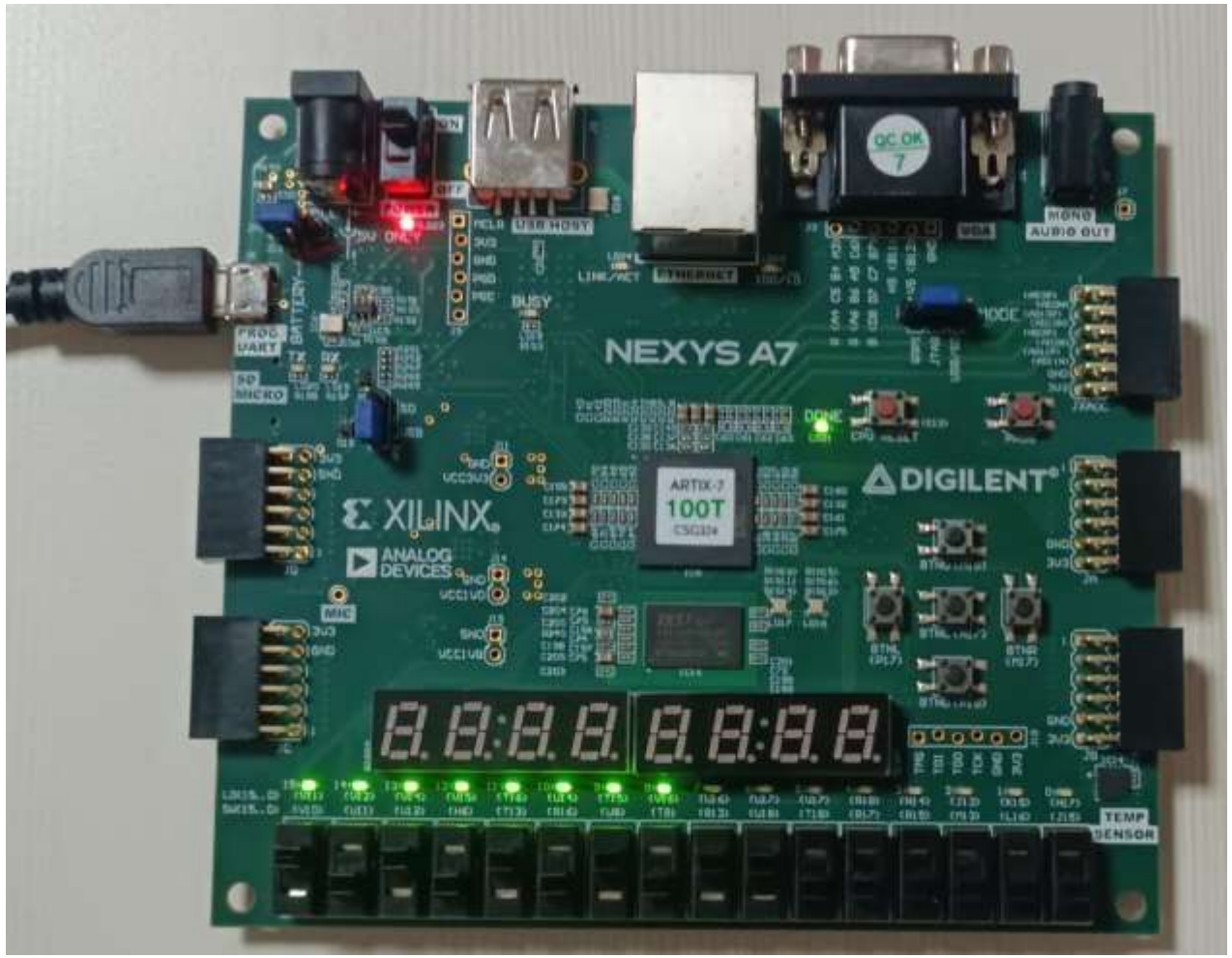
# BLOCK DIAGRAM



# FLOWCHART


Start
Inputs
A[7:0],B[7:0]
Generate 8x8
Partial products
[X1,X2,X3,X4,X5,X6,
X7,X8]
4:3 Compressor
[X1,X2,X3,X4]
4:2 Compressor
[X5,X6,X7,X8]
S1
C1
C2
S2
C3
5:2 Compressor
[S1,C1,C2,S2,C3]
S3
C4
Final Addition[S3,C4]
Product[7:0]
End
Made with Whimsical

# RESULT


NEXYS A7
XILINX
DIGILENT
ANALOG DEVICES
ARTIX-7
100T
QC OK
7
MONO AUDIO OUT
USB HOST
ETHERNET
PROG UART
SD MICRO
TEMP SENSOR

# RESULT TEST CASES

| TEST CASES | INPUT A[8 bit] | INPUT B[8 bit] | COMPRESSOR STAGE OUTPUT | FINAL PRODUCT |
|---|---|---|---|---|
| 1 | 00001111 (15) | 00000011 (3) | S = 45, C = 0 | 45 |
| 2 | 00010101 (21) | 00000101 (5) | S = 105, C = 0 | 105 |
| 3 | 11110000 (240) | 00001111 (15) | S = 3600, C = 0 | 3600 |
| 4 | 11111111 (255) | 11111111 (255) | S = 65025, C = 0 | 65025 |
| 5 | 00011000 (24) | 00000010 (2) | S = 48, C = 0 | 48 |
| 6 | 10000000 (128) | 00000001 (1) | S = 128, C = 0 | 128 |

# THANK YOU !