EENG 5560 FINIAL PROJECT Report

Name: Kavya Sree Maddikara

Joshua Sannareddy

Due: 05/02/2024

Table of Contents

Design

Block diagrams

Overall design

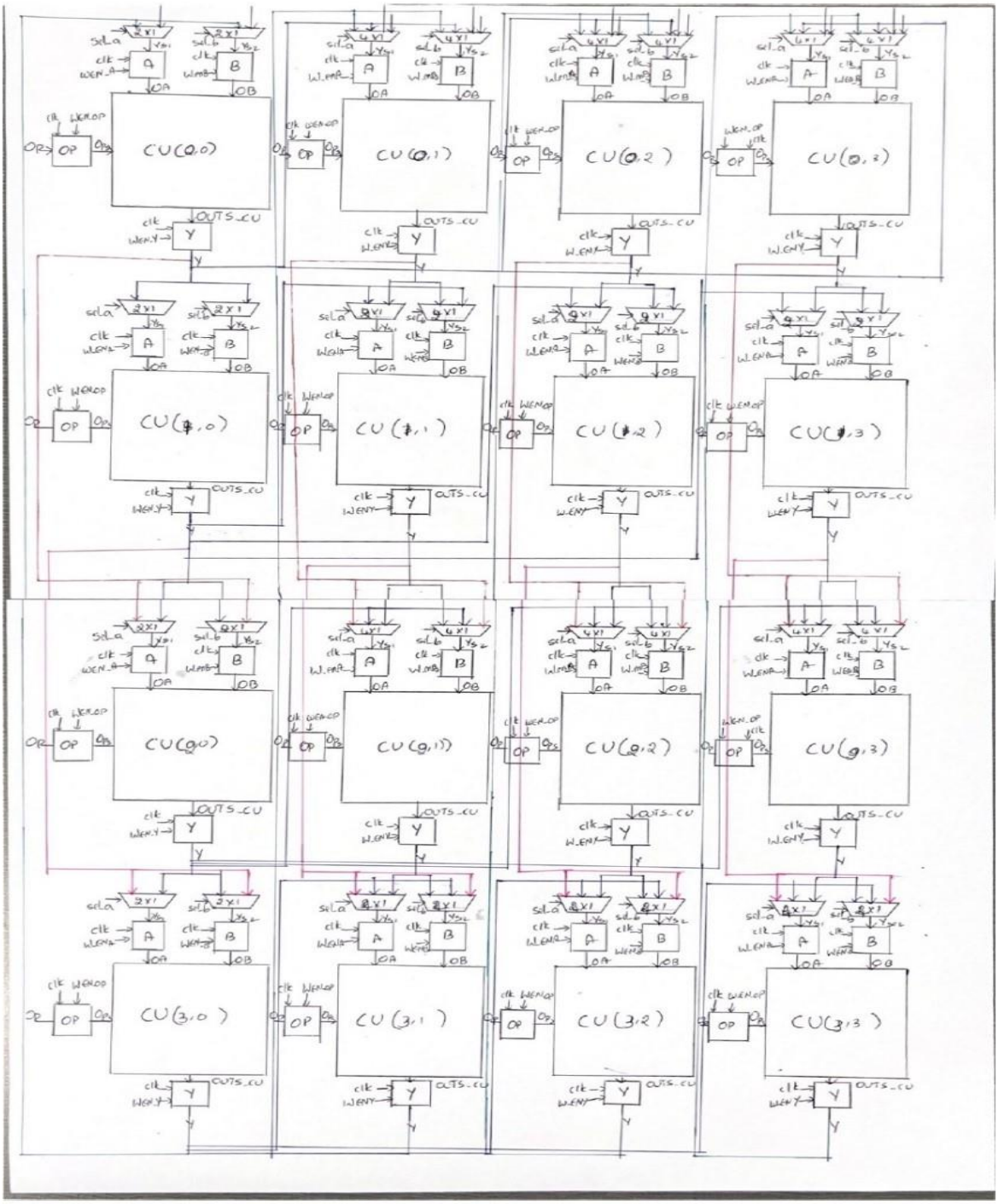


Figure 1- top module: CU with overall ports, subcomponents and intermediate signals shown

Overall component: All CUs.

Parameters: d_w – 1 (for inputs and outputs)

Rows-number of rows=4

Cols-number of columns=4

Input ports:

Port name	Bit width	Purpose
A,B	d_w = 4	1D arrays Data inputs
Sel_a, Sel_b	3	2D arrays Selection line for A and B
W_enA,W_enB, W_enY,W_enOP	1	2D arrays containing write/read values for all storage units
CLK	1	Clock signal
Op	5	2D array op code

Output ports:

Port name	Bit width	Purpose
C	4	2D array outputs of all CU
Y	d_w = 4	1D array of all last row CU

Necessary intermediate signals:

Port name	Bit width	Purpose
OUTS_CU	d_w = 4	2D array of Outputs of CU to input of storage
OY	4	2D array output of storage y to input of muxes
OA	4	2D array output of storage A to input of CU A
OB	4	2D array output of storage B to input of CU B
OPS	5	2D array of CU op codes
Ys1,Ys2	4	2D array of outputs of Z of muxes to inputs of storage A and B
K, L, M, N	4	2D arrays of all 4x1 muxes
I, J	4	2D arrays of all 2x1 muxes

Subcomponents

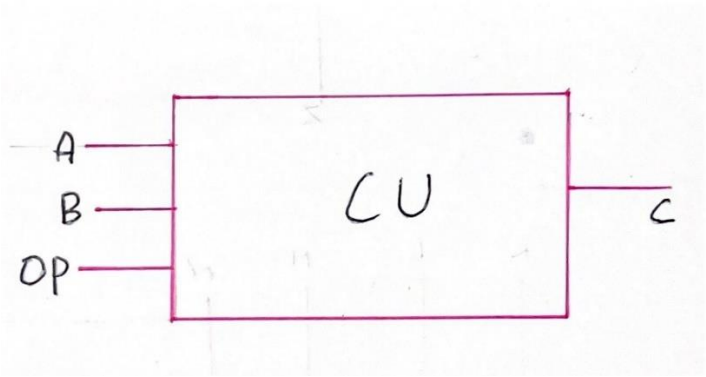


Figure 2- CU with overall ports

Subcomponent: CU

Input ports:

Port name	Bit width	Purpose
A,B	d_w = 4	Data inputs
Op	5	Op code

Output ports:

Port name	Bit width	Purpose
C	d_w = 4	Data output

Necessary intermediate signals:

Port name	Bit width	Purpose
Temp	8	Stores full length multiplication

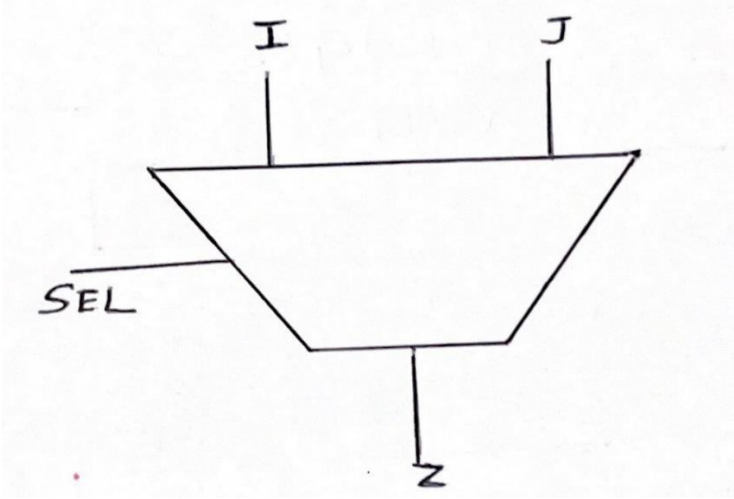


Figure 3- MUX 2:1 with overall ports

Subcomponent: MUX 2:1

Input ports:

Port name	Bit width	Purpose
I,J	d_w = 4	Data inputs
SEL	3	Select line, selects data inputs to send to data output

Output ports:

Port name	Bit width	Purpose
Z	d_w = 4	Data output

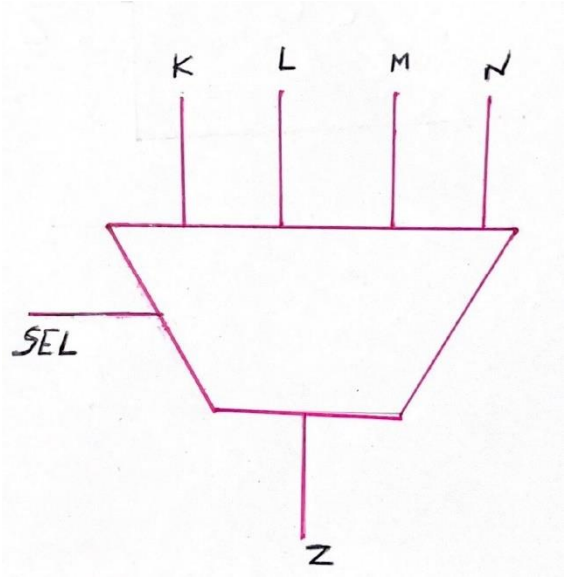


Figure 4-MUX 4:1 with overall ports

Subcomponent: MUX 4:1

Input ports:

Port name	Bit width	Purpose
K,L,M,N	d_w = 4	Data inputs
SEL	3	Select line, selects data inputs to send to data output

Output ports:

Port name	Bit width	Purpose
Z	d_w = 4	Data output

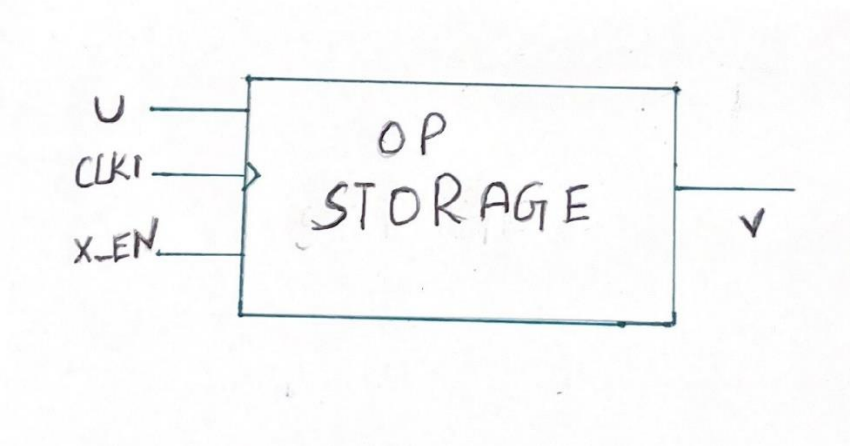


Figure 5- GATE OP STORAGE with overall ports

Subcomponent: Gate Op Storage

Input ports:

Port name	Bit width	Purpose
U		Data inputs(gate_oper)
Clk1	1	Clock signal
X_EN	1	Read/Write

Output ports:

Port name	Bit width	Purpose
V		Data output(gate_oper)

Necessary intermediate signals:

Port name	Bit width	Purpose
Stored_data		Stores data input if w_en=1

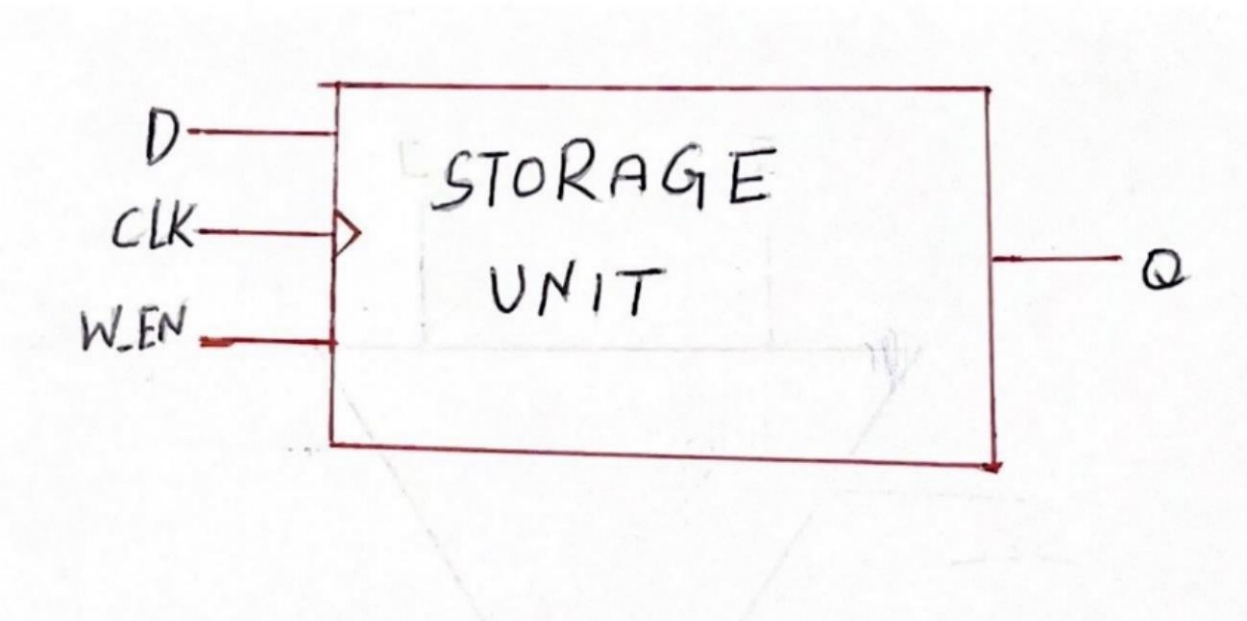


Figure 6-STORAGE UNIT with overall ports

Subcomponent: Storage unit

Input ports:

Port name	Bit width	Purpose
D	4	Data inputs
Clk	1	Clock signal
W_EN	1	Read/Write

Output ports:

Port name	Bit width	Purpose
Q	4	Data output

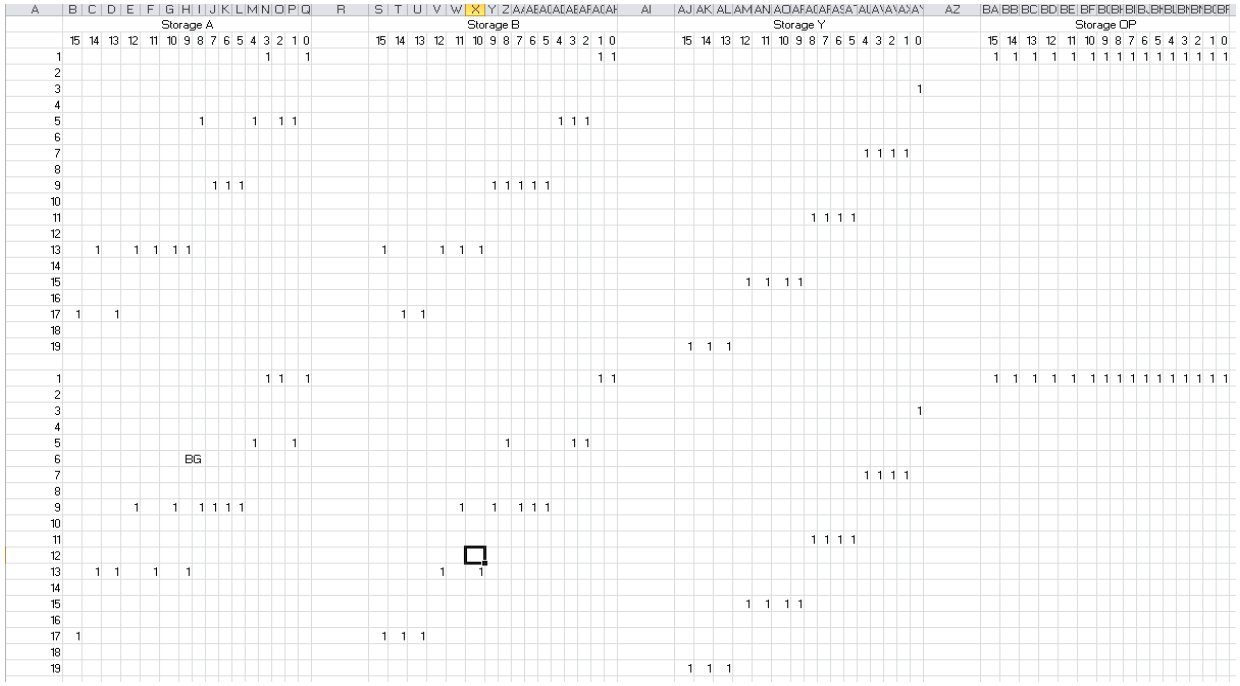
Necessary intermediate signals:

Port name	Bit width	Purpose
Stored_data	4	Stores data input if W_en=1

Design explanation

Functionality

This design aims to use virtualization technology, which enables an 8x4 structure to operate on a 4x4 framework. To achieve this, feed the final row back into the first row as though it were entering a fresh set of four rows. This design also makes advantage of multi-level vertical and horizontal interconnection, necessitating the employment of storage units. Communication between CUs in the same row is made possible by horizontal connectivity. With multi-level vertical connectivity, a CU in one row can communicate with CUs one or more rows below it. Because two CUs in a row will operate at the same clock signal, storage units are necessary for both horizontal and vertical connection; As a result, they are unable to use their neighbor's output as one of their own. In a similar vein, input A of a CU may originate from three rows above, whereas input B may originate from a single row above in a multi-level vertical connection. As a result, if every CU uses the same clock, the CU will operate logically incorrectly since it will use known operand A and undefined operator B. In order for operand A to be received at time t0, operand B to be received at time t0+t = t1, and for the CU to execute the computation at time t1 + t, storage units introduce a delay. The storage units are configured to do two actions in response to the clock signal's rising edge: writing to storage when enable = 1 and reading from storage when enable = 0. The graphic below illustrates how to determine the read/write order for each storage unit using a timing diagram. The following graphic displays four sets of columns: A storage, B storage, Y storage, and OP storage. There are sixteen units per every set of columns, numbered fifteen through zero. The cycle number, with a 10 ns cycle per cycle, is shown in Column A. The storage unit is receiving input D and writing it for later use whenever there is a 1. The storage unit's output Q is whatever it stored since a 0 indicates a where a 1 is implied. A few cycles ahead of the B, certain CUs will receive the A input, and vice versa. Columns I and AY both display this. In cycle 5, storage A (2,0) receives the payload from store Y (0,0), whose output is ready in cycle 3. But storage Y (2,0) does not have any data to store until cycle 11 since storage B (2,0) does not receive any input until cycle 9. I utilized this timing diagram to get my findings.



Select input	operation
ANDo	AND
ORo	OR
NANDo	NAND
NORo	NOR
XORo	XOR
XNORo	XNOR
addo	ADDITION
subo	SUBTRACTION
multo	MULTIPLICATION
gthano	GREATHER THAN
lthano	LESS THAN
equalo	EQUAL
geqo	GREATHER THAN EQUAL
leqo	LESS THAN EQUAL
neqo	NOT EQUAL
aslo	ARITHMETIC SHIFT LEFT
asro	ARITHMETIC SHIFT RIGHT
rslo	ROTATE SHIFT LEFT
rsro	ROTATE SHIFT RIGHT
lslo	LOGIC SHIFT LEFT
lsro	LOGIC SHIFT RIGHT

Results

Generated Schematics

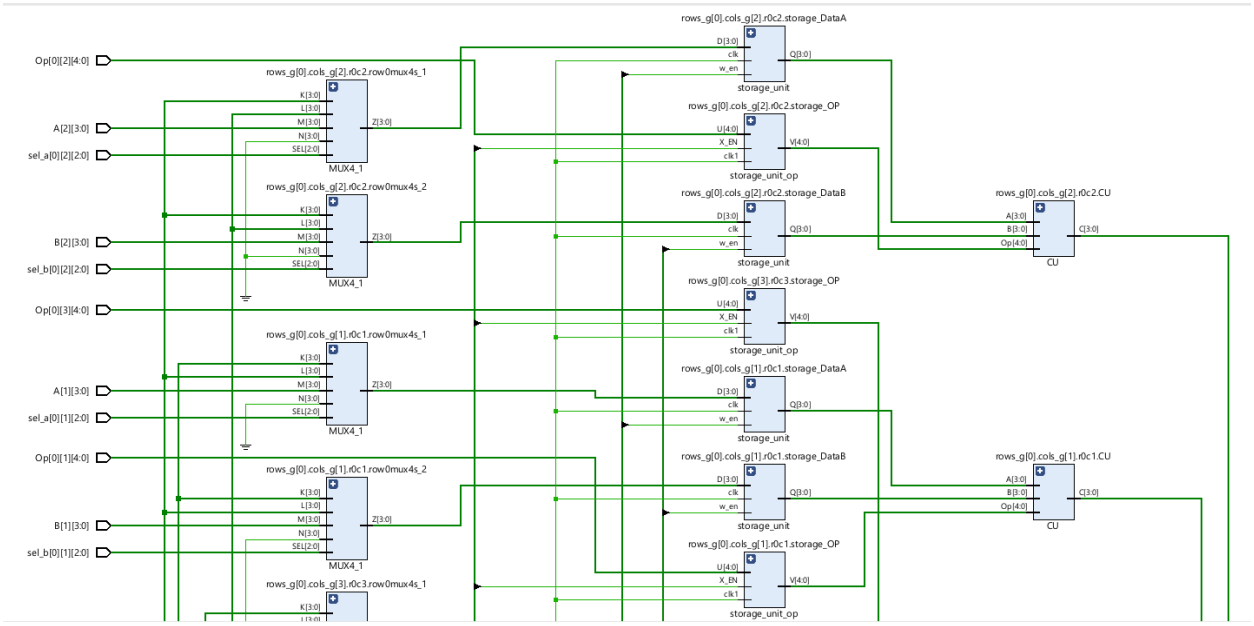


Figure 7- Computational unit RTL schematic with overall ports

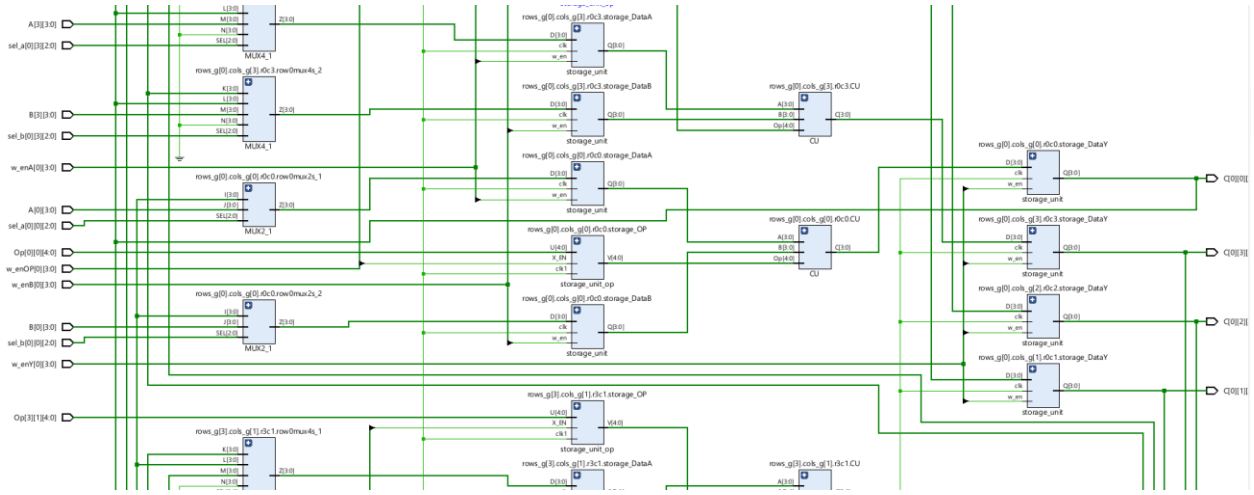


Figure 8- Computational unit RTL schematic with overall ports

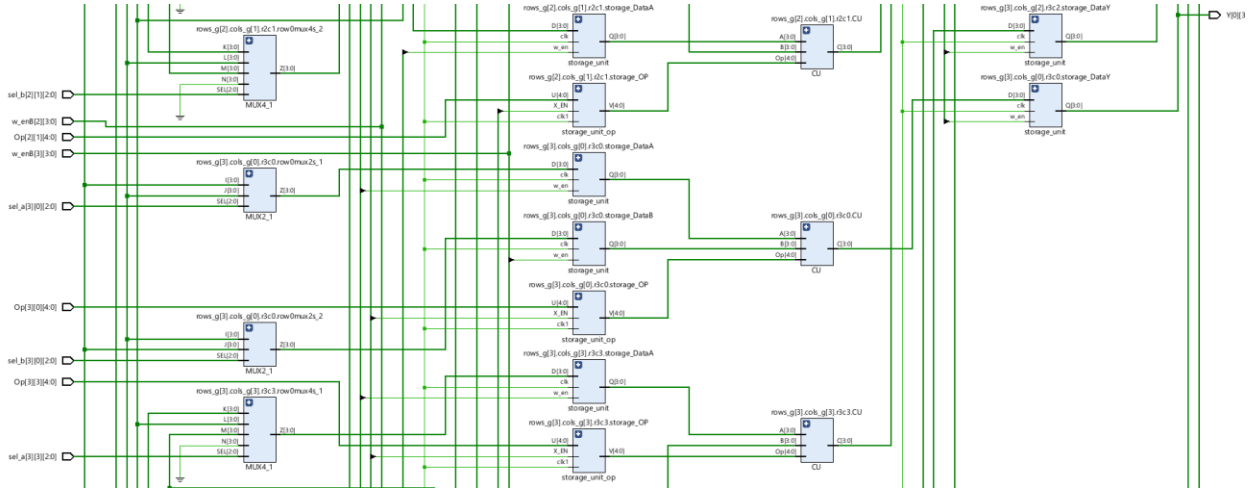


Figure 9- Computational unit RTL schematic with overall ports

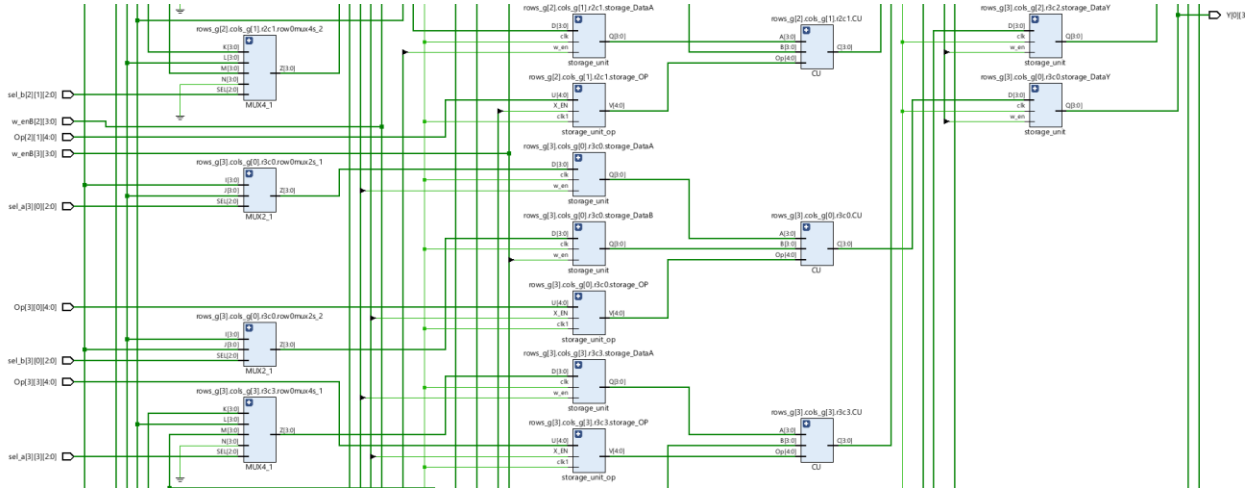


Figure 10- Computational unit RTL schematic with overall ports

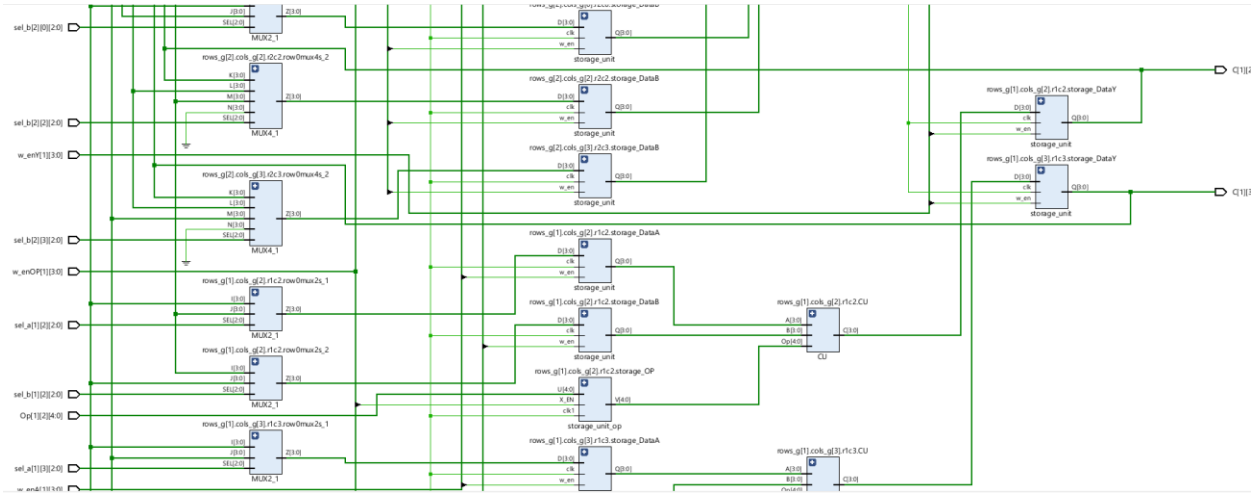


Figure 11- Computational unit RTL schematic with overall ports

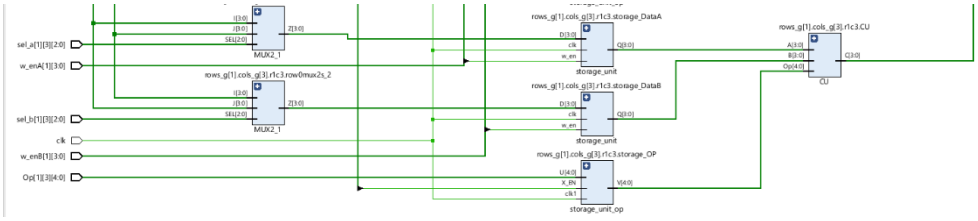


Figure 12- Computational unit RTL schematic with overall ports

Waveforms

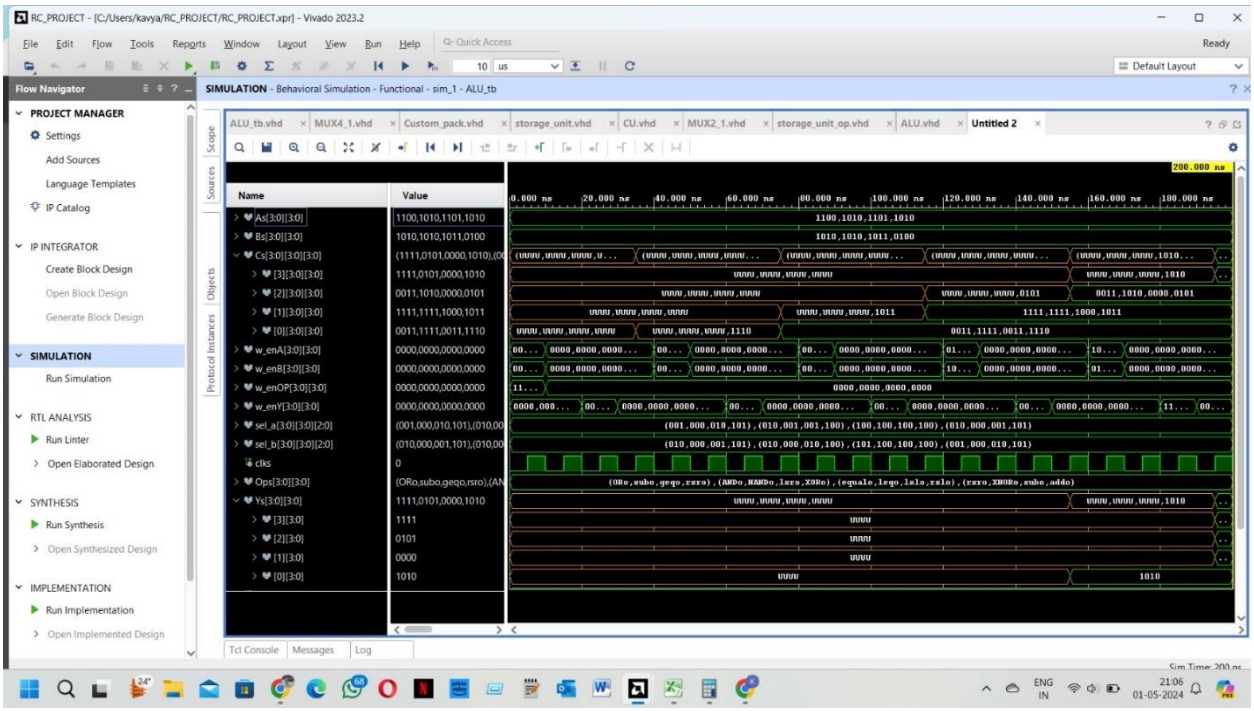


Figure 13- Waveform for first run

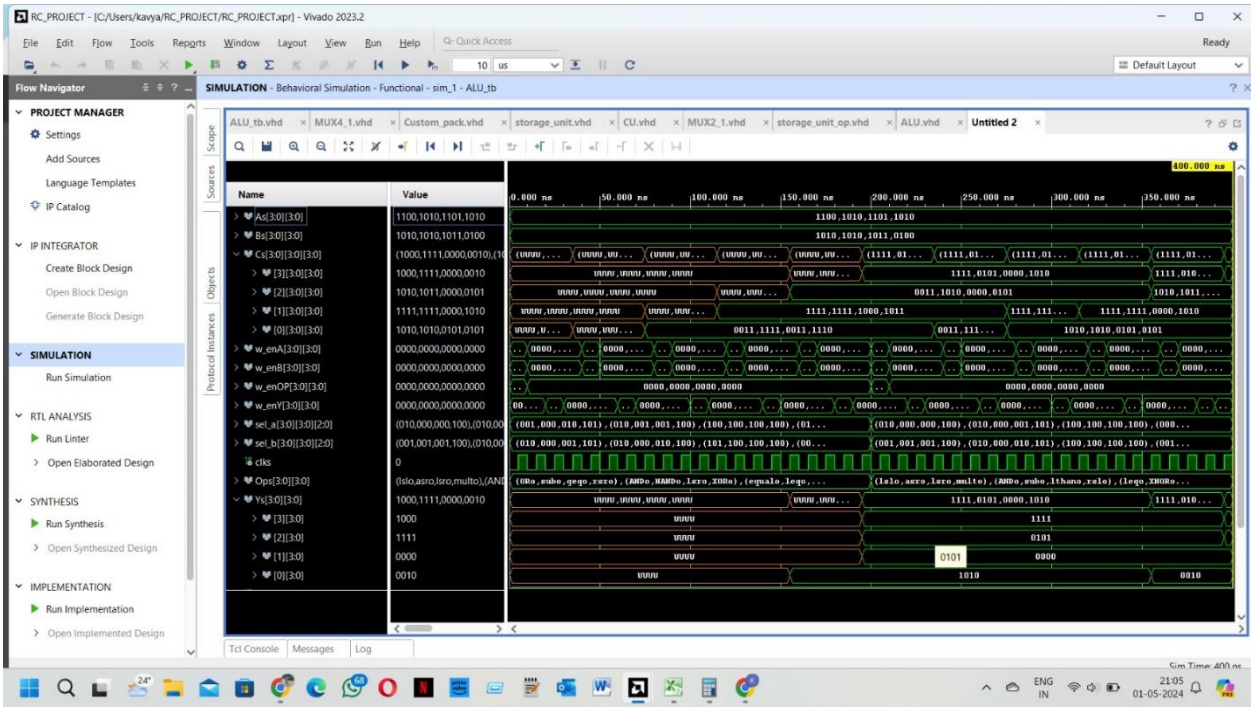


Figure 14- Waveform for first run

Table/Calculations

Overall Design

Test Case 1 :

CU#	Source A	Source B	A	B	Oper	Calculated Op	Simulated Op	Match
CU (0,0)	EXTERNAL	EXTERNAL	1010	0100	ADD	1110	1110	Yes
CU (0,1)	CU (0,0)	EXTERNAL	1101	1011	SUB	0011	0011	Yes
CU (0,2)	CU (0,0)	CU (0,0)	1010	1010	XNOR	1111	1111	Yes
CU (0,3)	EXTERNAL	CU (0,0)	1100	1010	RSR	0011	0011	Yes
CU (1,0)	CU (0,0)	CU (0,0)	1110	1110	ROL	1011	1011	Yes
CU (1,1)	CU (1,0)	CU (0,1)	1011	0011	LSL	1000	1000	Yes
CU (1,2)	CU (1,0)	CU (0,2)	1011	1111	LTE	1111	1111	Yes
CU (1,3)	CU (1,0)	CU (1,0)	1011	1011	EQ	1111	1111	Yes
CU (2,0)	CU (0,0)	CU (1,0)	1110	1011	XOR	0101	0101	Yes
CU (2,1)	CU (2,0)	CU (0,1)	0101	0011	LSR	0000	0000	Yes
CU (2,2)	CU (2,0)	CU (1,2)	0101	1111	NAND	1010	1010	Yes
CU (2,3)	CU (1,3)	CU (0,3)	1111	0011	AND	0011	0011	Yes
CU (3,0)	CU (2,0)	CU (1,0)	0101	1011	RSR	1010	1010	Yes
CU (3,1)	CU (2,1)	CU (3,0)	0000	1010	GTE	0000	0000	Yes
CU (3,2)	CU (1,2)	CU (2,2)	1111	1010	SUB	0101	0101	Yes
CU (3,3)	CU (3,0)	CU (1,3)	1010	1111	OR	1111	1111	Yes

Test Case 2:

CU#	Source A	Source B	A	B	Oper	Calculated Op	Simulated Op	Match
CU (0,0)	CU (3,0)	CU (3,0)	1010	1010	NOR	0101	0101	Yes
CU (0,1)	CU (0,0)	CU (3,1)	0101	0000	XOR	0101	0101	Yes
CU (0,2)	CU (3,2)	CU (0,0)	0101	0101	NAND	1010	1010	Yes
CU (0,3)	CU (3,3)	CU (0,0)	1111	0101	SUB	1010	1010	Yes
CU (1,0)	CU (0,0)	CU (0,0)	0101	0101	ADD	1010	1010	Yes
CU (1,1)	CU (1,0)	CU (0,1)	0101	1010	EQ	0000	1000	Yes
CU (1,2)	CU (1,0)	CU (0,2)	1010	1010	XNOR	1111	1111	Yes
CU (1,3)	CU (1,0)	CU (0,3)	1010	1010	LTE	1111	1111	Yes
CU (2,0)	CU (1,0)	CU (0,0)	1010	0101	ROL	0101	0101	Yes
CU (2,1)	CU (2,0)	CU (0,1)	0101	0101	LT	0000	0000	Yes
CU (2,2)	CU (0,2)	CU (1,2)	1010	1111	SUB	1011	1011	Yes
CU (2,3)	CU (1,3)	CU (0,3)	1111	1010	AND	1010	1010	Yes
CU (3,0)	CU (1,0)	CU (2,0)	1010	0101	MULT	0010	0010	Yes
CU (3,1)	CU (1,1)	CU (3,0)	0000	0010	LSR	0000	0000	Yes
CU (3,2)	CU (1,2)	CU (3,0)	1111	0010	ASR	1111	1111	Yes
CU (3,3)	CU (2,3)	CU (3,0)	1010	0010	LSL	1000	1000	Yes

The above table shows the output of every CU which matches the signal C.