Matrix Multiplication Unit

DEVELOPED BY CADEN SANDERS

Project Overview

- 4 x 4 Matrix Math

- Matrix Addition
- Matrix Subtraction
- Scalar Multiplication
- Matrix Multiplication
- Matrix Transposition

- RAM

Project Goals

- Create a Verilog ecosystem of modules
- Model basic computer architecture

Assumptions

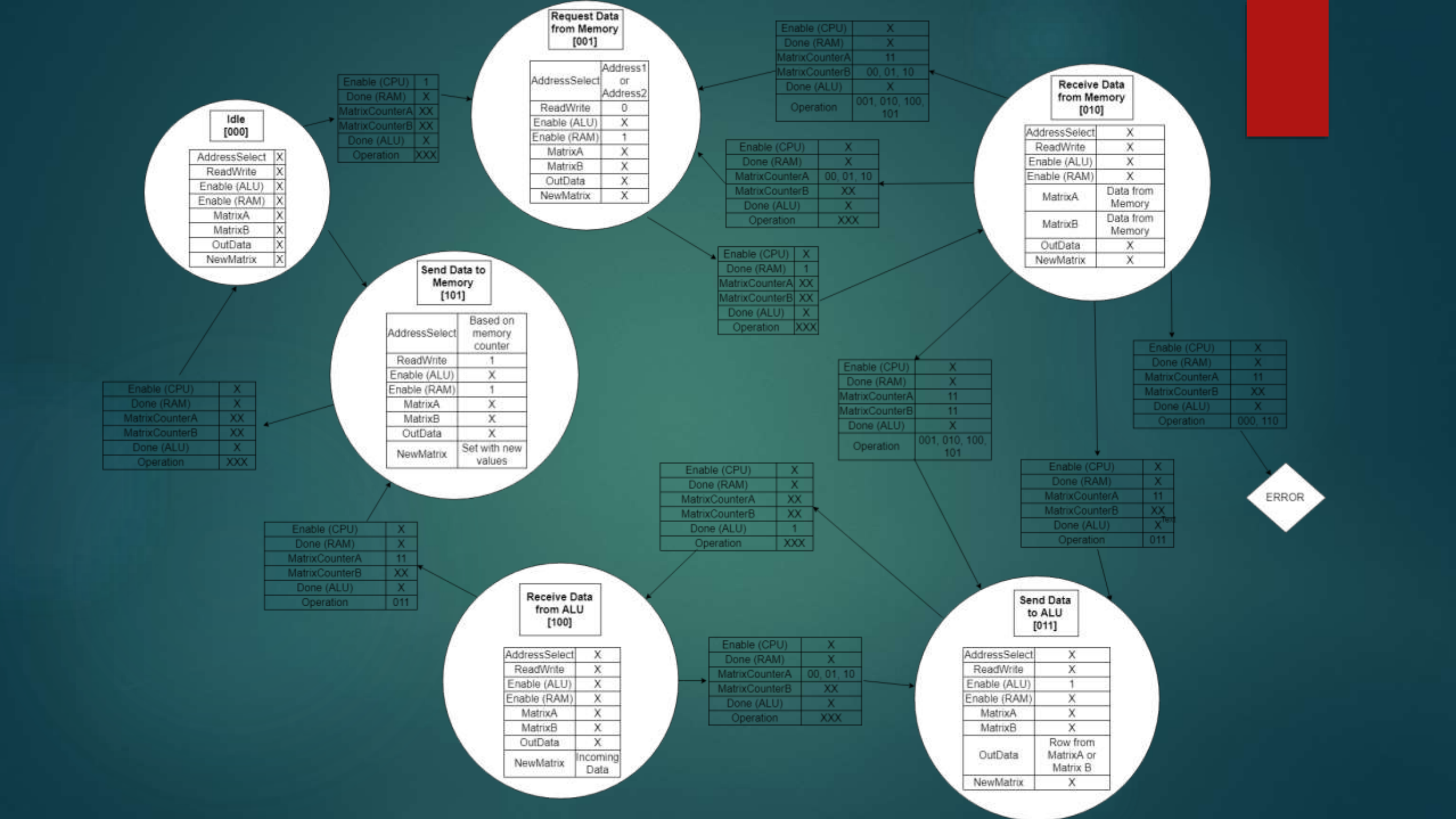
Memory

- Modeled after Random Access Memory (RAM)
- Memory will be volatile SRAM. Data will be stored only while the device has power. Memory will be latched into place and will not need to be "refreshed" to keep an electric charge.
- reg signed [31:0] Memory [3:0] [255:0]
- Each element in the memory is a 32-bit integer

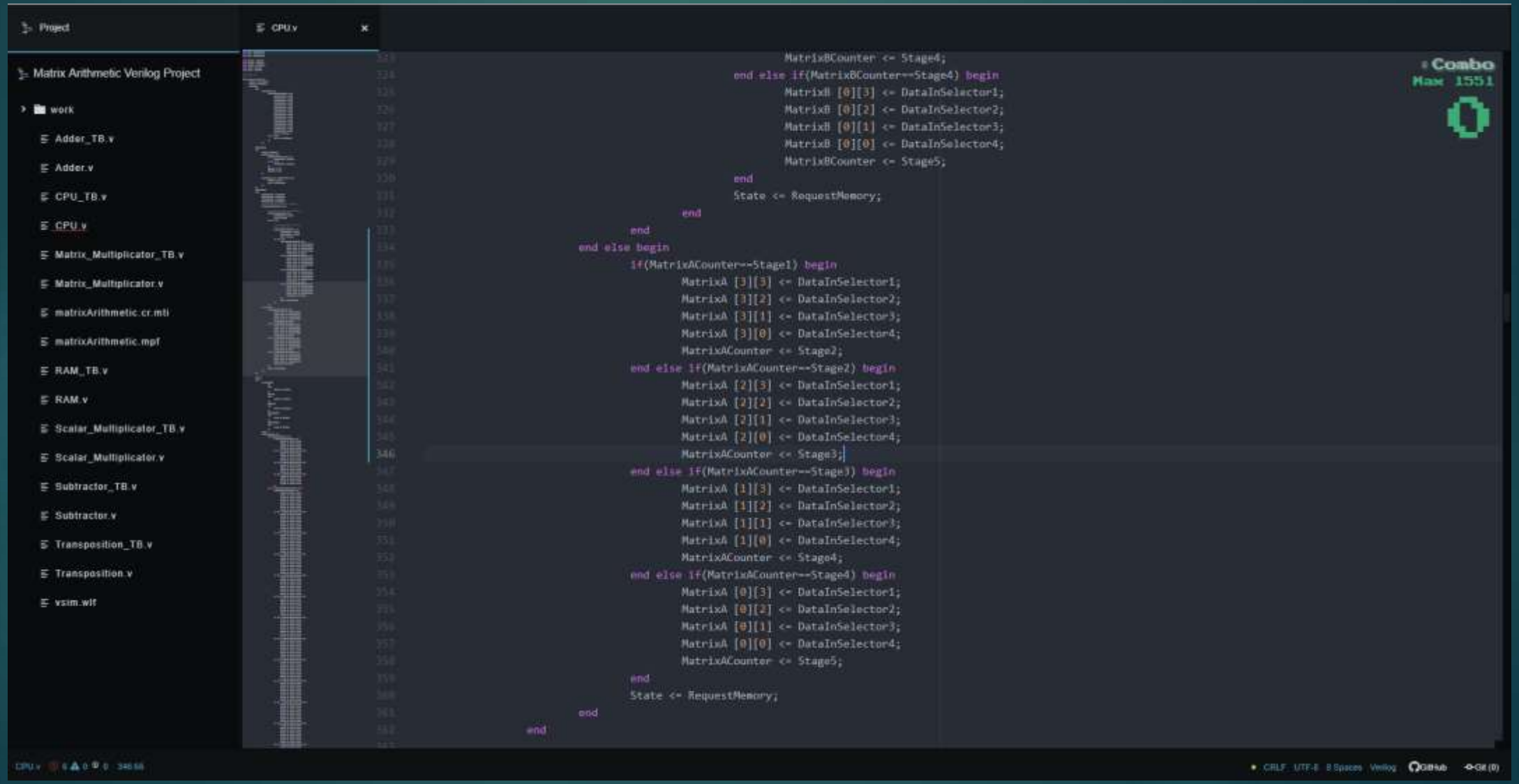
CPU

- The CPU is not independent and will be considered the top-level module.
- Control signals are sent from a test bench with “initial” procedural blocks for testing and control.
- Could be connected to an external system that has a functional operating system.

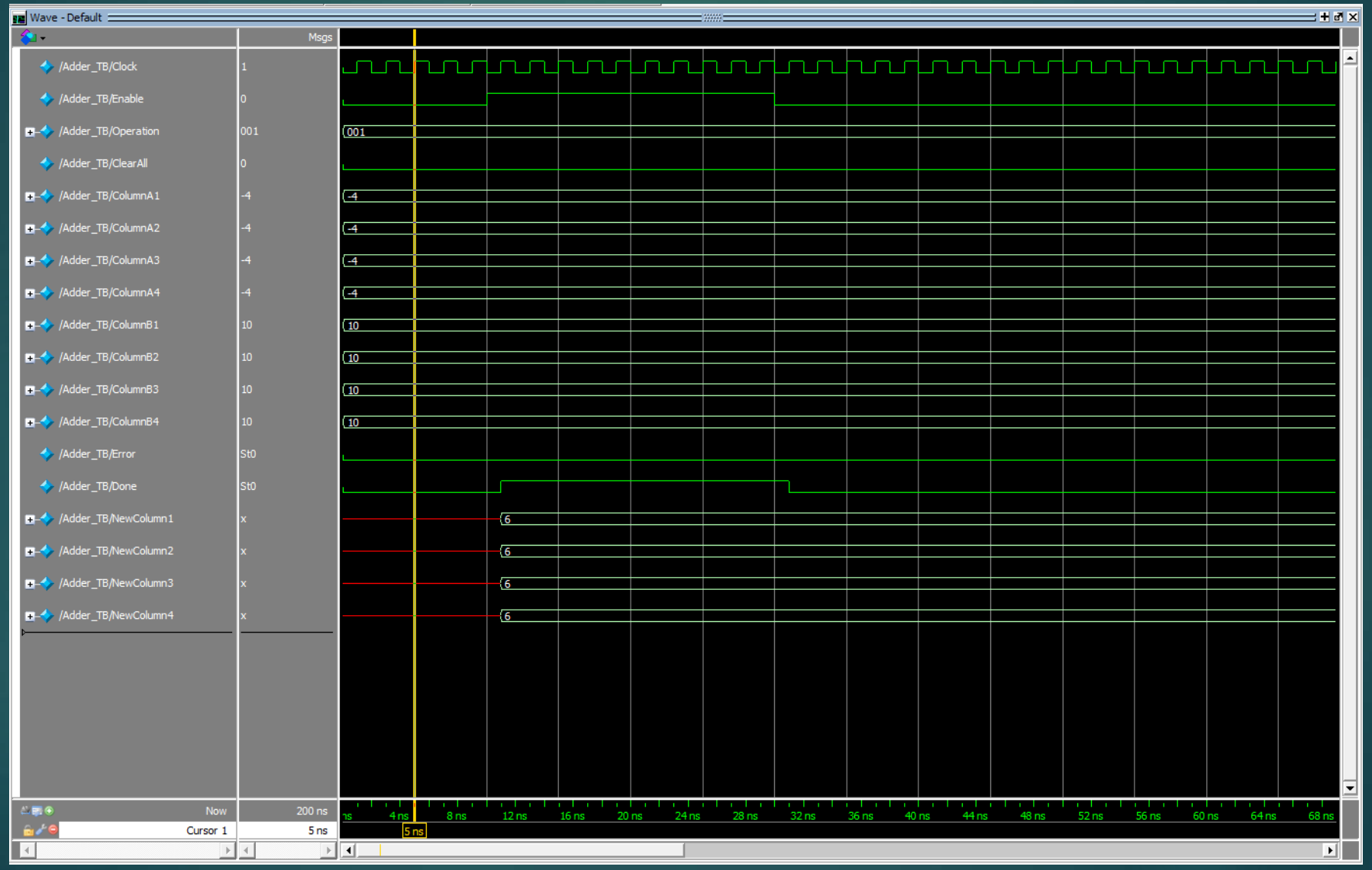

Adder.v
input wire Clock
input wire [2:0] Operation
input wire ClearAll
input wire signed [31:0] ColumnA1
input wire signed [31:0] ColumnA2
input wire signed [31:0] ColumnA3
input wire signed [31:0] ColumnA4
input wire signed [31:0] ColumnB1
input wire signed [31:0] ColumnB2
input wire signed [31:0] ColumnB3
input wire signed [31:0] ColumnB4
output reg Error
output reg Done
output reg signed [31:0] NewColumn1
output reg signed [31:0] NewColumn2
output reg signed [31:0] NewColumn3
output reg signed [31:0] NewColumn4
CPU.v [TOP LEVEL]
input wire [9:0] Address1
input wire [9:0] Address2
input wire [2:0] OperationIn
input wire Clock
input wire Reset
input wire CPUEnable
input wire ClearAllIn
input wire [31:0] C*_*
output reg [31:0] MemoryCounter
output reg [31:0] MatrixACounter
output reg [31:0] MatrixBCounter
output reg ClearAll
output reg [31:0] AddressSelect
output reg ReadWrite
output reg Enable
output reg Done
output reg [2:0] State
output reg signed [31:0] A*_*
output reg signed [31:0] C*_*
output reg signed [31:0] DataOut1
output reg signed [31:0] DataOut2
output reg signed [31:0] DataOut3
output reg signed [31:0] DataOut4
output reg signed [31:0] DataOut5
output reg signed [31:0] DataOut6
output reg signed [31:0] DataOut7
output reg signed [31:0] DataOut8
output reg signed [31:0] DataInSelector1-4
reg signed [31:0] MatrixA [3:0] [3:0]
reg signed [31:0] MatrixB [3:0] [3:0]
reg [2:0] Operation
wire Error
wire Done
wire signed [31:0] DataIn1
wire signed [31:0] DataIn2
wire signed [31:0] DataIn3
wire signed [31:0] DataIn4
Subtractor.v
input wire Clock
input wire ClearAll
input wire [2:0] Operation
input wire signed [31:0] ColumnA1
input wire signed [31:0] ColumnA2
input wire signed [31:0] ColumnA3
input wire signed [31:0] ColumnA4
input wire signed [31:0] ColumnB1
input wire signed [31:0] ColumnB2
input wire signed [31:0] ColumnB3
input wire signed [31:0] ColumnB4
output reg Error
output reg Done
output reg signed [31:0] NewColumn1
output reg signed [31:0] NewColumn2
output reg signed [31:0] NewColumn3
output reg signed [31:0] NewColumn4
RAM.v
input wire Clock
input wire Enable
input wire ReadWrite
input wire [9:0] AddressSelect
input wire signed [31:0] InColumn1
input wire signed [31:0] InColumn2
input wire signed [31:0] InColumn3
input wire signed [31:0] InColumn4
output reg Done
output reg signed [31:0] OutColumn1
output reg signed [31:0] OutColumn2
output reg signed [31:0] OutColumn3
output reg signed [31:0] OutColumn4
reg signed [31:0] Memory [3:0] [255:0]
Transposition.v
input wire Clock
input wire ClearAll
input wire Enable
input wire [2:0] Operation
input wire signed [31:0] Column1
input wire signed [31:0] Column2
input wire signed [31:0] Column3
input wire signed [31:0] Column4
output reg Error
output reg Done
output reg signed [31:0] NewRow1
output reg signed [31:0] NewRow2
output reg signed [31:0] NewRow3
output reg signed [31:0] NewRow4
Scalar_Multiplicator.v
input wire Clock
input wire ClearAll
input wire Enable
input wire [2:0] Operation
input wire signed [31:0] ColumnA1
input wire signed [31:0] ColumnA2
input wire signed [31:0] ColumnA3
input wire signed [31:0] ColumnA4
input wire signed [31:0] ColumnB1
input wire signed [31:0] ColumnB2
input wire signed [31:0] ColumnB3
input wire signed [31:0] ColumnB4
output reg Error
output reg Done
output reg signed [31:0] NewColumn1
output reg signed [31:0] NewColumn2
output reg signed [31:0] NewColumn3
output reg signed [31:0] NewColumn4
Matrix_Multiplicator.v
input wire Clock
input wire ClearAll
input wire Enable
input wire [2:0] Operation
input wire signed [31:0] ColumnA1
input wire signed [31:0] ColumnA2
input wire signed [31:0] ColumnA3
input wire signed [31:0] ColumnA4
input wire signed [31:0] RowB1
input wire signed [31:0] RowB2
input wire signed [31:0] RowB3
input wire signed [31:0] RowB4
output reg Error
output reg Done
output reg signed [31:0] Result


Idle [000]
AddressSelect X
ReadWrite X
Enable (ALU) X
Enable (RAM) X
MatrixA X
MatrixB X
OutData X
NewMatrix X
Enable (CPU) 1
Done (RAM) X
MatrixCounterA XX
MatrixCounterB XX
Done (ALU) X
Operation XXX
Request Data from Memory [001]
AddressSelect Address1 or Address2
ReadWrite 0
Enable (ALU) X
Enable (RAM) 1
MatrixA X
MatrixB X
OutData X
NewMatrix X
Enable (CPU) X
Done (RAM) X
MatrixCounterA 11
MatrixCounterB 00, 01, 10
Done (ALU) X
Operation 001, 010, 100, 101
Enable (CPU) X
Done (RAM) X
MatrixCounterA 00, 01, 10
MatrixCounterB XX
Done (ALU) X
Operation XXX
Receive Data from Memory [010]
AddressSelect X
ReadWrite X
Enable (ALU) X
Enable (RAM) X
MatrixA Data from Memory
MatrixB Data from Memory
OutData X
NewMatrix X
Enable (CPU) X
Done (RAM) 1
MatrixCounterA XX
MatrixCounterB XX
Done (ALU) X
Operation XXX
Send Data to Memory [101]
AddressSelect Based on memory counter
ReadWrite 1
Enable (ALU) X
Enable (RAM) 1
MatrixA X
MatrixB X
OutData X
NewMatrix Set with new values
Enable (CPU) X
Done (RAM) X
MatrixCounterA XX
MatrixCounterB XX
Done (ALU) X
Operation XXX
Enable (CPU) X
Done (RAM) X
MatrixCounterA 11
MatrixCounterB 11
Done (ALU) X
Operation 001, 010, 100, 101
Enable (CPU) X
Done (RAM) X
MatrixCounterA 11
MatrixCounterB XX
Done (ALU) X
Operation 000, 110
ERROR
Enable (CPU) X
Done (RAM) X
MatrixCounterA 11
MatrixCounterB XX
Done (ALU) X
Operation 011
Enable (CPU) X
Done (RAM) X
MatrixCounterA XX
MatrixCounterB XX
Done (ALU) 1
Operation XXX
Enable (CPU) X
Done (RAM) X
MatrixCounterA 11
MatrixCounterB XX
Done (ALU) X
Operation 011
Receive Data from ALU [100]
AddressSelect X
ReadWrite X
Enable (ALU) X
Enable (RAM) X
MatrixA X
MatrixB X
OutData X
NewMatrix Incoming Data
Enable (CPU) X
Done (RAM) X
MatrixCounterA 00, 01, 10
MatrixCounterB XX
Done (ALU) X
Operation XXX
Send Data to ALU [011]
AddressSelect X
ReadWrite X
Enable (ALU) 1
Enable (RAM) X
MatrixA X
MatrixB X
OutData Row from MatrixA or Matrix B
NewMatrix X

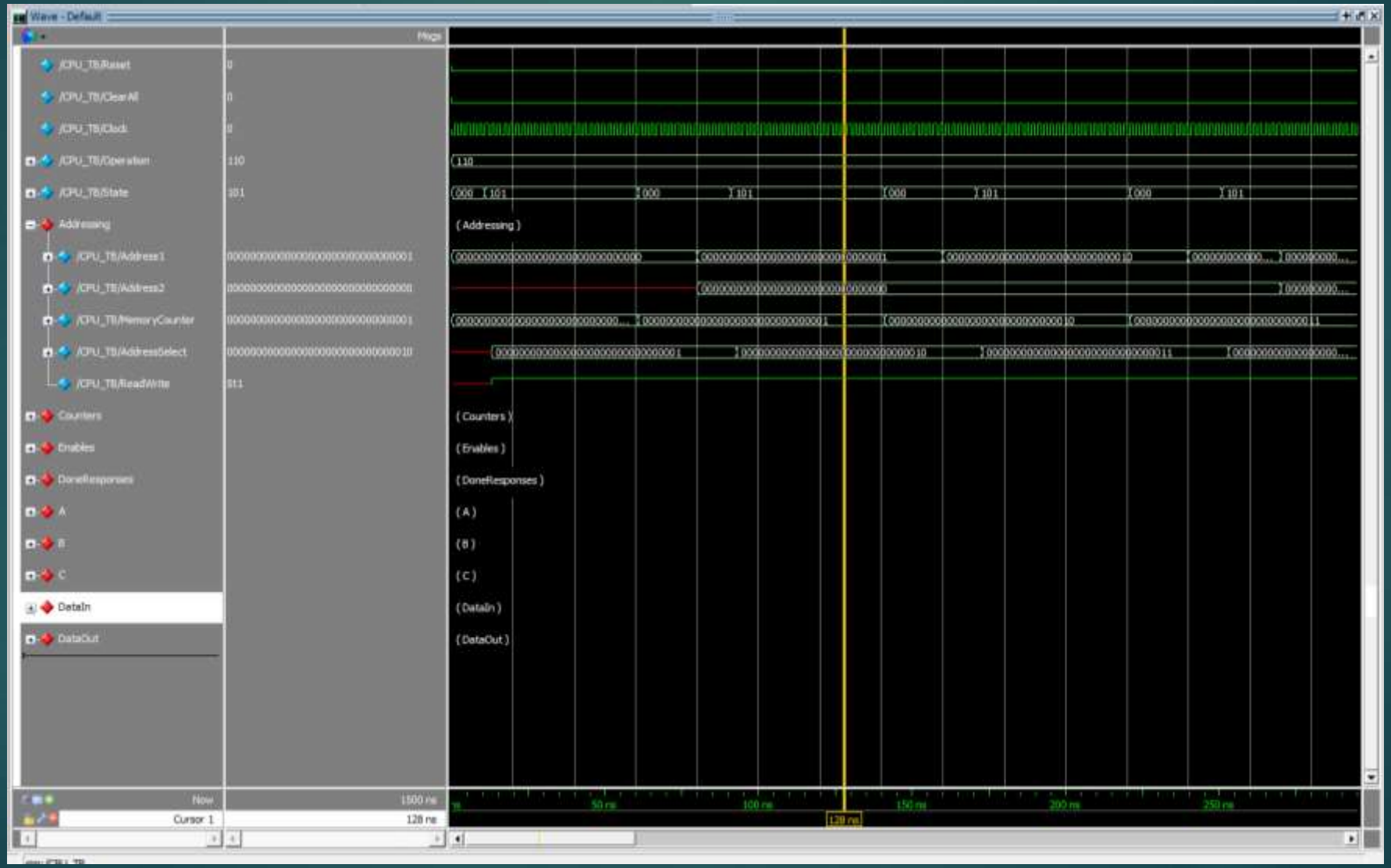
Development Environment



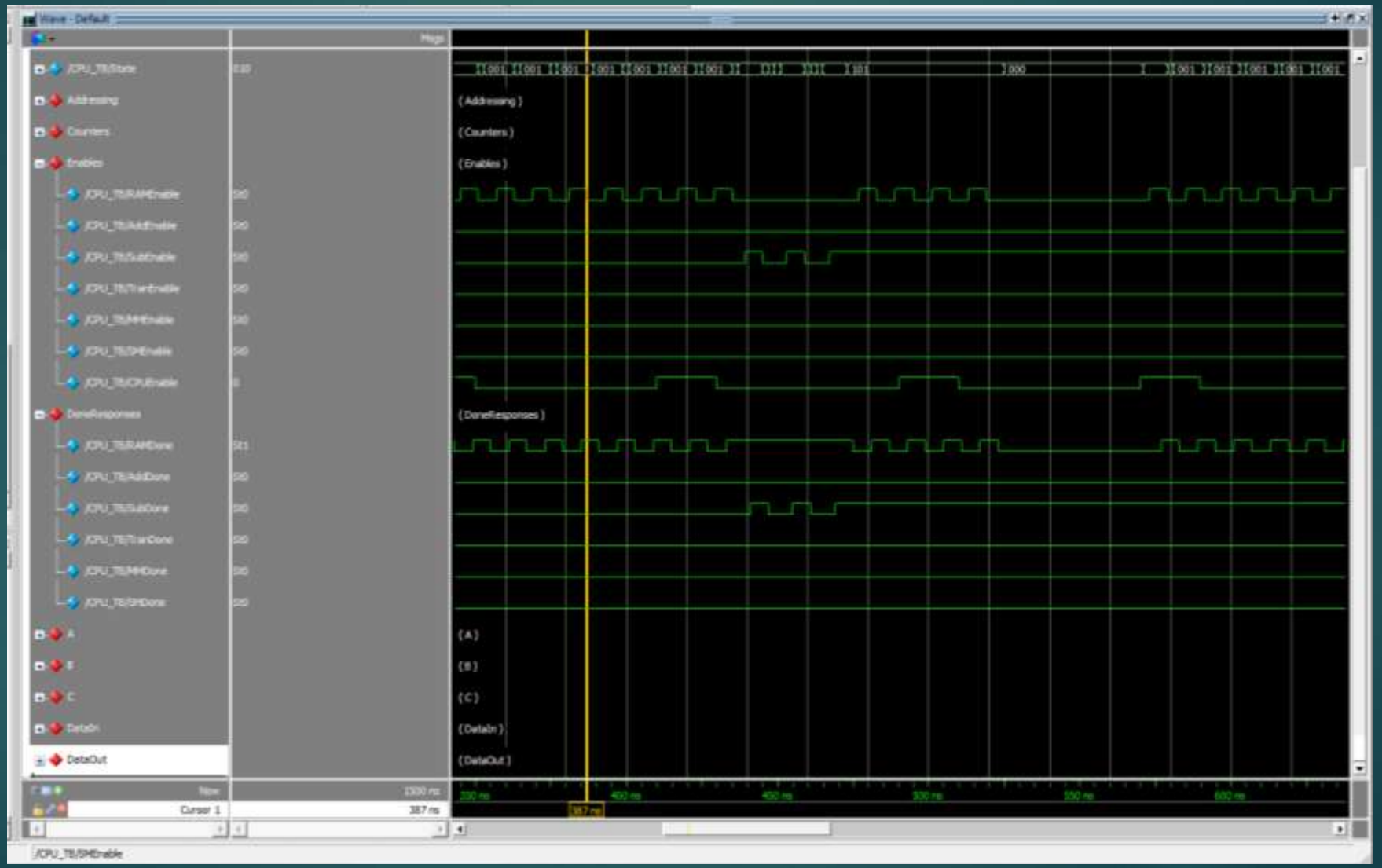
Addition


Wave - Default
Msgs
/Adder_TB/Clock
/Adder_TB/Enable
/Adder_TB/Operation
/Adder_TB/ClearAll
/Adder_TB/ColumnA1
/Adder_TB/ColumnA2
/Adder_TB/ColumnA3
/Adder_TB/ColumnA4
/Adder_TB/ColumnB1
/Adder_TB/ColumnB2
/Adder_TB/ColumnB3
/Adder_TB/ColumnB4
/Adder_TB/Error
/Adder_TB/Done
/Adder_TB/NewColumn1
/Adder_TB/NewColumn2
/Adder_TB/NewColumn3
/Adder_TB/NewColumn4
1
0
001
0
-4
-4
-4
-4
10
10
10
10
St0
St0
x
x
x
x
6
6
6
6
Now
200 ns
Cursor 1
5 ns
4 ns
8 ns
12 ns
16 ns
20 ns
24 ns
28 ns
32 ns
36 ns
40 ns
44 ns
48 ns
52 ns
56 ns
60 ns
64 ns
68 ns

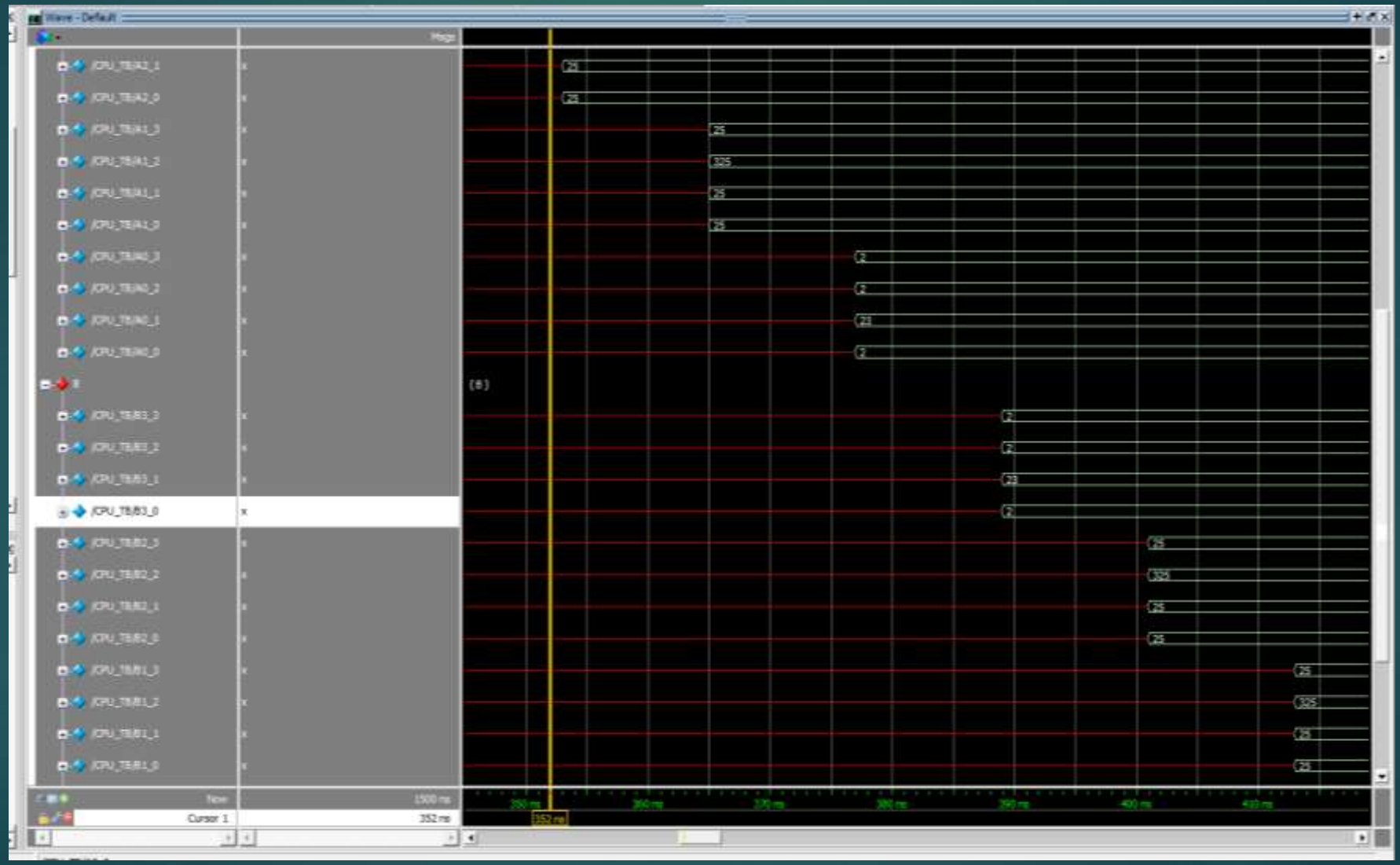
CPU Memory Addressing



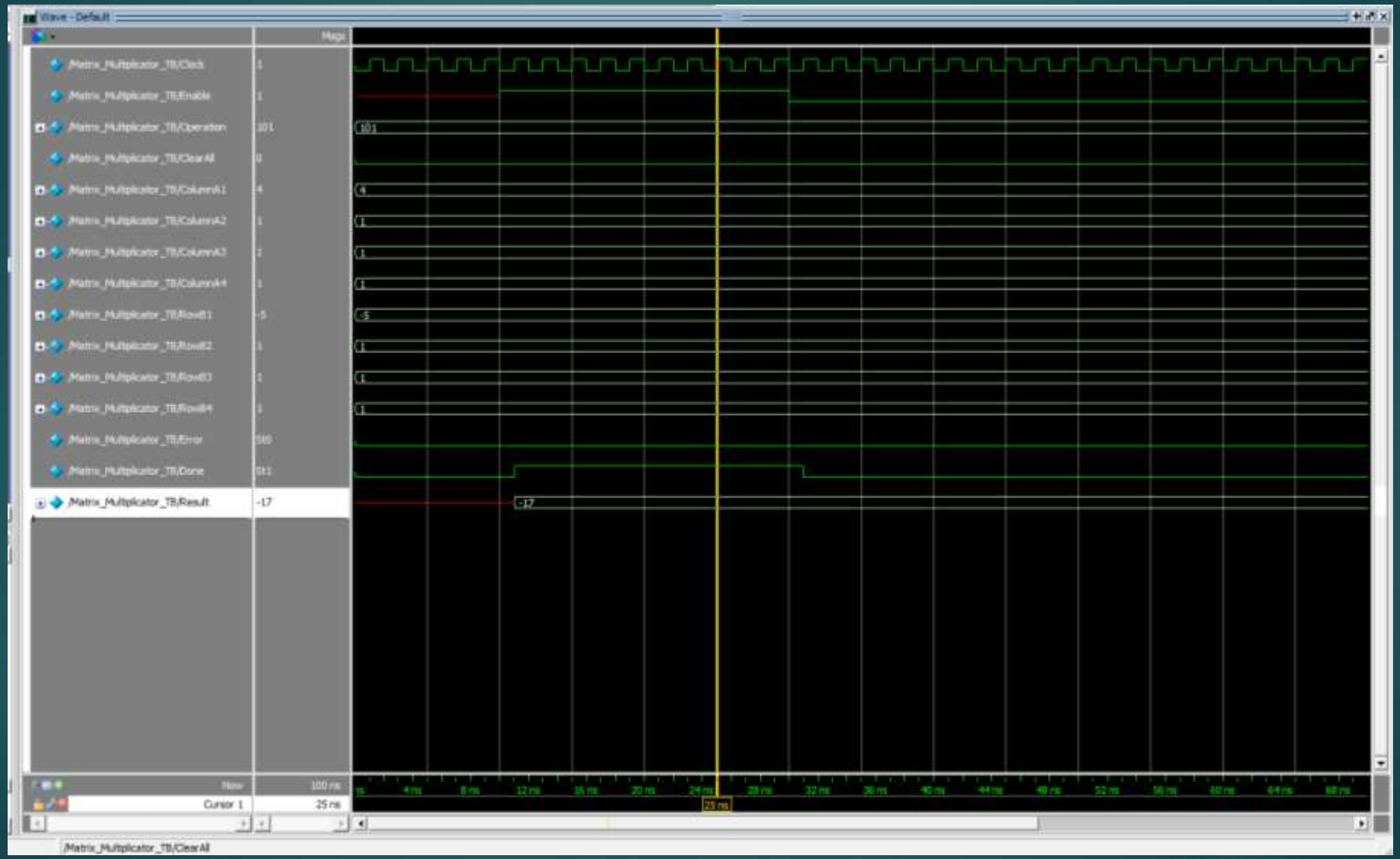
ALU Enables and Done Responses



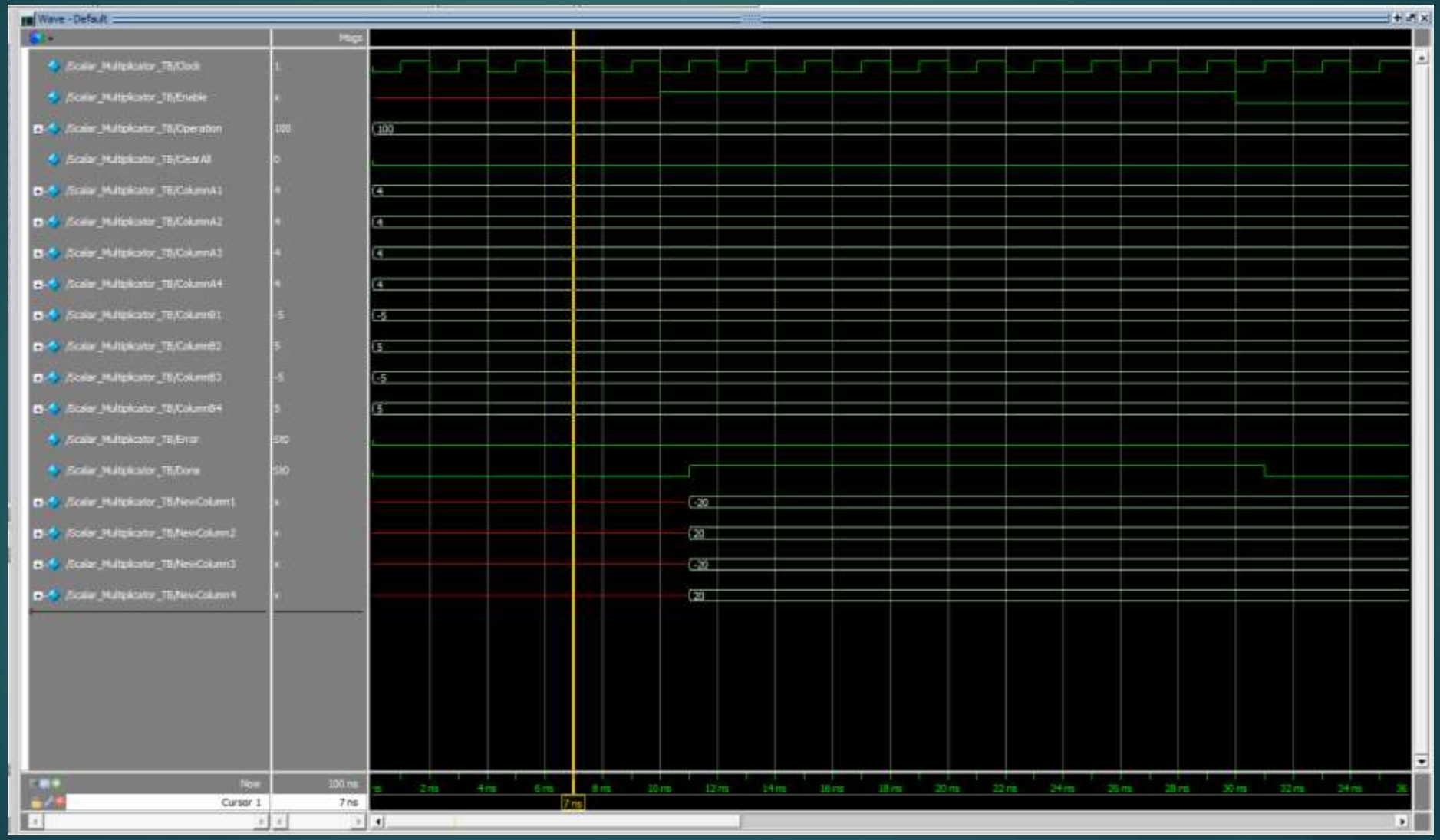
Loading CPU Registers from Memory



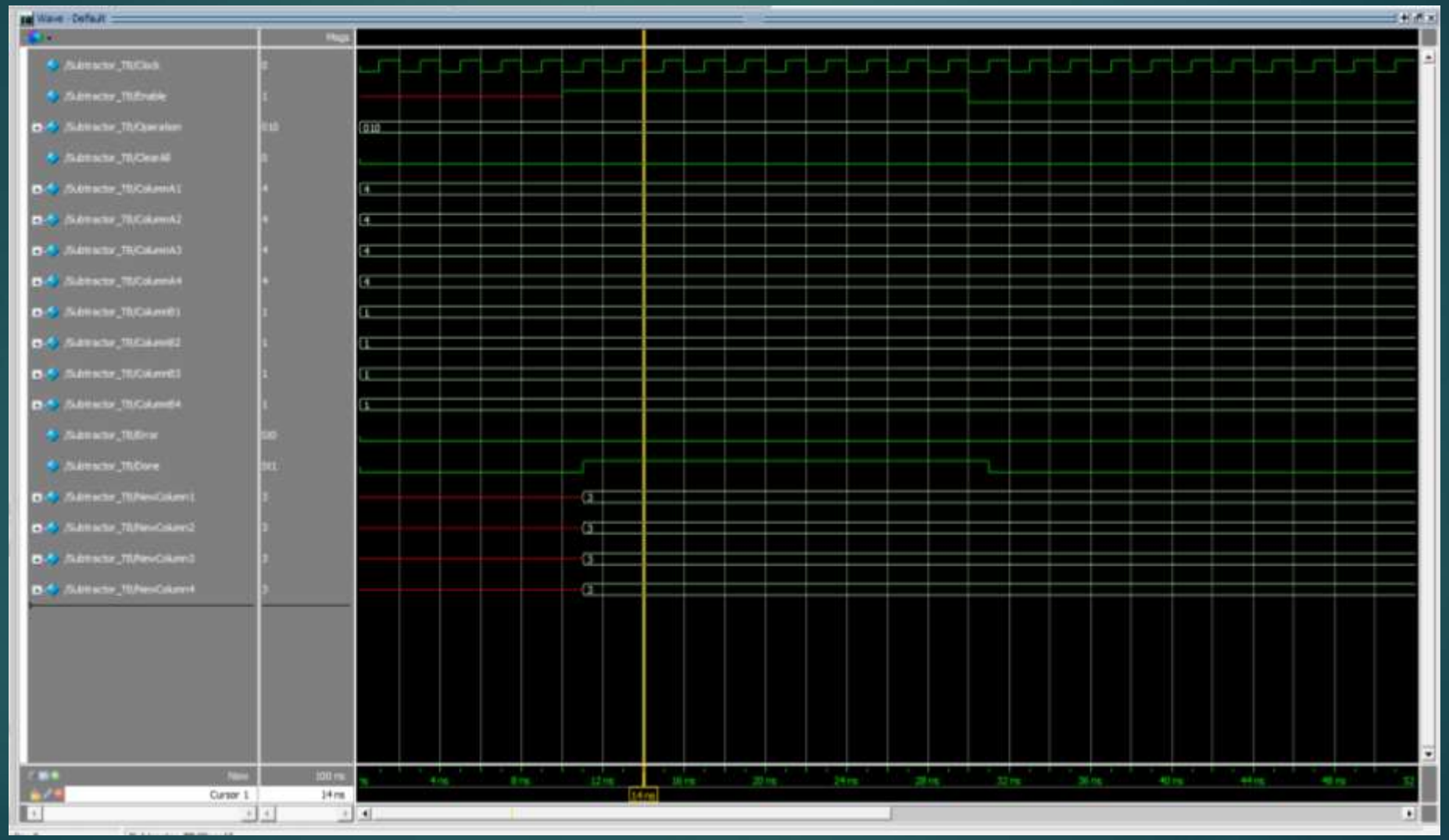
Matrix Multiplication



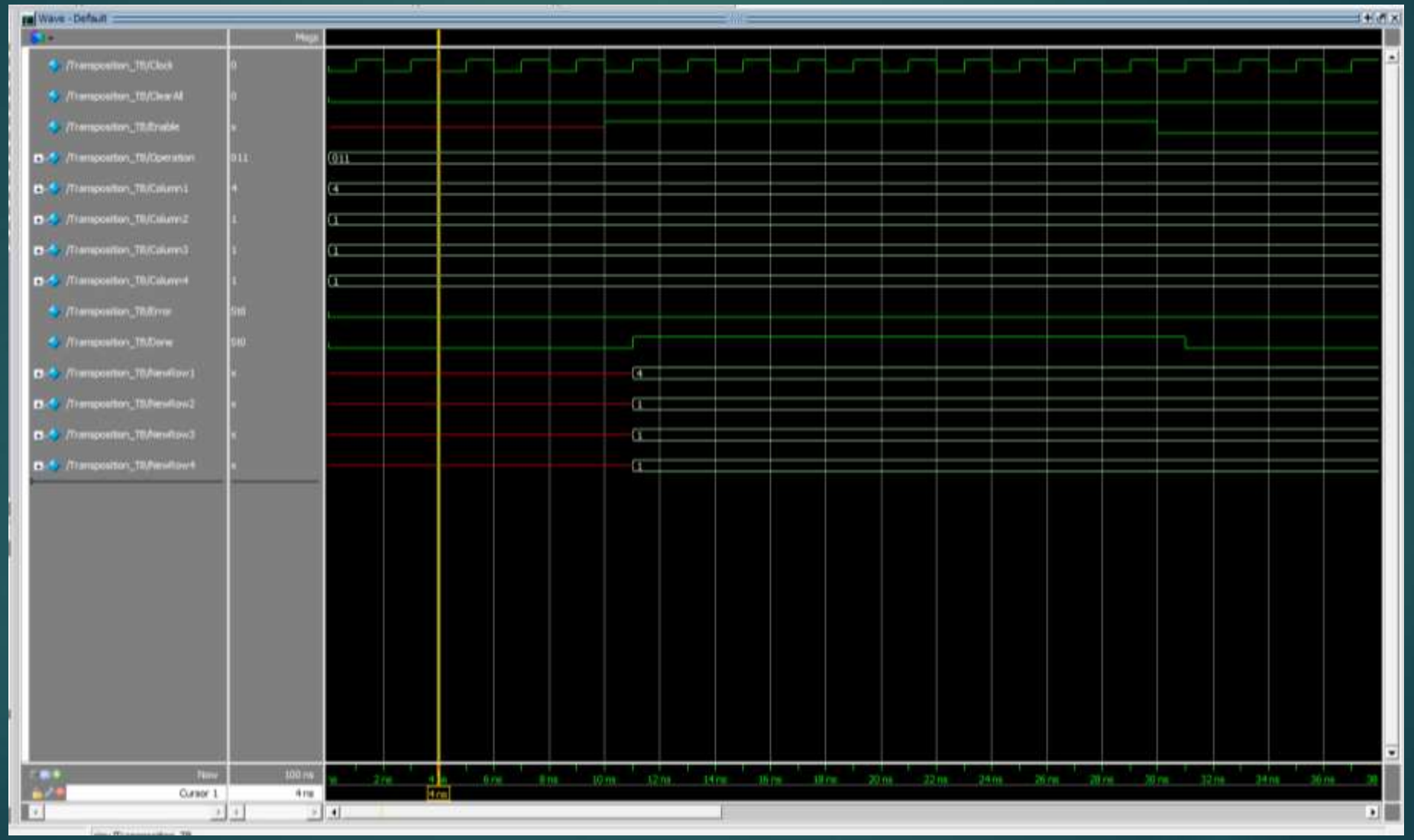
Scalar Multiplication



Subtraction



Transposition



Questions?