EEE 468
VLSI DESIGN PROJECT

Designing a RV32i compatible core with SPI interface

Group No:G1 (01)

Presented By

Mrinmoy Kundu	-1706001
Md. Jawad Ul Islam	-1706003
Aroni Ghosh	-1706004
Swapnil Siddiky	-1706018
Saleh Ahmed Khan	-1706053
Adib Md. Ridwan	-1706055
Sadat Tahmeed Azad	-1706064

Submitted to:

Dr. A.B.M. Harun-Ur-Rashid

Professor, Department of EEE, BUET

Mumtahina Islam Sukanya

Lecturer, Department of EEE, BUET

Objectives

- Design (with RTL) and simulate a 32-bit rv32i compatible core.
- Use core to execute instructions written in C that's converted into 32-bit hexadecimal format.
- Load disassembled instructions into the instruction memory of the processor during verification with the help of a chosen interface (SPI).
- Synthesize and perform the physical design of Hardware Description of their RISC-V processor.

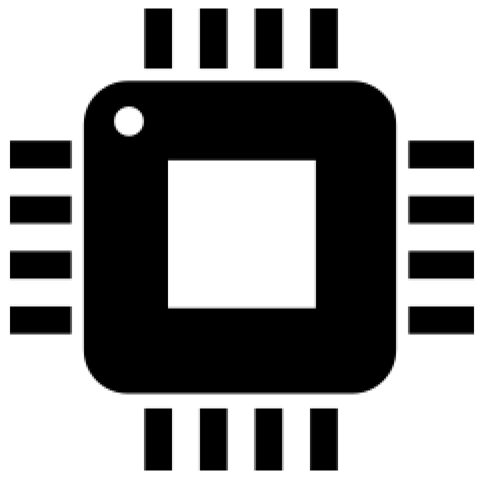
Specifications

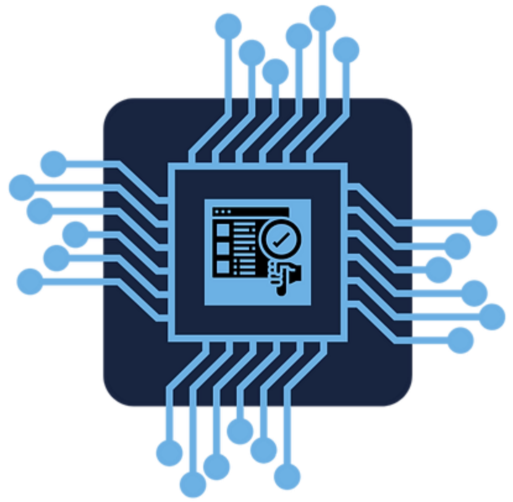
1. Core must be compatible with rv32i instruction set.
2. Implement RISC-V ISA spec. Reference: RV32i-ISA
3. Use input global clock (positive edge triggered: clk), global reset (negative edge triggered: rst_n) and core_select signal.
4. Create a memory controller with an industry standard interface (SPI) to load the instruction data to the memory.
5. core_select functionality:

core_select	Functionality
0	1. Memory controller interface will be active to write data into the memory.
1	1. Core will start to work 2. Core side interface of the memory will be active to read and write data into the memory.

6. Core HDL descriptions need to be Synthesizable and optimized.
7. No pipelining stages are necessary.

Workflow





Core design

- Datapath design
- RTL implementation
- Testing and verification

Interface design

- RTL implementation for standard SPI protocol
- Testing and verification

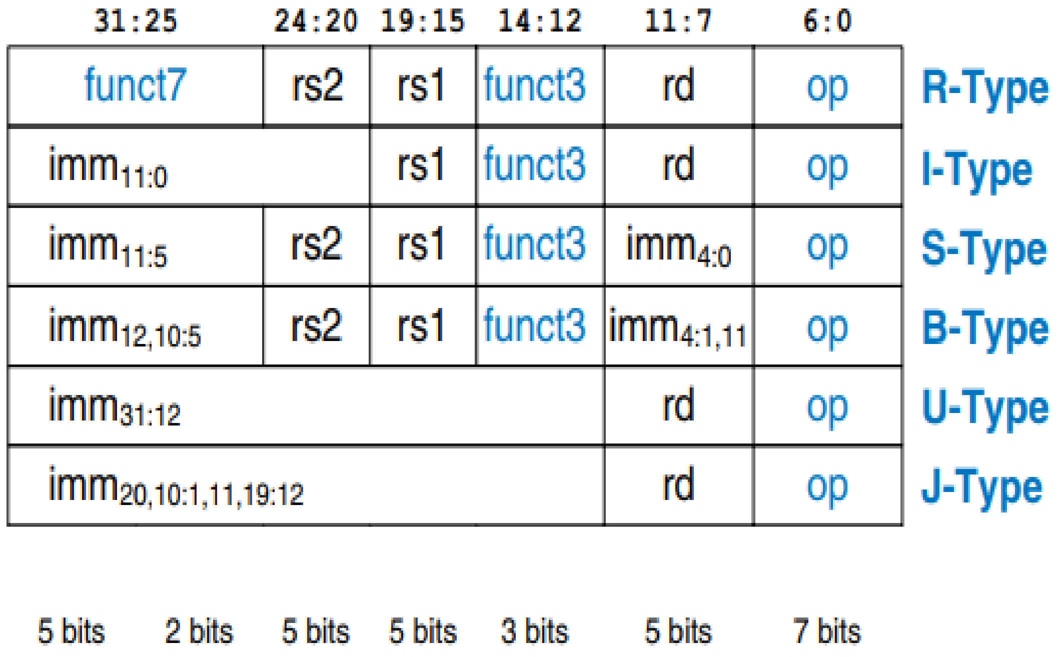
Physical design

- Compile necessary resources for synthesis and physical design while design gets ready

Core Design

Core: ISA

- The instruction set architecture - the entire group of commands that the processor can perform to execute the program instructions.
- RV32I ISA is a 32-bit version of the RISC-V ISA --> data and address buses are 32 bits wide.
- Includes 6 types of instructions.
- Consists of 37 unique instructions.

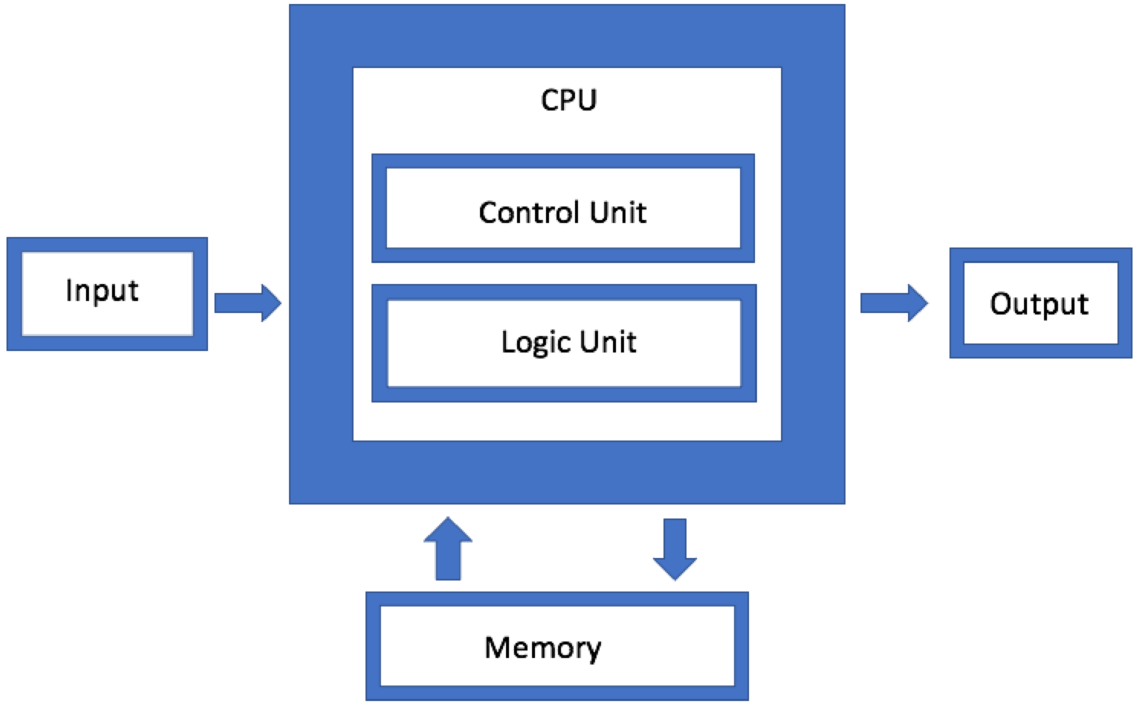


31:25	24:20	19:15	14:12	11:7	6:0	
funct7	rs2	rs1	funct3	rd	op	R-Type
$imm_{11:0}$		rs1	funct3	rd	op	I-Type
$imm_{11:5}$	rs2	rs1	funct3	$imm_{4:0}$	op	S-Type
$imm_{12,10:5}$	rs2	rs1	funct3	$imm_{4:1,11}$	op	B-Type
$imm_{31:12}$				rd	op	U-Type
$imm_{20,10:1,11,19:12}$				rd	op	J-Type

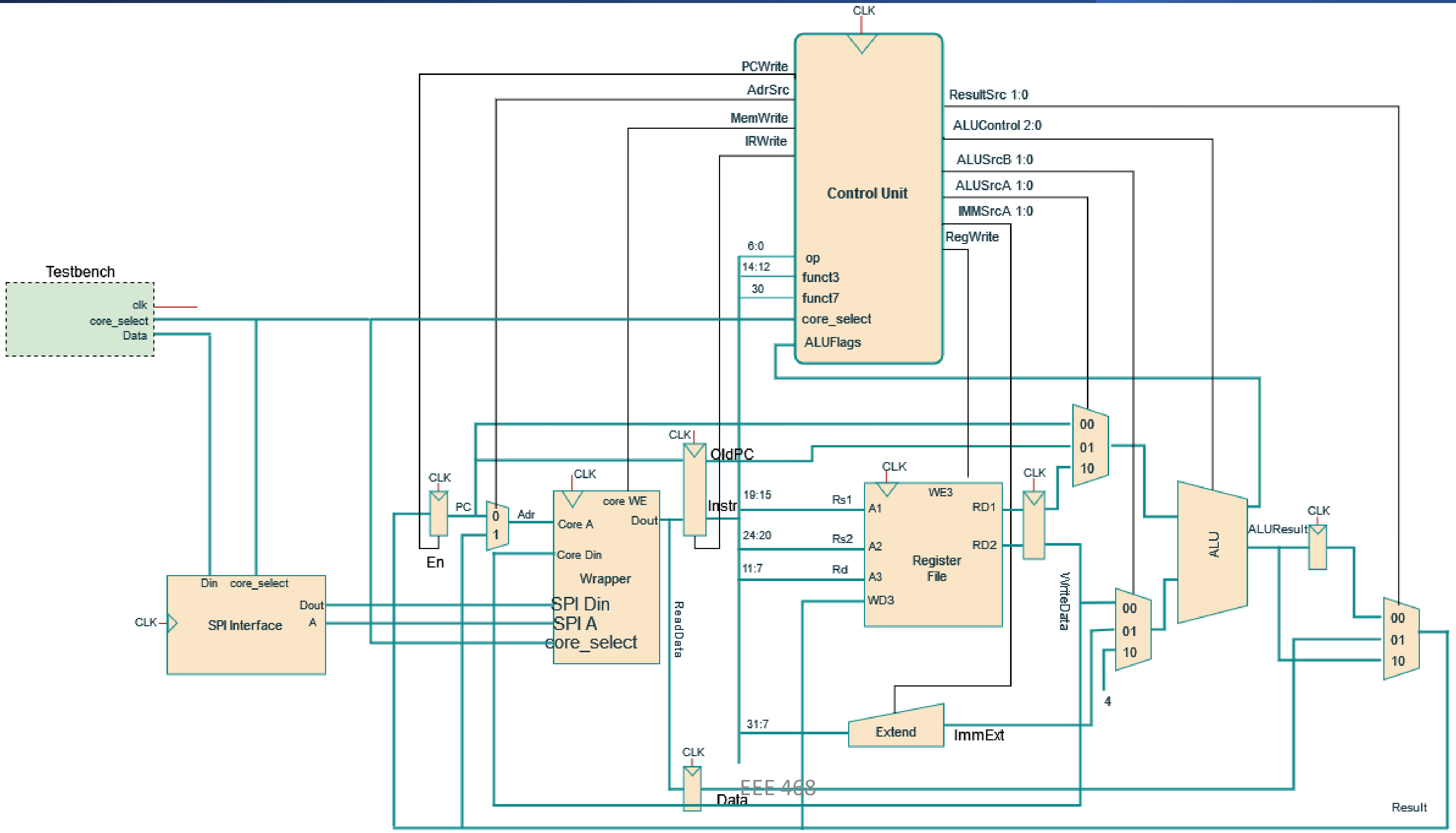
5 bits 2 bits 5 bits 5 bits 3 bits 5 bits 7 bits

Core: Micro Architecture

- Von Neumann Architecture
- 2048x32 bit memory inside Wrapper for Data and Instructions
- 32 bits *multicycle processor* with 32 architectural registers
- ALU functions extended for logical & arithmetical bit shifting



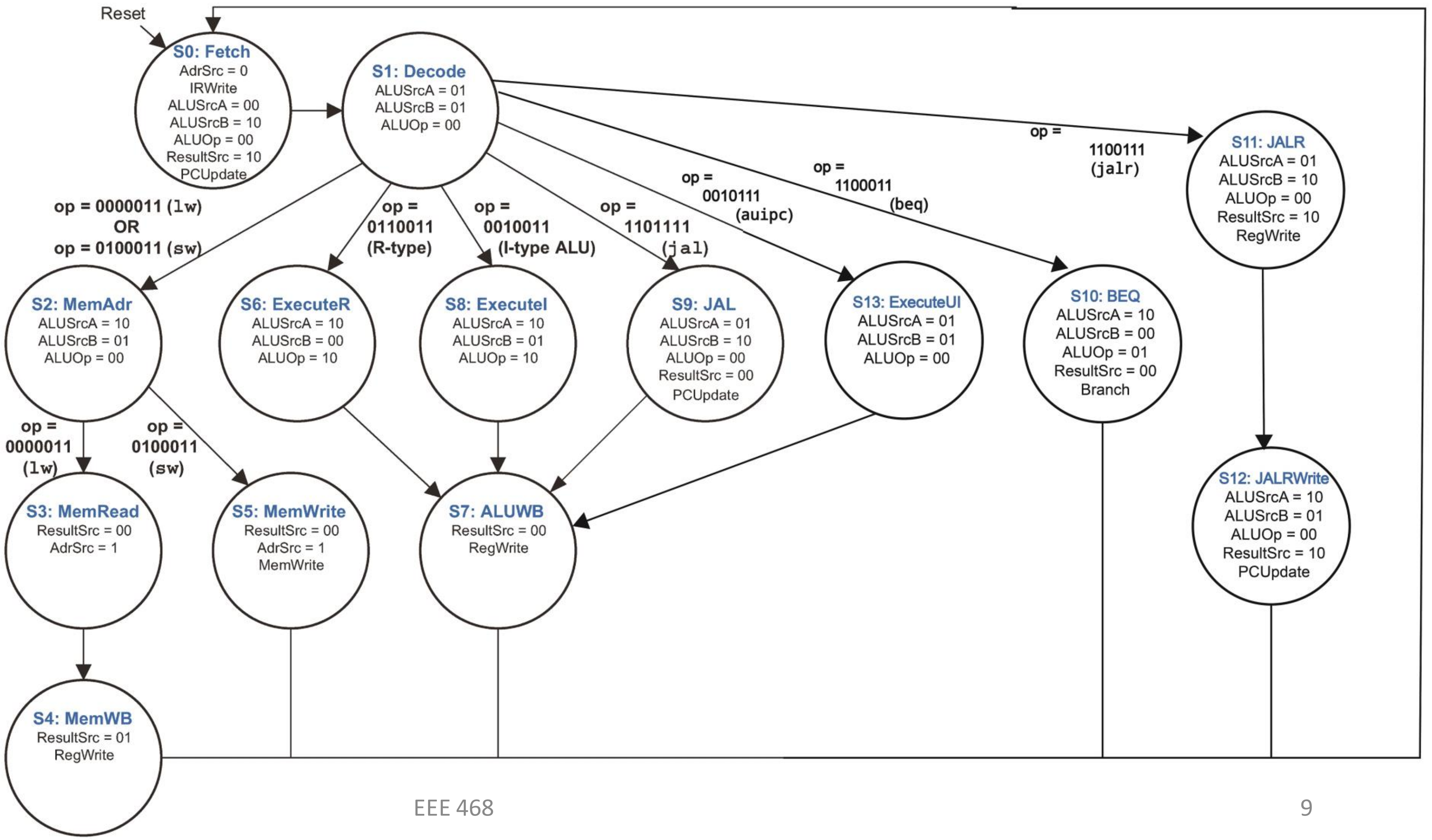
Core: Datapath



Core: Control Unit - FSM

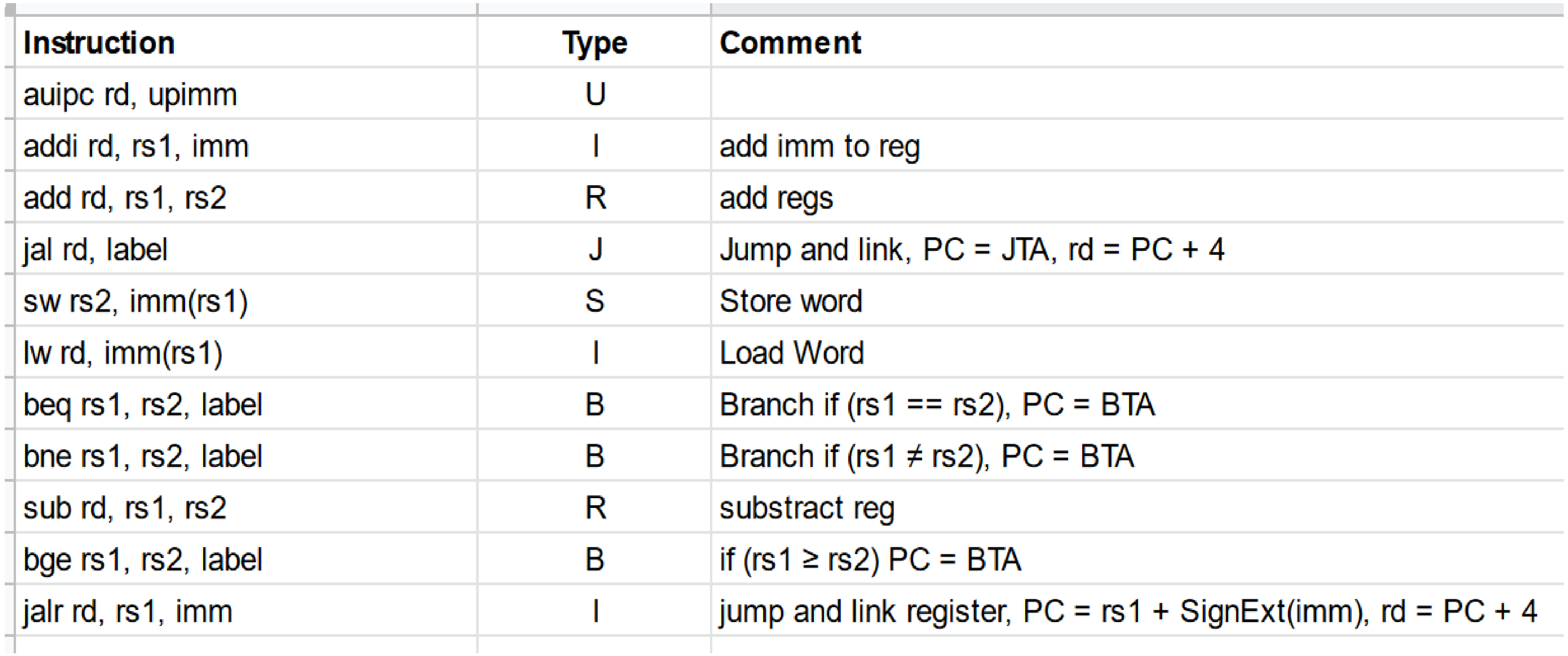
15 State FSM for 6 different types of instruction.

One idle state for core_select = 0



Core: ISA support

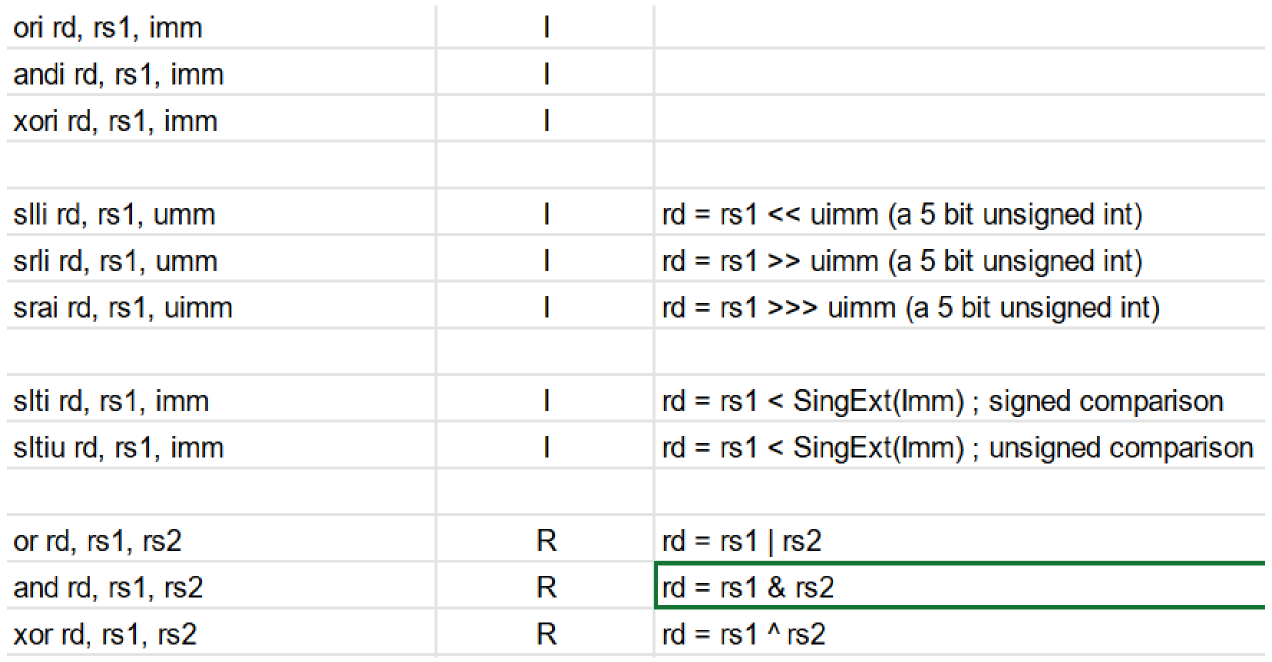
Required Instructions



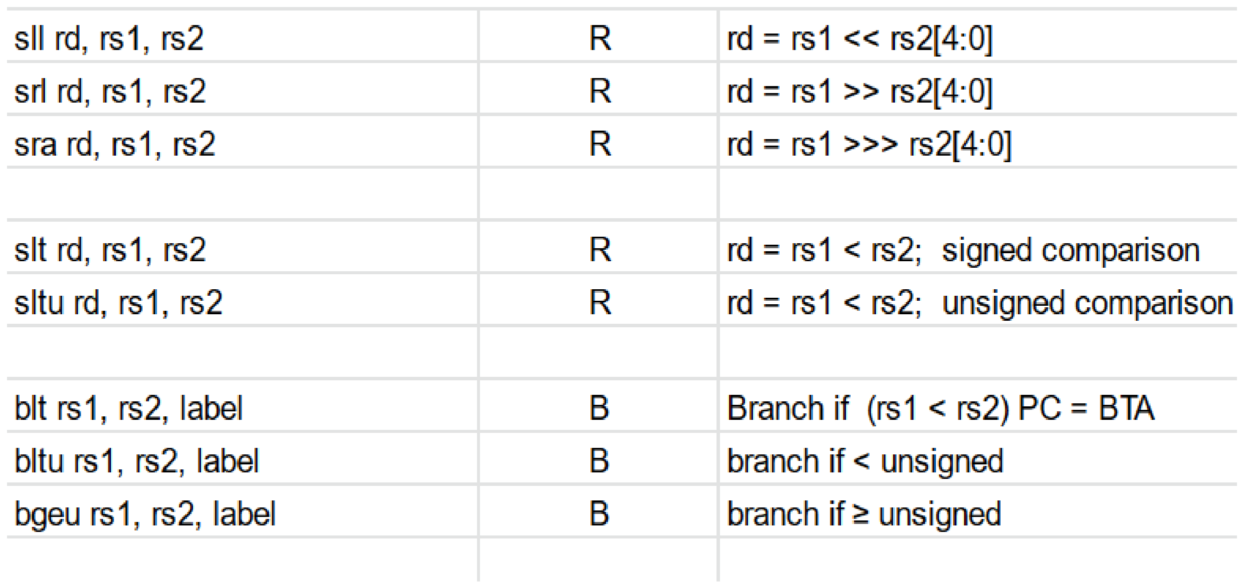
Instruction	Type	Comment
auipc rd, upimm	U	
addi rd, rs1, imm	I	add imm to reg
add rd, rs1, rs2	R	add regs
jal rd, label	J	Jump and link, PC = JTA, rd = PC + 4
sw rs2, imm(rs1)	S	Store word
lw rd, imm(rs1)	I	Load Word
beq rs1, rs2, label	B	Branch if (rs1 == rs2), PC = BTA
bne rs1, rs2, label	B	Branch if (rs1 ≠ rs2), PC = BTA
sub rd, rs1, rs2	R	substract reg
bge rs1, rs2, label	B	if (rs1 ≥ rs2) PC = BTA
jalr rd, rs1, imm	I	jump and link register, PC = rs1 + SignExt(imm), rd = PC + 4

Core: ISA support

Additional Instructions Implemented



ori rd, rs1, imm	I	
andi rd, rs1, imm	I	
xori rd, rs1, imm	I	
slli rd, rs1, umm	I	rd = rs1 << uimm (a 5 bit unsigned int)
srli rd, rs1, umm	I	rd = rs1 >> uimm (a 5 bit unsigned int)
srai rd, rs1, uimm	I	rd = rs1 >>> uimm (a 5 bit unsigned int)
slti rd, rs1, imm	I	rd = rs1 < SingExt(Imm) ; signed comparison
sltiu rd, rs1, imm	I	rd = rs1 < SingExt(Imm) ; unsigned comparison
or rd, rs1, rs2	R	rd = rs1 | rs2
and rd, rs1, rs2	R	rd = rs1 & rs2
xor rd, rs1, rs2	R	rd = rs1 ^ rs2



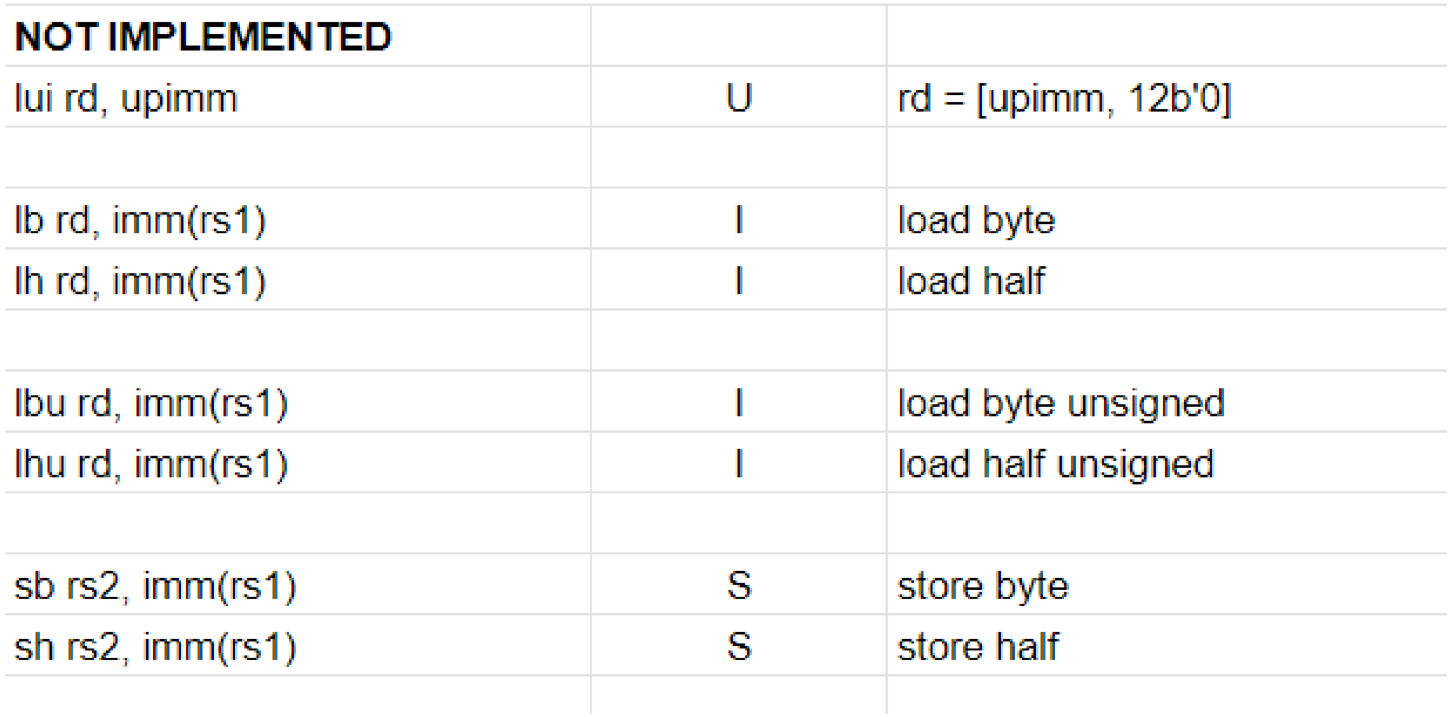
sll rd, rs1, rs2	R	rd = rs1 << rs2[4:0]
srl rd, rs1, rs2	R	rd = rs1 >> rs2[4:0]
sra rd, rs1, rs2	R	rd = rs1 >>> rs2[4:0]
slt rd, rs1, rs2	R	rd = rs1 < rs2; signed comparison
sltu rd, rs1, rs2	R	rd = rs1 < rs2; unsigned comparison
blt rs1, rs2, label	B	Branch if (rs1 < rs2) PC = BTA
bltu rs1, rs2, label	B	branch if < unsigned
bgeu rs1, rs2, label	B	branch if ≥ unsigned

Core: ISA support

Instructions Not Implemented

Some instructions have not been implemented mainly due to **byte addressability.**

Thus, in total, **30 out of 37** rv32i instructions are functional in our design.



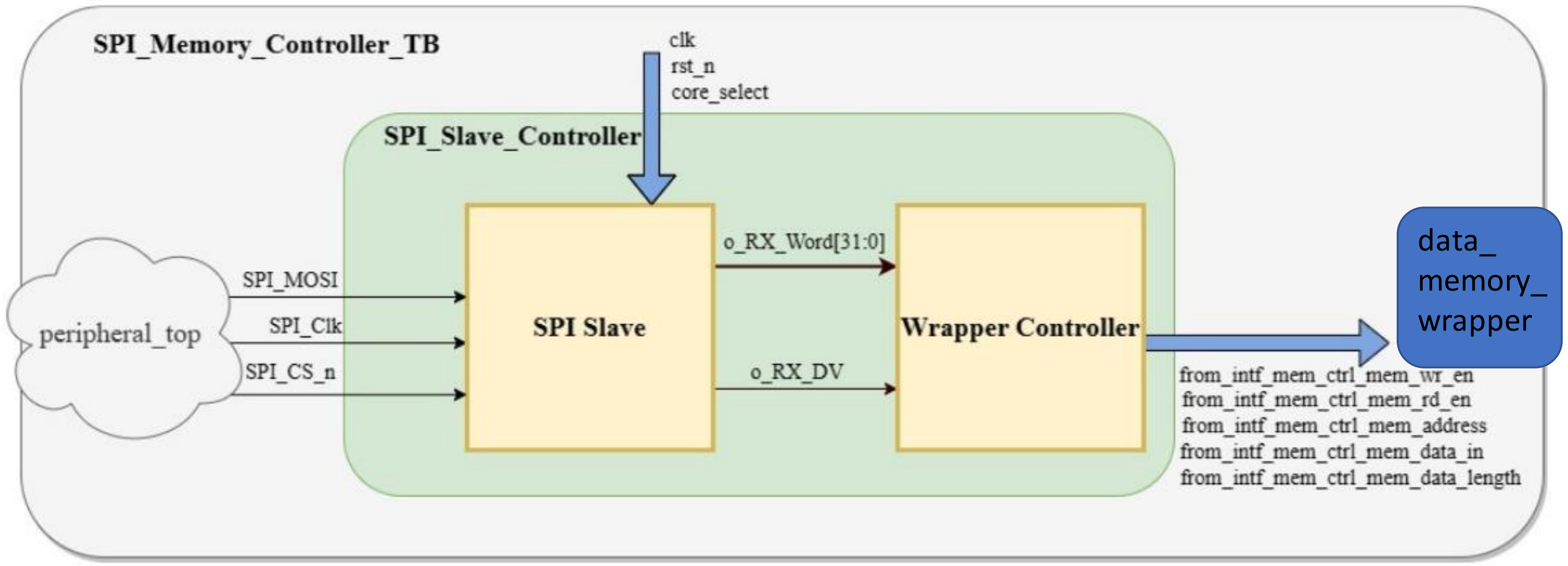
NOT IMPLEMENTED		
lui rd, upimm	U	rd = [upimm, 12b'0]
lb rd, imm(rs1)	I	load byte
lh rd, imm(rs1)	I	load half
lbu rd, imm(rs1)	I	load byte unsigned
lhu rd, imm(rs1)	I	load half unsigned
sb rs2, imm(rs1)	S	store byte
sh rs2, imm(rs1)	S	store half

Interface Design

Interface(SPI) : Introduction

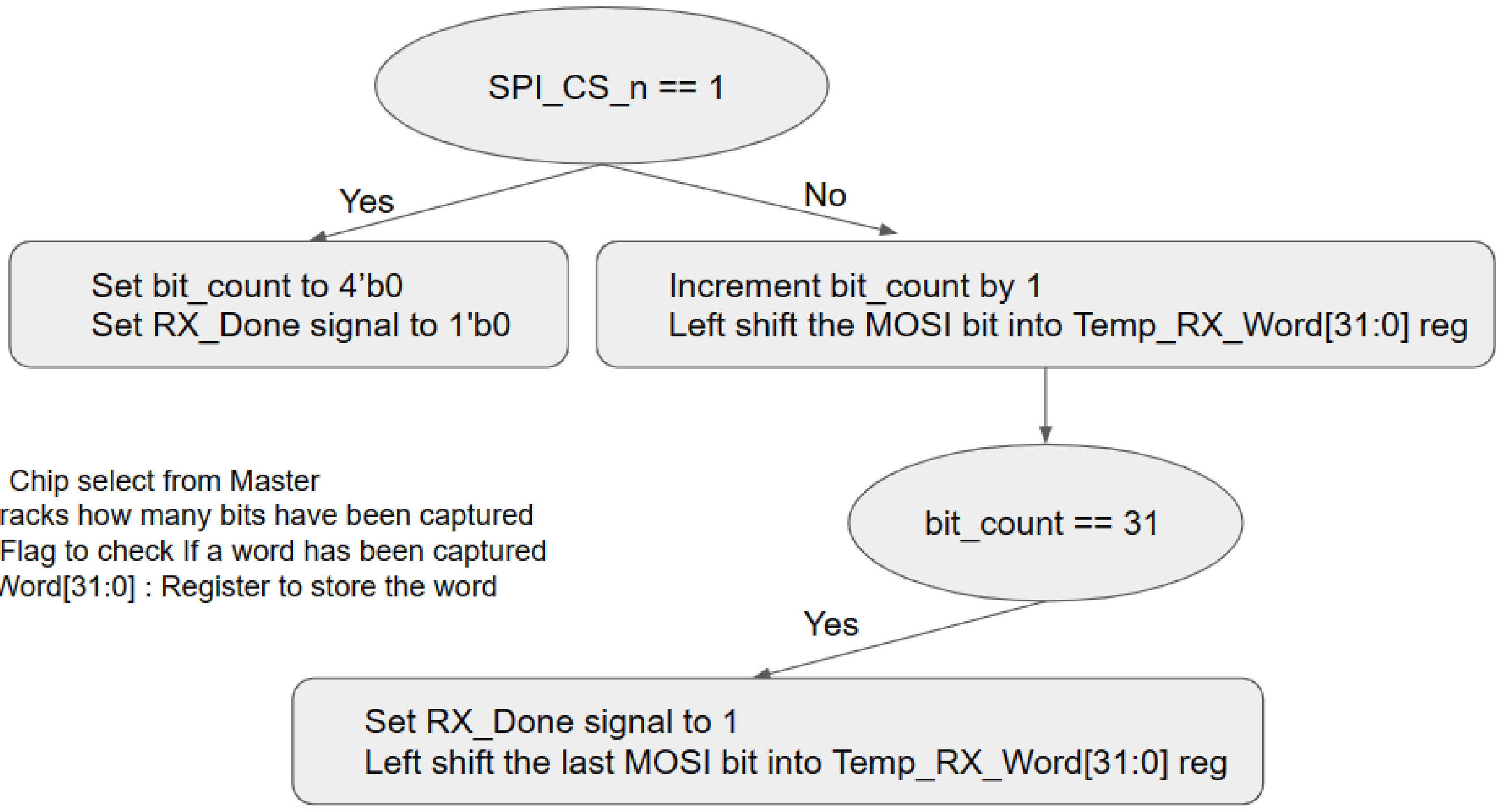
- Peripheral Master generates SPI_Clk, SPI_MOSI signals
- 32-bit data serially transmitted as 1 bit MOSI signals
- A chip select signal SPI_CS_n to control the data transmission
- Data decoded by SPI Slave & sent to memory wrapper

Interface(SPI) : Block Diagram



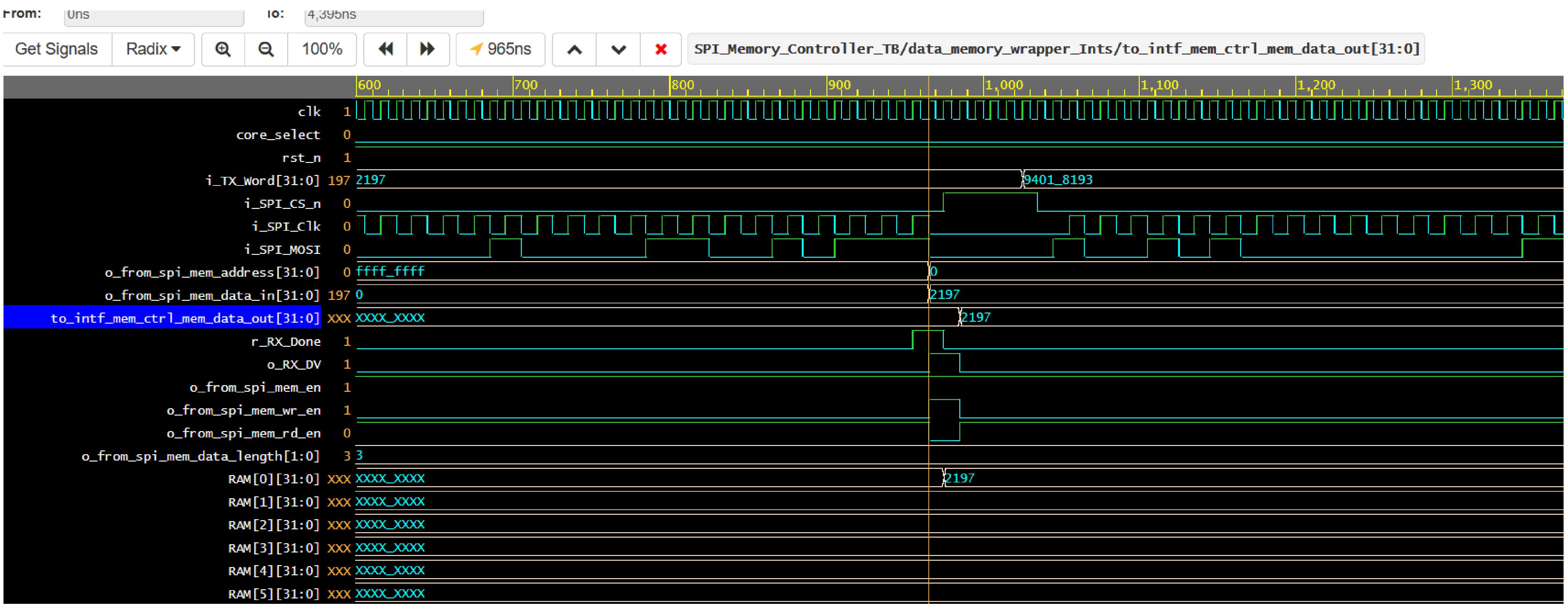
Interface(SPI) : Flowchart for Word Extraction

After detecting posedge of **SPI_Clk or SPI_CS_n**, the following behavioral logic will be implemented


SPI_CS_n == 1
Yes
No
Set bit_count to 4'b0
Set RX_Done signal to 1'b0
Increment bit_count by 1
Left shift the MOSI bit into Temp_RX_Word[31:0] reg
bit_count == 31
Yes
Set RX_Done signal to 1
Left shift the last MOSI bit into Temp_RX_Word[31:0] reg

SPI_CS_n : Chip select from Master
bit_count : tracks how many bits have been captured
RX_Done : Flag to check If a word has been captured
Temp_RX_Word[31:0] : Register to store the word

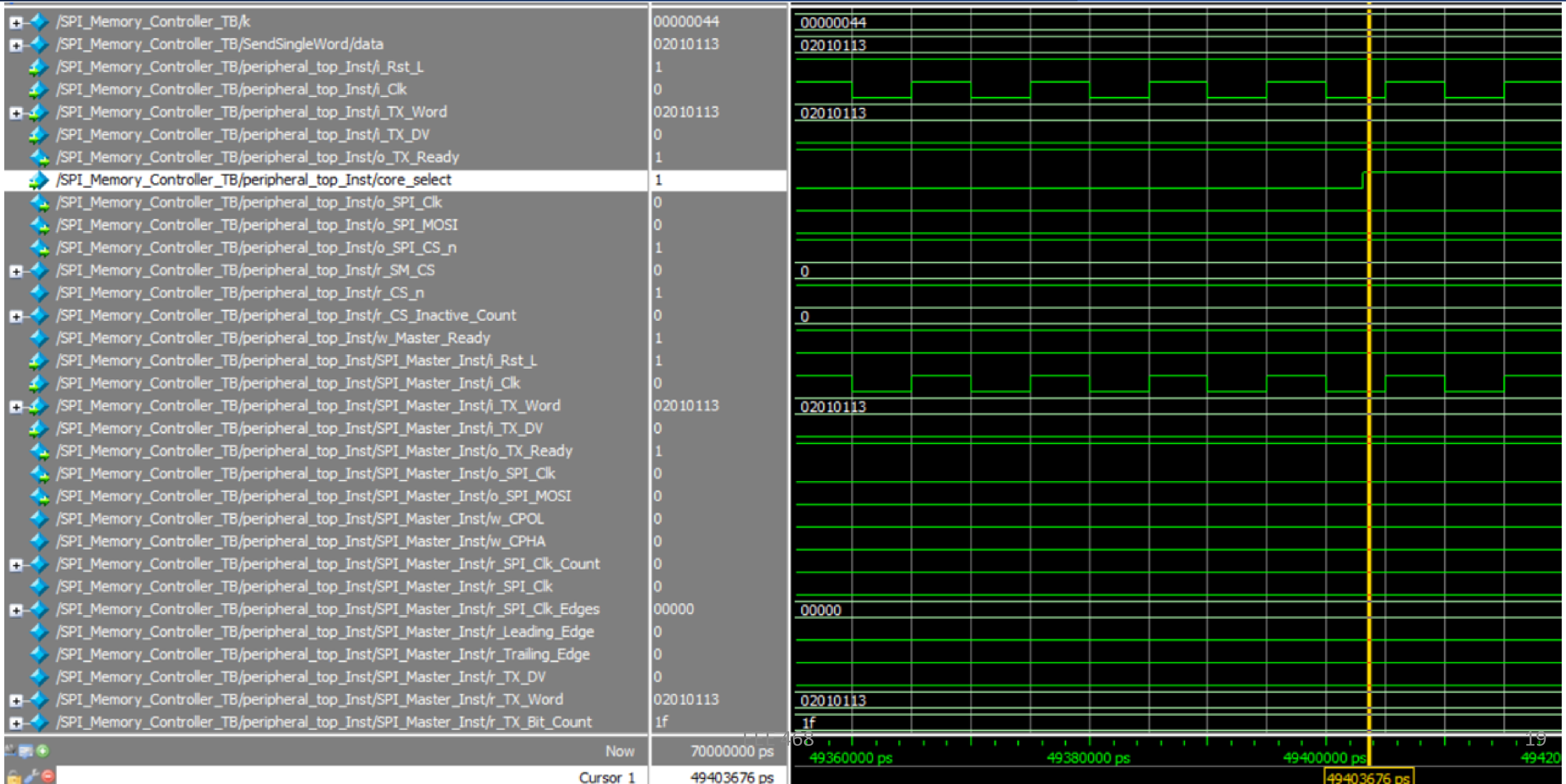
Interface(SPI) : Timing Diagram


From: 0ns
To: 4,395ns
Get Signals
Radix
100%
965ns
SPI_Memory_Controller_TB/data_memory_wrapper_Ints/to_intf_mem_ctrl_mem_data_out[31:0]
600
700
800
900
1,000
1,100
1,200
1,300

```
clk                                  1
core_select                          0
rst_n                                1
i_TX_Word[31:0]                    197  2197          9401_8193
i_SPI_CS_n                           0
i_SPI_Clk                            0
i_SPI_MOSI                           0
o_from_spi_mem_address[31:0]         0  ffff_ffff     0
o_from_spi_mem_data_in[31:0]       197  0             2197
to_intf_mem_ctrl_mem_data_out[31:0] XXX XXXX_XXXX     2197
r_RX_Done                            1
o_RX_DV                              1
o_from_spi_mem_en                    1
o_from_spi_mem_wr_en                 1
o_from_spi_mem_rd_en                 0
o_from_spi_mem_data_length[1:0]      3  3
RAM[0][31:0]                       XXX  XXXX_XXXX     2197
RAM[1][31:0]                       XXX  XXXX_XXXX
RAM[2][31:0]                       XXX  XXXX_XXXX
RAM[3][31:0]                       XXX  XXXX_XXXX
RAM[4][31:0]                       XXX  XXXX_XXXX
RAM[5][31:0]                       XXX  XXXX_XXXX
```

Outputs

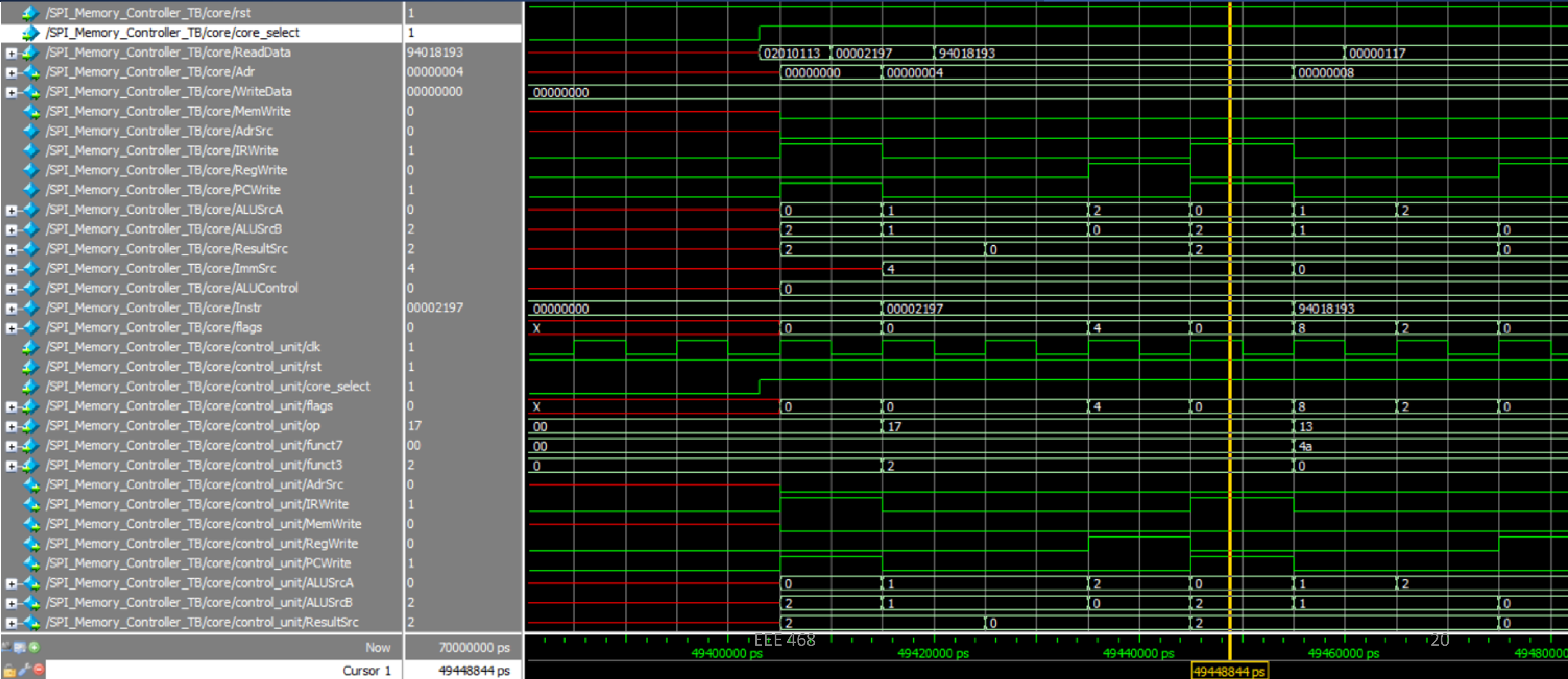
Outputs



```
/SPI_Memory_Controller_TB/k                                                      00000044
/SPI_Memory_Controller_TB/SendSingleWord/data                                    02010113
/SPI_Memory_Controller_TB/peripheral_top_Inst/i_Rst_L                            1
/SPI_Memory_Controller_TB/peripheral_top_Inst/i_Clk                              0
/SPI_Memory_Controller_TB/peripheral_top_Inst/i_TX_Word                          02010113
/SPI_Memory_Controller_TB/peripheral_top_Inst/i_TX_DV                            0
/SPI_Memory_Controller_TB/peripheral_top_Inst/o_TX_Ready                         1
/SPI_Memory_Controller_TB/peripheral_top_Inst/core_select                        1
/SPI_Memory_Controller_TB/peripheral_top_Inst/o_SPI_Clk                          0
/SPI_Memory_Controller_TB/peripheral_top_Inst/o_SPI_MOSI                         0
/SPI_Memory_Controller_TB/peripheral_top_Inst/o_SPI_CS_n                         1
/SPI_Memory_Controller_TB/peripheral_top_Inst/r_SM_CS                            0
/SPI_Memory_Controller_TB/peripheral_top_Inst/r_CS_n                             1
/SPI_Memory_Controller_TB/peripheral_top_Inst/r_CS_Inactive_Count                0
/SPI_Memory_Controller_TB/peripheral_top_Inst/w_Master_Ready                     1
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/i_Rst_L            1
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/i_Clk              0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/i_TX_Word          02010113
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/i_TX_DV            0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/o_TX_Ready         1
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/o_SPI_Clk          0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/o_SPI_MOSI         0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/w_CPOL             0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/w_CPHA             0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/r_SPI_Clk_Count    0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/r_SPI_Clk          0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/r_SPI_Clk_Edges    00000
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/r_Leading_Edge     0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/r_Trailing_Edge    0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/r_TX_DV            0
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/r_TX_Word          02010113
/SPI_Memory_Controller_TB/peripheral_top_Inst/SPI_Master_Inst/r_TX_Bit_Count     1f

Now         70000000 ps
Cursor 1    49403676 ps
```

Outputs

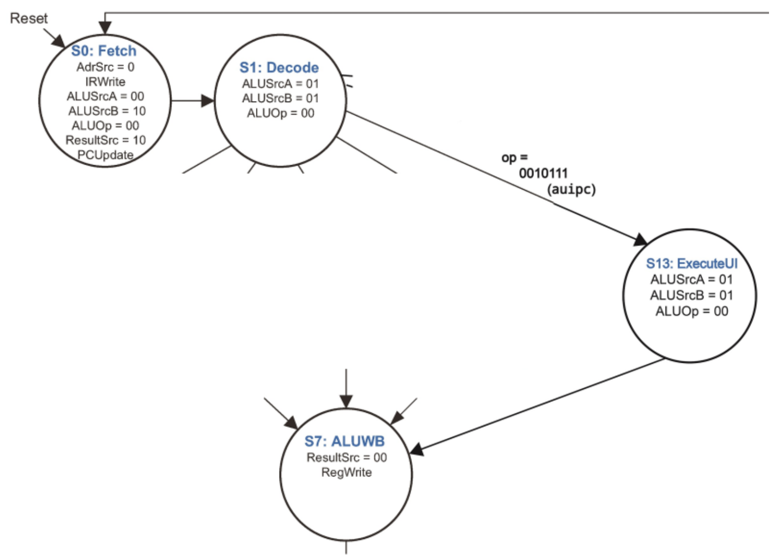


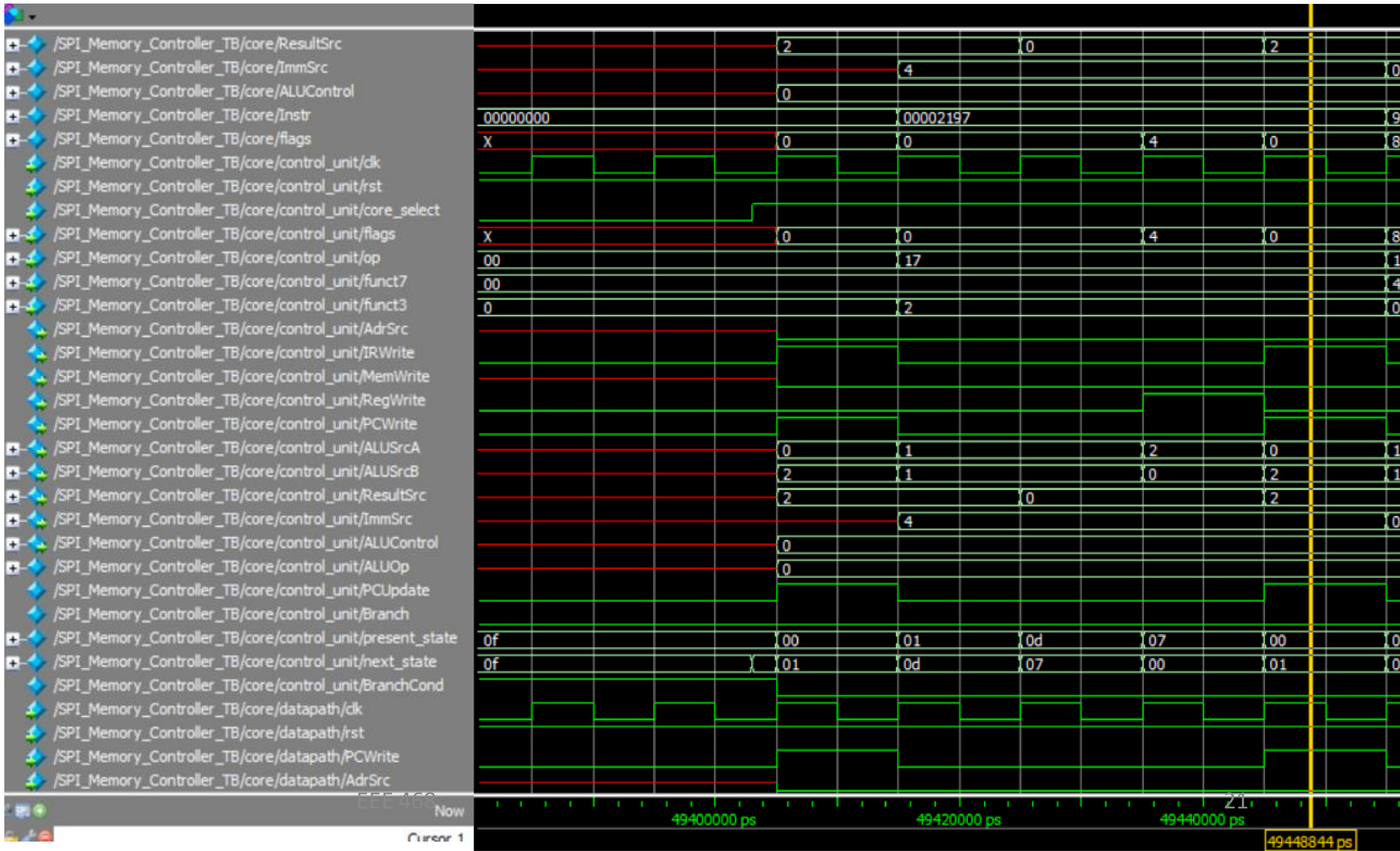
```
/SPI_Memory_Controller_TB/core/rst                        1
/SPI_Memory_Controller_TB/core/core_select                1
/SPI_Memory_Controller_TB/core/ReadData                   94018193
/SPI_Memory_Controller_TB/core/Adr                        00000004
/SPI_Memory_Controller_TB/core/WriteData                  00000000
/SPI_Memory_Controller_TB/core/MemWrite                   0
/SPI_Memory_Controller_TB/core/AdrSrc                     0
/SPI_Memory_Controller_TB/core/IRWrite                    1
/SPI_Memory_Controller_TB/core/RegWrite                   0
/SPI_Memory_Controller_TB/core/PCWrite                    1
/SPI_Memory_Controller_TB/core/ALUSrcA                    0
/SPI_Memory_Controller_TB/core/ALUSrcB                    2
/SPI_Memory_Controller_TB/core/ResultSrc                  2
/SPI_Memory_Controller_TB/core/ImmSrc                     4
/SPI_Memory_Controller_TB/core/ALUControl                 0
/SPI_Memory_Controller_TB/core/Instr                      00002197
/SPI_Memory_Controller_TB/core/flags                      0
/SPI_Memory_Controller_TB/core/control_unit/clk           1
/SPI_Memory_Controller_TB/core/control_unit/rst           1
/SPI_Memory_Controller_TB/core/control_unit/core_select   1
/SPI_Memory_Controller_TB/core/control_unit/flags         0
/SPI_Memory_Controller_TB/core/control_unit/op            17
/SPI_Memory_Controller_TB/core/control_unit/funct7        00
/SPI_Memory_Controller_TB/core/control_unit/funct3        2
/SPI_Memory_Controller_TB/core/control_unit/AdrSrc        0
/SPI_Memory_Controller_TB/core/control_unit/IRWrite       1
/SPI_Memory_Controller_TB/core/control_unit/MemWrite      0
/SPI_Memory_Controller_TB/core/control_unit/RegWrite      0
/SPI_Memory_Controller_TB/core/control_unit/PCWrite       1
/SPI_Memory_Controller_TB/core/control_unit/ALUSrcA       0
/SPI_Memory_Controller_TB/core/control_unit/ALUSrcB       2
/SPI_Memory_Controller_TB/core/control_unit/ResultSrc     2

Now        70000000 ps
Cursor 1   49448844 ps
```

Outputs

```
0:  00002197          auipc  gp,0x2
```

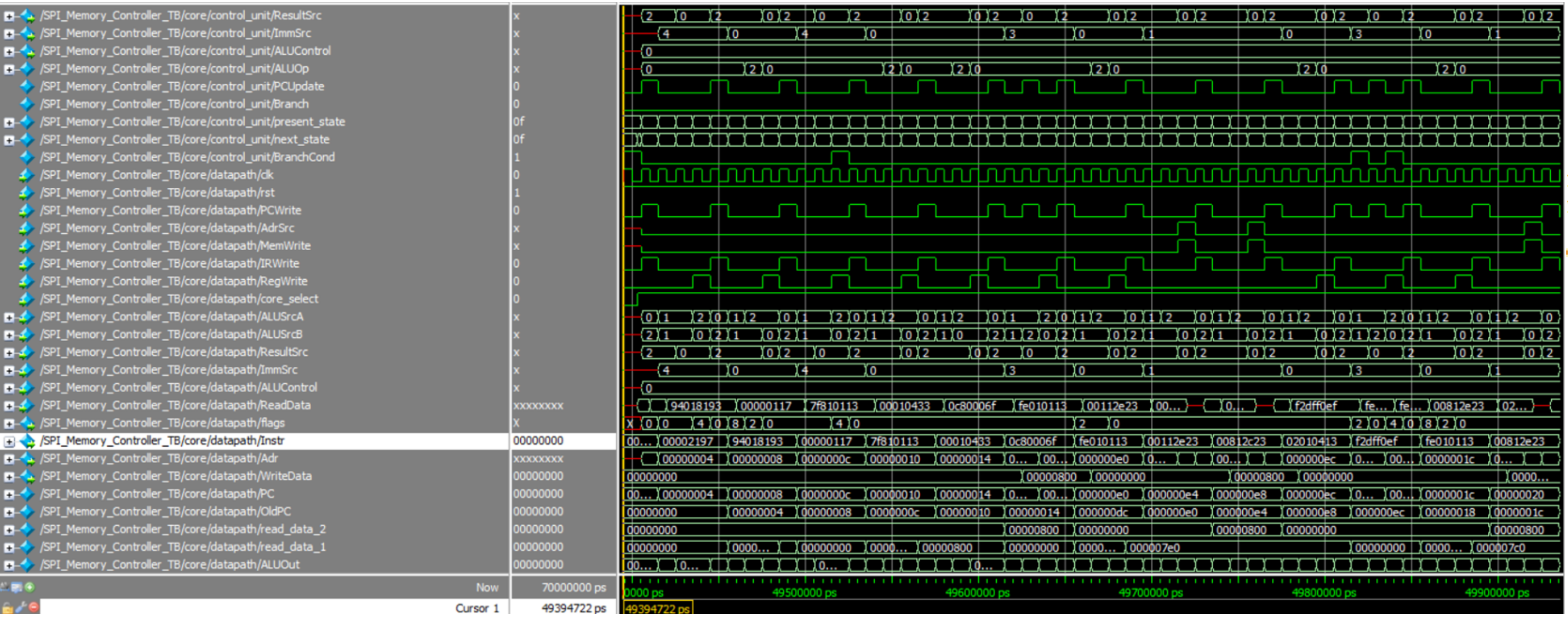




```
/SPI_Memory_Controller_TB/core/ResultSrc                    2  0  2
/SPI_Memory_Controller_TB/core/ImmSrc                       4  0
/SPI_Memory_Controller_TB/core/ALUControl                   0
/SPI_Memory_Controller_TB/core/Instr                        00000000  00002197  9
/SPI_Memory_Controller_TB/core/flags                        X  0  0  4  0  8
/SPI_Memory_Controller_TB/core/control_unit/clk
/SPI_Memory_Controller_TB/core/control_unit/rst
/SPI_Memory_Controller_TB/core/control_unit/core_select
/SPI_Memory_Controller_TB/core/control_unit/flags           X  0  0  4  0  8
/SPI_Memory_Controller_TB/core/control_unit/op              00  17  1
/SPI_Memory_Controller_TB/core/control_unit/funct7          00  4
/SPI_Memory_Controller_TB/core/control_unit/funct3          0  2  0
/SPI_Memory_Controller_TB/core/control_unit/AdrSrc
/SPI_Memory_Controller_TB/core/control_unit/IRWrite
/SPI_Memory_Controller_TB/core/control_unit/MemWrite
/SPI_Memory_Controller_TB/core/control_unit/RegWrite
/SPI_Memory_Controller_TB/core/control_unit/PCWrite
/SPI_Memory_Controller_TB/core/control_unit/ALUSrcA         0  1  2  0  1
/SPI_Memory_Controller_TB/core/control_unit/ALUSrcB         2  1  0  2  1
/SPI_Memory_Controller_TB/core/control_unit/ResultSrc       2  0  2
/SPI_Memory_Controller_TB/core/control_unit/ImmSrc          4  0
/SPI_Memory_Controller_TB/core/control_unit/ALUControl      0
/SPI_Memory_Controller_TB/core/control_unit/ALUOp           0
/SPI_Memory_Controller_TB/core/control_unit/PCUpdate
/SPI_Memory_Controller_TB/core/control_unit/Branch
/SPI_Memory_Controller_TB/core/control_unit/present_state   0f  00  01  0d  07  00  0
/SPI_Memory_Controller_TB/core/control_unit/next_state      0f  01  0d  07  00  01  0
/SPI_Memory_Controller_TB/core/control_unit/BranchCond
/SPI_Memory_Controller_TB/core/datapath/clk
/SPI_Memory_Controller_TB/core/datapath/rst
/SPI_Memory_Controller_TB/core/datapath/PCWrite
/SPI_Memory_Controller_TB/core/datapath/AdrSrc

Now                49400000 ps     49420000 ps     49440000 ps
Cursor 1                                                49448844 ps
```

Outputs



```
/SPI_Memory_Controller_TB/core/control_unit/ResultSrc     x
/SPI_Memory_Controller_TB/core/control_unit/ImmSrc        x
/SPI_Memory_Controller_TB/core/control_unit/ALUControl    x
/SPI_Memory_Controller_TB/core/control_unit/ALUOp         x
/SPI_Memory_Controller_TB/core/control_unit/PCUpdate      0
/SPI_Memory_Controller_TB/core/control_unit/Branch        0
/SPI_Memory_Controller_TB/core/control_unit/present_state 0f
/SPI_Memory_Controller_TB/core/control_unit/next_state    0f
/SPI_Memory_Controller_TB/core/control_unit/BranchCond    1
/SPI_Memory_Controller_TB/core/datapath/clk               0
/SPI_Memory_Controller_TB/core/datapath/rst               1
/SPI_Memory_Controller_TB/core/datapath/PCWrite           0
/SPI_Memory_Controller_TB/core/datapath/AdrSrc            x
/SPI_Memory_Controller_TB/core/datapath/MemWrite          x
/SPI_Memory_Controller_TB/core/datapath/IRWrite           0
/SPI_Memory_Controller_TB/core/datapath/RegWrite          0
/SPI_Memory_Controller_TB/core/datapath/core_select       0
/SPI_Memory_Controller_TB/core/datapath/ALUSrcA           x
/SPI_Memory_Controller_TB/core/datapath/ALUSrcB           x
/SPI_Memory_Controller_TB/core/datapath/ResultSrc         x
/SPI_Memory_Controller_TB/core/datapath/ImmSrc            x
/SPI_Memory_Controller_TB/core/datapath/ALUControl        x
/SPI_Memory_Controller_TB/core/datapath/ReadData          xxxxxxxx
/SPI_Memory_Controller_TB/core/datapath/flags             X
/SPI_Memory_Controller_TB/core/datapath/Instr             00000000
/SPI_Memory_Controller_TB/core/datapath/Adr               xxxxxxxx
/SPI_Memory_Controller_TB/core/datapath/WriteData         00000000
/SPI_Memory_Controller_TB/core/datapath/PC                00000000
/SPI_Memory_Controller_TB/core/datapath/OldPC             00000000
/SPI_Memory_Controller_TB/core/datapath/read_data_2       00000000
/SPI_Memory_Controller_TB/core/datapath/read_data_1       00000000
/SPI_Memory_Controller_TB/core/datapath/ALUOut            00000000

Now       70000000 ps
Cursor 1  49394722 ps
```

```
ReadData:    94018193 00000117 7f810113 00010433 0c80006f fe010113 00112e23 00... 0... f2dff0ef fe... fe... 00812e23 02...
Instr:       00... 00002197 94018193 00000117 7f810113 00010433 0c80006f fe010113 00112e23 00812c23 02010413 f2dff0ef fe010113 00812e23
Adr:         00000004 00000008 0000000c 00000010 00000014 0... 00... 000000e0 0... 00... 000000ec 0... 00... 0000001c 0...
WriteData:   00000000 00000800 00000000 00000800 00000000 0000...
PC:          00... 00000004 00000008 0000000c 00000010 00000014 0... 00... 000000e0 000000e4 000000e8 000000ec 0... 00... 0000001c 00000020
OldPC:       00000000 00000004 00000008 0000000c 00000010 00000014 000000dc 000000e0 000000e4 000000e8 000000ec 00000018 0000001c
read_data_2: 00000000 00000800 00000000 00000800 00000000 00000800
read_data_1: 00000000 0000... 00000000 0000... 00000800 00000000 0000... 000007e0 00000000 0000... 000007c0

0000 ps   49500000 ps   49600000 ps   49700000 ps   49800000 ps   49900000 ps
49394722 ps
```

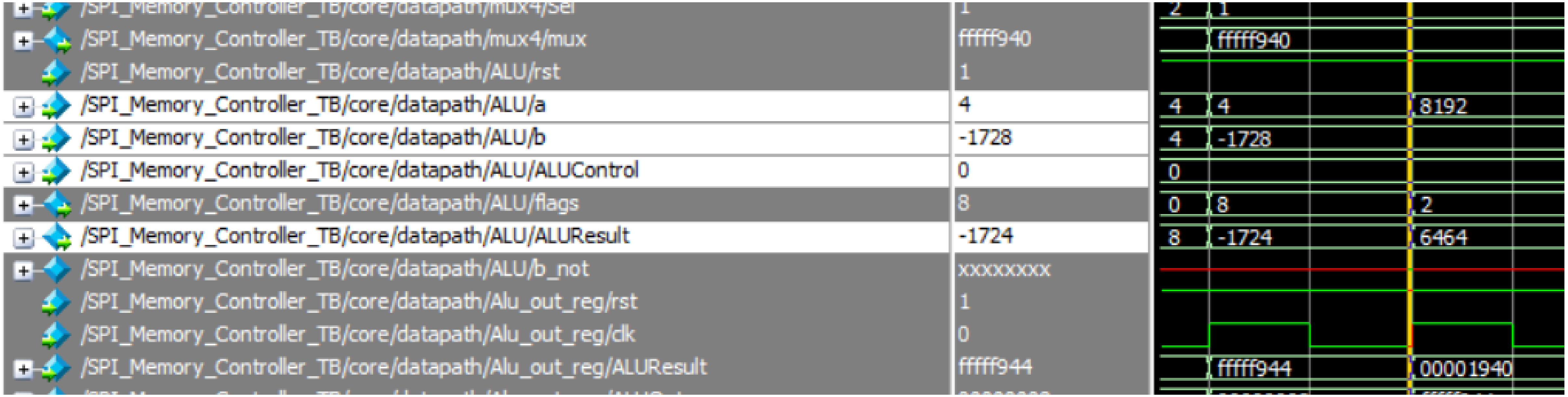
```
 0:  00002197
 4:  94018193
 8:  00000117
 c:  7f810113
10:  00010433
14:  0c80006f

000000dc <main>
 dc:  fe010113
 e0:  00112e23
 e4:  00812c23
 e8:  02010413
 ec:  f2dff0ef

00000018 <prime_
  18:  fe010113
  1c:  00812e23
  20:  02010413
  24:  03200793
  28:  fef42623
```

Outputs

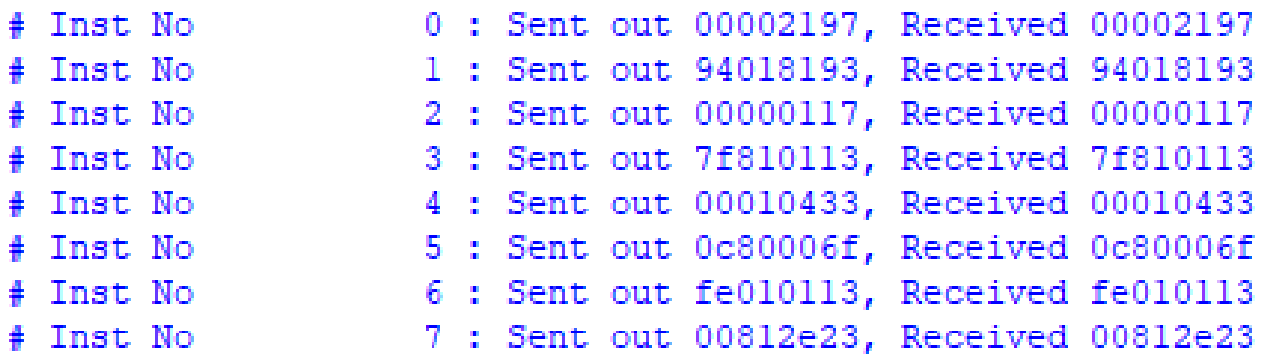
```
4:  94018193          addi   gp,gp,-1728 # 1940 <__global_pointer$>
```



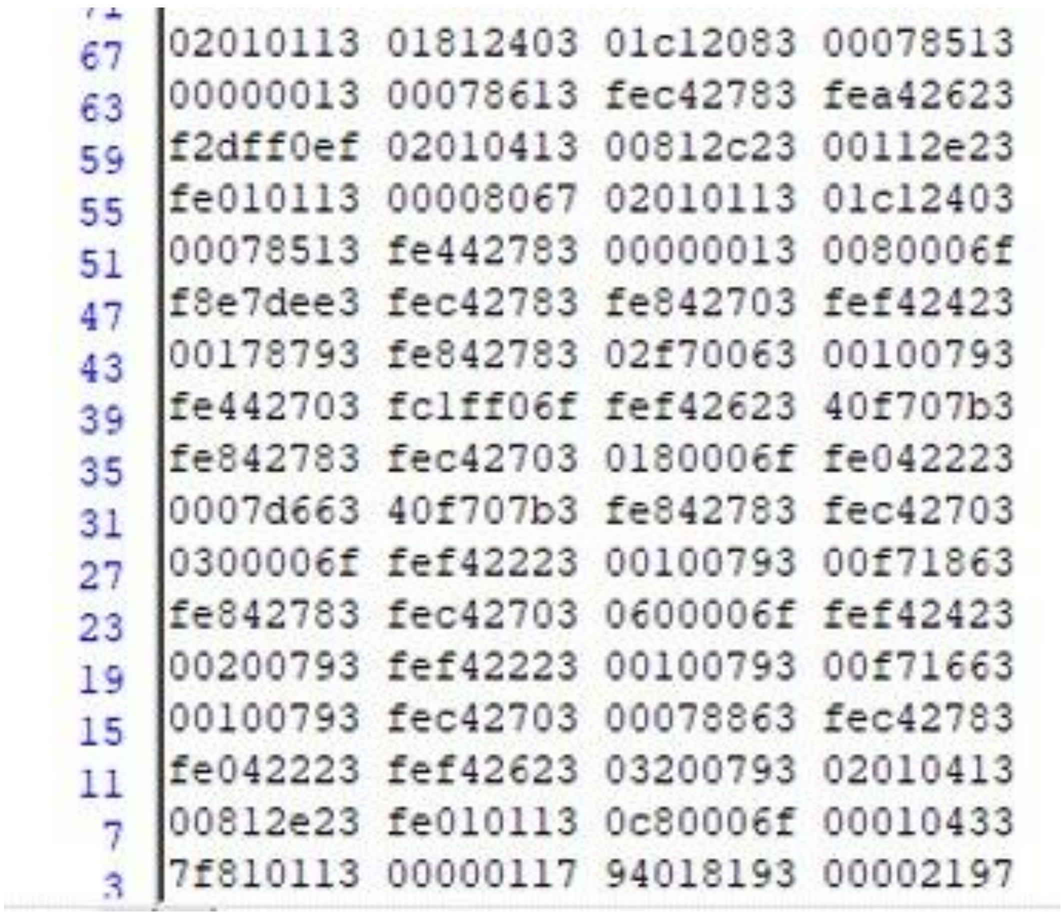
```
/SPI_Memory_Controller_TB/core/datapath/mux4/sel           1          2 1
/SPI_Memory_Controller_TB/core/datapath/mux4/mux           fffff940     fffff940
/SPI_Memory_Controller_TB/core/datapath/ALU/rst            1
/SPI_Memory_Controller_TB/core/datapath/ALU/a              4          4 4              8192
/SPI_Memory_Controller_TB/core/datapath/ALU/b              -1728      4 -1728
/SPI_Memory_Controller_TB/core/datapath/ALU/ALUControl     0          0
/SPI_Memory_Controller_TB/core/datapath/ALU/flags          8          0 8              2
/SPI_Memory_Controller_TB/core/datapath/ALU/ALUResult      -1724      8 -1724          6464
/SPI_Memory_Controller_TB/core/datapath/ALU/b_not          xxxxxxxx
/SPI_Memory_Controller_TB/core/datapath/Alu_out_reg/rst    1
/SPI_Memory_Controller_TB/core/datapath/Alu_out_reg/clk    0
/SPI_Memory_Controller_TB/core/datapath/Alu_out_reg/ALUResult  fffff944   fffff944   00001940
```

Outputs: Detect Non-prime

- DFFRAM memory data after loading the instructions through SPI (Right).
- Verification from TB after SPI slave received data(Bottom)



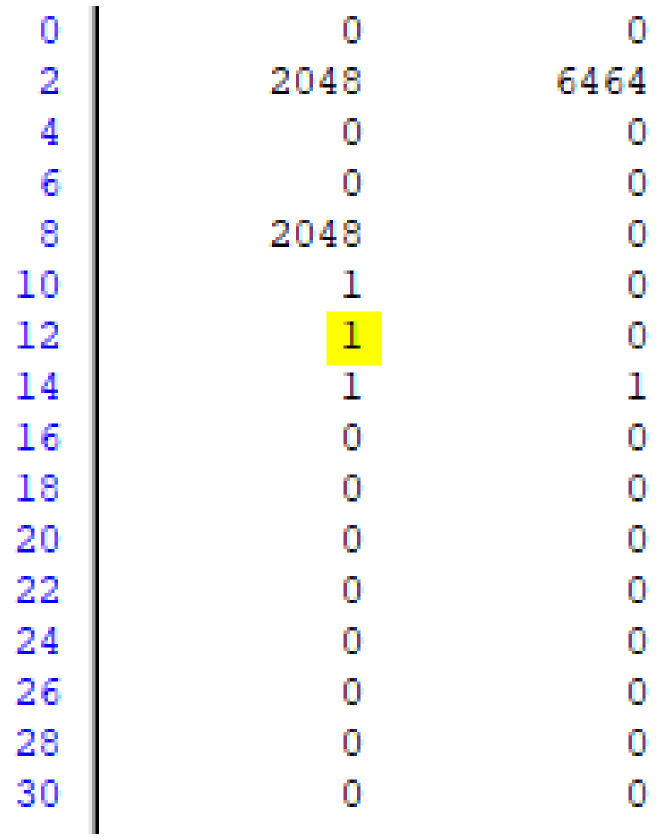
```
# Inst No          0 : Sent out 00002197, Received 00002197
# Inst No          1 : Sent out 94018193, Received 94018193
# Inst No          2 : Sent out 00000117, Received 00000117
# Inst No          3 : Sent out 7f810113, Received 7f810113
# Inst No          4 : Sent out 00010433, Received 00010433
# Inst No          5 : Sent out 0c80006f, Received 0c80006f
# Inst No          6 : Sent out fe010113, Received fe010113
# Inst No          7 : Sent out 00812e23, Received 00812e23
```



```
67 |02010113 01812403 01c12083 00078513
63 |00000013 00078613 fec42783 fea42623
59 |f2dff0ef 02010413 00812c23 00112e23
55 |fe010113 00008067 02010113 01c12403
51 |00078513 fe442783 00000013 0080006f
47 |f8e7dee3 fec42783 fe842703 fef42423
43 |00178793 fe842783 02f70063 00100793
39 |fe442703 fc1ff06f fef42623 40f707b3
35 |fe842783 fec42703 0180006f fe042223
31 |0007d663 40f707b3 fe842783 fec42703
27 |0300006f fef42223 00100793 00f71863
23 |fe842783 fec42703 0600006f fef42423
19 |00200793 fef42223 00100793 00f71663
15 |00100793 fec42703 00078863 fec42783
11 |fe042223 fef42623 03200793 02010413
 7 |00812e23 fe010113 0c80006f 00010433
 3 |7f810113 00000117 94018193 00002197
```

Outputs: Detect Non-prime

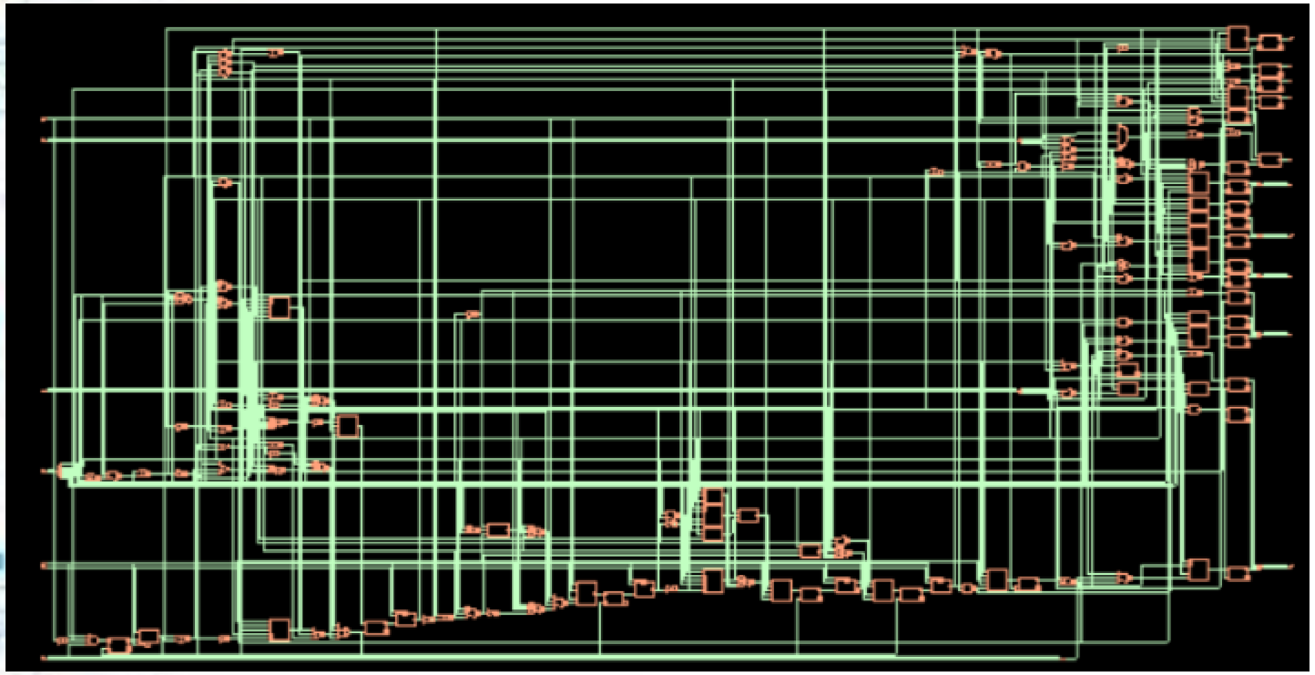
- Register File contents state after execution of program
- Return value stored in R12
  - 1 = non-prime
  - 0 = prime
- Test done on a = 50
  - Result shows non-prime



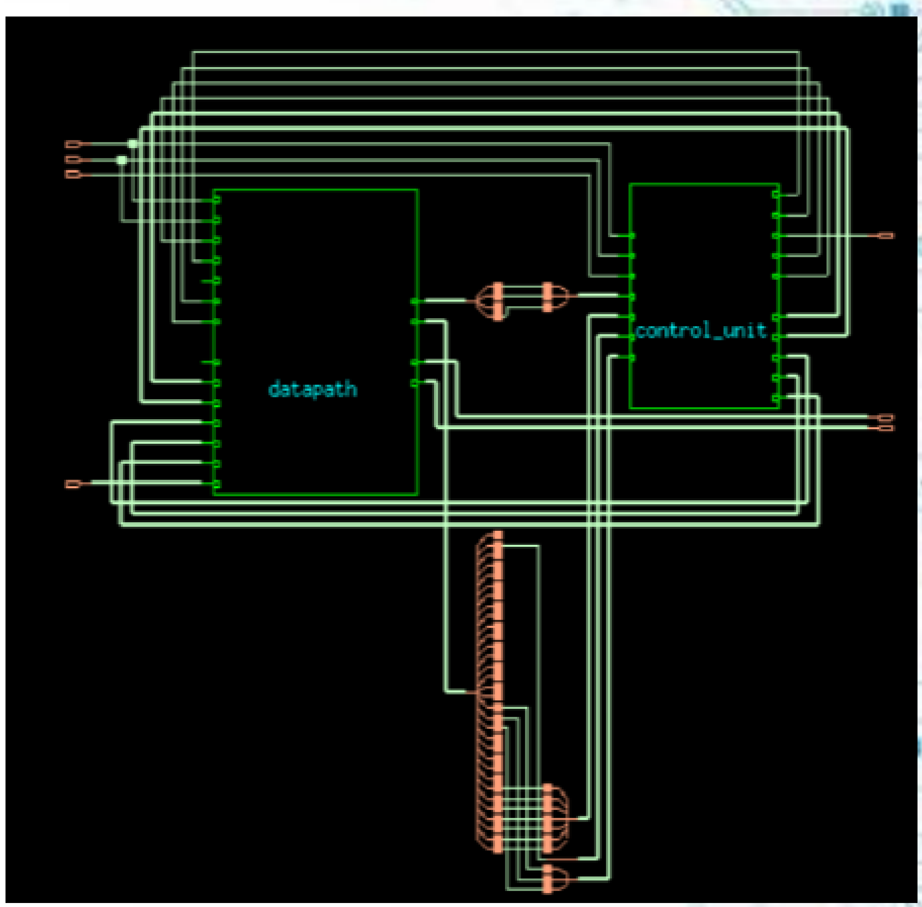
```
 0 |    0     0
 2 | 2048  6464
 4 |    0     0
 6 |    0     0
 8 | 2048     0
10 |    1     0
12 |    1     0
14 |    1     1
16 |    0     0
18 |    0     0
20 |    0     0
22 |    0     0
24 |    0     0
26 |    0     0
28 |    0     0
30 |    0     0
```

Synthesis

Gate Level Diagram - RISC-V Core



Control Unit of the Core



Top Module of the Core

Physical Design of RV32i Core

- We could not obtain the Physical Design, Clock Tree Synthesis and the Subsequent Steps due to the recent Server Issue.

Acknowledgements

To Dr. A.B.M Harun-Ur-Rashid Sir

Professor, Department of EEE, BUET

and

Mumtahina Islam Sukanya

Lecturer, Department of EEE, BUET

To A.K.M Uday Hasan Bhuiyan

Lead DFT Engineer, Neural Semiconductor Limited

and his team

for this collaborative opportunity

References

1. Digital Design and Computer Architecture, RISC-V Edition
   Sarah L. Harris, David Harris
2. The RISC-V Instruction Set Manual Volume I: User-Level ISA

THANK YOU

FEEL FREE TO ASK ANY QUESTIONS