



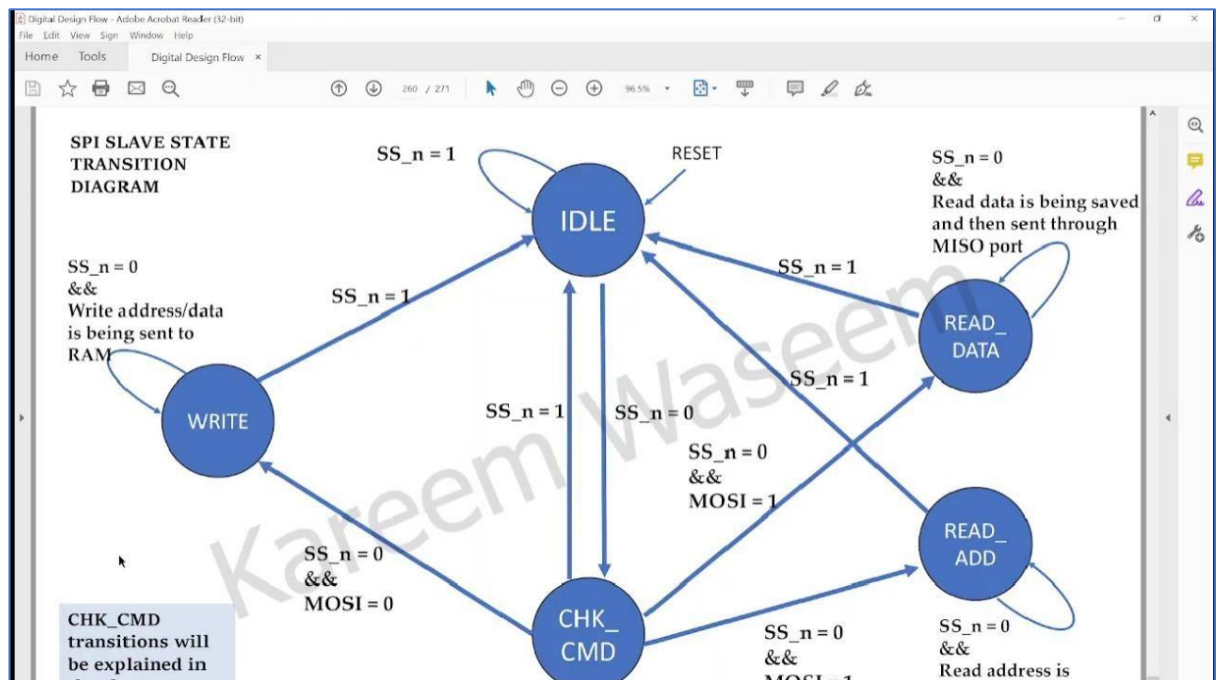
Digital IC Design SPI Slave with Single Port RAM

Supervised by Eng. Kareem Waseem

1.0 QUESTION

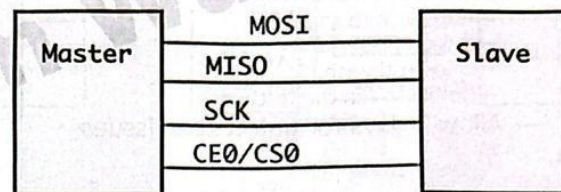
RAM Write Command – Write Address

1. Master will start the write command by sending the write address value, $rx_data[9:8] = din[9:8] = 2'b00$
2. $SS_n = 0$ to tell the SPI Slave that the master will begin communication
3. SPI Slave check the first received bit on MOSI port '0' which is a control bit to let the slave determine which operation will take place "write in this case". SPI Slave then expects to receive 10 more bits, the first 2 bits are "00" on two clock cycles and then the $wr_address$ will be sent on 8 more clock cycles
4. Now the data is converted from serial "MOSI" to parallel after writing the $rx_data[9:0]$ bus
5. rx_valid will be HIGH to inform the RAM that it should expect data on din bus
6. din takes the value of rx_data
7. RAM checks on $din[9:8]$ and find that they hold "00"
8. RAM stores $din[7:0]$ in the internal write address bus
9. $SS_n = 1$ to end communication from Master side



SPI Interface

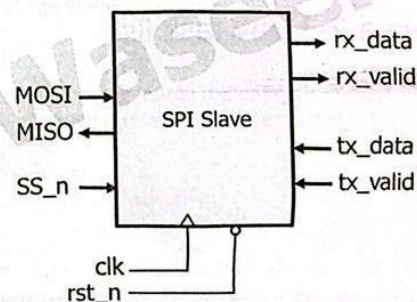
- One of the most popular Interfaces nowadays
- Stands for Serial-Peripheral Interface
- Four Wires
 - MOSI: Master-Out-Slave-In
 - MISO: Master-In-Slave-Out
 - SCK: Clock
 - SS_n: Slave Select
- High Data Rates



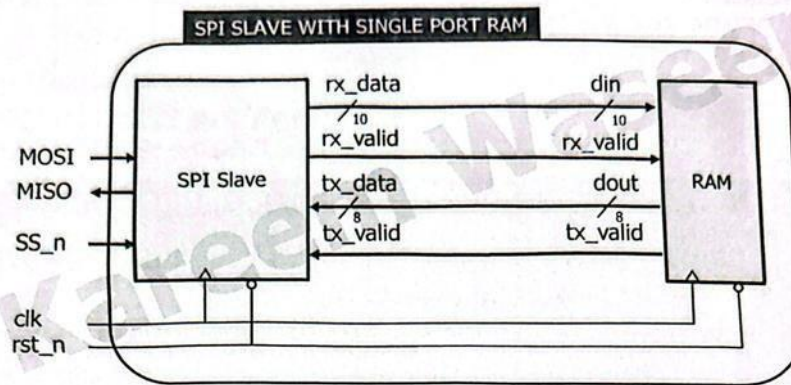
299

Project: 1- SPI Slave Interface

- One of the most popular Interfaces nowadays
- Stands for Serial-Peripheral Interface
- Four Wires
 - MOSI: Master-Out-Slave-In
 - MISO: Master-In-Slave-Out
 - SCK: Clock
 - SS_n: Slave Select
- High Data Rates

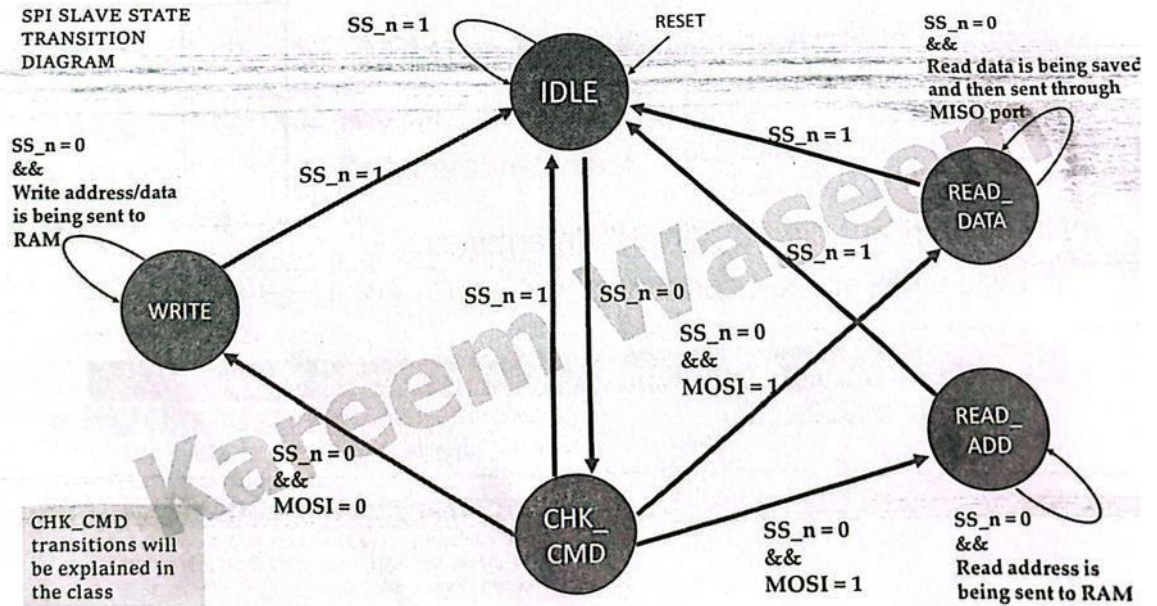


Project: 3- SPI Wrapper



303

SPI SLAVE STATE TRANSITION DIAGRAM



2.0 CODE

2.1 Ram

```
1  module RAM_MODULE (clk,rst_n,rx_valid,din,dout,tx_valid);
2
3      parameter MEM_DEPTH = 256;
4      parameter ADDR_SIZE = 8;
5
6      input clk;
7      input rst_n;
8      input rx_valid;
9      input [9:0] din;
10
11     output reg [7:0] dout;
12     output reg tx_valid;
13
14     reg [ADDR_SIZE-1:0] mem [MEM_DEPTH-1:0];
15     reg [7:0] addr_wr;
16     reg [7:0] addr_re;
17
18
19     always @(posedge clk) begin
20         if (!rst_n) begin
21             dout <= 8'b0;
22             tx_valid <= 1'b0;
23             addr_wr <= 8'b0;
24             addr_re <= 8'b0;
25         end else begin
26             if (rx_valid) begin
27                 case (din[9:8])
28                     2'b00: begin
29                         addr_wr <= din[7:0];
30                         tx_valid <= 1'b0;
31                     end
32                     2'b01: begin
33                         mem[addr_wr] <= din[7:0];
34                         tx_valid <= 1'b0;
35                     end
36                     2'b10: begin
37                         addr_re <= din[7:0];
38                         tx_valid <= 1'b0;
39                     end
40                     2'b11: begin
41                         dout <= mem[addr_re];
42                         tx_valid <= 1'b1;
43                     end
44                 endcase
45             end
46         end
47     end
48 endmodule
```

2.2 SPI Module

2.2.1 Definition

```
1 module SPI_MODULE(clk,rst_n,SS_n,MOSI,tx_valid,tx_data,MISO,rx_valid,rx_data);
2     input MOSI , tx_valid , clk , rst_n , SS_n ;
3     input [7:0] tx_data ;
4     output reg MISO , rx_valid ;
5     output reg [9:0] rx_data;
6     //////////////////////////////////////
7     parameter IDLE = 3'b000;
8     parameter CHK_CMD = 3'b001;
9     parameter WRITE = 3'b010;
10    parameter READ_ADD = 3'b011;
11    parameter READ_DATA = 3'b100;
12
13    reg [2:0] cs , ns ;
14    reg ADD_DATA_checker ;
15    reg [3:0] counter1 ;
16    reg [2:0] counter2 ;
17    reg [9:0] bus ;
```

2.2.2 State Memory

```
1 always @(posedge clk)
2 begin
3     if(~rst_n)
4         cs <= IDLE;
5     else
6         cs <= ns ;
7 end
```

2.2.3 Output Logic

```
1  always @(posedge clk or negedge rst_n) begin
2      if (~rst_n) begin
3          counter1 <= 9;
4          counter2 <= 7;
5          ADD_DATA_checker <= 1;
6          bus <= 0;
7          rx_data <= 0;
8          rx_valid <= 0;
9          MISO <= 0;
10     end
11
12     else begin
13         if(cs == IDLE) begin
14             rx_valid <= 0;
15             counter1 <= 9 ;
16             counter2 <= 7 ;
17         end
18
19         else if(cs == WRITE) begin
20             if (counter1 >= 0)begin
21                 bus[counter1] <= MOSI;
22                 counter1 <= counter1 - 1;
23             end
24             if(counter1 == 4'b1111) begin
25                 rx_valid = 1;
26                 rx_data <= bus ;
27             end
28         end
29
30         else if (cs == READ_ADD) begin
31             if (counter1 >= 0)begin
32                 bus[counter1] <= MOSI;
33                 counter1 <= counter1 - 1;
34             end
35             if(counter1 == 4'b1111) begin
36                 rx_valid <= 1;
37                 rx_data <= bus ;
38                 ADD_DATA_checker <= 0;
39             end
40         end
41
42         else if (cs == READ_DATA) begin
43             if (counter1 >= 0)begin
44                 bus[counter1] <= MOSI;
45                 counter1 <= counter1 - 1;
46             end
47             if(counter1 == 4'b1111) begin
48                 rx_valid <= 1;
49                 rx_data <= bus ;
50                 counter1 <= 9 ;
51             end
52             if(rx_valid == 1) rx_valid <= 0;
53             if(tx_valid==1 && counter2 >=0)begin
54                 MISO <= tx_data[counter2] ;
55                 counter2 <= counter2 - 1 ;
56             end
57             if(counter2 == 3'b111)begin
58                 ADD_DATA_checker <= 1;
59             end
60         end
61     end
62 end
```

2.2.4 Next State Logic

```
1  always @(*) begin
2      ns = cs ;
3      case(cs)
4          IDLE : begin
5              if(SS_n)
6                  ns = IDLE;
7              else
8                  ns = CHK_CMD;
9          end
10         CHK_CMD : begin
11             if(SS_n)
12                 ns = IDLE;
13             else begin
14                 if((~SS_n) && (MOSI == 0))
15                     ns = WRITE;
16                 else if ((~SS_n) && (MOSI == 1) && (ADD_DATA_checker == 1))
17                     ns = READ_ADD;
18                 else if ((~SS_n) && (MOSI == 1) && (ADD_DATA_checker == 0))
19                     ns = READ_DATA;
20             end
21         end
22         WRITE : begin
23             if(SS_n || counter1 == 4'b1111)
24                 ns = IDLE;
25             else
26                 ns = WRITE;
27         end
28         READ_ADD : begin
29             if(SS_n || counter1 == 4'b1111)
30                 ns = IDLE;
31             else
32                 ns = READ_ADD;
33         end
34         READ_DATA : begin
35             if(SS_n)
36                 ns = IDLE;
37             else
38                 ns = READ_DATA;
39         end
40     endcase
41 end
42
43 endmodule
```


2.3 SPI Wrapper

```
1 module SPI_Wrapper (clk,ss_n,MOSI,MISO,rst_n);
2   parameter MEM_DEPTH = 256 ;
3   parameter ADDR_SIZE = 8;
4   input clk , ss_n , MOSI , rst_n;
5   output MISO;
6   wire [9:0] rxdata ;
7   wire [7:0] txdata ;
8   wire rx_valid , tx_valid ;
9
10  SPI_MODULE SPI(.MOSI(MOSI),.MISO(MISO),.clk(clk),.SS_n(ss_n),.rst_n(rst_n)
11  ,.rx_data(rxdata),.tx_data(txdata),.rx_valid(rx_valid),.tx_valid(tx_valid));
12
13  RAM_MODULE #(.MEM_DEPTH(MEM_DEPTH),.ADDR_SIZE(ADDR_SIZE)) RAM (.din(rxdata),.dout(txdata),.clk(clk)
14  ,.rx_valid(rx_valid),.tx_valid(tx_valid),.rst_n(rst_n));
15 endmodule
16
```

2.4 Testbench

2.4.1 Definition

```
1 module test();
2   reg clk,rst_n,ss_n,mosi;
3   wire miso;
4   SPI_Wrapper DUT(.clk(clk),.rst_n(rst_n),.MOSI(mosi),.MISO(miso),.ss_n(ss_n));
5   reg [9:0]temp;
6   integer i=0;
7
8   initial begin
9     clk=0;
10    forever
11      #1 clk=~clk;
12    end
13
```

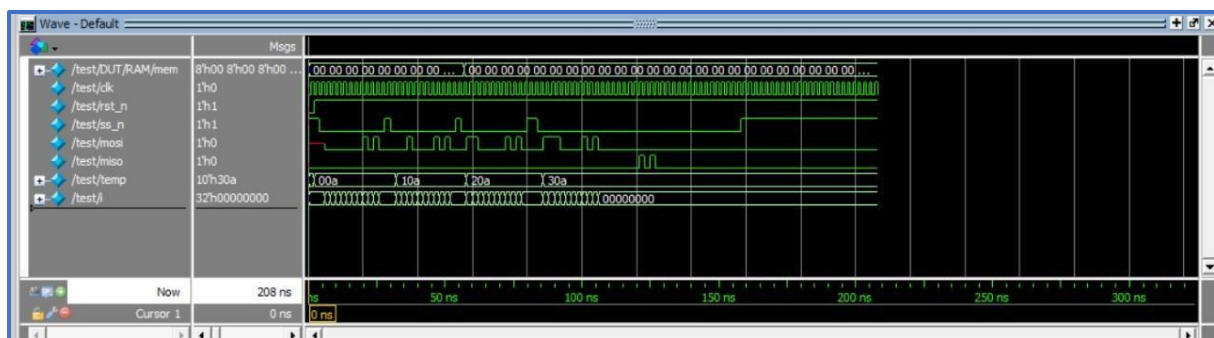
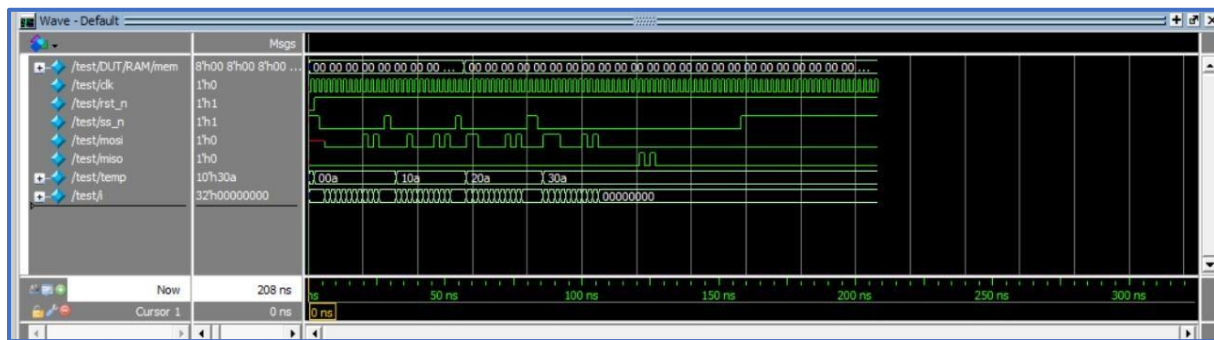
2.4.2 Main stimulus process

```
1  initial begin
2  $readmemh("mem.dat",DUT.RAM.mem);
3  rst_n=0;
4  ss_n=1;
5  temp=0;
6  @(negedge clk);
7  temp=10'b0000_01010;
8  rst_n=1;
9
10
11  @(negedge clk)begin
12  ss_n=0;
13  end
14  @(negedge clk)begin
15  mosi=0;
16  end
17  for(i=10;i>0;i=i-1)begin
18  @(negedge clk)
19      mosi=temp[i-1];
20  end
21
22  @(negedge clk)
23  ss_n=1;
24
25  @(negedge clk)
26  ss_n=0;
27  @(negedge clk)begin
28  mosi=0;
29
30  temp=10'b01000_01010;
31  end
```



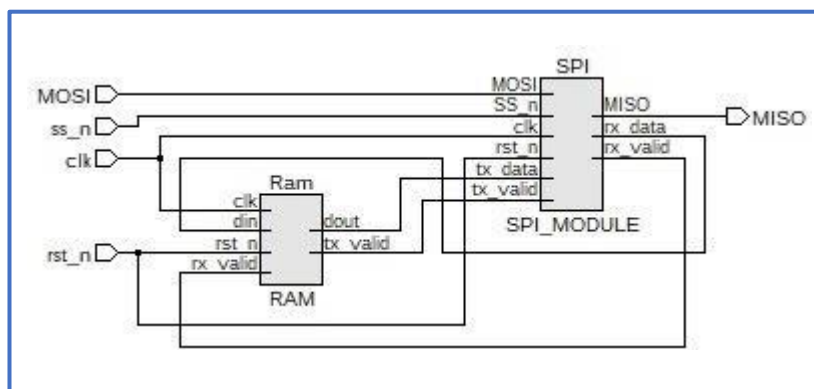
```
1  for(i=10;i>0;i=i-1)begin
2  @(negedge clk)
3      mosi=temp[i-1];
4  end
5  @(negedge clk)
6  ss_n=1;
7
8  @(negedge clk)
9  ss_n=0;
10 @(negedge clk)begin
11 mosi=1;
12
13 temp=10'b10000_01010;
14 end
15 for(i=10;i>0;i=i-1)begin
16 @(negedge clk)
17     mosi=temp[i-1];
18 end
19 @(negedge clk)
20 ss_n=1;
21 @(negedge clk)
22 ss_n=1;
23
24 @(negedge clk)
25 ss_n=0;
26 @(negedge clk)begin
27 mosi=1;
28
29 temp=10'b11000_01010;
30 end
31 for(i=10;i>0;i=i-1)begin
32 @(negedge clk)
33     mosi=temp[i-1];
34 end
35
36 repeat(25)@(negedge clk);
37 @(negedge clk)
38 ss_n=1;
39 repeat(25)@(negedge clk);
40 $stop;
41 end
42
43 initial begin
44     $monitor("mosi=%b,miso=%b,ss_n=%b",mosi,miso,ss_n);
45 end
46 endmodule
```

3.0 WAVEFORM



4.0 LINTING

4.1 Schematic



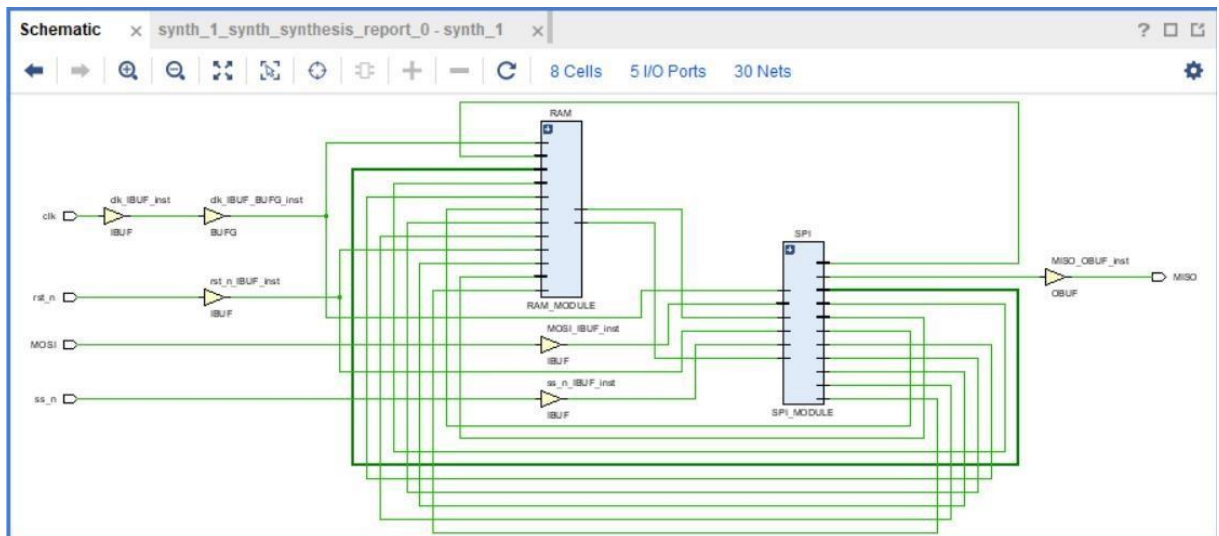
4.2 Errors

```
# End time: 22:26:24 on Jul 31,2025, Elapsed time: 0:00:00
#
# Errors: 0, Warnings: 0
#
```


5.0 SYNTHESIS

5.1 Grey Coding

5.1.1 Schematic



5.1.2 Synthesis Report

```
Schematic x synth_1_synth_synthesis_report_0 - synth_1 x
C:/Users/Mina/Downloads/digital_ic_design/project2_vivado_gray/project_2/runs/synth_1/SPI_Wrapper.vds
Read-only

20 Starting RTL Elaboration : Time (s): cpu = 00:00:02 ; elapsed = 00:00:03 . Memory (MB): peak = 356.004 ; gain = 99.176
21 -----
22 INFO: [Synth 8-6157] synthesizing module 'SPI_Wrapper' [C:/Users/Mina/Downloads/digital_ic_design/project2_vivado_gray/
23   Parameter MEM_DEPTH bound to: 256 - type: integer
24   Parameter ADDR_SIZE bound to: 8 - type: integer
25 INFO: [Synth 8-6157] synthesizing module 'SPI_MODULE' [C:/Users/Mina/Downloads/digital_ic_design/project2_vivado_gray/S
26   Parameter IDLE bound to: 3'b000
27   Parameter CHK_CMD bound to: 3'b001
28   Parameter WRITE bound to: 3'b010
29   Parameter READ_ADD bound to: 3'b011
30   Parameter READ_DATA bound to: 3'b100
31 INFO: [Synth 8-5534] Detected attribute (* fsm_encoding = "gray" *) [C:/Users/Mina/Downloads/digital_ic_design/project2
32 INFO: [Synth 8-155] case statement is not full and has no default [C:/Users/Mina/Downloads/digital_ic_design/project2_v
33 INFO: [Synth 8-6155] done synthesizing module 'SPI_MODULE' (1#1) [C:/Users/Mina/Downloads/digital_ic_design/project2_vi
34 INFO: [Synth 8-6157] synthesizing module 'RAM_MODULE' [C:/Users/Mina/Downloads/digital_ic_design/project2_vivado_gray/R
35   Parameter MEM_DEPTH bound to: 256 - type: integer
```

5.1.3 Timing Report

Tcl ConsoleMessagesLogReportsDesign RunsTiming xDebug

Design Timing Summary

General Information

Timer Settings

Design Timing Summary

Clock Summary (1)

Check Timing (4)

Intra-Clock Paths

Inter-Clock Paths

Other Path Groups

User Ignored Paths

Unconstrained Paths

Setup

Hold

Pulse Width

Worst Negative Slack (WNS): 6.261 ns

Total Negative Slack (TNS): 0.000 ns

Number of Failing Endpoints: 0

Total Number of Endpoints: 108

Worst Hold Slack (WHS): 0.146 ns

Total Hold Slack (THS): 0.000 ns

Number of Failing Endpoints: 0

Total Number of Endpoints: 108

Worst Pulse Width Slack (WPWS): 4.500 ns

Total Pulse Width Negative Slack (TPWS): 0.000 ns

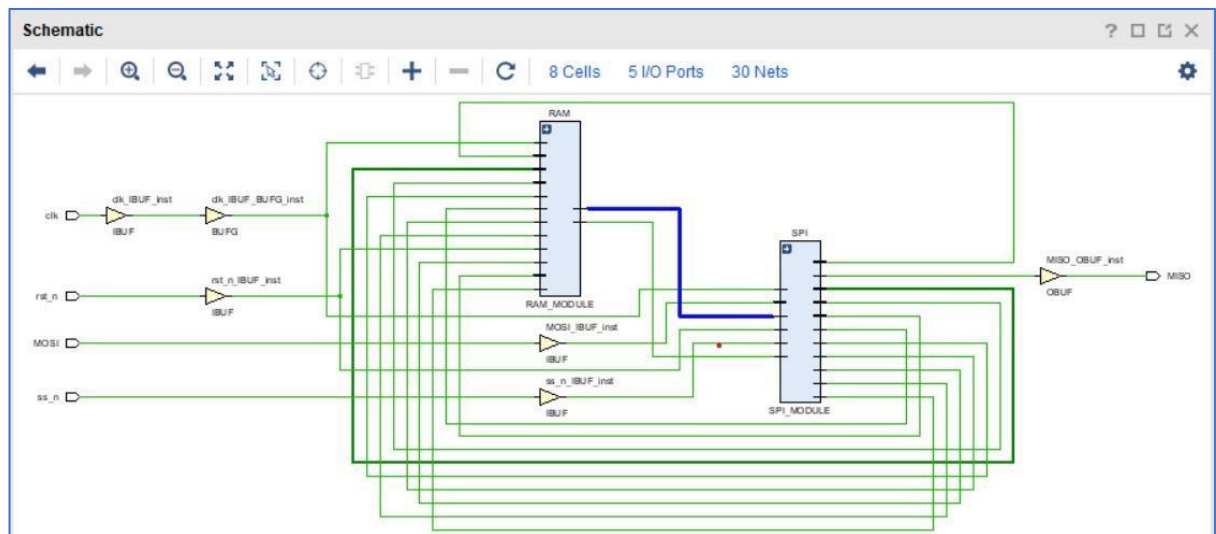
Number of Failing Endpoints: 0

Total Number of Endpoints: 53

All user specified timing constraints are met.

Timing Summary - timing_1

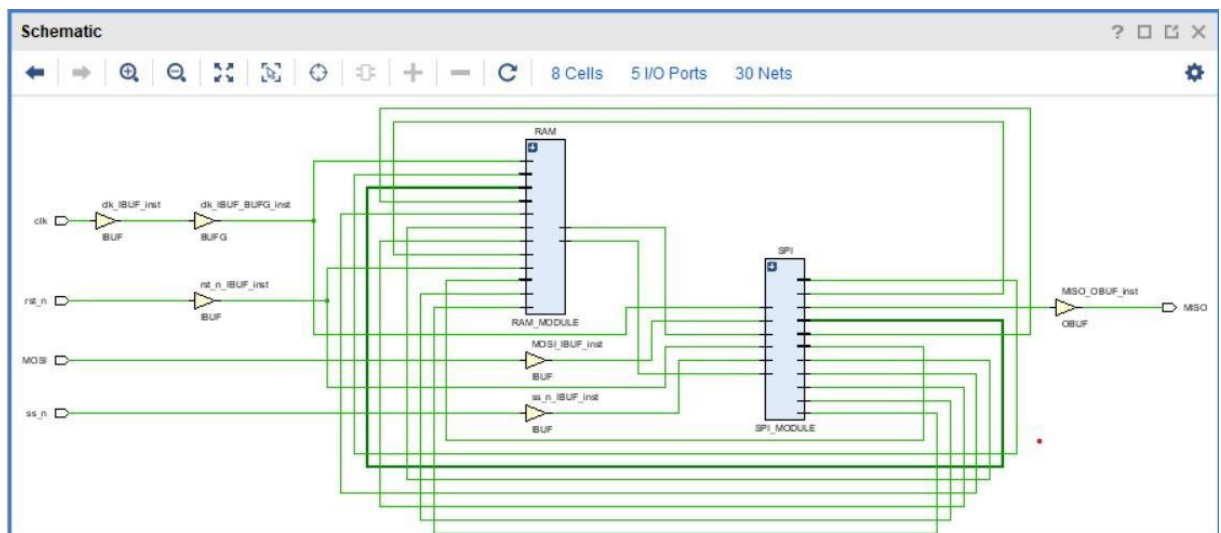
5.1.4 Critical Path



Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source ...
Path 1	6.261	2	3	1	RAM/mem_reg/CLKBWRCLK	SPI/MISO_reg/D	3.623	2.823	0.800	10.0	sys_clk_pin

5.2 One Hot

5.2.1 Schematic



5.2.2 Synthesis Report

```

22 INFO: [Synth 8-6157] synthesizing module 'SPI_Wrapper' [C:/Users/Mina/Downloads/digital_ic_design/project2_vivado_one_h
23   Parameter MEM_DEPTH bound to: 256 - type: integer
24   Parameter ADDR_SIZE bound to: 8 - type: integer
25 INFO: [Synth 8-6157] synthesizing module 'SPI_MODULE' [C:/Users/Mina/Downloads/digital_ic_design/project2_vivado_one_ho
26   Parameter IDLE bound to: 3'b000
27   Parameter CHK_CMD bound to: 3'b001
28   Parameter WRITE bound to: 3'b010
29   Parameter READ_ADD bound to: 3'b011
30   Parameter READ_DATA bound to: 3'b100
31 INFO: [Synth 8-5534] Detected attribute (* fsm_encoding = "one_hot" *) [C:/Users/Mina/Downloads/digital_ic_design/proje
32 INFO: [Synth 8-155] case statement is not full and has no default [C:/Users/Mina/Downloads/digital_ic_design/project2_v
33 INFO: [Synth 8-6155] done synthesizing module 'SPI_MODULE' (1#1) [C:/Users/Mina/Downloads/digital_ic_design/project2_vi
34 INFO: [Synth 8-6157] synthesizing module 'RAM_MODULE' [C:/Users/Mina/Downloads/digital_ic_design/project2_vivado_one_ho
35   Parameter MEM_DEPTH bound to: 256 - type: integer
36   Parameter ADDR_SIZE bound to: 8 - type: integer
37 INFO: [Synth 8-6155] done synthesizing module 'RAM_MODULE' (2#1) [C:/Users/Mina/Downloads/digital_ic_design/project2_vi

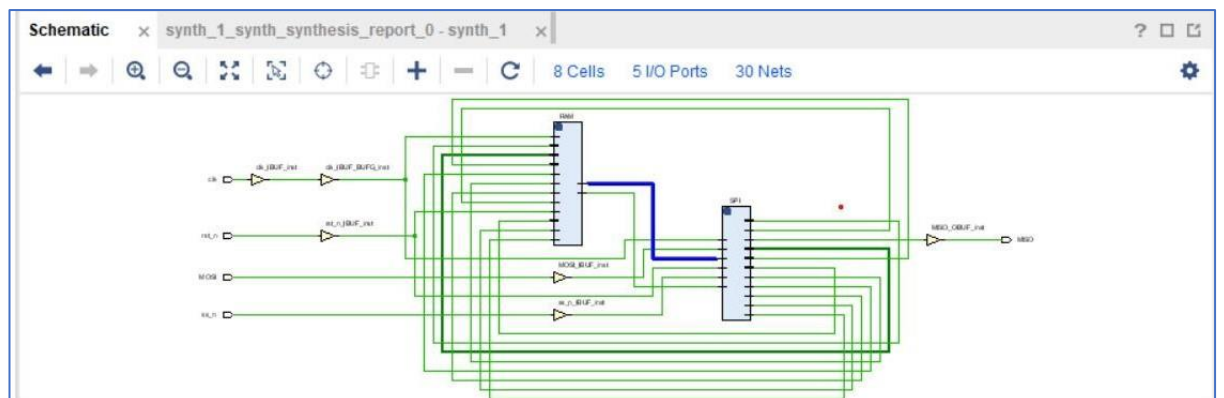
```

5.2.3 Timing Report

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): 6.261 ns	Worst Hold Slack (WHS): 0.148 ns	Worst Pulse Width Slack (WPWS): 4.500 ns
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns	Total Pulse Width Negative Slack (TPWS): 0.000 ns
Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	Number of Failing Endpoints: 0
Total Number of Endpoints: 115	Total Number of Endpoints: 115	Total Number of Endpoints: 55

All user specified timing constraints are met.

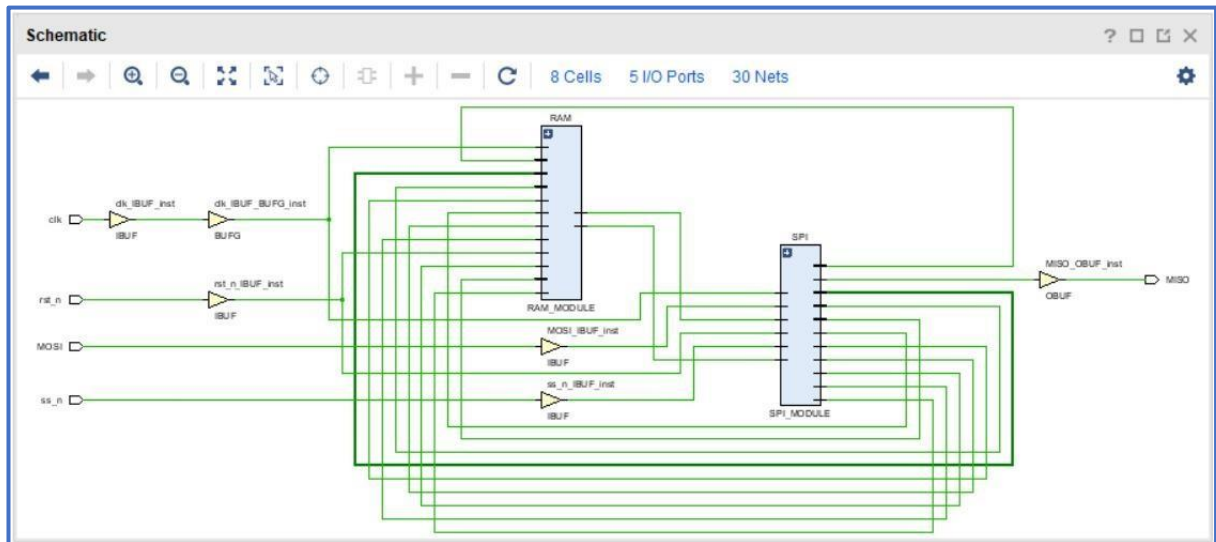
5.2.4 Critical Path



Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement
Path 1	6.261	2	3	1	RAM/mem_reg/CLKBWRCLK	SPI/MISO_reg/D	3.623	2.823	0.800	10.0

5.3 Seq

5.3.1 Schematic



5.3.2 Synthesis Report

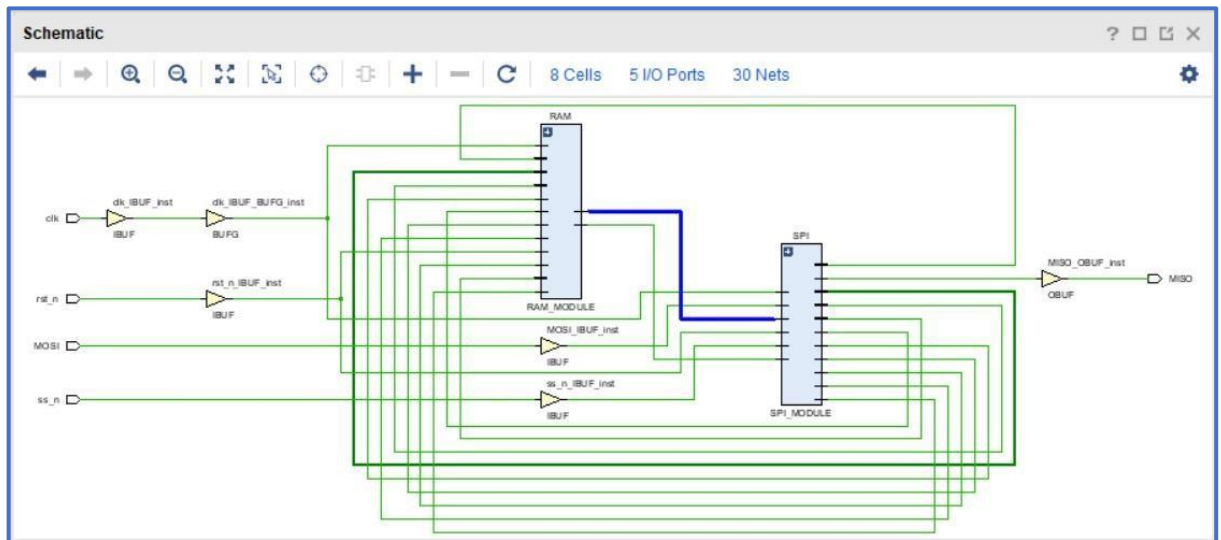
```
Schematic x synth_1_synth_synthesis_report_0 - synth_1 x
C:/Users/Mina/Downloads/digital_ic_design/project2_seq/project_1/project_1.runs/synth_1/SPI_Wrapper.vds

24 Parameter ADDR_SIZE bound to: 8 - type: integer
25 INFO: [Synth 8-6157] synthesizing module 'SPI_MODULE' [C:/Users/Mina/Downloads/digital_ic_design/project2_seq/SPI.v:1]
26 Parameter IDLE bound to: 3'b000
27 Parameter CHK_CMD bound to: 3'b001
28 Parameter WRITE bound to: 3'b010
29 Parameter READ_ADD bound to: 3'b011
30 Parameter READ_DATA bound to: 3'b100
31 INFO: [Synth 8-5534] Detected attribute (* fsm_encoding = "seq" *) [C:/Users/Mina/Downloads/digital_ic_design/project2_
32 INFO: [Synth 8-155] case statement is not full and has no default [C:/Users/Mina/Downloads/digital_ic_design/project2_s
33 INFO: [Synth 8-6155] done synthesizing module 'SPI_MODULE' (1#1) [C:/Users/Mina/Downloads/digital_ic_design/project2_se
34 INFO: [Synth 8-6157] synthesizing module 'RAM_MODULE' [C:/Users/Mina/Downloads/digital_ic_design/project2_seq/RAM.v:1]
35 Parameter MEM_DEPTH bound to: 256 - type: integer
36 Parameter ADDR_SIZE bound to: 8 - type: integer
37 INFO: [Synth 8-6155] done synthesizing module 'RAM_MODULE' (2#1) [C:/Users/Mina/Downloads/digital_ic_design/project2_se
38 INFO: [Synth 8-6155] done synthesizing module 'SPI_Wrapper' (3#1) [C:/Users/Mina/Downloads/digital_ic_design/project2_s
39
```

5.3.3 Timing Report

Design Timing Summary			
Setup	Hold	Pulse Width	
Worst Negative Slack (WNS): 6.261 ns	Worst Hold Slack (WHS): 0.146 ns	Worst Pulse Width Slack (WPWS): 4.500 ns	
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns	Total Pulse Width Negative Slack (TPWS): 0.000 ns	
Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	
Total Number of Endpoints: 108	Total Number of Endpoints: 108	Total Number of Endpoints: 53	
All user specified timing constraints are met.			

5.3.4 Critical Path



Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Clock
Path 1	6.261	2	3	1	RAM/mem_reg/CLKBWRCLK	SPI/MISO_reg/D	3.623	2.823	0.800	10.0	sys_clk_pin	sys_clk_pin

6.0 IMPLEMENTATION

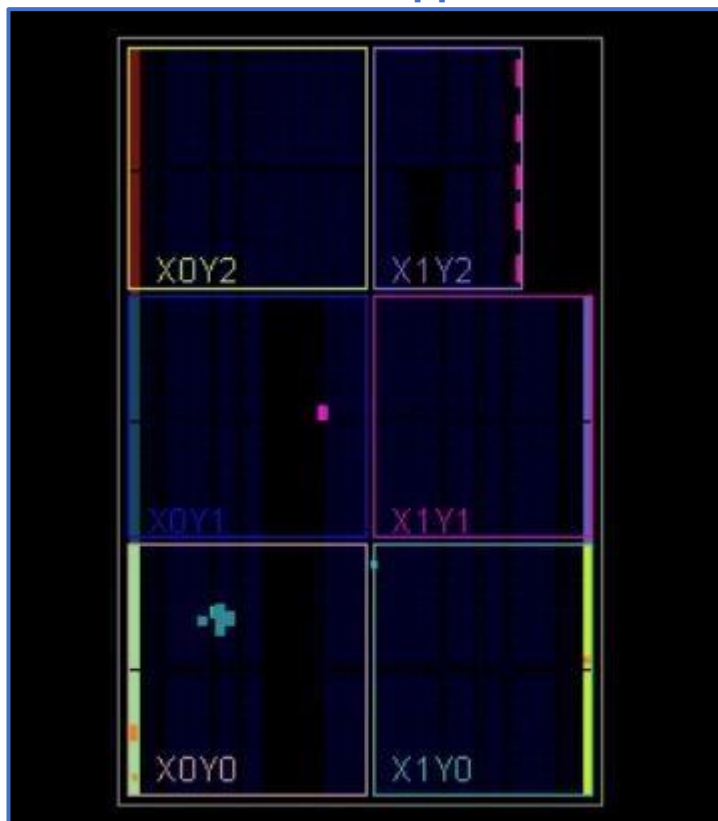
6.1 Utilization Report

Hierarchy										
Name	Slice LUTs (20800)	Slice Registers (41600)	F7 Muxes (16300)	Slice (8150)	LUT as Logic (20800)	LUT Flip Flop Pairs (20800)	Block RAM Tile (50)	Bonded IOB (106)	BUFGCTRL (32)	
▼ SPI_Wrapper	42	50	1	21	42	13	0.5	5	1	
RAM (RAM_MODULE)	3	17	1	4	3	0	0.5	0	0	
SPI (SPI_MODULE)	39	33	0	19	39	12	0	0	0	

6.2 Timing Report

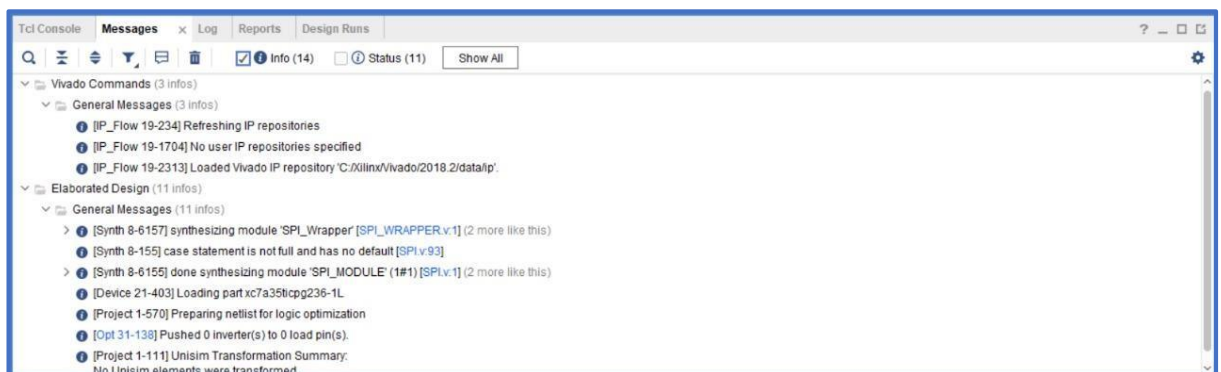
Design Timing Summary			
Setup	Hold	Pulse Width	
Worst Negative Slack (WNS): 5.925 ns	Worst Hold Slack (WHS): 0.069 ns	Worst Pulse Width Slack (WPWS): 4.500 ns	
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns	Total Pulse Width Negative Slack (TPWS): 0.000 ns	
Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	
Total Number of Endpoints: 109	Total Number of Endpoints: 109	Total Number of Endpoints: 53	
All user specified timing constraints are met.			

6.3 FPGA device snippet



7.0 CRITICAL WARNINGS

7.1 Elaboration



7.2 Synthesis



7.3 Implementation

