SPI SLAVE WITH SINGLE PORT RAM

Under Supervision of:

ENG. Kareem Waseem

Submitted by:

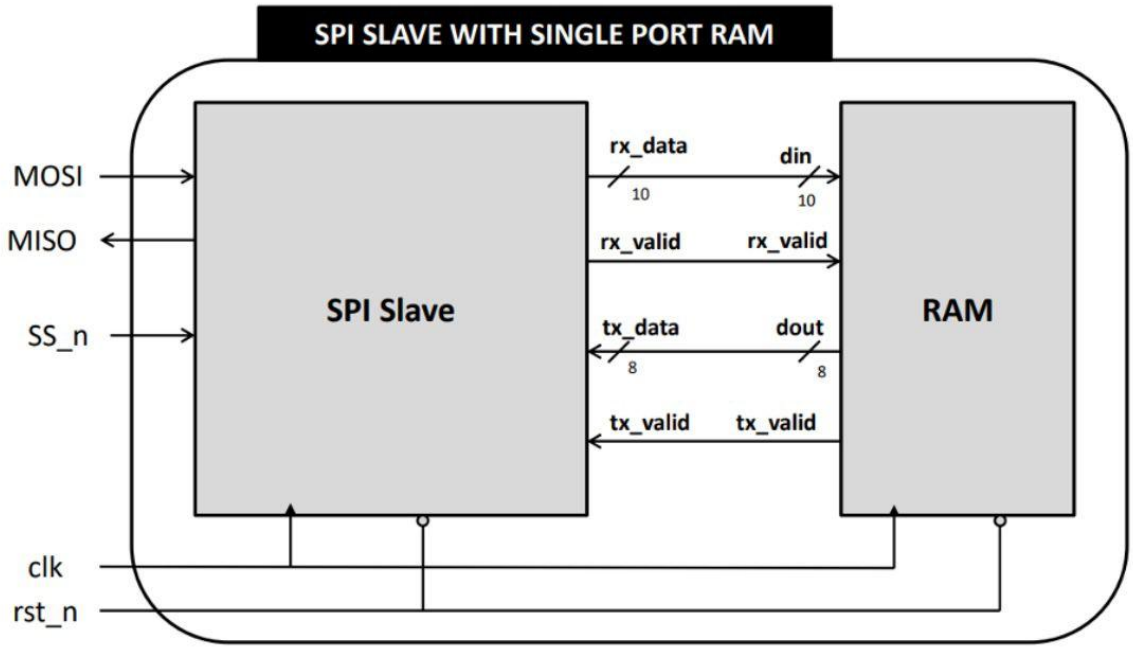
Ahmed Belal

Omar Waleed

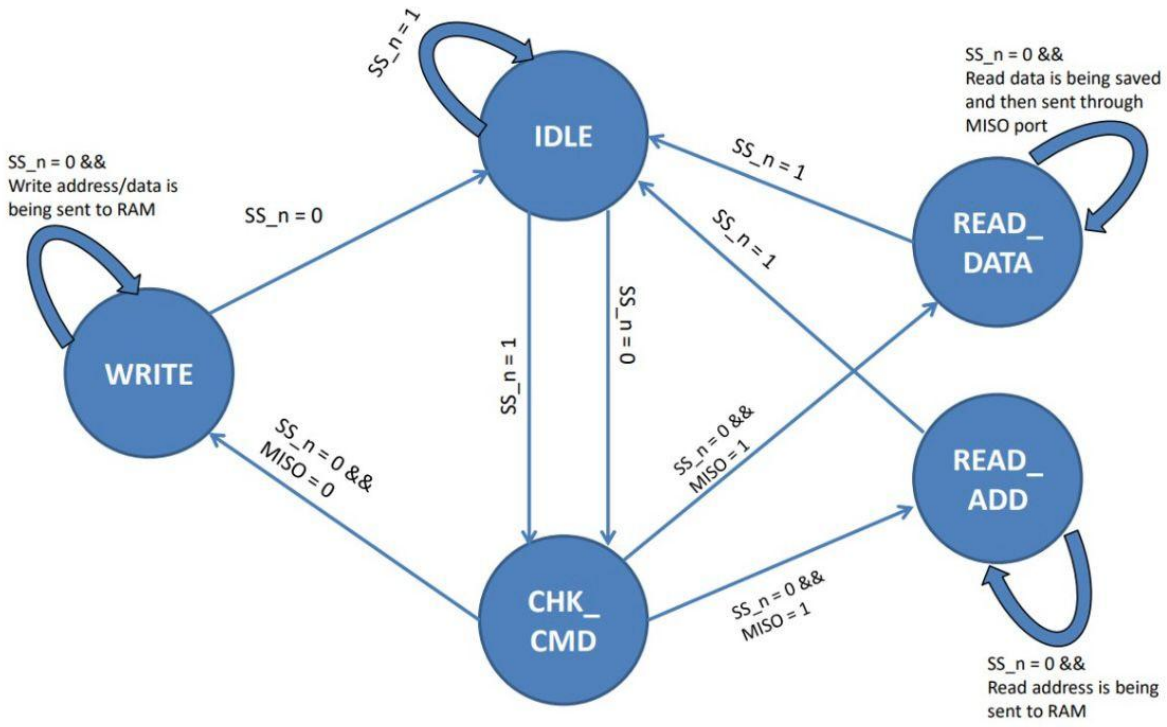
Overview

This project implements a Serial Peripheral Interface (SPI) Slave on an FPGA that communicates with an SPI Master to store and retrieve data using an on-chip single-port RAM. The SPI Slave receives commands and data over SS_n, SCLK, and MOSI, writes data into RAM on write commands, and returns the requested bytes over MISO on read commands. A finite-state machine (FSM) coordinates command decoding, address handling, and data transfers to ensure reliable, synchronous operation with an active-low reset.

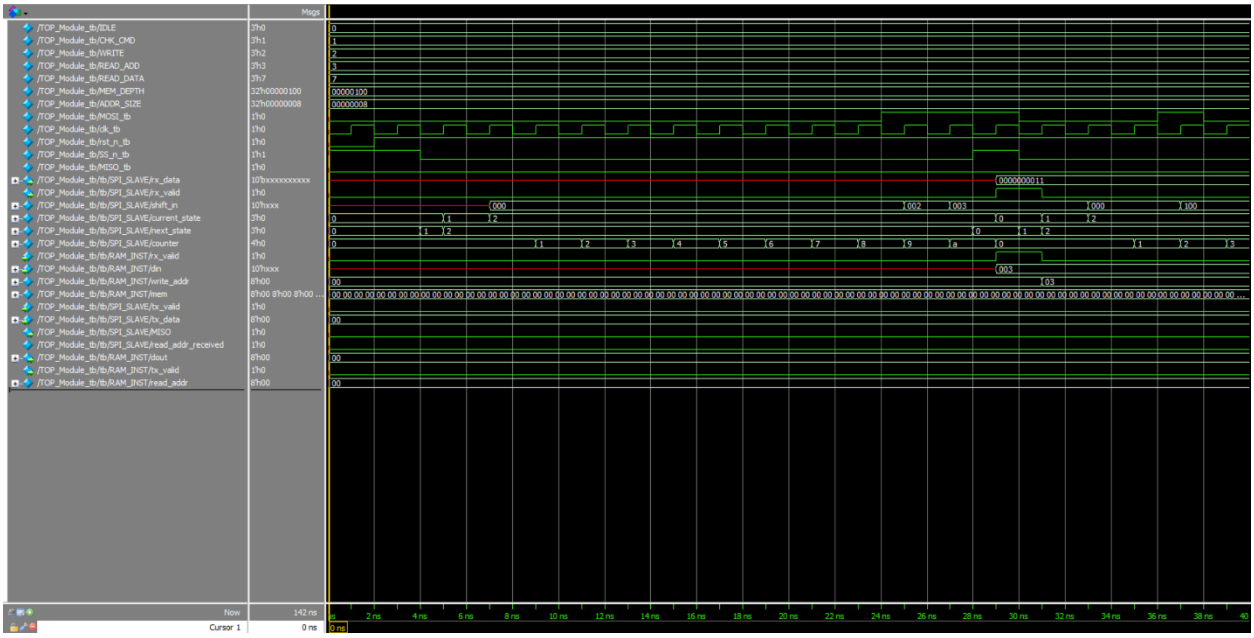
Beyond demonstrating a practical SPI memory-mapped peripheral, the design compares Gray, One-Hot, and Sequential FSM encodings to show how state encoding choices affect timing and resource usage on FPGA, highlighting performance/area trade-offs in real digital systems.

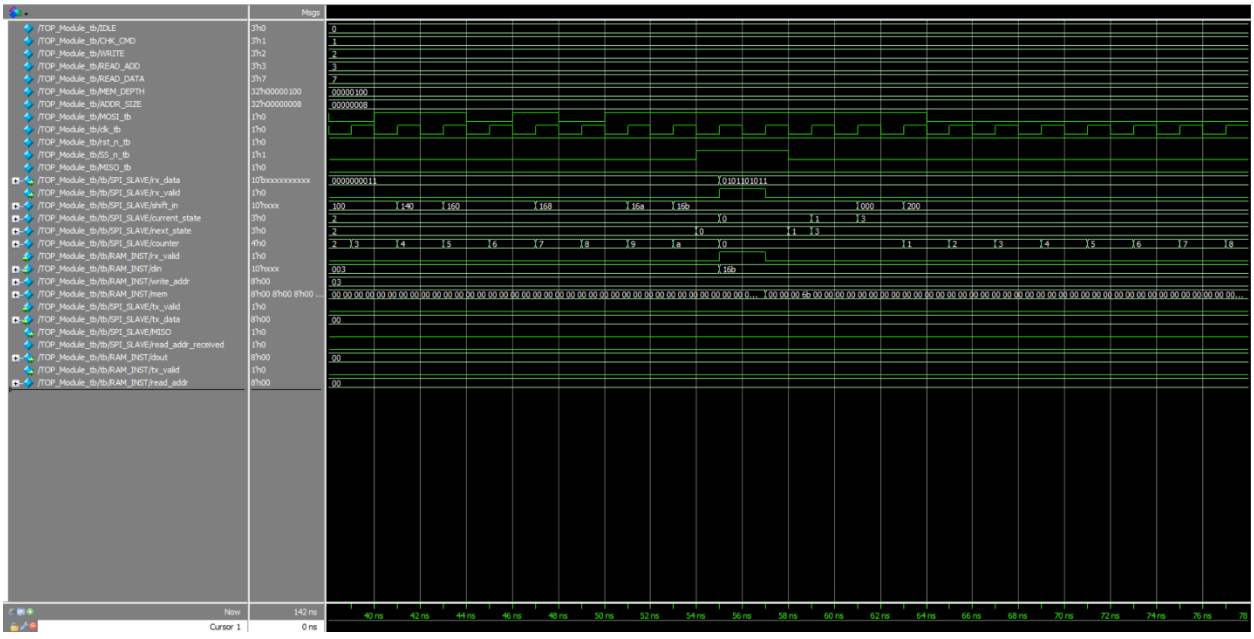


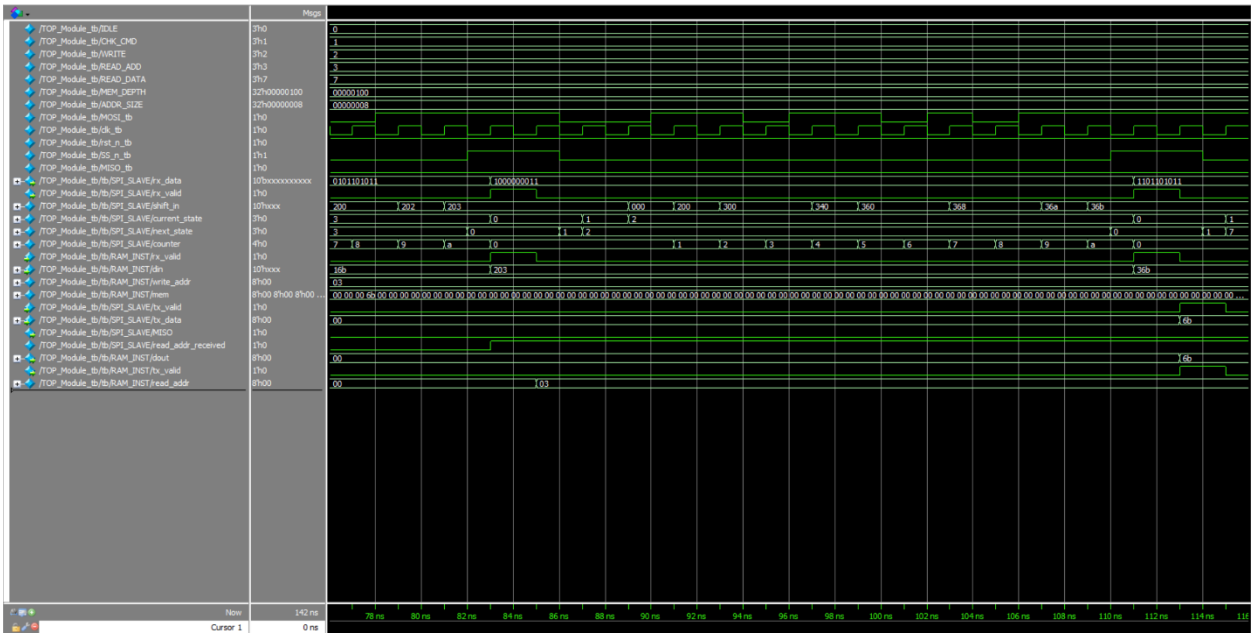
A Finite State Machine (FSM) is used to manage the communication process:

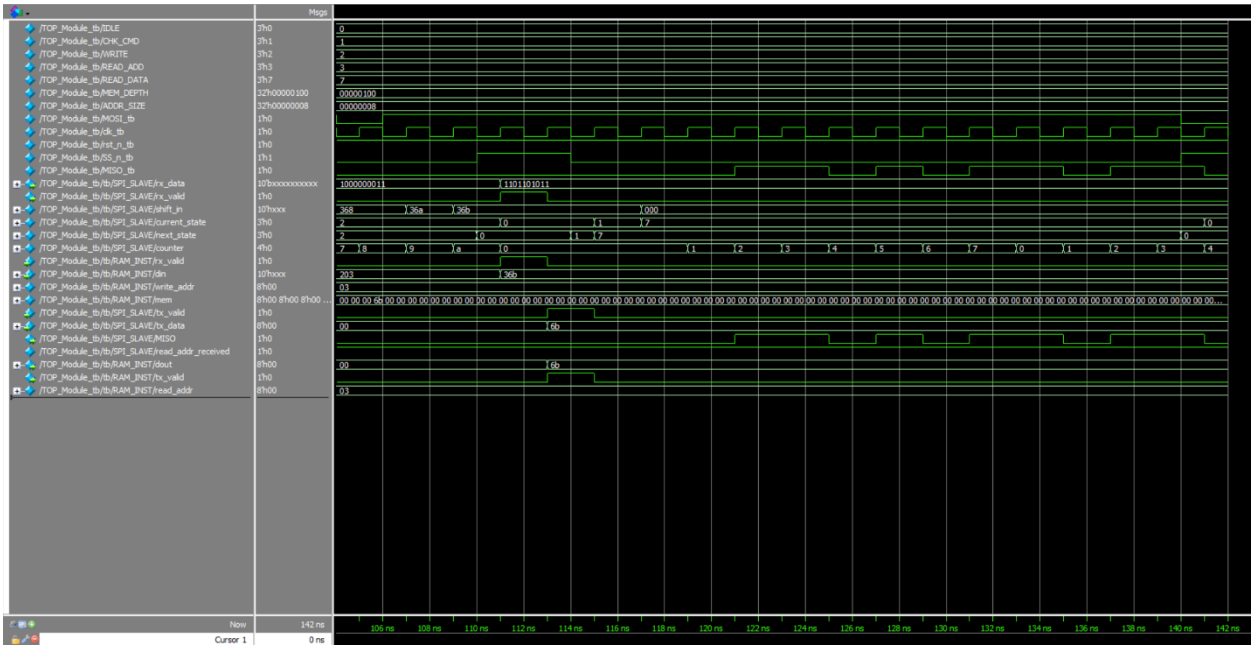


- **Questa Sim Snippets:**

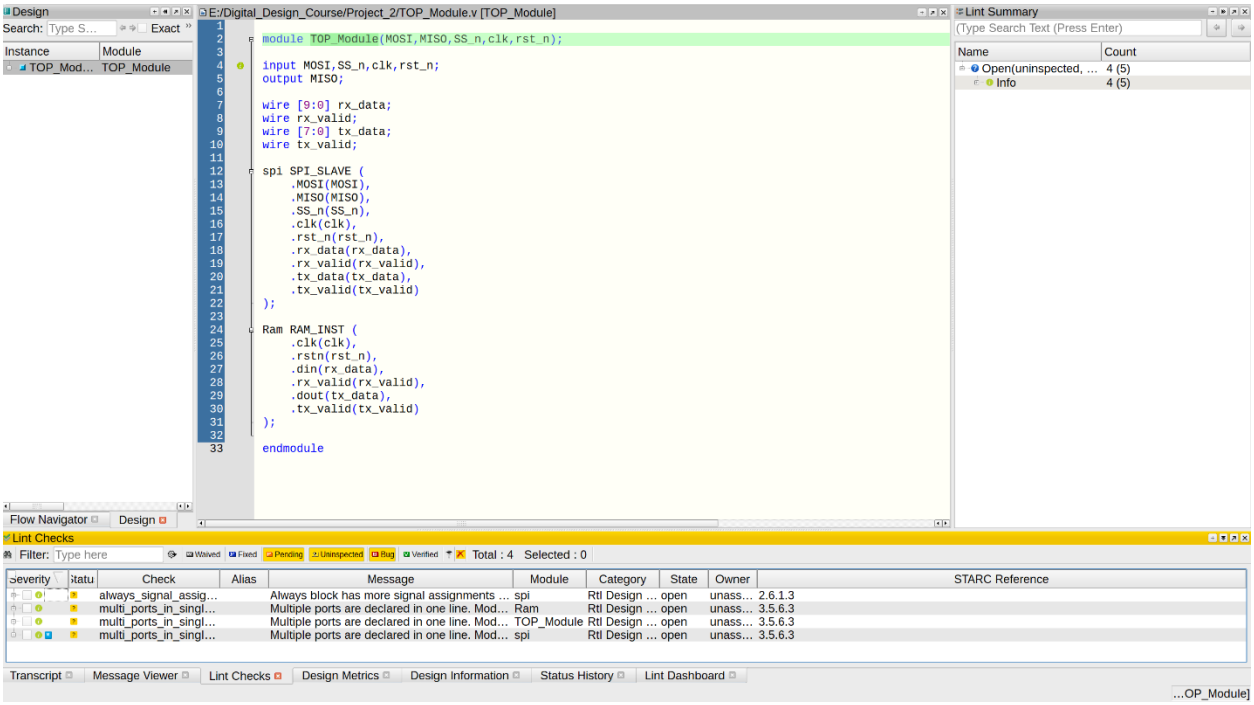




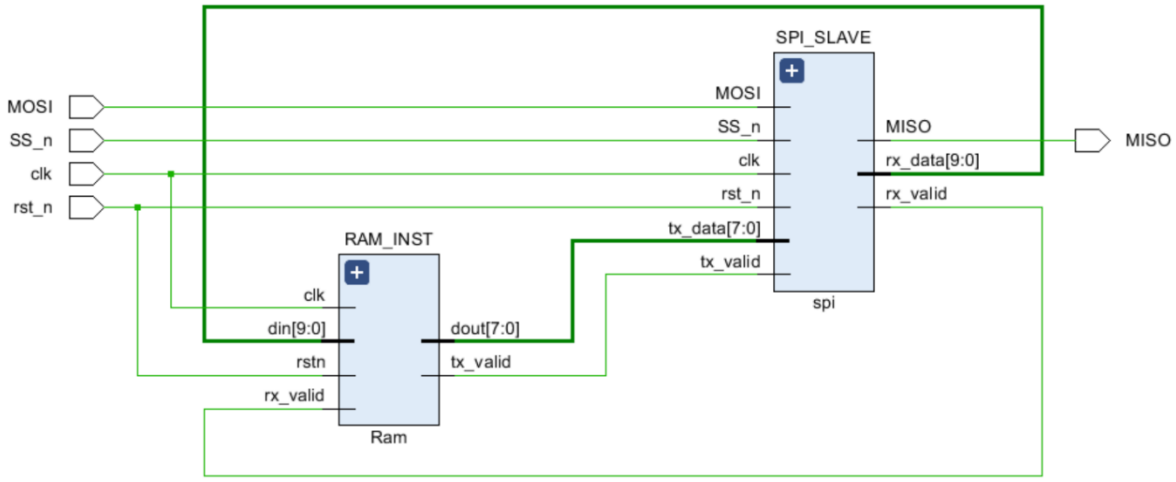

/TOP_Module_tb/IDLE
/TOP_Module_tb/CHK_CMD
/TOP_Module_tb/WRITE
/TOP_Module_tb/READ_ADD
/TOP_Module_tb/READ_DATA
/TOP_Module_tb/MEM_DEPTH
/TOP_Module_tb/ADDR_SIZE
/TOP_Module_tb/MOSI_tb
/TOP_Module_tb/clk_tb
/TOP_Module_tb/rst_n_tb
/TOP_Module_tb/SS_n_tb
/TOP_Module_tb/MISO_tb
/TOP_Module_tb/tb/SPI_SLAVE/rx_data
/TOP_Module_tb/tb/SPI_SLAVE/rx_valid
/TOP_Module_tb/tb/SPI_SLAVE/shift_in
/TOP_Module_tb/tb/SPI_SLAVE/current_state
/TOP_Module_tb/tb/SPI_SLAVE/next_state
/TOP_Module_tb/tb/SPI_SLAVE/counter
/TOP_Module_tb/tb/RAM_INST/rx_valid
/TOP_Module_tb/tb/RAM_INST/din
/TOP_Module_tb/tb/RAM_INST/write_addr
/TOP_Module_tb/tb/RAM_INST/mem
/TOP_Module_tb/tb/SPI_SLAVE/tx_valid
/TOP_Module_tb/tb/SPI_SLAVE/tx_data
/TOP_Module_tb/tb/SPI_SLAVE/MISO
/TOP_Module_tb/tb/SPI_SLAVE/read_addr_received
/TOP_Module_tb/tb/RAM_INST/dout
/TOP_Module_tb/tb/RAM_INST/tx_valid
/TOP_Module_tb/tb/RAM_INST/read_addr
Msgs
Now
Cursor 1


/TOP_Module_tb/IDLE
/TOP_Module_tb/CHK_CMD
/TOP_Module_tb/WRITE
/TOP_Module_tb/READ_ADD
/TOP_Module_tb/READ_DATA
/TOP_Module_tb/MEM_DEPTH
/TOP_Module_tb/ADDR_SIZE
/TOP_Module_tb/MOSI_tb
/TOP_Module_tb/clk_tb
/TOP_Module_tb/rst_n_tb
/TOP_Module_tb/SS_n_tb
/TOP_Module_tb/MISO_tb
/TOP_Module_tb/tb/SPI_SLAVE/rx_data
/TOP_Module_tb/tb/SPI_SLAVE/rx_valid
/TOP_Module_tb/tb/SPI_SLAVE/shift_in
/TOP_Module_tb/tb/SPI_SLAVE/current_state
/TOP_Module_tb/tb/SPI_SLAVE/next_state
/TOP_Module_tb/tb/SPI_SLAVE/counter
/TOP_Module_tb/tb/RAM_INST/rx_valid
/TOP_Module_tb/tb/RAM_INST/din
/TOP_Module_tb/tb/RAM_INST/write_addr
/TOP_Module_tb/tb/RAM_INST/mem
/TOP_Module_tb/tb/SPI_SLAVE/tx_valid
/TOP_Module_tb/tb/SPI_SLAVE/tx_data
/TOP_Module_tb/tb/SPI_SLAVE/MISO
/TOP_Module_tb/tb/SPI_SLAVE/read_addr_received
/TOP_Module_tb/tb/RAM_INST/dout
/TOP_Module_tb/tb/RAM_INST/tx_valid
/TOP_Module_tb/tb/RAM_INST/read_addr
Msgs
Now
Cursor 1

- **QuestLint Snippets:**



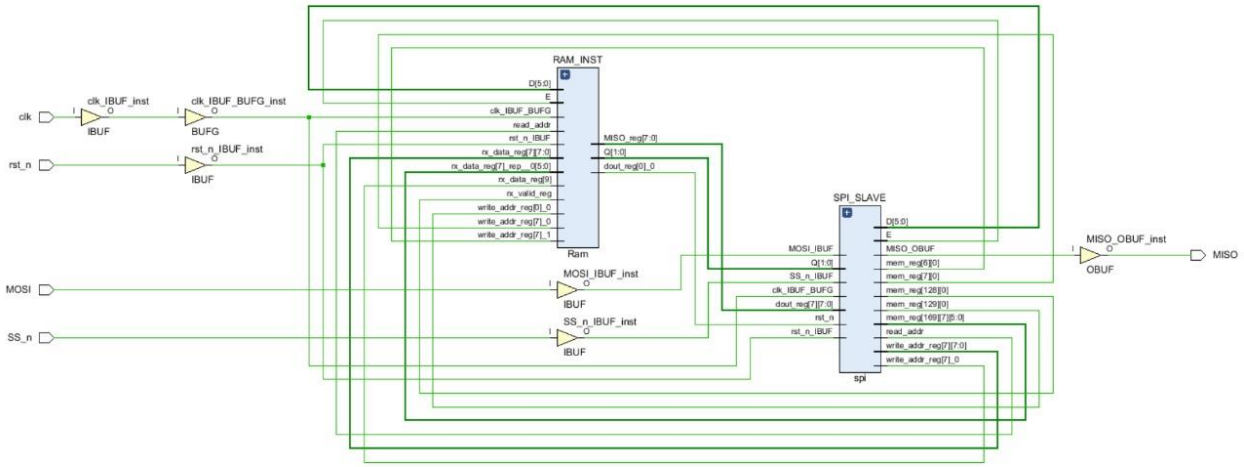
- **Elaboration schematic:**



- **Elaboration Massage:**

➢ Gray Code:

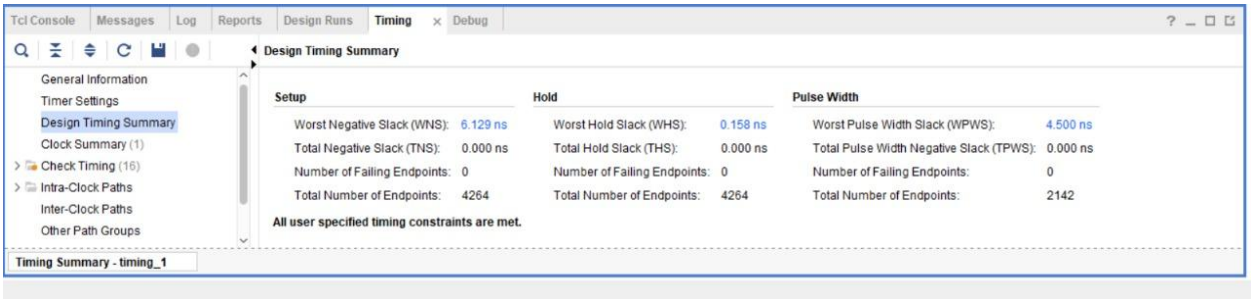
- Synthesis Schematic of Gray Code:



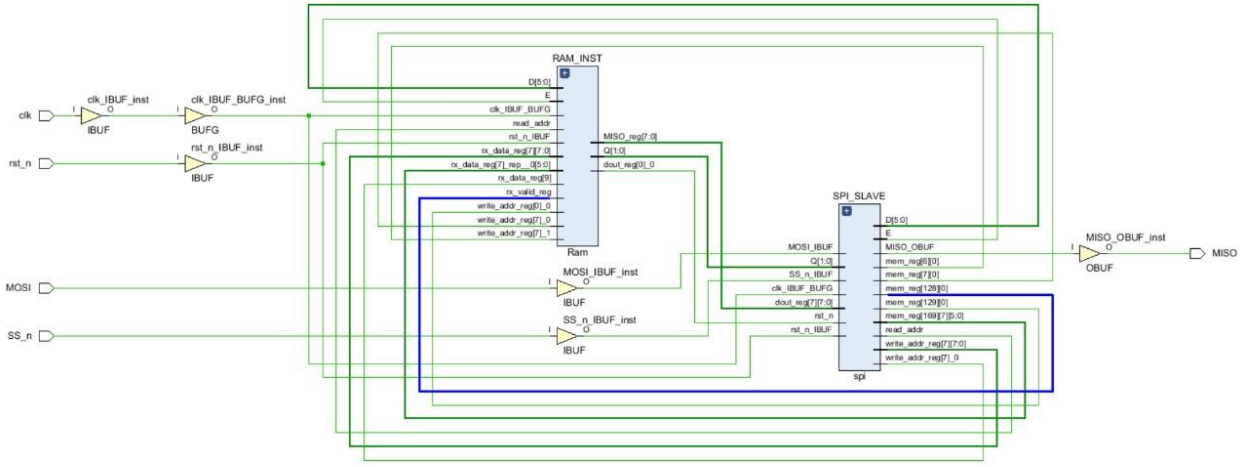
- Synthesis report showing the encoding used:

```
109 | ---------------------------------------------------------------------------------------------------
110 |                   State |                     New Encoding |               Previous Encoding
111 | ---------------------------------------------------------------------------------------------------
112 |                    IDLE |                              000 |                              000
113 |                 CHK_CMD |                              001 |                              001
114 |               READ_DATA |                              011 |                              111
115 |                READ_ADD |                              010 |                              011
116 |                   WRITE |                              111 |                              010
117 | ---------------------------------------------------------------------------------------------------
118 | INFO: [Synth 8-3354] encoded FSM with state register 'current_state_reg' using encoding 'gray' in module 'spi'
119 | WARNING: [Synth 8-327] inferring latch for variable 'FSM_gray_next_state_reg' [C:/Users/omarw/Desktop/spi_project/spi.v
120 | ---------------------------------------------------------------------------
```

- Synthesis Timing report snippet:

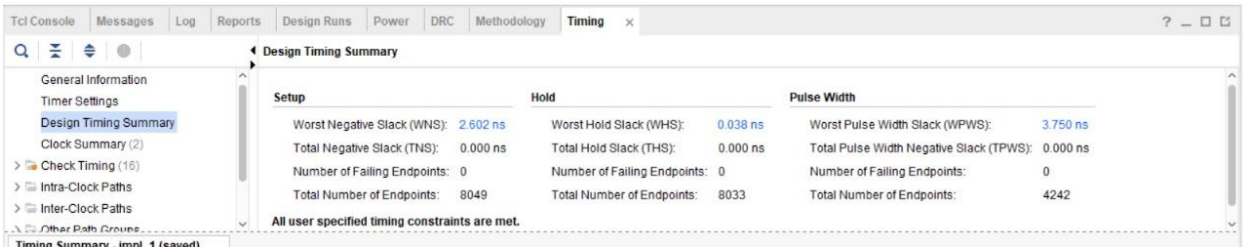


- Snippet of the critical path highlighted in the schematic:

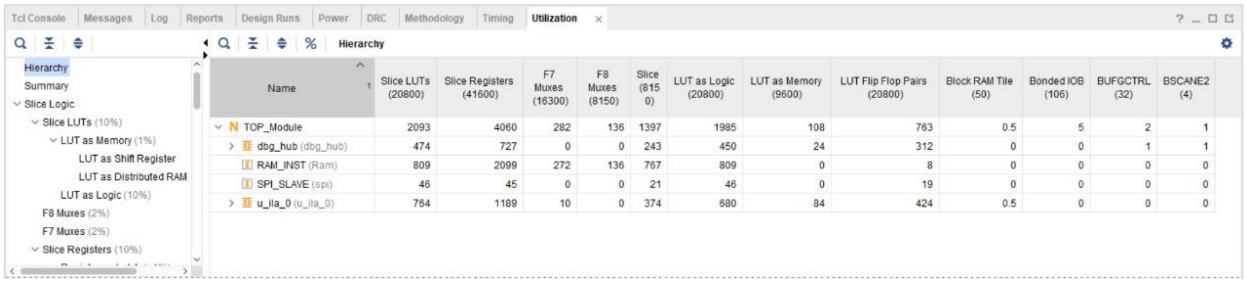


- Massage after Synthesis:

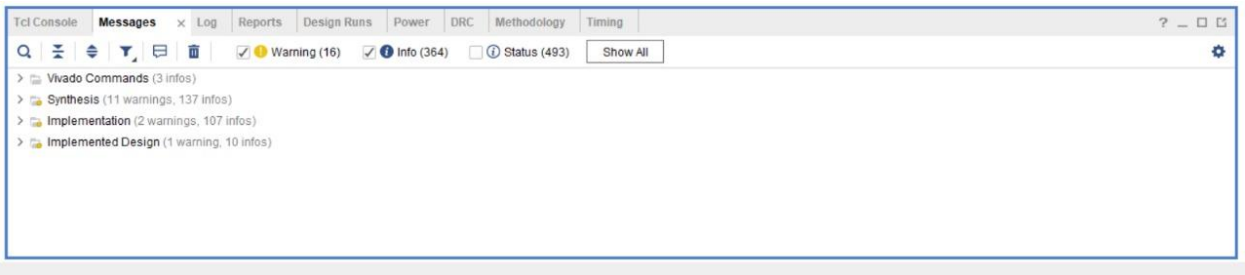
- Implementation Timing report:



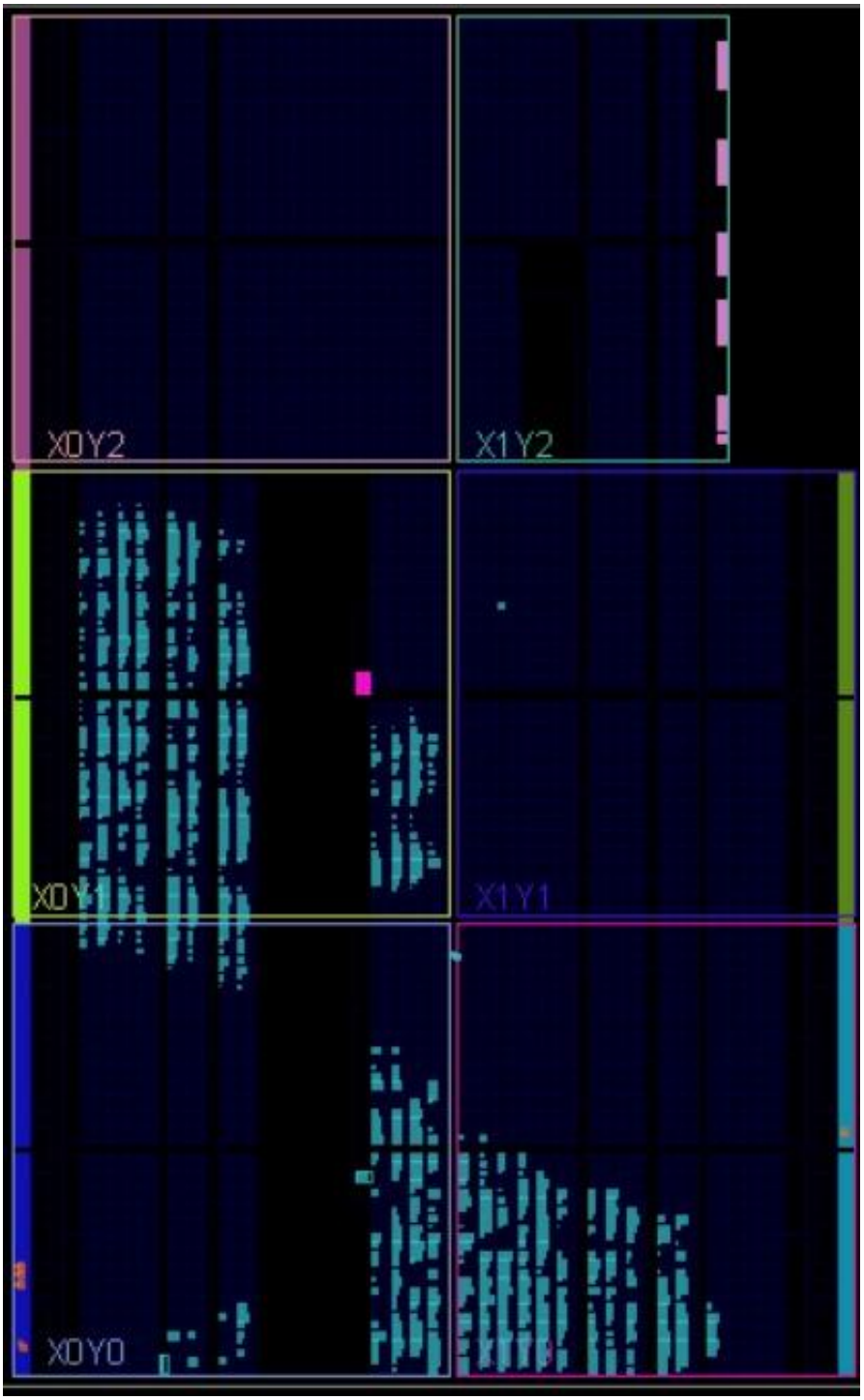
- Implementation Utilization report:



- Massage after Implementation:

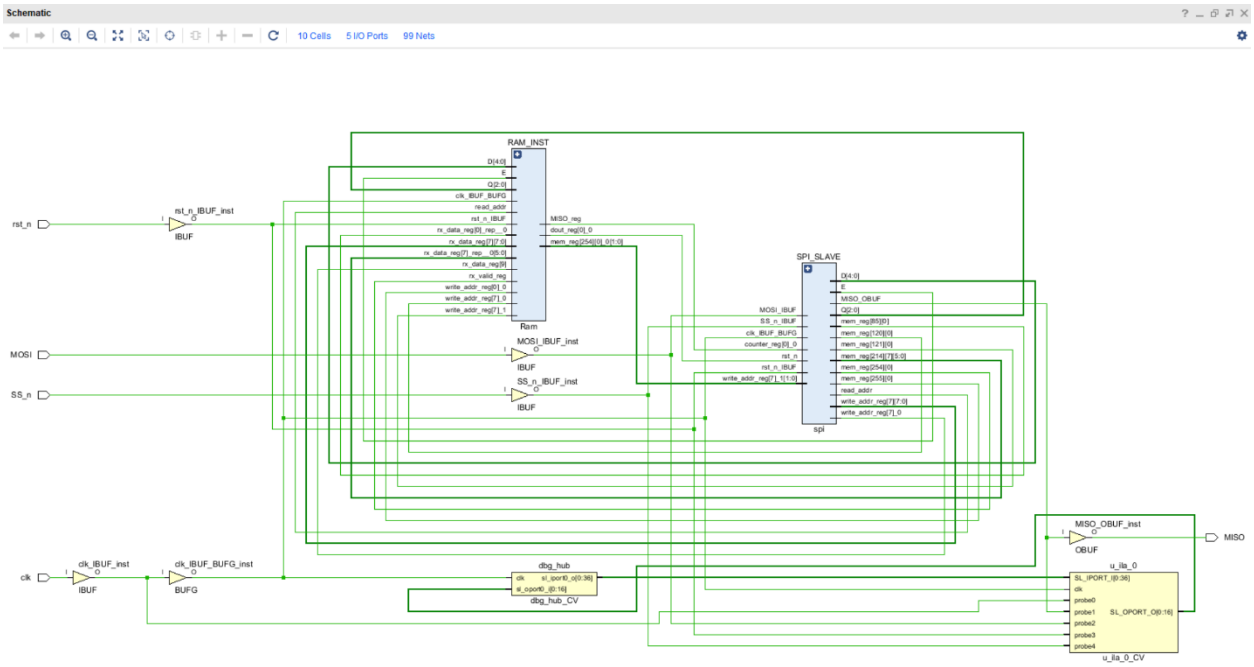


- FPGA device snippet:



➢ One_hot Code:

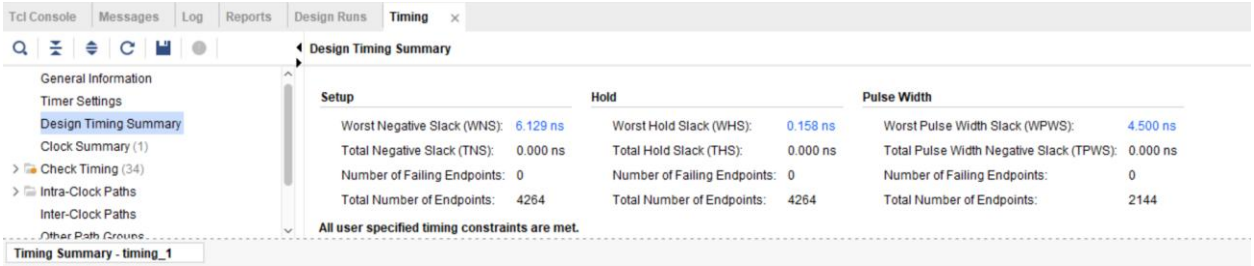
- Synthesis Schematic of one_hot Code:



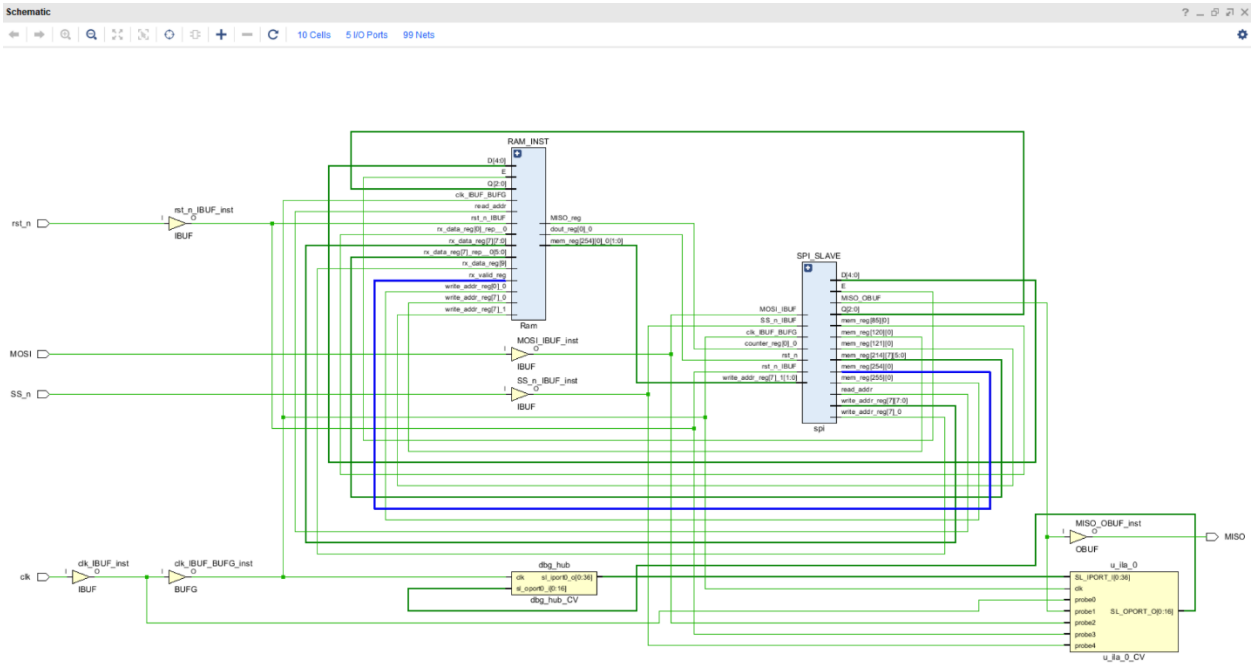
- Synthesis report showing the encoding used:

```
105 | ----------------------------------------------------------------------------------------------------
106 |                    State |                     New Encoding |               Previous Encoding
107 | ----------------------------------------------------------------------------------------------------
108 |                     IDLE |                            00001 |                                  000
109 |                  CHK_CMD |                            00010 |                                  001
110 |                READ_DATA |                            00100 |                                  111
111 |                 READ_ADD |                            01000 |                                  011
112 |                    WRITE |                            10000 |                                  010
113 | ----------------------------------------------------------------------------------------------------
114 | INFO: [Synth 8-3354] encoded FSM with state register 'current_state_reg' using encoding 'one-hot' in module 'spi'
115 | WARNING: [Synth 8-327] inferring latch for variable 'FSM_onehot_next_state_reg' [E:/Digital_Design_Course/Project_2/spi.v:32]
116 | ---------------------------------------------------------------------------
117 | Finished RTL Optimization Phase 2 : Time (s): cpu = 00:00:13 ; elapsed = 00:00:16 . Memory (MB): peak = 786.770 ; gain = 529.246
```

- Synthesis Timing report snippet:



- Snippet of the critical path highlighted in the schematic:

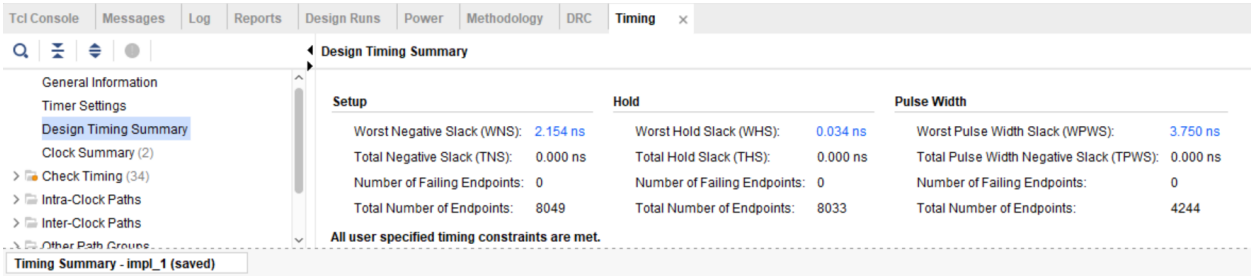


- Massage after Synthesis:

Synthesis (9 warnings, 135 infos)

- [Common 17-349] Got license for feature 'Synthesis' and/or device 'xc7a35ti'
- [Synth 8-6157] synthesizing module 'TOP_Module' [TOP_Module.v:2] (2 more like this)
- [Synth 8-5534] Detected attribute (* fsm_encoding = "one_hot" *) [spi.v:18]
- [Synth 8-5788] Register shift_in_reg in module spi is has both Set and reset with same priority. This may cause simulation mismatches. Consider rewriting code [spi.v:88] (2 more like this)
- [Synth 8-226] default block is never used [Ram.v:24]
- [Synth 8-4767] Trying to implement RAM 'mem_reg' in registers. Block RAM or DRAM implementation is not possible; see log for reasons.
- [Synth 8-6155] done synthesizing module 'spi' (1#1) [spi.v:1] (2 more like this)
- [Synth 8-3331] design spi has unconnected port tx_valid
- [Device 21-403] Loading part xc7a35ticpg236-1L
- [Project 1-236] Implementation specific constraints were found while reading constraint file [E:/Digital_Design_Course/Project_2/spi.xdc]. These constraints will be ignored for synthesis but will be used in implementation. Impacted constraints are listed in the file [.Xil/TOP_Module_propImpl.xdc].
  Resolution: To avoid this warning, move constraints listed in [Undefined] to another XDC file and exclude this new file from synthesis with the used_in_synthesis property (File Properties dialog in GUI) and re-run elaboration/synthesis.
- [Synth 8-802] inferred FSM for state register 'current_state_reg' in module 'spi'
- [Synth 8-5544] ROM "read_addr_received" won't be mapped to Block RAM because address size (4) smaller than threshold (5) (8 more like this)
- [Synth 8-3354] encoded FSM with state register 'current_state_reg' using encoding 'one-hot' in module 'spi'
- [Synth 8-327] inferring latch for variable 'FSM_onehot_next_state_reg' [spi.v:32]
- [Synth 8-5546] ROM "p_0_out" won't be mapped to RAM because it is too sparse (99 more like this)
- [Common 17-14] Message 'Synth 8-5546' appears 100 times and further instances of the messages will be disabled. Use the Tcl command set_msg_config to change the current settings.
- [Synth 8-3332] Sequential element (RAM_INST/tx_valid_reg) is unused and will be removed from module TOP_Module.
- [Project 1-571] Translating synthesized netlist
- [Netlist 29-17] Analyzing 417 Unisim elements for replacement
- [Netlist 29-28] Unisim Transformation completed in 0 CPU seconds
- [Netlist 29-101] Netlist 'TOP_Module' is not ideal for floorplanning, since the cellview 'Ram' contains a large number of primitives. Please consider enabling hierarchy in synthesis if you want to do floorplanning.
- [Project 1-570] Preparing netlist for logic optimization (1 more like this)
- [Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s).
- [Project 1-111] Unisim Transformation Summary:
  No Unisim elements were transformed. (1 more like this)
- [Common 17-83] Releasing license: Synthesis
- [Constraints 18-5210] No constraint will be written out.
- [Common 17-1381] The checkpoint 'E:/Digital_Design_Course/Project_2/VIVADO/Project_2/Project_2.runs/synth_1/TOP_Module.dcp' has been generated.
- [runtcl-4] Executing : report_utilization -file TOP_Module_utilization_synth.rpt -pb TOP_Module_utilization_synth.pb
- [Common 17-206] Exiting Vivado at Tue Aug 5 23:07:56 2025...

- Implementation Timing report:

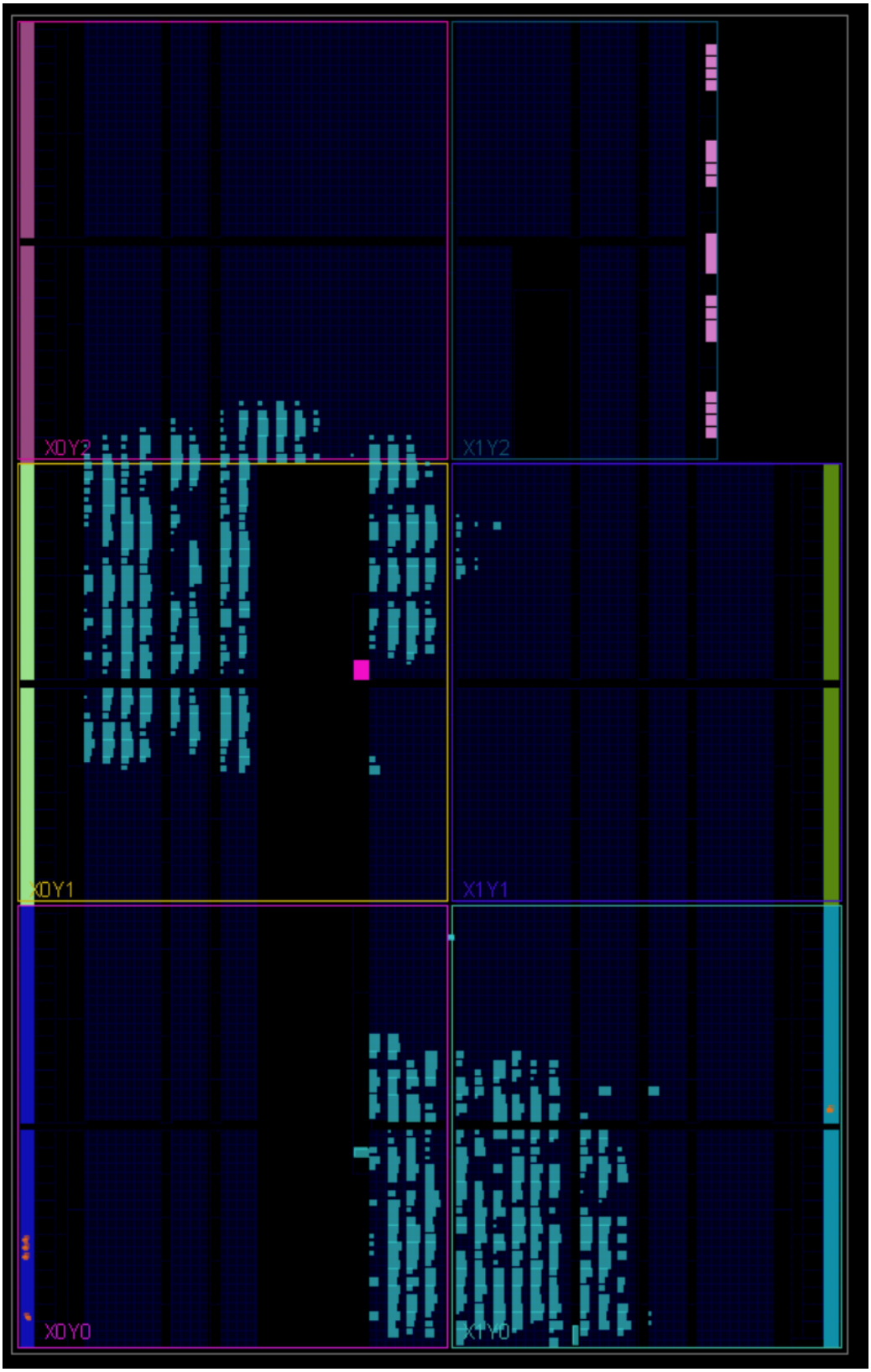


- Implementation Utilization report:

Name	Slice LUTs (20800)	Slice Registers (41600)	F7 Muxes (16300)	F8 Muxes (8150)	Slice (8150)	LUT as Logic (20800)	LUT as Memory (9600)	LUT Flip Flop Pairs (20800)	Block RAM Tile (50)	Bonded IOB (106)	BUFGCTRL (32)	BSCANE2 (4)
TOP_Module	2090	4064	282	136	1366	1982	108	752	0.5	5	2	1
dbg_hub (dbg_hub)	475	727	0	0	246	451	24	304	0	0	1	1
RAM_INST (Ram)	812	2099	272	136	739	812	0	8	0	0	0	0
SPI_SLAVE (spi)	40	49	0	0	20	40	0	21	0	0	0	0
u_ila_0 (u_ila_0)	763	1189	10	0	372	679	84	417	0.5	0	0	0

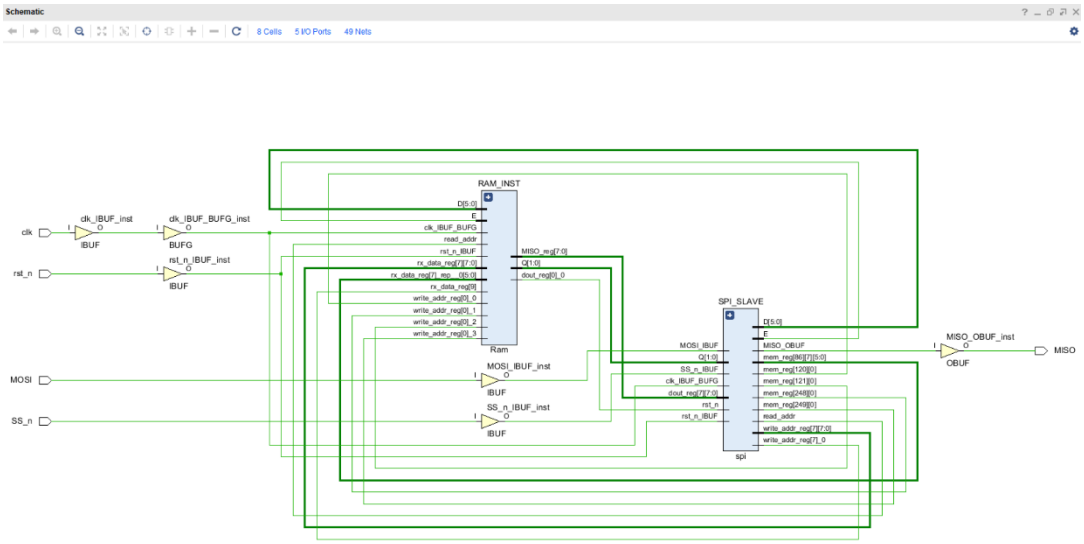
- Massage after Implementation:

- FPGA device snippet:



➢ Sequential Code:

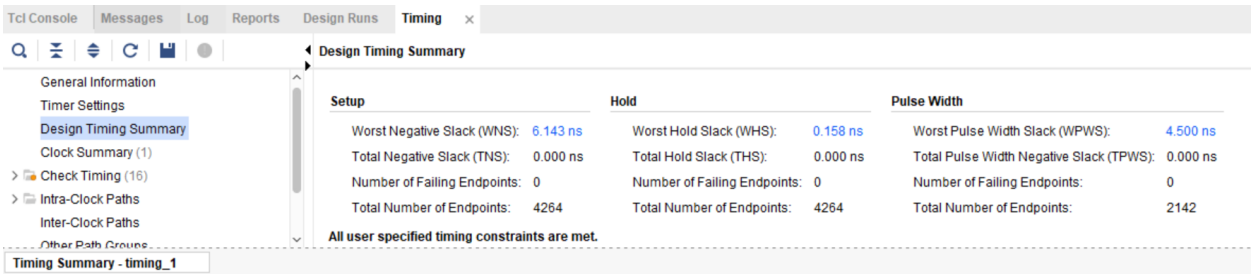
- Synthesis Schematic of Seq Code:



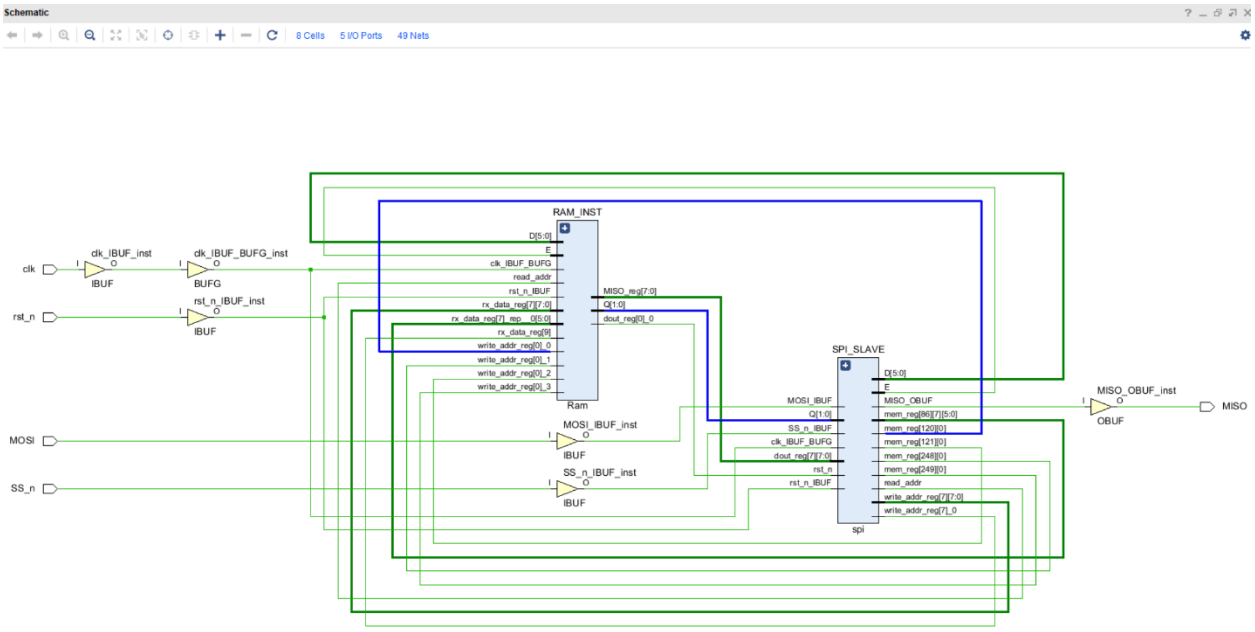
- Synthesis report showing the encoding used:

```
105 | ---------------------------------------------------------------------------------------------------
106 |                   State |                     New Encoding |               Previous Encoding
107 | ---------------------------------------------------------------------------------------------------
108 |                    IDLE |                              000 |                              000
109 |                 CHK_CMD |                              001 |                              001
110 |               READ_DATA |                              010 |                              111
111 |               READ_ADD |                              011 |                              011
112 |                  WRITE |                              100 |                              010
113 | ---------------------------------------------------------------------------------------------------
114 | INFO: [Synth 8-3354] encoded FSM with state register 'current_state_reg' using encoding 'sequential' in module 'spi'
115 | WARNING: [Synth 8-327] inferring latch for variable 'FSM_sequential_next_state_reg' [E:/Digital_Design_Course/Project_2/spi.v:32]
116 | ---------------------------------------------------------------------------
117 | Finished RTL Optimization Phase 2 : Time (s): cpu = 00:00:14 ; elapsed = 00:00:16 . Memory (MB): peak = 785.449 ; gain = 527.711
```

- Synthesis Timing report snippet:

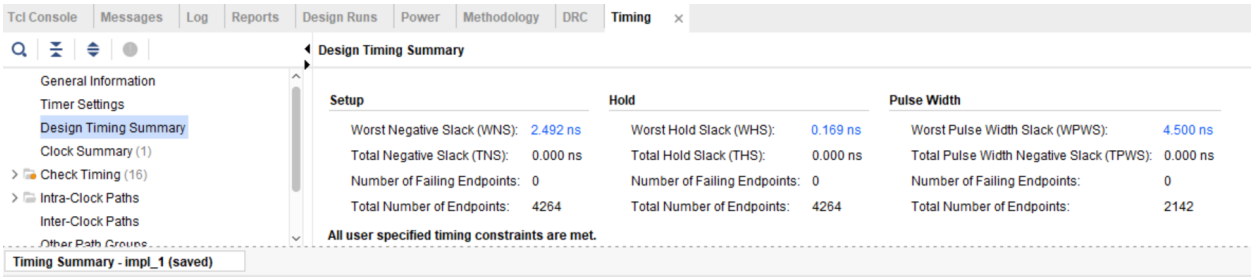


- Snippet of the critical path highlighted in the schematic:

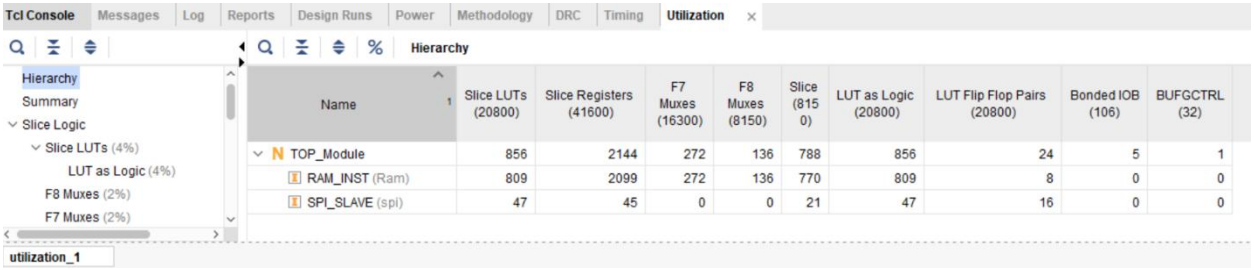


- Massage after Synthesis:

- Implementation Timing report:

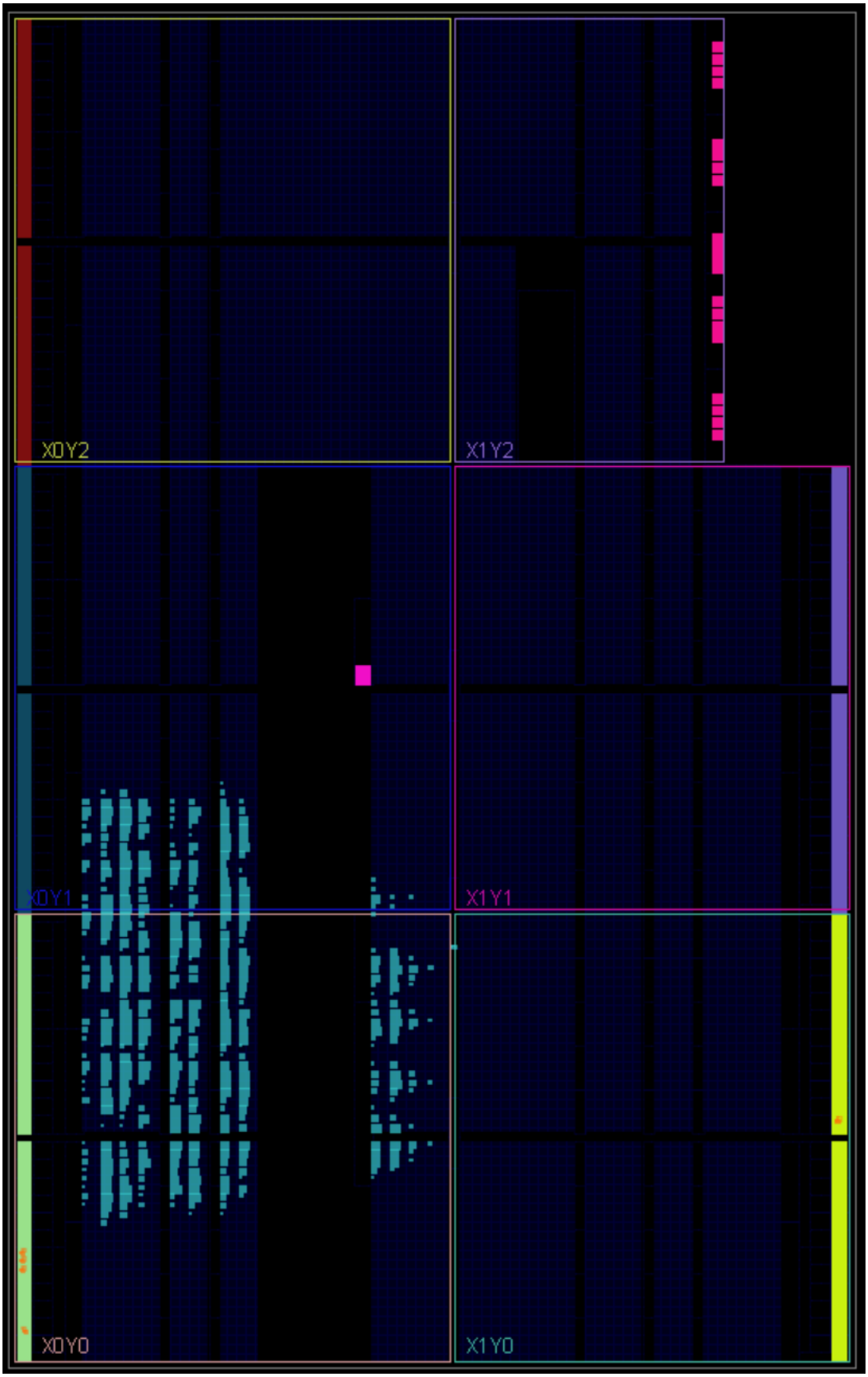


- Implementation Utilization report:



- Massage after Implementation:

- FPGA device snippet:



- Note that: The best timing is the **Gray Code**

```
161
162   create_debug_core u_ila_0 ila
163   set_property ALL_PROBE_SAME_MU true [get_debug_cores u_ila_0]
164   set_property ALL_PROBE_SAME_MU_CNT 1 [get_debug_cores u_ila_0]
165   set_property C_ADV_TRIGGER false [get_debug_cores u_ila_0]
166   set_property C_DATA_DEPTH 1024 [get_debug_cores u_ila_0]
167   set_property C_EN_STRG_QUAL false [get_debug_cores u_ila_0]
168   set_property C_INPUT_PIPE_STAGES 0 [get_debug_cores u_ila_0]
169   set_property C_TRIGIN_EN false [get_debug_cores u_ila_0]
170   set_property C_TRIGOUT_EN false [get_debug_cores u_ila_0]
171   set_property port_width 1 [get_debug_ports u_ila_0/clk]
172   connect_debug_port u_ila_0/clk [get_nets [list clk_IBUF_BUFG]]
173   set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe0]
174   set_property port_width 1 [get_debug_ports u_ila_0/probe0]
175   connect_debug_port u_ila_0/probe0 [get_nets [list clk_IBUF]]
176   create_debug_port u_ila_0 probe
177   set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe1]
178   set_property port_width 1 [get_debug_ports u_ila_0/probe1]
179   connect_debug_port u_ila_0/probe1 [get_nets [list MISO_OBUF]]
180   create_debug_port u_ila_0 probe
181   set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe2]
182   set_property port_width 1 [get_debug_ports u_ila_0/probe2]
183   connect_debug_port u_ila_0/probe2 [get_nets [list MOSI_IBUF]]
184   create_debug_port u_ila_0 probe
185   set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe3]
186   set_property port_width 1 [get_debug_ports u_ila_0/probe3]
187   connect_debug_port u_ila_0/probe3 [get_nets [list rst_n_IBUF]]
188   create_debug_port u_ila_0 probe
189   set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe4]
190   set_property port_width 1 [get_debug_ports u_ila_0/probe4]
191   connect_debug_port u_ila_0/probe4 [get_nets [list SS_n_IBUF]]
192   set_property C_CLK_INPUT_FREQ_HZ 300000000 [get_debug_cores dbg_hub]
193   set_property C_ENABLE_CLK_DIVIDER false [get_debug_cores dbg_hub]
194   set_property C_USER_SCAN_CHAIN 1 [get_debug_cores dbg_hub]
195   connect_debug_port dbg_hub/clk [get_nets clk_IBUF_BUFG]
196
```