Small Microprocessor (SM) Design
Using AXI4 Full of ARM Cortex-A9

By

HITESH SOJITRA
ABD ALRAHMAN AL NOUNOU

Advisor

PROF. DANIEL LIAMOCCA OBREGON

Final Project

ECE-5736: Reconfigurable Computing

Electrical and Computer Engineering Department

School of Engineering and Computer Science

At

Oakland University

Jun 2022

Outlines:

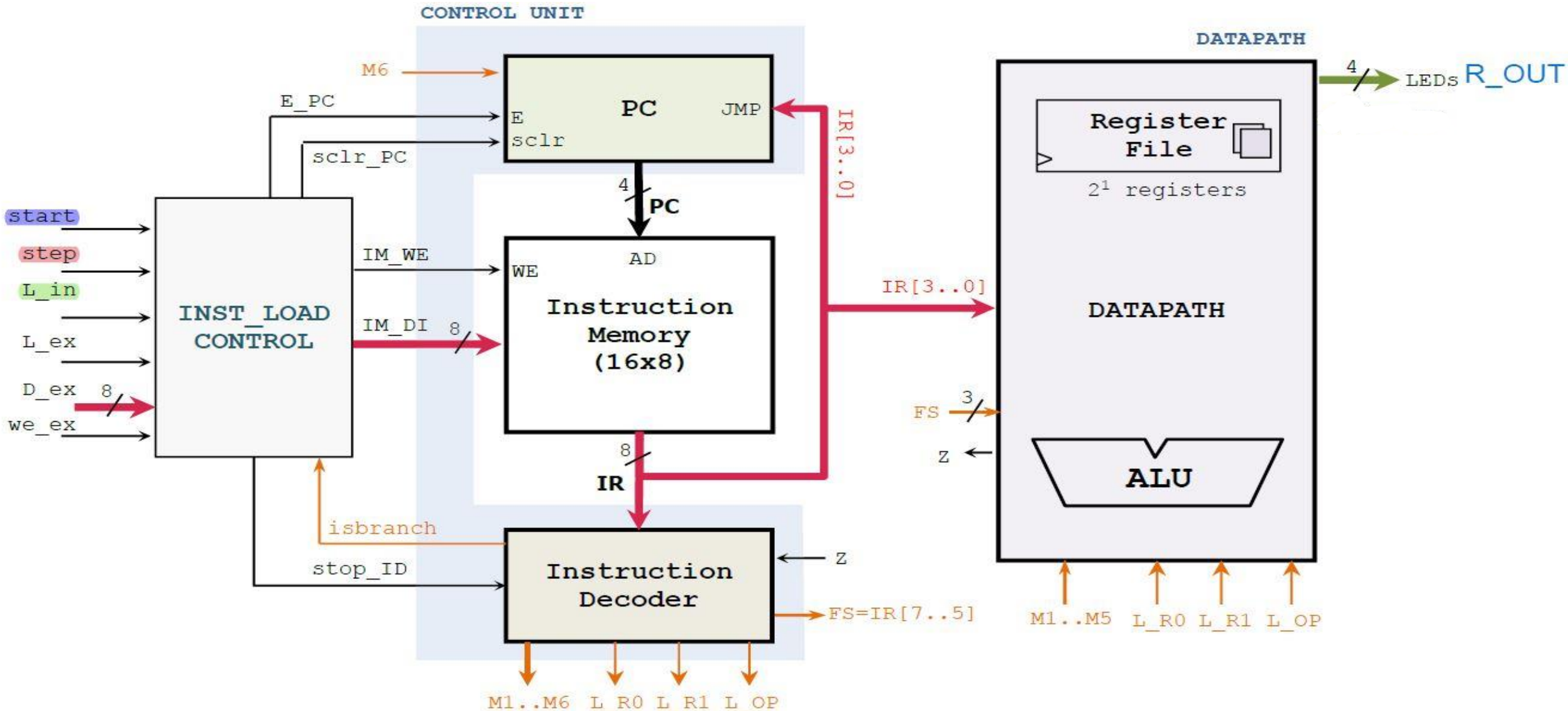
- Overview
- Methodology & Block Diagram
- Result & discussions
- Conclusions & Future work
- Demo

Overview

- We implement a simple microprocessor where instructions are processed in one clock cycle.
- We design an AXI4_Full interface for a small microprocessor (SM) peripheral.
- We integrate this custom peripheral into an embedded system in Vivado.
- We also create a software application in the SDK tool that can communicate with the peripheral.
- The output is mapped to external outputs on the development board LEDs.

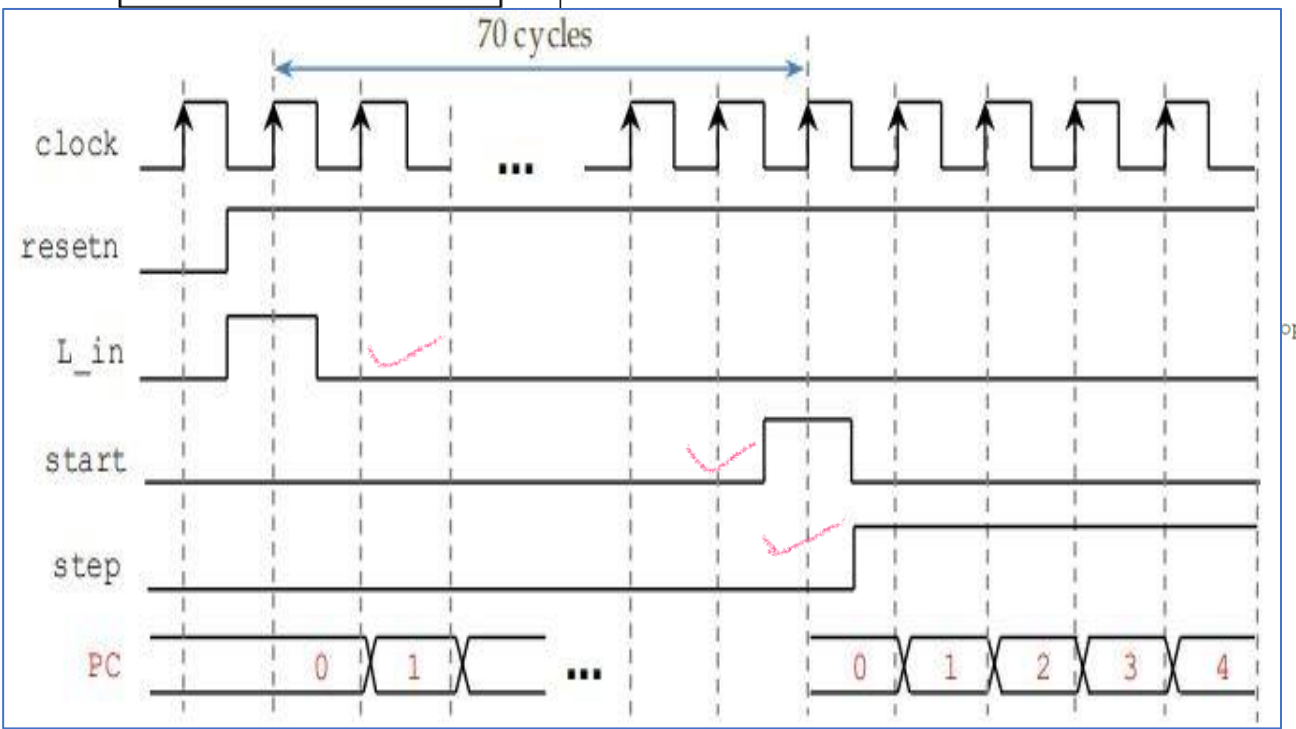
Methodology & Block Diagram

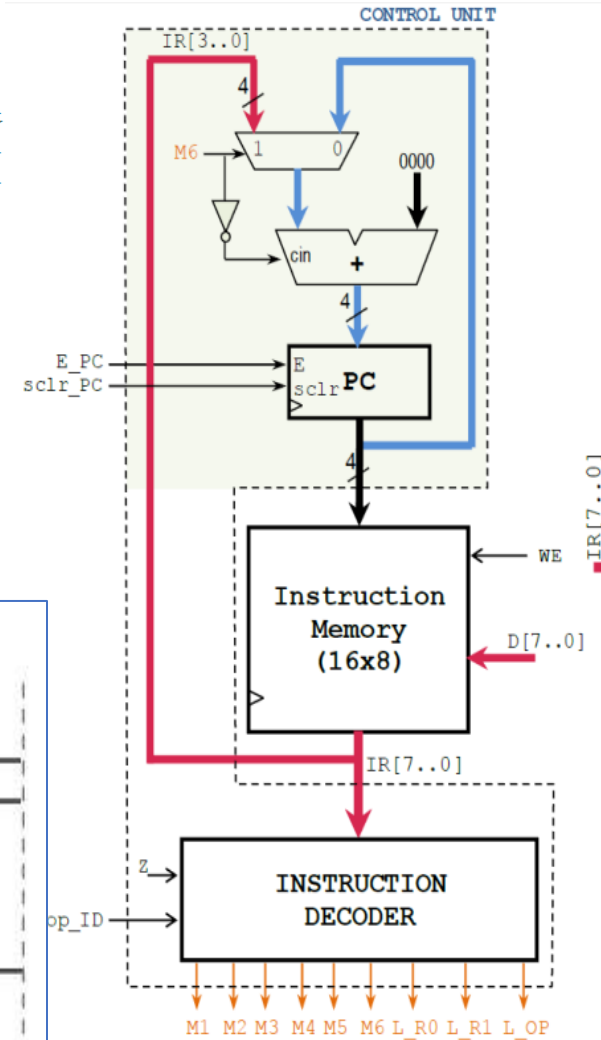
VBC_Architecture

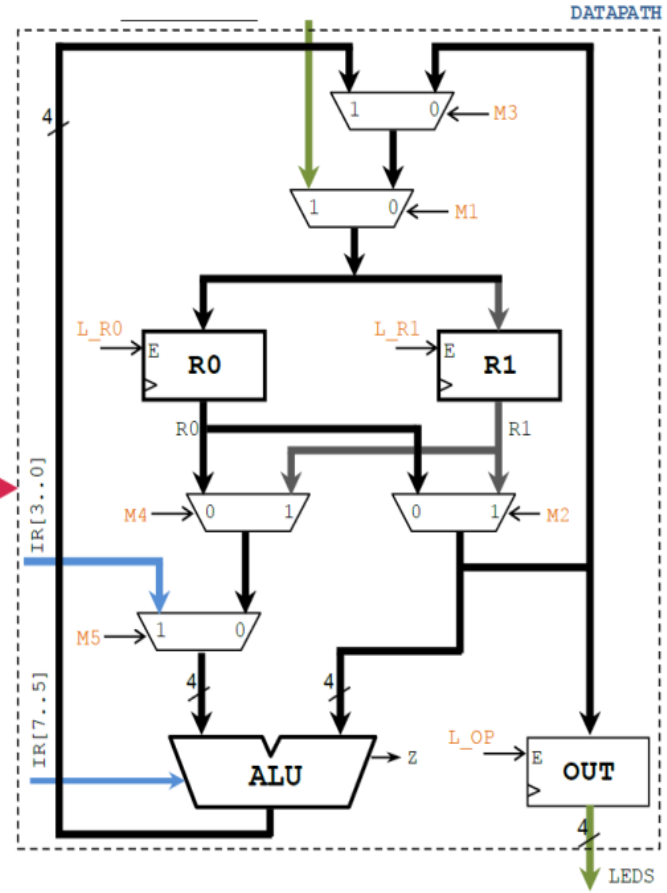


address	INSTRUCTION MEMORY
0000	00100010
0001	00110100
0010	11000000
0011	01100001
0100	01110001
0101	11110010
0110	00100001
0111	11100000
1000	00000000
1001	00000000
1010	00000000
1011	00000000
1100	00000000
1101	00000000
1110	00000000
1111	00000000

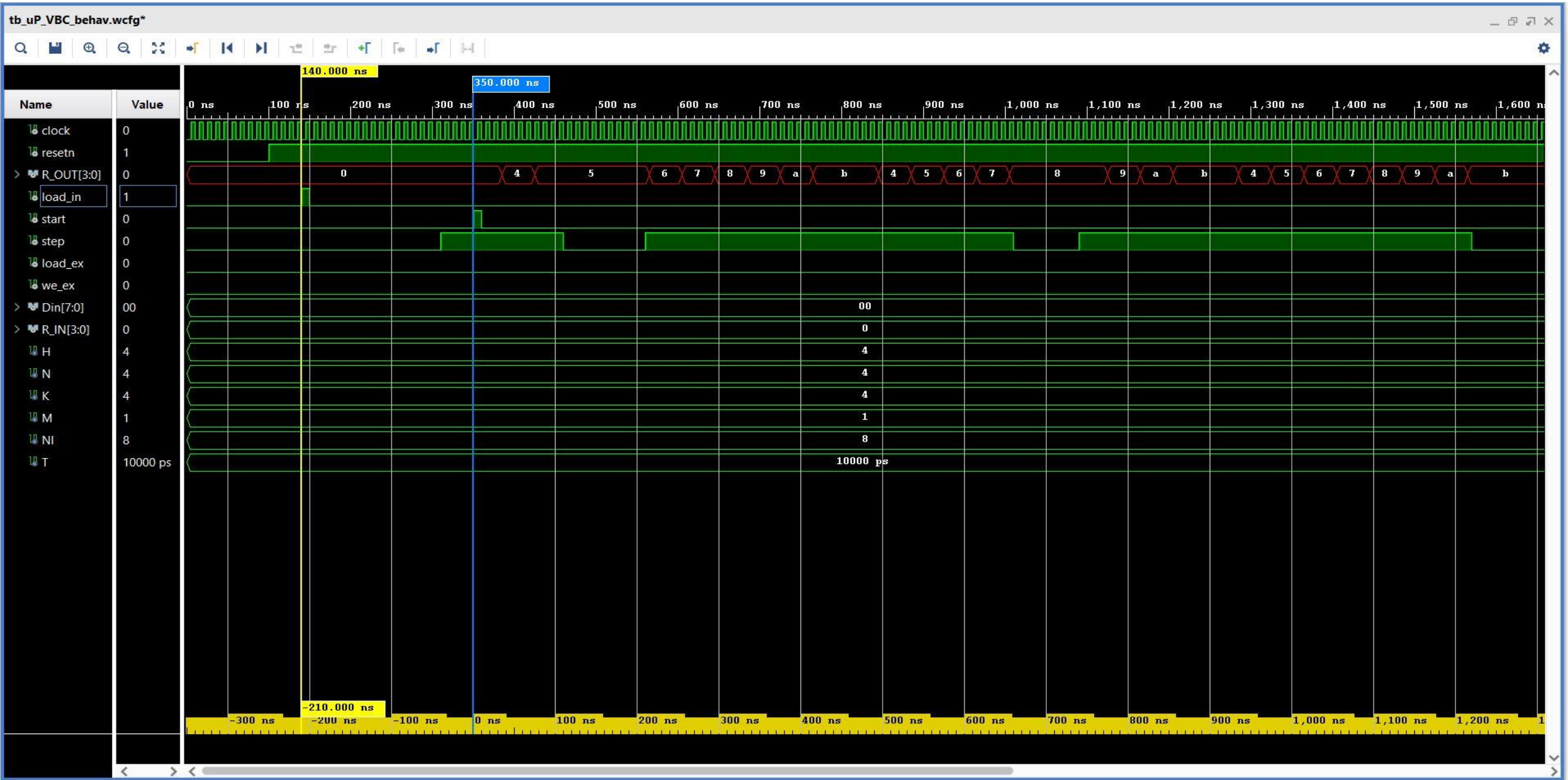
```
* 2 to 13 ≡ 4 to 15


start:  loadi R0,2          R0 ← 2
        loadi R1,4          R1 ← 4
loop:   out R0 → OUT: shows the count
        addi R0,1           R0 ← R0+1
        addi R1,1           R1 ← R1+1
        jnz R1, loop
        loadi R0,1          R0 ← 1
        jnz R0, start
```



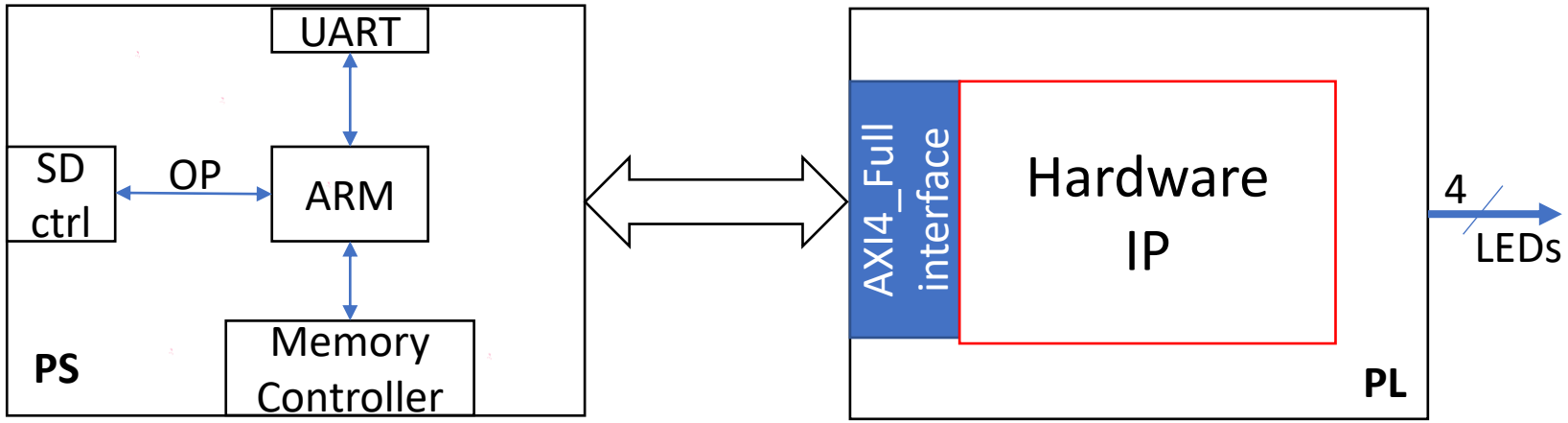


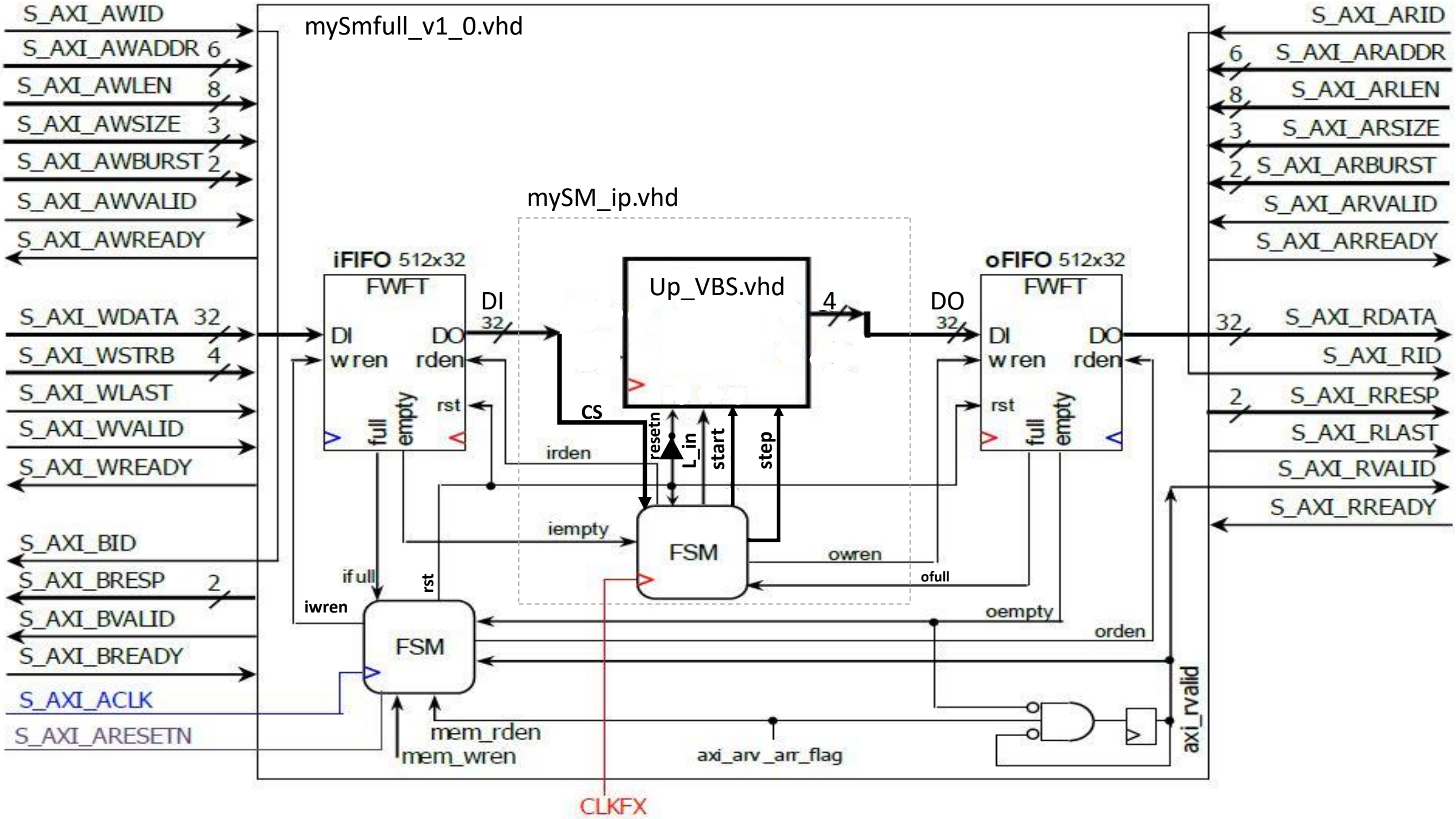


VBC_MultiRound

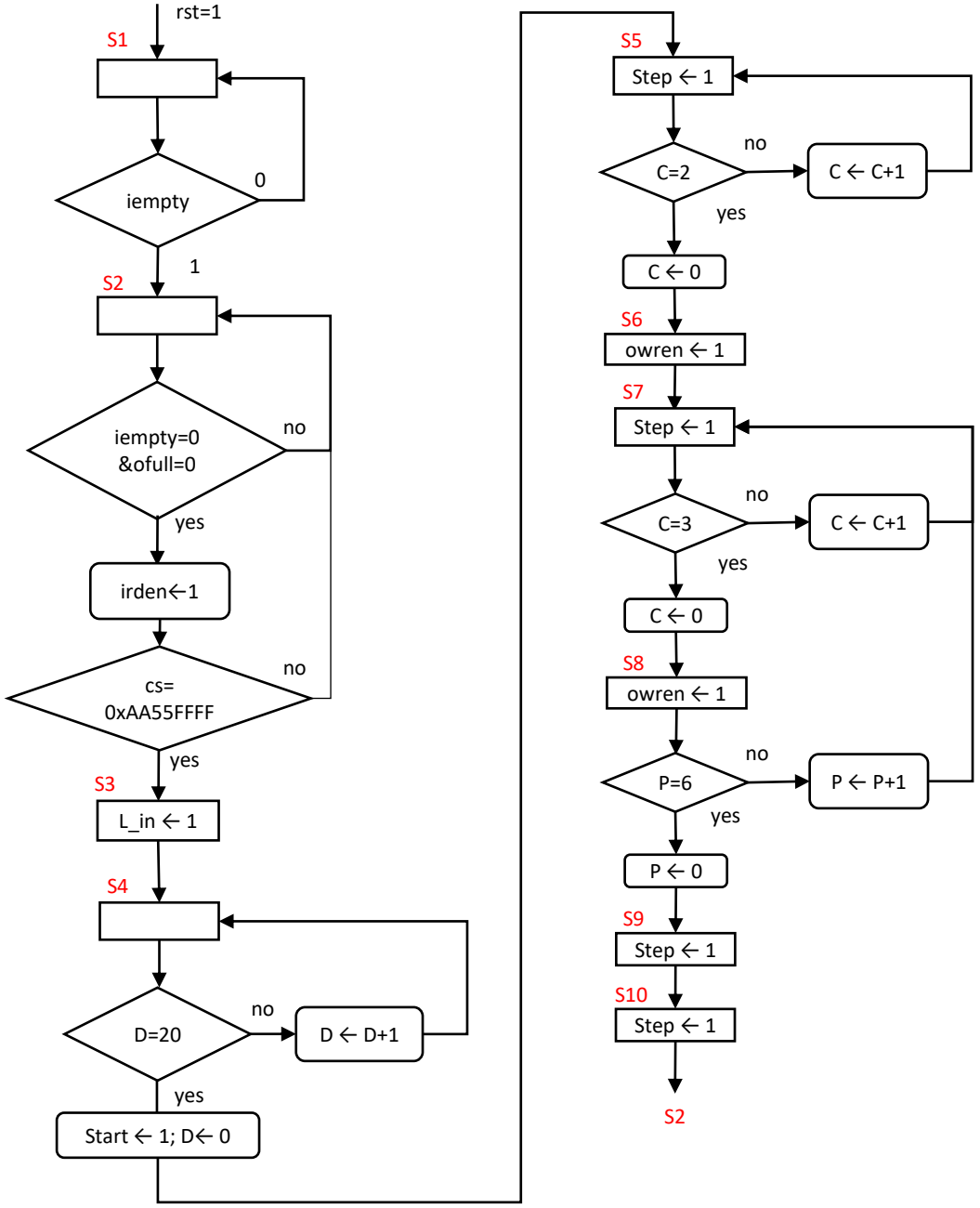


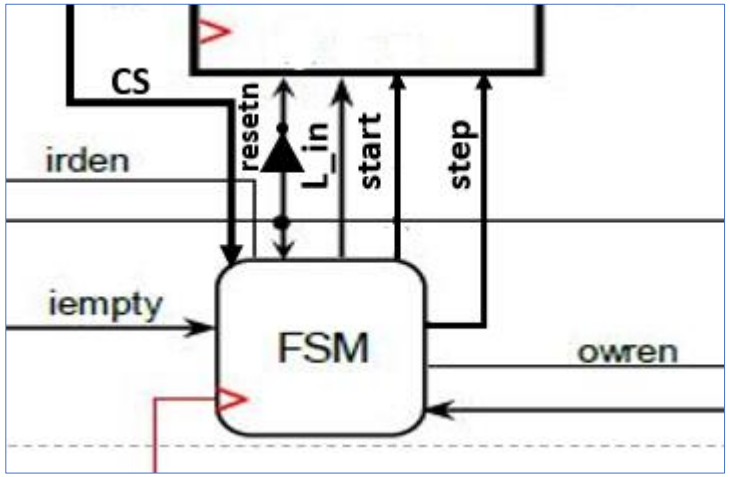
AXI4_Full_SM_LEDs

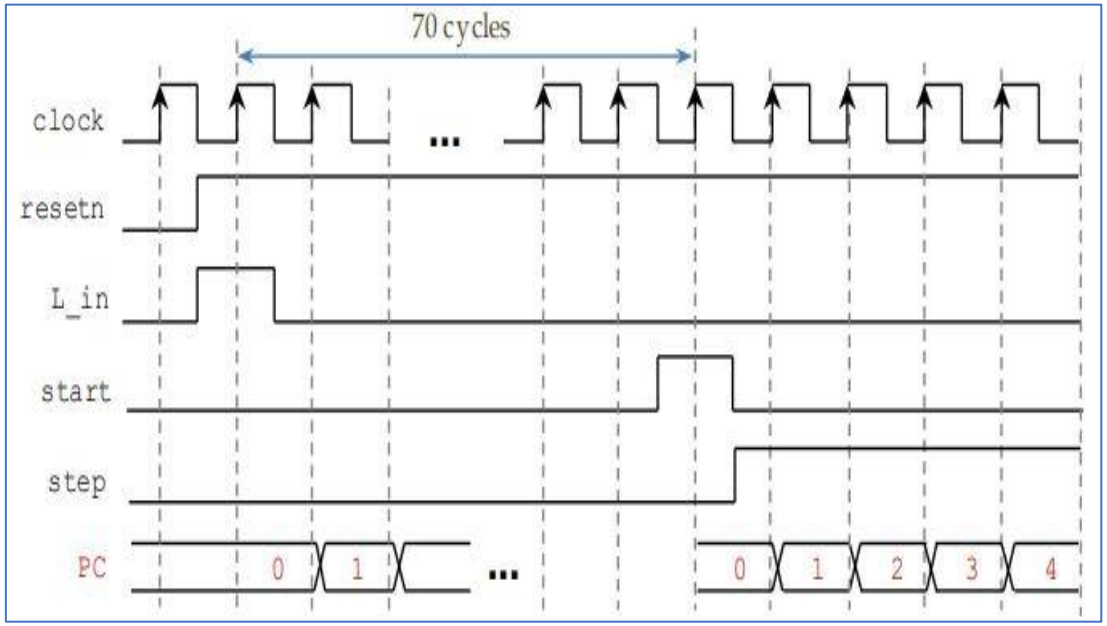



mySmfull_v1_0.vhd
mySM_ip.vhd
S_AXI_AWID
S_AXI_AWADDR 6
S_AXI_AWLEN 8
S_AXI_AWSIZE 3
S_AXI_AWBURST 2
S_AXI_AWVALID
S_AXI_AWREADY
S_AXI_WDATA 32
S_AXI_WSTRB 4
S_AXI_WLAST
S_AXI_WVALID
S_AXI_WREADY
S_AXI_BID
S_AXI_BRESP 2
S_AXI_BVALID
S_AXI_BREADY
S_AXI_ACLK
S_AXI_ARESETN
S_AXI_ARID
S_AXI_ARADDR 6
S_AXI_ARLEN 8
S_AXI_ARSIZE 3
S_AXI_ARBURST 2
S_AXI_ARVALID
S_AXI_ARREADY
S_AXI_RDATA 32
S_AXI_RID
S_AXI_RRESP 2
S_AXI_RLAST
S_AXI_RVALID
S_AXI_RREADY
iFIFO 512x32
FWFT
DI
wren
DO
rden
full
empty
rst
DI 32
Up_VBS.vhd
4
DO 32
oFIFO 512x32
FWFT
DI
wren
DO
rden
rst
full
empty
cs
resetn
L_in
start
step
irden
iempty
FSM
owren
ofull
ifull
rst
iwren
FSM
oempty
orden
mem_rden
mem_wren
axi_arv_arr_flag
axi_rvalid
CLKFX

- mySMfull_v1_0.vhd
  - mySMfull_v1_o_S00_AXI.vhd
    - myAXI_IP.vhd
      - my_AXIfifo.vhd
        - my_genpulse_sclr.vhd
        - mySM_ip.vhd
          - Up_VBS.vhd
            - instload_ctrl.vdh (FSM)
            - pcounter.vhd
              - super_addsub.vhd
                - fulladd.vhd … [5]
              - my_rege.vhd
            - Inst_Dec8.vhd
            - RAM_emul.vhd
              - my_rege.vhd … [16]
            - datapath_VBC.vhd
              - my_rege.vhd … [3]
              - alu_vbc.vhd
                - my_addsub.vhd
                  - fulladd.vhd … [4]

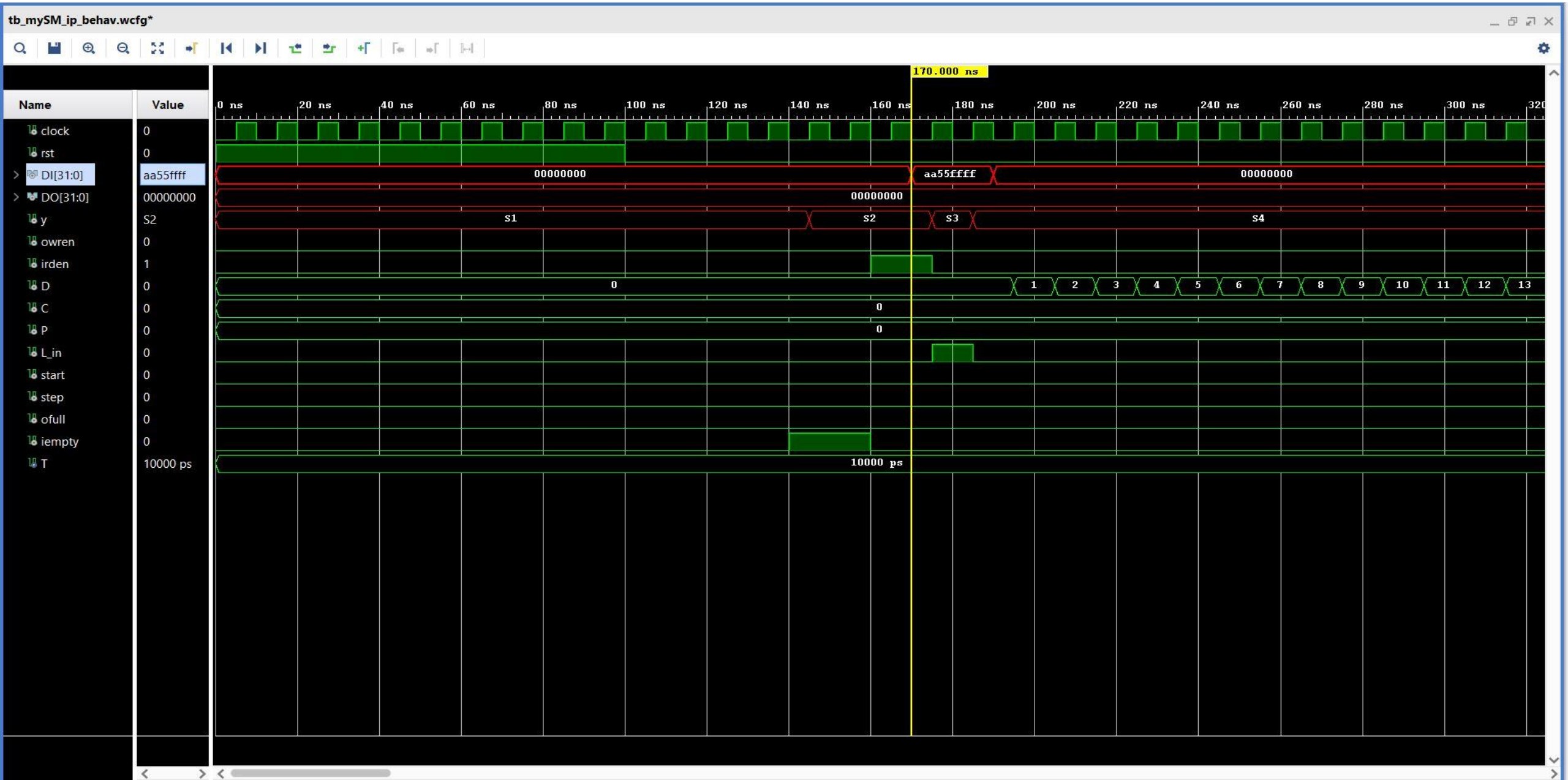

rst=1
S1
iempty
0
1
S2
iempty=0
&ofull=0
no
yes
irden←1
cs=
0xAA55FFFF
no
yes
S3
L_in ← 1
S4
D=20
no
D ← D+1
yes
Start ← 1; D← 0
S5
Step ← 1
C=2
no
C ← C+1
yes
C ← 0
S6
owren ← 1
S7
Step ← 1
C=3
no
C ← C+1
yes
C ← 0
S8
owren ← 1
P=6
no
P ← P+1
yes
P ← 0
S9
Step ← 1
S10
Step ← 1
S2


CS
resetn
L_in
start
step
irden
iempty
FSM
owren

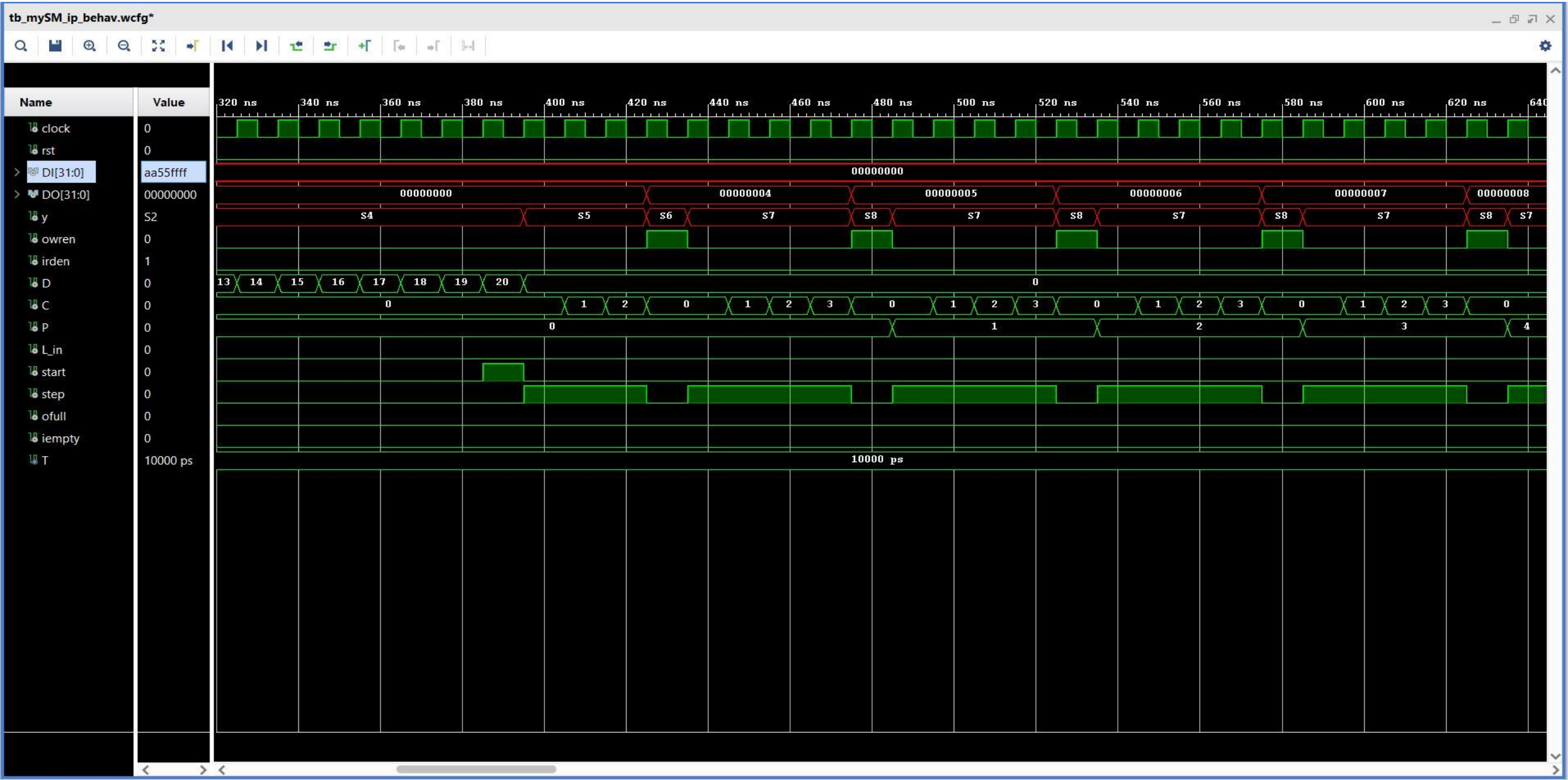

70 cycles
clock
resetn
L_in
start
step
PC
0
1
0
1
2
3
4

Result & discussions

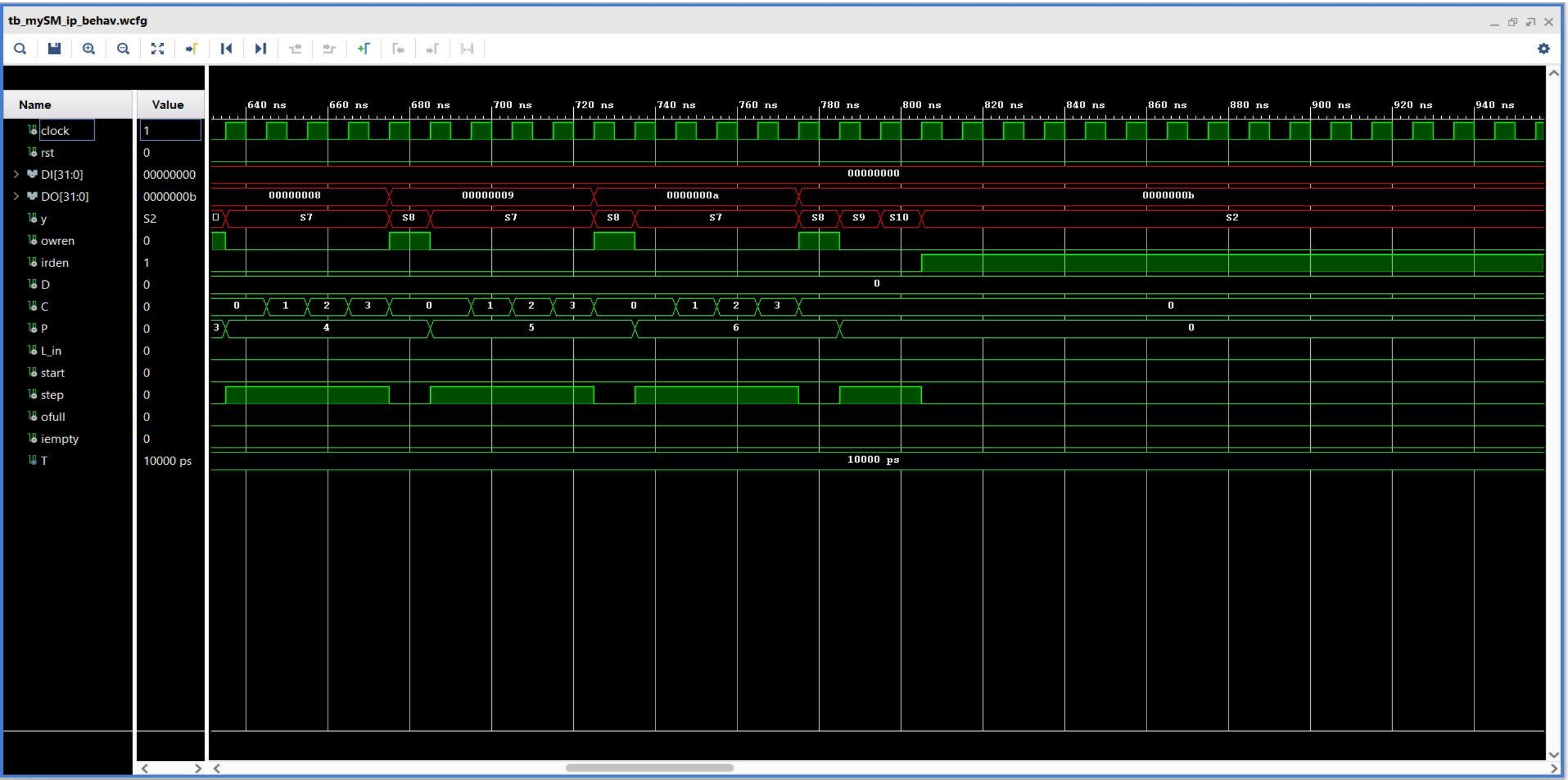
mySM_ip_OneRound1



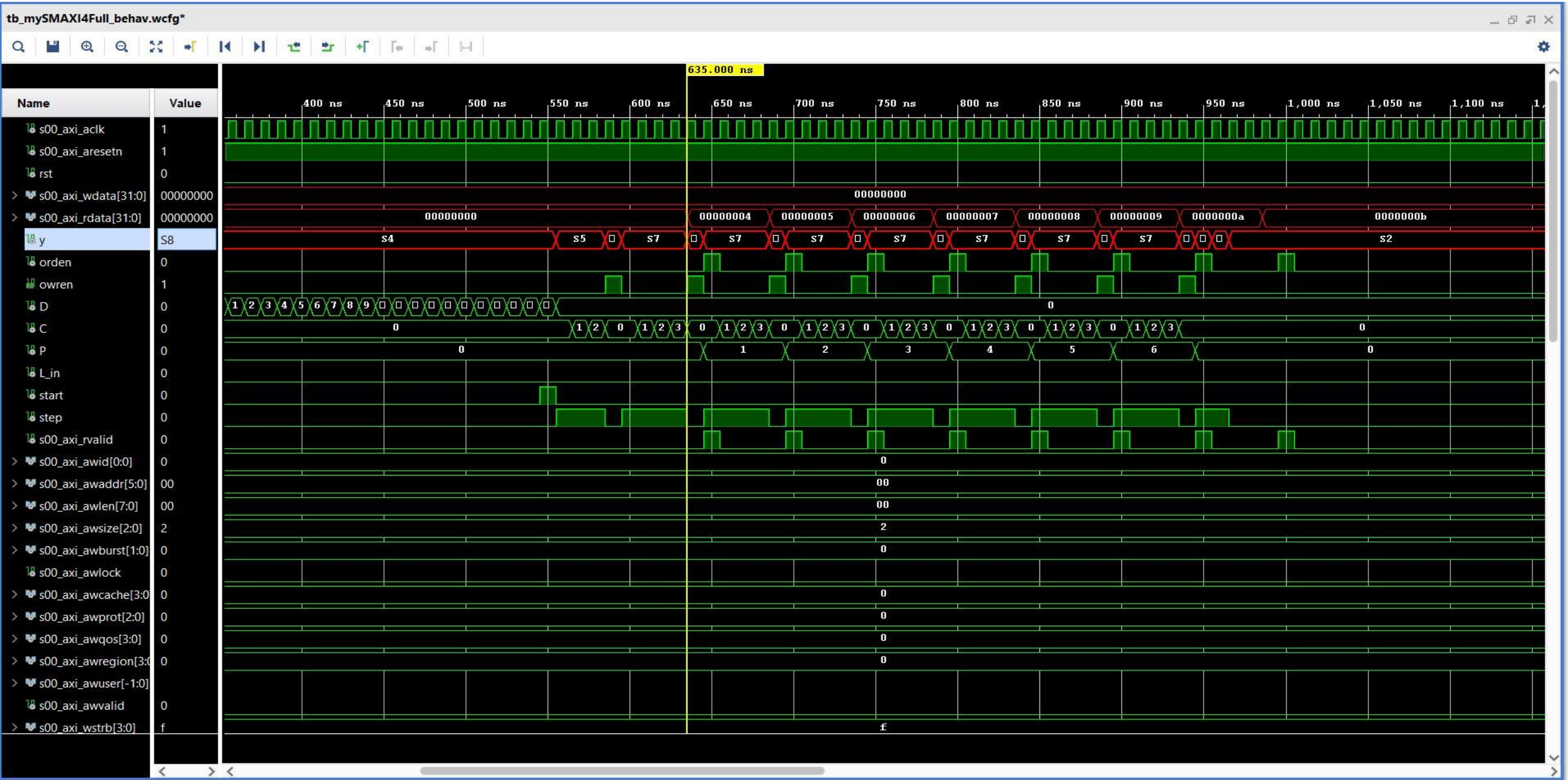
mySM_ip _OneRound2

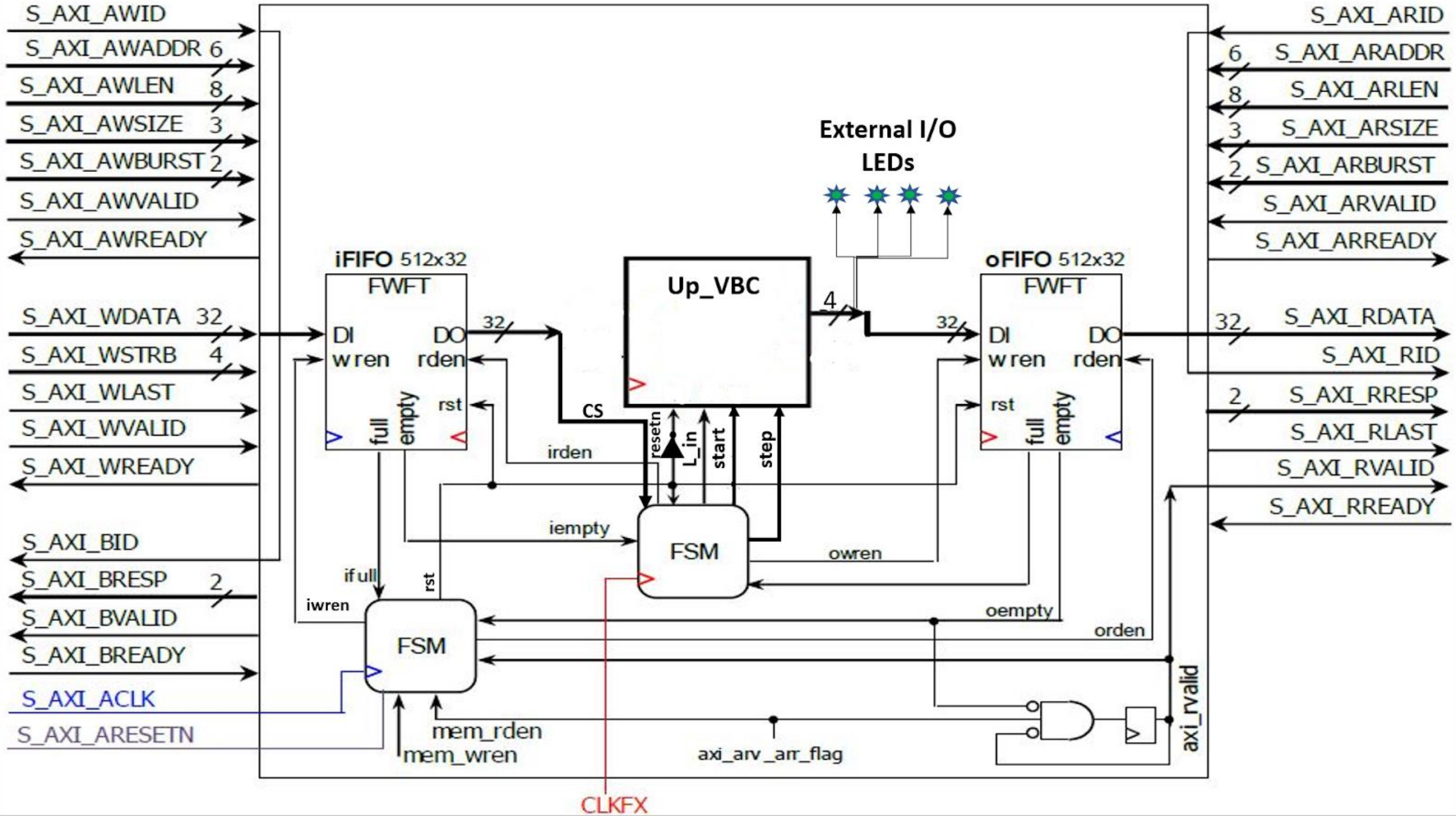


mySM_ip _OneRound3

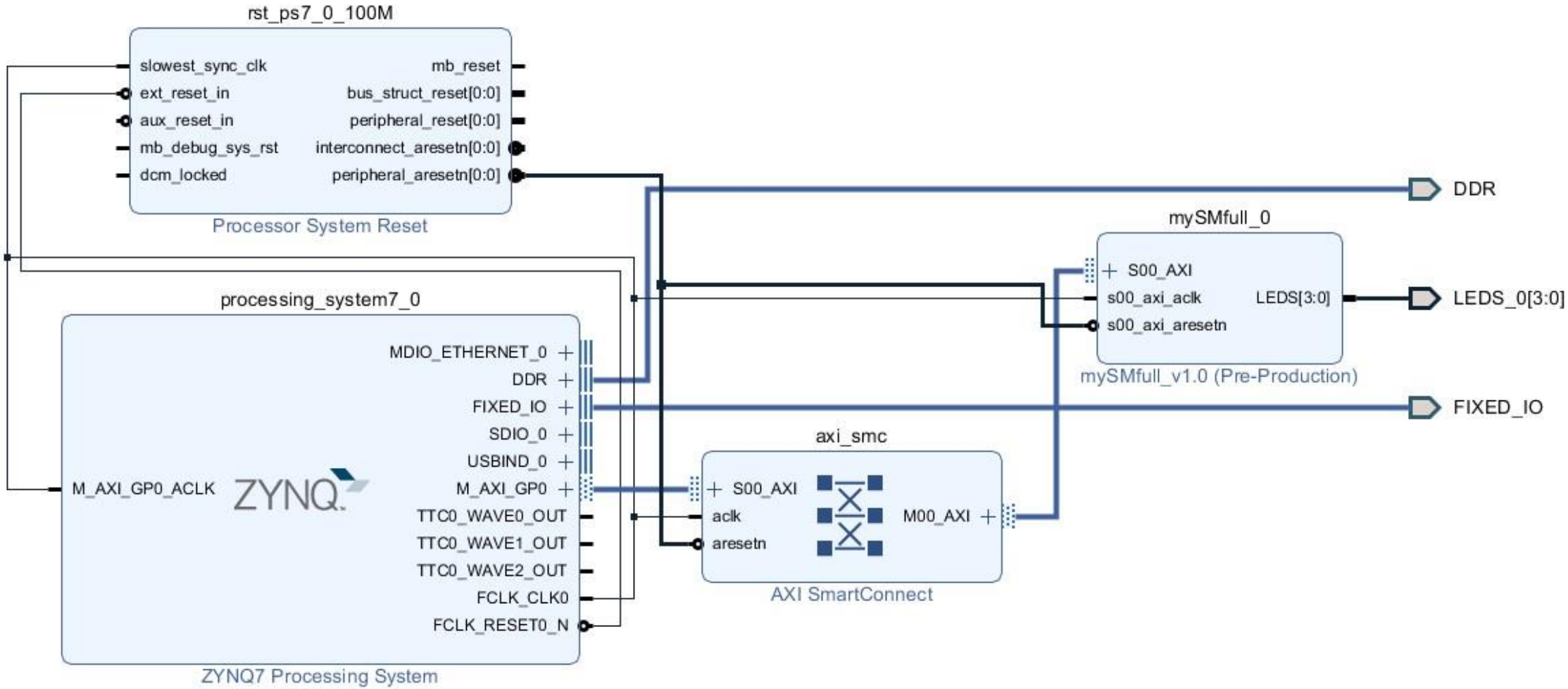


mySMfull_v1_0 _OneRound

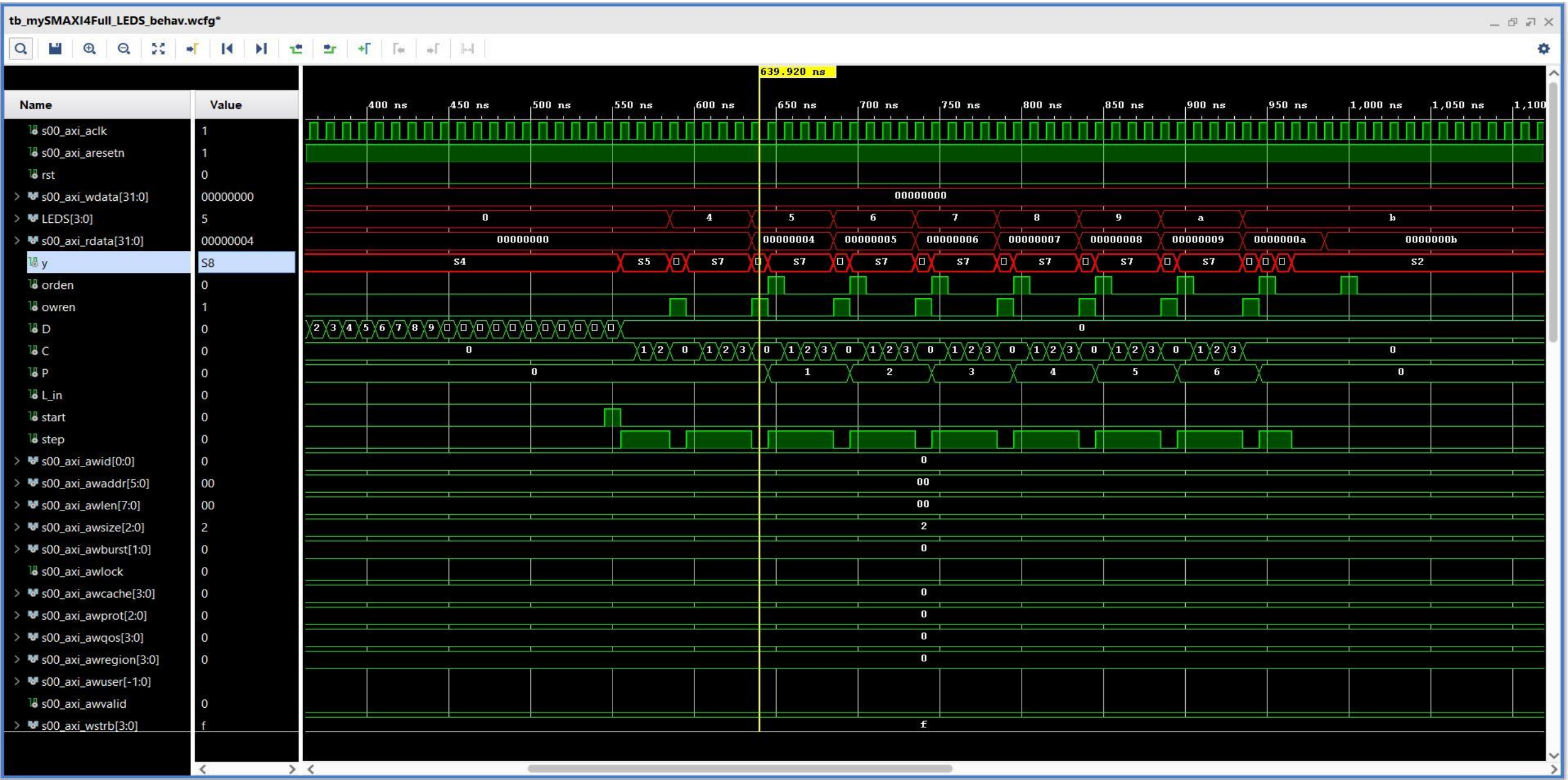


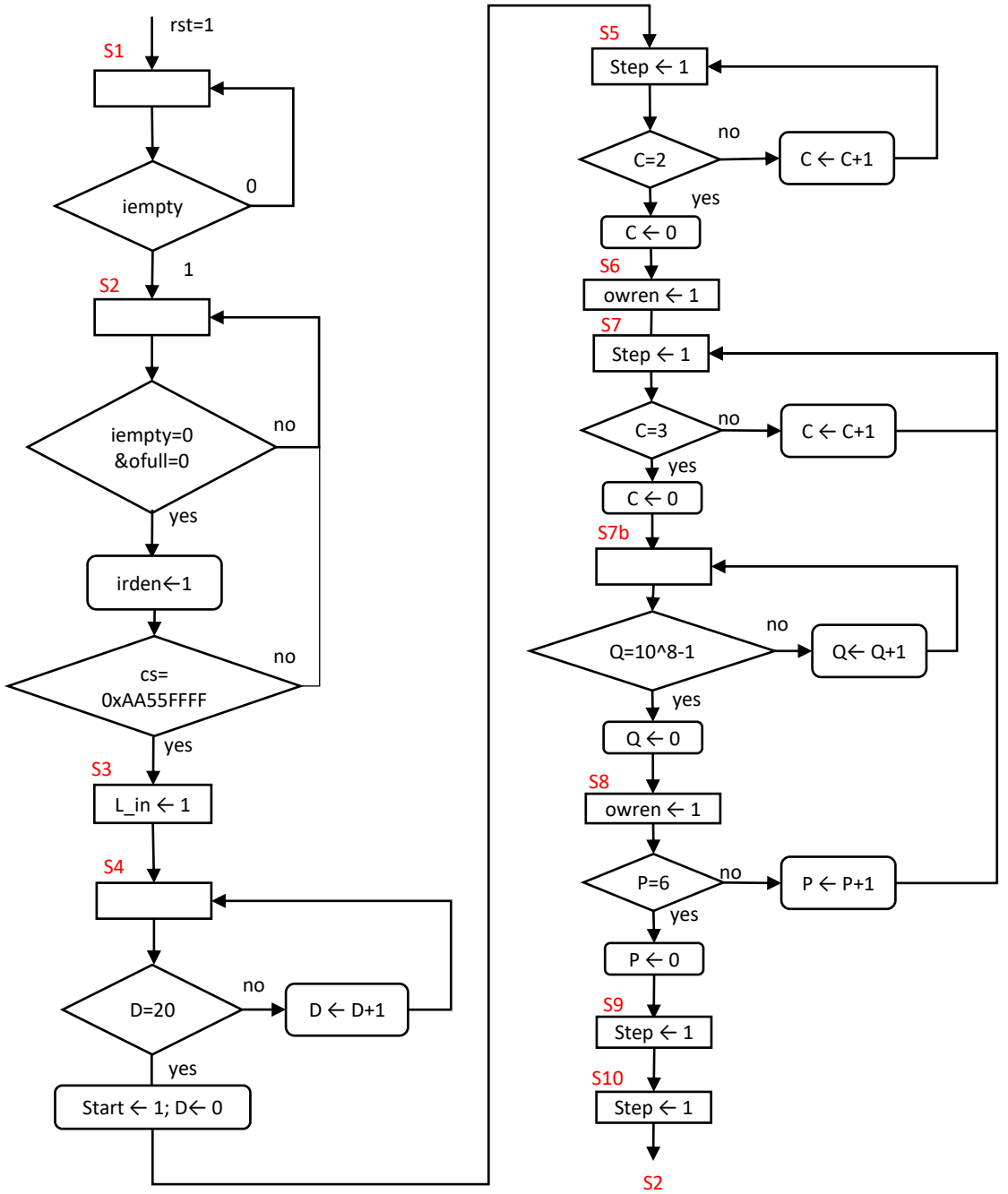

S_AXI_AWID
S_AXI_AWADDR 6
S_AXI_AWLEN 8
S_AXI_AWSIZE 3
S_AXI_AWBURST 2
S_AXI_AWVALID
S_AXI_AWREADY
S_AXI_WDATA 32
S_AXI_WSTRB 4
S_AXI_WLAST
S_AXI_WVALID
S_AXI_WREADY
S_AXI_BID
S_AXI_BRESP 2
S_AXI_BVALID
S_AXI_BREADY
S_AXI_ACLK
S_AXI_ARESETN
S_AXI_ARID
S_AXI_ARADDR 6
S_AXI_ARLEN 8
S_AXI_ARSIZE 3
S_AXI_ARBURST 2
S_AXI_ARVALID
S_AXI_ARREADY
S_AXI_RDATA 32
S_AXI_RID
S_AXI_RRESP 2
S_AXI_RLAST
S_AXI_RVALID
S_AXI_RREADY
External I/O
LEDs
iFIFO 512x32
FWFT
DI
wren
DO
rden
full
empty
rst
32
Up_VBC
4
CS
resetn
L_in
start
step
irden
iempty
FSM
owren
oFIFO 512x32
FWFT
DI
wren
DO
rden
rst
full
empty
32
ifull
rst
iwren
FSM
oempty
orden
mem_rden
mem_wren
axi_arv_arr_flag
axi_rvalid
CLKFX

mySMfull IP Design



mySMfull_v1_0 _LEDs_OneRound

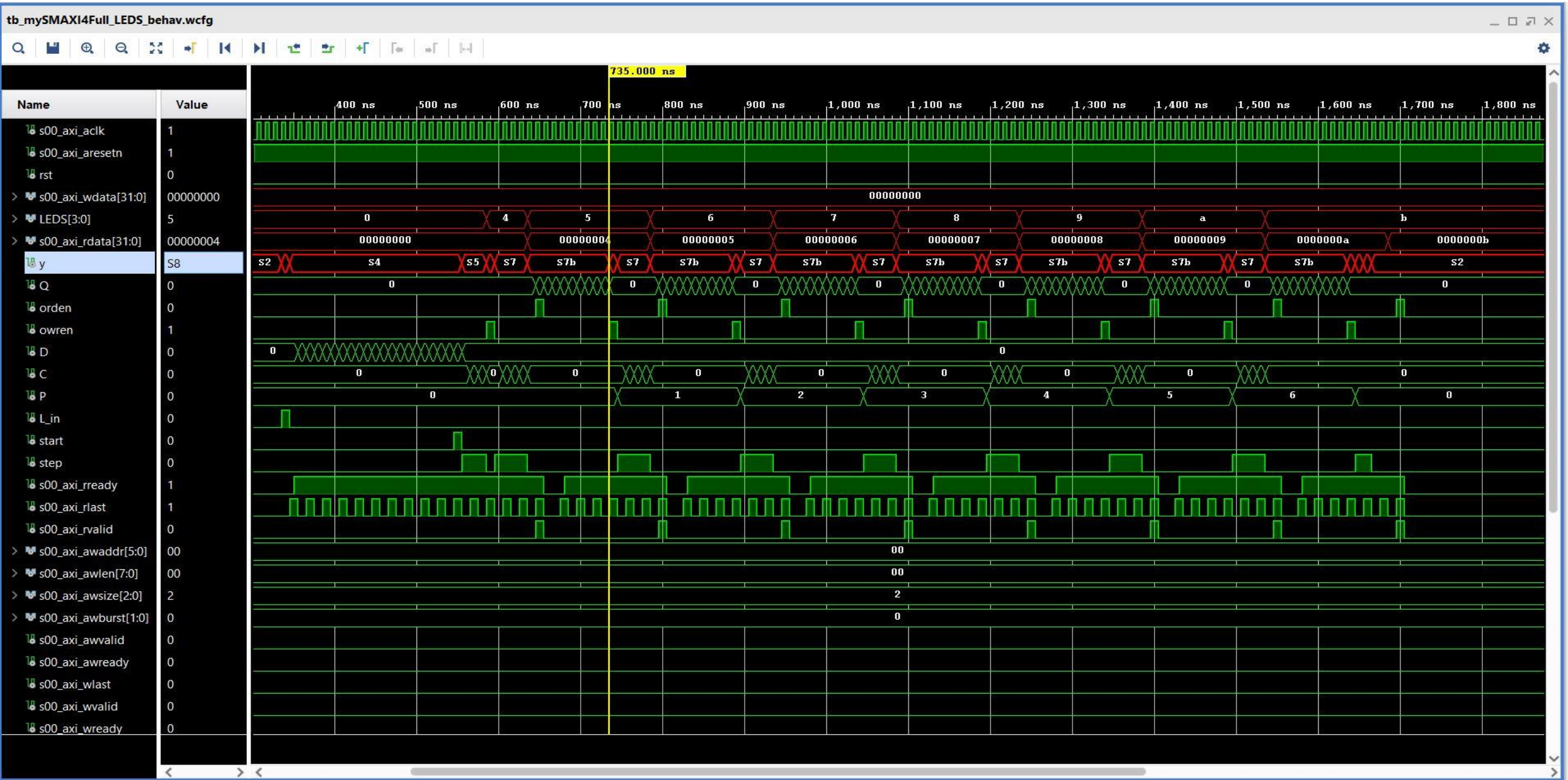




T0 = 8*10^(-9)*4
= 32 nS

T = 8*10^(-9)*10^8
= 0.8 S

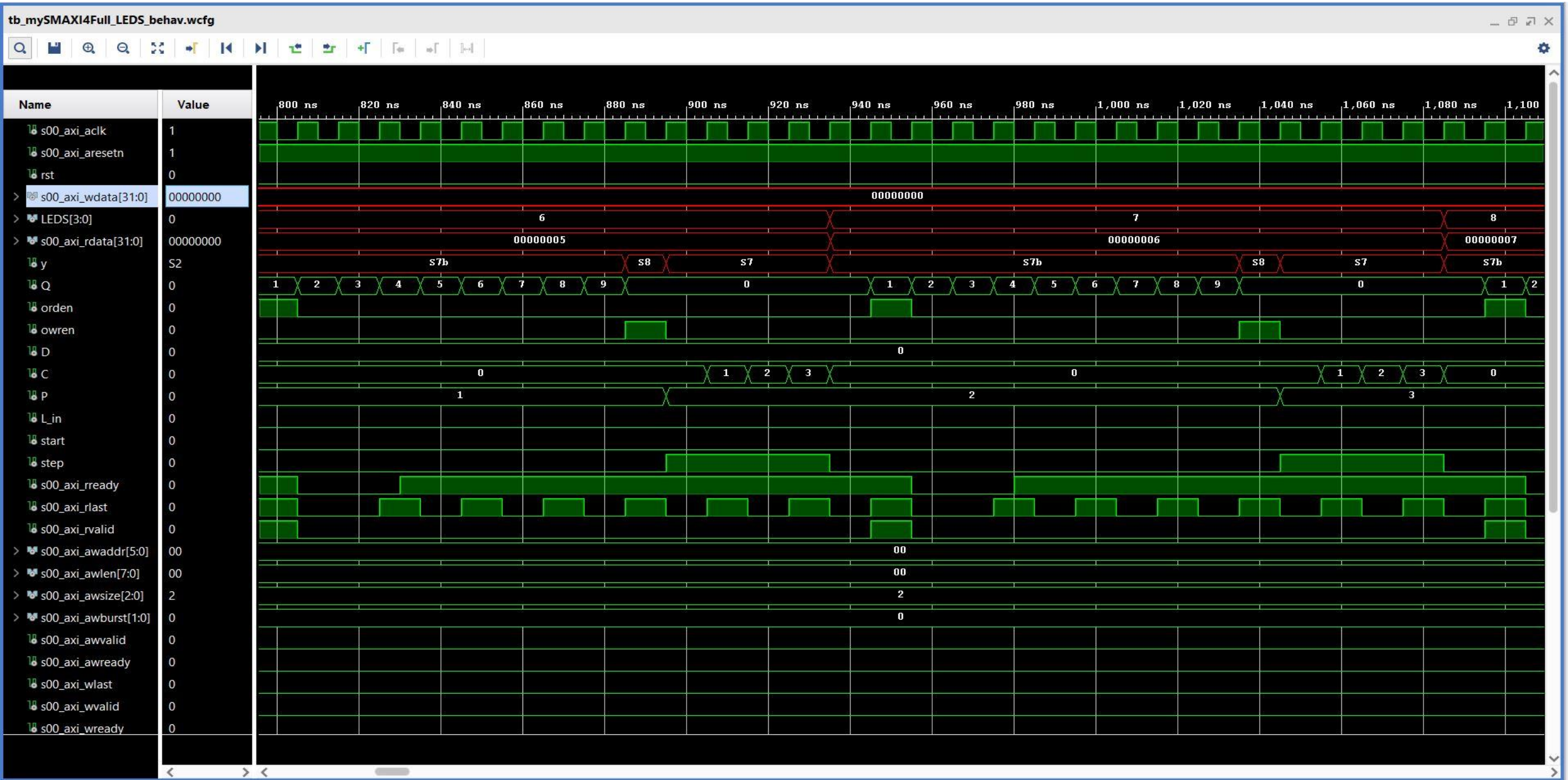
mySMfull_v1_0 _LEDs_Delay_OneRound



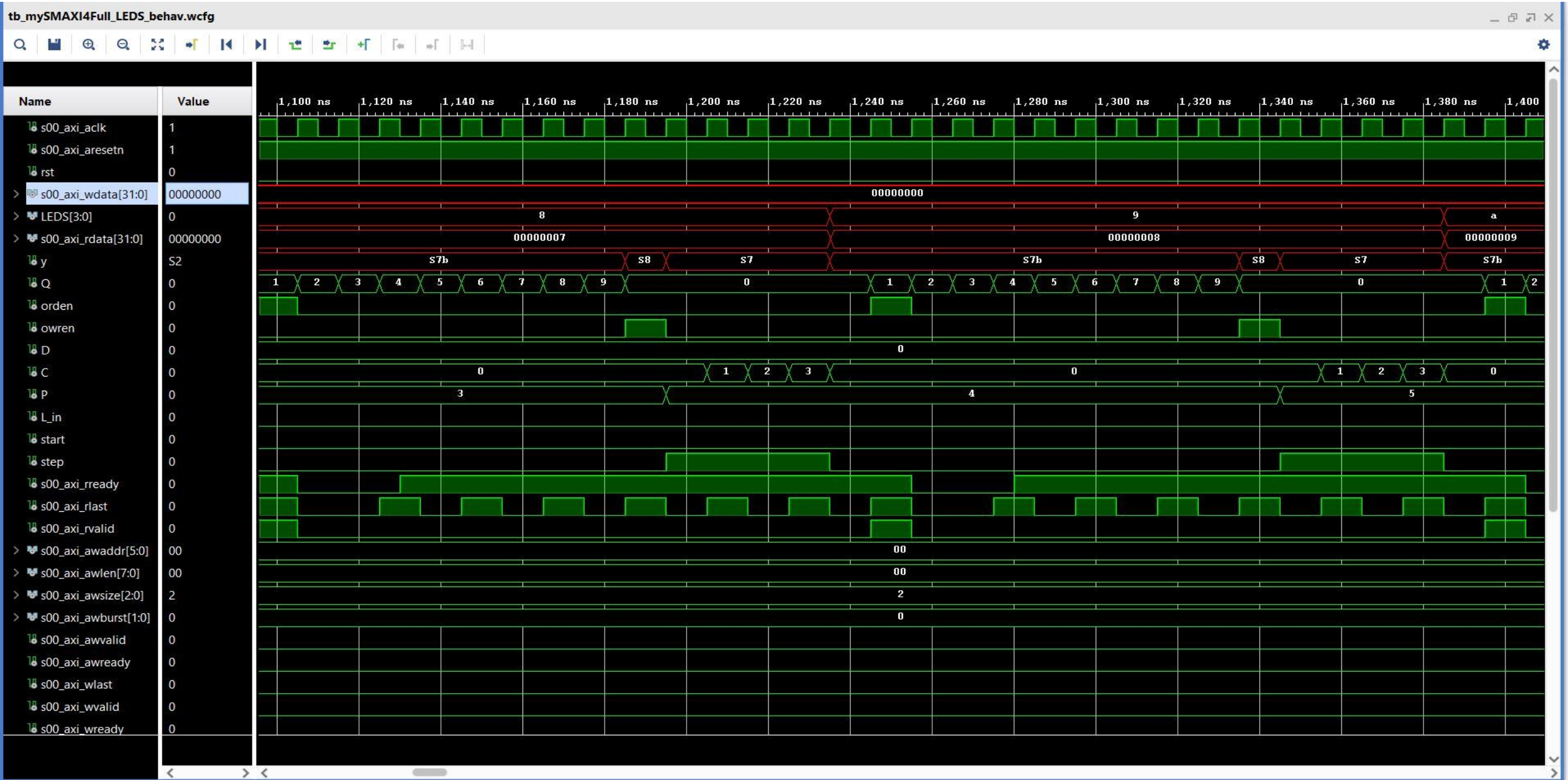
mySMfull_v1_0 _LEDs_Delay_MultiRound1

mySMfull_v1_0 _LEDs_Delay_MultiRound2

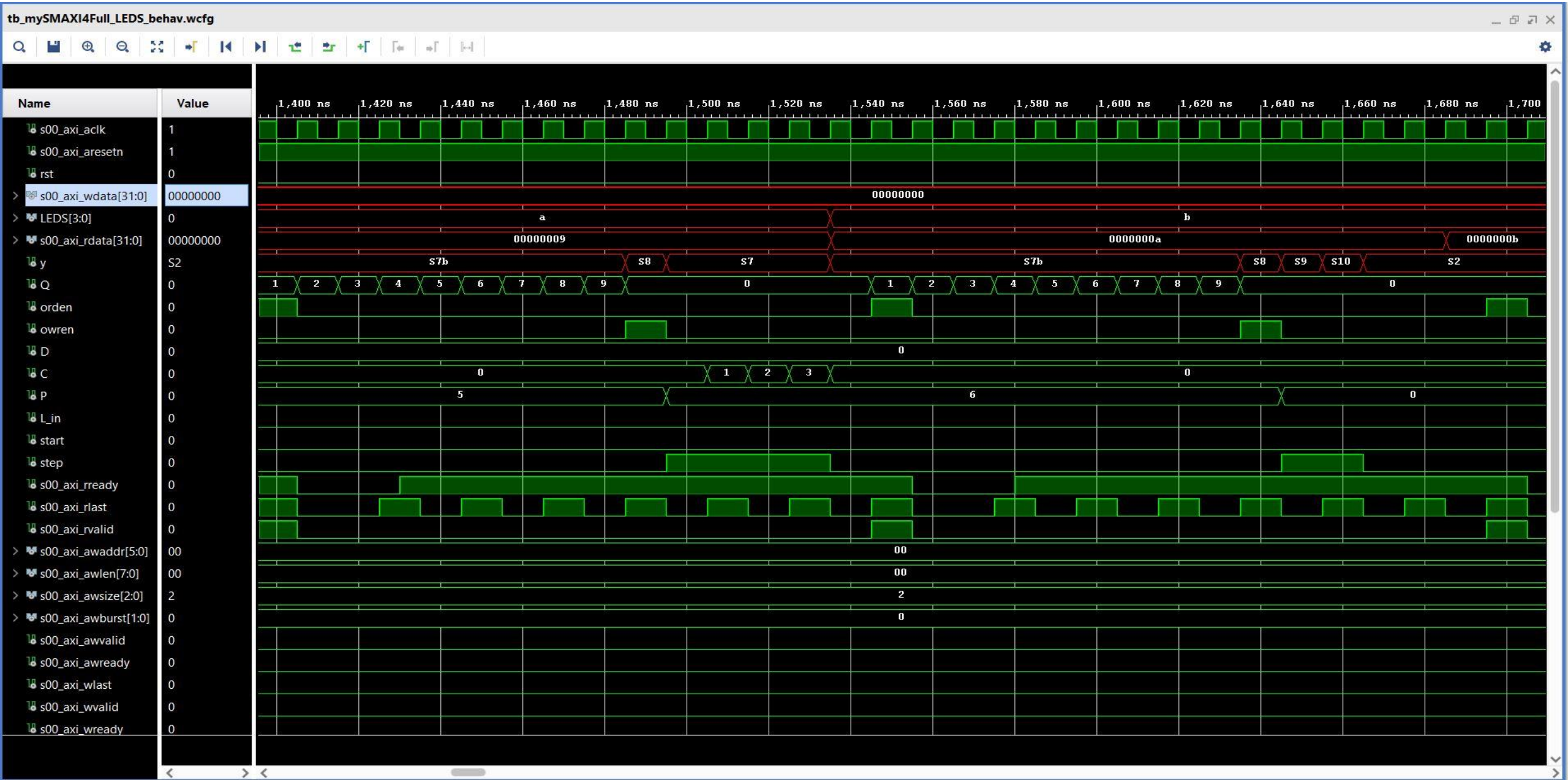
mySMfull_v1_0 _LEDs_Delay_MultiRound3



mySMfull_v1_0 _LEDs_Delay_MultiRound4

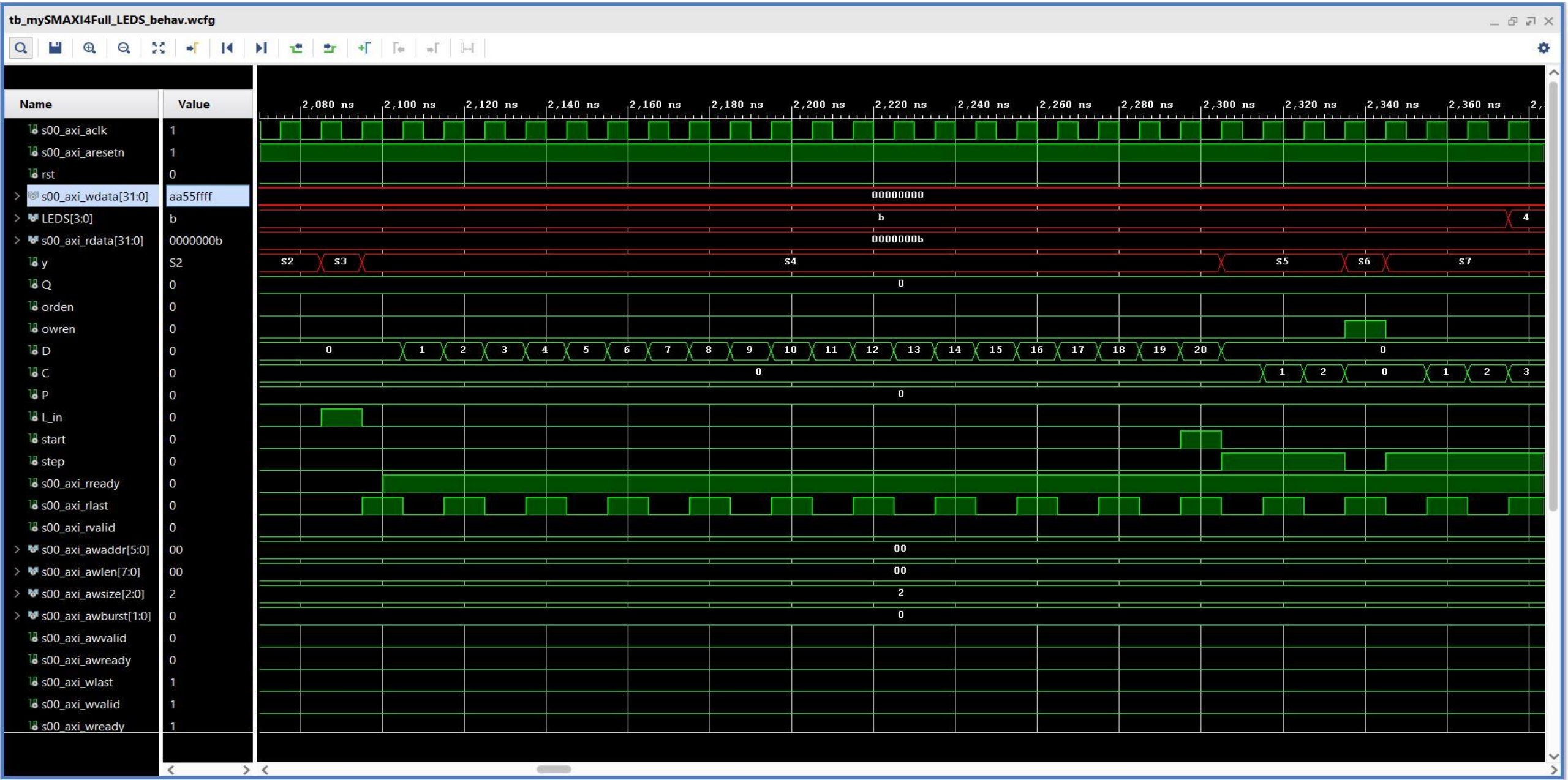


mySMfull_v1_0 _LEDs_Delay_MultiRound5

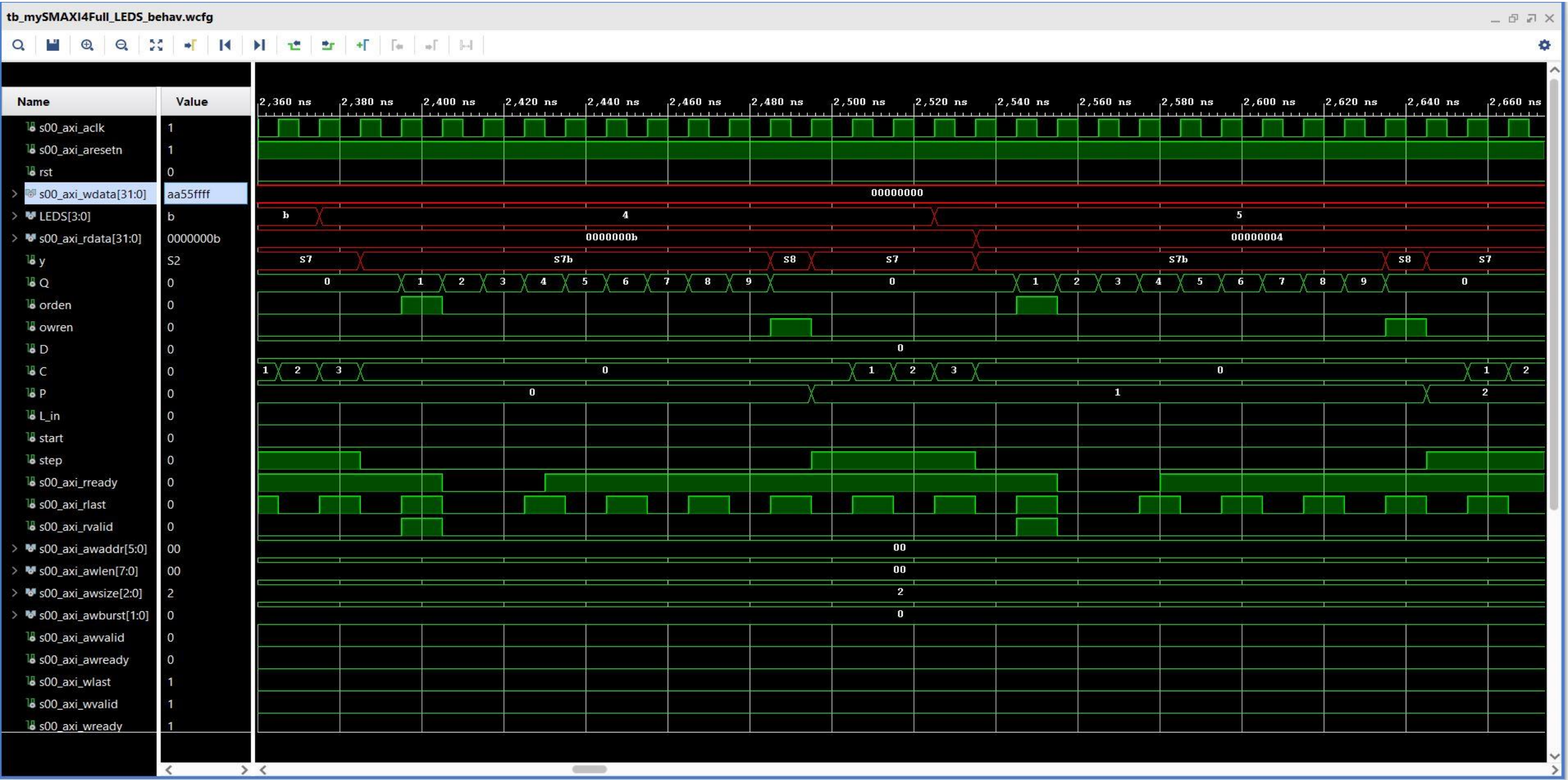


mySMfull_v1_0 _LEDs_Delay_MultiRound6

mySMfull_v1_0 _LEDs_Delay_MultiRound7

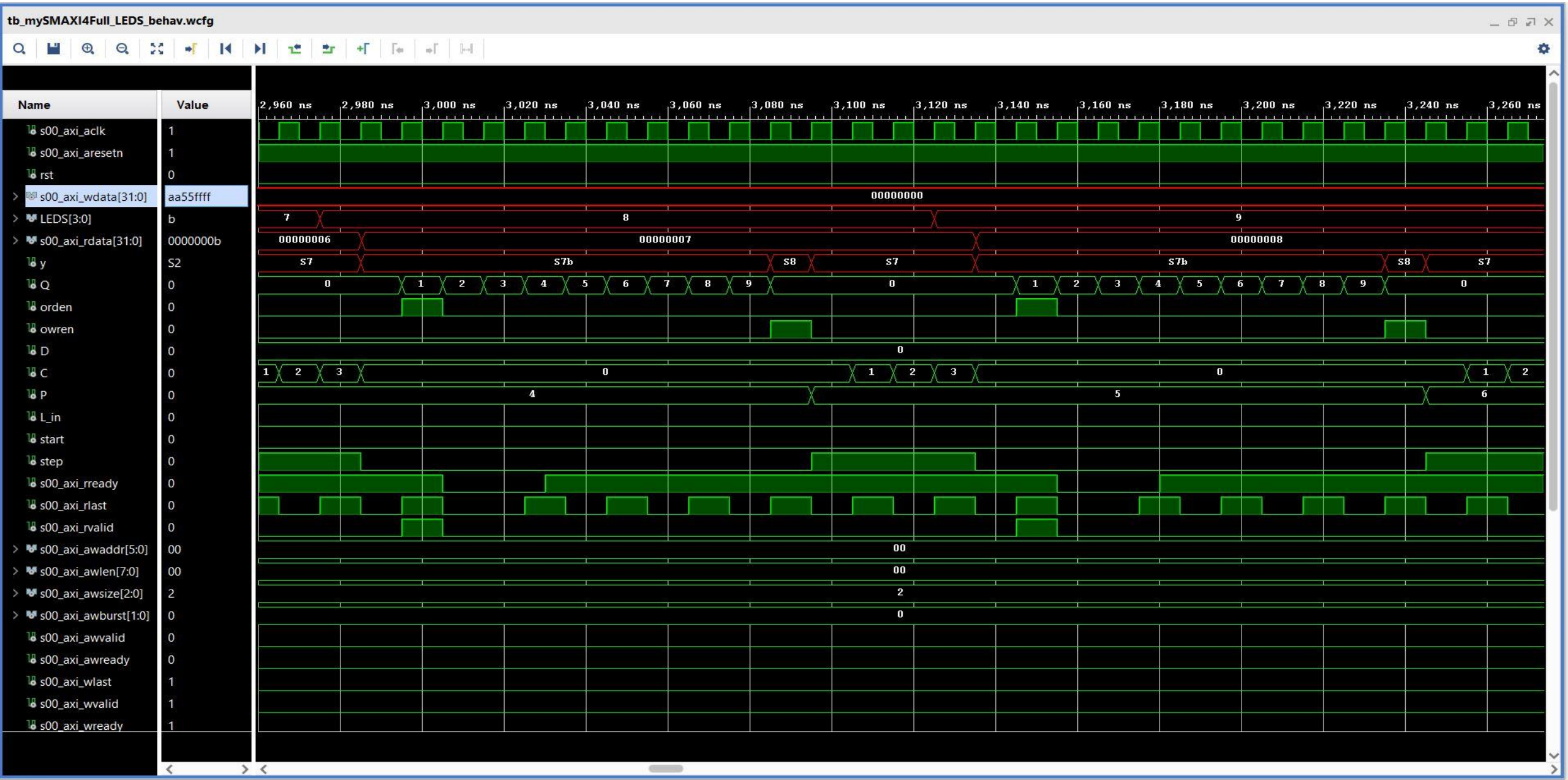


mySMfull_v1_0 _LEDs_Delay_MultiRound8



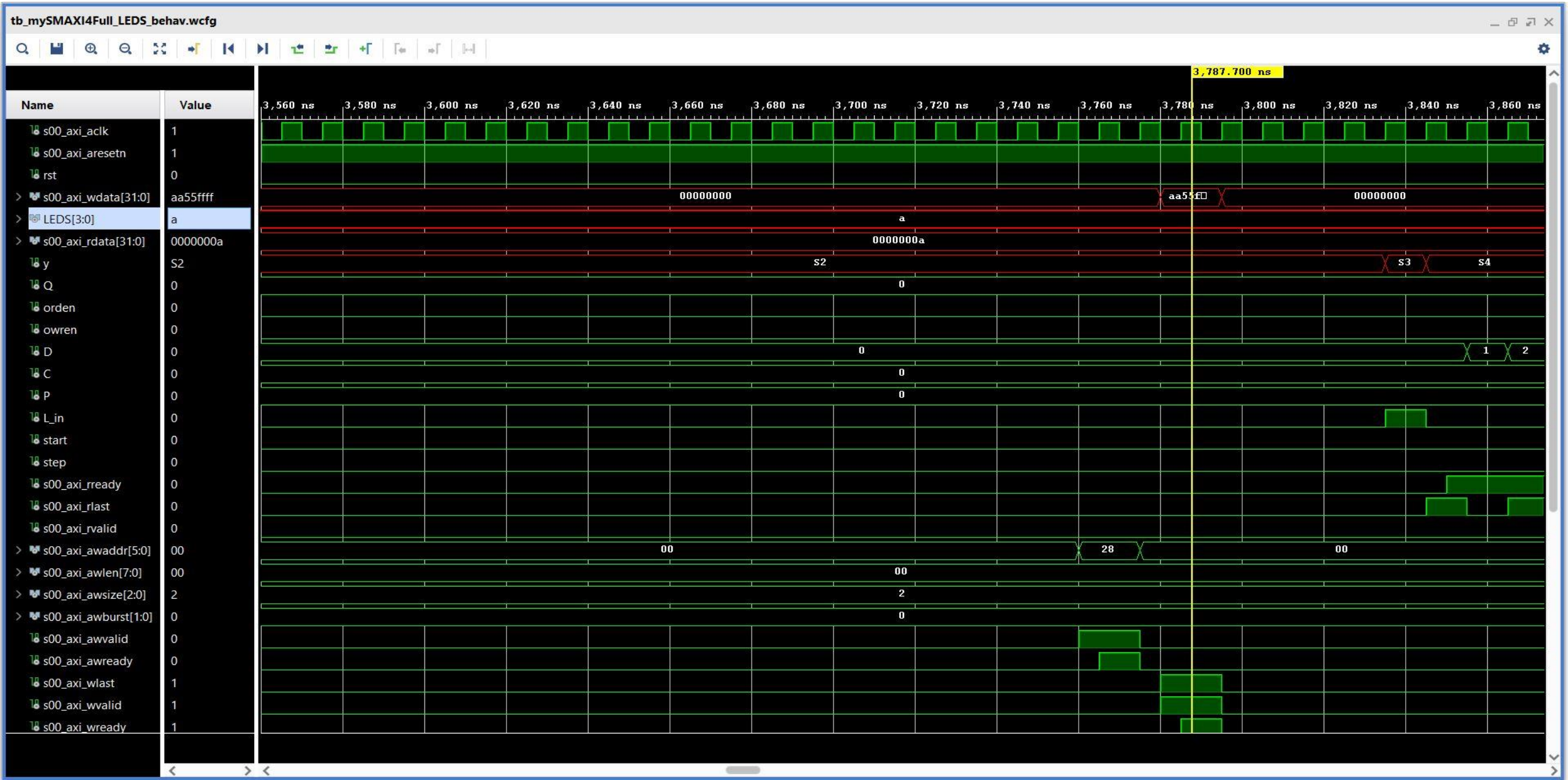
mySMfull_v1_0 _LEDs_Delay_MultiRound9

mySMfull_v1_0 _LEDs_Delay_MultiRound10



mySMfull_v1_0 _LEDs_Delay_MultiRound11

mySMfull_v1_0 _LEDs_Delay_MultiRound12



Conclusions & Future work

- The control signal input data **0xAA55FFFF** is provided to the designed.
- The CPU operation is triggered by the **FSM** and start counting provided counter sequences.
- The counter will count value from **0x04 to 0x0B** and repeats the same counter if the state machine is triggered again.
- The **LEDs** are used as external outputs of the Zybo board.
- **All over**, this project was great learning from the class materials and also provides the **platform** to grow our skills in related filed within industries.
- We **believe** that this project work was very **didactic** to learn about **real-world technology**.
- This design could be **expanded** into a more complex processor with a stack to allow for less reading and writing between the software and hardware.
- The **instruction set** could be expanded from a **3-bit** to a **4-bit** or higher set depending on the required project complexity.
- The **software** can be used with more than two processors.
- The **reconfigurable partition** can be designed to allow multiple configuration switches at run time.

Demo

- Project_Demo_Final.mp4 - Google Drive

- (Also provided with submission folder)

Thank you !!

Any Question?