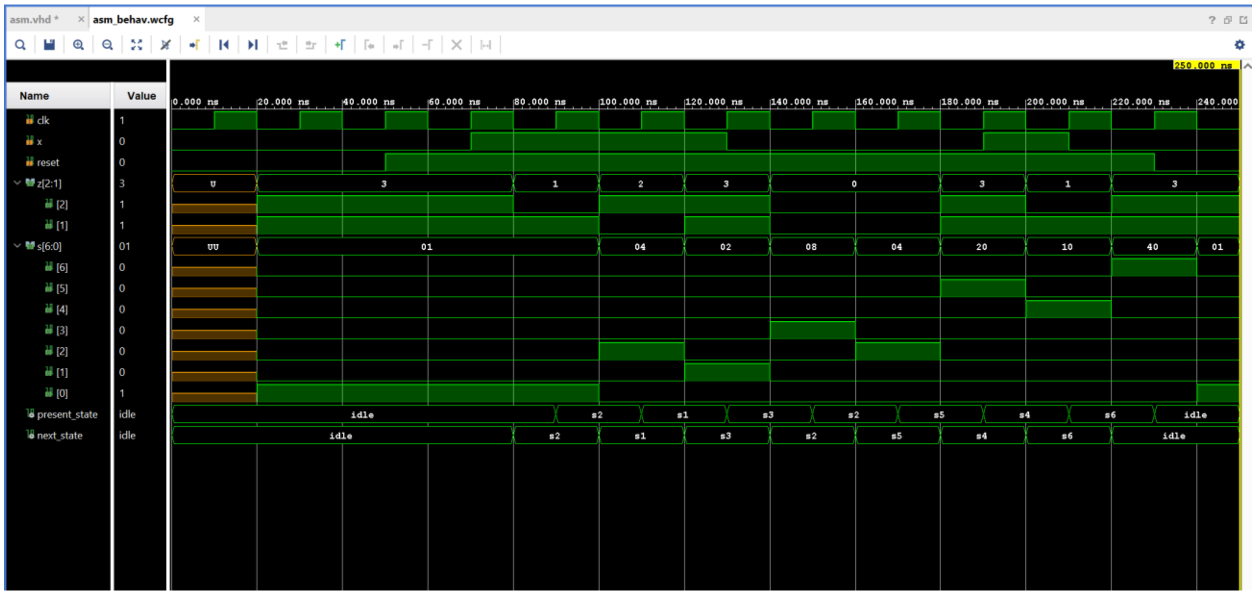
Mack Usmanova

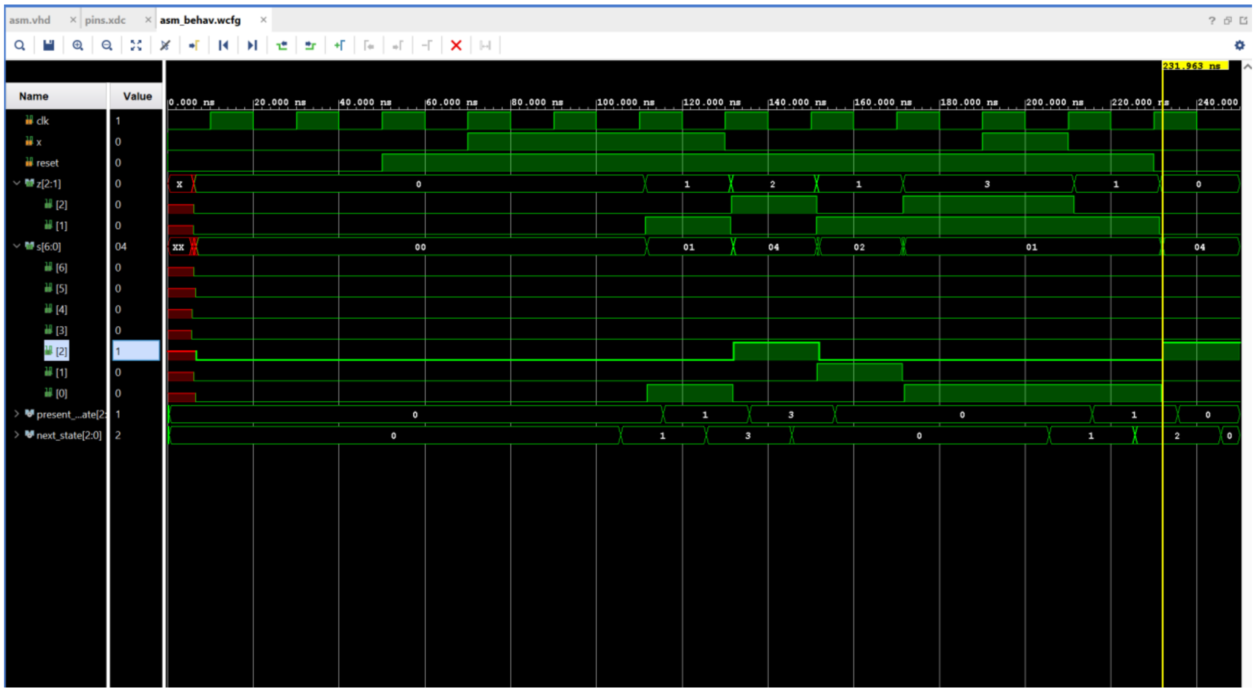
Computer engineer

Homework 4

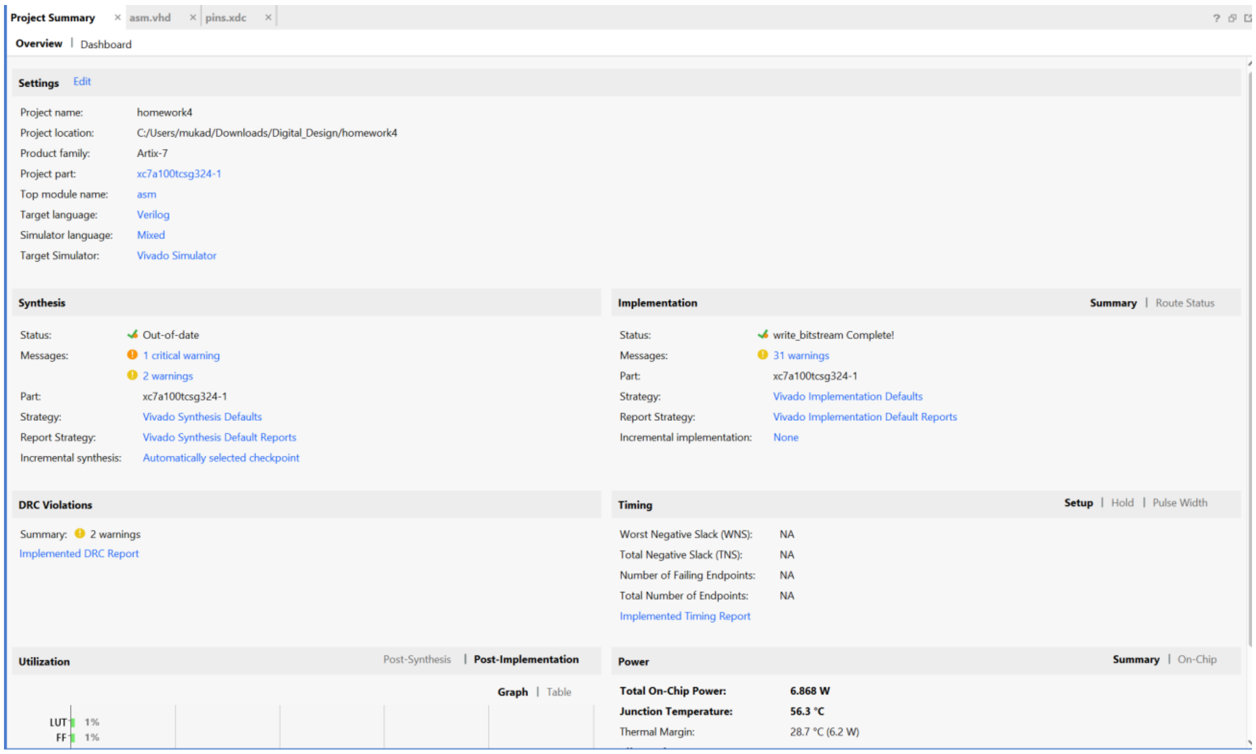
10/2/2025



The clock cycle was every 20 ns. Even though I made the x input every rising edge, you can see its not synchronized by "next state" until falling edge occurs. the present state on the other hand, which is the state we care about isn't changing until a rising edge occurs. I made sure to arrange my x inputs in a way that every state is outputted at least once. The z outputs also are correct depending to x input and present state.



this simulation has a different output then the behavioral simulation. I believe it is because this one is being more delayed then the behavioral simulation. For the output of this simulation to look similar to the behavoral I could add ns delays to the earlier inputs.


Project Summary | asm.vhd | pins.xdc

Overview | Dashboard

Settings Edit

Project name:	homework4
Project location:	C:/Users/mukad/Downloads/Digital_Design/homework4
Product family:	Artix-7
Project part:	xc7a100tcsg324-1
Top module name:	asm
Target language:	Verilog
Simulator language:	Mixed
Target Simulator:	Vivado Simulator

Synthesis

Status:	Out-of-date
Messages:	1 critical warning
	2 warnings
Part:	xc7a100tcsg324-1
Strategy:	Vivado Synthesis Defaults
Report Strategy:	Vivado Synthesis Default Reports
Incremental synthesis:	Automatically selected checkpoint

Implementation Summary | Route Status

Status:	write_bitstream Complete!
Messages:	31 warnings
Part:	xc7a100tcsg324-1
Strategy:	Vivado Implementation Defaults
Report Strategy:	Vivado Implementation Default Reports
Incremental implementation:	None

DRC Violations

Summary: 2 warnings

Implemented DRC Report

Timing Setup | Hold | Pulse Width

Worst Negative Slack (WNS):	NA
Total Negative Slack (TNS):	NA
Number of Failing Endpoints:	NA
Total Number of Endpoints:	NA

Implemented Timing Report

Utilization Post-Synthesis | Post-Implementation

Graph | Table

LUT 1%

FF 1%

Power Summary | On-Chip

Total On-Chip Power:	6.868 W
Junction Temperature:	56.3 °C
Thermal Margin:	28.7 °C (6.2 W)

vhd

```
----------------------------------------------------------------------------------
-- Company: 
-- Engineer: 
-- 
-- Create Date: 10/01/2025 10:01:50 PM
-- Design Name: 
-- Module Name: asm - Behavioral
-- Project Name: 
-- Target Devices: 
-- Tool Versions: 
-- Description: 
-- 
-- Dependencies: 
-- 
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
-- 
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity asm is
  Port (
        clk, x, reset: in std_logic;
        z: out std_logic_vector(2 downto 1);
        s: out std_logic_vector(6 downto 0)
  );
end asm;

architecture Behavioral of asm is
type sreg is (idle,s1,s2,s3,s4,s5,s6);
signal present_state, next_state: sreg:= idle; --initial state idle
begin
    statem: process
    begin
        wait until clk' event and clk ='0';
                case present_state is
                    when idle =>
                    s <= "0000001";
                        if x='0' then
                            z <= "11";
                            next_state <= idle;
                        else
                            z <= "01";
                            next_state <= s2;
                        end if;
                    when s1 =>
                    s <= "0000010";
                        if x='0' then
                            z <= "01";
                            next_state <= idle;
                        else
                            z <= "11";
                            next_state <= s3;
                        end if;
                    when s2 =>
                    s <= "0000100";
                        if x='0' then
                            z <= "00";
                            next_state <= s5;
                        else
                            z <= "10";
                            next_state <= s1;
                        end if;
                    when s3 =>
                    s <= "0001000";
                        if x='0' then
                            z <= "00";
                            next_state <= s2;
                        else
                            z <= "10";
                            next_state <= s4;
                        end if;
                    when s4 =>
                     s <= "0010000";
                        if x='0' then
                            z <= "11";
                            next_state <= s1;
                        else
                            z <= "01";
                            next_state <= s6;
                        end if;
                    when s5 =>
                    s <= "0100000";
                        if x='0' then
                            z <= "11";
                            next_state <= s4;
                        else
```

```
                             z <= "00";
                             next_state <= s5;
                         end if;
                     when s6 =>
                     s <= "1000000";
                         if x='0' then
                             z <= "11";
                             next_state <= idle;
                         else
                             z <= "01";
                             next_state <= s2;
                         end if;
                     when others =>
                         z <= "00";
                         next_state <= idle;
                     end case;
    end process statem;

clk_process: process
       begin
       wait until clk' event and clk = '1'; --wait until the rising edge
       if reset = '0' then
          present_state <= idle;
       else
          present_state <= next_state;
    end if;
end process clk_process;

end Behavioral;
```

Xdc

```
## This file is a general .xdc for the Nexys A7-100T

## To use it in a project:
## - uncomment the lines corresponding to used pins
## - rename the used ports (in each line, after get_ports) according to the top level signal
names in the project
## Clock signal
#set_property CLOCK_DEDICATED_ROUTE FALSE [get_nets clk_IBUF]

set_property -dict { PACKAGE_PIN E3 IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L12P_T1_MRCC_35
Sch=clk100mhz

create_clock -add -name clk -period 10.00 -waveform {0 5} [get_ports {clk}];

set_property -dict { PACKAGE_PIN J15 IOSTANDARD LVCMOS33 } [get_ports { x }]; #IO_L24N_T3_RS0_15
Sch=sw[0]

set_property  -dict  {  PACKAGE_PIN  H17  IOSTANDARD  LVCMOS33  }  [get_ports  {  s[0]  }];
#IO_L18P_T2_A24_15 Sch=led[0]
set_property  -dict  {  PACKAGE_PIN  K15  IOSTANDARD  LVCMOS33  }  [get_ports  {  s[1]  }];
#IO_L24P_T3_RS1_15 Sch=led[1]
set_property  -dict  {  PACKAGE_PIN  J13  IOSTANDARD  LVCMOS33  }  [get_ports  {  s[2]  }];
#IO_L17N_T2_A25_15 Sch=led[2]
set_property -dict { PACKAGE_PIN N14 IOSTANDARD LVCMOS33 } [get_ports { s[3] }]; #IO_L8P_T1_D11_14
Sch=led[3]
set_property -dict { PACKAGE_PIN R18 IOSTANDARD LVCMOS33 } [get_ports { s[4] }]; #IO_L7P_T1_D09_14
Sch=led[4]
set_property  -dict  {  PACKAGE_PIN  V17  IOSTANDARD  LVCMOS33  }  [get_ports  {  s[5]  }];
#IO_L18N_T2_A11_D27_14 Sch=led[5]
set_property  -dict  {  PACKAGE_PIN  U17  IOSTANDARD  LVCMOS33  }  [get_ports  {  s[6]  }];
#IO_L17P_T2_A14_D30_14 Sch=led[6]
#set_property  -dict  {  PACKAGE_PIN  U16  IOSTANDARD  LVCMOS33  }  [get_ports  {  clokk  }];
#IO_L18P_T2_A12_D28_14 Sch=led[7]

set_property  -dict  {  PACKAGE_PIN  V16  IOSTANDARD  LVCMOS33  }  [get_ports  {  z[2]  }];
#IO_L16N_T2_A15_D31_14 Sch=led[8]
```

```
set_property -dict { PACKAGE_PIN T15 IOSTANDARD LVCMOS33 } [get_ports { z[1] }]; #IO_L14N_T2_SRCC_14 Sch=led[9]
#set_property -dict { PACKAGE_PIN N17 IOSTANDARD LVCMOS33 } [get_ports { clok }]; #IO_L9P_T1_DQS_14 Sch=btnc
#set_property -dict { PACKAGE_PIN V10 IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L21P_T3_DQS_14 Sch=sw[15]

set_property -dict { PACKAGE_PIN P17 IOSTANDARD LVCMOS33 } [get_ports { reset }]; #IO_L12P_T1_MRCC_14 Sch=btnl
```

Tcl

```
#inputs:
# clk, x, reset STD_LOGIC
#outputs:
# z, s STD_LOGIC

restart

# create train of clock pulses of 100MHz
add_force clk {0 0ns} {1 10ns} -repeat_every 20ns

# apply active low CPU_RESETN
add_force reset {0 0ns}

# initialize inputs
add_force x {0 0ns}

# wait out 100ns
run 50ns

# set active low reset back to inactive state
add_force reset {1 0ns}
run 20ns

# turn on x
add_force x {1 0ns}
run 20ns

# turn off x
add_force x {1 0ns}
run 20ns

# turn on x
add_force x {1 0ns}
run 20ns

# turn off x
add_force x {0 0ns}
run 20ns

# turn on x
add_force x {0 0ns}
run 20ns

# turn off x
add_force x {0 0ns}
run 20ns

# turn on x
add_force x {1 0ns}
run 20ns

# turn off x
add_force x {0 0ns}
```

```
run 20ns

add_force reset {0 0ns}
run 20ns
```