Lab 2 - Task A: Timing simulation

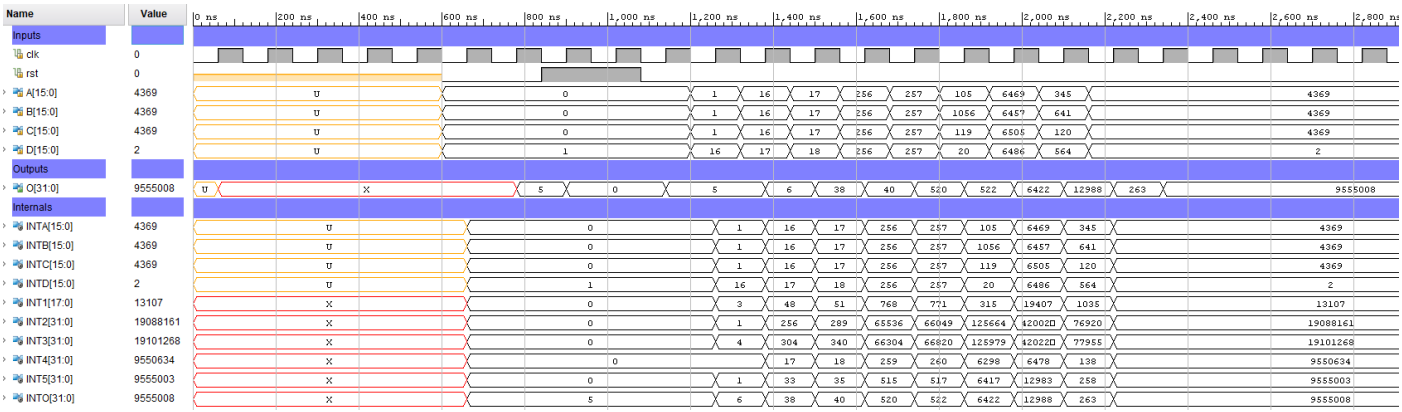
1.1.1: Print out of the self-checking testbench

```
library ieee;
use ieee.std_logic_1164.ALL;
use ieee.numeric_std.ALL;
entity two_process_TB IS

end two_process_TB;
architecture behavior OF two_process_TB IS

    -- Testing Strategy:
    --   This is a two process test bench. This means that it consists of a set of test
      --    vectors and two processes. One of these processes sets the inputs based on the
      --    test vectors and one of the processes checks that the outputs match what is
      --    expected as specified in the test vectors.
    --   There are 10 test vectors in total. They consist of a range of values designed
      --    so that all the different input variables are tested along with the floor
      --    division functionality.

    -- Inputs
    signal A : std_logic_vector(15 downto 0);
    signal B : std_logic_vector(15 downto 0);
    signal C : std_logic_vector(15 downto 0);
    signal D : std_logic_vector(15 downto 0);
    signal clk : std_logic;
    signal rst : std_logic;

    -- Outputs
    signal O : std_logic_vector(31 downto 0);
    constant clk_period : time := 120ns; -- clock period
    constant wait_period: time := 500ns;
    constant latency : natural := 2; -- circuit latency, defined as the number of clock
periods
                                        --   it takes to go from input to output

    -- Define and create a record of test patterns to test the circuit
    type test_vector is record
        A : std_logic_vector(15 downto 0);
        B : std_logic_vector(15 downto 0);
        C : std_logic_vector(15 downto 0);
        D : std_logic_vector(15 downto 0);
        O : std_logic_vector(31 downto 0);
    end record;

    type test_vector_array is array
        (natural range <>) of test_vector;
    constant test_vectors : test_vector_array := (
        -- A, B, C, D, O
        (X"0000", X"0000", X"0000", X"0001", X"00000005"),
        (X"0001", X"0001", X"0001", X"0010", X"00000006"),
        (X"0010", X"0010", X"0010", X"0011", X"00000026"),
        (X"0011", X"0011", X"0011", X"0012", X"00000028"),
        (X"0100", X"0100", X"0100", X"0100", X"00000208"),
        (X"0101", X"0101", X"0101", X"0101", X"0000020A"),
        (X"0069", X"0420", X"0077", X"0014", X"00001916"),
        (X"1945", X"1939", X"1969", X"1956", X"000032BC"),
        (X"0159", X"0281", X"0078", X"0234", X"00000107"),
        (X"1111", X"1111", X"1111", X"0002", X"0091CC40"));

BEGIN
    -- Instantiate the Unit Under Test (UUT)
    uut: entity work.algorithm
    PORT MAP (
        clk => clk,
        rst => rst,
        A => A,
        B => B,
        C => C,
        D => D,
        O => O
```

```
    );

    -- Clock process
    clkProcess : process
    begin
        clk <= '0';
        wait for clk_period/2;

        clk <= '1';
        wait for clk_period/2;
    end process;

    set_inputs: process
    begin
           -- Initial pause followed by syncing to the falling edge
        wait for wait_period;
        wait until falling_edge(clk);

        -- Initialise inputs
        rst <= '0';
        A <= x"0000";
        B <= x"0000";
        C <= x"0000";
        D <= x"0001";

        -- Initial reset
        wait for clk_period*2;
        rst <= '1';
        wait for clk_period*2;
        rst <= '0';

        -- test pattern input loop
        for i in test_vectors'range loop
            A <= test_vectors(i).A;
            B <= test_vectors(i).B;
            C <= test_vectors(i).C;
            D <= test_vectors(i).D;
            wait for clk_period;
        end loop;
        wait;
    end process;

    check_outputs: process
    begin
           -- Initial pause followed by syncing to the falling edge
        wait for wait_period;
        wait until falling_edge(clk);

           -- Pause to account for  the delay between input and output
        wait for clk_period*latency;

           -- Pause to account for the timing of pressing the reset button
        wait for clk_period*2;
        wait for clk_period*2;

        -- test pattern check loop
        for i in test_vectors'range loop
            assert (O = test_vectors(i).O)
            report "Test vector failed"
            severity error;
            wait for clk_period;
        end loop;
        wait;
    end process;
end;
```

1.1.2: Print out a screenshot of the behavioural simulation window, zoomed in to display, in readable format, all inputs, the output, and internal signals INT1-INT5 and INTO in unsigned decimal format. Include the console output (as a separate screenshot). All inputs; outputs; and internal signals INT1-INT5, and INTO in unsigned decimal format.

Screenshot



Console output:

```
launch_simulation
INFO: [Vivado 12-5682] Launching behavioral simulation in 'D:/files/uni/digitalengineering/lab2/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/behav/xsim'
INFO: [SIM-utils-51] Simulation object is 'sim_1'
INFO: [SIM-utils-54] Inspecting design source files for 'two_process_TB' in fileset 'sim_1'...
INFO: [USF-XSim-97] Finding global include files...
INFO: [USF-XSim-98] Fetching design files from 'sim_1'...
INFO: [USF-XSim-2] XSim::Compile design
INFO: [USF-XSim-61] Executing 'COMPILE and ANALYZE' step in 'D:/files/uni/digitalengineering/lab2/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/behav/xsim'
"xvhdl --incr --relax -prj two_process_TB_vhdl.prj"
INFO: [USF-XSim-69] 'compile' step finished in '2' seconds
INFO: [USF-XSim-3] XSim::Elaborate design
INFO: [USF-XSim-61] Executing 'ELABORATE' step in 'D:/files/uni/digitalengineering/lab2/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/behav/xsim'
Vivado Simulator 2017.4
Copyright 1986-1999, 2001-2016 Xilinx, Inc. All Rights Reserved.
Running: D:/programs/uni/xilinx/Vivado/Vivado/2017.4/bin/unwrapped/win64.o/xelab.exe -wto 14a433ed9f5e47a497118842e777e334 --incr --debug typical --relax --mt 2 -L xil_defaultlib -L secureip --snapshot two_process_TB_behav xil
Using 2 slave threads.
Starting static elaboration
Completed static elaboration
INFO: [XSIM 43-4323] No Change in HDL. Linking previously generated obj files to create kernel
INFO: [USF-XSim-69] 'elaborate' step finished in '2' seconds
INFO: [USF-XSim-4] XSim::Simulate design
INFO: [USF-XSim-61] Executing 'SIMULATE' step in 'D:/files/uni/digitalengineering/lab2/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/behav/xsim'
INFO: [USF-XSim-98] *** Running xsim
   with args "two_process_TB_behav -key {Behavioral:sim_1:Functional:two_process_TB} -tclbatch {two_process_TB.tcl} -view {D:/files/uni/digitalengineering/lab2/Performance_Check_On_Algorithm_Circuit/two_process_TB_behav.wcfg}
INFO: [USF-XSim-8] Loading simulator feature
Vivado Simulator 2017.4
Time resolution is 1 ps
open_wave_config D:/files/uni/digitalengineering/lab2/Performance_Check_On_Algorithm_Circuit/two_process_TB_behav.wcfg
source two_process_TB.tcl
# set curr_wave [current_wave_config]
# if { [string length $curr_wave] == 0 } {
#   if { [llength [get_objects]] > 0} {
#     add_wave /
#     set_property needs_save false [current_wave_config]
#   } else {
#     send_msg_id Add_Wave-1 WARNING "No top level signals found. Simulator will start without a wave window. If you want to open a wave window go to 'File->New Waveform Configuration' or type 'create_wave_config' in the TCL
#   }
# }
# run 5us
INFO: [USF-XSim-96] XSim completed. Design snapshot 'two_process_TB_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 5us
launch_simulation: Time (s): cpu = 00:00:03 ; elapsed = 00:00:06 . Memory (MB): peak = 999.715 ; gain = 0.000
```

1.1.3: Screenshot of the “Design Runs” tab; with all columns up to “DSP”. In your own words, and in relation to the lecture material, explain how the WNS is calculated. What is the maximum frequency at which the circuit can run, according to the tools?

Screenshot of the “Design Runs” tab; with all columns up to “DSP”

Name	Const...	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete!	29.183	0.000	0.610	0.000	0.000	0.108	0	886	96	0.00	0	0

In your own words, and in relation to the lecture material, explain how the WNS is calculated.

The WNS, or Worse Negative Slack, is calculated by taking the clock period and subtracting the time for a signal to propagate across the critical path. The critical path is the slowest path through the combinational logic part of the circuit.

This also means that worse negative slack is a measure of how much spare time is left over after the combinational part of the circuit before the next clock cycle.

What is the maximum frequency at which the circuit can run, according to the tools?

The negative slack is the amount of time left over after the critical path before the next clock cycle so we can subtract it from the clock period to get the theoretical shortest valid clock period.

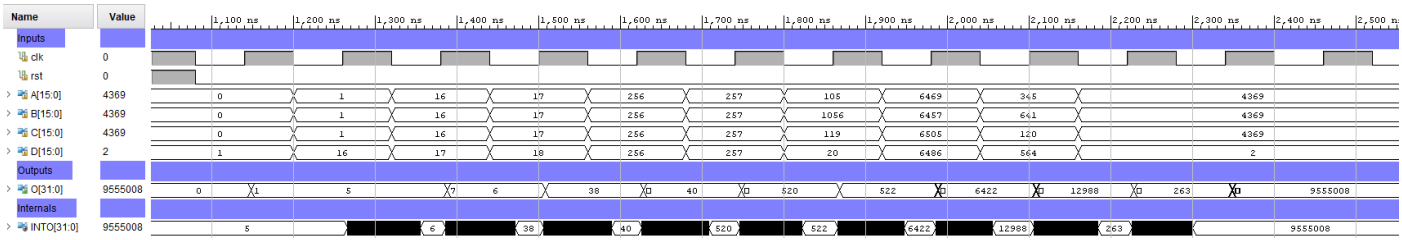
Clock Period (120ns) – Worse Negative Slack (29.183ns) = 90.817ns.

To get the frequency all we need to do is get the reciprocal of the frequency.

$$\text{Max frequency} = \frac{1}{\text{Min Clock Period}} = \frac{1}{90.817\text{ns}} \approx 11.011\text{MHz}$$

1.1.4: Screenshots of the post-implementation timing simulation window

All inputs and the outputs and INTO in unsigned decimal format.



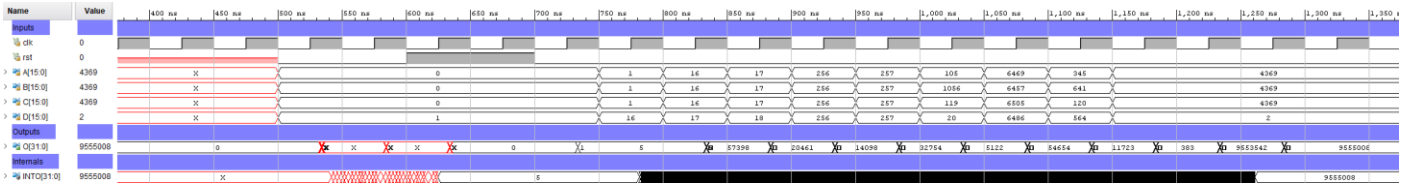
Console output:

```
launch_simulation -mode post-implementation -type timing
INFO: [Vivado 12-5682] Launching post-implementation timing simulation in 'H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/si
INFO: [SIM-utils-51] Simulation object is 'sim_1'
INFO: [SIM-utils-20] The target language is set to VHDL, it is not supported for simulation type 'timing', using Verilog instead.
INFO: [Netlist 29-17] Analyzing 289 Unisim elements for replacement
INFO: [Netlist 29-28] Unisim Transformation completed in 0 CPU seconds
WARNING: [Netlist 29-101] Netlist 'algorithm' is not ideal for floorplanning, since the cellview 'algorithm' contains a large number of primitives.  Please consider enablin
INFO: [Project 1-479] Netlist was created with Vivado 2017.4
INFO: [Project 1-570] Preparing netlist for logic optimization
Parsing XDC File [C:/Users/nt823/AppData/Local/Temp/.Xil_nt823/Vivado-5312-elclas003/dcp1/algorithm.xdc]
Finished Parsing XDC File [C:/Users/nt823/AppData/Local/Temp/.Xil_nt823/Vivado-5312-elclas003/dcp1/algorithm.xdc]
Reading XDEF placement.
Reading placer database...
Reading XDEF routing.
Read XDEF File: Time (s): cpu = 00:00:00 ; elapsed = 00:00:00.171 . Memory (MB): peak = 2032.695 ; gain = 0.000
Restored from archive | CPU: 0.000000 secs | Memory: 0.000000 MB |
Finished XDEF File Restore: Time (s): cpu = 00:00:00 ; elapsed = 00:00:00.171 . Memory (MB): peak = 2032.695 ; gain = 0.000
INFO: [Project 1-111] Unisim Transformation Summary:
No Unisim elements were transformed.
INFO: [SIM-utils-31] Writing simulation netlist file for design 'impl_1'...
INFO: [SIM-utils-32] write_verilog -mode timesim -nolib -sdf_anno true -force -file "H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circ
INFO: [SIM-utils-34] Writing SDF file...
INFO: [SIM-utils-35] write_sdf -mode timesim -process_corner slow -force -file "H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.s
write_sdf: Time (s): cpu = 00:00:01 ; elapsed = 00:00:15 . Memory (MB): peak = 2124.371 ; gain = 0.016
INFO: [SIM-utils-36] Netlist generated:H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/impl/timing/xsim/two_process_TB_
INFO: [SIM-utils-37] SDF generated:H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/impl/timing/xsim/two_process_TB_time_
INFO: [SIM-utils-54] Inspecting design source files for 'two_process_TB' in fileset 'sim_1'...
INFO: [USF-XSim-2] XSim::Compile design
INFO: [USF-XSim-61] Executing 'COMPILE and ANALYZE' step in 'H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/impl/timin
"xvlog --incr --relax -prj two_process_TB_vlog.prj"
INFO: [VRFC 10-2263] Analyzing Verilog file "H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/impl/timing/xsim/two_proce
INFO: [VRFC 10-311] analyzing module algorithm
INFO: [VRFC 10-311] analyzing module glbl
"xvhdl --incr --relax -prj two_process_TB_vhdl.prj"
INFO: [USF-XSim-69] 'compile' step finished in '4' seconds
INFO: [USF-XSim-3] XSim::Elaborate design
INFO: [USF-XSim-61] Executing 'ELABORATE' step in 'H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/impl/timing/xsim'
Vivado Simulator 2017.4
Copyright 1986-1999, 2001-2016 Xilinx, Inc. All Rights Reserved.
Running: C:/Xilinx/Vivado/2017.4/bin/unwrapped/win64.o/xelab.exe -wto 14a433ed9f5e47a497118842e777e334 --incr --debug typical --relax --mt 2 --maxdelay -L xil_defaultlib -L
Using 2 slave threads.
Starting static elaboration
Completed static elaboration
Starting simulation data flow analysis
Completed simulation data flow analysis
INFO: [XSIM 43-3451] SDF backannotation process started with SDF file "two_process_TB_time_impl.sdf", for root module "two_process_TB/uut".
INFO: [XSIM 43-3452] SDF backannotation was successful for SDF file "two_process_TB_time_impl.sdf", for root module "two_process_TB/uut".
Time Resolution for simulation is 1ps
```

```
Compiling package std.standard
Compiling package std.textio
Compiling package ieee.std_logic_1164
Compiling package ieee.numeric_std
Compiling package vl.vl_types
Compiling module xil_defaultlib.glbl
Compiling module simprims_ver.IBUF
Compiling module simprims_ver.FDRE_default
Compiling module simprims_ver.FDSE_default
Compiling module simprims_ver.x_lut2_mux4
Compiling module simprims_ver.LUT2
Compiling module simprims_ver.x_lut3_mux8
Compiling module simprims_ver.LUT3
Compiling module simprims_ver.LUT4
Compiling module simprims_ver.x_lut1_mux2
Compiling module simprims_ver.LUT1
Compiling module simprims_ver.LUT5
Compiling module simprims_ver.LUT6
Compiling module simprims_ver.OBUF
Compiling module simprims_ver.CARRY4
Compiling module simprims_ver.BUFG
Compiling module xil_defaultlib.algorithm
Compiling architecture behavior of entity xil_defaultlib.two_process_tb
Built simulation snapshot two_process_TB_time_impl
run_program: Time (s): cpu = 00:00:00 ; elapsed = 00:00:20 . Memory (MB): peak = 2124.371 ; gain = 0.000
INFO: [USF-XSim-69] 'elaborate' step finished in '20' seconds
INFO: [USF-XSim-4] XSim::Simulate design
INFO: [USF-XSim-61] Executing 'SIMULATE' step in 'H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_On_Algorithm_Circuit.sim/sim_1/impl/timing/xsim'
INFO: [USF-XSim-98] *** Running xsim
   with args "two_process_TB_time_impl -key {Post-Implementation:sim_1:Timing:two_process_TB} -tclbatch {two_process_TB.tcl} -view {H:/Downloads/Performance_Check_On_Algori
INFO: [USF-XSim-8] Loading simulator feature
Vivado Simulator 2017.4
Time resolution is 1 ps
open_wave_config H:/Downloads/Performance_Check_On_Algorithm_Circuit/two_process_TB_behav.wcfg
WARNING: Simulation object /two_process_TB/uut/INT1 was not found in the design.
WARNING: Simulation object /two_process_TB/uut/INT2 was not found in the design.
WARNING: Simulation object /two_process_TB/uut/INT3 was not found in the design.
WARNING: Simulation object /two_process_TB/uut/INT5 was not found in the design.
source two_process_TB.tcl
# set curr_wave [current_wave_config]
# if { [string length $curr_wave] == 0 } {
#   if { [llength [get_objects]] > 0} {
#     add_wave /
#     set_property needs_save false [current_wave_config]
#   } else {
#     send_msg_id Add_Wave-1 WARNING "No top level signals found. Simulator will start without a wave window. If you want to open a wave window go to 'File->New Waveform C
#   }
# }
# run 1000ns
INFO: [USF-XSim-96] XSim completed. Design snapshot 'two_process_TB_time_impl' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:12 ; elapsed = 00:00:48 . Memory (MB): peak = 2124.371 ; gain = 91.676
run 5 us
```

1.1.5: For 50ns clock period in testbench, screenshots of post-implementation timing simulation window

All inputs and the outputs and INTO in unsigned decimal format.



Console output.

```
Time resolution is 1 ps
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[3]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[11]/TChk142_1
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[5]/TChk142_17
Error: Test vector failed
Time: 850 ns  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Chec
Error: Test vector failed
Time: 900 ns  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Chec
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[2]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 145: Timing violation in scope /two_process_TB/uut/O_reg[6]/TChk145_17
Error: Test vector failed
Time: 950 ns  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Chec
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[3]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 145: Timing violation in scope /two_process_TB/uut/O_reg[1]/TChk145_17
Error: Test vector failed
Time: 1 us  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Check_
relaunch_sim: Time (s): cpu = 00:00:04 ; elapsed = 00:00:43 . Memory (MB): peak = 2124.371 ; gain = 0.000
run 5 us
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[5]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[4]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[11]/TChk142_1
Error: Test vector failed
Time: 1050 ns  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Che
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[12]/TChk142_1
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 145: Timing violation in scope /two_process_TB/uut/O_reg[0]/TChk145_17
Error: Test vector failed
Time: 1100 ns  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Che
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[3]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[7]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[6]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 145: Timing violation in scope /two_process_TB/uut/O_reg[1]/TChk145_17
Error: Test vector failed
Time: 1150 ns  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Che
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 145: Timing violation in scope /two_process_TB/uut/O_reg[10]/TChk145_1
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 145: Timing violation in scope /two_process_TB/uut/O_reg[6]/TChk145_17
Error: Test vector failed
Time: 1200 ns  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Che
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 145: Timing violation in scope /two_process_TB/uut/O_reg[1]/TChk145_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[7]/TChk142_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 145: Timing violation in scope /two_process_TB/uut/O_reg[8]/TChk145_17
WARNING: "C:\Xilinx\Vivado\2017.4\data/verilog/src/unisims/FDRE.v" Line 142: Timing violation in scope /two_process_TB/uut/O_reg[10]/TChk142_1
Error: Test vector failed
Time: 1250 ns  Iteration: 0  Process: /two_process_TB/check_outputs  File: H:/Downloads/Performance_Check_On_Algorithm_Circuit/Performance_Che
```

1.1.6: In your own words, and in relation to the lecture material, explain:

1) Why most internal signals are not available for display in the post-implementation simulation, whereas INTO is.

Synthesis rewrites the logic of the circuit to reduce the number of components. An example of this can be seen in RTL component statistics, where the last two adders in the circuit have been merged into a single three input adder.

```
---------------------------------------------------------------------------------
Start RTL Component Statistics
---------------------------------------------------------------------------------
Detailed RTL Component Info :
+---Adders :
      3 Input     32 Bit       Adders := 1
+---Registers :
                  32 Bit    Registers := 1
                  16 Bit    Registers := 4
---------------------------------------------------------------------------------
Finished RTL Component Statistics
---------------------------------------------------------------------------------
```

Whilst many of the internal signals have disappeared due to synthesis rearranging the circuit, INTO is an output rather than an intermediate signal, so it is required.

2) The behaviour of INTO and O in the post-implementation simulation of step 1.1.4, compared to the behavioural simulation of step 1.1.2

In the behavioural simulation for question 1.1.2, the simulation is treated as if all wires and combinational logic is instantaneous. This leads to the correct value reaching INTO as soon as the input register receives the new values.

In a real circuit, and in the timing simulation for question 1.1.4, the signal takes time to propagate down the wire. This leaves a time where the INTO has changed from its previous value, but the next value has not yet resolved itself as multiple changes to bits are coming in at different times. This leads to the blocks that look like solid colour. This

effect does not make its way to O because it must first travel through a register and the registers only update on a rising clock edge, by which point the INTO has already resolved.

3) The behaviour of INTO and O in the post-implementation simulation of step 1.1.5, compared to the post-implementation simulation of step 1.1.4

The timing simulation for question 1.1.5 was similar to the timing simulation 1.1.4, with the main difference being that INTO never resolved to a value. This is because the simulated clock period was 50ns but as established in question 1.1.3 the fastest clock speed the circuit can handle is around 90ns. What is happening in the timing simulation for step 1.1.5 is the signal is that the next clock cycle is starting, and the values are changing before INTO has resolved. For this reason, INTO never resolves.

This also means that the value the register picks up doesn't match the final value and is instead one value whilst INTO is still fluctuating between different numbers. This is why the tests fail and why O seems to have random or arbitrary numbers.

Lab 2 - Task B: Tool optimizations

1.2.1: Screenshots of the "Design Runs" tab with all columns up to "DSP"

For 80ns clock period in XDC file and in testbench

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete!	0.297	0.000	0.599	0.000	0.000	0.111	0	885	96	0.00	0	0

For 75ns clock period in XDC file and in testbench

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	Synthesis Out-of-date								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete!	0.232	0.000	1.030	0.000	0.000	0.111	0	886	96	0.00	0	0

For 70ns clock period in XDC file and in testbench

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete, Failed Timing!	-3.951	-110.914	1.216	0.000	0.000	0.112	0	895	96	0.00	0	0

Maximum frequency for each implementation of the circuit.

For 80ns clock period in XDC file and in testbench

$$\text{Max frequency} = \frac{1}{80ns - 0.297ns} \approx 12.547MHz$$

For 75ns clock period in XDC file and in testbench

$$\text{Max frequency} = \frac{1}{75ns - 0.232ns} \approx 13.375MHz$$

For 70ns clock period in XDC file and in testbench

$$\text{Max frequency} = \frac{1}{70ns - (-3.951)ns} \approx 15.140MHz$$

Why does the WNS change?

The worst negative slack is the clock period minus the time to resolve the critical path. As the clock period reduces so does the worst negative slack.

Why doesn't the WNS change by the amount the clock period is reduced by?

The compiler optimises the circuit for several things. One of the things it tries to do is reduce the critical path time. When the clock period is reduced so that the circuit has potential to fail if it is takes too long on the critical path the compiler is forced to prioritise reducing the critical path time.

What happens when the timing requirements aren't met?

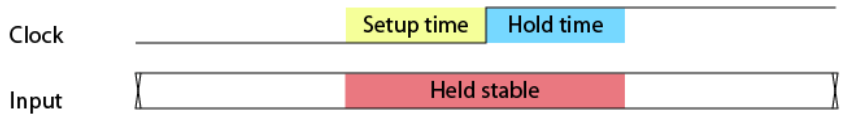
The circuit will not function properly as things will not resolve in time.

How does this relate to what was demonstrated in the timing simulations?

An example of the timing requirements being met was the timing simulation for question 1.1.4. Conversely, an example of them not being met was the timing simulation for question 1.1.5. As we saw in the latter, the path will not resolve so the output will not be correct.

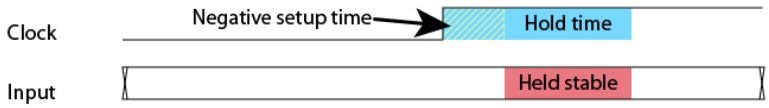
1.2.2: What a setup violation is and how does it relate to the critical path.

An input signal, i.e. "0011", going through sequential components in the critical path will be held stable at that value during Setup time and Hold time. Here is the visualization:



[1]

A setup violation is when setup time is negative and thus the input signal is allowed to change, when it wasn't expected to, after the onset of clock edge until Hold time shows up later on. Here is the visualization:



This unexpected change leads to different and fluctuating sequential outcome generated from the critical path which leads to wrong, short-term outputs where they are filled with block colour.

Lab 2 - Task C: Logic optimizations

1.3.1: Print out the modified VHDL code; screenshot of "Design Runs" tab with all columns up to "DSP", and comment on the comparison between the WNS value here and in the pre-modification case.

Print out the modified VHDL code

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

-- Entity description:
-- The entity implements, with no optimization, a sequence of operations:
--      O <= (A*3 + B*C)/D + C + 5
-- where A,B,C, and D are UNSIGNED vectors of parameterizable size

-- Note 1: There is no particular "meaning" to the equation - it is designed for
--  experimentation with logic optimization for performance
-- Note 2: There is no provision for overflow. Some input vectors can cause
--  overflow and the result will be incorrect.
-- Note 3: Inputs and outputs are registered (rising edge, synchronous reset).
--  This introduces a latency of 2 clock cycles between inputs and outputs.
-- Note 4: D is the divisor in one of the operations, so can never have value 0

entity algorithm is
    generic (data_size : integer := 16); -- defines the size of the data
    Port ( clk : in  STD_LOGIC;
           rst : in  STD_LOGIC;
           -- The four (parameterizable) data inputs
           A : in  STD_LOGIC_VECTOR (data_size-1 downto 0);
           B : in  STD_LOGIC_VECTOR (data_size-1 downto 0);
           C : in  STD_LOGIC_VECTOR (data_size-1 downto 0);
           D : in  STD_LOGIC_VECTOR (data_size-1 downto 0);
           -- Output = (A*3 + B*C)/D + C +5
           O : out  STD_LOGIC_VECTOR (data_size*2-1 downto 0)
           );
end algorithm;
```

```
architecture Behavioral of algorithm is

-- Registered inputs
signal INTA, INTB, INTC, INTD : UNSIGNED (data_size-1 downto 0);
-- Internal signals for intermediate operations (note the sizes)
signal INT1 : UNSIGNED (data_size+1 downto 0);   -- INTA + 3
signal INT2 : UNSIGNED (data_size*2-1 downto 0); -- INTB * INTC
signal INT3 : UNSIGNED (data_size*2-1 downto 0); -- INT1 + INT2
signal INT4 : UNSIGNED (data_size*2-1 downto 0); -- INT3 / INTD
signal INT5 : UNSIGNED (data_size*2-1 downto 0); -- INTC + INT4
signal INTO : UNSIGNED (data_size*2-1 downto 0); -- INT5 + 5

begin

-- Input registers (D-type, rising edge, synchronous reset)
input_regs: process (clk) is
begin
  if rising_edge(clk) then
    if rst = '1' then
      INTA <= (others => '0');
      INTB <= (others => '0');
      INTC <= (others => '0');
--    INTD <= (0 => '1', others => '0');  -- aggregate notation
      INTD <= to_unsigned(1,INTD'length); -- type conversion notation
    else
      INTA <= unsigned(A);
      INTB <= unsigned(B);
      INTC <= unsigned(C);
      INTD <= unsigned(D);
    end if;
  end if;
end process input_regs;

-- Mathematical operations on the data (combinational)
INT1 <= INTA * to_unsigned(3, 2);
INT2 <= INTB * INTC;
INT3 <= INT1 + INT2;
INT4 <= INT3 / INTD;
INT5 <= INTC + to_unsigned(5, INT5'length);
INTO <= INT5 + INT4;

-- Input registers (D-type, rising edge, synchronous reset)
output_regs: process (clk) is
begin
  if rising_edge(clk) then
    if rst = '1' then
      O <= (others => '0');
    else
      O <= std_logic_vector(INTO);
    end if;
  end if;
end process output_regs;
end Behavioral;
```

"Design Runs" tab with all columns up to "DSP" for 70ns clock period

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete, Failed Timing!	-3.951	-110.914	1.216	0.000	0.000	0.112	0	895	96	0.00	0	0

Comparison between the WNS value here and in the pre-modification case

"Design Runs" of post-modified code

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete, Failed Timing!	-3.951	-110.914	1.216	0.000	0.000	0.112	0	895	96	0.00	0	0

"Design Runs" of pre-modified code

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete, Failed Timing!	-3.951	-110.914	1.216	0.000	0.000	0.112	0	895	96	0.00	0	0

WNS in both cases are the same despite attempt to break up the critical path. The same applies when clock period is 80ns

“Design Runs” of post-modified code

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete!	0.297	0.000	0.599	0.000	0.000	0.111	0	885	96	0.00	0	0

“Design Runs” of pre-modified code

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								897	96	0.00	0	0
impl_1	constrs_1	route_design Complete!	0.297	0.000	0.599	0.000	0.000	0.111	0	885	96	0.00	0	0

1.3.2: Screenshot of the first few lines of the first Max Delay Path in the Post-Route Timing Report

```
Max Delay Paths
--------------------------------------------------------------------------------------
Slack (MET) :             0.297ns  (required time - arrival time)
  Source:                 INTB_reg[11]/C
                            (rising edge-triggered cell FDRE clocked by clk  {rise@0.000ns fall@40.000ns period=80.000ns})
  Destination:            O_reg[29]/D
                            (rising edge-triggered cell FDRE clocked by clk  {rise@0.000ns fall@40.000ns period=80.000ns})
  Path Group:             clk
  Path Type:              Setup (Max at Slow Process Corner)
  Requirement:            80.000ns  (clk rise@80.000ns - clk rise@0.000ns)
  Data Path Delay:        79.542ns  (logic 49.897ns (62.730%)  route 29.645ns (37.270%))
```