Kyle Keislar

ECE524/L FPGA/ASIC Design and Optimization Using VHDL Lab



Lab 1

Exploring Xilinx Vivado IDE and Zedboard

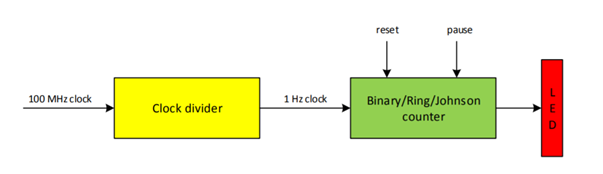
09/15/2020

## 

## Introduction & Problem Statements

This experiment explores the features of the Zedboard with onboard ZYNQ 7000 processor through design with the Xilnix Vivado IDE. This is accomplished through the design simple switchable counter which will select a counter type form the user via a switch input, and display the current output of the counter as a binary value on the onboard LEDs. To enable human observation, the counters input clock will be reduced from a master clock frequency of 100 MHz to a drastically slower rate of 1 Hz. Figure 1.0.A below shows the high level design of the counter.

Fig 1.0.A:High Level Design of Switchable-Counter



An implemented design of this counter was not realized using the Zedboard for this experiment.

## Procedure:

When simulating these designs, the Freq\_Div.vhd design module was used to create an input clock for each counter with the appropriate frequency. However, to demonstrate the functionality of each counter this was altered to be ½ of the 100 MHz design clock. The intended clock rate of the switchable counter is shown below.

Task 1: Design two counters: frequency divider counter that slows down the input clock frequency to the FPGA and 8 bit binary counter that lights up the LEDs. Make a new project and import all the files to your project. These counters can be seen in Appendix as items A.9 Freq\_Div.vhd and A.3 Bin\_Up.vhd respectively

Task 2: Synthesize VHDL code and make sure it is error free.

Task 3: Add location constraints to your design in order to use Zedboard development board pins. Connect the LEDs to the 8 bit counter outputs, and slide switches to reset, direction, and pause inputs to the 8 bit counter. The design constraint file can be seen in the Appendix as item A.10 Lab\_1\_Const.xdc.

Task 4: Modify your code to convert the counter to a full 8‐bit BCD counter, Johnson counter, Gray Code counter, Ring counter and 8 bit Fibonacci counter and verify the hardware. These counters can be seen in the Appendix as A.4 BCD\_Counter.vhd, A.5 Johnson\_Counter.vhd, A.6 Gray\_Counter.vhd, A.7 Ring\_Counter.vhd and A.8 Fib\_Counter.vhd respectively.

Task 5: There are 6 different counters you have designed. Use a multiplexer with select lines connected to the sliding switches to switch among all the counters. For instance, setting the switch to 001 selects binary counter, 010 selects BCD counter, and so on. The design top level can be seen in the Appendix as A.2 Lab\_1\_Design\_Top.vhd. The testbench can be seen in the Appendix as A.1 tb\_Lab\_1\_Design\_Top.vhd.

Task 6: Prove your design works by simulating your design.

## Results (Data):

## *Simulation Waveforms:*

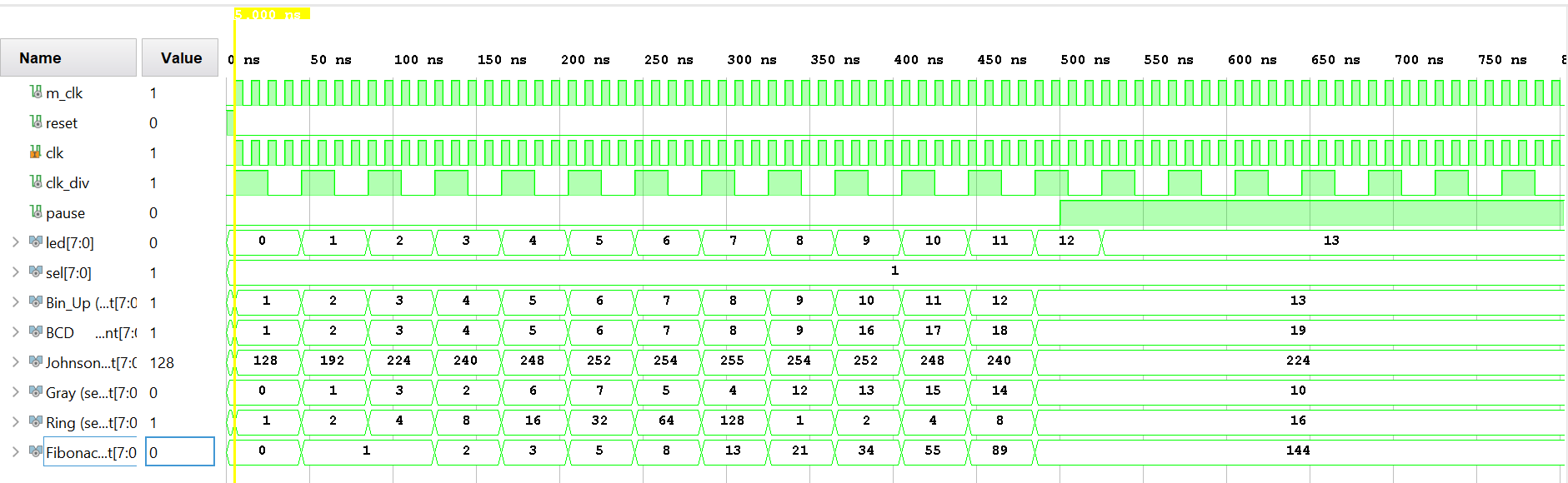
Fig 1.1: Binary Up Counter & Pause

Fig 1.2: BCD Counter & Unpause

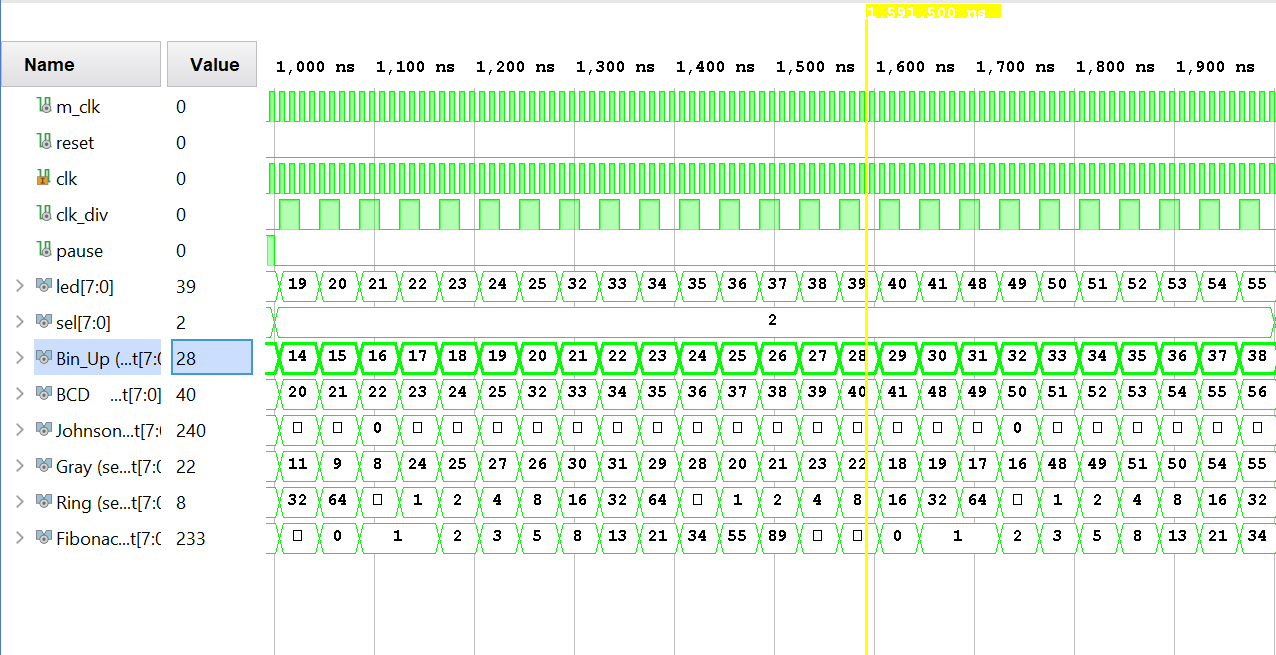


Fig 1.3: Johnson Counter

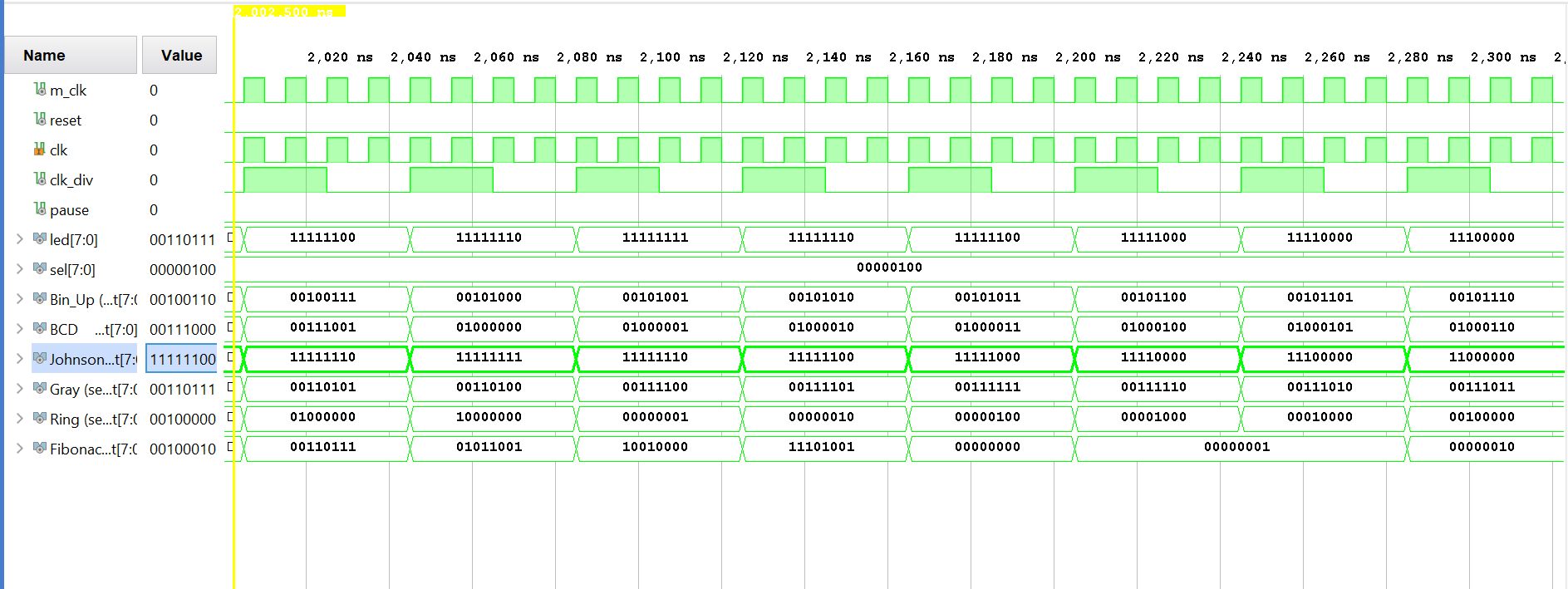


Fig 1.4: Gray Counter

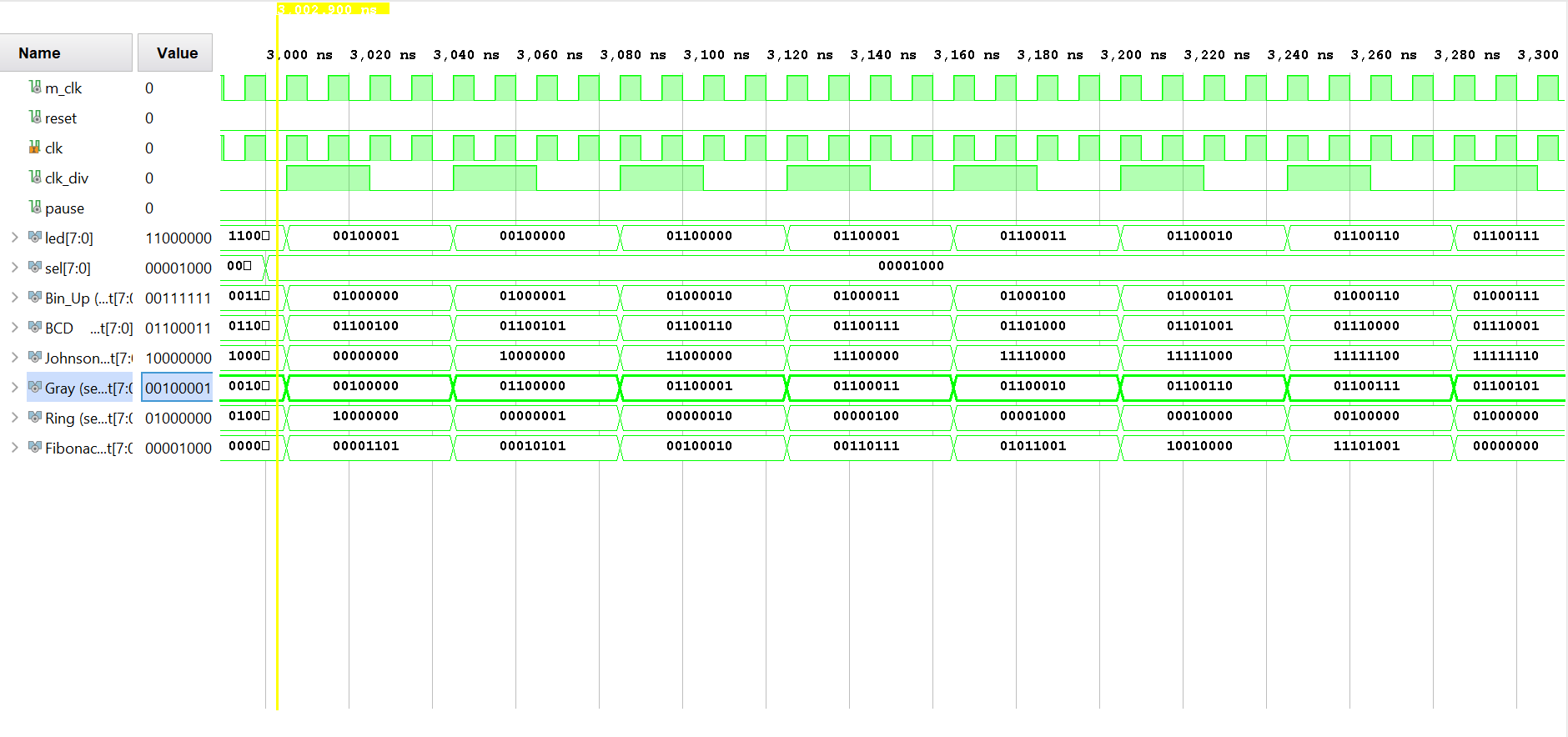


Fig 1.5: Ring Counter

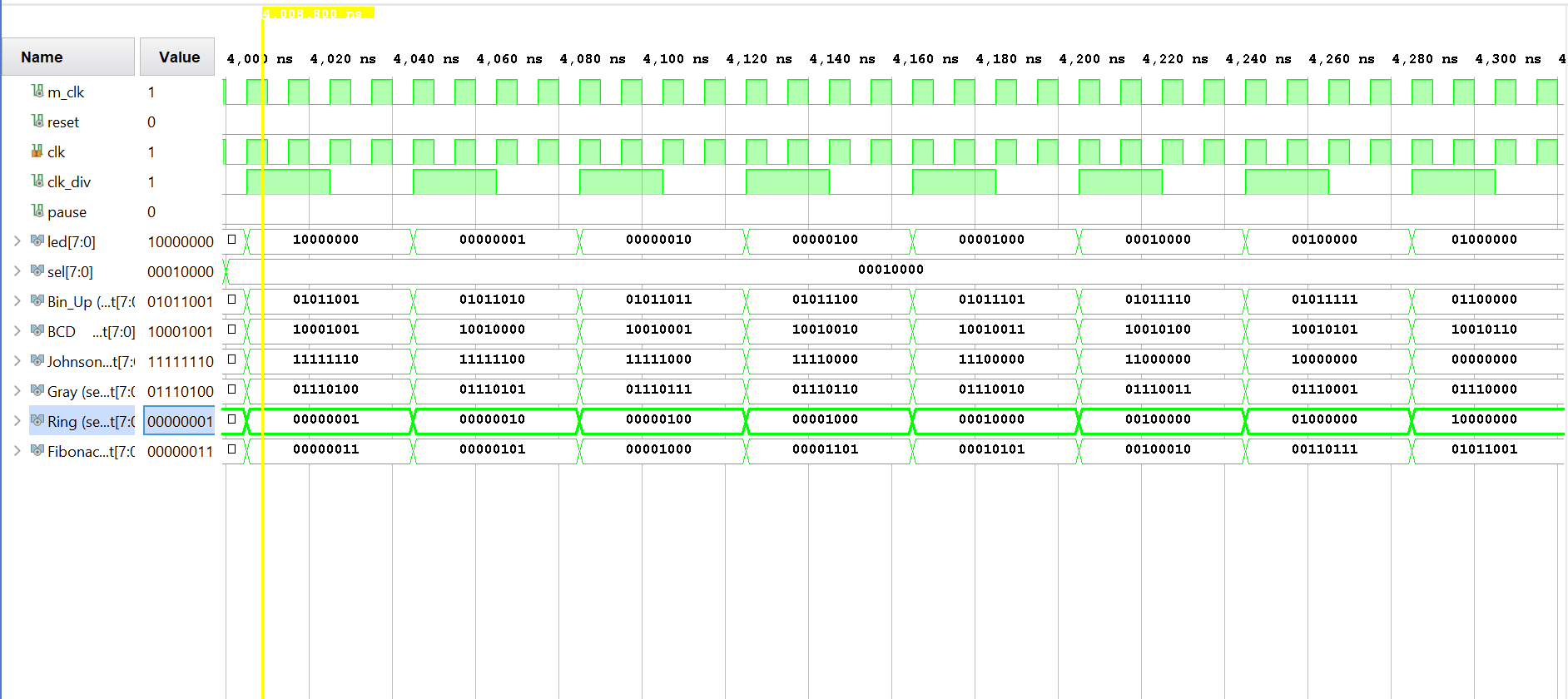


Fig 1.6: Fibonacci Counter

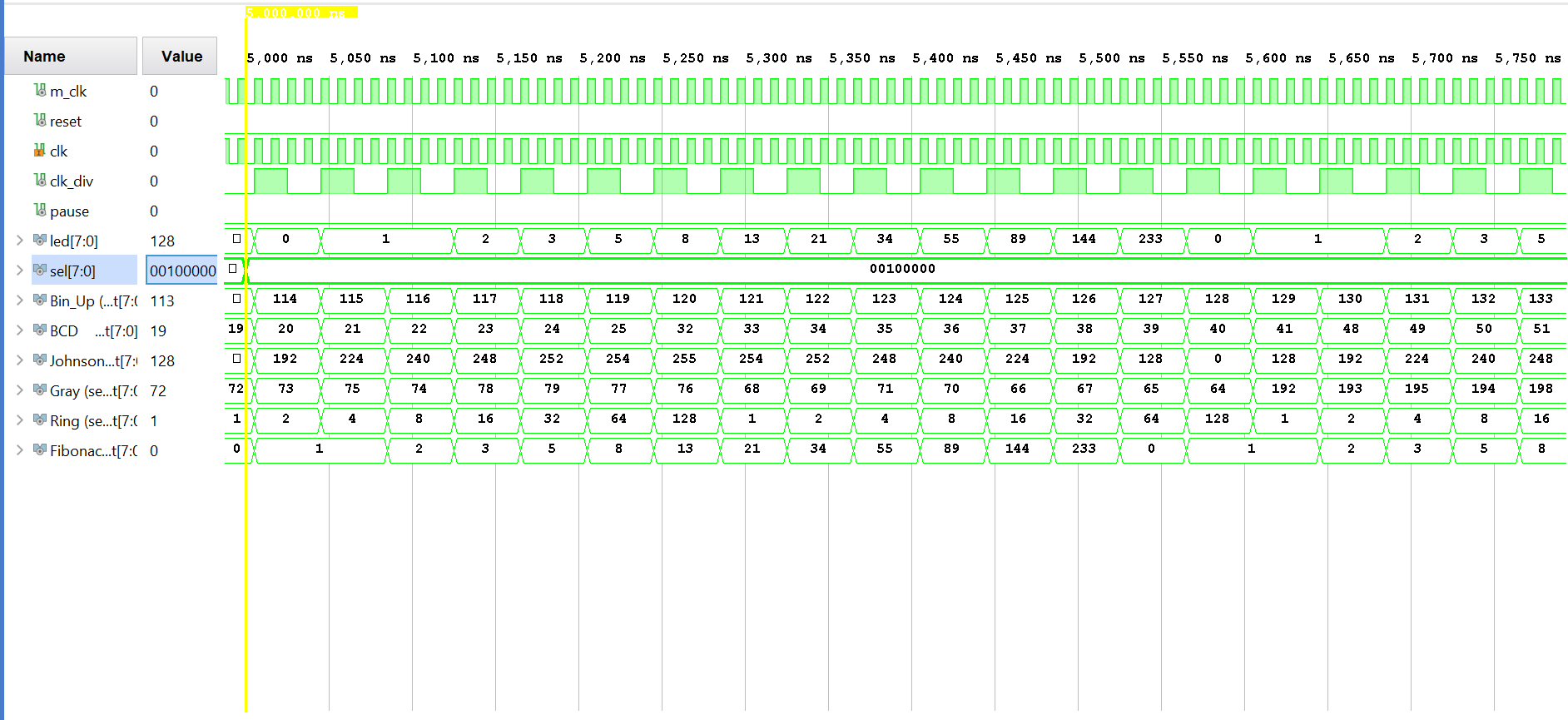


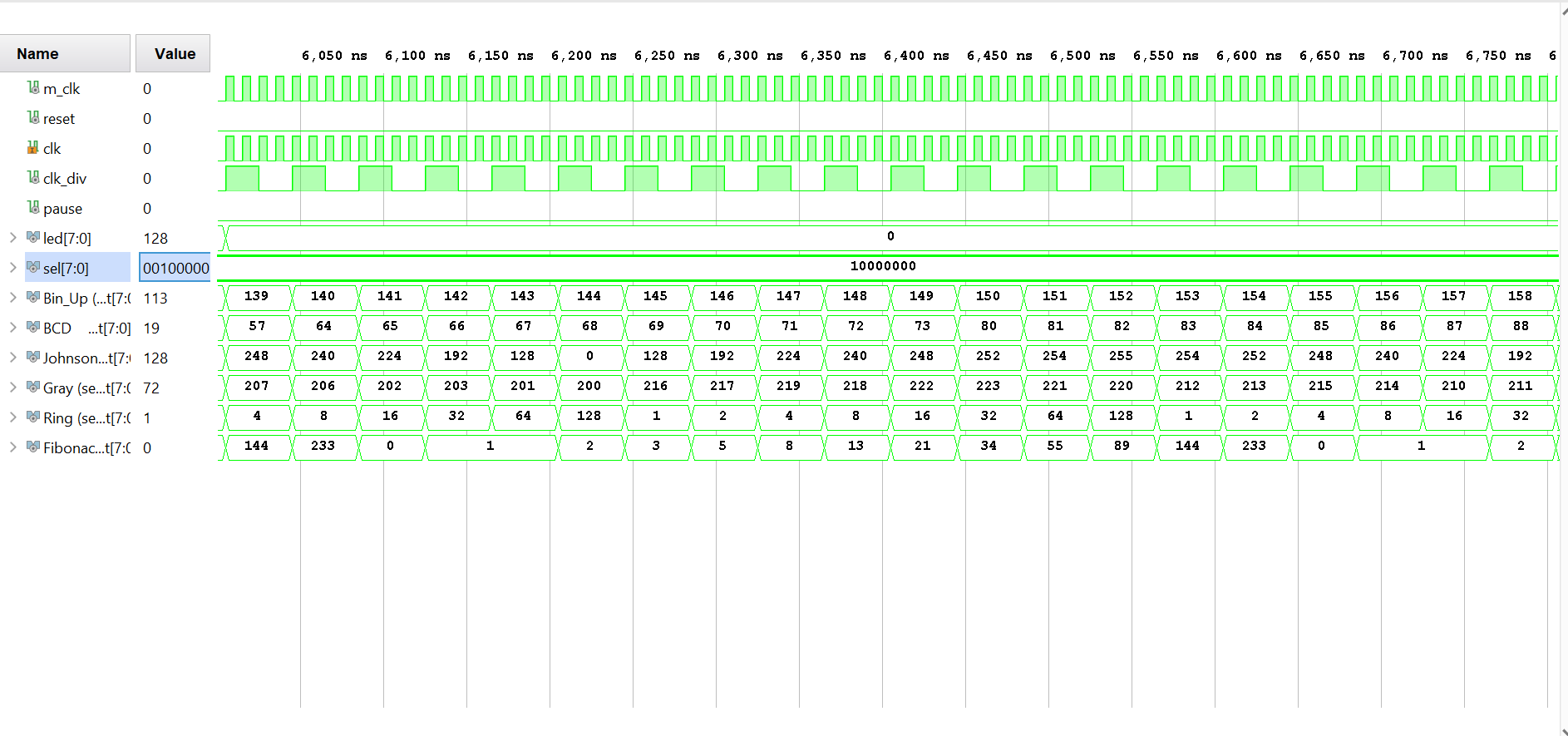
Fig 1.7: Other Select

Fig 1.8: Frequency Divider, Input 100 MHz Clock

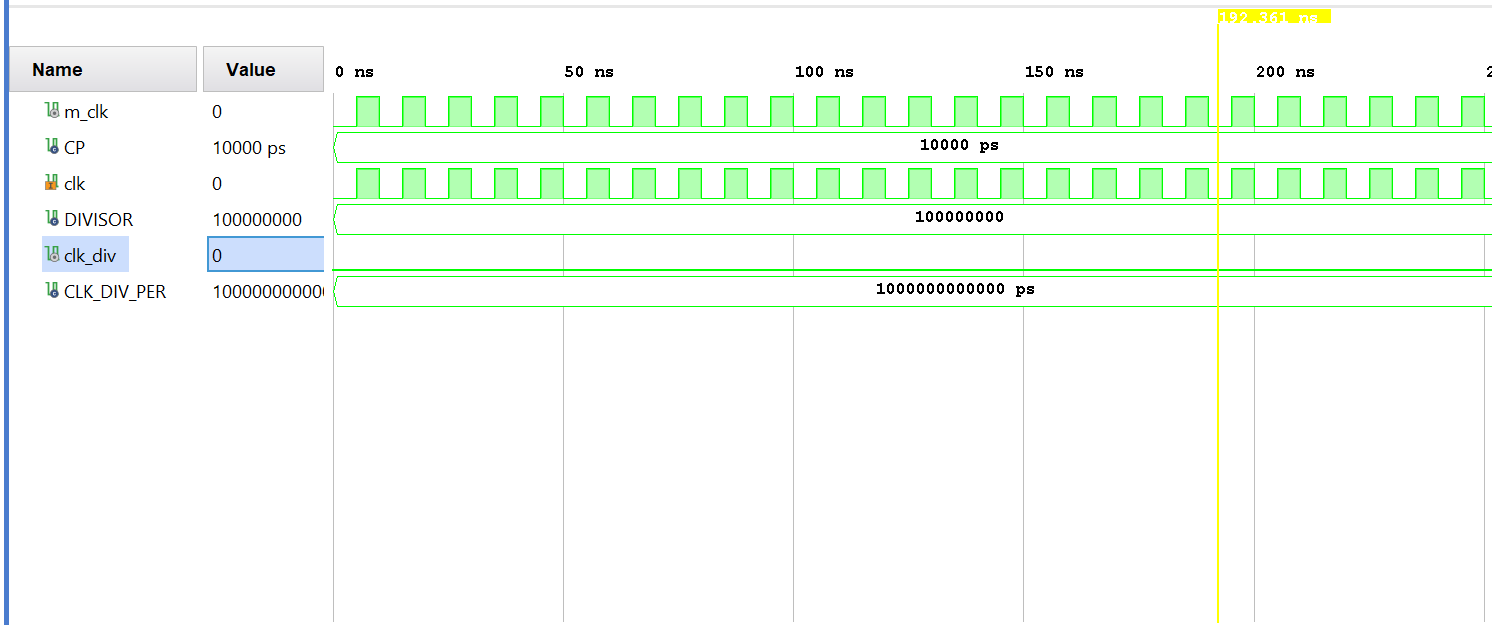
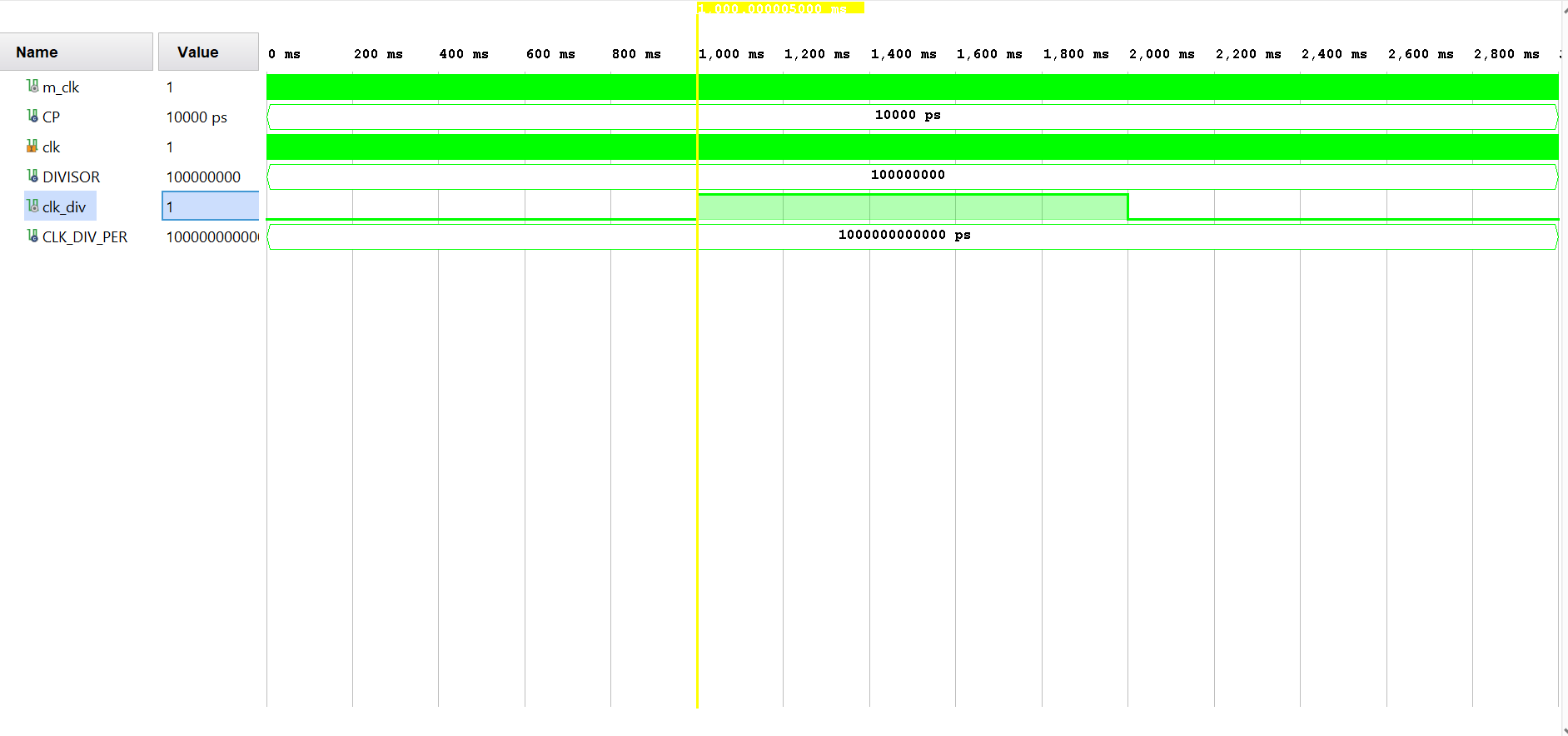


Fig 1.9: Frequency Divider, Output 1 Hz Clock



## Analysis:

The simulated design for this experiment successfully demonstrates the functionality of the proposed Switchable Counter as described in Fig 1.0.A. This is evidenced by the simulation waveforms above in figures 1.1-1.9. Each counter can be shown to be counting on each of the input clocks.

Fig 1.1 shows the functionality of the pause and reset inputs, as well as the functionality of the Binary Up counter. When reset is high, the counter’s output is set to 0, and the value displayed on the LEDs is also 0 (all LEDs off). In addition when the pause signal is asserted to high at ~500 ns, all of the counter’s outputs hold their current value. The selected counter at this time, the Binary Counter, holds its output value of 13.

Fig 1.2 shows the functionality of the Binary Coded Decimal (BCD) counter. When selected the output shows an 8-bit representation of two decimal digits. The counter has an initial value of 00 (seen in Fig 1.1) when reset and a maximum value of 99.

Fig 1.3 shows the functionality of the Johnson Counter. When selected the output shows an 8-bit Johnson Counter beginning and ending its sequence with the MSB. The counter has an initial value of “1000 0000” (seen in Fig 1.1) when reset and incrementally traverses each bit weight until its maximum value of “1111 1111” is reached. The counter then proceeds in reverse until the initial value is reached again.

Fig 1.4 shows the functionality of the Gray Counter. When selected the output shows an 8-bit Gray sequence which repeats indefinitely. The counter has an initial value of “0000 0000” when reset and proceeds through each value of the gray sequence until its maximum value of “1111 1111”. This value was not shown in the previous figures’ simulation time.

Fig 1.5 shows the functionality of the Ring counter. When selected the output shows an 8-bit ring counter in which only one of the bits is a ‘1’ and every other bit is a ‘0’. This counter proceeds in an increasing bit order from bit 0 to bit 7. The counter has an initial value of “0000 0001” when reset and a maximum value of “1000 0000”.

Fig 1.6 shows the functionality of the Fibonacci Counter. When seleted, the output displays the values of the Fibonacci sequence up to the maximum sequence value able to be represented with 8-bits ( ≤ 255). Each element of the Fibonacci sequence is equal to the sum of its previous two elements. The counter has an initial value of 0 when reset and a maximum value of 233.

Fig 1.7 shows the switchable counter’s functionality when an unused input is selected. For the multiplexing mechanism used in this design, a maximum of 6 inputs are required. Each bit of the sel line represents a single counter’s select line. This left multiple unused input values that, when selected, will display no count on the board LEDs. This can be seen in Fig 1.7 as an unused input “1000 000” is selected.

Finally, Fig 1.8 & 1.9 show the functionality of the Frequency Divider module used to calculate the clock input frequencies. As noted in the procedure, Figs 1.1-1.7 show the behavioral simulation of the design at an accelerated clock rate. This value is chosen by the user as a selected constant (DIVISOR, see tb\_Lab\_1\_Design\_Top.vhd) which subdivides the master clock (m\_clk) to create the counter input clock (clk\_div). For this experiment a value of 2 was selected for all simulated cases. This resulted in a clk\_div with a frequency equal to half of the m\_clk rate (50 MHz).

In these two figures, the desired clock rate of 1 Hz is shown to partially illustrate the timing functionality of the design. From the m\_clk frequency of 100 MHz, a divisor value of 100,000,000 is selected. These parameters are passed to the Freq\_Div.vhd module by the design’s top level. The output of Freq\_Div results in a counter input clock frequency of 1 Hz. Fig 1.8 shows the Freq\_Div’s inputs of clk and DIVISOR. Figure 1.9 shows the output clk\_div at a period of 1 second.

APPENDIX

Questions

#### 1) Explain what the constraint file is and how it helps you use the resources on FPGA development board? Explain how you add a constraint file to an existing design?

The constraint file is a set of software instructions that dictate what physical pins on the FPGA are intended to be connected or utilized by a design module. The file allows the user to enable or disable certain connections within the FGPA internal architecture and allows for more flexible designs.

To add a constraint file to a design you must add a constraint source to the project, and code which specific pins you wish to use. This can be done with the aid of the board developer provided master constraint file. For this experiment the constraint file below (Lab\_1\_Const.xdc) was constructed form the master constraint file provided by Xilnix.

#### 2) How do you access the FPGA resource information for any design? i.e. number of flip flops or LUTs?

The FPGA resource utilization for a design can be viewed in the Reports tab of the Vivado IDE. A detailed list of what flip-flops, LUTs and other resources are tabulated in the synthesis utilization reports that are automatically generated when a design is synthesized.

#### 3) How do you observe real time delays on simulation waveforms that is generated by Xilinx Vivado software? How do you select between different types of simulation in ISE software and what are the differences?

To observe real timing information from a design simulation, the design must be implemented. Then the Post Implementation Functional or Post Implementation Timing simulations show how the design will function in the design constraints in real time. To select among different types of simuation simply click the Run Simulation link beneath the Simulation tab in the Vivado IDE Flow Navigator. Then the user can select from among behavorial, post-synthesis functional/timing, or post-implementation functional/timing.

#### 4) What file is used to program the FPGA? Where is this file located?

The file used to program an FPGA is the bitstream file. This file is exported from an implemented design and read by the FPGA before the code is executed by the processor. The bitstream file is located at ~\Project\_Workspace\Project\_1\Project\_1.runs\impl\_1.

* 5) What is the input clock frequency to the FPGA?

100 MHz.

* 6) Assuming the LED switching frequency is to be 1 Hz, how can you make the FPGA input clock faster or slower than 1 Hz?

One way to change the input clock frequency of the FPGA is to use a frequency divider composed of one or more flip-flops with direct feedback. The clock could also be modified using Xilnix Custom IP

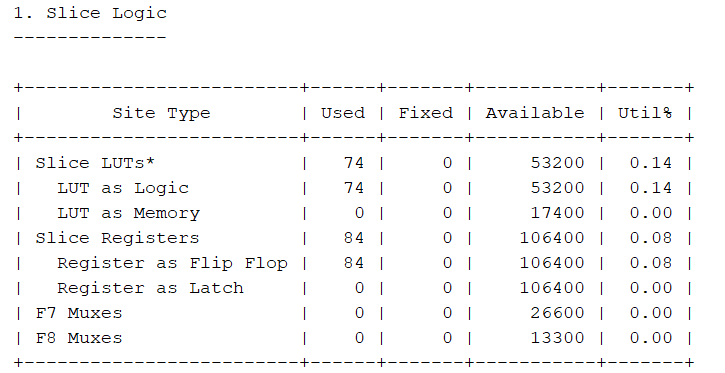
* 7) What is the contents of the constraint file to use development board resources such as LEDs, sliding switches, clock, and push buttons?

The constraint file contains instructions about what onboard ports are connected to which onboard elements. For example the line:

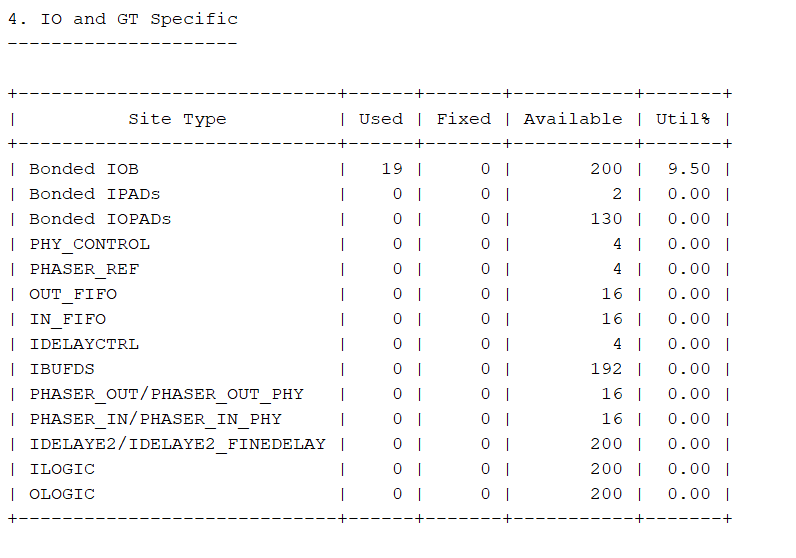
set\_property PACKAGE\_PIN T22 [get\_ports {LD0}]; # "LD0"

sets the pin package T22 to be connected to the ports of LED\_0.

**Utilization report**



**I/O report**



1. *Tb\_Lab\_1\_Design.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: tb\_Lab\_1\_Design\_top.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design\_top.vhd

--

-- Date Last Revised: 09/14/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**use** IEEE**.**NUMERIC\_STD**.ALL;**

**use** IEEE**.**math\_real**.**"ceil"**;**

**use** IEEE**.**math\_real**.**"log2"**;**

**entity** tb\_Lab\_1\_Design **is**

-- Port ( );

**end** tb\_Lab\_1\_Design**;**

**architecture** Behavioral **of** tb\_Lab\_1\_Design **is**

**component** Lab\_1\_Design\_top **is**

**generic** **(** divisor**:** integer **:=** 10**;**

div\_width**:** integer **:=** 1**);**

**Port** **(** clk**:** **in** std\_logic**;**

reset**:** **in** std\_logic**;**

pause**:** **in** std\_logic**;**

sel**:** **in** std\_logic\_vector**(**7 **downto** 0**);** --select line form switches

led**:** **out** std\_logic\_vector**(**7 **downto** 0**));** --binary count value to display from selected counter

**end** **component;**

--Constant Definitions

**CONSTANT** CP**:** TIME **:=** 10ns**;**--clock period

**CONSTANT** DIVISOR**:** INTEGER **:=** 2**;** --clock divisor, integer value that the input signal m\_clk will be divided by

**CONSTANT** DIV\_WIDTH**:** INTEGER **:=** integer**(**ceil**(**log2**(**real**(**DIVISOR**))));** --# of bits needed to hold DIVISOR value

**CONSTANT** CLK\_DIV\_PER**:** TIME **:=** CP**\***DIVISOR**;**--amount of time needed to wait for output of next count

**CONSTANT** SEL\_WAIT\_TIME**:** TIME **:=** CLK\_DIV\_PER**\***50**;**--amount of time select is to remain unchanged (used for observation)

--Signal Definitions

**SIGNAL** PULSE**:** TIME**:=**CP**\***0.5**;**--pulse width

**signal** m\_clk**,**reset**,**pause**:** std\_logic **:=** '0'**;** --master clock

**signal** led**,** sel**,** sel\_in**:** std\_logic\_vector**(**7 **downto** 0**)** **:=** **(Others** **=>**'0'**);**-- led value to display, select line, input selected from switches by user

**begin**

uut**:** Lab\_1\_Design\_top

**Generic** **map(** divisor **=>** DIVISOR**,**

div\_width **=>** DIV\_WIDTH**)**

**Port** **map** **(**clk **=>** m\_clk**,**

reset**=>**reset**,**

pause **=>** pause**,**

sel**=>**sel**,**

led **=>** led**);**

--Clock Processs

Rst**:process** --resets counters to 0

**begin**

reset **<=** '1'**;**

**wait** **for** pulse**;**

reset **<=** '0'**;**

**wait** **for** pulse**;**

**wait;**

**end** **process;**

clock**:process**

**begin**

m\_clk **<=** '0'**;**

**wait** **for** PULSE **;**

m\_clk **<=** '1'**;**

**wait** **for** PULSE**;**

**end** **process;**

test**:process**

**begin**

**if(**reset**=**'1'**)** **then**

sel**<=**"00000000"**;**

pause **<=** '0'**;**

**else**

sel**<=**"00000001"**;**

**wait** **for** SEL\_WAIT\_TIME**\***0.5**;**

pause **<=** '1'**;**

**wait** **for** SEL\_WAIT\_TIME**\***0.5**;**

pause **<=**'0'**;**

sel**<=**"00000010"**;**

**wait** **for** SEL\_WAIT\_TIME**;**

sel**<=**"00000100"**;**

**wait** **for** SEL\_WAIT\_TIME**;**

sel**<=**"00001000"**;**

**wait** **for** SEL\_WAIT\_TIME**;**

sel**<=**"00010000"**;**

**wait** **for** SEL\_WAIT\_TIME**;**

sel**<=**"00100000"**;**

**wait** **for** SEL\_WAIT\_TIME**;**

sel**<=**"10000000"**;**

**wait** **for** SEL\_WAIT\_TIME**;**

**wait;**

**end** **if;**

**end** **process;**

**end** Behavioral**;**

--Unused User Processes

--read switches

--read\_led: process(sel\_in,m\_clk) --Provides synchronous select change

--variable temp: std\_logic\_vector(7 downto 0):= (Others =>'0');

--begin

-- if(reset='0') then

-- if(m\_clk'event AND m\_clk='1') then

-- temp:= sel\_in;

-- end if;

-- end if;

--sel <= temp;

--end process;

1. *Lab\_1\_Design\_top.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: Lab\_1\_Design\_top.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design.vhd

--

-- Date Last Revised: 09/14/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**entity** Lab\_1\_Design\_top **is**

**generic** **(** divisor**:** integer **:=** 10**;**

div\_width**:** integer **:=** 1**);**

**Port** **(** clk**:** **in** std\_logic**;**

reset**:** **in** std\_logic**;**

pause**:** **in** std\_logic**;**

sel**:** **in** std\_logic\_vector**(**7 **downto** 0**);**

led**:** **out** std\_logic\_vector**(**7 **downto** 0**));**

**end** Lab\_1\_Design\_top**;**

**architecture** Behavioral **of** Lab\_1\_Design\_top **is**

--Component Definitions

**component** Freq\_Div **is**

**generic(** divisor**:** integer **:=** 10**;**

div\_width**:** integer **:=** 1**);**

**Port** **(** m\_clk **:** **in** STD\_LOGIC**;**

clk **:** **out** STD\_LOGIC**);**

**end** **component;**

**component** Bin\_Up **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** **component;**

**component** BCD\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** **component;**

**component** Johnson\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** **component;**

**component** Gray\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** **component;**

**component** Ring\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** **component;**

**component** Fib\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** **component;**

--Signal Definitions

**Signal** sel\_0\_cnt**,** sel\_1\_cnt**,**sel\_2\_cnt**,**sel\_3\_cnt**,**sel\_4\_cnt**,**sel\_5\_cnt **:** std\_logic\_vector**(**7 **downto** 0**)** **:=** **(Others** **=>**'0'**);** --counter outputs

**signal** clk\_div**:** std\_logic **:=** '0'**;**--divided clock value

**signal** temp\_sel **:** std\_logic\_vector **(**7 **downto** 0**)** **:=** **(Others** **=>**'0'**);** --used to read from select and hold last clocked value

**begin**

--Component instantiations

f\_div**:**Freq\_Div

**Generic** **map(** divisor **=>** DIVISOR**,**

div\_width **=>** DIV\_WIDTH**)**

**Port** **map** **(**m\_clk **=>** clk**,**

clk**=>**clk\_div**);**

bin\_up\_c**:** Bin\_Up --Select 0

**Port** **map** **(**clk **=>** clk\_div**,**

rst**=>**reset**,**

pause **=>** pause**,**

count **=>** sel\_0\_cnt**);**

BCD\_c**:** BCD\_Counter --Select 1

**Port** **map** **(**clk **=>** clk\_div**,**

rst**=>**reset**,**

pause **=>** pause**,**

count **=>** sel\_1\_cnt**);**

John\_c**:** Johnson\_Counter --Select 2

**Port** **map** **(**clk **=>** clk\_div**,**

rst**=>**reset**,**

pause **=>** pause**,**

count **=>** sel\_2\_cnt**);**

Gray\_c**:** Gray\_Counter --Select 3

**Port** **map** **(**clk **=>** clk\_div**,**

rst**=>**reset**,**

pause **=>** pause**,**

count **=>** sel\_3\_cnt**);**

Ring\_c**:** Ring\_Counter --Select 4

**Port** **map** **(**clk **=>** clk\_div**,**

rst**=>**reset**,**

pause **=>** pause**,**

count **=>** sel\_4\_cnt**);**

Fib\_c**:** Fib\_Counter --Select 5

**Port** **map** **(**clk **=>** clk\_div**,**

rst**=>**reset**,**

pause **=>** pause**,**

count **=>** sel\_5\_cnt**);**

--select 6,7 == LEDS off

--MUX processes

MUX**:** **process(**sel**,**clk\_div**)**

**begin**

**if(**reset**=**'1'**)** **then**

led **<=** **(Others** **=>**'0'**);**

**else**

**if(**clk\_div'**event** **AND** clk\_div**=**'1'**)** **then**

**case** temp\_sel **is**

**when**"00000001" **=>** led **<=** sel\_0\_cnt**;**--Bin up

**when**"00000010" **=>** led **<=** sel\_1\_cnt**;**--BCD

**when**"00000100" **=>** led **<=** sel\_2\_cnt**;**--Johnson

**when**"00001000" **=>** led **<=** sel\_3\_cnt**;**--Gray

**when**"00010000" **=>** led **<=** sel\_4\_cnt**;**--Ring

**when**"00100000" **=>** led **<=** sel\_5\_cnt**;**--Fib Sequence

**when** **others** **=>** led **<=** **(Others** **=>**'0'**);**

**end** **case;**

**end** **if;**

**end** **if;**

**end** **process;**

temp\_sel **<=** sel**;** --get passed sel value from User

**end** Behavioral**;**

1. *Bin\_Up.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: Bin\_Up.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design\_top.vhd

--

-- Date Last Revised: 09/12/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**use** IEEE**.**NUMERIC\_STD**.ALL;**

**entity** Bin\_Up **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** Bin\_Up**;**

**architecture** Behavioral **of** Bin\_Up **is**

**signal** cnt**:** std\_logic\_vector**(**7 **downto** 0**):=** **(Others** **=>** '0'**);**

**begin**

**process(**rst**,**clk**,**pause**)**

**begin**

**if(**rst**=**'1'**)** **then**

cnt **<=** **(Others=>** '0'**);**

**else**

**if** pause**=**'1' **then**

cnt **<=** cnt**;**

**else**

**if** clk'**event** **AND** clk**=**'1' **then**

cnt **<=** std\_logic\_vector**(**unsigned**(**cnt**)** **+** 1**);**

**end** **if;**

**end** **if;**

**end** **if;**

**end** **process;**

count **<=** cnt**;**

**end** Behavioral**;**

1. *BCD\_Counter.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: BCD\_Counter.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design\_top.vhd

--

-- Date Last Revised: 09/14/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**use** IEEE**.**NUMERIC\_STD**.ALL;**

--apply directional inputs

**entity** BCD\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** BCD\_Counter**;**

**architecture** Behavioral **of** BCD\_Counter **is**

--min value == 0

--max value == 99 as '9' '9' or 1001 1001

**signal** first\_cnt**:** std\_logic\_vector**(**3 **downto** 0**):=** **(Others** **=>** '0'**);**

**signal** second\_cnt**:** std\_logic\_vector**(**3 **downto** 0**):=** **(Others** **=>** '0'**);**

**begin**

**process(**rst**,**clk**,**pause**)**

**begin**

**if(**rst**=**'1'**)** **then**

first\_cnt **<=** **(Others=>** '0'**);**

second\_cnt **<=** **(Others=>** '0'**);**

**else**

**if** pause**=**'1' **then**

first\_cnt **<=** first\_cnt**;**

second\_cnt **<=** second\_cnt**;**

**else**

**if** clk'**event** **AND** clk**=**'1' **then**

first\_cnt **<=** std\_logic\_vector**(**unsigned**(**first\_cnt**)** **+** 1**);**

**if** first\_cnt **=** "1001" **then**

second\_cnt **<=** std\_logic\_vector**(**unsigned**(**second\_cnt**)** **+** 1**);**

first\_cnt **<=** **(Others** **=>** '0'**);**

**if** second\_cnt **=** "1001" **then**

second\_cnt **<=** **(Others** **=>** '0'**);**

first\_cnt **<=** **(Others** **=>**'0'**);**

**end** **if;**

**end** **if;**

**end** **if;**

**end** **if;**

**end** **if;**

**end** **process;**

count **<=** second\_cnt **&** first\_cnt**;**

**end** Behavioral**;**

1. *Johnson\_Counter.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: Johnson\_Counter.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design\_top.vhd

--

-- Date Last Revised: 09/12/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**use** IEEE**.**NUMERIC\_STD**.ALL;**

**entity** Johnson\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** Johnson\_Counter**;**

**architecture** Behavioral **of** Johnson\_Counter **is**

**signal** bit\_chg**:** std\_logic\_vector**(**3 **downto** 0**)** **:=** **(Others** **=>**'0'**);** --tracks # of bit state changes during count

**signal** cnt **:** std\_logic\_vector**(**7 **downto** 0**)** **:=** **(Others** **=>**'0'**);**

**begin**

**process(**rst**,**clk**,**pause**)**

**begin**

**if(**rst**=**'1'**)** **then**

cnt **<=** **(Others=>** '0'**);**

bit\_chg **<=** **(Others** **=>** '0'**);**

**else**

**if** pause**=**'1' **then**

cnt **<=** cnt**;**

bit\_chg **<=** bit\_chg**;**

**else**

**if** clk'**event** **AND** clk**=**'1' **then**

**case** bit\_chg **is**

**when** "0000" **=>** cnt **<=** "10000000"**;**

**when** "0001" **=>** cnt **<=** "11000000"**;**

**when** "0010" **=>** cnt **<=** "11100000"**;**

**when** "0011" **=>** cnt **<=** "11110000"**;**

**when** "0100" **=>** cnt **<=** "11111000"**;**

**when** "0101" **=>** cnt **<=** "11111100"**;**

**when** "0110" **=>** cnt **<=** "11111110"**;**

**when** "0111" **=>** cnt **<=** "11111111"**;**

**when** "1000" **=>** cnt **<=** "11111110"**;**

**when** "1001" **=>** cnt **<=** "11111100"**;**

**when** "1010" **=>** cnt **<=** "11111000"**;**

**when** "1011" **=>** cnt **<=** "11110000"**;**

**when** "1100" **=>** cnt **<=** "11100000"**;**

**when** "1101" **=>** cnt **<=** "11000000"**;**

**when** "1110" **=>** cnt **<=** "10000000"**;**

**when** **others** **=>** cnt **<=** "00000000"**;**

**end** **case;**

bit\_chg **<=** std\_logic\_vector**(**unsigned**(**bit\_chg**)** **+** 1**);** -- increment count

**end** **if;**

**end** **if;**

**end** **if;**

**end** **process;**

count **<=** cnt**;**

**end** Behavioral**;**

1. *Gray\_Counter.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: Gray\_Counter.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design\_top.vhd

--

-- Date Last Revised: 09/12/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**use** IEEE**.**NUMERIC\_STD**.ALL;**

**entity** Gray\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** Gray\_Counter**;**

**architecture** Behavioral **of** Gray\_Counter **is**

**signal** bit\_chg**:** std\_logic\_vector**(**3 **downto** 0**)** **:=** **(Others** **=>**'0'**);** --tracks # of bit state changes during count

**signal** b\_cnt**,** g\_cnt **:** std\_logic\_vector**(**7 **downto** 0**)** **:=** **(Others** **=>**'0'**);** --Binary Count, Gray Count

**begin**

**process(**rst**,**clk**,**pause**)**

**begin**

**if(**rst**=**'1'**)** **then**

b\_cnt **<=** **(Others=>** '0'**);**

g\_cnt **<=** **(Others=>** '0'**);**

bit\_chg **<=** **(Others** **=>** '0'**);**

**else**

**if** pause**=**'1' **then**

b\_cnt **<=** b\_cnt**;**

g\_cnt **<=** g\_cnt**;**

bit\_chg **<=** bit\_chg**;**

**else**

**if** clk'**event** **AND** clk**=**'1' **then**

g\_cnt **(**7**)** **<=** b\_cnt**(**7**);** --MSB is the same

**for** i **in** 0 **to** 6 **loop**

--gray[i] = (binary[i+1] ^ binary[i])

g\_cnt**(**i**)** **<=** b\_cnt**(**i**+**1**)** **xor** b\_cnt**(**i**);**

**end** **loop;**

b\_cnt **<=** std\_logic\_vector**(**unsigned**(**b\_cnt**)** **+** 1**);**

**end** **if;**

**end** **if;**

**end** **if;**

**end** **process;**

count **<=** g\_cnt**;**

**end** Behavioral**;**

1. *Ring\_Counter.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: Ring\_Counter.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design\_top.vhd

--

-- Date Last Revised: 09/12/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**use** IEEE**.**NUMERIC\_STD**.ALL;**

--apply directional inputs

**entity** Ring\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** Ring\_Counter**;**

**architecture** Behavioral **of** Ring\_Counter **is**

**signal** bit\_cnt**:** std\_logic\_vector**(**2 **downto** 0**)** **:=** **(Others** **=>**'0'**);** --tracks # of bit state changes during count

**signal** cnt **:** std\_logic\_vector**(**7 **downto** 0**)** **:=** **(Others** **=>**'0'**);**

**begin**

**process(**rst**,**clk**,**pause**)**

**begin**

**if(**rst**=**'1'**)** **then**

cnt **<=** **(Others=>** '0'**);**

bit\_cnt**<=** **(Others** **=>** '0'**);**

**else**

**if** pause**=**'1' **then**

cnt **<=** cnt**;**

bit\_cnt **<=** bit\_cnt**;**

**else**

**if** clk'**event** **AND** clk**=**'1' **then**

**case** bit\_cnt **is**

**when** "000" **=>** cnt **<=** "00000001"**;**

**when** "001" **=>** cnt **<=** "00000010"**;**

**when** "010" **=>** cnt **<=** "00000100"**;**

**when** "011" **=>** cnt **<=** "00001000"**;**

**when** "100" **=>** cnt **<=** "00010000"**;**

**when** "101" **=>** cnt **<=** "00100000"**;**

**when** "110" **=>** cnt **<=** "01000000"**;**

**when** "111" **=>** cnt **<=** "10000000"**;**

**when** **others** **=>** cnt **<=** **(Others** **=>**'0'**);**

**end** **case;**

bit\_cnt **<=** std\_logic\_vector**(**unsigned**(**bit\_cnt**)** **+** 1**);** -- increment count

**end** **if;**

**end** **if;**

**end** **if;**

**end** **process;**

count **<=** cnt**;**

**end** Behavioral**;**

1. *Fib\_Counter.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: Fib\_Counter.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design\_top.vhd

--

-- Date Last Revised: 09/12/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**use** IEEE**.**NUMERIC\_STD**.ALL;**

--apply directional inputs

**entity** Fib\_Counter **is**

**Port** **(** count **:** **out** STD\_LOGIC\_VECTOR **(**0 **to** 7**);**

clk **:** **in** STD\_LOGIC**;**

rst **:** **in** STD\_LOGIC**;**

pause **:** **in** STD\_LOGIC**);**

**end** Fib\_Counter**;**

**architecture** Behavioral **of** Fib\_Counter **is**

**signal** cnt**:** std\_logic\_vector**(**7 **downto** 0**)** **:=** **(Others** **=>**'0'**);**

**signal** fib\_prev**,** fib\_cur**,** fib\_nxt**:** std\_logic\_vector**(**7 **downto** 0**)** **:=** "00000001"**;**

**signal** fib\_count**:** std\_logic\_vector**(**3 **downto** 0**):=** **(Others** **=>**'0'**);**

**begin**

**process(**rst**,**clk**,**pause**)**

**begin**

**if(**rst**=**'1'**)** **then** --set first element

cnt **<=** **(Others** **=>** '0'**);**

fib\_count**<=** "1101"**;**

**else**

**if** pause**=**'1' **then**

cnt **<=** cnt**;**

**else**

--Fib sequence bounded by 0,255

-- 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233

--13 elements

**if** clk'**event** **AND** clk**=**'1' **then**

fib\_count **<=** std\_logic\_vector**(**unsigned**(**fib\_count**)** **+**1**);**

**if(**fib\_count**=**"1101"**)** **then** --reset after 13th element

cnt **<=** **(Others** **=>**'0'**);**

fib\_cur **<=** **(Others** **=>**'0'**);**

fib\_prev **<=** "00000001"**;**

fib\_count**<=** **(Others** **=>**'0'**);**

**else**

fib\_prev**<=**cnt**;**

fib\_cur**<=**fib\_nxt**;**

cnt **<=** fib\_nxt**;**

**end** **if;**

**end** **if;**

**end** **if;**

**end** **if;**

**end** **process;**

count **<=** cnt**;**

fib\_nxt **<=** std\_logic\_vector**(**unsigned**(**fib\_prev**)+**unsigned**(**fib\_cur**));**

**end** Behavioral**;**

1. *Freq\_Div.vhd*

--Author: Kyle Keislar

--CSUN ECE 524L Mirzaei

--Due Date: 09/15/2020

--

--File: Freq\_Div.vhd

--

--Design Units

--

--

--Library:

--Synthesis and Verification

-- Software: Xilnix Vivado 2019.1.1

-- Test bench: tb\_Lab\_1\_Design\_top.vhd

--

-- Date Last Revised: 09/12/2020

--

--

----------------------------------------------------------------------------------

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**use** IEEE**.**NUMERIC\_STD**.ALL;**

**entity** Freq\_Div **is**

**generic(** divisor**:** integer **:=** 10**;**

div\_width**:** integer **:=** 1**);**

**Port** **(** m\_clk **:** **in** STD\_LOGIC**;**

clk **:** **out** STD\_LOGIC**);**

**end** Freq\_Div**;**

**architecture** Behavioral **of** Freq\_Div **is**

**signal** div\_cnt **:** std\_logic\_vector**(**div\_width**-**1 **downto** 0**):=** **(Others** **=>** '1'**);**

**signal** max\_cnt **:** std\_logic\_vector**(**div\_width**-**1 **downto** 0**):=** std\_logic\_vector**(to\_unsigned(**divisor**,**div\_width**));**

**signal** clk\_out**:** std\_logic **:=**'0'**;**

**begin**

**process** **(**m\_clk**)**

**begin**

**if(**m\_clk'**event)** **AND** **(**m\_clk**=**'1'**)** **then**

**if** **(**div\_cnt**=**std\_logic\_vector**(**unsigned**(**max\_cnt**)** **-** 1**))** **then**

clk\_out **<=** **not** clk\_out**;**

div\_cnt **<=** **(Others** **=>**'0'**);**

**else**

div\_cnt **<=** std\_logic\_vector**(**unsigned**(**div\_cnt**)** **+** 1**);**

**end** **if;**

**end** **if;**

**end** **process;**

**with** clk\_out **select**

clk **<=** '1' **when** '1'**,**

'0' **when** '0'**,**

'0' **when** **others;**

**end** Behavioral**;**

1. *Lab\_1\_Const.xdc*

create\_clock -period 10.000 -name clk -waveform {0.000 5.000} [get\_ports clk]

create\_generated\_clock -name f\_div/clk\_out\_reg\_0 -source [get\_ports clk] -divide\_by 100000000 [get\_pins f\_div/clk\_out\_reg/Q]

create\_clock -period 1000000000.000 -name VIRTUAL\_f\_div/clk\_out\_reg\_0 -waveform {0.000 500000000.000}

set\_input\_delay -clock [get\_clocks VIRTUAL\_f\_div/clk\_out\_reg\_0] -min -add\_delay 2000.000 [get\_ports {sel[\*]}]

set\_input\_delay -clock [get\_clocks VIRTUAL\_f\_div/clk\_out\_reg\_0] -max -add\_delay 2000.000 [get\_ports {sel[\*]}]

set\_input\_delay -clock [get\_clocks VIRTUAL\_f\_div/clk\_out\_reg\_0] -min -add\_delay 2000.000 [get\_ports pause]

set\_input\_delay -clock [get\_clocks VIRTUAL\_f\_div/clk\_out\_reg\_0] -max -add\_delay 2000.000 [get\_ports pause]

set\_input\_delay -clock [get\_clocks VIRTUAL\_f\_div/clk\_out\_reg\_0] -min -add\_delay 2000.000 [get\_ports reset]

set\_input\_delay -clock [get\_clocks VIRTUAL\_f\_div/clk\_out\_reg\_0] -max -add\_delay 2000.000 [get\_ports reset]

set\_output\_delay -clock [get\_clocks VIRTUAL\_f\_div/clk\_out\_reg\_0] -min -add\_delay 0.000 [get\_ports {led[\*]}]

set\_output\_delay -clock [get\_clocks VIRTUAL\_f\_div/clk\_out\_reg\_0] -max -add\_delay 2000.000 [get\_ports {led[\*]}]