EECS 140: Lab 7

EECS 140 Introduction Structural VHDL

Morgan Bergen

KUID: 3073682

Date submitted: 10/22/2022

1. Introduction & Background

The purpose of this lab was to learn the basics of structural VHDL models and design simple cir suits using structural VHDL. The prelab consisted of understanding a high level overview of VHDL component declaration, signal declaration, and component instantiation/port mapping. The procedure consisted of maintaining a design that confirmed with Jorge Ortiz's guidelines for the generic procedures for the design implementation of a bottom up approach to building a six input to one output gate that was able to facilitate a one to one input to output relationship. First we needed to make an entity that accepted six bits of inputs and output the result of a 6-way XOR gate, then declare and define an exclusive or gate between inputs that yielded true whenever the inputs were on, thereafter we designed the 3-bit XOR gate that was the input to the 2-input XOR gate from the previous step, next we made an entity that simply accepted six bits, and after lastly we built a my_xor6_gate that accepted 2 three inputs which allowed for the same result as the top level entity using the 6-bit inputs. This process of bottom up and then top down design implementation allow for the decomposition of problems to yield the correct solution.

The following are the definitions needed to understanding the structural design architectures of this lab,

component declaration: defines the input and output ports for all of the differing components.
component instantiation: naming variable assignments for each component
port mapping: connected each port to every designed component name aka each instantiated components, which allowed for the connections to be made directly to the encapsulated entity's inputs and outputs or to internal signals.

2. Implementation Process

The implementation process consisted of designing the sources for the my_xor6_gate.vhd, my_xor2_gate.vhd, my_xor3_gate.vhd, and designing the constraints for these three source files. The first top level entity was a xor2 gate that ported a, b as the STD_LOGIC inputs, and g as the STD_LOGIC output, this was the port mapping of the architecture behavioral of my_xor2_gate.

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
ENTITY my_xor2_gate IS
   PORT (
      A,B : IN STD_LOGIC;
      G   : OUT STD_LOGIC);
END my_xor2_gate;
ARCHITECTURE Behavioral of my_xor2_gate IS
BEGIN
   G <= (A AND (NOT B)) OR ((NOT A) AND B);
END ARCHITECTURE Behavioral;
```

This was our first component which lead us to create another xor gate with more inputs in order to reach out goal of a 6-bit input XOR gate. The next was to design an entity that had the associativity property of the XOR gates to allow for the following expression to be valid within the flow of control x = a xor b xor c = (a xor b) xor c = g xor c, where g can be the output of the 2-bit xor gate.

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
ENTITY my_xor3_gate IS
   PORT (
      A,B,C : IN STD_LOGIC;
      H     : OUT STD_LOGIC);
END my_xor3_gate;
ARCHITECTURE Structural of my_xor3_gate IS
   -- Component Declaration
   COMPONENT my_xor2_gate IS
      PORT (
         A,B : IN STD_LOGIC;
         G   : OUT STD_LOGIC);
   END COMPONENT my_xor2_gate;
   -- Signal Declaration
   SIGNAL AxorB: STD_LOGIC;
BEGIN
   -- Component Instantiation
   AB: my_xor2_gate
      PORT MAP (
         A => A,
         B => B,
G => AxorB);
   ABC: my_xor2_gate
      PORT MAP (
         A => AxorB,
B => C, G => H);
END ARCHITECTURE Structural;
```

The last component that was defined was the my_xor6_gate that pulled every component together, and this allowed for the components to all speak to each other, via the port maps. Lastly we needed to create the constraints file for each input and output.

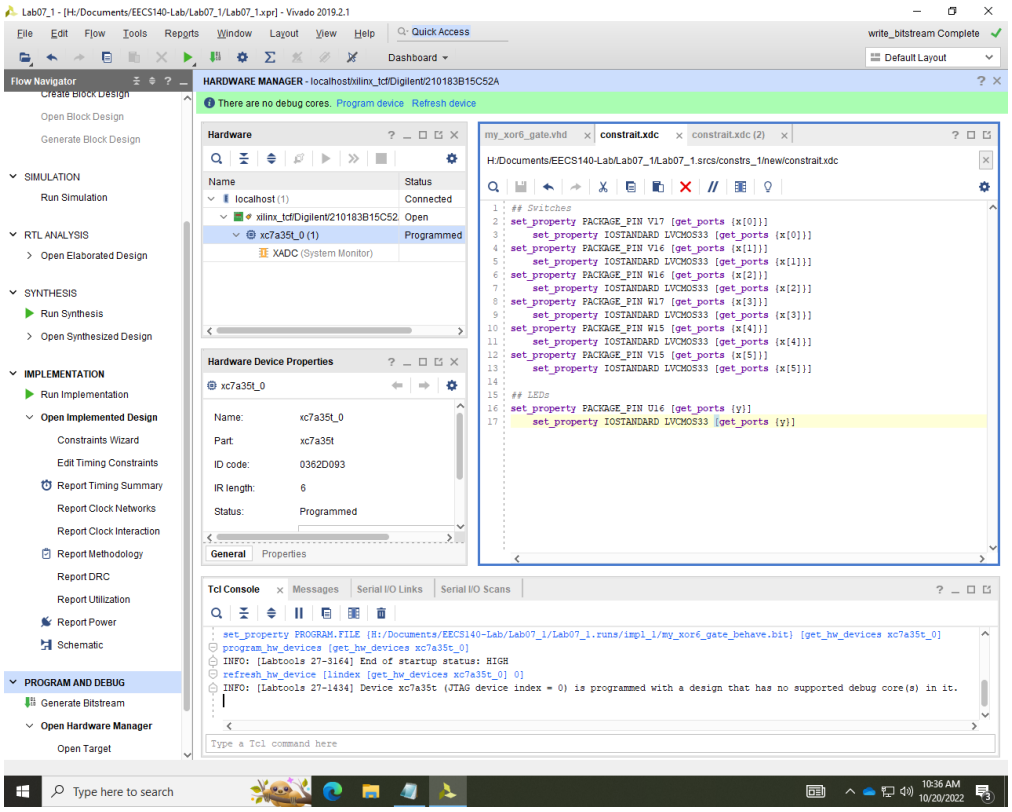
```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
ENTITY my_xor6_gate IS
   PORT (
      U,V,W,X,Y,Z: IN STD_LOGIC;
      I          : OUT STD_LOGIC);
END my_xor6_gate;
ARCHITECTURE Structural of my_xor6_gate IS
```

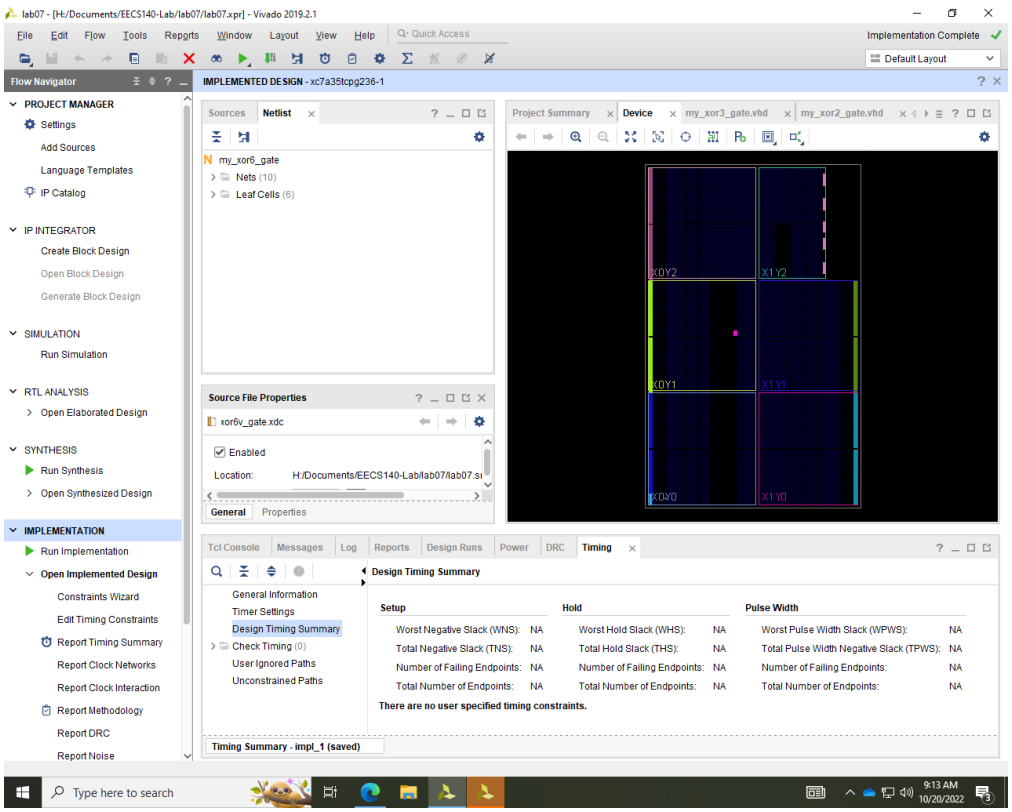
```
   -- Component Declaration
   COMPONENT my_xor2_gate IS
      PORT (
         A,B : IN STD_LOGIC;
         G   : OUT STD_LOGIC);
   END COMPONENT my_xor3_gate;
   COMPONENT my_xor3_gate IS
      PORT (
         A,B,C : IN STD_LOGIC;
         H     : OUT STD_LOGIC);
   END COMPONENT my_xor3_gate;
   -- Signal Declaration
   SIGNAL UxorVxorW: STD_LOGIC;
   SIGNAL XxorYxorZ: STD_LOGIC;
BEGIN
   -- Component Instantiation
   UVW: my_xor3_gate
      PORT MAP (
         A => U,
         B => V,
         C => W,
         H => UxorVxorW);
   XYZ: my_xor3_gate
      PORT MAP (
         A => X,
         B => Y,
         C => Z,
         H => XxorYxorZ);
   UVWXYZ: my_xor2_gate
      PORT MAP (
         A => UxorVxorW,
         B => XxorYxorZ,
         G => I);
END ARCHITECTURE Structural;
```

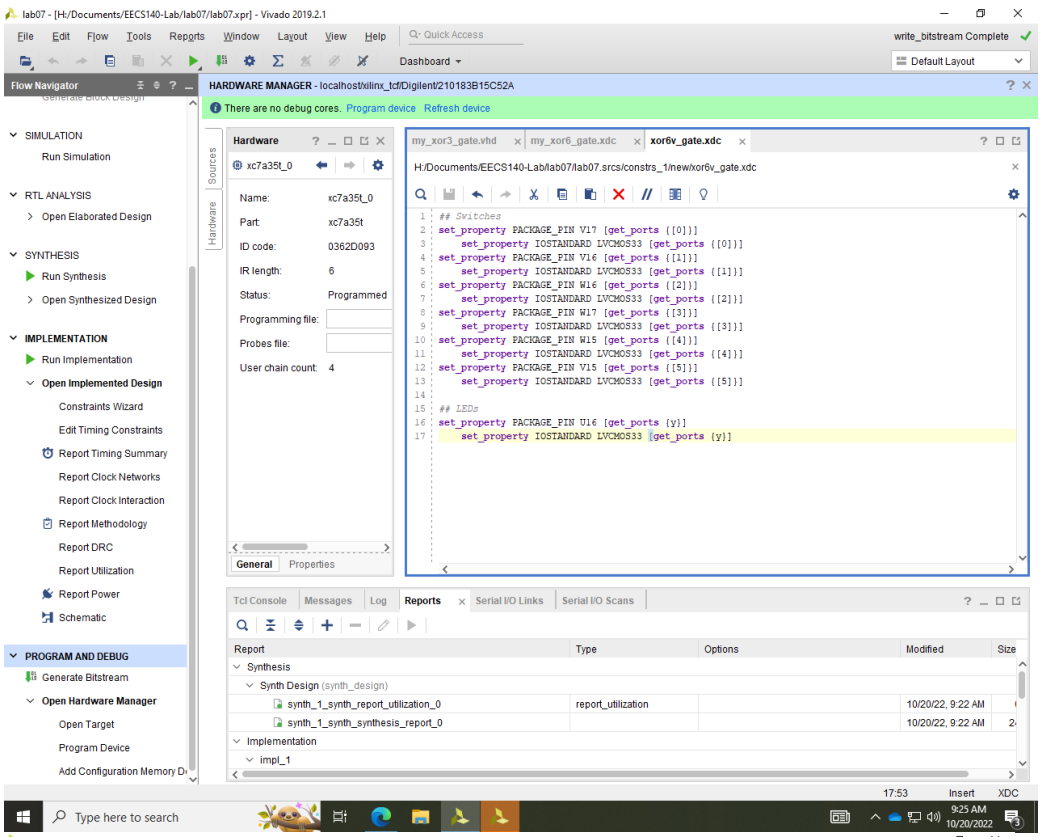
This process consisted of the bottom up approach, which contained more verbose code and more encapsulated files, however the support for the std_logioc_vectors allowed for multiple different bits to be addressed as a input/output signal and address each individual via as a vector similar to an array data type, the logical expression allowed for the same output,

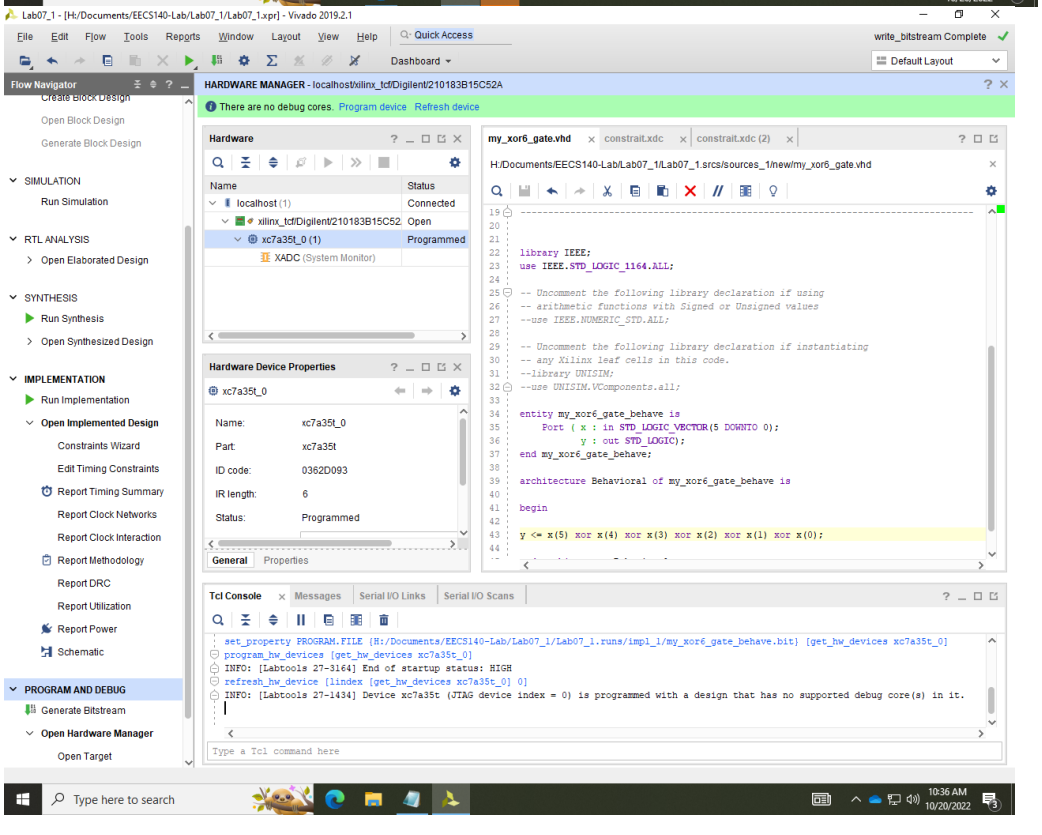
```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
ENTITY my_xor6_gate IS
   PORT (
      X : IN STD_LOGIC_VECTOR(5 DOWNTO 0);
      Y : OUT STD_LOGIC);
```

```
END my_xor6_gate;
ARCHITECTURE Behavioral of my_xor6_gate IS
BEGIN
   Y <= X(5) XOR X(4) XOR X(3) XOR X(2) XOR X(1) XOR X(0);
END ARCHITECTURE Structural;


## xor6v_gate.xdc constraints file
## Switches
set_property PACKAGE_PIN V17 [get_ports {[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {[0]}]
set_property PACKAGE_PIN V16 [get_ports {[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {[1]}]
set_property PACKAGE_PIN W16 [get_ports {[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {[2]}]
set_property PACKAGE_PIN W17 [get_ports {[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {[3]}]
set_property PACKAGE_PIN W15 [get_ports {[4]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {[4]}]
set_property PACKAGE_PIN V15 [get_ports {[5]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {[5]}]

## LEDs
set_property PACKAGE_PIN U16 [get_ports {Y}]
    set_property IOSTANDARD LVCMOS33 [get_ports {Y}]


## xor6_gate.xdc constraints file
## Switches
set_property PACKAGE_PIN V17 [get_ports {U}]
    set_property IOSTANDARD LVCMOS33 [get_ports {U}]
set_property PACKAGE_PIN V16 [get_ports {V}]
    set_property IOSTANDARD LVCMOS33 [get_ports {V}]
set_property PACKAGE_PIN W16 [get_ports {W}]
    set_property IOSTANDARD LVCMOS33 [get_ports {W}]
set_property PACKAGE_PIN W17 [get_ports {X}]
    set_property IOSTANDARD LVCMOS33 [get_ports {X}]
set_property PACKAGE_PIN W15 [get_ports {Y}]
    set_property IOSTANDARD LVCMOS33 [get_ports {Y}]
set_property PACKAGE_PIN V15 [get_ports {Z}]
    set_property IOSTANDARD LVCMOS33 [get_ports {Z}]

## LEDs
set_property PACKAGE_PIN U16 [get_ports {I}]
    set_property IOSTANDARD LVCMOS33 [get_ports {I}]
```


Lab07_1 - [H:/Documents/EECS140-Lab/Lab07_1/Lab07_1.xpr] - Vivado 2019.2.1
write_bitstream Complete
HARDWARE MANAGER - localhost/xilinx_tcf/Digilent/210183B15C52A
There are no debug cores. Program device Refresh device
Hardware
Hardware Device Properties
xc7a35t_0
Name: xc7a35t_0
Part: xc7a35t
ID code: 0362D093
IR length: 6
Status: Programmed
my_xor6_gate.vhd
constrait.xdc
constrait.xdc (2)
H:/Documents/EECS140-Lab/Lab07_1/Lab07_1.srcs/constrs_1/new/constrait.xdc
Switches
set_property PACKAGE_PIN V17 [get_ports {x[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {x[0]}]
set_property PACKAGE_PIN V16 [get_ports {x[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {x[1]}]
set_property PACKAGE_PIN W16 [get_ports {x[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {x[2]}]
set_property PACKAGE_PIN W17 [get_ports {x[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {x[3]}]
set_property PACKAGE_PIN W15 [get_ports {x[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {x[4]}]
set_property PACKAGE_PIN V15 [get_ports {x[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {x[5]}]
LEDs
set_property PACKAGE_PIN U16 [get_ports {y}]
set_property IOSTANDARD LVCMOS33 [get_ports {y}]
Tcl Console
INFO: [Labtools 27-1434] Device xc7a35t (JTAG device index = 0) is programmed with a design that has no supported debug core(s) in it.
Type a Tcl command here
Type here to search
10:36 AM
10/20/2022


lab07 - [H:/Documents/EECS140-Lab/lab07/lab07.xpr] - Vivado 2019.2.1
Implementation Complete
IMPLEMENTED DESIGN - xc7a35tcpg236-1
Sources
Netlist
my_xor6_gate
Nets (10)
Leaf Cells (6)
Source File Properties
xor6v_gate.xdc
Enabled
Location: H:/Documents/EECS140-Lab/lab07/lab07.s
Project Summary
Device
my_xor3_gate.vhd
my_xor2_gate.vhd
X0Y2
X1Y2
X0Y1
X1Y1
X0Y0
X1Y0
Tcl Console
Messages
Log
Reports
Design Runs
Power
DRC
Timing
Design Timing Summary
General Information
Timer Settings
Design Timing Summary
Check Timing (0)
User Ignored Paths
Unconstrained Paths
Setup
Worst Negative Slack (WNS): NA
Total Negative Slack (TNS): NA
Number of Failing Endpoints: NA
Total Number of Endpoints: NA
Hold
Worst Hold Slack (WHS): NA
Total Hold Slack (THS): NA
Number of Failing Endpoints: NA
Total Number of Endpoints: NA
Pulse Width
Worst Pulse Width Slack (WPWS): NA
Total Pulse Width Negative Slack (TPWS): NA
Number of Failing Endpoints: NA
Total Number of Endpoints: NA
There are no user specified timing constraints.
Timing Summary - impl_1 (saved)
Type here to search
9:13 AM
10/20/2022


lab07 - [H:/Documents/EECS140-Lab/lab07/lab07.xpr] - Vivado 2019.2.1
HARDWARE MANAGER - localhost/xilinx_tcf/Digilent/210183B15C52A
There are no debug cores. Program device Refresh device
Hardware
xc7a35t_0
Name: xc7a35t_0
Part: xc7a35t
ID code: 0362D093
IR length: 6
Status: Programmed
Programming file:
Probes file:
User chain count: 4
my_xor3_gate.vhd
my_xor6_gate.xdc
xor6v_gate.xdc
H:/Documents/EECS140-Lab/lab07/lab07.srcs/constrs_1/new/xor6v_gate.xdc
Switches
set_property PACKAGE_PIN V17 [get_ports {[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {[0]}]
set_property PACKAGE_PIN V16 [get_ports {[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {[1]}]
set_property PACKAGE_PIN W16 [get_ports {[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {[2]}]
set_property PACKAGE_PIN W17 [get_ports {[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {[3]}]
set_property PACKAGE_PIN W15 [get_ports {[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {[4]}]
set_property PACKAGE_PIN V15 [get_ports {[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {[5]}]
LEDs
set_property PACKAGE_PIN U16 [get_ports {y}]
set_property IOSTANDARD LVCMOS33 [get_ports {y}]
Reports
Synthesis
Synth Design (synth_design)
synth_1_synth_report_utilization_0
synth_1_synth_synthesis_report_0
Implementation
impl_1


Lab07_1 - [H:/Documents/EECS140-Lab/Lab07_1/Lab07_1.xpr] - Vivado 2019.2.1
HARDWARE MANAGER - localhost/xilinx_tcf/Digilent/210183B15C52A
There are no debug cores. Program device Refresh device
Hardware
localhost (1)
xilinx_tcf/Digilent/210183B15C52
xc7a35t_0 (1)
XADC (System Monitor)
Hardware Device Properties
Name: xc7a35t_0
Part: xc7a35t
ID code: 0362D093
IR length: 6
Status: Programmed
my_xor6_gate.vhd
constrait.xdc
constrait.xdc (2)
H:/Documents/EECS140-Lab/Lab07_1/Lab07_1.srcs/sources_1/new/my_xor6_gate.vhd
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 my_xor6_gate_behave is
Port (x : in STD_LOGIC_VECTOR(5 DOWNTO 0);
y : out STD_LOGIC);
end my_xor6_gate_behave;
architecture Behavioral of my_xor6_gate_behave is
begin
y <= x(5) xor x(4) xor x(3) xor x(2) xor x(1) xor x(0);
Tcl Console
set_property PROGRAM.FILE {H:/Documents/EECS140-Lab/Lab07_1/Lab07_1.runs/impl_1/my_xor6_gate_behave.bit} [get_hw_devices xc7a35t_0]
program_hw_devices [get_hw_devices xc7a35t_0]
INFO: [Labtools 27-3164] End of startup status: HIGH
refresh_hw_device [lindex [get_hw_devices xc7a35t_0] 0]
INFO: [Labtools 27-1434] Device xc7a35t (JTAG device index = 0) is programmed with a design that has no supported debug core(s) in it.
Type a Tcl command here

3. Evaluation Process

The top level entity design too creat can entity using the generic produces for the implementation accepted 6 - 1-bit input and the 1 - output. The evaluation process consisted of the input output relationships, the truth table outlines the following correct evaluations,

input						output
v17	v16	w16	w17	w15	v15	u16
0	0	1	0	0	0	1
1	0	0	0	0	0	1
0	0	0	0	0	0	0
…						

The testing procedure using the XOR truth tables results allowed for the circuits to work as accordingly. When three inputs for the U, V, W from the project of the my_xor3_gate and the x, y, z of the other my_xor3_gate (top down design) both contained outputs that allowed or the indicated to be on. I tested and ran the functionality using the 6-FPGA slide switches and 1 output led.

4. Results & Discussion

The testing procedure using the XOR truth tables resulted in the circuit working correctly. Figure 7 shows the block diagram of how the inputs are being implemented. 3 inputs enter the UVW of my_xor3_gate and 3 inputs enter XYZ of the second my_xor3_gate. Both outputs are then used for the inputs of my_xor2_gate which result in Output I. After the test was successful, I modified the constraints and behavior of the my_xor6_gate to support STD_LOGIC_VECTORs. Created a new source with 1 input as X and 1 output as Y. I specified the BUS MSB 0 and LSB 5 for the switches (X) input. The constraints were modified using [ ]. Then ran a successful synthesis, implementation and bitstream generation. Circuit worked correctly per the truth tables.

5. Conclusions & Recommendation

The conclusive discoveries consisted of understanding on an implementation level what it means to instantiate using component declaration functions, signal declaration, and component instantiation/port mapping. The Tutorial allowed me to learn what it means to “divide and conquer” the correct way and break up the design in manageable steps that made the implementation and evaluation process easier. I think one way to improve on this would be to provide less explicit code and allow for the programmer to implement the logic on their own.