Experiment 1: Combinational Circuit 1

Anubhav Bhatla | 200070008

EE-214, WEL, IIT Bombay

August 10, 2021

## Overview of the experiment:

|  |
| --- |
| The purpose of this experiment was to design a set of Multiplexers, understand the theory behind it and then implementing the design using VHDL.  Quartus Prime was used to implement the given specifications and my own designs with VHDL. RTL and Gate Level simulations were then carried out using the created Testbench to check for the correctness of the design and its implementation.  The report consists of the following sections:   * **Approach:** An overview of how I approached the experiment and the logic behind my designs. * **Design document and VHDL code:** A brief description of all the design components. * **RTL View:** A screenshot of the RTL View generated using Quartus Prime. * **RTL Simulation:** Screenshots for the RTL and Gate Level Simulations. |

## Approach to the experiment:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Part A: 2x1 Mux**  The objective was to write the VHDL code for the design given below:    Given below is the truth table for the same:   |  |  |  |  | | --- | --- | --- | --- | | **I1** | **I0** | **S** | **Y** | | 0 | 0 | 0 | 0 | | 0 | 1 | 0 | 1 | | 1 | 0 | 0 | 0 | | 1 | 1 | 0 | 1 | | 0 | 0 | 1 | 0 | | 0 | 1 | 1 | 0 | | 1 | 0 | 1 | 1 | | 1 | 1 | 1 | 1 |   The Boolean expression for the output Y can be found in the following way:    **Part B: 4x1 Mux**        **Part C: 4-bit 4x1 Mux** |

## 

## Design document and VHDL code if relevant:

|  |
| --- |
| **Part A: 2x1 Mux**  Given below is the code for the architecture of the entity **Mux\_2x1** which is the 2x1 Multiplexer we had to implement in Quartus using the given design:    **Part B: 4x1 Mux**  Given below is the code for the architecture of the entity **Mux\_4x1** which is the 4x1 multiplexer I implemented using my design and only making use of 2x1 multiplexers:    **Part C: 4-bit 4x1 Mux**  Given below is the code for the architecture of the entity **Mux\_4x1\_4bit** which is the 4-bit 4x1 multiplexer implemented using my design and only making use of 4x1 multiplexers: |

## RTL View:

|  |
| --- |
| **Part A: 2x1 Mux**    **Part B: 4x1 Mux**    **Part C: 4-bit 4x1 Mux** |

## DUT Input/Output Format:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Part A: 2x1 Mux**   |  |  | | --- | --- | | Input  <In1><In0><S> | Output  <Y> | | 010 | 1 | | 110 | 1 | | 011 | 0 | | 111 | 1 |   **Part B: 4x1 Mux**   |  |  | | --- | --- | | Input  <In3><In2><In1><In0><S1><S0> | Output  <Y> | | 110100 | 1 | | 010110 | 1 | | 011111 | 0 | | 110011 | 1 |   **Part C: 4-bit 4x1 Mux**   |  |  | | --- | --- | | Input  <d3 d2 d1 d0><c3 c2 c1 c0><b3 b2 b1 b0><a3 a2 a1 a0><sel1 sel0> | Output  <Y3 Y2 Y1 Y0> | | 000001010100000100 | 0001 | | 000011010011111000 | 1110 | | 000101000010100000 | 1000 | | 111001100110100100 | 1001 | |

## RTL Simulation:

|  |
| --- |
| **Part A: 2x1 Mux**    **Part B: 4x1 Mux**    **Part C: 4-bit 4x1 Mux** |

## Gate-level Simulation:

|  |
| --- |
| **Part A: 2x1 Mux**    **Part B: 4x1 Mux**    **Part C: 4-bit 4x1 Mux** |

## Krypton board\*:

|  |
| --- |
| Map the logic circuit to the Krypton board and attach the images of the pin assignment and output observed on the board (switches/LEDs). |

## Observations\*:

|  |
| --- |
| You must summarize your observations, either in words, using figures and/or tables. |

## References:

|  |
| --- |
| None |

\* To be submitted after the tutorial on “Using Krypton”.