Flip Flop Variants

Javier Mondragón Martin del Campo

Laboratory exercise number:

Lab. 01

Second Partial

Laboratory exercise name:

Assignment 05

|  |  |  |
| --- | --- | --- |
| Names: | Roll Number | Date |
| Javier Mondragón M. | A01365137 | 2/October/2019 |

Lab description:

The purpose of this laboratory is to learn how program an FPGA using VHDL with the architecture “dataflow” using the skills learnt in class. The Flip Flops are components to use to detect latches in a period while a circuit is running. They are set to LOW until a rising edge is detected on a clock the output sets to HIGH. The variants could be when it resets, clear or set a default value to it.

The material used was:

ISE Project Naviator for the VHDL compiler and editor

Schematics, block diagrams and/or timing diagrams:

Using the following truth of table, this Flip Flops were variants not seen in class. The code is shown in the image 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| D | RST | PRE | Clk | Q |
| X | X | 1 | ↓ | 1 |
| X | 1 | 0 | ↓ | 0 |
| 0 | 0 | 0 | ↓ | 0 |
| 1 | 0 | 0 | ↓ | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| D | EN | PRE | RST | Clk | Q |
| X | X | X | 1 | ↓ | 0 |
| X | X | 1 | 0 | ↓ | 1 |
| 0 | 1 | 0 | 0 | ↓ | 0 |
| 1 | 1 | 0 | 0 | ↓ | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| D | EN | PRE | Clk | Q |
| X | X | 1 | ↓ | 1 |
| 0 | 1 | 0 | ↓ | 0 |
| 1 | 1 | 0 | ↓ | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| D | EN | RST | Clk | Q |
| X | X | 1 | ↓ | 0 |
| 0 | 1 | 0 | ↓ | 0 |
| 1 | 1 | 0 | ↓ | 1 |

Image 1.1

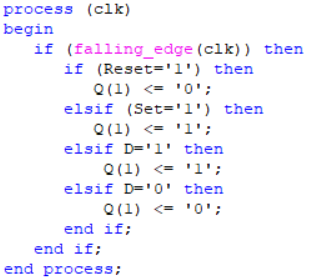


Image 1.2

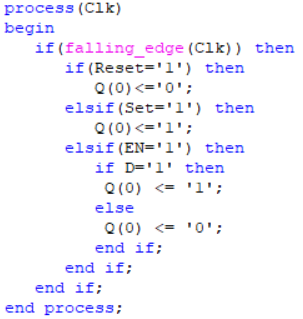


Image 1.3

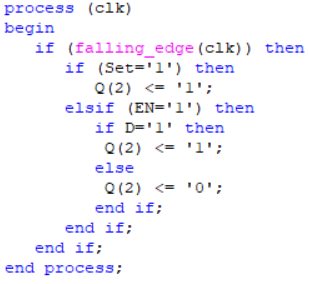
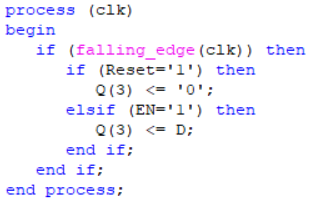
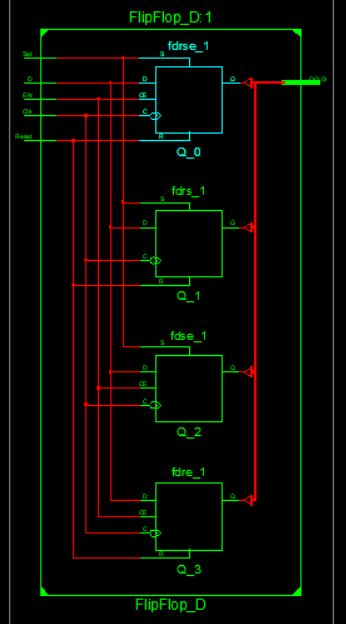


Image 1.4



Results obtained:

The results obtained was successfully created new ways to implement flip flops type D. We could see them in the following diagram.



Conclusions:

VHDL is open to new ways to program and resolve things to get done easily and faster, the prefabricated components and detecting them with lines of code was amazing and is easier to understand.

Problems encountered:

Sometimes VHDL has problems with which variables are inside to read the process and comprehend what type of flip flop is it.