

Entorno gráfico de active HDL

1° Practica

Integrants:

|  |  |
| --- | --- |
| Nombre | Expediente |
| Zuñiga Fragoso Diego Joel | 317684 |

Subject: Sistemas digitales de lógica reconfigurable I

Teacher: Marcos Romo Aviles

1. **Objective:**

In this practice session, we will learn about the basics and functionality of an FPGA. We will also learn how to use the IDE and perform real-time simulations using the Aldec Active-HDL software.

**Methodology:**

1. Go to the official website to download the executable installer.

2. Provide the requested information and complete the installation process.

3. Run the installed program and, following the professor's instructions, code the instructions that the FPGA will follow.

4. Learn to identify the inputs and outputs, and become familiar with the boolean operations of the language.

5. Create a benchmark file to test the output of all possible combinations of inputs.

6. Run the simulation of the file and observe the behavior of the output based on the different inputs.

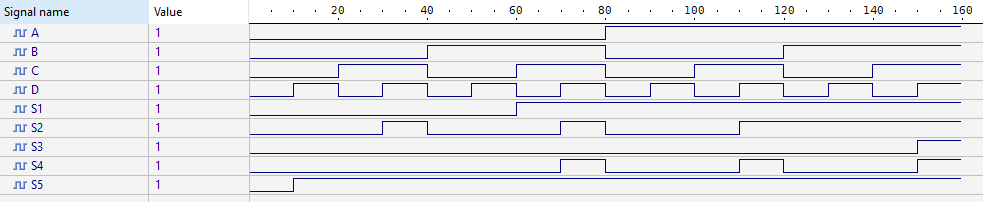
1. **Development:**
   1. **Code**

Next, I will provide the code with explanations for each part that composes it:

|  |
| --- |
| library IEEE;  use IEEE.std\_logic\_1164.all;  -- We initialize the variables as inputs or outputs  entity P1 is  port  (  A: in std\_logic;  B: in std\_logic;  C: in std\_logic;  D: in std\_logic;  S1: out std\_logic;  S2: out std\_logic;  S3: out std\_logic;  S4: out std\_logic;  S5: out std\_logic  );  end P1;  -- Based on the variables, we generate an architecture to relate the value of the outputs to the inputs  Architecture Pr1 of P1 is  begin  S1<= A or (B and C);  S2<= (C and D) or (A and B);  S3<= (B and A and C and D);  S4<= C and ( A or B) and D;  S5<= A or B or C or D;  end Pr1; |

|  |
| --- |
| library IEEE;  use IEEE.std\_logic\_1164.all;  entity P1\_TB is  end P1\_TB;  -- We generate a new architecture with the same variables  architecture TB of P1\_TB is  component P1  port  (  A: in std\_logic;  B: in std\_logic;  C: in std\_logic;  D: in std\_logic;  S1: out std\_logic;  S2: out std\_logic;  S3: out std\_logic;  S4: out std\_logic;  S5: out std\_logic  );  end component;  -- We simulate de input and output values  signal A,B,C,D:std\_logic; -- Inputs  signal S1,S2,S3,S4,S5:std\_logic; -- Outputs  begin  inicio: P1 port map (A=>A,B=>B,C=>C,D=>D,S1=>S1,S2=>S2,S3=>S3,S4=>S4,S5=>S5);  Parasimu: process  begin  A<='0'; B<='0'; C<='0'; D<='0'; wait for 10ns;  A<='0'; B<='0'; C<='0'; D<='1'; wait for 10ns;  A<='0'; B<='0'; C<='1'; D<='0'; wait for 10ns;  A<='0'; B<='0'; C<='1'; D<='1'; wait for 10ns;  A<='0'; B<='1'; C<='0'; D<='0'; wait for 10ns;  A<='0'; B<='1'; C<='0'; D<='1'; wait for 10ns;  A<='0'; B<='1'; C<='1'; D<='0'; wait for 10ns;  A<='0'; B<='1'; C<='1'; D<='1'; wait for 10ns;  A<='1'; B<='0'; C<='0'; D<='0'; wait for 10ns;  A<='1'; B<='0'; C<='0'; D<='1'; wait for 10ns;  A<='1'; B<='0'; C<='1'; D<='0'; wait for 10ns;  A<='1'; B<='0'; C<='1'; D<='1'; wait for 10ns;  A<='1'; B<='1'; C<='0'; D<='0'; wait for 10ns;  A<='1'; B<='1'; C<='0'; D<='1'; wait for 10ns;  A<='1'; B<='1'; C<='1'; D<='0'; wait for 10ns;  A<='1'; B<='1'; C<='1'; D<='1'; wait for 10ns;  wait;  end process Parasimu;  end TB; |

* 1. **Simulation**

****

1. **Conclusion**

I have successfully learned to use the development environment effectively and am beginning to familiarize myself with FPGA programming. This process has allowed me to better understand the fundamental concepts and advanced techniques necessary to work with these devices. Additionally, I have successfully completed the simulation of the Boolean operations assigned in class, which has provided me with valuable practical experience and reinforced my theoretical understanding of logical operations.