Workshop on Programmable Logic Devices

Łukasz Sajewski

Tasks Part-1

Burak ELHAMAN

**Introduction**

We have 3 inputs and 4 outputs. We prepare logic device that converts natural binary code to “2 from 4” code. We only have a natural binary in input. On the exit we have two “1” containing codes. All outputs are different each other.

**What is FPGA?**

FPGA is mean Field Programmable Gate Array. FPGA can operate at high frequencies and has a parallel operation. FPGA has got two programmable species HDL and Block design, HDL has got two option Verilog HDL and VHDL. VHDL is more popular than Verilog.

 There are keys, push buttons, LED, 7 segment displays, LCD and Pins on FPGA. If we make group our inputs are keys, push buttons and our outputs are 7 segment display, LED, LCD. If we want to use as input we can connect some sensor to FPGA and if we want to use as output and for example we can connect motor with a USB cable to the computer and using some programs like Xilinx or Altera Quartus we can load code or logic circuit to FPGA.

**What is Verilog?**

Verilog is using for design electronic models. It is a hardware description language. Verilog support design and verify in analog, digital and mixed mark circuits. Verilog is very close to C programmable language that's why people who know C programmable language they can learn faster Verilog. it has got command like "if, else, while" like C programmable language. but Verilog has got "begin, end" instead of C programmable language and Finally if we talk about the history of Verilog we can say that, Verilog invented by Phil Moorby and Prabhu Goel in 1983 Verilog Purchased by Cadence Company and Right now this company has got all rights of Verilog.

**Verilog Code of First Project**

module firstproject(input [2:0] A, output reg [3:0] B);

always @(\*)

case (A)

3'b000:B = 4'b1001;

3'b001:B = 4'b1010;

3'b010:B = 4'b1100;

3'b011:B = 4'b0110;

3'b100:B = 4'b0101;

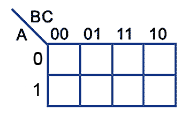
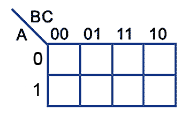
3'b101:B = 4'b0011;

default:B= 4'b1111;

endcase

endmodule

**OPTIMIZATION**



**0 0 1 1**

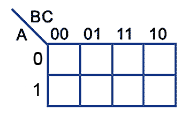
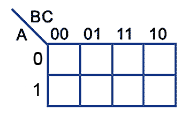
**1 0 X X**

**1 1 0 1**

**0 0 X X**

X=B + AC’

W=A’B’ + BC’



**0 1 1 0**

**0 1 X X**

**1 0 0 0**

**1 1 X X**

Y=C

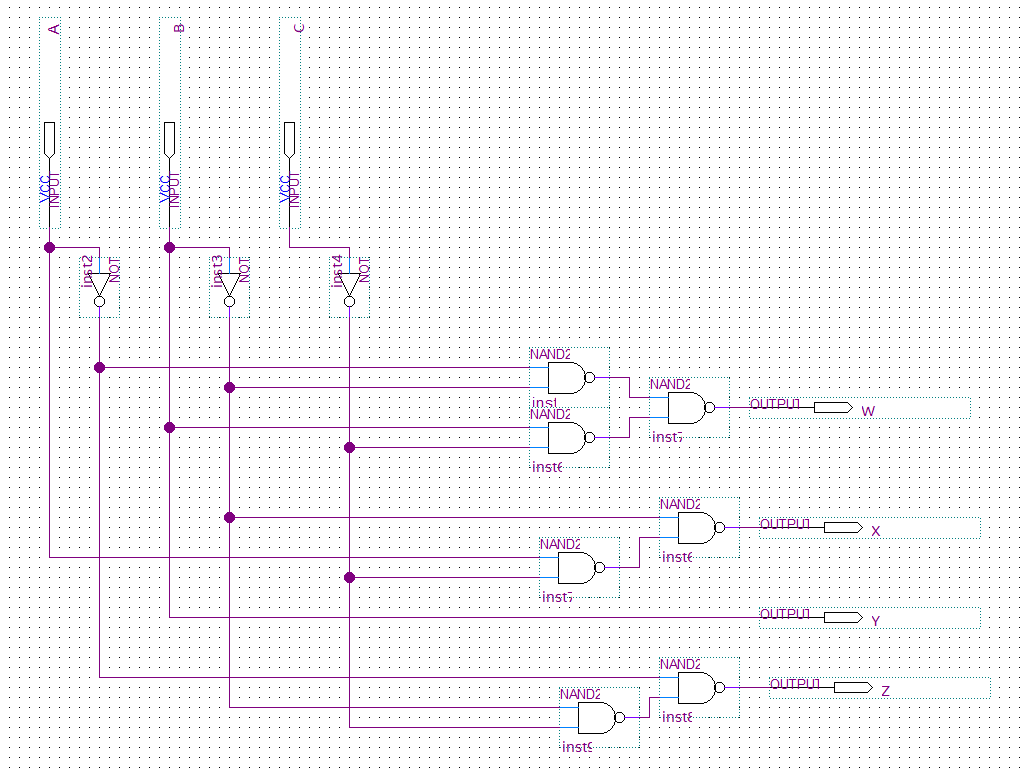
Z=A + B’C’

**TABLE OF TRUTH**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| A | B | C | W | X | Y | Z |
| 0 | 0 | 0 | **1** | **0** | **0** | **1** |
| 0 | 0 | 1 | **1** | **0** | **1** | **0** |
| 0 | 1 | 0 | **1** | **1** | **0** | **0** |
| 0 | 1 | 1 | **0** | **1** | **1** | **0** |
| 1 | 0 | 0 | **0** | **1** | **0** | **1** |
| 1 | 0 | 1 | **0** | **0** | **1** | **1** |
| 1 | 1 | 0 | X | X | X | X |
| 1 | 1 | 1 | X | X | X | X |

**What is Logic Block?**

There are logic blocks on Quartus we can design our circuit with this logic blocks there are so type gates like "or, and, nand" we can connect these gates with each other and we can create the complex systems these blocks work with 1 and 0 after we create our circuit we must add our inputs and outputs these can be led, key and button we can define on Quartus.

**Gate Realization and Cad Simulation**

