Project

The essence of the project is cooperation in small groups, study of the assigned topic, design of your own solution, simulation, implementation, and presentation of results.

Students solve the project from the 9th to the 13th week of the semester in the same groups as in the computer labs. Source codes for CPLD must be written in VHDL language and implemented on Xilinx CoolRunner-II CPLD starter board (XC2C256-TQ144).

Students may consult with their tutor from the computer labs and may not "assign" anyone else, including consultations with other student, academics or PhD students.

The project is submitted on Thursday April 30, 2020 via the GitHub repository. It must contain:

- own README file with a complete description of the problem, circuit diagram of peripherals, graphical representation of the application (flowchart / state diagram), screenshots of simulations, links to source files in the repository, discussion of results, list of references, link to your own video with a demo, etc. The README file must be written in English
- complete project in Xilinx Vivado
- testbench of all developed components
- all source files must be written in English

The complexity of the created application, functionality, compliance with VHDL guidelines, division of code into modules, code purity, use of comments, completeness and clarity of documentation, application wiring diagrams, discussion of results, compliance with assignment instructions are evaluated.

Use VHDL modules from previous labs (especially `clock_enable` , `binary_cnt` , `driver_7seg`) and not codes taken from the Internet. Design the application as a synchronous sequential system.

Topics 2019/20

1. Code lock with 4x3 matrix keyboard, time limit for entering the correct pin code and signaling an incorrect attempt.

2. PWM dimmer with adjustable "timing" with rotary encoder KY-040 and push button. After the time has elapsed, the output of 100% PWM is continuously dimmed to zero.

3. UART (Universal asynchronous receiver / transmitter) transmitter with adjustable UART frame, data word and one of two preset standard symbol rate parameters while the application is running.

4. Ultrasonic distance meter HC-SR04. Output on multiplexed 7-segment display.

5. Own ALU (Arithmetic Logic Unit) unit. Possibility to select instructions and input values while the application is running. Output on multiplexed 7-segment display.

6. Countdown from set value to zero. Value is given by rotary encoder KY-040 with button, output is shown on serial 4-digit 7-segment display with control circuit TM1637.

Mandatory structure of your README file:

Your project title

Objectives

TBD

Hardware description

TBD

Code description and simulations

TBD

Video

TBD

References

TBD