CSCI Projects guidelines

1- Team should be from 4 to 5 max within the same lecture group and all team members should attend at the same time

2- Submit the following:

a) Report that includes:

- **Names and Ids**
- **Description of your project**
- **Any equations, K maps and Truth Table**
- **Components list and costs**
- **Simulation screenshot from Proteus or circuitverse and a screenshot TinkerCad**

3- Evaluation Criteria (20)

a) Discussion (5 points)

b) Software Implementation (Proteus, TinkerCad) (10 points)

c) Equations, K maps and Truth table (5 points)

4- Deadline 2/1 (There will be a submission link on Moodle, One submission per team)

5- Discussion from (2/1-6/1) in the lab time

***The grading scheme is 10 marks (after scaling) for project to be discussed in the last semester week.**

Students participating in UGRF will be given 2.5 marks and those implementing a hardware prototype will have extra 2.5 marks.

Project ideas:

1. Design a system to count number of present students in class and display the number of absent ones. The system consists of a photocell to detect each student while entering the classroom, a counter, 3 registers to carry the total number of enrolled students, the number of present ones and the number absent ones, and a subtractor.

2. Design a system to detect number of vans passing by a certain gate. The system consists of a photocell to detect the passing vehicles, a comparator to decide whether that vehicle is a van or not based on its weight on a ground

scale, a counter to count only vans, and a register to carry the number of vans

3. Design a system to count number of present students in class and display the number of absent ones. The system consists of a photocell to detect each student while entering the classroom, a counter, 3 registers to carry the total number of enrolled students, the number of present ones and the number absent ones, and a subtractor.

4. Design a traffic light system to control the flow of the vehicles. The project aims to implement a 4-way intersection traffic controller. The basic idea behind the design is to avoid the collision of the vehicles by providing appropriate signals to different directions a limited time slot, after which the next waiting driver will be given the same treatment. In this way a cycle will be established which will control the traffic.

5. Design a system to count number of present students in class and display the number of absent ones. The system consists of a photocell to detect each student while entering the classroom, a counter, 3 registers to carry the total number of enrolled students, the number of present ones and the number absent ones, and a subtractor.

6. Design a system to detect number of vans passing by a certain gate. The system consists of a photocell to detect the passing vehicles, a comparator to decide whether that vehicle is a van or not based on its weight on a ground scale, a counter to count only vans, and a register to carry the number of vans

7. Design a digital Stopwatch Circuit LED display. Stopwatch is a timer which is used to measure an event anywhere from milliseconds to several hours. The stopwatch we are going to design is in “MM:SS:mS” (Minutes:Seconds:Milliseconds) format, the maximum time duration it can measure is 99:59:99, next count will be reset “00:00:00”.

8. Design an Automatic Water Level Controller circuit without using any micro-controllers. You have a tank with 2 sensors, low-level sensor, and high-level sensor. Your circuit will trigger the water pump motor to fill up a tank once the water is below the low-level sensor and stop the motor once the water reaches the high-level sensor.

9. Create a four-bit Bidirectional Shift register. A bidirectional, or reversible, shift register is one in which the data can be shifted either left or right. Your circuit should have a display of the current binary number, means to enter data to be shifted, to select shift direction, and to start /stop shifting.

10. The objective is to design and implement a digital circuit that acts like a digital lock for a safe which accepts a binary passcode, compare it to a preset code by a comparator and takes an action depending on the validity of the entered value. If a wrong code is entered for three successive trials, the lock should no longer accept a new value unless a Reset switch is pressed.

11. An arithmetic logic unit (ALU) is a digital circuit able to carry out arithmetic and logical operations between two operands and is considered the main component of a computer processor. The objective of the project is to design and implement a simple 3-bit ALU.

    Required arithmetic operations are Addition and subtraction. On the other hand, required logical operations are: AND, OR and NOT only (Bitwise operation).