CMPE 443 PRINCIPLES OF EMBEDDED SYSTEMS DESIGN

The Final Project

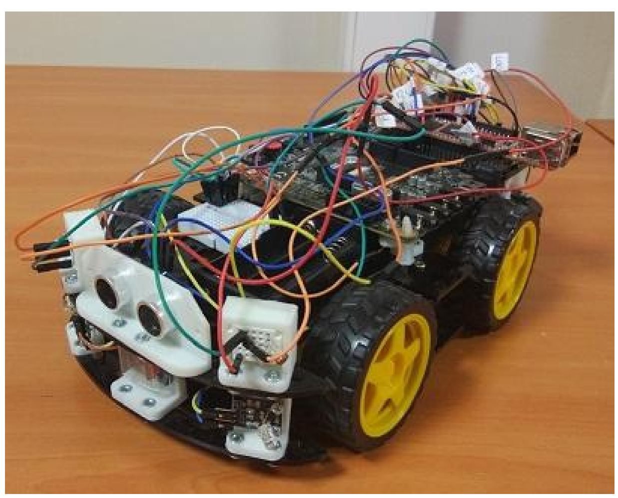
“Autonomous Car”

Motivation

This project is designed to enable the students to put together their learnings throughout CmpE443 to design an embedded system, the Autonomous Car, with its hardware and software components. This Car should support both manual and autonomous modes. The project will consist of three phases: Simulation, Execution and Race. In the Simulation phase, both hardware and software parts will be designed and simulated. The successful students will have the chance to port their designs to execute on the CmpE443 Car. The students who pass the Execution phase will be eligible to enter the Autonomous Car Race, which has been held every year except 2020.

1) Problem Description

The final project consists of simulation and execution phases with two different scenarios. In the simulation phase, you will use **LPC1343FBD48** to complete the tasks which are given for simulation. In the execution phase, you will use an **LPC4088FET208** microcontroller with the car. For the execution phase, a car with the necessary sensors and controllers will be given to you so that you can develop your software at home. However, the race environment will be at HWLAB.



The car contains 1 Motor Controller, 1 Push Button, 1 Joystick, 4 LEDs, 2 LDRs and 1 Ultrasonic Sensor. With these components, you can execute two scenarios described below. These scenarios are Manual and Autonomous. You will switch between the scenarios via the push button (presses and then release).

a. Manual Scenario

In the manual scenario, you will control the car via Joystick, you will change the speed of the car via a global variable and when the Ultrasonic sensor detects an obstacle, the car will stop.

- ➢ When the Joystick Left button is pressed, your robot should start to rotate counter-clockwise direction (Point Turn is necessary)
- ➢ When the Joystick Up button is pressed, your robot should start to travel in the forward direction.
- ➢ When the Joystick Down button is pressed, your robot should start to travel in backward direction.
- ➢ When the Joystick Center button is pressed, your robot should stop.
- ➢ When Joystick Right button is pressed, your robot should rotate in clockwise direction (Point Turn is necessary)

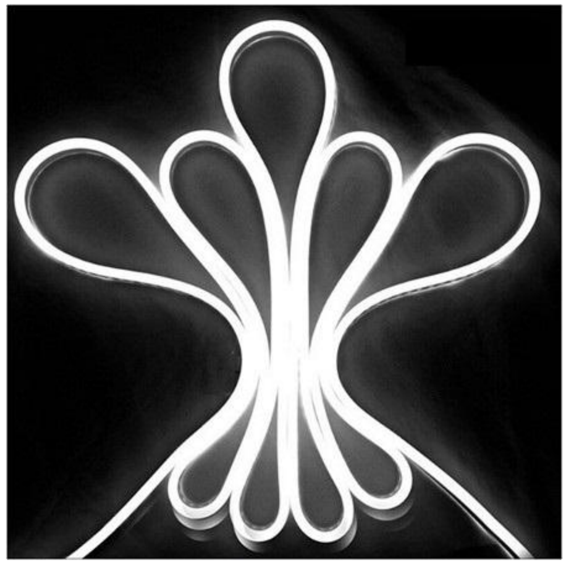
Your robot car has 4 LEDs which are located to the Front-Left, Front-Right, Back-Left and Back-Right The state of the LED is changed according the action which robot performs:

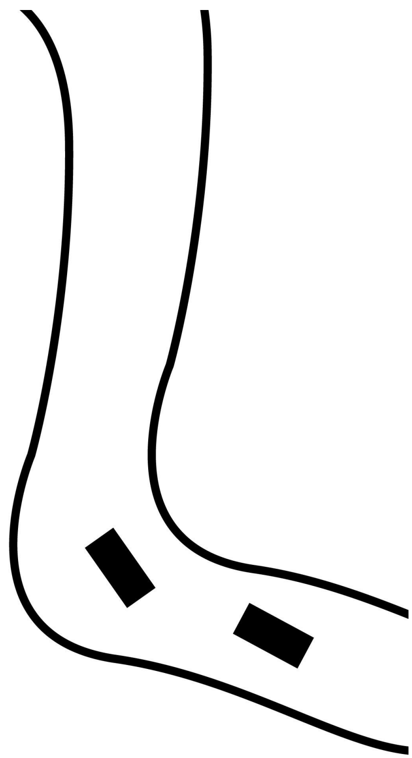
- ➢ When the robot stops, all the LEDs should be turned off.
- ➢ When the robot drives in forward direction, Front-Left and Front-Right LEDs should be turned on and the other LEDs should be turned off.
- ➢ When a robot drives in a backward direction, Back-Left and Back-Right LEDs should be turned on and the other LEDs should be turned off.
- ➢ When the robot rotates counter-clockwise, Front-Left and Back-Left LEDs should blink (2 times in a second) and the other LEDs should be turned off.
- ➢ When the robot rotates clockwise, Front-Right and Back-Right LEDs should blink (2 times in a second) and the other LEDs should be turned off.

When the Ultrasonic sensor detects an obstacle, the car will stop and until a joystick button is pressed, it will not move.

b. Autonomous Scenario

In this scenario, we will put two LED strips on the left and right of the robot.





The robot should follow the road and always be at the center of the road. Also, there can be more than one robot on the road. Therefore, the robot should not hit another robot in front of it. (approximately 10 cm away)

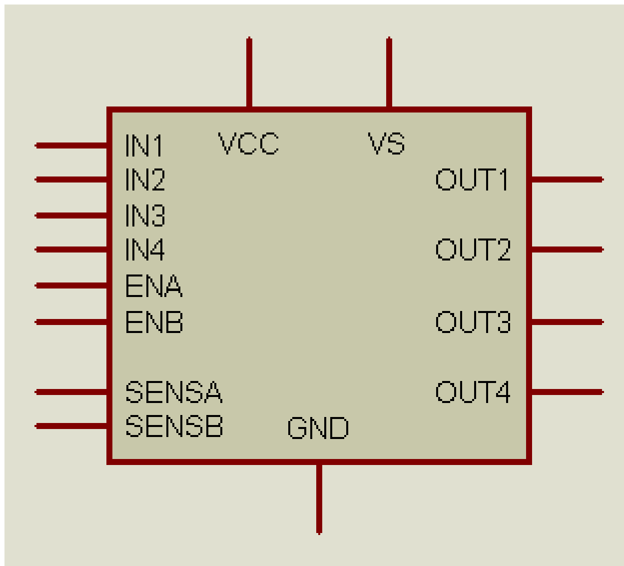
In the Autonomous Scenario, you will still use two cases of the Joystick:

- When the Joystick Up button is pressed, your robot should start to execute Autonomous Scenario
- When the Joystick Center button is pressed, your robot should stop.

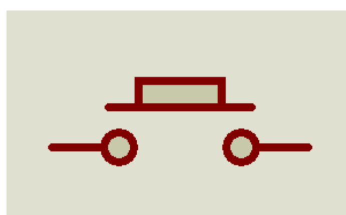
2) Simulation Phase

In the simulation phase, you will use **LPC1343FBD48** with Proteus. You will use 1 Motor Controller, 1 Push Button, 1 Joystick, 4 LEDs, 2 LDRs, and 1 Ultrasonic Sensor in the proteus.

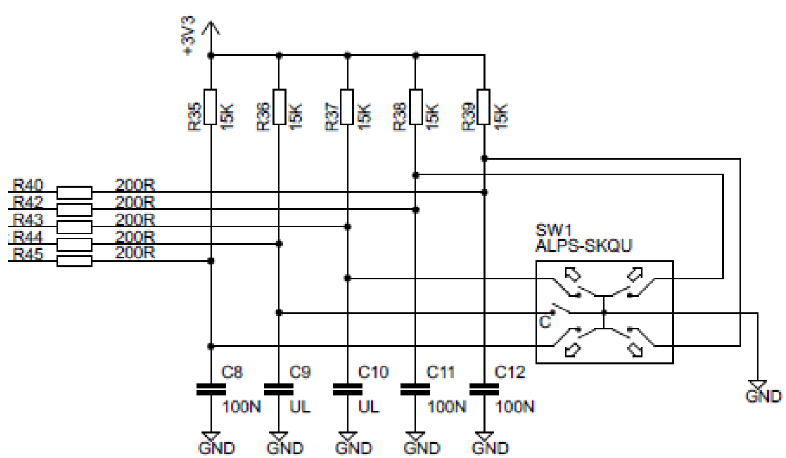
For the motor controller, you will use L298:



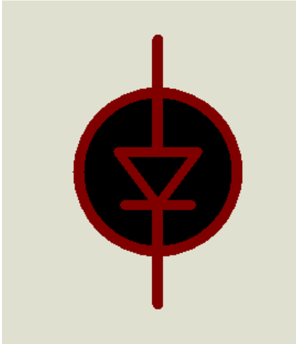
For the Joystick and push button, you will use 5 + 1 push buttons (SPST Push Button). Please see the schematic of LPC4088 QuickStart Experiment Baseboard for related pin connections. A sample view is shown below for your reference:



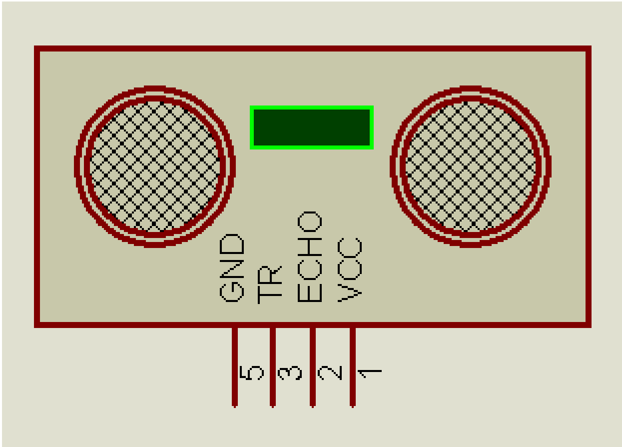
5-key Joystick Switch



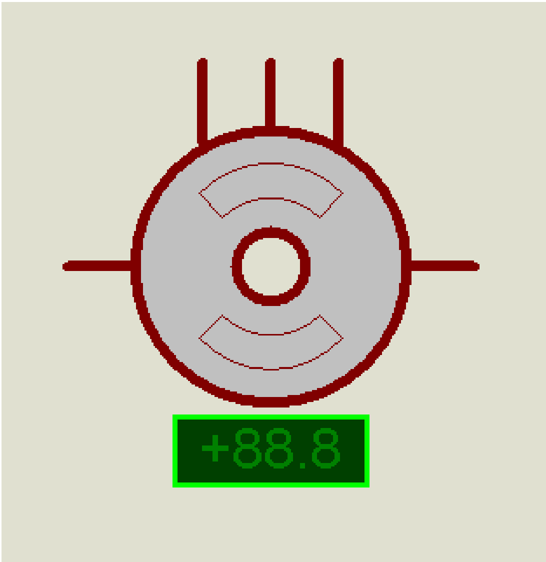
For the LED, you will use 2 Green and 2 Red LED (Animated LED Model):



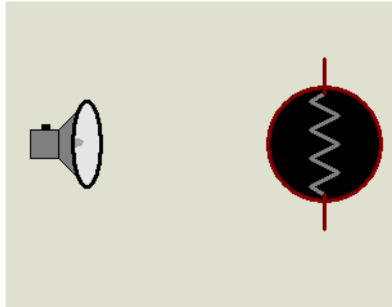
For the ultrasonic sensor, you will use HCSR04:



For the motor, you will use MOTOR-ENCODER:



For the LDRs, you will use Torch LDR:



For the autonomous phase, we want to see changes on the motor behaviour via the changes on the LDR sensors. For example, when LDR values are almost equal, the speed of the motors should be the same, when the left LDR value is more than the right LDR value, the speed of the right motor should be more than the left motor, when the right LDR value is more than the left LDR value, the speed of the left motor should be more than the right motor. (stopping the motor is giving 0 speed and changing the direction of the motor is giving negative speed the motor)

3) Execution Phase

In the execution phase, you will use **LPC4088FET208** with the car. All the necessary components will be available on the car and board. You will use the push button which is on the QuickStart board and you will use the joystick which is on the Experimental Base Board. Ultrasonic Sensor, LDRs, Motor Controller and Motors will be placed on the car.

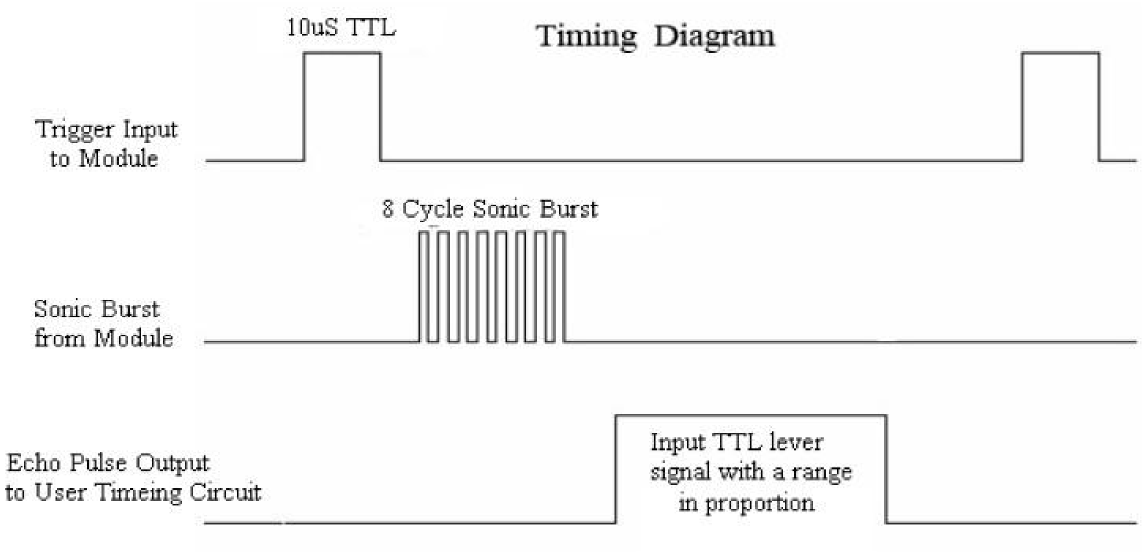
4) Sensors

a. Ultrasonic Sensor

HC - SR04 Ultrasonic sensor (US) has 4 pins which are VCC, Trigger, Echo and GND. User manual of the HC - SR04 is given in Moodle.

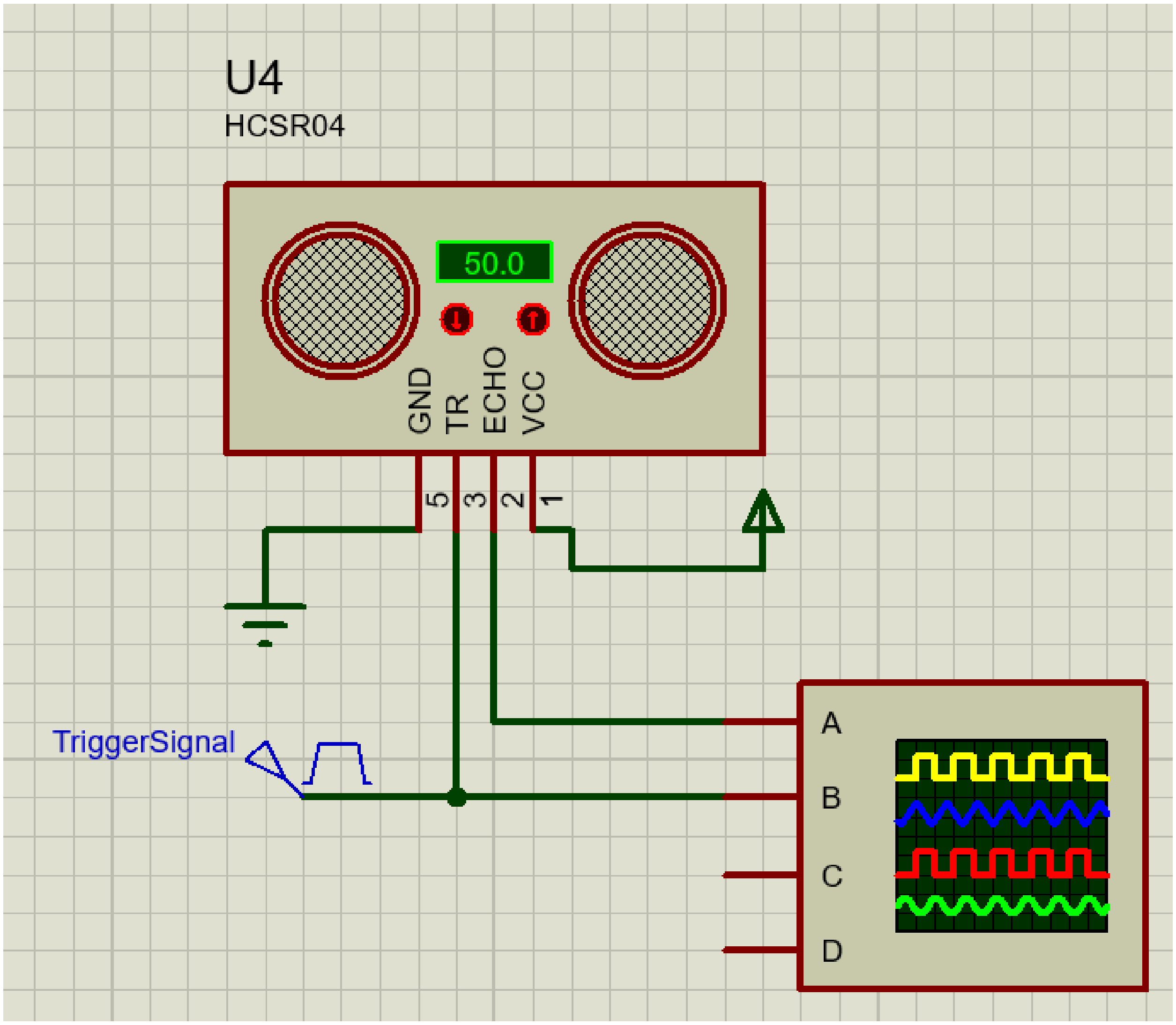
In order to use HC - SR04 Ultrasonic sensor, you should give a trigger signal which is a square signal. After that, the ultrasonic sensor will send 8 cycle bursts of ultrasound at 40 KHz and it will give **HIGH** value to its Echo pin. The Echo signal pulse width is related to the distance of the object:

- If the obstacle moves away, pulse width on Echo will increase.
- If an obstacle approaches, pulse width on Echo will decrease.



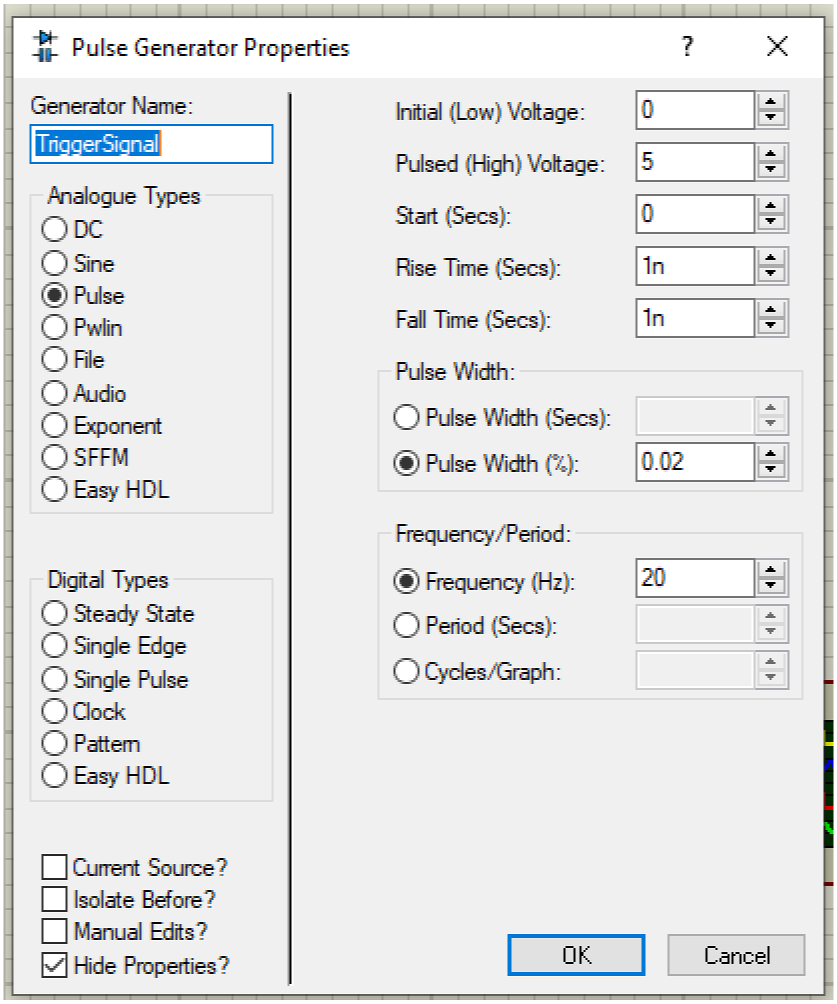
b. Ultrasonic Sensor on Proteus

You can simulate the Ultrasonic Sensor without any code to understand the behavior of the sensor.

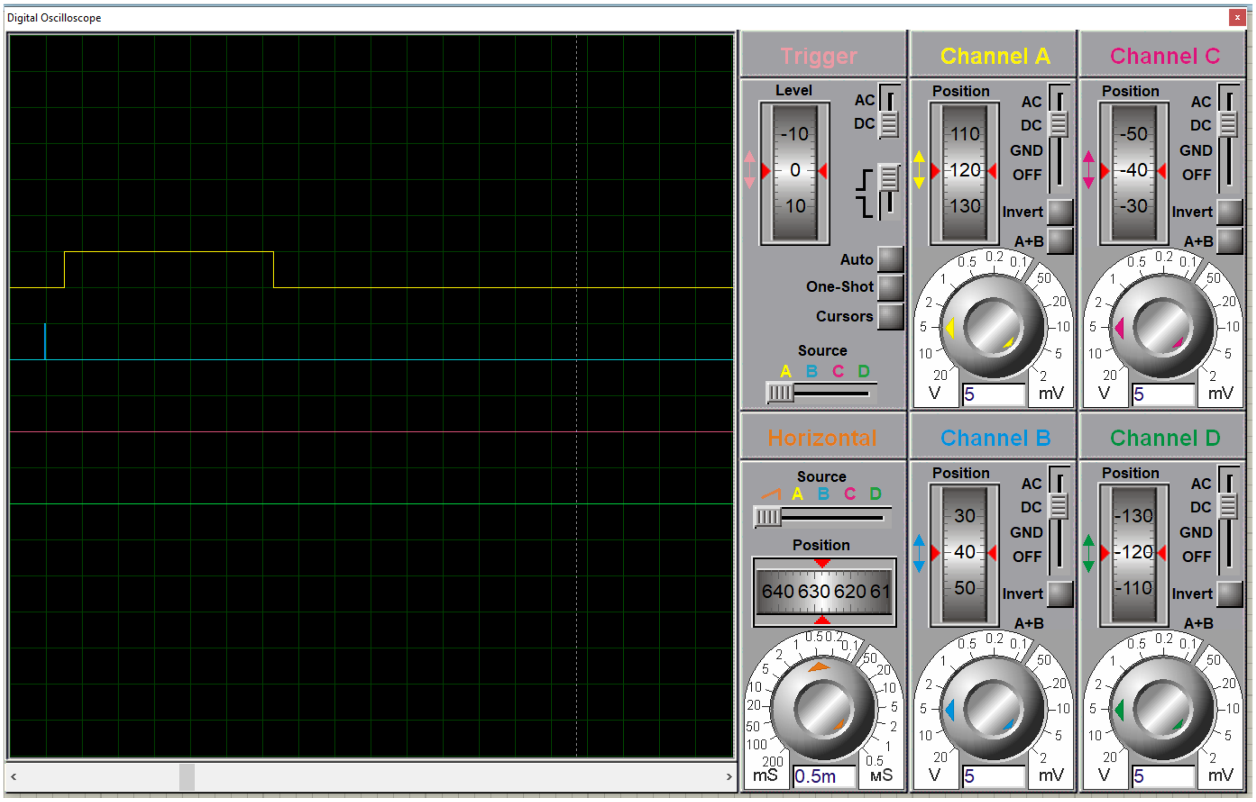


You will use HCSR04 as a sensor. You can give power to the sensor from Terminals - Power.

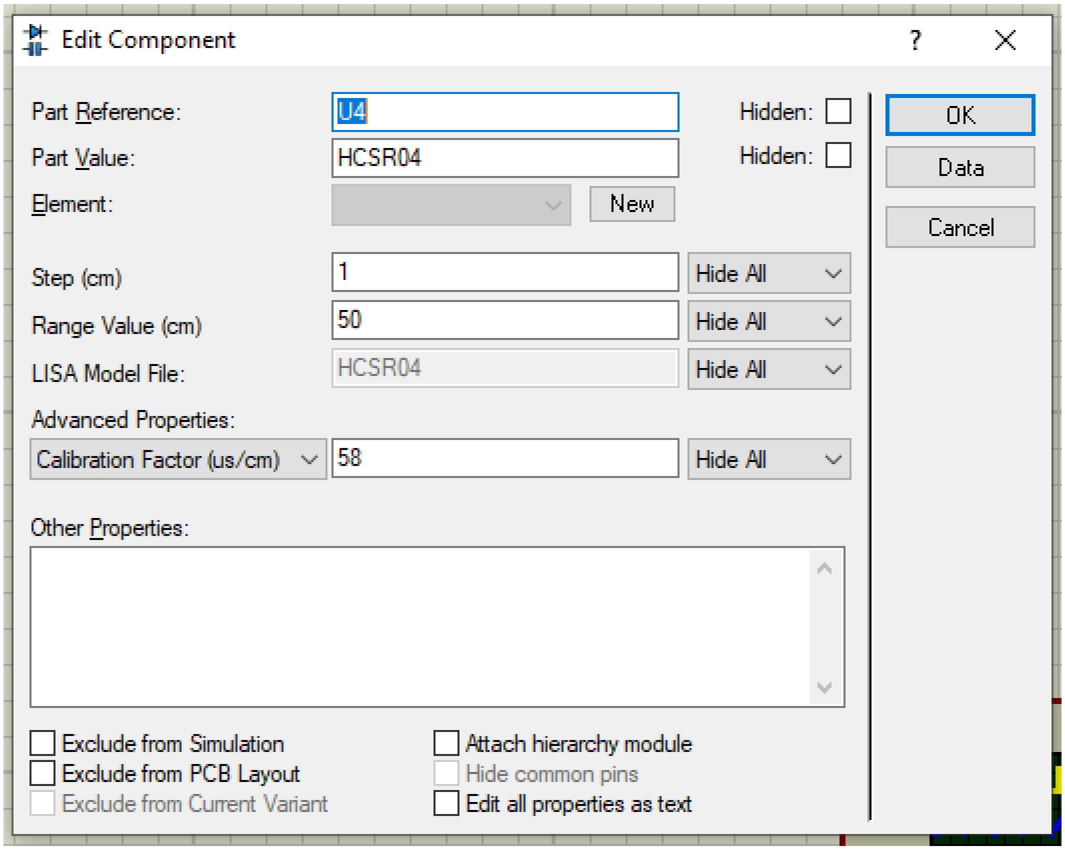
You can configure the Trigger Signal as:



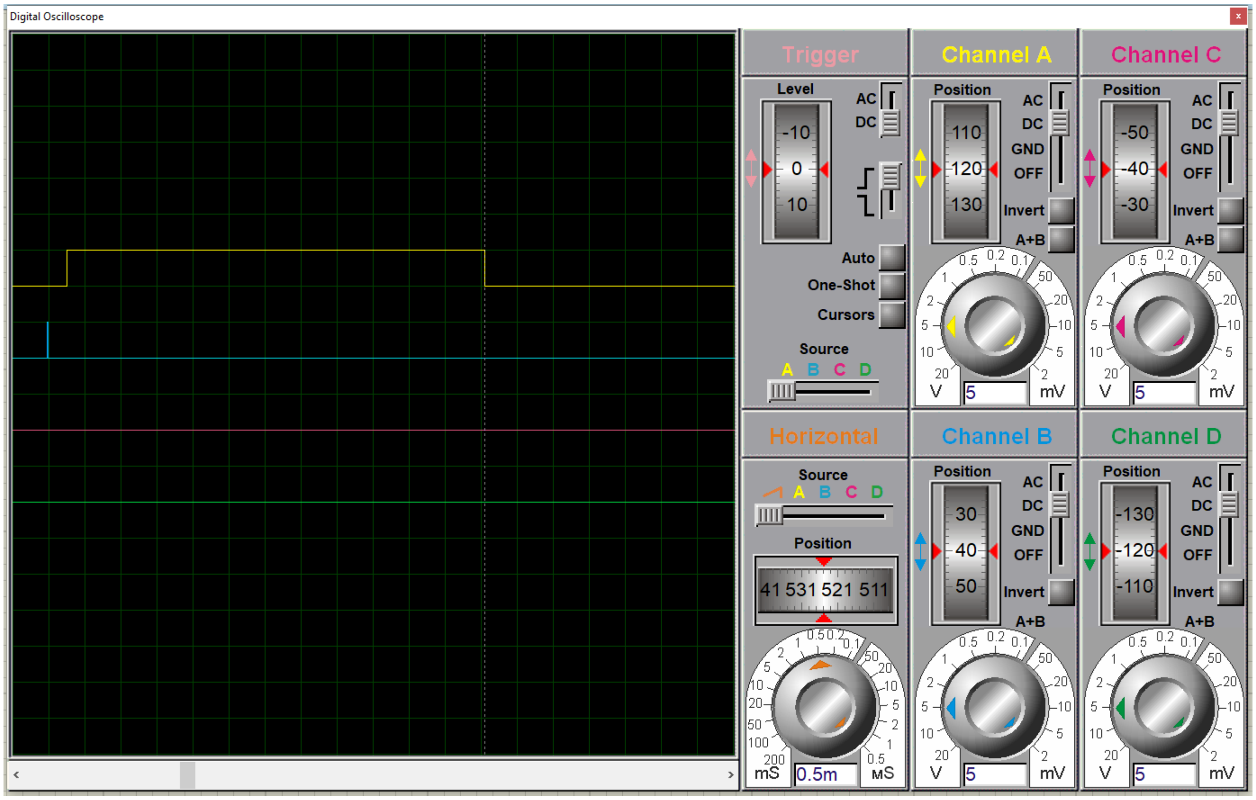
In this oscilloscope display below, you can see the Trigger Signal as blue and Echo signal as yellow (for 50cm):



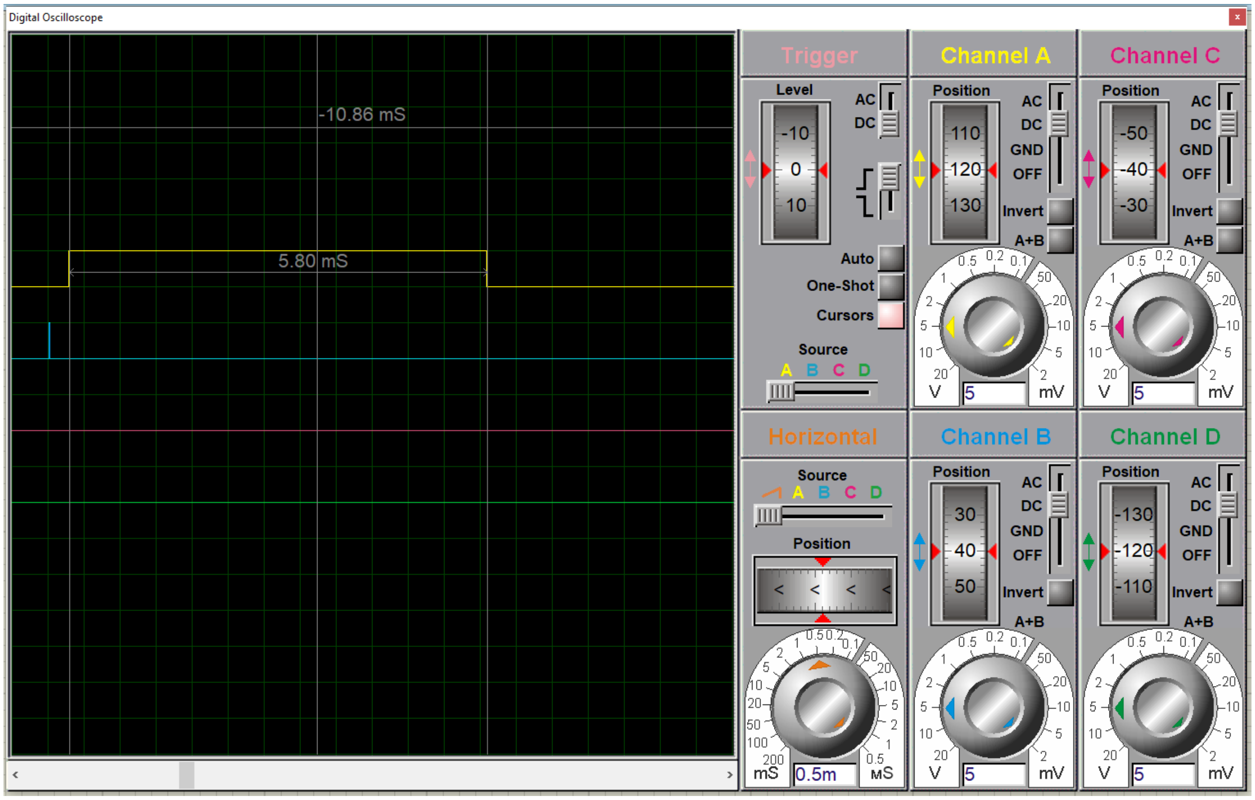
The distance value is shown on the Ultrasonic sensor, you can increase the value from them or you can edit the properties of the sensor to set new distance value by the Range Value:



If you increase the distance to 100cm, you can see that the length of the Echo signal also increases:



Ultrasonic sensor calibration factor is determined with 58 us/cm which is for the traveling speed of sound in the air. You can use a cursor on the oscilloscope. By clicking and dragging you can see the difference between the time. In the shown image, the echo signal length is 5.80ms, which means the Ultrasonic sensor detects an obstacle at 50cm (5800 (us) / 58 (us/cm) = 50 cm) away:



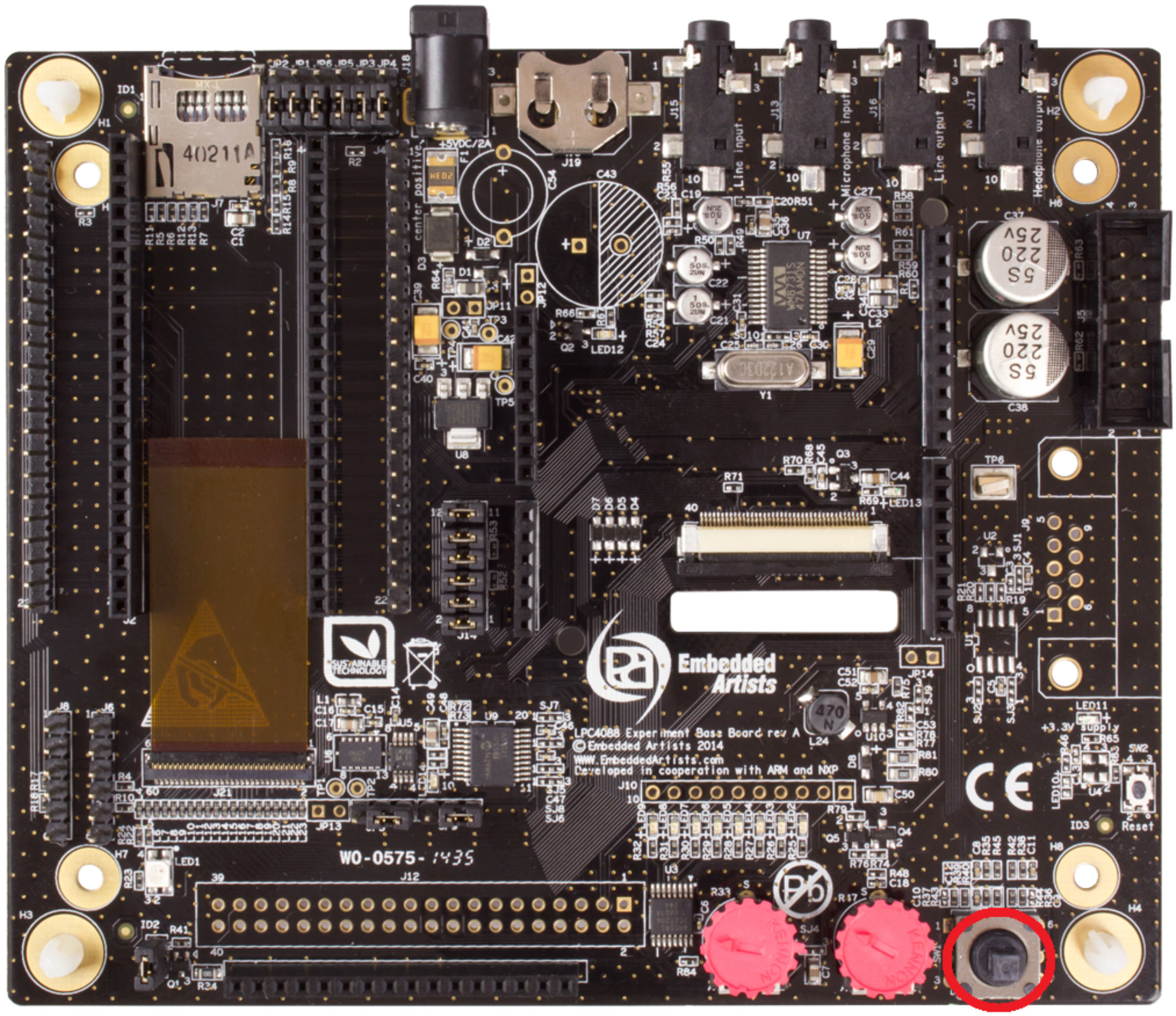
c. Ultrasonic Sensor with Timer

You can generate the Trigger signal with Timer Match and you can capture the echo signal via Timer Capture.

d. Joystick

On the Experiment base board, there is a joystick with 5 push buttons which are Left, Up, Down, Right and Center. You can find which pins are connected to the joystick from the Experimental Board shematic.

In the Proteus environment, you will use push buttons instead of joysticks.



5) Report

Your report must include these sections:

- Structural Block Diagram(s)
- Pin Table(s)
- Circuit Schematic(s) (your drawings on Proteus)
- Shared Variable Table(s)

6) Grading

In order to enter the execution phase, you need to complete the simulation phase with the max score. The simulation phase grade is 60% of the Term Project and execution phase is %40.

For the simulation phase, you need to complete all the tasks with a single Proteus project and a single Keil Projects. The grading of the simulation phase (total 60):

- Report *10 pts*
- Motors should rotate with different RPMs (based on global variable) *5 pts*
- When the Joystick Left button is pressed, the left motor should rotate clockwise direction and the right motor should rotate counter-clockwise direction *2 pts*
- When the Joystick Up button is pressed, the left and right motors should rotate counter-clockwise *2 pts*
- When the Joystick Down button is pressed, the left and right motors should rotate clockwise *2 pts*
- When the Joystick Center button is pressed, motors should stop. *2 pts*
- When Joystick Right button is pressed, the left motor should rotate counter-clockwise direction and the right motor should rotate clockwise direction *2 pts*
- When the robot stops, all the LEDs should be turned off. *2 pts*
- When the robot drives in forward direction, Front-Left and Front-Right LEDs should be turned on and the other LEDs should be turned off. *2 pts*
- When a robot drives in a backward direction, Back-Left and Back-Right LEDs should be turned on and the other LEDs should be turned off. *2 pts*
- When the robot rotates counter-clockwise, Front-Left and Back-Left LEDs should blink (2 times in a second) and the other LEDs should be turned off. *2 pts*
- When the robot rotates clockwise, Front-Right and Back-Right LEDs should blink 3 times in a second) and the other LEDs should be turned off. *2 pts*
- When the Ultrasonic sensor detects an obstacle, the car will stop and until a joystick button is pressed, it will not move. *10 pts*
- Scenario changes via push button *5 pts*
- Motor behaviour changes according to LDR values *10 pts*

The grading of the execution phase (total 40):

- Report *5 pts*
- Motors should rotate with different RPMs (based on global variable) *2.5 pts*
- When the Joystick Left button is pressed, the left motor should rotate clockwise direction and the right motor should rotate counter-clockwise direction *1 pts*
- When the Joystick Up button is pressed, the left and right motors should rotate counter-clockwise *1 pts*
- When the Joystick Down button is pressed, the left and right motors should rotate clockwise *1 pts*
- When the Joystick Center button is pressed, motors should stop. *1 pts*

- When Joystick Right button is pressed, the left motor should rotate counter-clockwise direction and the right motor should rotate clockwise direction *1 pts*
- When the robot stops, all the LEDs should be turned off. *1 pts*
- When the robot drives in forward direction, Front-Left and Front-Right LEDs should be turned on and the other LEDs should be turned off. *1 pts*
- When a robot drives in a backward direction, Back-Left and Back-Right LEDs should be turned on and the other LEDs should be turned off. *1 pts*
- When the robot rotates counter-clockwise, Front-Left and Back-Left LEDs should blink (2 times in a second) and the other LEDs should be turned off. *1 pts*
- When the robot rotates clockwise, Front-Right and Back-Right LEDs should blink (2 times in a second) and the other LEDs should be turned off. *1 pts*
- When the Ultrasonic sensor detects an obstacle, the car will stop and until a joystick button is pressed, it will not move. *5 pts*
- Scenario changes via push button 2.*5 pts*
- Motor behaviour changes according to LDR values *5 pts*
- Completing the parkour *10 pts*

Some Important Notes

- You are only allowed to use peripherals which are shown in the course lecture. If you want to use other types of peripherals, you need to ask for permission.
- The students who successfully complete the Simulation phase will be given the cars. Other students will be graded over their simulation performance, code and reports.
- The tentative deadlines are given below. These deadlines will be discussed and finalized on Jan 6th at 11:00
- Note that there are two submissions for the report and the code. We expect the second submissions to show the modifications that are done on the simulation part.
- The Winner of the race will get 10 pts which will be added on the total course grade.

Tentative Deadlines:

- Simulation Demo and Code Submission: Jan 17, 2021 @ 10:00 - 13:00 on Zoom (link will be provided)
- Submission of Reports: Jan 17, 2021 @ 17:00 on Moodle
- Execution Demo and Code Submission (only for the students who pass the Simulation phase): Jan 27, 2021 @ 11:00 - 14:00
- Race (only for the students who pass the Execution phase): Jan 27, 2021 after the Execution Demo
- Submission of Final Reports: Jan 27, 2021 @ 17:00