Laboratory #3
Basic Robot Navigation Using Sensors

Objective:

- Install wheel encoders and create a test program to verify correct operation
- Use ultrasonic sensor for detecting and avoiding obstacles
- Use light sensors for basic robot navigation

Introduction:

Sensors are essential elements for intelligent operation of autonomous robots. In this week's lab you will be working with wheel encoders, an ultrasonic sensor, and light sensors.

Tasks:

Part I (due at the beginning of 4^{th} week lab period)

1. Install the wheel encoders and collect test data

The basic motors that are currently on your VEX robot are the standard 2-wire Motor 393. This motor does not come with built in encoder. However, VEX sells Integrated Encoder Modules that can be integrated inside the standard motors for measuring the movement of the motors. The Integrated Encoder Module is installed by replacing the plastic cap on your 2-Wire Motor 393. A complete installation instruction is available here:

https://content.vexrobotics.com/docs/inventors-guide/276-1321-INST-0112.pdf

Figure 1: VEX motor encoder modules

Follow the installation instructions to install the encoder modules on both of your VEX motors.

The encoders use I2C communication protocol to communicate with a microcontroller. An Arduino library for accessing the encoder data is available from the following link:

https://github.com/alexhenning/I2CEncoder

Download and install this library into your system. Go through the documentation and example programs to understand how the encoder works and how you can access it in your program.

For this lab exercise, you are required to implement a program that will test your motor encoders and collect speed data that helps you understand the relationship between the motor power and speed for each of the motors. Use the getSpeed() function to obtain the encoder readings for the motor speed values. In your setup() function of your Arduino program use the following code lines to initialize the encoders for the VEX 393 motors in its default high torque mode.

```
leftEncoder.init(MOTOR_393_TORQUE_ROTATIONS, MOTOR_393_TIME_DELTA);

rightEncoder.init(MOTOR_393_TORQUE_ROTATIONS, MOTOR_393_TIME_DELTA);
```

In your test program, collect the motor speed measurements for both left and right motors for motor power settings in the range 0 to 100, with increments of 10.

Tabulate the average speed at each power setting for each of the motors. And create a graph showing the relationship between the motor power (horizontal axis) and the motor speed (vertical axis).

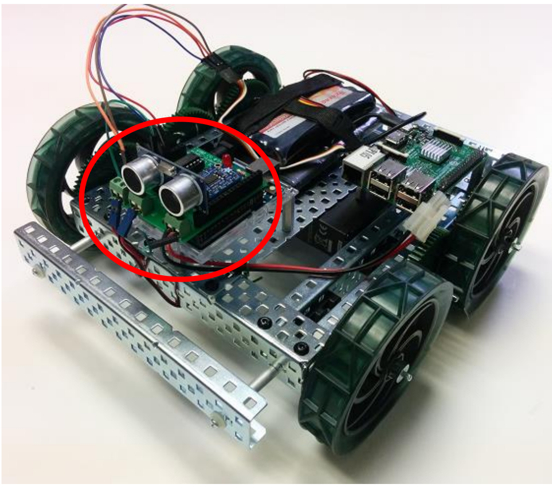
2. Use encoder measurements to control robot motion

Assume that the nominal wheel diameters for each of the wheels in your VEX robot chassis is 10 cm, and the distance between the centers of the left and right wheels is L = 23 cm. The VEX encoder you installed in the previous step has a resolution of 627.2 tick counts per revolution of the wheels. Using this information write a program that drives the robot to navigate in a square path of 4 feet sides. Use power levels of about 50%. If the motors do not behave exactly the same, use the information you obtained in the previous step to adjust the power levels as necessary to achieve the desired trajectory. Your program needs to use information from the two wheel encoders.

3. Use ultrasonic sensor for object detection

The motor driver shield that you used in the last lab assignment has a connector for interfacing an ultrasonic sensor. See Figure 2 below for details.

Mount your HC-SR04 ultrasonic sensor as shown in Figure 2. This sensor has a total of four pins (power, ground, trigger, and echo). The power and ground pins connect to the 5V supply voltage and Ground from the Arduino. The trigger and echo pins are connected to Arduino digital pins 12 and 13, respectively.



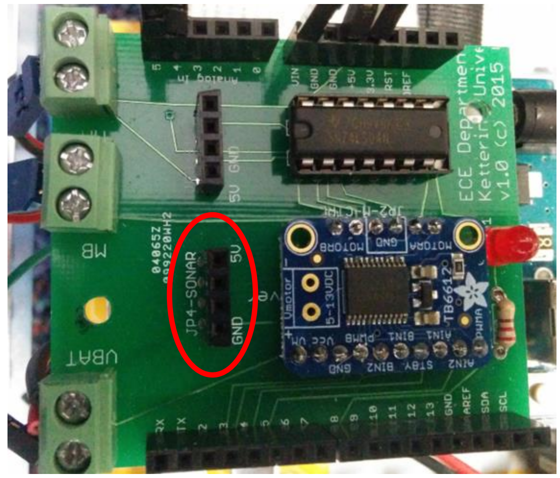


Figure 2: Interfacing ultrasonic sensor

You can learn more about the ultrasonic sensor from the following website:

https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/

Download and install the NewPing Arduino library for the ultrasonic sensor from the link below:

https://playground.arduino.cc/Code/NewPing

Go through the documentation and the example programs to understand how the ultrasonic sensor works.

Write a test program that uses the ultrasonic sensor for detecting objects in front of the robot and prints the distance measurement on to your serial monitor.

4. Obstacle detection and avoidance using the ultrasonic sensors

The way the obstacle detection works is by first establishing a certain threshold distance to keep the robot away from obstacles. Thus, any time the ultrasonic sensors register distances smaller than the threshold due to an object, e.g. a wall, a foot, a cat, etc. on its way then the robot is instructed to take some evasive action to avoid the object.

In this task, when no obstacles are detected on its way the robot drives forward. If it sees an obstacle (i.e., distance below threshold value) by its ultrasonic sensor, it should avoid the obstacle by turning to the left or right by about 90 degrees. Pick your turn directions in a random way.

<u>Part II (due at the beginning of 5th week lab period)</u>

5. Interfacing light sensors to your robot

If you don't have the light sensors mounted on the underside of your robot on its front side, you need to do that as the first step for this task.

The light sensors use photo-resistors that are wired through a voltage division circuit to produce analog output that is proportional to the light intensity detected by the sensor. Therefor you need to use three analog input pins of the Arduino (any of the pins A0 … A5).

Write a program that measures the values from the three light sensors and displays them on the serial monitor.

6. Wander-bot within enclosed environment

Create a program that extends the obstacle avoidance program from task 4 into a wandering robot that stays within an enclosed environment.

The robot will use its ultrasonic sensors to detect and avoid obstacles. While doing that it will use its light sensors to detect boundary lines on the floor and makes sure it does not drive outside its defined environment. Use black electrical tape to create your boundary on the floor for the robot's operating environment, as shown in Figure 3 below.

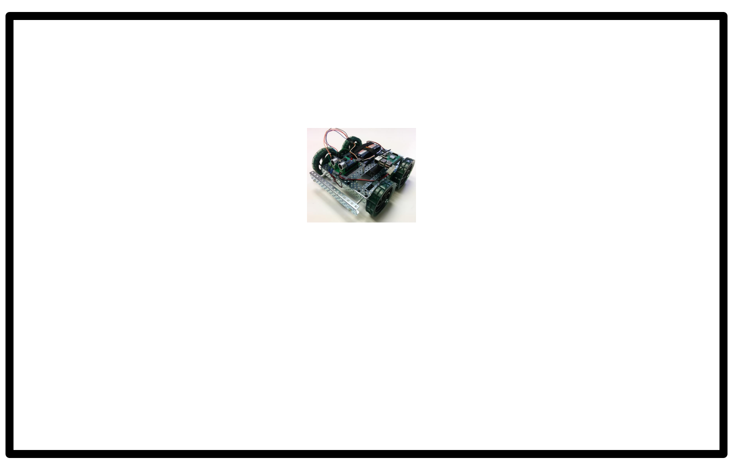


Figure 3: Robot operating zone

Since the light sensors are sensitive to lighting conditions in the environment, it is highly recommended that you incorporate automatic calibration methods in the program that will allow it to calculate the desired threshold value at runtime.

7. Line-following robot

In this task, you are required to implement a program that utilizes the light sensors and generates the necessary motor control commands for the robot so that the robot drives by keeping its middle light sensor on top of the black line on the floor.

8. Line-following robot with safety

In this task, you will use the ultrasonic sensor to add a safety feature for your line following robot you created in task 7. While the robot is following a line, if you detect an obstacle in front of the robot within 20 cm range, the robot must stop and wait until the obstacle is removed from its way. After the obstacle is removed the robot then continues with its line following operation.

Deliverables:

Demonstrate successful completion of each of the eight tasks to your lab instructor and submit your lab report via Blackboard. Your lab report is expected to follow the format and structure guidelines given in "Laboratory and Project Report Guidelines" document which is available on Blackboard. It needs to include the title page, table of contents, objectives, schematic diagrams for the tasks that involve circuit wiring, well commented program source code for each of the tasks that involve programming, and conclusion.