# Lab4: Robot Wall avoidance

## Pre-Requisite Knowledge and Skills

1. Understand basic structure of robot programming
2. Be able to write C code.
3. Be able to compile and upload a code.
4. Be able to move Sparki move forward, backward, and making turns.

## Learning Objective:

1. Understand what the sensor is and how it is implemented.
2. Understand Ultrasonic range sensor and its usage in Sparki.
3. Be able to use Ultrasonic range sensor to let Sparki avoid the wall.

## Recommended Running Environment and Software:

1. Computers Running Windows OS, OSX, or Linux
2. SparkiDuino IDE

## Instructional Material:

1. Sparki Robot
2. Instructions of this activity
3. Sample codes

## Video Demonstration:

1. to be developed

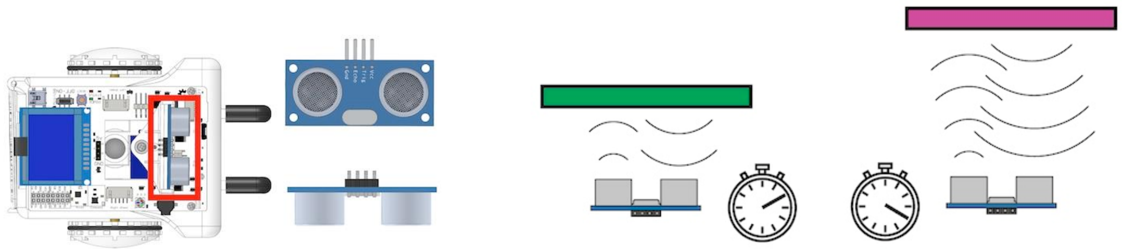
## Lab Assessment:

1. Exercises
2. Quiz

## Lab Instructions

Before lab starts, check the batteries of Sparki and remote controller.

Sparki has one Ultrasonic range sensor in its eye as shown in Figure 1; one is to transmit sound wave where the other receives the sound from reflection. The distance is calculated using the time difference of transmission and reception as shown in Figure 1.



(a) Ultrasonic range finder sensor (b) distance calculation using time difference

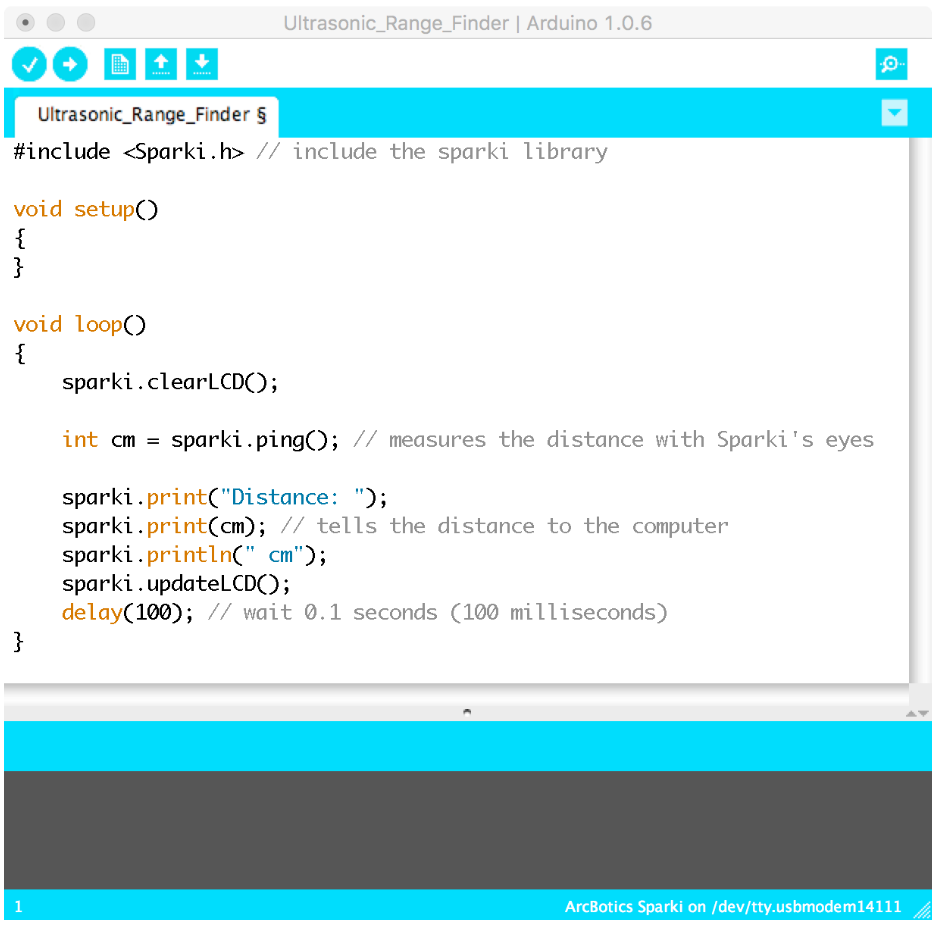
**Figure 1.** Ultrasonic range finder sensor and distance calculation.

The formula that is used in distance calculation is described in below.

Distance = Speed * Travel_Time

**Step 1:** Answer following question.
Assume that the round trip time detected at Sparki is about 0.01 second where the speed of sound is 343 m/s. What is the measured distance of the object from the Sparki in meter?

**Step 2:** Now let's learn how to use Ultrasonic range sensor. Write a code as shown in Figure 2.


Ultrasonic_Range_Finder | Arduino 1.0.6
Ultrasonic_Range_Finder §
1
ArcBotics Sparki on /dev/tty.usbmodem14111

```
#include <Sparki.h> // include the sparki library

void setup()
{
}

void loop()
{
    sparki.clearLCD();

    int cm = sparki.ping(); // measures the distance with Sparki's eyes

    sparki.print("Distance: ");
    sparki.print(cm); // tells the distance to the computer
    sparki.println(" cm");
    sparki.updateLCD();
    delay(100); // wait 0.1 seconds (100 milliseconds)
}
```

**Figure 2.** Ultrasonic range finder sensor code

**Step 3:** Compile and upload the code. Then, disconnect the cable and turn the Sparki on. Place any object in front of Sparki and move it back and forth. Observe the number change in LCD monitor.

**Step 4:** Move the object to left or right side and observe the number displayed in LCD monitor. Locate the position when display shows "-1" (i.e., meaning "error").

**Step 5:** Find the angle from the Sparki to the right and left that Sparki can detect an object.

**Step 6:** Use following code shown in Figure 3 to let Sparki avoid the wall. When Sparki gets closer than 15 cm, it moves backward by 5 cm and turn to the right in 45 degree and keeps moving forward. Find the x, y, and z value in the code.

```
#include <Sparki.h>                              // include the sparki library

void setup() {
        sparki.servo(SERVO_CENTER);      // Center the Servo
}

void loop() {
        sparki.moveForward();   // move Sparki forward
        int  cm =  sparki.ping();  // measures the distance with Sparki's eyes

        if(cm  ??  x)                    // if the distance measured is less than x
cm
        {
                        sparki.RGB(RGB_RED);         // turn the light red
            sparki.beep();                        // beep!
            sparki.moveBackward(y);         // back up y centimeters
            sparki.moveRight(z);                          // rotate right z degrees
        }
        delay(100);                      // wait 0.1 seconds (100 milliseconds)
}
```

**Figure 3.** Sparki move forward and backward code

There are several cases that Ultrasonic range sensor fails to measure the distance such as out of range, no reflection, too far, or tilted reflection, as shown in Figure 4.

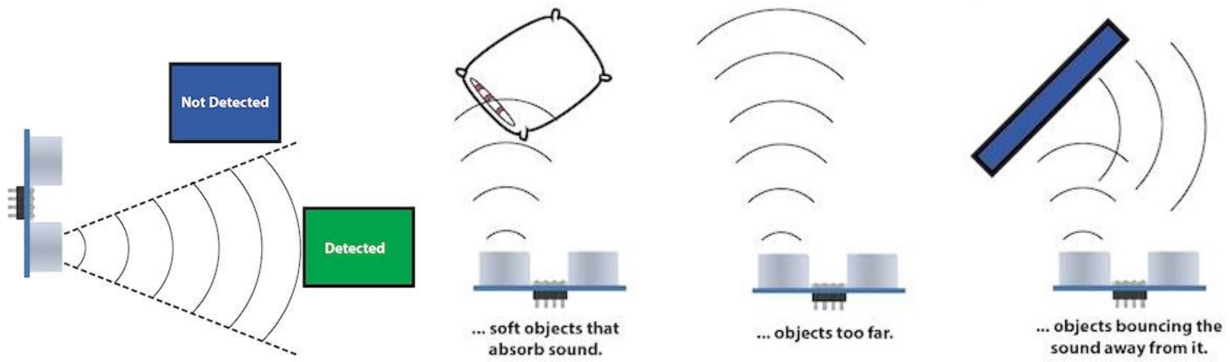

Not Detected
Detected
... soft objects that absorb sound.
... objects too far.
... objects bouncing the sound away from it.

**Figure 4.** Cases that Ultrasonic range finder sensor fails to measure

**Step 7:** Use code in Figure 2 and test the Ultrasonic range finder sensor for each cases in Figure 4.

<u>**Discussion**</u>

- **Code in Figure 3 has one command line in Setup() function. What is the purpose of**

**this?**

sparki.servo(SERVO_CENTER) will set the Sparki's eye to center so that it can detect the object in front.

- **Code in Figure 3 has a vulnerability that Spark may act as if there is an object even if there is no object in front. Which case may it occur?**

  If the object is too close or too far away. "sparki.ping()" returns "-1" as an error. Since "-1" is smaller than desired value "x", so it will react as if there is a object within a range.

- **How this vulnerability can be protected?**

  You may want to add more layer to check the reading – more numbers of protections. See sample code.