FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Licenciatura em Engenharia Informática

TECNOLOGIA DOS COMPUTADORES Year 2018/2019

Laboratorial assignment # 4

Distance sensor using time-of-fly

Components list:

- Arduino UNO
- 1 USB cable
- 1 white breadboard
- LED’s and resistors
- Ultrasonic sensor
- Wires
- Software: Arduino IDE
-

1. Measure distance

The HC-SR04 ultrasonic sensor (also called proximity sensor) measures the time interval between the transmission of a sound pulse and its echo. Since the velocity of the sound is known, one can convert this time interval into a distance. The sensor has four pins that have the following functions:

• GND (0V)	• Echo – pin to receive the echo signal
• Vcc (5V)	• Trigger – pin to send a pulse

When the Arduino sends a pulse of 10 microseconds to the trigger pin, the ultrasonic sensor produces a pulse and then waits for its echo. The echo is waited at the echo pin. The distance in centimeters is computed by dividing the interval time (measured by the function **`pulseIn()`**) by 58. The Arduino should wait at least 60ms until the next measure of distance. The code in figure 1 illustrates an example function to measure the distance using an ultrasonic sensor. Analyze the datasheet available at Nónio for more information.

2. Serial communication

The serial port of the Arduino is able to communicate with the computer to send and receive data. The operation of this serial communication is made using the following functions:

- **`Serial.begin(baud_rate);`** - indicates to the Arduino the velocity used to communicate. It should be set in the **`setup()`** function.

- **`Serial.print(value);`** - send the **`value`** through the serial port. It is able to send variables in several formats, including characters or strings.
- **`Serial.println(value);`** - send the **`value`** through the serial port and change the line (introduces an '\n' at the end).

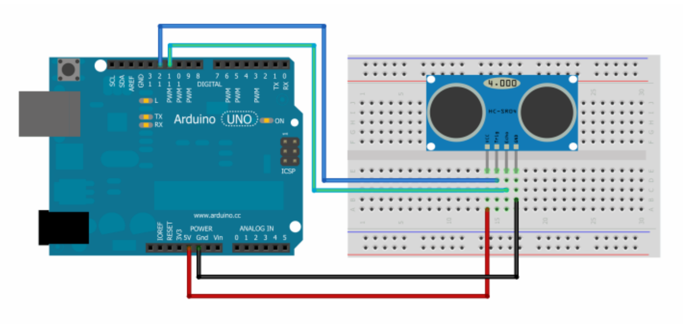

Fig. 1 – Circuit with an ultrasonic sensor

3. Work

3.1 - Mount the ultrasonic sensor as explained above and send the distance value to the computer by the serial port. Use the Serial Monitor (Tools → Serial Monitor) to observe the values sent through the port.

3.2 – Mount a scale of 5 LEDs (Green, Green, Yellow, Yellow, Red) to be lit depending on the distance to the objects. If distance is lower than 40cm all LEDs should be turned off. If distance is between 40cm and 80cm, then the leftmost Green LED should be turned on. This operation mode should be replicated to the remaining LEDs in steps of 40cms. Notice that when the measured distance is 2 meters or higher, all LEDs are turned on.

3.3 – Produce an alert message and send it through the serial port each time the red LED is turned on.

3.4 – Use the mounted circuit to measure the size of one object at your choice.

3.5 - Use the mounted circuit to measure the height of your group colleague and the height of the ceiling.

```
#define trigPin 13
#define echoPin 12

long measureDistance() {
  long duration, distance;

  //Send pulse
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  //Wait for echo and measure time until it happens
  duration = pulseIn(echoPin, HIGH);

  //Compute distance
  distance = duration/58;

  return distance;
}
void setup() {
  Serial.begin (9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  ...
}
...
```