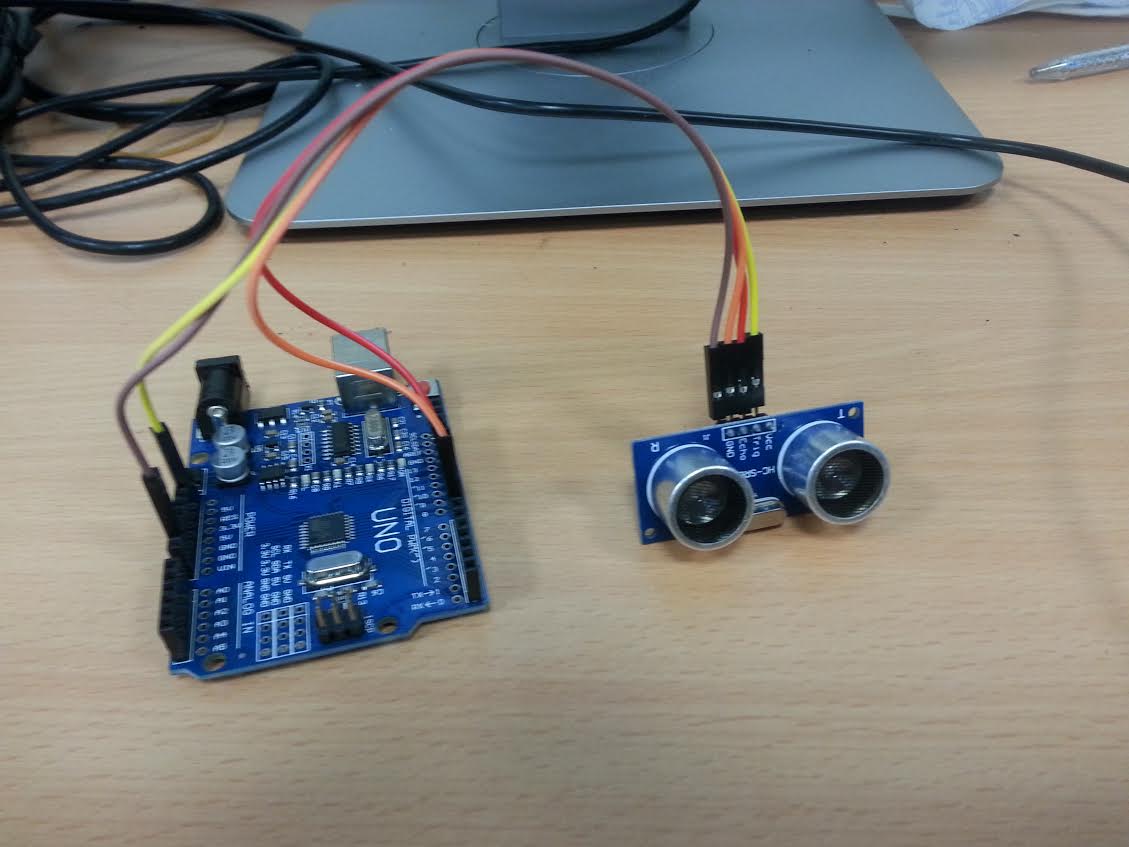
## Ultrasonic Sensor

The ultrasonic sensor is more complicated than reading a binary value. It not only must be driven using a trigger pin, but it’s output comes in the form of a measurement of how long it drives one of the pins High. The wiring can be seen in the picture below.

Here, we connected: Vcc to 5v and GND to GND of course, but we also connect the Trig pin from the module to the Arduino digital 8 pin and the echo pin from the module to the Arduino’s digital 7 pin. The Trig pin is an output pin, so we will be writing HIGH or LOW to it in order to initiate protocol with the module. The echo pin is an input pin, we will measure how long it remains high, and this length can be converted to cm in order to give us the distance of objects from the module, in centimetres.

The code is shown in the figure below. As stated, the trigger pin is pin 7 and the echo pin is pin 8. Here we define a minimum and a maximum range from 0 to 200cm. We also declare two variables, distance and duration. We will write the length of time in which the echo pin remains high into the duration variable, and the distance variable will be the object’s distance in cm.

As usual in our setup function, we initialize our Serial communication object, and of course set the pin modes using the pinMode function, trigPin is an output and echoPin is an input.

To begin communication with the module, we drive trigPin LOW for 2 microseconds using the digitalWrite function and the delayMicroseconds function. The delayMicroseconds delays by the amount of microseconds passed as the first argument. After this we drive trigPin HIGH for 10 microseconds using the same strategy.

Now we begin to measure the distance our echo pin remains HIGH. First, we drive trigPin LOW, Next we use a function called pulseIn(x,y). The pin measures the time the x pin remains in state y. In our case we measure the duration for which echoPin remains HIGH. Using our duration variable, we then convert this to cm. If our distance is in the range we declared earlier, then we write it to the Serial Monitor.

/\*

VCC to arduino 5v

GND to arduino GND

Echo to Arduino pin 7

Trig to Arduino pin 8

\*/

#define echoPin 7 // Echo Pin

#define trigPin 8 // Trigger Pin

int maximumRange = 200; // Maximum range needed

int minimumRange = 0; // Minimum range needed

float duration, distance; // Duration used to calculate distance

void setup() {

Serial.begin (9600);

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

}

void loop() {

//initialize communication protocol, using the trigger pin

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

//bring the pin low

digitalWrite(trigPin, LOW);

//pulse in records how long the echoPin remains high to obtain the input

duration = (float)pulseIn(echoPin, HIGH);

//Compute the distance based on the duration

distance = duration/58.2;

if (distance <= maximumRange && distance > minimumRange) //if not in range

Serial.println("object detected at: " + String(distance) + " cm.");

delay(100);

}

void loop(void)

{

//request temperature

sensor.requestTemperatures();

Serial.print("Temperature := ");

Serial.println(sensor.getTempCByIndex(0));

delay(100); //delay

}