UNITED INTERNATIONAL UNIVERSITY

Lab Report 4

Course Name: Microprocessors and Microcontrollers Laboratory

Course Code: CSE 4326

Section: D

Group Members:

Abdulla Al Samir	011192039
Md Abdul Quddus Sagor	011192047

Prepared by:

Md Abdul Quddus Sagor	011192047

Submitted to:

Sumaiya Jahan Tabassum

Lecturer, United International University

Submission Date: 09/07/2023

Experiment no. 5

Conditional Ultrasonic Distance Measurement and RF Transmission

Objective:

You are an adventurous person and have gone to find a rare species of spider in the Amazons. Whenever you see a spider from a distance, you push a button that activates an ultrasonic sensor, which then measures the distance between you and the spider. However, the spider is venomous and you don't want it getting too close to you. If it is closer than 500 cm to you, you need to signal your friend via RF transmission that you are in danger by saying "The spider is ** cm close to me. Help!". The ultrasonic sensor and push button are connected to the master Arduino. Connect a serial monitor to the slave Arduino to check whether the sent message is received properly.

Equipment:

- Arduino Uno R3
- Arduino IDE (Compiler)
- Proteus (Simulator)

Introduction:

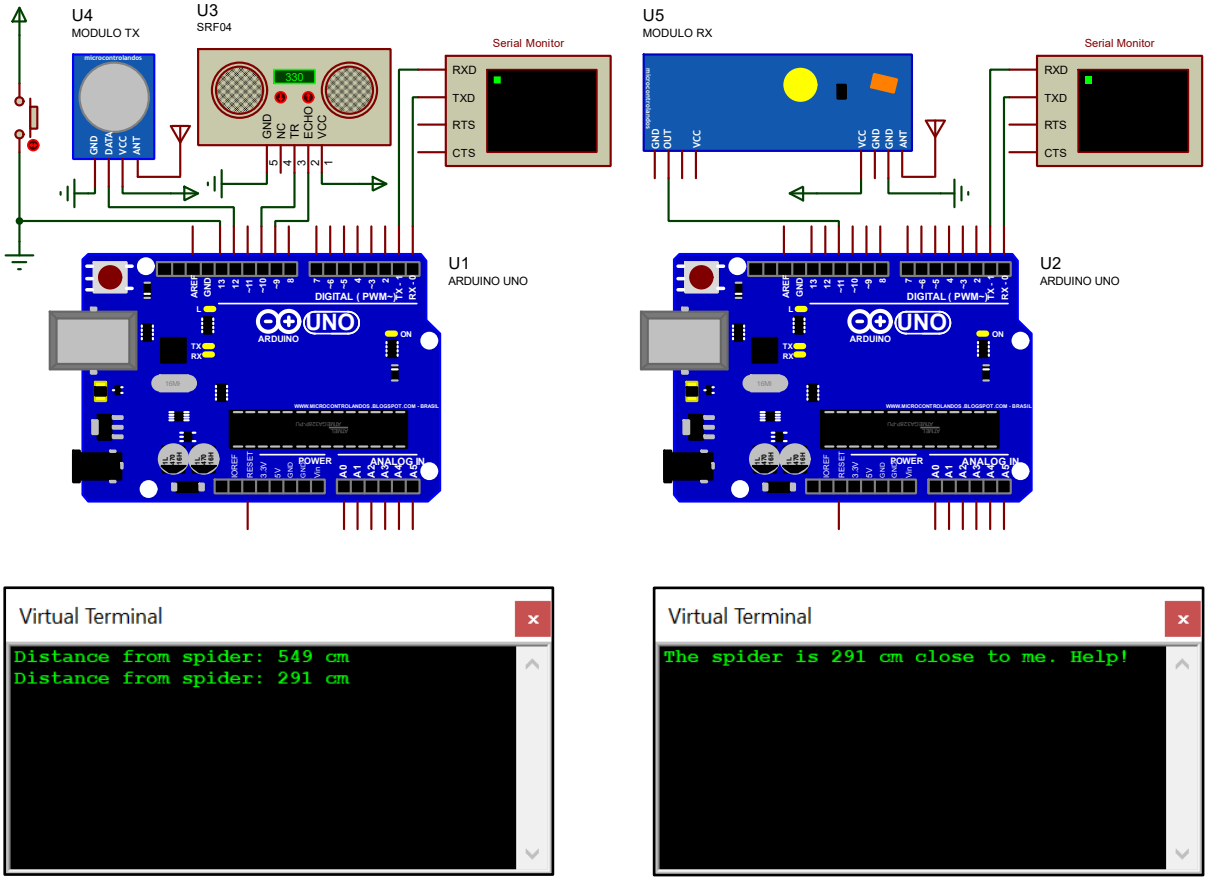
This experiment combines the capabilities of an ultrasonic sensor and an RF transmitter-receiver pair to enable selective distance measurement and wireless data transmission. Traditionally, ultrasonic sensors measure distances continuously, which may not always be desired. In this experiment, the ultrasonic sensor is only activated when a push button is pressed, allowing for greater control over when distance measurements are taken. The measured distance is then transmitted wirelessly using the RF transmitter-receiver pair, enabling remote monitoring and data transfer. This experiment offers flexibility and interactivity by providing the ability to choose when to activate the ultrasonic sensor and transmit the distance data.

Components and its Functions:

1. **Arduino Uno:** The Arduino board serves as the main control unit for the experiment, responsible for running the code, processing sensor data, and transmitting/receiving data using the RF modules.

2. **Ultrasonic Sensor:** The ultrasonic sensor is used to measure the distance to an object. It emits ultrasonic pulses and measures the time it takes for the pulses to bounce back after hitting an object. This information is used to calculate the distance.
3. **Push Button:** The push button serves as a manual trigger for activating the ultrasonic sensor. When pressed, it enables the ultrasonic sensor to measure the distance, allowing the user to control when measurements are taken.
4. **RF Transmitter Module:** The RF transmitter module is responsible for wirelessly transmitting data from the Arduino to the RF receiver module. It converts the data into radio frequency signals that can be transmitted over the air.
5. **RF Receiver Module:** The RF receiver module receives the transmitted data from the RF transmitter module. It converts the received radio frequency signals back into digital data, which can be processed by the Arduino.
6. **Transmitter Serial Monitor:** The Serial Monitor on the transmitter Arduino is used to display the distance measurements obtained from the ultrasonic sensor. It shows the measured distance in centimeters when the ultrasonic sensor is active.
7. **Receiver Serial Monitor:** The Serial Monitor on the receiver Arduino is utilized to display the received data from the RF receiver module. It shows the transmitted distance data received wirelessly from the transmitter Arduino. This allows for monitoring and verification of the successful transmission and reception of the distance measurements.

Circuit Diagram:



Transmitter Arduino Code:

```
#include <VirtualWire.h>

char msg[4];
int buttonPin = 13;

void setup() {
  Serial.begin(9600);
  vw_set_ptt_inverted(true);
  vw_setup(2000);
  vw_set_tx_pin(12);
  pinMode(buttonPin, INPUT_PULLUP);
  pinMode(10, OUTPUT);
  pinMode(9, INPUT);
}

void loop() {
  if (digitalRead(buttonPin) == HIGH) {
    digitalWrite(10, LOW);
    delayMicroseconds(10);
    digitalWrite(10, HIGH);
    delayMicroseconds(10);
    digitalWrite(10, LOW);

    long duration = pulseIn(9, HIGH);
    long distance = (duration * 0.034) / 2;

    Serial.print("Distance from spider: ");
    Serial.print(distance);
    Serial.println(" cm");
    delay(100);

    if (distance <= 500) {
      itoa(distance, msg, 10);
      vw_send((uint8_t*)msg, sizeof(distance));
      delay(100);
    }
  }
}
```

Receiver Arduino Code:

```
#include <VirtualWire.h>

int i;

void setup() {
  Serial.begin(9600);
  vw_set_ptt_inverted(true);
  vw_setup(2000);
  vw_set_rx_pin(11);
  vw_rx_start();
}

void loop() {
  uint8_t buf[VW_MAX_MESSAGE_LEN];
  uint8_t buflen = VW_MAX_MESSAGE_LEN;

  if (vw_get_message(buf, &buflen)) {
    Serial.print("The spider is ");

    for (i = 0; i < buflen; i++) {
      char ch = buf[i];
      Serial.print(ch);
    }

    Serial.println(" cm close to me. Help!");
  }
}
```