



# Faculty of Computing

Year 1 Semester 1 (2024)

IT1140 – Fundamental of Computing

Lab Sheet 09

---

## Objectives:

This practical provides a guidance to the students to develop a visualization dashboard to monitor the behavior of the distance factor of the Ultrasonic sensor.

## Activity

Design the circuit diagram given in figure 9.1 in Wokwi. Use the below items to design your circuit diagram.

- ESP32
  - LEDs - 2
  - Ultrasonic Distance Sensor
- a) Connect the components correctly and design the circuit as shown in figure 9.1.
- a. Connect the PIN\_TRIG of the sensor to the ESP:5
  - b. Connect the PIN\_ECHO of the sensor to the ESP:17
  - c. Connect the red colour LED (Low\_LED\_Pin) to ESP: 12
  - d. Connect the green colour LED (High\_LED\_Pin) to ESP: 14

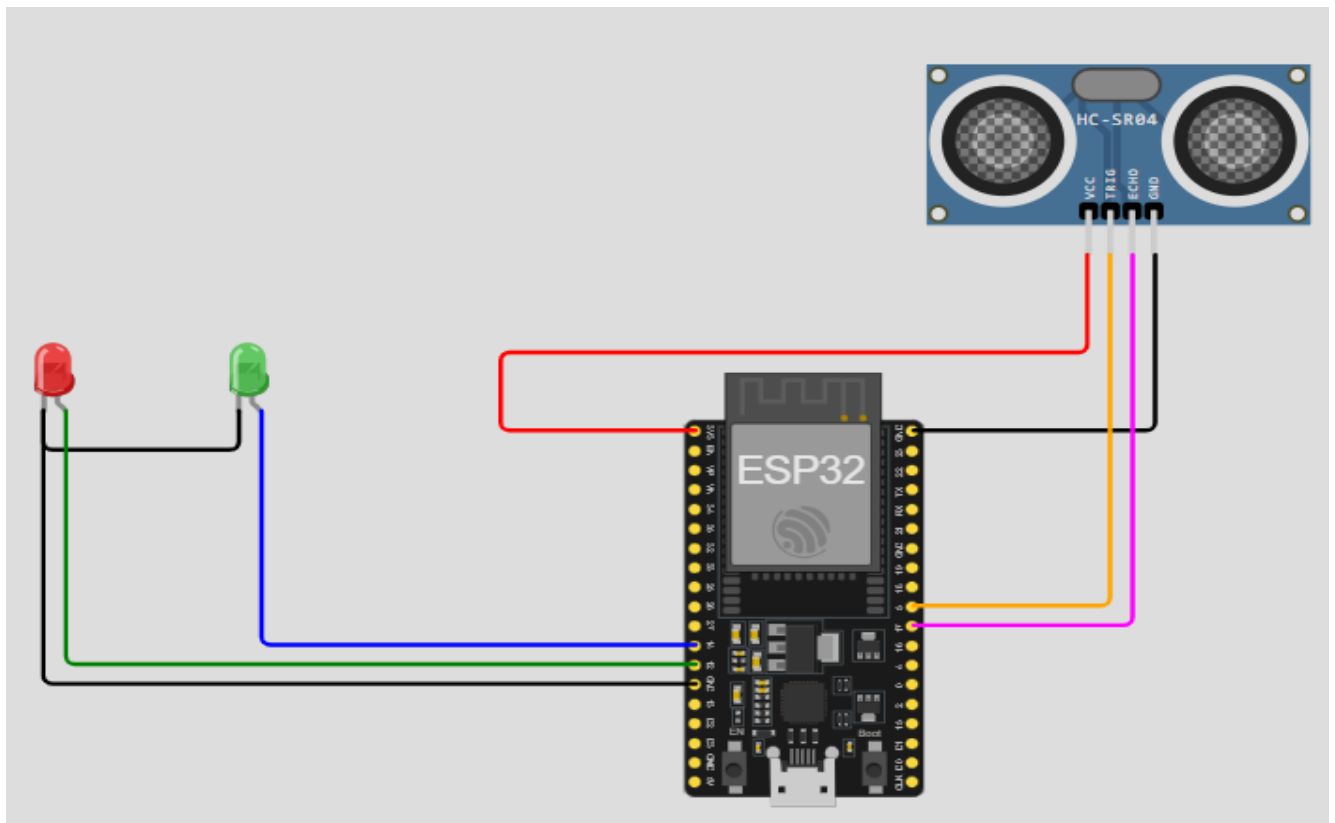


Figure 9.1

- a) Add the following libraries in your code from the Library Manager.
  - a. WiFi
  - b. SimpleWiFiClient
  - c. ThingSpeak

In this activity, the circuit will measure the distance using the sensor and connects with the ThingSpeak to display the measurements to monitor the functionality of the circuit. The ESP32 provides the Wifi connectivity so, that the Wokwi simulator will connect with the ThingSpeak visualizer.

- b) Add the following C++ code to your program.

```
#include <WiFi.h>
#include <WiFiClient.h>
#include "ThingSpeak.h"
```

Include header files

```
#define Low_LED_Pin 12
#define High_LED_Pin 14
#define PIN_TRIG 5
#define PIN_ECHO 17
```

Connect the wires

```
const char* WIFI_NAME = "Wokwi-GUEST";
const char* WIFI_PASSWORD = "";
const int myChannelNumber = 2682285 ;
const char* myApiKey = "SE6MI04FDX1CPS2N";
const char* server = "api.thingspeak.com";
```

Configure the connection with  
thingspeak over wifi

ChannelNumber and the  
ApiKey are subject to change

```
WiFiClient client;
```

```
void setup() {
  Serial.begin(115200);
  pinMode(Low_LED_Pin, OUTPUT);
  pinMode(High_LED_Pin, OUTPUT);
  pinMode(PIN_TRIG, OUTPUT);
  pinMode(PIN_ECHO, INPUT);
  WiFi.begin(WIFI_NAME, WIFI_PASSWORD);
  ThingSpeak.begin(client);
}
```

setup

```

void loop() {
  digitalWrite(PIN_TRIG, HIGH);
  delayMicroseconds(10);
  digitalWrite(PIN_TRIG, LOW);

  int duration = pulseIn(PIN_ECHO, HIGH);
  int distance = duration / 58;
  Serial.print("Distance in CM: ");
  Serial.println(distance);
  ThingSpeak.setField(1,distance);
  ThingSpeak.writeFields(myChannelNumber,myApiKey);
}

```

Calculate the distance and pass to the dashboard

```

if (distance < 20) {
  digitalWrite(High_LED_Pin, HIGH);
  digitalWrite(Low_LED_Pin, LOW);
}
else if (distance >300){
  digitalWrite(Low_LED_Pin, HIGH);
  digitalWrite(High_LED_Pin, LOW);
}
else{
  digitalWrite(High_LED_Pin, LOW);
  digitalWrite(Low_LED_Pin, LOW);
}
delay(1000);
}

```

Control the circuit using the distance measurement

c) Load the ThingSpeak page using the link (<https://thingspeak.mathworks.com/>) and create a free account for you. (Follow the steps below)

- Click on “Create One” and create a new account by providing the details.
- Goto the email and click on “verify email” button in the email received from ThingSpeak.
- Then, go back to the “Sign Up” page and set a password.
- Then, click on “Continue”.
- You will get the below message. Fill the details and click “OK”.

ThingSpeak Usage Intent

How are you planning to use ThingSpeak?

☐ Commercial work (including research)

☐ Government work (including research)

☐ Personal, non-commercial projects

☒ Student use, Teaching, or Research in academia

What is the name of your University?\*

SLIIT

What best describes your current role?\*

☒ Student ☐ Professor ☐ Researcher

What is the name of your Course or Project?\*

FC

Tell us something about your project (optional)

d) Create a new channel and develop the dashboard. (Follow the steps below)

- Click on “New Channel”

My Channels

New Channel

Search by tag

- Fill the details.

New Channel

Name lab9

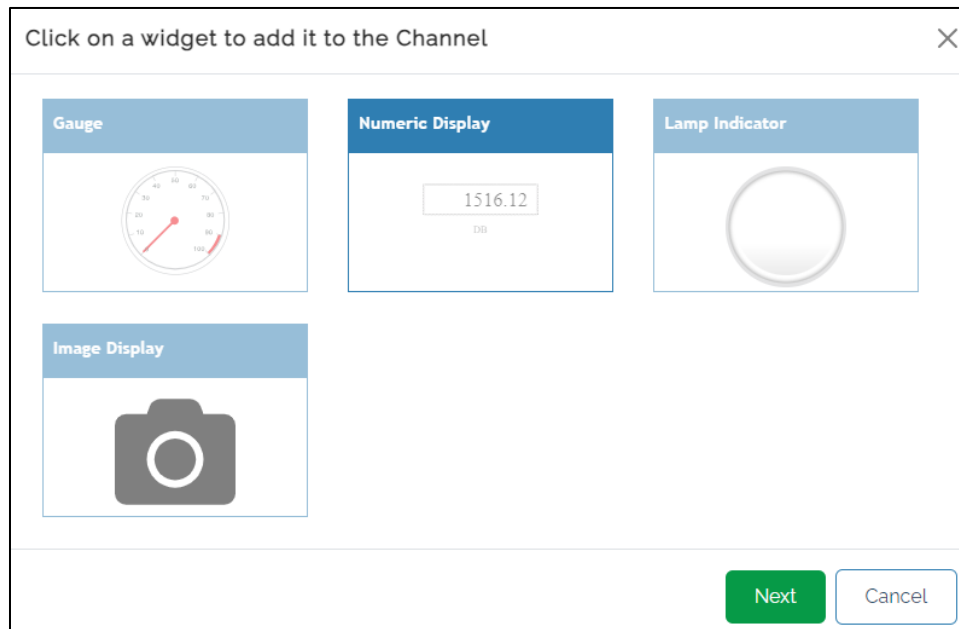
Description

Field 1 distance ☒

Field 2 ☐

Field 3 ☐

- c. Click on “Add Widgets” and add a Numeric Display.



- d. Then, configure the widget.

A dialog box titled "Configure widget parameters" with a help icon (?) and a close button (X) in the top right corner. It contains the following configuration fields: "Name" with the text "distance"; "Field" with a dropdown menu showing "Field 1"; "Update Interval" with a text box containing "15" and the label "second(s)"; "Units" with a text box containing "Enter Measurement Units"; and "Data Type" with radio buttons for "Integer" (selected) and "Decimal", followed by a dropdown menu showing "1" and the label "(# of places)". At the bottom right, there are "Create" and "Cancel" buttons.

- e. Go to Chart and click on the icon shown below and fill the given details.



Field 1 Chart Options
? X

Title:	Distance	Timescale:	
X-Axis:		Average:	
Y-Axis:		Median:	
Color:	#d62020	Sum:	
Background:	#ffffff	Rounding:	
Type:	line	Data Min:	
Dynamic?:	true	Data Max:	
Days:		Y-Axis Min:	
Results:	60	Y-Axis Max:	

Save
Cancel

- f. Then, click on the “Channel Setting” tab. Copy the Channel ID and save it in the *myChannelNumber* variable in the code.

Private View
Public View
Channel Settings
Sharing
API Keys

## Channel Settings

**Percentage Complete** 30%

**Channel ID** 2682285

**Name** lab9

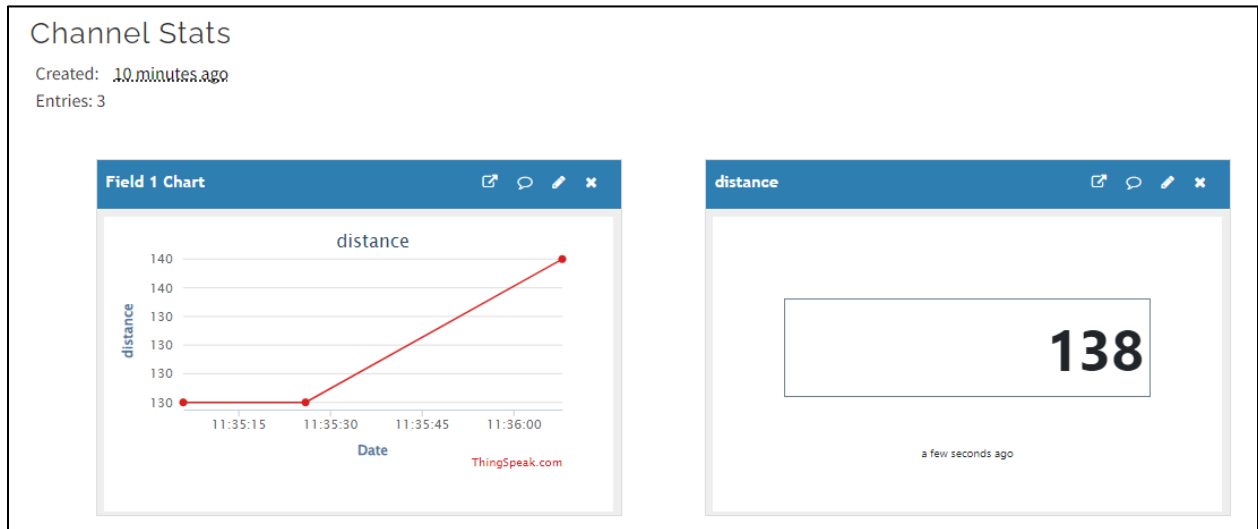
**Description**

**Field 1** distance ☒

**Field 2** ☐

**Field 3** ☐

- g. Goto API Keys and copy the key to *myApiKey* variable in the code.
- e ) Now, simulate your Wokwi circuit and check the output shown in the ThingSpeak dashboard in “Private View”. (You need to have a strong internet connection to view the result)



- e) Change the distance measurement manually and observe the behavior of the changes in the chart and the LEDs.

