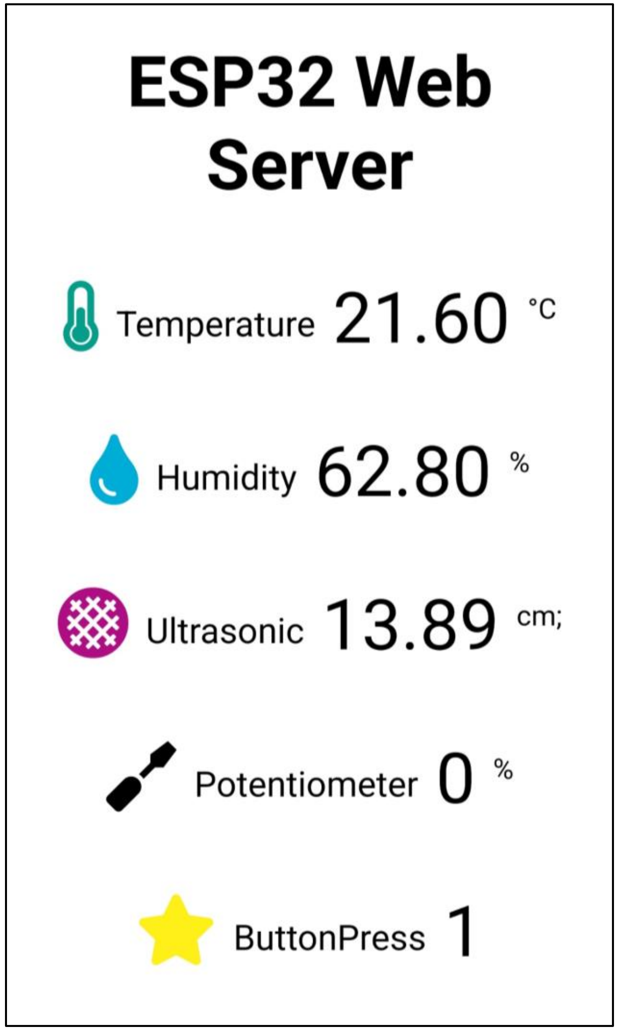
MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

ALBANY SENIOR HIGH SCHOOL

ESP32 Remote monitoring activity



ESP32 Web Server

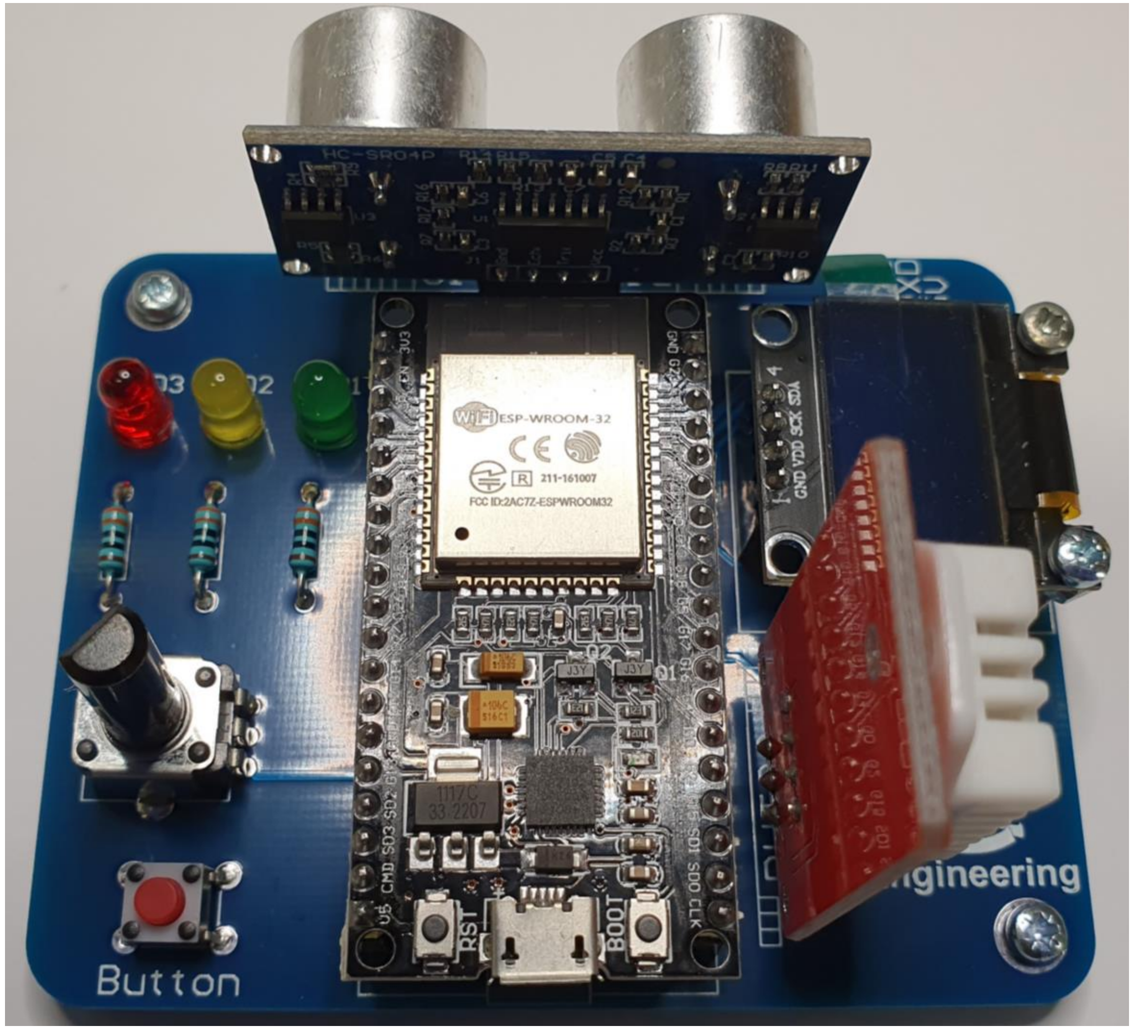
Temperature 21.60 °C

Humidity 62.80 %

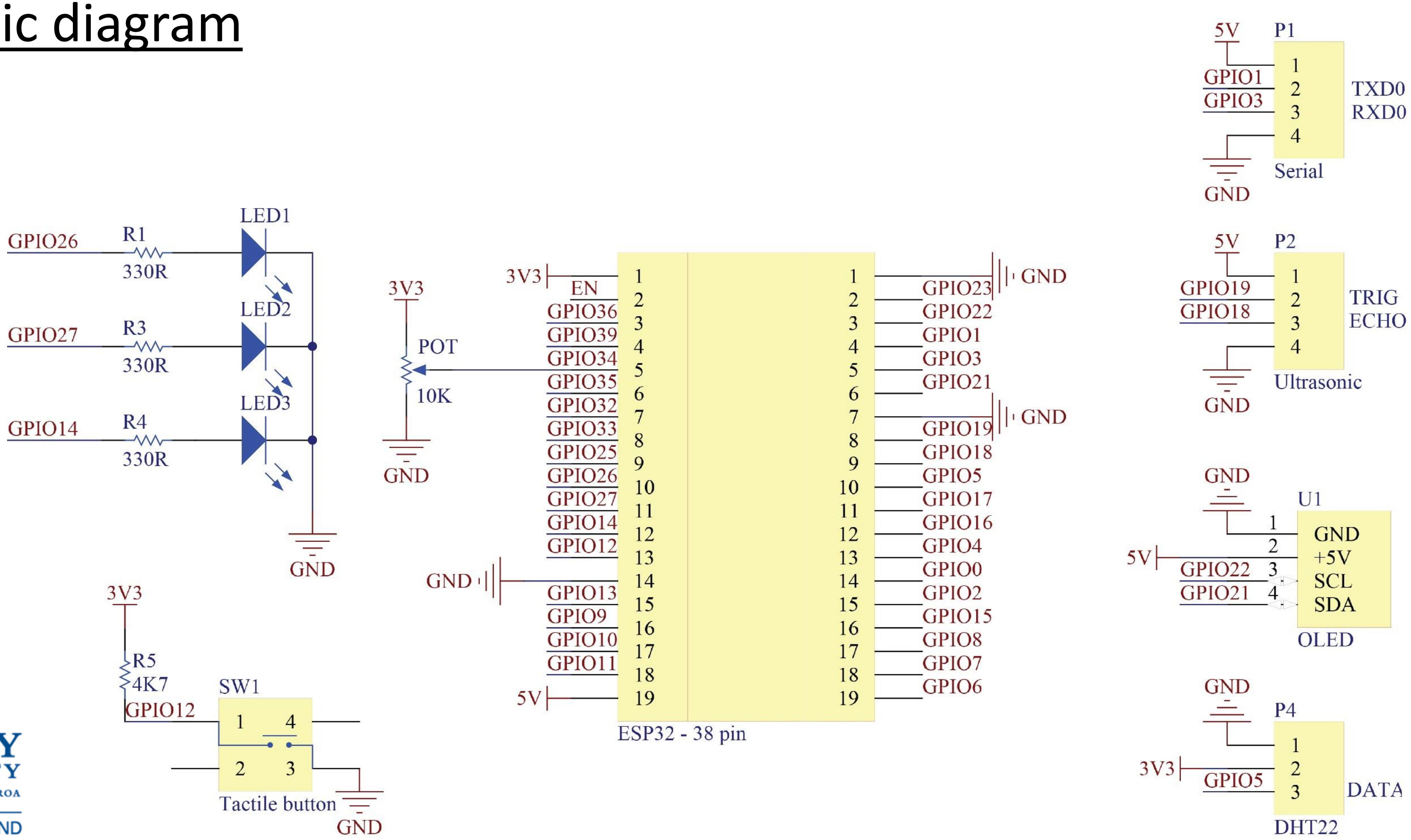
Ultrasonic 13.89 cm;

Potentiometer 0 %

ButtonPress 1



Schematic diagram

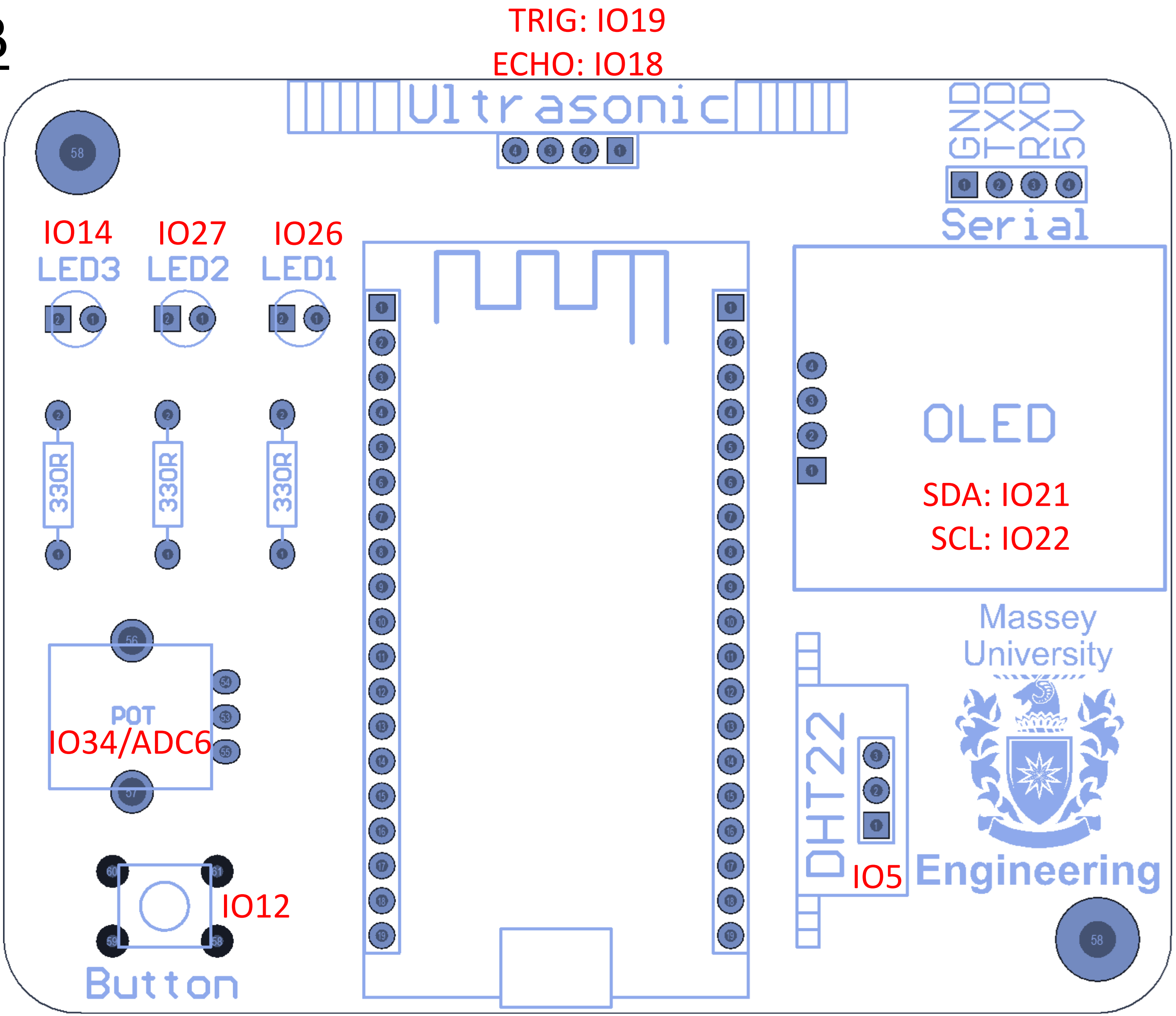


Components and PCB

Part	Quantity
ESP32 – 38 pin	1
LED - Green	1
LED - Red	1
LED - Yellow	1
Resistor - 330R	3
DHT22 sensor	1
Potentiometer	1
OLED – 0.96 inch	1
Ultrasonic sensor	1
Tactile button	1
M2.5, 10mm screw & nut for OLED	2
PCB	1
MDF Base	1
Female headers – 38 pin	2
Female header – 3 pin	1
Female header – 4 pin	2

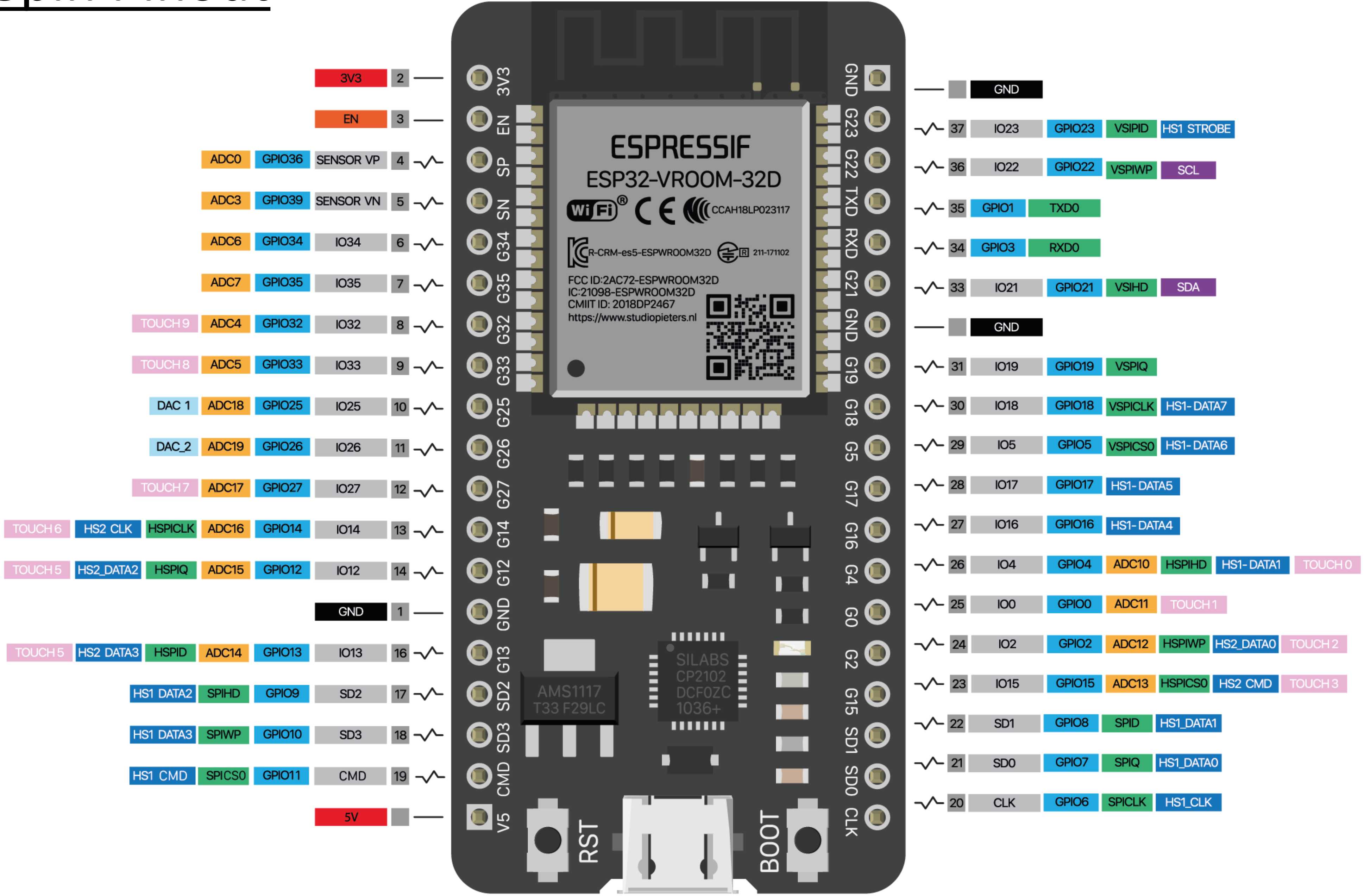
Note: ESP32 pins are labelled in red for the corresponding peripheral

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND



ESP32 – 38pin Pinout

∿	PWM
■	PIN NUMBER
■	NAME
■	GROUND
■	POWER
■	CONTROL
■	I/O
■	ADC
■	COMM. INTERFACE
■	DAC
■	I2C
■	HS
■	TOUCH



					3V3	2	—
					EN	3	—
			ADC0	GPIO36	SENSOR VP	4	∿
			ADC3	GPIO39	SENSOR VN	5	∿
			ADC6	GPIO34	IO34	6	∿
			ADC7	GPIO35	IO35	7	∿
		TOUCH9	ADC4	GPIO32	IO32	8	∿
		TOUCH8	ADC5	GPIO33	IO33	9	∿
		DAC 1	ADC18	GPIO25	IO25	10	∿
		DAC_2	ADC19	GPIO26	IO26	11	∿
		TOUCH7	ADC17	GPIO27	IO27	12	∿
TOUCH6	HS2 CLK	HSPICLK	ADC16	GPIO14	IO14	13	∿
TOUCH5	HS2_DATA2	HSPIQ	ADC15	GPIO12	IO12	14	∿
					GND	1	—
TOUCH5	HS2 DATA3	HSPID	ADC14	GPIO13	IO13	16	∿
		HS1 DATA2	SPIHD	GPIO9	SD2	17	∿
		HS1 DATA3	SPIWP	GPIO10	SD3	18	∿
		HS1 CMD	SPICS0	GPIO11	CMD	19	∿
					5V		—

—		GND					
∿	37	IO23	GPIO23	VSPID	HS1 STROBE		
∿	36	IO22	GPIO22	VSPIWP	SCL		
∿	35	GPIO1	TXD0				
∿	34	GPIO3	RXD0				
∿	33	IO21	GPIO21	VSIHD	SDA		
—		GND					
∿	31	IO19	GPIO19	VSPIQ			
∿	30	IO18	GPIO18	VSPICLK	HS1-DATA7		
∿	29	IO5	GPIO5	VSPICS0	HS1-DATA6		
∿	28	IO17	GPIO17	HS1-DATA5			
∿	27	IO16	GPIO16	HS1-DATA4			
∿	26	IO4	GPIO4	ADC10	HSPIHD	HS1-DATA1	TOUCH0
∿	25	IO0	GPIO0	ADC11	TOUCH1		
∿	24	IO2	GPIO2	ADC12	HSPIWP	HS2_DATA0	TOUCH2
∿	23	IO15	GPIO15	ADC13	HSPICS0	HS2 CMD	TOUCH3
∿	22	SD1	GPIO8	SPID	HS1_DATA1		
∿	21	SD0	GPIO7	SPIQ	HS1_DATA0		
∿	20	CLK	GPIO6	SPICLK	HS1_CLK		

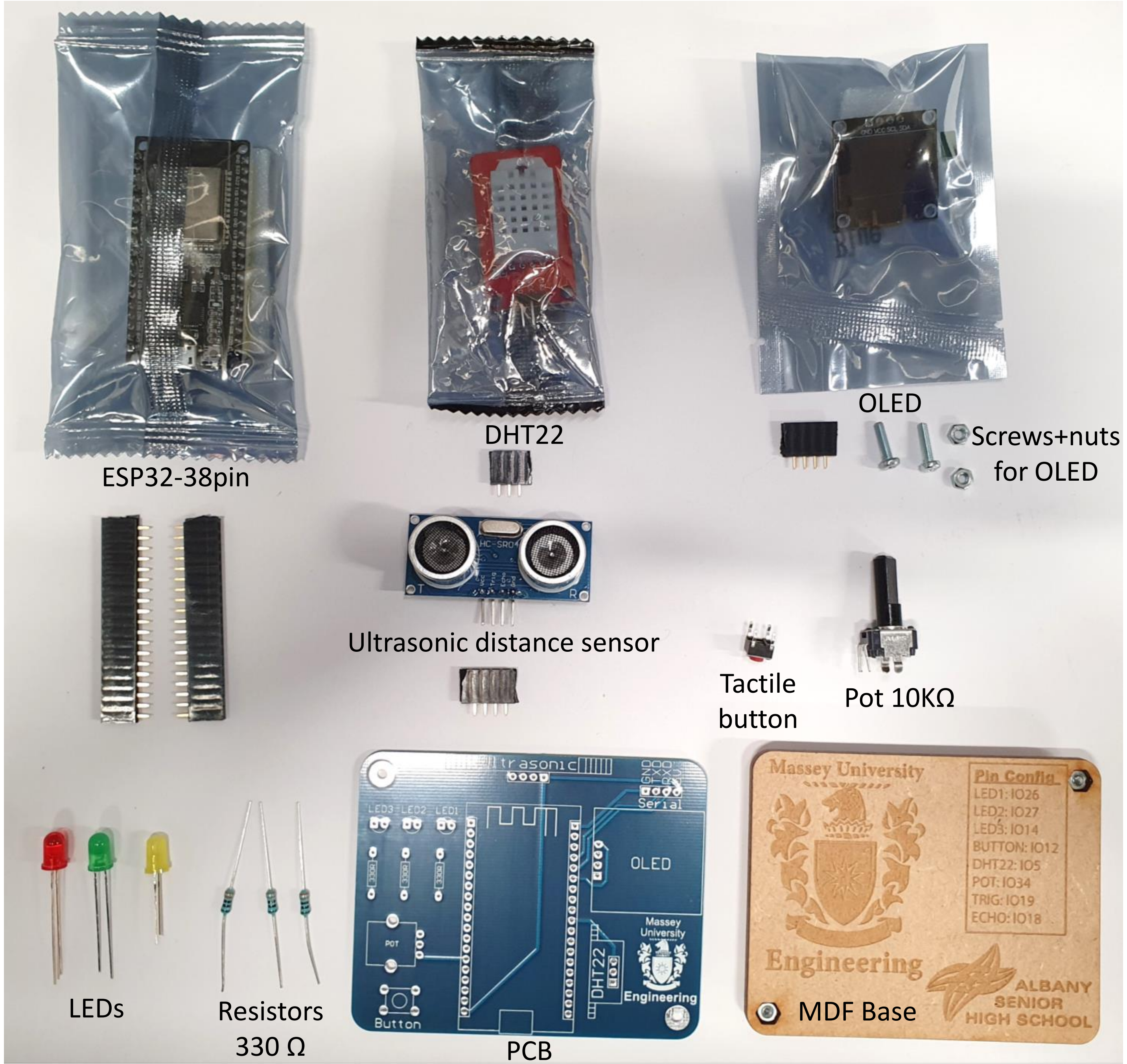
MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

Components

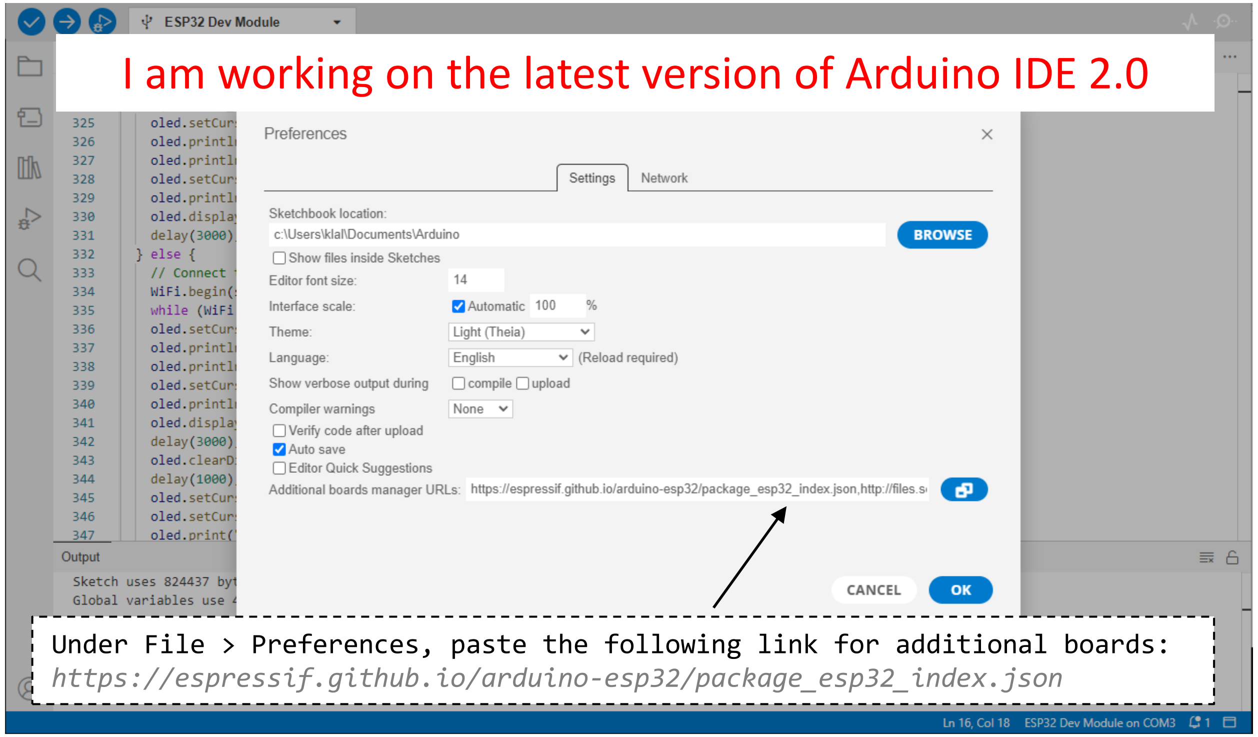
Notes:

- Resistors are not polarised and can be soldered on the PCB in any direction.
- The LEDs are polarised. Long leg is positive and short leg is negative, denoted on the PCB by a square pad for positive and round pad for negative.
- Solder the provided female headers in place of ESP32, DHT22, OLED, and Ultrasonic. Mount these components on top of the female headers, so that, the component can be replaced if broken.
- The 4-pin connector labelled Serial on top right corner of the PCB provides an added functionality to communicate with another device/microcontroller such as Arduino. This pin can be left unconnected.

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND



Sample Arduino Code – Installing the ESP32 board and libraries



I am working on the latest version of Arduino IDE 2.0

```
        oled.setCur
        oled.printl
        oled.printl
        oled.setCur
        oled.printl
        oled.displa
        delay(3000)
      } else {
        // Connect 
        WiFi.begin(
        while (WiFi
        oled.setCur
        oled.printl
        oled.printl
        oled.setCur
        oled.printl
        oled.displa
        delay(3000)
        oled.clearD
        delay(1000)
        oled.setCur
        oled.setCur
        oled.print(
```

```
Sketch uses 824437 byt
Global variables use 4
```

Preferences

Settings | Network

Sketchbook location: c:\Users\klal\Documents\Arduino BROWSE

□ Show files inside Sketches

Editor font size: 14

Interface scale: ☑ Automatic 100 %

Theme: Light (Theia)

Language: English (Reload required)

Show verbose output during □ compile □ upload

Compiler warnings: None

□ Verify code after upload

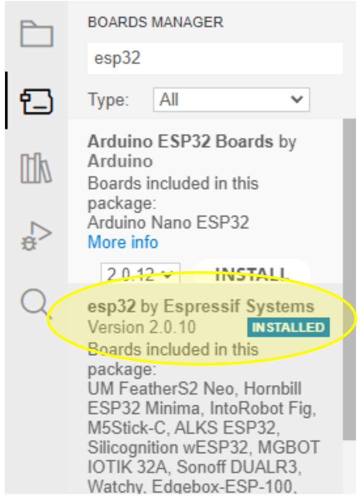
☑ Auto save

□ Editor Quick Suggestions

Additional boards manager URLs: https://espressif.github.io/arduino-esp32/package_esp32_index.json,http://files.s

CANCEL OK

Under File > Preferences, paste the following link for additional boards:
https://espressif.github.io/arduino-esp32/package_esp32_index.json



BOARDS MANAGER

esp32

Type: All

Arduino ESP32 Boards by Arduino
Boards included in this package:
Arduino Nano ESP32
More info

2.0.12 INSTALL

esp32 by Espressif Systems
Version 2.0.10 INSTALLED
Boards included in this package:
UM FeatherS2 Neo, Hornbill ESP32 Minima, IntoRobot Fig, M5Stick-C, ALKS ESP32, Silicognition wESP32, MGBOT IOTIK 32A, Sonoff DUALR3, Watchy, Edgebox-ESP-100,

Under Tools > Board > Boards Manager, search for ESP32. Install "esp32 by Espressif Systems".

The required libraries can be downloaded from the links below. To install them, you will need to extract the contents of the zip files and place it in the Arduino libraries folder which is usually found in Documents > Arduino.

- https://github.com/me-no-dev/AsyncTCP
- https://github.com/me-no-dev/ESPAsyncWebServer/archive/master.zip
- https://github.com/adafruit/Adafruit_SSD1306
- https://github.com/adafruit/Adafruit-GFX-Library
- https://downloads.arduino.cc/libraries/github.com/adafruit/DHT_sensor_library-1.4.4.zip

Download a copy of the code from my GitHub page
https://github.com/KlalEngineer/ASHS.git

Sample Arduino Code – Network SSID & Password

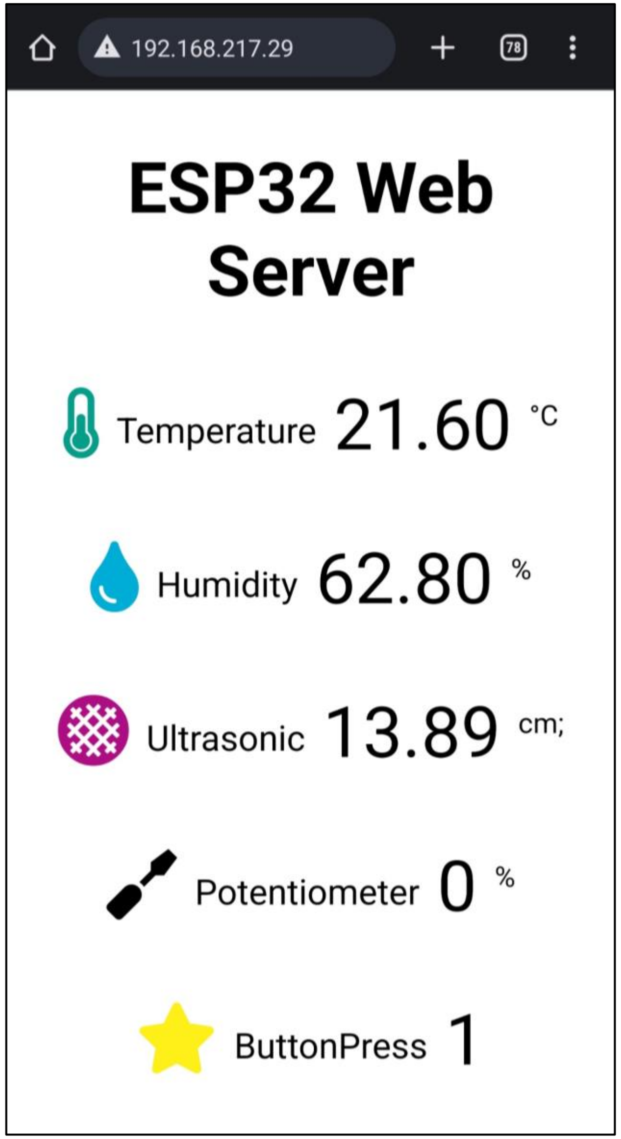
```
// Import required libraries
#include "WiFi.h"
#include "ESPAsyncWebServer.h"
#include <Wire.h>
#include <DHT22.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

// set this to 'true' when testing all the peripherals on the PCB to e
works
// set this to 'false' to enable the web server
#define HARDWARE_TEST true

// Replace with your network credentials
const char* ssid = "place-your-network-name-here";
const char* password = "place-your-network-password-here";
/* Ultrasonic sensor variables */
#define TRIG_PIN 19 // ultrasonic sensor - trig pin
#define ECHO_PIN 18 // ultrasonic sensor - echo pin
//define sound speed in cm/uS
#define SOUND_SPEED 0.034
#define CM_TO_INCH 0.393701
long duration = 0;
float distance = 0.0;
/* DHT22 sensor variables */
#define DHT_PIN 5 // data pin of DHT22 sensor
DHT22 dht22(DHT_PIN);
float t_prev = 0.0;
float h_prev = 0.0;
/* OLED size in pixels */
#define SCREEN_WIDTH 128 // OLED display width,  in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels
/* Button, Potentiometer and LEDs */
#define BUTTON_PIN 12
#define POT 34 // Potentiometer
#define LED_RED 27
#define LED_GREEN 26
#define LED_YELLOW 14
// declare an SSD1306 display object connected to I2C
Adafruit_SSD1306 oled(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
// Create AsyncWebServer object on port 80
AsyncWebServer server(80);
```

Set the HARDWARE_TEST variable to ‘true’ for complete hardware test. Set to ‘true’ when you are using the board for the first time after soldering the PCB to ensure everything works correctly. When set to ‘false’, ESP32 will connect to the local network

ESP32 needs to connect to your local network. Place your network name and password here. Your smartphone needs to be connected to the same network. Upon power up, ESP32 will create a generic web server to display sensor data on a specific IP address that your local network will assign. The OLED will display the IP address and the web server will be hosted on that IP which can be accessed through your smartphone.

Note: To upload the program on ESP32, you may need to hold down ‘BOOT’ button.

Screenshot of web server receiving data from sensors



192.168.217.29

ESP32 Web Server

Temperature 21.60 °C

Humidity 62.80 %

Ultrasonic 13.89 cm;

Potentiometer 0 %

ButtonPress 1

Finished product