

Assignment -4

Name:C.Subhiksha

Regno:950819106092

```
#include <stdio.h>
#include <stdbool.h>
#include <freertos/FreeRTOS.h>
#include <freertos/task.h>
#include <esp_err.h>
#include "ultrasonic.h"
#define ECHO_GPIO 12
#define TRIGGER_GPIO 13
#define MAX_DISTANCE_CM 500 // Maximum of 5 meters
void ultrasonic_test(void *pvParameters)
{
    float distance;
    ultrasonic_sensor_t sensor = {
        .trigger_pin = TRIGGER_GPIO,
        .echo_pin = ECHO_GPIO
    };
    ultrasonic_init(&sensor);
    while (true) {
        esp_err_t res = ultrasonic_measure(&sensor, MAX_DISTANCE_CM, &distance);
        if (res == ESP_OK) {
            printf("Distance: %0.04f m\n", distance);
        } // Print error
        else {
            printf("Error %d: ", res);
            switch (res) {
                case ESP_ERR_ULTRASONIC_PING:
                    printf("Cannot ping (device is in invalid state)\n");
                    break;
                case ESP_ERR_ULTRASONIC_PING_TIMEOUT:
                    printf("Ping timeout (no device found)\n");
                    break;
                case ESP_ERR_ULTRASONIC_ECHO_TIMEOUT:
                    printf("Echo timeout (i.e. distance too big)\n");
                    break;
                default:
                    printf("%s\n", esp_err_to_name(res));
            }
        }
    }
}
```

Assignment -4

```
}  
vTaskDelay(pdMS_TO_TICKS(500));  
}  
}  
void app_main()  
{  
xTaskCreate(ultrasonic_test, "ultrasonic_test", configMINIMAL_STACK_SIZE * 3,  
NULL, 5, NULL);  
}
```

Wokwi IDE interface showing the Ultrasonic_sensor.ino code and a simulation of the circuit.

Code (Ultrasonic_sensor.ino):

```
8  
9 #define ECHO_PIN 2  
10 #define TRIG_PIN 3  
11  
12 void setup() {  
13   Serial.begin(115200);  
14   pinMode(LED_BUILTIN, OUTPUT);  
15   pinMode(TRIG_PIN, OUTPUT);  
16   pinMode(ECHO_PIN, INPUT);  
17 }  
18  
19 float readDistanceCM() {  
20   digitalWrite(TRIG_PIN, LOW);  
21   delayMicroseconds(2);  
22   digitalWrite(TRIG_PIN, HIGH);  
23   delayMicroseconds(10);  
24   digitalWrite(TRIG_PIN, LOW);  
25   int duration = pulseIn(ECHO_PIN, HIGH);  
26   return duration * 0.034 / 2;  
27 }  
28  
29 void loop() {  
30   float distance = readDistanceCM();  
31  
32   bool isNearby = distance < 100;  
33   digitalWrite(LED_BUILTIN, isNearby);  
34  
35   Serial.print("Measured distance: ");  
36   Serial.println(readDistanceCM());  
37  
38   delay(100);  
39 }  
40
```

Simulation:

The simulation shows an Arduino Uno connected to an HC-SR04 ultrasonic sensor. The sensor's VCC is connected to the 5V pin, GND to GND, TRIG to pin 3, and ECHO to pin 2. A red LED is connected to the Arduino's GND and a digital pin (likely 13). The simulation output shows the measured distance in centimeters:

```
Measured distance: 96.68  
Measured distance: 96.68  
Measured distance: 96.76  
Measured distance: 96.76  
Measured distance: 96.66  
Measured distance: 96.68  
Measured distance: 96.68
```

The interface includes a Wokwi logo, a SAVE button, a SHARE button, and a heart icon. The top bar shows the project name "Ultrasonic_sensor.ino - Wokwi" and a "Docs" button. The bottom status bar displays the temperature (22°C), weather (Partly cloudy), and system time (10:57 PM, 25-10-2022).