**REPORT**

**INTRODUCTION**

The ESP32-WROOM-32 is a potent, universal Wi-Fi + Bluetooth + Bluetooth LE MCU module that can handle a wide range of functions, including voice encoding, music streaming, and decoding, as well as low-power sensor networks.

**Figure (1.1) :- ESP32**

**Components That Used**

* ESP32 Wroom Dev Module
* 2 LEDs
* Breadboard
* Jumper Cables
* Push Buttons
* Registers
* Oscilloscope

**Assignment -1 Question Evolution**

Table

Description automatically generated

Calculation of parameters Each student will have a set of (a, b, c, d, and mode) parameters. To calculate your set of parameters, use the following information. Alphabet is numbered - a=1, b=2, c=3, … m=13 n=13, o=12, p=11, … z=1

Letters => “AKHUN”

A = 1

K = 11

H = 8

U = 6

N = 13

a = first letter \* 100μS = 100μS

b = second letter \* 100μS = 11 \* 100 = 1100μS

c = third letter + 4 = 8 + 4 = 12 Pulse

d = fourth letter \* 500μS = 6 \* 500μS = 3000μS

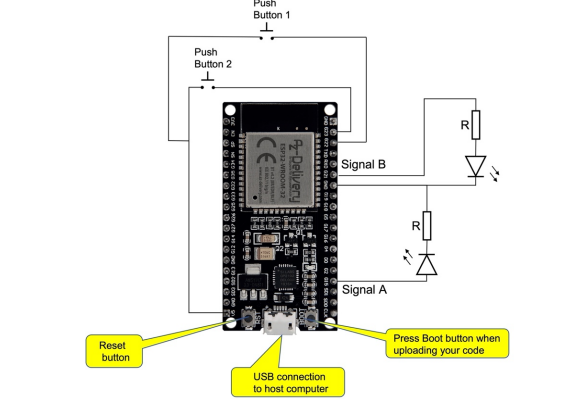
mode = remainder (fifth letter/4) + 1 = (13 / 4) + 1 = 2

As I have Mode = 2 so, I have to generate the inverted form of complete Sig A waveform Generate (from the largest pulse to the shortest) until switch 2 set back to 0.

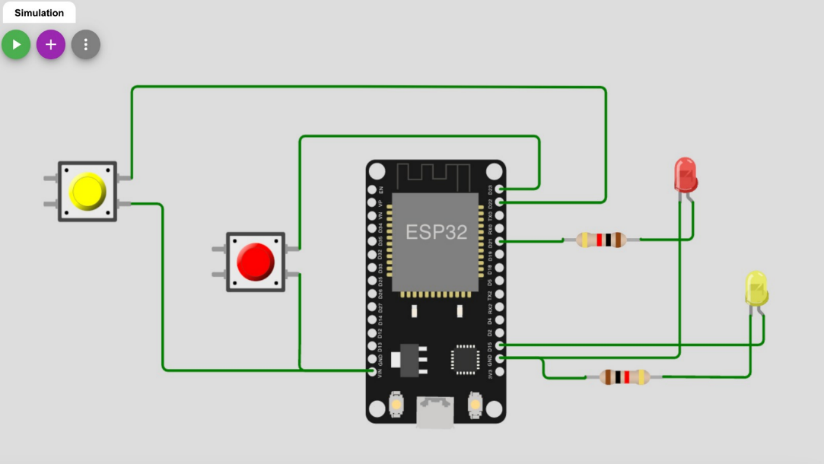
Build the circuit using the provided equipment (ESP32, 2 switches, 2 LEDs and an oscilloscope to visualize Signals A and B)

**4. ESP32 Connection Diagram**

ESP32 circuit diagram simulates on WOKWI, Wokwi is an**online simulator for Arduino and Electronics**. It's designed for makers, by makers. You can use Wokwi to learn Arduino programming, prototype your ideas, and share your projects with other makers. Look below figure for the understanding of a circuit diagram.



**Figure (1.2) :- Circuit Diagram**



**Figure (1.3) :- Simulation of circuit diagram**

**Task 1 :-**

**Arduino IDE Code:**

//Assignment\_1 ESP32 Pulse generator

const int led\_1 = 15; // ESP32 GPIO PIN 15

const int led\_2 = 21; // ESP32 GPIO PIN 21

const int btn\_1 = 22; // ESP32 GPIO PIN 22

const int btn\_2 = 23; // ESP32 GPIO PIN 23

void setup() {

Serial.begin(115200); //BaudRate for Serial Moniter Visulization of output

pinMode(15, OUTPUT); // GPIO PIN 15 is for led\_1 output

pinMode(21, OUTPUT); // GPIO PIN 21 is for Led\_2 output

pinMode(22, INPUT); // GPIO PIN 22 is for Led\_2 input

pinMode(23, INPUT); // GPIO PIN 23 is for Led\_2 input

}

// mod\_a is for noemal pulse run

void mod\_a(int pin, int puls\_a, int time\_inc, int pause\_b, int d) {

// First led\_2 blink for 50 microseconds

digitalWrite(led\_2, HIGH);

delayMicroseconds(50);

digitalWrite(led\_2, LOW);

delayMicroseconds(50);

// The loop is consist of 12 pulses = 1 waveform

for(int i = 0; i < 12; i ++) {

digitalWrite(pin, HIGH);

delayMicroseconds(puls\_a);

// Increase the pulse duration with 50 microseconds

puls\_a = puls\_a + time\_inc;

Serial.println(puls\_a);

digitalWrite(pin, LOW);

// It will Pause between every pulse

delayMicroseconds(pause\_b);

}

// After loop it will delay waveform for given time

digitalWrite(pin, LOW);

delayMicroseconds(d);

Serial.println("Mode A");

}

// mod\_b is for modified pulse run

void mod\_b(int pin, int puls\_a, int time\_inc, int pause\_b, int d) {

// First led\_2 blink for 50 microseconds

digitalWrite(led\_2, HIGH);

delayMicroseconds(50);

digitalWrite(led\_2, LOW);

delayMicroseconds(50);

// The loop is consist of 12 pulses = 1 waveform

for(int i = 0; i < 12; i ++) {

digitalWrite(pin, HIGH);

delayMicroseconds(puls\_a);

// Decrease the pulse duration with 50 microseconds

puls\_a = puls\_a - time\_inc;

Serial.println(puls\_a);

digitalWrite(pin, LOW);

// It will Pause between every pulse

delayMicroseconds(pause\_b);

}

// After loop it will delay waveform for given time

digitalWrite(pin, LOW);

delayMicroseconds(d);

Serial.println("Mode B");

}

void loop() {

// If button 1 is pressed then disablke pulses

if(digitalRead(btn\_1)==HIGH)

{

digitalWrite(led\_1, LOW);

digitalWrite(led\_2, LOW);

}

// If button 2 is pressed then change pulse to Modified pulse

else if(digitalRead(btn\_2)==HIGH)

{

mod\_b(led\_1,650, 50, 1100, 3000);

}

//if none of the buttons is pressed then pulse running continiously

else

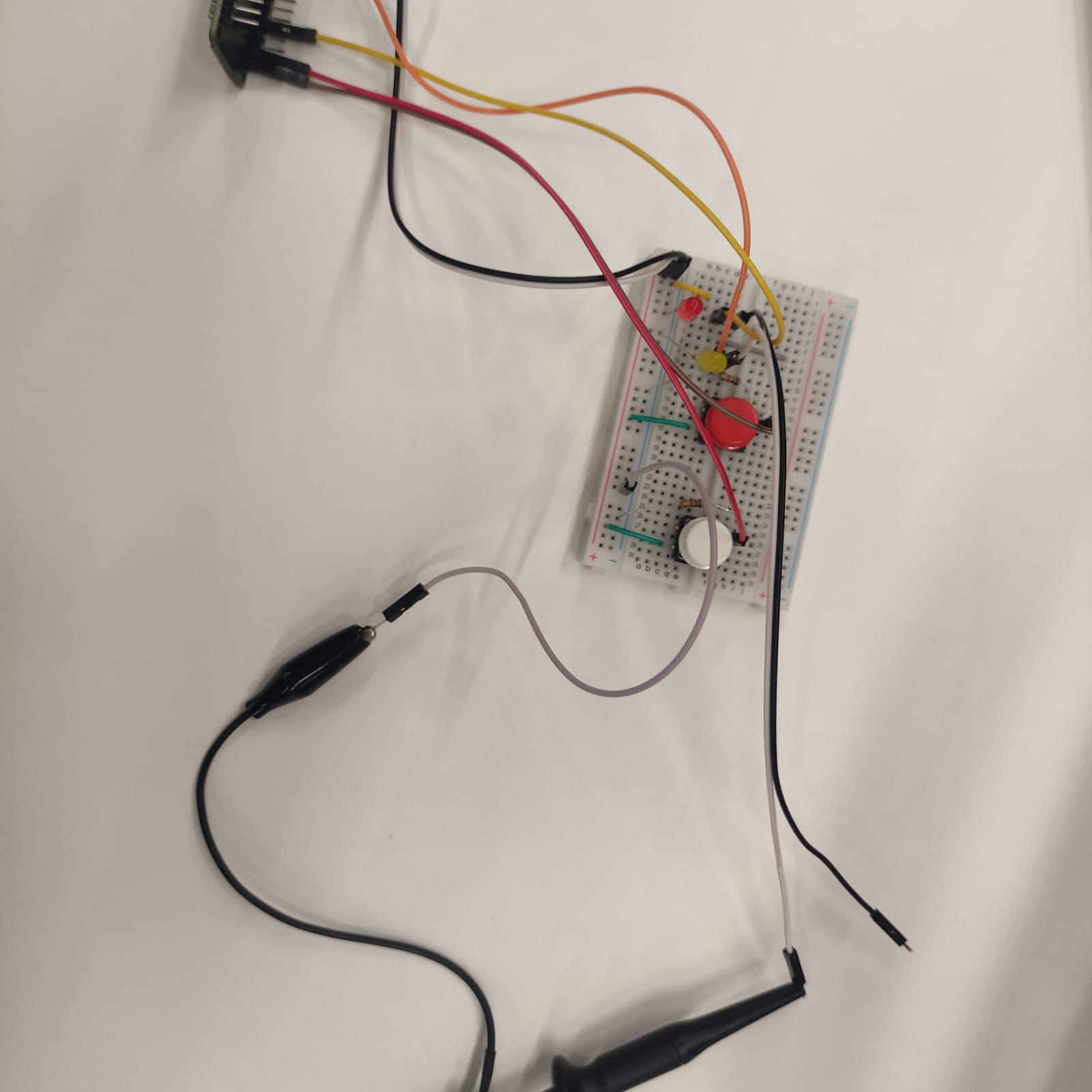
{

mod\_a(led\_1,100, 50, 1100, 3000);

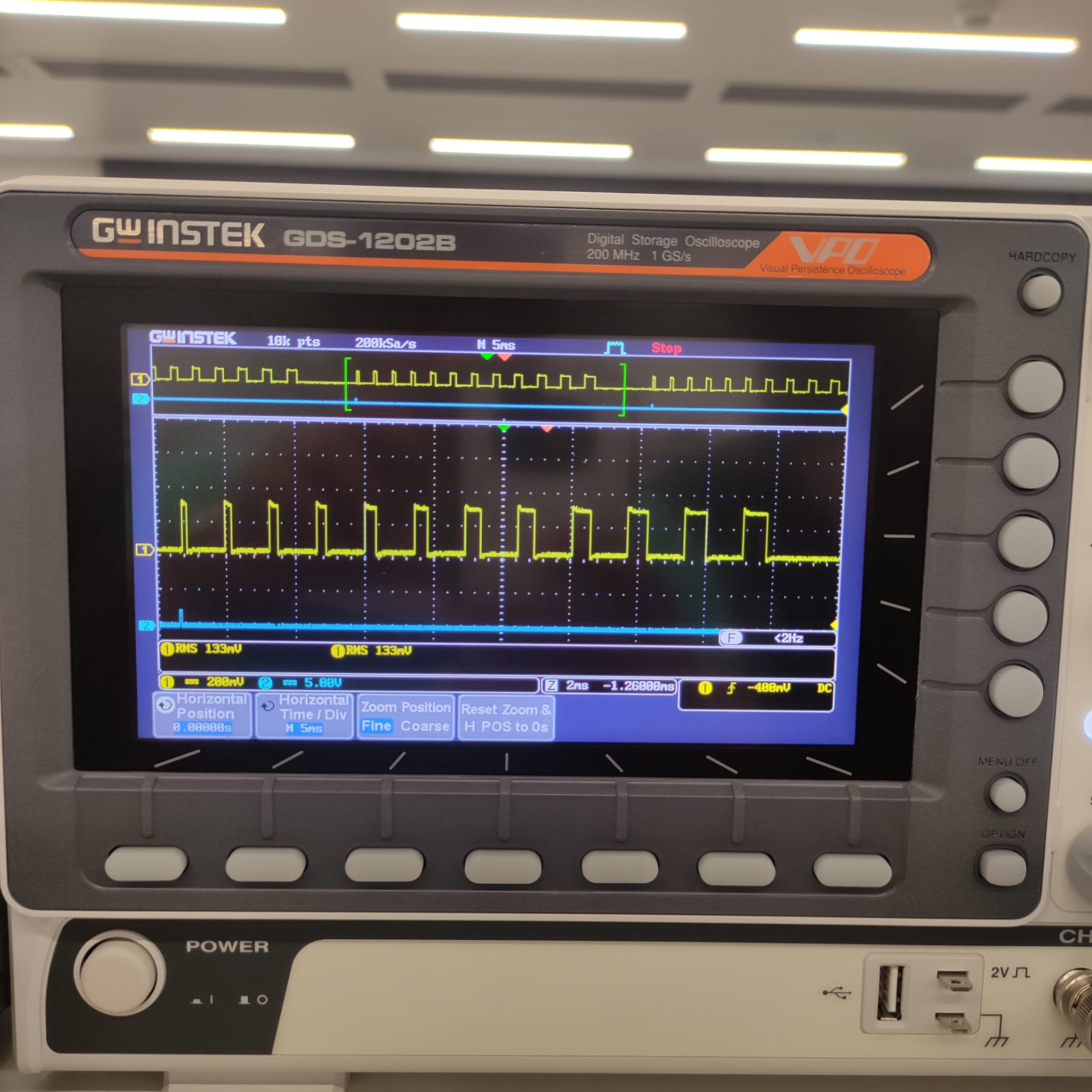
}

}

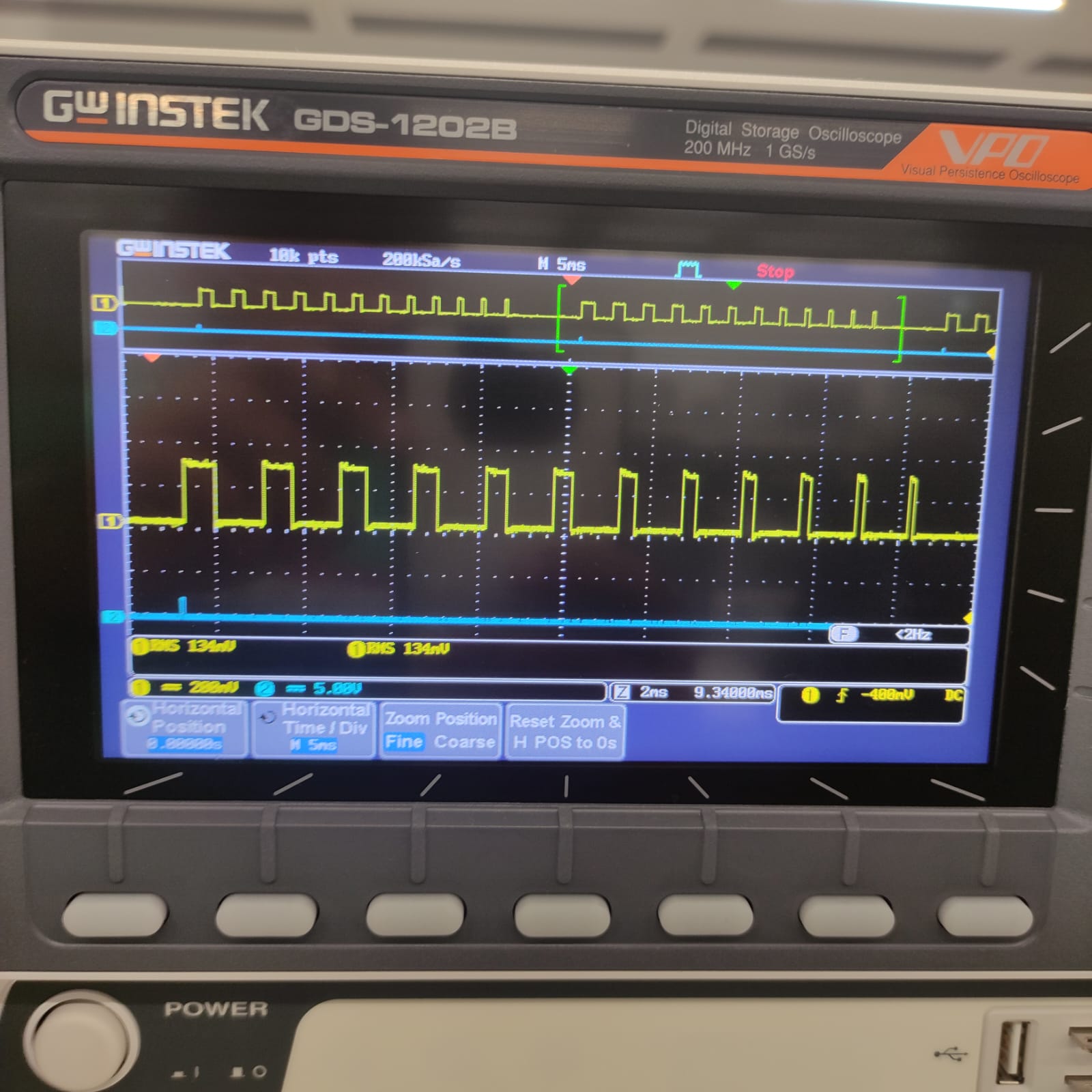
**Task 2 :-**



**Figure (2.1): Actual Circuit Diagram**



**Figure (2.2): Pulse in Normal Mode**



**Figure (2.3): Pulse in mode 1**

**Task 3 :-**

Table

Description automatically generated

**Figure (3.1): N – S Diagram**

**Github Link :-**

[Mohmadnoor\_Akhunji/ESP32-signal-regerator: Generating signal for 2 pulse (github.com)](https://github.com/noor-akhunji/Embeded-Software-Assignments/blob/main/arduino_assignment_1_f_code.ino)