Assignment 1

Embedded Software(B31DG)

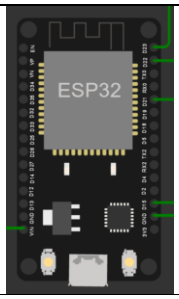
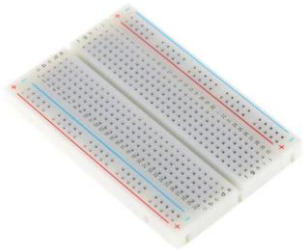
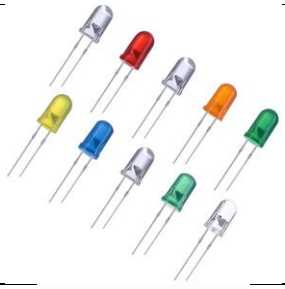
Abram Jijimon Theckayil

ID: H00421874

Introduction:

The ESP32 is a low-cost, low-power system with Wi-Fi and dual-mode Bluetooth and in this project, we will create a pulse generator using the ESP32. We will generate two distinct pulses, one in normal mode and the other in the specified mode.

Components:

ESP 32	
Wires	
Breadboard	
LEDs	
Button	

Resistor	

Table 1: Components for ESP 32 Circuit

Diagram:

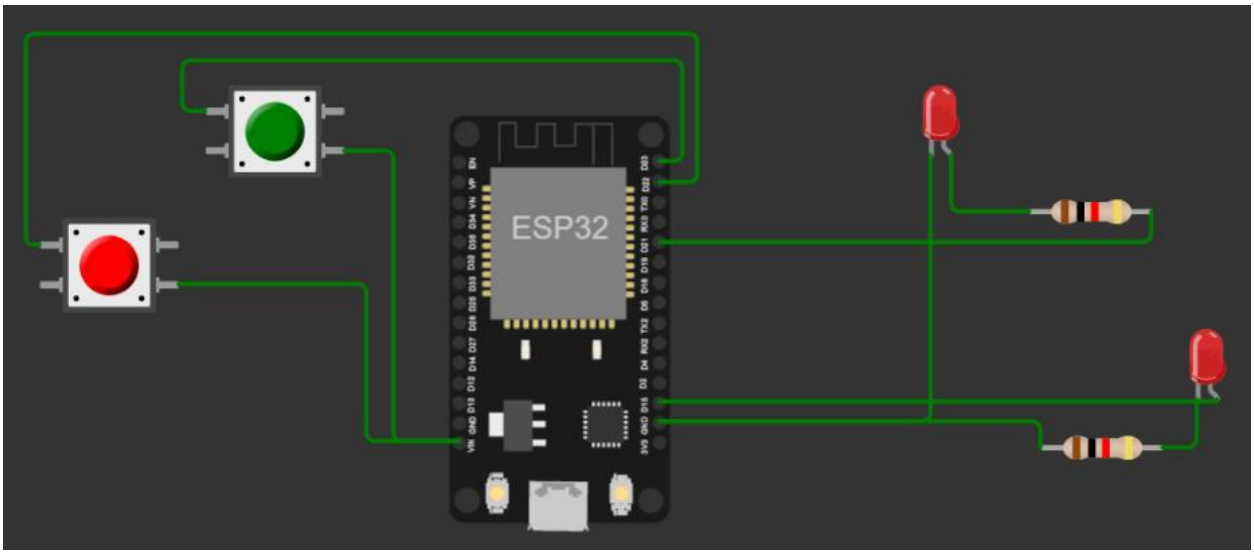


Fig 1: ESP32 Circuit

- Led Pin 1 is connected to D15
- Led Pin 2 is connected to D21
- Push Button 1 is connected to D22
- Push Button 2 is connected to D23

Calculations:

a = 1 * 100µs = 100µs

b = 1 * 100µs = 200µs

c = 9+4 = 13

d = 1 * 500µs = 500µs = 0.5ms

Mode = (13/4) +1 = 4.25 = 4

Calculations for Mode 4:

Half of d & b

d = 0.5/2 = 0.25ms

b = 200/2 = 100µs

Code:

```
// We assigned a name LED pin to pin number 15
const int LEDPIN1 = 15;
const int LEDPIN2 = 21;
// this will assign the name PushButton to pin number 22
const int PushButton1 = 22;
const int PushButton2 = 23;
// This Setup function is used to initialize everything
void setup()
{
// This statement will declare pin 15 as digital output
pinMode(LEDPIN1, OUTPUT);
// This statement will declare pin 21 as digital output
pinMode(LEDPIN2, OUTPUT);
// This statement will declare pin 22 as digital input
pinMode(PushButton1, INPUT);
// This statement will declare pin 23 as digital output
```

```
pinMode(PushButton2, INPUT);
}


void loop()

{

if(digitalRead(PushButton1)==LOW) //If button 1 is not pressed this will happen
{
  if(digitalRead(PushButton2)==LOW) // //If button 2 is not pressed this will happen
  {
   digitalWrite(LEDPIN1, HIGH);
   delayMicroseconds(100);
   digitalWrite(LEDPIN1, LOW);
   delayMicroseconds (200);
   digitalWrite(LEDPIN1, HIGH);
   delayMicroseconds (150);
   digitalWrite(LEDPIN1, LOW);
   delayMicroseconds (200);
   digitalWrite(LEDPIN1, HIGH);
   delayMicroseconds (200);
   digitalWrite(LEDPIN1, LOW);
   delayMicroseconds (200);
   digitalWrite(LEDPIN1, HIGH);
   delayMicroseconds (250);
   digitalWrite(LEDPIN1, LOW);
```

```
delayMicroseconds (200);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds (300);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds (200);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds (350);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds (200);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds (400);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds (200);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds (450);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds (200);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds (500);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds (200);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds (550);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds (200);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds (600);
```

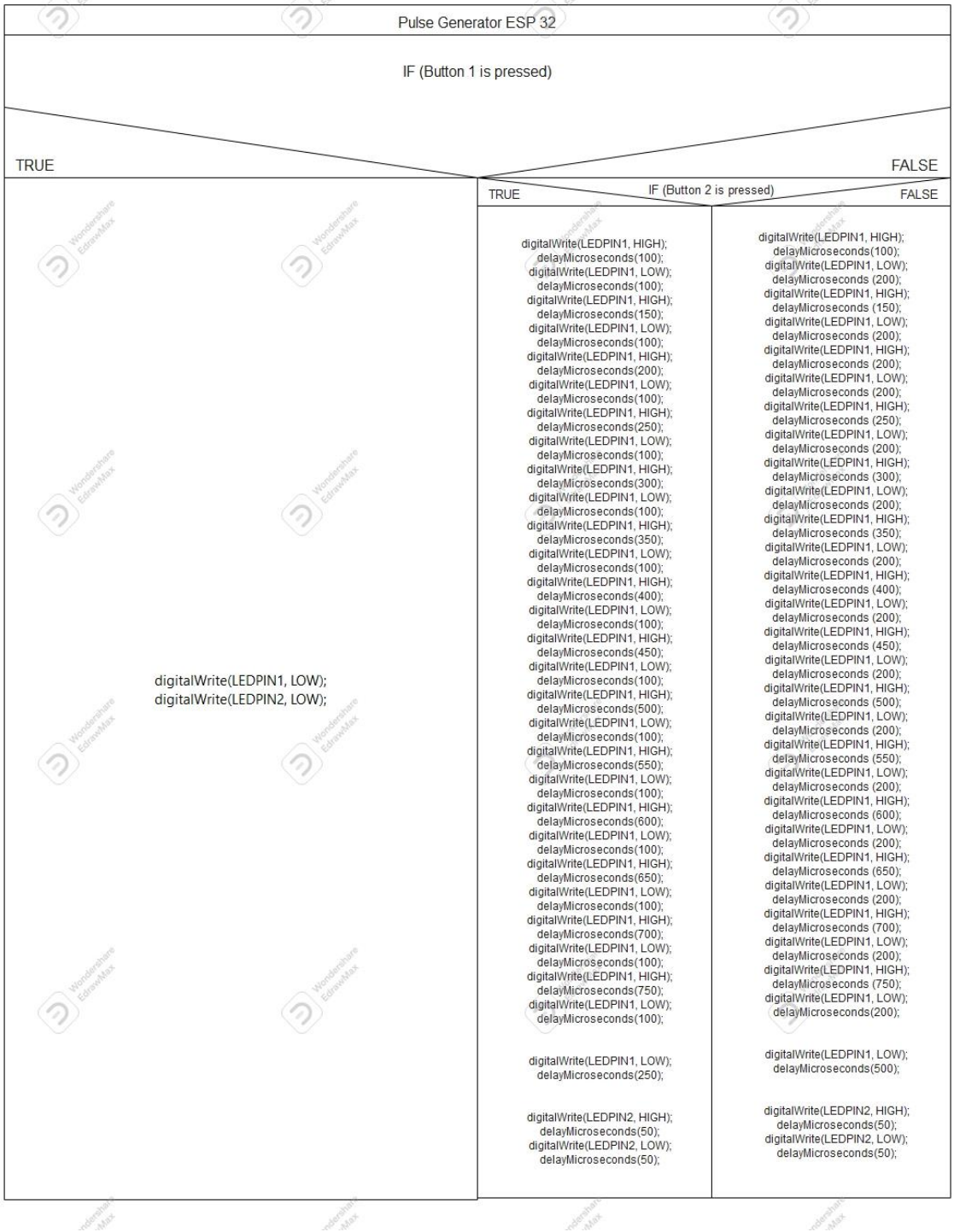
```
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds (200);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds (650);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds (200);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds (700);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds (200);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds (750);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds(200);


  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds(500);


  digitalWrite(LEDPIN2, HIGH);
  delayMicroseconds(50);
  digitalWrite(LEDPIN2, LOW);
  delayMicroseconds(50);
 }

else //If button 2 is pressed this will happen
```

```
{
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds(100);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds(100);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds(150);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds(100);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds(200);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds(100);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds(250);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds(100);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds(300);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds(100);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds(350);
  digitalWrite(LEDPIN1, LOW);
  delayMicroseconds(100);
  digitalWrite(LEDPIN1, HIGH);
  delayMicroseconds(400);
```

```
digitalWrite(LEDPIN1, LOW);
delayMicroseconds(100);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds(450);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds(100);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds(500);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds(100);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds(550);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds(100);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds(600);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds(100);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds(650);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds(100);
digitalWrite(LEDPIN1, HIGH);
delayMicroseconds(700);
digitalWrite(LEDPIN1, LOW);
delayMicroseconds(100);
digitalWrite(LEDPIN1, HIGH);
```

```
    delayMicroseconds(750);
    digitalWrite(LEDPIN1, LOW);
    delayMicroseconds(100);



    digitalWrite(LEDPIN1, LOW);
    delayMicroseconds(250);



    digitalWrite(LEDPIN2, HIGH);
    delayMicroseconds(50);
    digitalWrite(LEDPIN2, LOW);
    delayMicroseconds(50);
  }
}

else //If button 1 is pressed this will happen
{
    digitalWrite(LEDPIN1, LOW);
    digitalWrite(LEDPIN2, LOW);
}


}
```

Nassi–Shneiderman diagram:



Github:

https://github.com/AbramJT/Pulse-Generator-ESP32/blob/main/Abram%20Assignment%201%20(ESP32)