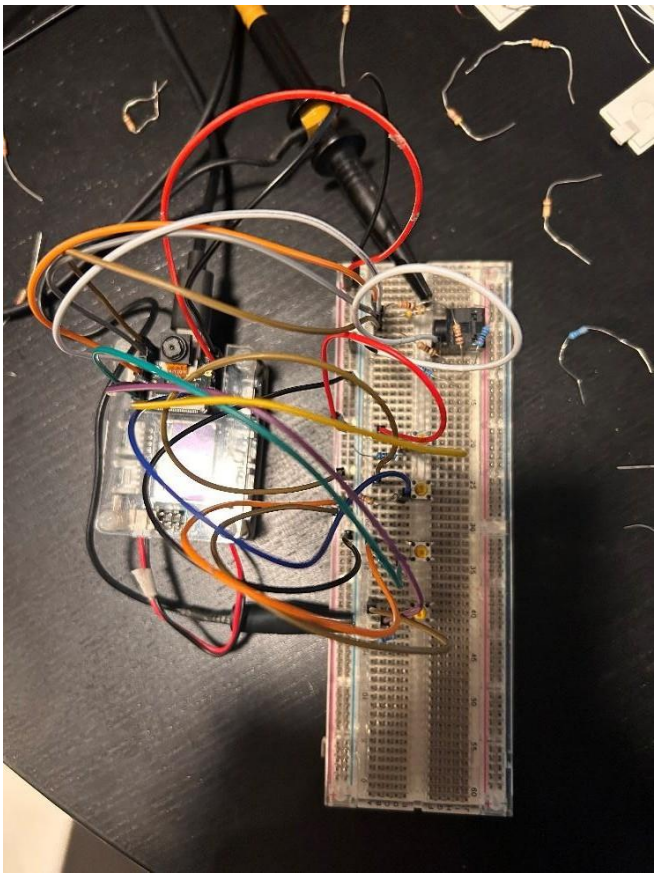
Lab-8

NAME: K. MEGAHNA CHOWDARY
NUM : 11599585

The following is a picture of my wiring setup:



The two parts of this system consist of the speaker and the keys.

The speaker is powered by the GPIO pins, and powers the speaker through the resistors. I changed my resistors to 500 Ohm, 1 kOhm, 2kOhm in series, and 4kOhm in series because I did not have enough resistors and wanted to make it double each time to make a better sine wave.

The keys are powered by the battery, goes through the switch, sends the signal to the board, and then goes through the resistor to ground.

The following highlighted code is what I added and changed:

```
#include "esp32-hal-timer.h"

// === 3-bit Sine Wave Table (0 to 7) ===
```

```
const uint8_t SineWave[18] = {

  8, 10, 12, 14, 15, 15, 14, 12, 10, 8, 6, 4, 2, 0, 0, 2, 4, 6

}; -> added a 4 and 6 to make it a complete loop so had to increase to 18

volatile uint8_t Index = 0;

volatile bool waveformActive = false;   // Track if wave generation is on

// === DAC output pins (simulate 3-bit DAC on GPIO 0, 1, 2) ===

const int DAC_PINS[4] = {0, 1, 2, 3};

// === Switch pin (active-high when pressed) ===

const int KEY_PINS[4] = {D7, D8, D6, D5}; // For notes C, D, E, G

const uint16_t timerPeriods[4] = {120, 106, 95, 80}; // Approximate values

// === ESP32 Hardware Timer ===

hw_timer_t * timer = NULL;

// For edge detection

bool lastSwitchState = LOW;

// === Initialize DAC Pins ===

void DAC_Init() {

 for (int i = 0; i < 4; i++) {

   pinMode(DAC_PINS[i], OUTPUT);

   digitalWrite(DAC_PINS[i],  LOW);

 }
```

```
}

// === Output 3-bit Value to DAC ===
void DAC_Out(uint8_t value) {
        for (int i = 0; i < 4; i++) {
                digitalWrite(DAC_PINS[i], (value >> i) C 0x01);
        }
}

// === Timer Interrupt: Update DAC Value ===
void IRAM_ATTR onTimer() {
        if (waveformActive) {
                Index = (Index + 1) % 18; -> had to make it 18 because I changed the array
                DAC_Out(SineWave[Index]);
        }
}

// === Setup Function ===
void setup() {
        Serial.begin(115200); // Start serial communication
        DAC_Init();

        for (int i = 0; i < 4; i++) {
                pinMode(KEY_PINS[i], INPUT_PULLDOWN);
        }
```

```
    // Timer config: 1 tick = 1 µs (80 MHz / 80)
    timer = timerBegin(1000000);  // 1600 Hz = every 625 µs -> changed to 1 microsec
    timerAttachInterrupt(timer, ConTimer);  // No edge param
    //timerAlarm(timer, 625, true, 0); // alarm value, auto-reload, reload count
}

// === Main Loop ===
void loop() {
    for (int i = 0; i < 4; i++) {
        if (digitalRead(KEY_PINS[i]) == HIGH) {
            waveformActive = true;
            timerAlarm(timer, timerPeriods[i], true, 0);
            return;
        }
    }
    waveformActive = false;
    DAC_Out(0); // Silence when no key is pressed
}
```