

Embedded Systems Laboratory

- Lap8:
- มีความรู้ความเข้าใจในรายละเอียดของ PWM และ DAC ของ ESP32
 - การโปรแกรมเพื่อควบคุมการทำงานโดยใช้ PWM และ DAC
 - การโปรแกรมประยุกต์ในการใช้งาน PWM และ DAC ของ ESP32

อุปกรณ์ Lab8

- | | |
|-------------------------------|-----------|
| 1. บอร์ดทดลอง Embedded System | 1 กล่อง |
| 2. สายไฟสำหรับการต่อวงจร | 1 ชุด |
| 3. Laptop หรือ Notebook | 1 เครื่อง |
| 4. LED | 1 ชุด |

8.1 ข้อมูลเบื้องต้น PWM และ DAC ของ ESP32

ให้นักศึกษาร่วมกันหาข้อมูลเพื่อนำมาตอบคำถามข้างล่างดังนี้

คำถาม	คำตอบ
PWM คือ	เทคนิคการแปลงสัญญาณ analog เป็น digital โดยการหาแรงดันเฉลี่ยจากการปรับความกว้างของสัญญาณ
ESP32 มี PWM จำนวนกี่ชุด? ในแต่ละชุดมีจำนวนกี่ Channel?	2 ชุด ชุดละ 8 channel
Ton = 7ms Toff = 12ms 1. Period มีขนาด? 2. ความถี่ กี่ Hz? 3. ในกรณี Vmax = 3.3V Avg Voltage มีขนาดเท่าใด? จงแสดงการคำนวณทั้งหมดในช่องคำตอบ	1. 19 ms 2. $1 / (19 \times 10^{-3}) = 52.63 \text{ Hz}$ 3. $3.3 \times 7 / 19 = 1.922 \text{ V}$
DAC คือ?	การแปลงสัญญาณ analog เป็น digital จากตัวอุปกรณ์
ความละเอียดของ DAC ของ ESP32 มีขนาดกี่บิต?	8 บิต
ในกรณี ไฟเลี้ยงของ ESP32 ขนาด 3.0V ต้องการให้แรงดันออกที่ 3.3V ได้? เพราะ?	ไม่สามารถทำได้ เพราะ แรงดันไฟที่ต้องการมีค่ามากกว่าแรงดันไฟเลี้ยง
ในกรณี ไฟเลี้ยงของ ESP32 ขนาด 3.0V ต้องการให้แรงดันออกที่ 1.3V ต้องการ SET ค่าในโปรแกรมเท่าใด	8 bit = 256 (0-255) $1.3 \times 255 / 3 = 111$

จงอธิบายการทำงาน Function PWM ใน ArduinoIDE

Function	คำตอบ
ledcSetup(channel, freq, resolution);	สร้างและตั้งค่า channel
ledcAttachPin(GPIO, channel)	เชื่อมต่อขา GPIO กับ channel ที่สร้าง
ledcWrite(channel, dutycycle)	ให้ channel ที่เลือกส่งแรงดันตามค่า dutycycle

8.2 Dim LED with PWM

ให้นิสิต นำ ESP32 ต่อวงจรร่วมกับ LED ดังรูป

วงจรทดสอบ

Firmware

```
const int ledPin = 25;
const int freq = 5000;
const int ledChannel = 0;
const int resolution = 8;

void setup() {
  ledcSetup(ledChannel, freq, resolution);
  ledcAttachPin(ledPin, ledChannel);
}

void loop() {
  ledcWrite(ledChannel, ___Duty___ );
  delay(15);
}
```

เปลี่ยนค่า ___Duty___ และทำการทดลอง (ถ่ายรูป)

Duty = 0

Duty = 100

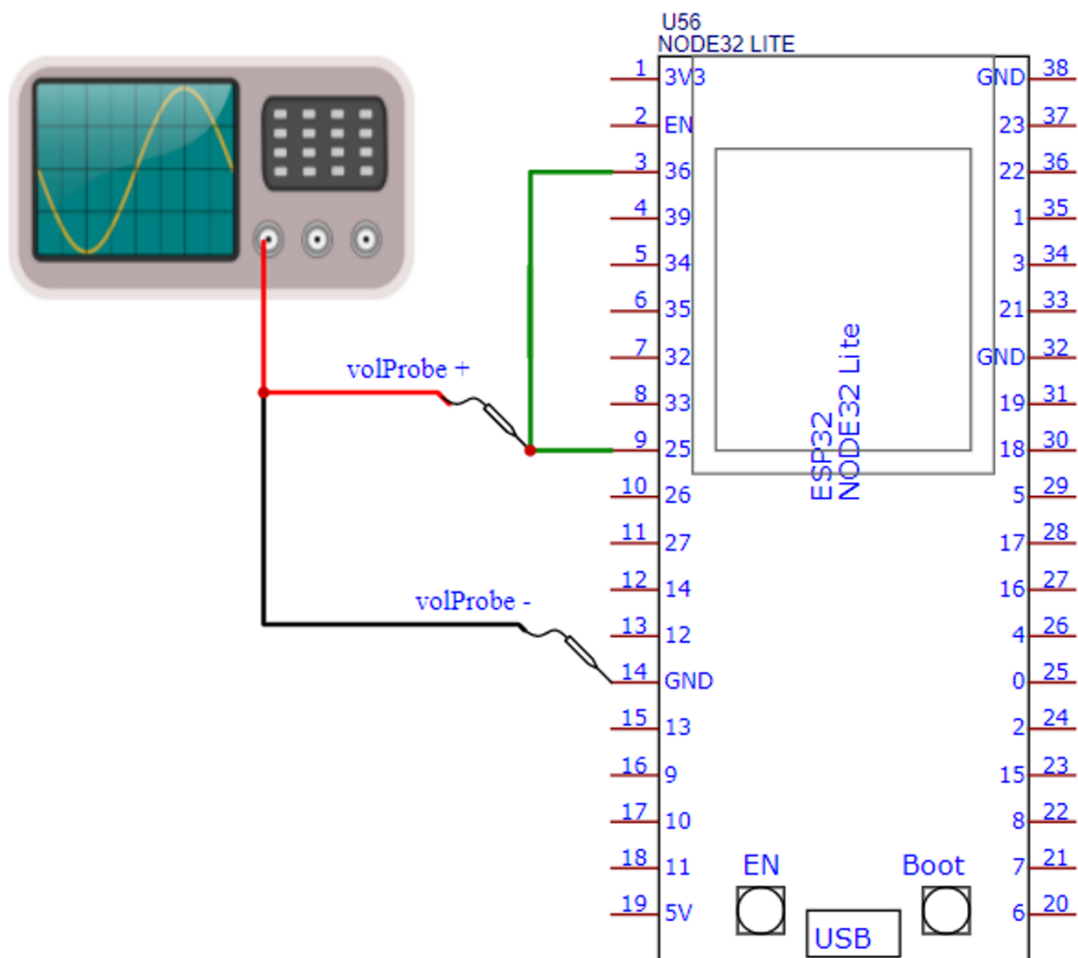
Duty = 255

8.3 DAC in ESP32

จงอธิบายการทำงาน Function DAC ของ ESP32 ใน ArduinoIDE

Function	คำตอบ
<code>dacWrite(uint8_t pin, uint8_t value);</code>	ส่งข้อมูลเป็นแรงดันไฟแบบ dac ไปตามขา pin ที่เลือกไว้

ให้นิสิต นำ ESP32 ต่อวงจรทดสอบ DAC ตามรูป



เนื่องจากการทดลองสัญญาณ Analog เครื่องมือวัดที่สำคัญคือ Oscilloscope แต่ในแลปจะประยุกต์ให้นำเอา ADC มาดูสัญญาณแทน ใช้ในกรณีไม่ต้องการวัดความละเอียดของสัญญาณ รวมถึงไม่ได้ทำงานในโหมด Trigger Signal

จากนั้นให้เขียนโค้ดเพื่อทดสอบการทำงานของ ESP32

```
#define Num_Samples 112
#define MaxWaveTypes 4
int i = 0;
const int adcPin = 36;
int adcValue = 0;

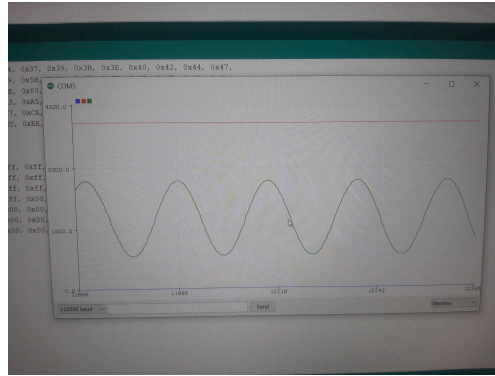
static byte WaveFormTable[MaxWaveTypes][Num_Samples] = {
  // Sin wave
  {
    0x80, 0x83, 0x87, 0x8A, 0x8E, 0x91, 0x95, 0x98, 0x9B, 0x9E, 0xA2, 0xA5, 0xA7, 0xAA, 0xAD, 0xAF,
    0xB2, 0xB4, 0xB6, 0xB8, 0xB9, 0xBB, 0xBC, 0xBD, 0xBE, 0xBF, 0xBF, 0xBF, 0xC0, 0xBF, 0xBF, 0xBF,
    0xBE, 0xBD, 0xBC, 0xBB, 0xB9, 0xB8, 0xB6, 0xB4, 0xB2, 0xAF, 0xAD, 0xAA, 0xA7, 0xA5, 0xA2, 0x9E,
    0x9B, 0x98, 0x95, 0x91, 0x8E, 0x8A, 0x87, 0x83, 0x80, 0x7C, 0x78, 0x75, 0x71, 0x6E, 0x6A, 0x67,
    0x64, 0x61, 0x5D, 0x5A, 0x58, 0x55, 0x52, 0x50, 0x4D, 0x4B, 0x49, 0x47, 0x46, 0x44, 0x43, 0x42,
    0x41, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x41, 0x42, 0x43, 0x44, 0x46, 0x47, 0x49, 0x4B,
    0x4D, 0x50, 0x52, 0x55, 0x58, 0x5A, 0x5D, 0x61, 0x64, 0x67, 0x6A, 0x6E, 0x71, 0x75, 0x78, 0x7C
  },
  // Triangular wave
  {
    0x80, 0x84, 0x89, 0x8D, 0x92, 0x96, 0x9B, 0x9F, 0xA4, 0xA8, 0xAD, 0xB2, 0xB6, 0xBB, 0xBF, 0xC4,
    0xC8, 0xCD, 0xD1, 0xD6, 0xDB, 0xDF, 0xE4, 0xE8, 0xED, 0xF1, 0xF6, 0xFA, 0xFF, 0xFA, 0xF6, 0xF1,
    0xED, 0xE8, 0xE4, 0xDF, 0xDB, 0xD6, 0xD1, 0xCD, 0xC8, 0xC4, 0xBF, 0xBB, 0xB6, 0xB2, 0xAD, 0xA8,
    0xA4, 0x9F, 0x9B, 0x96, 0x92, 0x8D, 0x89, 0x84, 0x7F, 0x7B, 0x76, 0x72, 0x6D, 0x69, 0x64, 0x60,
    0x5B, 0x57, 0x52, 0x4D, 0x49, 0x44, 0x40, 0x3B, 0x37, 0x32, 0x2E, 0x29, 0x24, 0x20, 0x1B, 0x17,
    0x12, 0x0E, 0x09, 0x05, 0x00, 0x05, 0x09, 0x0E, 0x12, 0x17, 0x1B, 0x20, 0x24, 0x29, 0x2E, 0x32,
    0x37, 0x3B, 0x40, 0x44, 0x49, 0x4D, 0x52, 0x57, 0x5B, 0x60, 0x64, 0x69, 0x6D, 0x72, 0x76, 0x7B
  },
  // Sawtooth wave
  {
    0x00, 0x02, 0x04, 0x06, 0x09, 0x0B, 0x0D, 0x10, 0x12, 0x14, 0x16, 0x19, 0x1B, 0x1D, 0x20, 0x22,
    0x24, 0x27, 0x29, 0x2B, 0x2D, 0x30, 0x32, 0x34, 0x37, 0x39, 0x3B, 0x3E, 0x40, 0x42, 0x44, 0x47,
    0x49, 0x4B, 0x4E, 0x50, 0x52, 0x54, 0x57, 0x59, 0x5B, 0x5E, 0x60, 0x62, 0x65, 0x67, 0x69, 0x6B,
    0x6E, 0x70, 0x72, 0x75, 0x77, 0x79, 0x7C, 0x7E, 0x80, 0x82, 0x85, 0x87, 0x89, 0x8C, 0x8E, 0x90,
    0x93, 0x95, 0x97, 0x99, 0x9C, 0x9E, 0xA0, 0xA3, 0xA5, 0xA7, 0xA9, 0xAC, 0xAE, 0xB0, 0xB3, 0xB5,
    0xB7, 0xBA, 0xBC, 0xBE, 0xC0, 0xC3, 0xC5, 0xC7, 0xCA, 0xCC, 0xCE, 0xD1, 0xD3, 0xD5, 0xD7, 0xDA,
    0xDC, 0xDE, 0xE1, 0xE3, 0xE5, 0xE8, 0xEA, 0xEC, 0xEE, 0xF1, 0xF3, 0xF5, 0xF8, 0xFA, 0xFC, 0xFE
  },
  // Square wave
  {
    0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
  }
};

void setup() {
  Serial.begin(115200);
}

void loop() {
  byte wave_type = 0;
  dacWrite(25, WaveFormTable[wave_type][i]);
  i++;
  if (i >= Num_Samples) i = 0;
  adcValue = analogRead(adcPin);
  Serial.printf("0, 4095, %d\n", adcValue);
}
```

เปลี่ยนค่า wave_type เป็น 0 สังเกตรูปร่างคลื่นจากเครื่อง Serial Plotter บันทึกผล

รูปร่างของตัวคลื่น



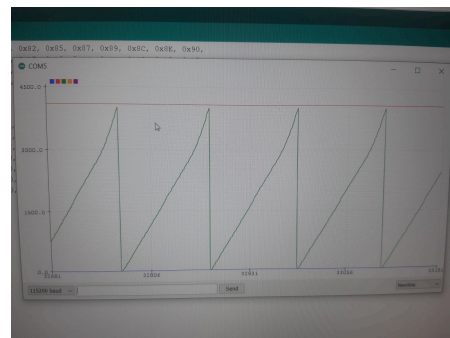
เปลี่ยนค่า wave_type เป็น 1 สังเกตรูปร่างคลื่นจากเครื่อง Serial Plotter บันทึกผล

รูปร่างของตัวคลื่น



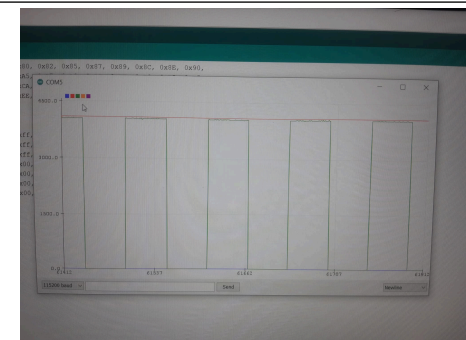
เปลี่ยนค่า wave_type เป็น 2 สังเกตรูปร่างคลื่นจากเครื่อง Serial Plotter บันทึกผล

รูปร่างของตัวคลื่น



เปลี่ยนค่า wave_type เป็น 3 สังเกตรูปร่างคลื่นจากเครื่อง Serial Plotter บันทึกผล

รูปร่างของตัวคลื่น



8.4 โจทย์ Assignment PWM and DAC

ให้นักศึกษาเขียนโปรแกรมให้ ดังนี้

เมื่อ MCU เริ่มต้นการทำงาน ให้ LED ติด (ความสว่างสูงสุด = 255)

1. Choose Mode

สามารถเลือก Mode การทำงานได้ 2 Mode จาก SW7 (ใช้สาย connect 3.3V/GND แทนได้)

1.1 ON Switch จะอยู่ใน Mode PWM (Connect 3.3V)

1.2 OFF Switch จะอยู่ใน Mode DAC (Connect 0V)

2. Mode PWM

2.1. เมื่อกดปุ่ม SW8 LED ค่อยๆสว่างขึ้น 1วินาที จนมีความสว่างสูงสุด

2.2. เมื่อกดปุ่ม SW9 LED ค่อยๆสว่างลง 1วินาที จนไฟ LED ดับไป

3. Mode DAC

3.1. ไม่กด SW8 และ SW9 (1 , 1) สร้างสัญญาณ Sinewave ไปยัง LED และ Oscilloscope/Serial Plotter

3.2. ไม่กด SW8 และ กด SW9 (1 , 0) สร้างสัญญาณ Triangular ไปยัง LED และ Oscilloscope/Serial Plotter

3.3. กด SW8 และ ไม่กด SW9 (0 , 1) สร้างสัญญาณ Sawtooth ไปยัง LED และ Oscilloscope/Serial Plotter

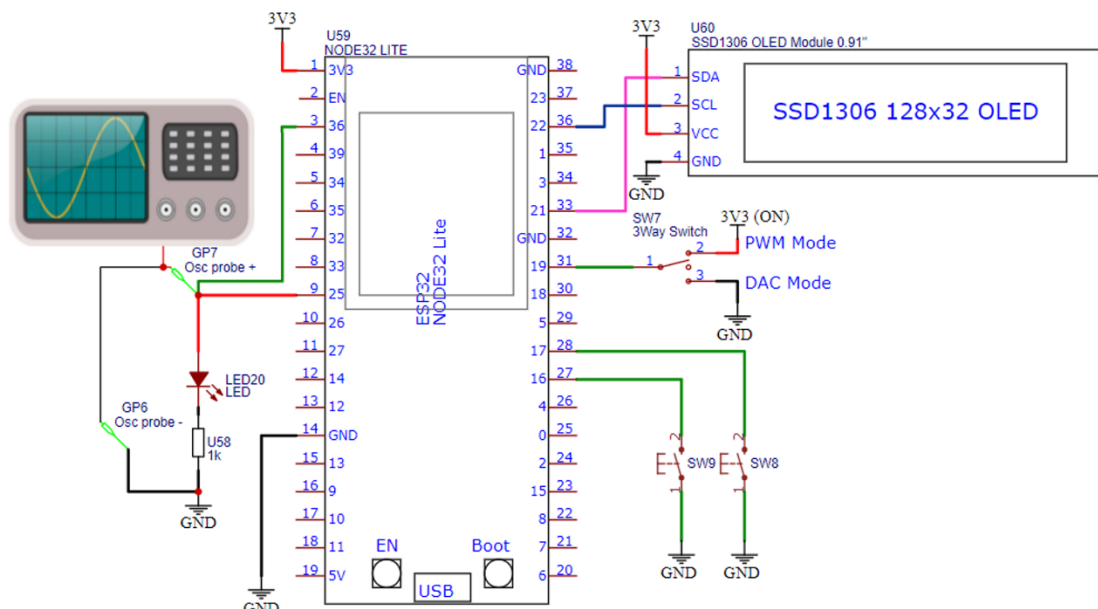
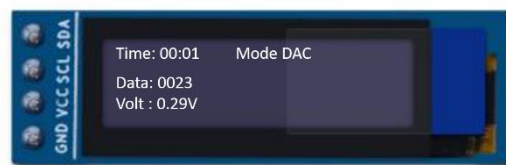
3.4. กด SW8 และ SW9 (0 , 0) สร้างสัญญาณ Square ไปยัง LED และ Oscilloscope/Serial Plotter

4. โดยมีการแสดงผลบน OLED ด้วยดังนี้

4.1. บรรทัดแรก แสดงเวลาในการทำงาน และ Mode

4.2. บรรทัดสอง Data

4.3. บรรทัดสาม ค่าแรงดัน



เขียนโปรแกรมลงในกล่องคำตอบด้านล่าง และถ่ายวิดีโอผลลัพธ์ของโจทย์นี้ Upload ไฟล์ตามหมู่เรียน

ในกรณีตัวหน้ากระดาษไม่พอให้เพิ่มหน้าแทรกในไฟล์แทน

```
#include <SPI.h>
#include <Wire.h>

#define SWMode 19
#define SW8 17
#define SW9 16
#define LED 25
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
Adafruit_SSD1306 display(128, 32, &Wire);
#define Num_Samples 112
#define MaxWaveTypes 4
#define adcPin 36

byte wave_type = 0;
long lasttime1000ms = 0;
long lasttime10ms = 0;
int mintime = 0, sectime = 0;
bool stateToggle = true, ramp = true;
int pwmdata = 0;
int i = 0, adcValue = 0;
static byte WaveFormTable[MaxWaveTypes][Num_Samples] = {
  // Sin wave
  {
    0x80, 0x83, 0x87, 0x8A, 0x8E, 0x91, 0x95, 0x98, 0x9B, 0x9E, 0xA2, 0xA5, 0xA7, 0xAA, 0xAD, 0xAF,
    0xB2, 0xB4, 0xB6, 0xB8, 0xB9, 0xBB, 0xBC, 0BD, 0xBE, 0xBF, 0xBF, 0xBF, 0xC0, 0xBF, 0xBF, 0xBF,
    0xBE, 0xBD, 0xBC, 0xBB, 0xB9, 0xB8, 0xB6, 0xB4, 0xB2, 0xAF, 0xAD, 0xAA, 0xA7, 0xA5, 0xA2, 0x9E,
    0x9B, 0x98, 0x95, 0x91, 0x8E, 0x8A, 0x87, 0x83, 0x80, 0x7C, 0x78, 0x75, 0x71, 0x6E, 0x6A, 0x67,
    0x64, 0x61, 0x5D, 0x5A, 0x58, 0x55, 0x52, 0x50, 0x4D, 0x4B, 0x49, 0x47, 0x46, 0x44, 0x43, 0x42,
    0x41, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x41, 0x42, 0x43, 0x44, 0x46, 0x47, 0x49, 0x4B,
    0x4D, 0x50, 0x52, 0x55, 0x58, 0x5A, 0x5D, 0x61, 0x64, 0x67, 0x6A, 0x6E, 0x71, 0x75, 0x78, 0x7C
  },
  // Triangular wave
  {
    0x80, 0x84, 0x89, 0x8D, 0x92, 0x96, 0x9B, 0x9F, 0xA4, 0xA8, 0xAD, 0xB2, 0xB6, 0xBB, 0xBF, 0xC4,
    0xC8, 0xCD, 0xD1, 0xD6, 0xDB, 0xDF, 0xE4, 0xE8, 0xED, 0xF1, 0xF6, 0xFA, 0xFF, 0xFA, 0xF6, 0xF1,
    0xED, 0xE8, 0xE4, 0xDF, 0xDB, 0xD6, 0xD1, 0xCD, 0xC8, 0xC4, 0xBF, 0xBB, 0xB6, 0xB2, 0xAD, 0xA8,
    0xA4, 0x9F, 0x9B, 0x96, 0x92, 0x8D, 0x89, 0x84, 0x7F, 0x7B, 0x76, 0x72, 0x6D, 0x69, 0x64, 0x60,
    0x5B, 0x57, 0x52, 0x4D, 0x49, 0x44, 0x40, 0x3B, 0x37, 0x32, 0x2E, 0x29, 0x24, 0x20, 0x1B, 0x17,
    0x12, 0x0E, 0x09, 0x05, 0x00, 0x05, 0x09, 0x0E, 0x12, 0x17, 0x1B, 0x20, 0x24, 0x29, 0x2E, 0x32,
    0x37, 0x3B, 0x40, 0x44, 0x49, 0x4D, 0x52, 0x57, 0x5B, 0x60, 0x64, 0x69, 0x6D, 0x72, 0x76, 0x7B
  },
}
```

เขียนโปรแกรมลงในกล่องคำตอบด้านล่าง และถ่ายวิดีโอผลลัพธ์ของโจทย์นี้ Upload ไฟล์ตามหมู่เรียน

ในกรณีตัวหน้ากระดาษไม่พอให้เพิ่มหน้าแทรกในไฟล์แทน

```
// Sawtooth wave
{ 0x00, 0x02, 0x04, 0x06, 0x09, 0x0B, 0x0D, 0x10, 0x12, 0x14, 0x16, 0x19, 0x1B, 0x1D, 0x20, 0x22,
  0x24, 0x27, 0x29, 0x2B, 0x2D, 0x30, 0x32, 0x34, 0x37, 0x39, 0x3B, 0x3E, 0x40, 0x42, 0x44, 0x47,
  0x49, 0x4B, 0x4E, 0x50, 0x52, 0x54, 0x57, 0x59, 0x5B, 0x5E, 0x60, 0x62, 0x65, 0x67, 0x69, 0x6B,
  0x6E, 0x70, 0x72, 0x75, 0x77, 0x79, 0x7C, 0x7E, 0x80, 0x82, 0x85, 0x87, 0x89, 0x8C, 0x8E, 0x90,
  0x93, 0x95, 0x97, 0x99, 0x9C, 0x9E, 0xA0, 0xA3, 0xA5, 0xA7, 0xA9, 0xAC, 0xAE, 0xB0, 0xB3, 0xB5,
  0xB7, 0xBA, 0xBC, 0xBE, 0xC0, 0xC3, 0xC5, 0xC7, 0xCA, 0xCC, 0xCE, 0xD1, 0xD3, 0xD5, 0xD7, 0xDA,
  0xDC, 0xDE, 0xE1, 0xE3, 0xE5, 0xE8, 0xEA, 0xEC, 0xEE, 0xF1, 0xF3, 0xF5, 0xF8, 0xFA, 0xFC, 0xFE
},
// Square wave
{ 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
  0x00
}
};

void setup() {
  Serial.begin(115200);
  pinMode(SWMode, INPUT);
  pinMode(SW8, INPUT_PULLUP);
  pinMode(SW9, INPUT_PULLUP);
  ledcSetup(0, 1000, 10); // Channel 0, Freq=1Khz, bit_res 10bit (0-1023)
  ledcAttachPin(LED, 0);
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(SSD1306_WHITE);
  display.setCursor(0, 0);
  display.printf("LAB8 Analog Out");
  display.display();
  delay(1000);
}

void loop() {
  // loop 10ms
  if (millis() - lasttime10ms >= 10) {
    lasttime10ms = millis();
    stateToogle = !stateToogle;
    if (digitalRead(SWMode) == LOW) //DAC
    { if(digitalRead(SW8) == HIGH&&digitalRead(SW9) == HIGH){wave_type = 0;}
      else if(digitalRead(SW8) == HIGH&&digitalRead(SW9) == LOW){wave_type = 1;}
      else if(digitalRead(SW8) == LOW&&digitalRead(SW9) == HIGH){wave_type = 2;}
      else{wave_type = 3;}
      dacWrite(25, WaveFormTable[wave_type][i]);
      i++;
      if (i >= Num_Samples) i = 0;
      adcValue = analogRead(adcPin);
      Serial.printf("0, 4095, %d\n", adcValue);
      display.clearDisplay();
      display.setTextSize(1);
      display.setTextColor(SSD1306_WHITE);
      display.setCursor(0, 0);
      display.printf("Time:%02d:%02d Mode DAC", mintime, sectime);
      display.setCursor(0, 10);
      display.printf("Data:%04d", adcValue);
      display.setCursor(0, 20);
      display.printf("Volt:%.2f", adcValue * 3.3 / 4095);
      display.display();
    }
  }
}
```


เขียนโปรแกรมลงในกล่องคำตอบด้านล่าง และถ่ายวิดีโอผลลัพธ์ของโจทย์นี้ Upload ไฟล์ตามหมู่เรียน

ในกรณีตัวหน้ากระดาษไม่พอให้เพิ่มหน้าแทรกในไฟล์แทน

```
if (digitalRead(SWMode) == HIGH) //PWM
{ // Setting LED2
  ledcWrite(0, pwmdata);
  if (digitalRead(SW8) == LOW) {
    ramp = true;
  }
  if (digitalRead(SW9) == LOW) {
    ramp = false;
  }
  if (ramp == true) {
    pwmdata = (pwmdata + (1023 / 100));
  }
  else {
    pwmdata = (pwmdata - (1023 / 100));
  }
  if (pwmdata > 1023) {
    pwmdata = 1023;
  }
  if (pwmdata < 0) {
    pwmdata = 0;
  }
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(SSD1306_WHITE);
  display.setCursor(0, 0);
  display.printf("Time:%02d:%02d Mode PWM", mintime, sectime);
  display.setCursor(0, 10);
  display.printf("Data:%04d", pwmdata);
  display.setCursor(0, 20);
  display.printf("Volt:%.2f", pwmdata * 3.3 / 1023);
  display.display();
}

}

if (millis() - lasttime1000ms >= 1000) {
  lasttime1000ms = millis();
  stateToogle = !stateToogle;
  sectime++;
  if (sectime >= 60) {
    sectime = 0;
    mintime++;
  }
  if (mintime >= 60) {
    mintime = 0;
  }
}
}
```