

## Embedded Systems Laboratory

- Lap3:
- มีความรู้ความเข้าใจในรายละเอียดของ Interrupt ของ ESP32
  - การโปรแกรมเพื่อควบคุมการทำงานโดยใช้ Timer Interrupt
  - การโปรแกรมเพื่อควบคุมการทำงานโดยใช้ External Interrupt

## อุปกรณ์ Lab3

1. ชุดอุปกรณ์ Embedded System 1 ชุด
2. Laptop หรือ Notebook 1 เครื่อง (Window/Mac/Linux ในการสอนจะใช้ Window เป็น OS)

## 3.1 ข้อมูลเบื้องต้น External Interrupt ของ ESP32

ให้นักศึกษาร่วมกันหาข้อมูลเพื่อนำมาตอบคำถามข้างล่างดังนี้

จงตอบคำถามทั่วไปสำหรับ PIN ของ ESP32

คำถาม	คำตอบ
ESP32 GPIO อะไรบ้างที่สามารถตรวจจับเหตุการณ์ Interrupt ภายนอกได้ (External Interrupt)	GPIO ทั้งหมดยกเว้น GPIO 6-11
ESP32 GPIO อะไรบ้างที่ไม่สามารถตรวจจับเหตุการณ์ Interrupt ภายนอกได้ (External Interrupt)	GPIO 6-11
จงเขียน เหตุการณ์ที่ ESP32 สามารถตรวจจับได้ทั้งหมดและอธิบายแต่ละเหตุการณ์ เช่น CHANGE ตรวจจับเมื่อมีการเปลี่ยนแปลงของค่าสัญญาณจาก HIGH เป็น LOW และ จาก LOW เป็น HIGH .....	FALLING จับสัญญาณจาก HIGH เป็น LOW RAISING จับสัญญาณจาก LOW เป็น HIGH CHANGE จับสัญญาณจาก LOW เป็น HIGH และ HIGH เป็น LOW HIGH จับสัญญาณที่สถานะ HIGH LOW จับสัญญาณที่สถานะ LOW

## 3.2 Workshop in External Interrupt

ทดลองต่อวงจรตามรูป และทดลองเขียนโปรแกรมให้ Output ตามรูป

```
SW1 press 1 times
SW1 press 2 times
SW2 press 1 times
SW2 press 2 times
SW2 press 3 times
. . . . .
```

เติมส่วนของ Code ที่หายไปให้ถูกต้อง

```
#define SW1 4
#define SW2 16

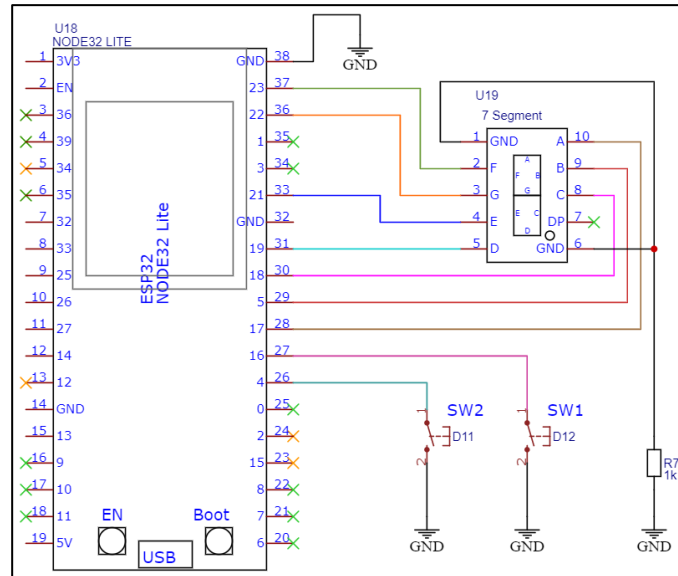
int KeyPressesSW1 = 0;
bool statusSW1 = false;
int KeyPressesSW2 = 0;
bool statusSW2 = false;

void IRAM_ATTR isrSW1() {
    KeyPressesSW1 += 1;
    statusSW1 = true;
}

void IRAM_ATTR isrSW2() {
    KeyPressesSW2 += 1;
    statusSW2 = true;
}

void setup() {
    Serial.begin(115200);
    pinMode(__SW1, INPUT_PULLUP);
    pinMode(__SW2, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(4), isrSW2, FALLING);
    attachInterrupt(digitalPinToInterrupt(16), isrSW1, FALLING);
}

void loop() {
    if (statusSW1) {
        Serial.printf("SW1 press %u times\n", KeyPressesSW1);
        statusSW1 = false;
    }
    if (statusSW2) {
        Serial.printf("SW2 press %u times\n", KeyPressesSW2);
        statusSW2 = false;
    }
}
```



จงอธิบายการทำงาน ตามที่เข้าใจ

เมื่อกด switch ทำให้สัญญาณตรงตามเงื่อนไขการ interrupt จึงเริ่มทำงานฟังก์ชัน interrupt ของ switch นั้นๆ

### 3.3 ข้อมูลเบื้องต้น Timer Interrupt ของ ESP32

จงตอบคำถามสำหรับ Timer Interrupt ใน ESP32

คำถาม	คำตอบ
ESP32 มี Timer จำนวนเท่าไร? นับเฉพาะ Hardware Timer	2
Timer ESP32 มีขนาดกี่ bit	64 bit
Prescaler สำหรับ ESP32 มีขนาดกี่ bit	16 bit
ในกรณีต้องการ Timer Interrupt ทุก 1 วินาที โดยมีรายละเอียดดังนี้ (จงแสดงวิธีการทำในช่องคำตอบ) <ul style="list-style-type: none"> <li>- Clock 80Mhz</li> <li>- Prescaler 40000</li> </ul> ต้อง Set Counter เท่าใดถึงจะเกิด Interrupt ทุกๆ 1 วินาที	2000

### 3.4 Workshop in Timer Interrupt

ทดลองต่อวงจรตามรูป และทดลองเขียนโปรแกรมให้ Output โดยให้ไฟ LED on Board (GPIO2) กระพริบทุกๆ 1 วินาที (ติด 1 วินาที ดับ 1 วินาที สลับกันไปมา) โดยใช้ Timer Interrupt  
 เติมส่วนของ Code ที่หายไปให้ถูกต้อง

```
#define LED_PIN 2    // GPIO-2 (built-in LED onboard)
#define LED_ON  LOW
#define LED_OFF HIGH
hw_timer_t *timer = NULL;
bool led_state = false;

void __IRAM_ATTR__ timer_callback( void ) {
    led_state = !led_state;
    digitalWrite( LED_PIN, led_state ? LED_ON : LED_OFF );
}

void setup() {
    Serial.begin( 115200 );
    Serial.println( "\n\n\n" );
    pinMode( LED_PIN, OUTPUT );
    digitalWrite( LED_PIN, led_state ? LED_ON : LED_OFF );

    timer = timerBegin( 0, 8000, true );
    timerAlarmWrite( timer, 10000, true );
    timerAttachInterrupt( timer, &timer_callback, true );
    timerAlarmEnable( timer );
}

void loop() {
}
```

### 3.5 Upgrade your Firmware

นำเอาการทดลองใน Lab2.3 มาปรับปรุงโดยใช้ Interrupt (Timer และ External) เข้ามาใช้ควบคุมการทำงาน

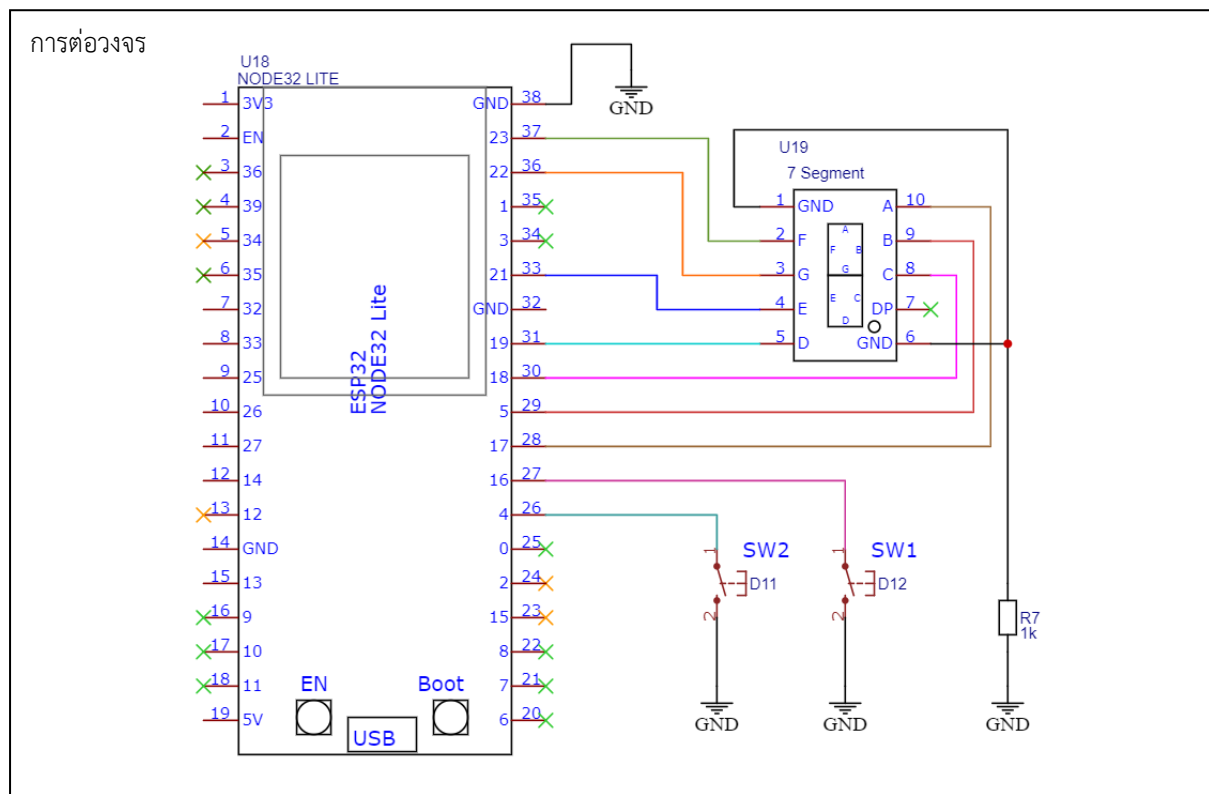
จะมี Switch อยู่ 2 ปุ่ม (GPIO4, GPIO16) และต่อกับ 7Segment Common Cathode ตามรูป

1. เมื่อไม่ได้กดปุ่มใดๆ ตัว 7Segment จะกระพริบเลข 0 (ติด 0.5วิ/ดับ 0.5วิ)
2. เมื่อกดปุ่ม D11 ตัว 7Segment จะกระพริบเลข 1 (ติด 0.5วิ/ดับ 0.5วิ)
3. เมื่อกดปุ่ม D12 ตัว 7Segment จะกระพริบเลข 2 (ติด 0.5วิ/ดับ 0.5วิ)
4. เมื่อกดปุ่ม D11 และ D12 พร้อมกัน ตัว 7Segment จะกระพริบเลข 3 (ติด 0.5วิ/ดับ 0.5วิ)

และ LED on Board (GPIO2) จะกระพริบติดดับ (ติด 0.5วิ/ดับ 0.5วิ) ตลอดการทำงานของ Firmware นี้

รายละเอียดตามตาราง

PIN	Status	7 Segment
Switch D11 & Switch D12	1/1	Show 0
Switch D11	0	Show 1
Switch D12	0	Show 2
Switch D11 & Switch D12	0/0	Show 3
LED on board (GPIO2)	Toggle 0.5sec	



เขียนโปรแกรมลงในกล่องคำตอบด้านล่าง และถ่ายวิดีโอผลลัพธ์ของโจทย์นี้ Upload ไฟล์ตามหมู่เรียน

โจทย์นี้ต้องใช้ Interrupt ทั้งหมดทั้ง External + Timer Interrupt (ใน function loop ห้ามใส่ code)

```
#define LED_PIN 2 // GPIO-2 (built-in LED onboard)
#define LED_ON LOW
#define LED_OFF HIGH
hw_timer_t *timer = NULL;
bool led_state = false;
#define A 17 // free memory
#define B 5
#define C 18
#define D 19
#define E 21
#define F 23
#define G 22
#define SW1 4
#define SW2 16
int num=0;

void SW()
{ if(digitalRead(SW1)==HIGH&&digitalRead(SW2)==HIGH){num=0;}
  else if(digitalRead(SW1)==LOW&&digitalRead(SW2)==HIGH){num=1;}
  else if(digitalRead(SW1)==HIGH&&digitalRead(SW2)==LOW){num=2;}
  else{num=3;}
}

void IRAM_ATTR isrSW1()
{ SW();
}

void IRAM_ATTR isrSW2()
{ SW();
}

void off(void){
  // OFF LED All Segment
  digitalWrite(A, LOW);
  digitalWrite(B, LOW);
  digitalWrite(C, LOW);
  digitalWrite(D, LOW);
  digitalWrite(E, LOW);
  digitalWrite(F, LOW);
  digitalWrite(G, LOW);
  //Serial.println("off");
}

void zero(void) {
  // LED 0
  digitalWrite(A, HIGH);
  digitalWrite(B, HIGH);
  digitalWrite(C, HIGH);
  digitalWrite(D, HIGH);
  digitalWrite(E, HIGH);
  digitalWrite(F, HIGH);
  digitalWrite(G, LOW);
  //Serial.println("0");
}

void one(void) {
  // LED 1
  digitalWrite(A, LOW);
  digitalWrite(B, HIGH);
  digitalWrite(C, HIGH);
  digitalWrite(D, LOW);
  digitalWrite(E, LOW);
  digitalWrite(F, LOW);
  digitalWrite(G, LOW);
  //Serial.println("1");
}

void two(void) {
  // LED 2
  digitalWrite(A, HIGH);
  digitalWrite(B, HIGH);
  digitalWrite(C, LOW);
  digitalWrite(D, HIGH);
  digitalWrite(E, HIGH);
  digitalWrite(F, LOW);
  digitalWrite(G, HIGH);
  //Serial.println("2");
}

void three(void){
  // LED 3
  digitalWrite(A, HIGH);
  digitalWrite(B, HIGH);
  digitalWrite(C, HIGH);
  digitalWrite(D, HIGH);
  digitalWrite(E, LOW);
  digitalWrite(F, LOW);
  digitalWrite(G, HIGH);
  //Serial.println("3");
}

void IRAM_ATTR timer_callback( void )
{ led_state = !led_state;
  digitalWrite( LED_PIN, led_state ? LED_ON : LED_OFF );
  if(led_state == true)
  { if(num==0)
    { zero();
    }else if(num==1)
    { one();
    }
    else if(num==2)
    { two();
    }else
    { three();
    }
  }else
  { off();
  }
}

void setup()
{ Serial.begin( 115200 );
  Serial.println( "\n\n\n" );
  pinMode( LED_PIN, OUTPUT );
  digitalWrite( LED_PIN, led_state ? LED_ON : LED_OFF );

  timer = timerBegin( 0, 8000, true );
  timerAlarmWrite( timer, 5000, true );
  timerAttachInterrupt( timer, &timer_callback, true );
  timerAlarmEnable( timer );
  pinMode(SW1, INPUT_PULLUP);
  pinMode(SW2, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(4), isrSW2, CHANGE);
  attachInterrupt(digitalPinToInterrupt(16), isrSW1, CHANGE);
  pinMode(A, OUTPUT); // SegA
  pinMode(B, OUTPUT); // SegB
  pinMode(C, OUTPUT); // SegC
  pinMode(D, OUTPUT); // SegD
  pinMode(E, OUTPUT); // SegE
  pinMode(F, OUTPUT); // SegF
  pinMode(G, OUTPUT); // SegG
}

void loop()
{
}
```