

Embedded Systems Laboratory

- Lap2:
- มีความรู้ความเข้าใจในรายละเอียดของ Peripheral ของ ESP32
 - สามารถออกแบบการ Control GPIO ทั้ง Input/Output ของ MCU ได้
 - การต่อวงจรอิเล็กทรอนิกส์ และสมองกลฝังตัว
 - การโปรแกรมเพื่อควบคุมการทำงานทั้ง Input/Output ของ MCU ได้

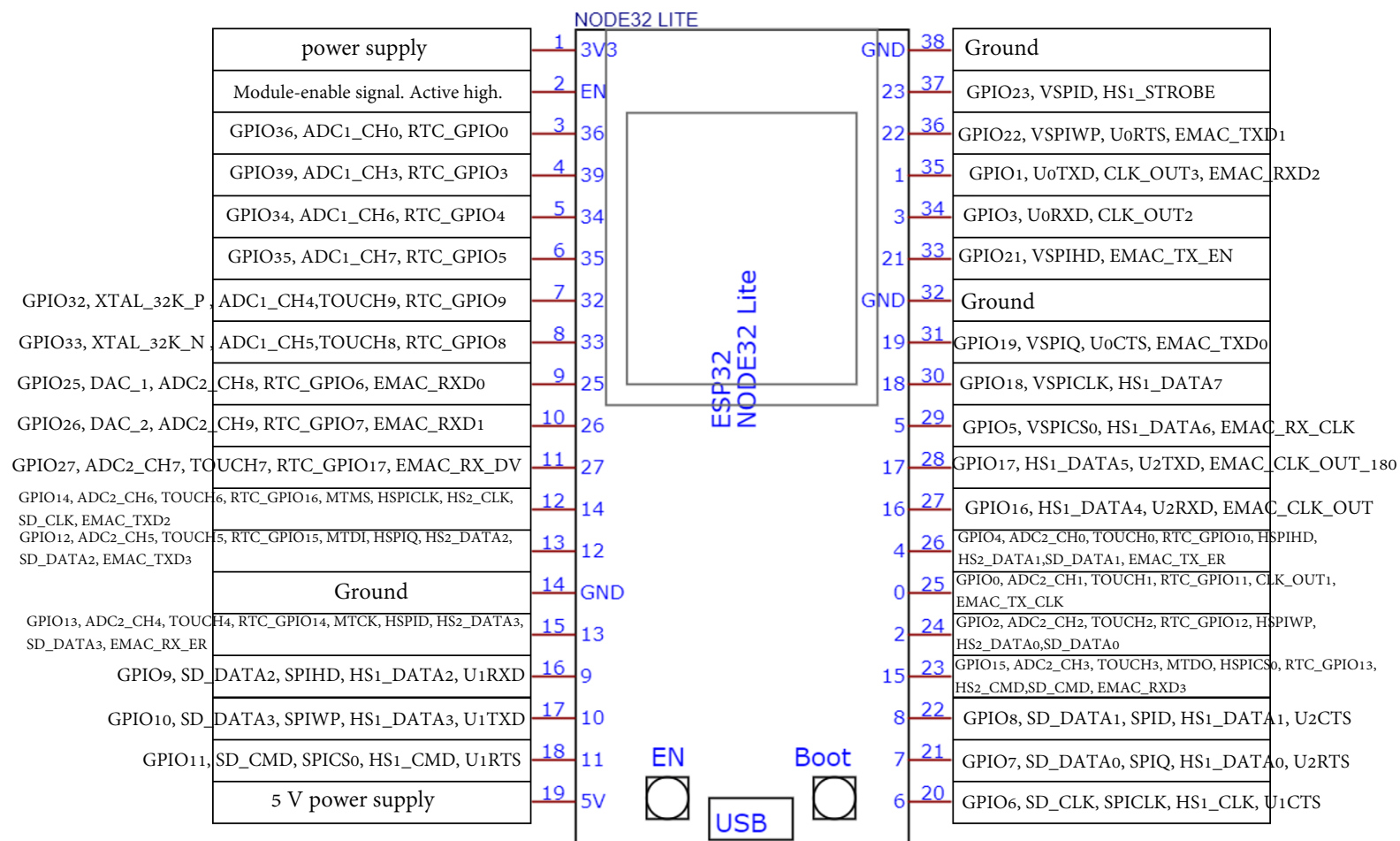
อุปกรณ์ Lab2

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

2.1 ข้อมูลเบื้องต้น Input/Output ของ ESP32

ให้นักศึกษาร่วมกันหาข้อมูลเพื่อนำมาตอบคำถามข้างล่างดังนี้

ให้ระบุรายละเอียดของแต่ละขาของ ESP32 ที่สามารถนำไปใช้งานอะไรได้บ้าง เช่น (input/output/ADC/UART1/....)



จงตอบคำถามทั่วไปสำหรับ PIN ของ ESP32

คำถาม	คำตอบ
ESP32 มีจำนวนขา GPIO ทั้งหมดเท่าไร	32
ESP32 มีขา PIN ที่สามารถนำไปใช้งานได้ทั้งหมดกี่ขา (รวมทุก Peripheral) (ขาที่ใช้งานไม่ได้คือ? อธิบายเหตุผล)	32 GPIO 6-11 เพราะเชื่อมกับ integrated spi flash
เฉพาะ INPUT มีกี่ขา GPIO ที่สามารถใช้งานได้ (พร้อมระบุ GPIO)	25 ขา GPIO 0, 2-5, 12-19, 21-23, 25-27, 32-36, 39
ขา PIN GPIO ไหนบ้างที่ใช้ INPUT ได้อย่างเดียว (พร้อมระบุ GPIO)	GPIO 3, 34-36, 39
เฉพาะ OUTPUT มีกี่ขา PIN ที่สามารถใช้งานได้ (พร้อมระบุ GPIO)	21 ขา GPIO 0-2, 4-5, 12-19, 21-23, 25-27, 32-33
เพราะเหตุใดบางขาถึงไม่สามารถใช้งานเป็น OUTPUT ได้	ไม่มีการทำงานภายในแบบ pull-up หรือ pull-down

Hint: http://www.weigu.lu/microcontroller/tips_tricks/esp32_tips_tricks/index.html

ข้อมูลของขา PIN เพิ่มเติม

PIN	รายละเอียดขา PIN	สามารถนำมาใช้งาน INPUT/ OUTPUT ได้? มีข้อจำกัด? เพราะเหตุใด
GPIO0	input pull up / output	ใช้สำหรับโหมด bootloader หรือ flashing
GPIO1	output	เปลี่ยนสถานะเป็น high หรือ pwm signal
GPIO3	input high at boot	เปลี่ยนสถานะเป็น high หรือ pwm signal
GPIO6	ไม่ควรใช้	เชื่อมกับ integrated SPI flash
GPIO7	ไม่ควรใช้	เชื่อมกับ integrated SPI flash
GPIO8	ไม่ควรใช้	เชื่อมกับ integrated SPI flash
GPIO9	ไม่ควรใช้	เชื่อมกับ integrated SPI flash
GPIO10	ไม่ควรใช้	เชื่อมกับ integrated SPI flash
GPIO11	ไม่ควรใช้	เชื่อมกับ integrated SPI flash
GPIO12	boot fail if pull high	ใช้สำหรับโหมด bootloader หรือ flashing
GPIO34	input only	ไม่มีการทำงานภายในแบบ pull-up หรือ pull-down
GPIO35	input only	ไม่มีการทำงานภายในแบบ pull-up หรือ pull-down
GPIO36	input only	ไม่มีการทำงานภายในแบบ pull-up หรือ pull-down
GPIO39	input only	ไม่มีการทำงานภายในแบบ pull-up หรือ pull-down

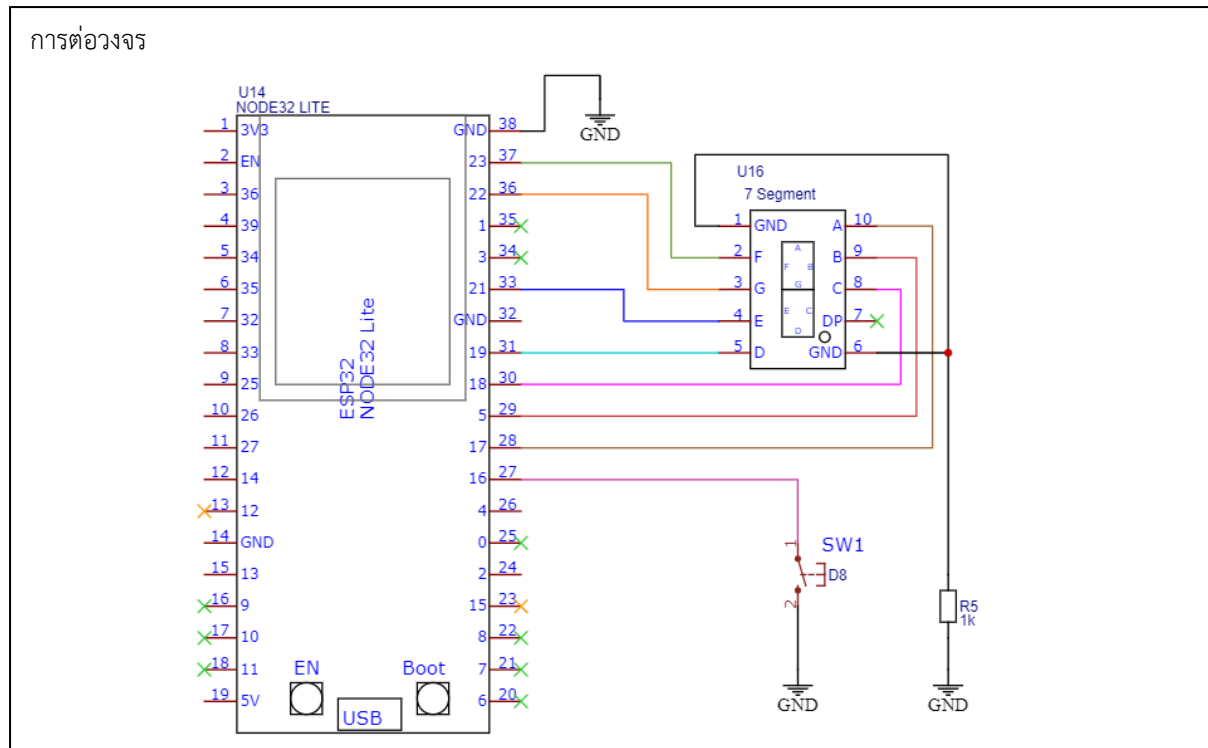
Hint: <https://randomnerdtutorials.com/esp32-pinout-reference-gpios/>

2.2 ทดลองเขียนควบคุม 7 Segment

จะเขียนโปรแกรมควบคุมดังนี้ (60XXXXXX_0202.ino)

จะมี Switch อยู่ 1 ปุ่ม และต่อกับ 7Segment Common Cathode

- เมื่อไม่ได้กดปุ่มใดๆ ตัว 7Segment จะกระพริบเลข (0 ติด 0.5วิ/ดับ 0.5วิ)
- เมื่อกดปุ่ม SW1 ตัว 7Segment จะกระพริบเลข 1/2/3/4/5/6/7/8/9/A/B/C/D/E/F



งตอบคำถามในตาราง

ตัวเลข/อักษร ที่แสดง	PIN GPIO ไหนที่ต้องเป็น HIGH LOW
0	27,5,18,19,21,31
1	5,18
2	27,5,19,21,22
3	27,5,18,19,22
4	5,18,23,22
5	27,18,19,23,22
6	27,18,19,21,23,22
7	27,5,18
8	27,5,18,19,21,23,22
9	27,5,18,19,23,22
A	27,5,18,19,21,22
F	27,21,23,22

เขียนโปรแกรมลงในกล่องคำตอบด้านล่าง

```

//Global
bool stateloop = true;
int state7Seg = 1;

void off(void)
{ digitalWrite(27, HIGH);
  digitalWrite(5, HIGH);
  digitalWrite(18, HIGH);
  digitalWrite(19, HIGH);
  digitalWrite(21, HIGH);
  digitalWrite(23, HIGH);
  digitalWrite(22, HIGH);
  Serial.println("off");
}

void zero(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(23, LOW);
  Serial.println("0");
}

void one(void)
{ digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  Serial.println("1");
}

void two(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(22, LOW);
  Serial.println("2");
}

void three(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(22, LOW);
  Serial.println("3");
}

void four(void)
{ digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(23, LOW);
  digitalWrite(22, LOW);
  Serial.println("4");
}

void five(void)
{ digitalWrite(27, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(23, LOW);
  digitalWrite(22, LOW);
  Serial.println("5");
}

void six(void)
{ digitalWrite(27, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(23, LOW);
  digitalWrite(22, LOW);
  Serial.println("6");
}

void seven(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  Serial.println("7");
}

void eight(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(23, LOW);
  digitalWrite(22, LOW);
  Serial.println("8");
}

void nine(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(23, LOW);
  digitalWrite(22, LOW);
  Serial.println("9");
}

void A(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(22, LOW);
  Serial.println("a");
}

void B(void)
{ digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(23, LOW);
  digitalWrite(22, LOW);
  Serial.println("b");
}

void C(void)
{ digitalWrite(27, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(23, LOW);
  Serial.println("c");
}

void D(void)
{ digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(22, LOW);
  Serial.println("d");
}

void E(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(23, LOW);
  digitalWrite(22, LOW);
  Serial.println("e");
}

void f(void)
{ digitalWrite(27, LOW);
  digitalWrite(21, LOW);
  digitalWrite(23, LOW);
  digitalWrite(22, LOW);
  Serial.println("f");
}

void state(int num)
{ if(num==1){one();}
  else if(num==2){two();}
  else if(num==3){three();}
  else if(num==4){four();}
  else if(num==5){five();}
  else if(num==6){six();}
  else if(num==7){seven();}
  else if(num==8){eight();}
  else if(num==9){nine();}
  else if(num==10){A();}
  else if(num==11){B();}
  else if(num==12){C();}
  else if(num==13){D();}
  else if(num==14){E();}
  else{f();}
}

void setup() {
  Serial.begin(115200);
  Serial.println("Start Firmware");
  // SW GPIO16
  pinMode(16, INPUT_PULLUP);
  // Seven Segment
  pinMode(27, OUTPUT); // SegA
  pinMode(5, OUTPUT); // SegB
  pinMode(18, OUTPUT); // SegC
  pinMode(19, OUTPUT); // SegD
  pinMode(21, OUTPUT); // SegE
  pinMode(23, OUTPUT); // SegF
  pinMode(22, OUTPUT); // SegG
  off(); // OFF LED ALL Segment
}

void loop()
{ if(digitalRead(16) == HIGH)
  { if(stateloop == true)
    { zero(); // 7Seg Display 0
    }else
    { off(); // 7Seg off Display
    }
    state7Seg = 1;
  }
  else
  { if(stateloop == true)
    { state7Seg++;
      state7Seg+=1;
    }else
    { off(); // 7Seg off Display
    }
  }
  stateloop = !stateloop;
  delay(500);
}

```

*<https://wokwi.com/projects/337596048885154388>

2.3 ทดลองเขียนควบคุม 7 Segment 2Bit input

จงเขียนโปรแกรมควบคุมดังนี้ (60XXXXXX_0203.ino)

จะมี Switch อยู่ 2 ปุ่ม และต่อกับ 7Segment Common Cathode และแสดงการกระพริบดังตารางด้านล่าง

โดยที่ไม่กดปุ่มจะเป็น Logic 0 ทั้ง Switch SW1 และ SW2

รายละเอียดตามตาราง

Switch SW1	Switch SW2	7 Segment
0	0	Show 0
0	1	Show 1
1	0	Show 2
1	1	Show 3

การกระพริบของ 7 Segment รายละเอียดดังนี้

<start Loop>

7Seg กระพริบ 1sec โห้ค่าตาม Input 3 ครั้ง

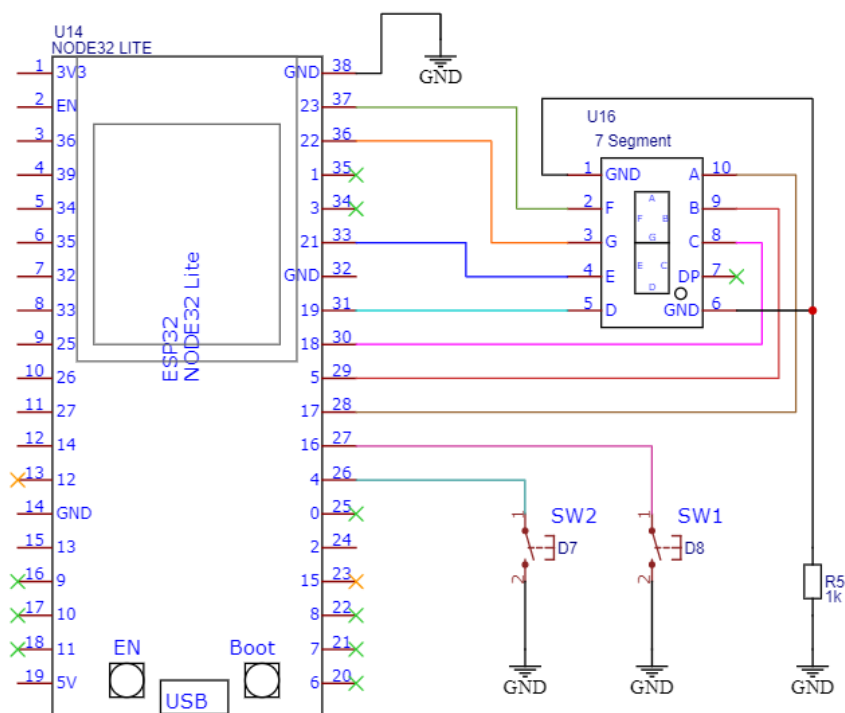
ไฟดับ 2วินาที

7Seg กระพริบ 0.5sec โห้ค่าตาม Input 6 ครั้ง ไฟดับ 2วินาที

<end Loop>

*** เมื่อเปลี่ยน input ค่าตัวเลขจะเปลี่ยนทันทีในรอบกระพริบ

ให้วาดการต่อวงจรด้านล่าง



เขียนโปรแกรมลงในกล่องคำตอบด้านล่าง

```

bool stateloop = true, loo = true;
int state7Seg = 1;

void off(void)
{ digitalWrite(27, HIGH);
  digitalWrite(5, HIGH);
  digitalWrite(18, HIGH);
  digitalWrite(19, HIGH);
  digitalWrite(21, HIGH);
  digitalWrite(23, HIGH);
  digitalWrite(22, HIGH);
  Serial.println("off");
}

void zero(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(23, LOW);
  Serial.println("0");
}

void one(void)
{ digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  Serial.println("1");
}

void two(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(19, LOW);
  digitalWrite(21, LOW);
  digitalWrite(22, LOW);
  Serial.println("2");
}

void three(void)
{ digitalWrite(27, LOW);
  digitalWrite(5, LOW);
  digitalWrite(18, LOW);
  digitalWrite(19, LOW);
  digitalWrite(22, LOW);
  Serial.println("3");
}

void state(int num)
{ if(num==0){zero();}
  else if(num==1){one();}
  else if(num==2){two();}
  else if(num==3){three();}
}

void loop1(int num)
{ int r=3;
  while(r--)
  { state(num);
    delay(1000);
    off();
    delay(1000);
  }
  loo=!loo;
}

void loop2(int num)
{ int r=6;
  while(r--)
  { state(num);
    delay(500);
    off();
    delay(500);
  }
  loo=!loo;
}

void setup() {
  Serial.begin(115200);
  Serial.println("Start Firmware");
  // SW GPIO16
  pinMode(16, INPUT_PULLUP);
  pinMode(4, INPUT_PULLUP);
  // Seven Segment
  pinMode(27, OUTPUT); // SegA
  pinMode(5, OUTPUT); // SegB
  pinMode(18, OUTPUT); // SegC
  pinMode(19, OUTPUT); // SegD
  pinMode(21, OUTPUT); // SegE
  pinMode(23, OUTPUT); // SegF
  pinMode(22, OUTPUT); // SegG
  off(); // OFF LED ALL Segment
}

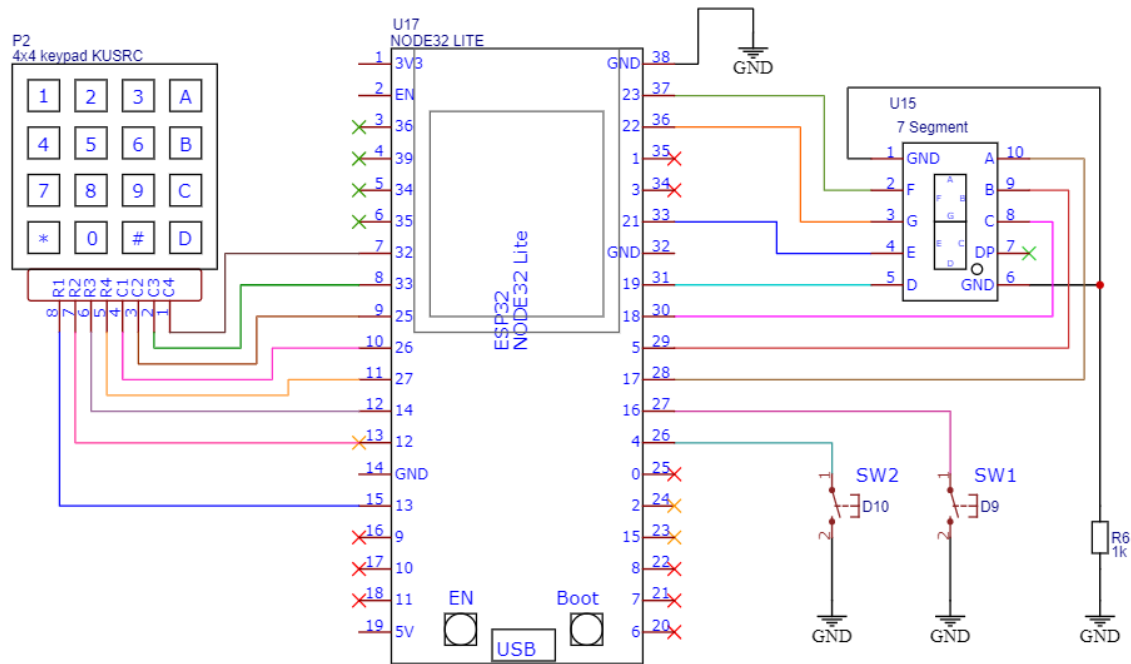
void loop()
{ if(digitalRead(16)==0 && digitalRead(4)==0)
  { if(loo){loop1(0);}
    else{loop2(0);}
  }else if(digitalRead(16)==0 && digitalRead(4)==1)
  { if(loo){loop1(1);}
    else{loop2(1);}
  }else if(digitalRead(16)==1 && digitalRead(4)==0)
  { if(loo){loop1(2);}
    else{loop2(2);}
  }else
  { if(loo){loop1(3);}
    else{loop2(3);}
  }
  delay(2000);
}

```

*<https://wokwi.com/projects/337596448943112787>

2.4 ทดลองเขียนควบคุม 7 Segment 2Bit input ด้วย Key-Pad

ปรับปรุงจาก Lab 2.3 โดยที่ Input รับมาจาก Numpad บนบอร์ด แล้วนำค่ามาแสดงที่ 7Segment ได้ตามที่กดจาก Numpad



ตัวเลขและตัวอักษรที่แสดงที่ 7Segment

0	1	2	3	4	5	6
7	8	9	A	B	C	D

โดยกดแล้วให้แสดงตามที่กดปุ่ม

Hint: <https://diyi0t.com/keypad-arduino-esp8266-esp32/>

Keypad by Mark Stanley, Alexander Brevig Version 3.1.1 **INSTALLED**

Keypad is a library for using matrix style keypads with the Arduino. As of version 3.0 it now supports multiple keypresses. This library is based upon the Keypad Tutorial. It was created to promote Hardware Abstraction. It improves readability of the code by hiding the pinMode and digitalWrite calls for the user.

[More info](#)

Select version Install

เขียนโปรแกรมลงในกล่องคำตอบด้านล่าง และถ่ายวิดีโอผลลัพธ์ของโจทย์นี้ Upload ไฟล์ตามหมู่เรียน

```

int stateNum = 0;

#define A 16 // free memory
#define B 5
#define C 18
#define D 19
#define E 21
#define F 23
#define G 22

#define R1 13 // output
#define R2 12 // output
#define R3 14 // output
#define R4 27 // output
#define C1 26 //input
#define C2 25 //input
#define C3 33 //input
#define C4 32 //input

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

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

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

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

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

void four(void){
  // LED 4
  digitalWrite(A, HIGH);
  digitalWrite(B, LOW);
  digitalWrite(C, LOW);
  digitalWrite(D, HIGH);
  digitalWrite(E, HIGH);
  digitalWrite(F, LOW);
  digitalWrite(G, LOW);
  //Serial.println("4");
}

void five(void){
  digitalWrite(A, LOW);
  digitalWrite(B, HIGH);
  digitalWrite(C, LOW);
  digitalWrite(D, LOW);
  digitalWrite(E, HIGH);
  digitalWrite(F, LOW);
  digitalWrite(G, LOW);
  // Serial.println("5");
}

void six(void){
  digitalWrite(A, LOW);
  digitalWrite(B, HIGH);
  digitalWrite(C, LOW);
  digitalWrite(D, LOW);
  digitalWrite(E, LOW);
  digitalWrite(F, LOW);
  digitalWrite(G, LOW);
  //Serial.println("6");
}

void seven(void){
  digitalWrite(A, LOW);
  digitalWrite(B, LOW);
  digitalWrite(C, LOW);
  digitalWrite(D, HIGH);
  digitalWrite(E, HIGH);
  digitalWrite(F, HIGH);
  digitalWrite(G, HIGH);
  //Serial.println("7");
}

void eight(void){
  digitalWrite(A, LOW);
  digitalWrite(B, LOW);
  digitalWrite(C, LOW);
  digitalWrite(D, LOW);
  digitalWrite(E, LOW);
  digitalWrite(F, LOW);
  digitalWrite(G, LOW);
  //Serial.println("8");
}

void nine(void){
  digitalWrite(A, LOW);
  digitalWrite(B, LOW);
  digitalWrite(C, LOW);
  digitalWrite(D, LOW);
  digitalWrite(E, HIGH);
  digitalWrite(F, LOW);
  digitalWrite(G, LOW);
  //Serial.println("9");
}

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

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

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

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

void setup() {
  Serial.begin(115200);
  Serial.println("Start Firmware");
  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
  off(); // OFF LED ALL Segment
  pinMode(R1, OUTPUT);
  pinMode(R2, OUTPUT);
  pinMode(R3, OUTPUT);
  pinMode(R4, OUTPUT);
  pinMode(C1, INPUT_PULLUP);
  pinMode(C2, INPUT_PULLUP);
  pinMode(C3, INPUT_PULLUP);
  pinMode(C4, INPUT_PULLUP);
}

void loop() {
  digitalWrite(R1, LOW);
  digitalWrite(R2, HIGH);
  digitalWrite(R3, HIGH);
  digitalWrite(R4, HIGH);
  if(digitalRead(C1) == LOW){one();}
  else if (digitalRead(C2) == LOW){two();}
  else if (digitalRead(C3) == LOW){three();}
  else if (digitalRead(C4) == LOW){a();}
  else{
    digitalWrite(R1, HIGH);
    digitalWrite(R2, LOW);
    digitalWrite(R3, HIGH);
    digitalWrite(R4, HIGH);
    if(digitalRead(C1) == LOW){four();}
    else if (digitalRead(C2) == LOW){five();}
    else if (digitalRead(C3) == LOW){six();}
    else if (digitalRead(C4) == LOW){b();}
    else{
      digitalWrite(R1, HIGH);
      digitalWrite(R2, HIGH);
      digitalWrite(R3, LOW);
      digitalWrite(R4, HIGH);
      if(digitalRead(C1) == LOW){seven();}
      else if (digitalRead(C2) == LOW){eight();}
      else if (digitalRead(C3) == LOW){nine();}
      else if (digitalRead(C4) == LOW){c();}
      else{
        digitalWrite(R1, HIGH);
        digitalWrite(R2, HIGH);
        digitalWrite(R3, HIGH);
        digitalWrite(R4, LOW);
        if(digitalRead(C1) == LOW){off();}
        else if (digitalRead(C2) == LOW){zero();}
        else if (digitalRead(C3) == LOW){off();}
        else if (digitalRead(C4) == LOW){d();}
        else{off();}
      }
    }
  }
  delay(500);
}

```

*<https://wokwi.com/projects/338265128157839956>