



เรื่อง ลานจอดรถ

จัดทำโดย

ธนภัทร ชนะเวศน์พงศ์ 6110506435

ภูมิภัทร เฝ้าโยธิน 6110503436

เสนอ

อาจารย์ ชัยพร ใจแก้ว

ภาควิชา

วิศวกรรมคอมพิวเตอร์

รายงานเล่มนี้เป็นส่วนหนึ่งของวิชา การฝึกปฏิบัติทางวิศวกรรมคอมพิวเตอร์

ภาคเรียนที่ 2 ปีการศึกษา 2562

มหาวิทยาลัยเกษตรศาสตร์

ไฟล์ที่เกี่ยวข้องในการรันโปรแกรม คือ **PracticumProjectCode**
โดยคำอธิบายจะcommentไว้ตามแต่ละบรรทัด

อุปกรณ์ที่ใช้

1.LED เขียว-แดง

2.ultrasonic

```
//ตั้งค่าPIN แต่ละตัวมีpin trigh,echoของ ultrasonicและ pin ledสีเขียว-แดง
const int TRIG_PIN[]={PIN_PC0,PIN_PC1,PIN_PB1,PIN_PB0};
const int ECHO_PIN[]={PIN_PC2,PIN_PC3,PIN_PB3,PIN_PB2};
const int RED_PIN[]={PIN_PD1,PIN_PD0,PIN_PD5};
const int GREEN_PIN[]={PIN_PD6};
```

```
long duration;//เป็นเวลาดั้งแต่ การส่งคลื่นเสียงออกจาก trig เข้า echo
int distance;//ระยะทางที่วัดได้จาก ultrasonic
int count = 2; ให้ที่จอดรถ วาง2คัน
```

```
//////// กำหนดแต่ละตัวว่าตัวไหนเป็น input หรือ output //////////
void setup() {
  for(int i=0;i<4;i++)
  {
    pinMode(TRIG_PIN[i], OUTPUT);
    pinMode(ECHO_PIN[i], INPUT);
  }
  // Serial.begin(9600);
}
for(int i=0;i<2;i++)
{
  pinMode(RED_PIN[i], OUTPUT);
  pinMode(GREEN_PIN[i], OUTPUT);
}
}
```

```

void loop() {
////////////////////////////////////////////////////
//ultrasonic สำหรับที่จอดรถ อันที่1

digitalWrite(TRIG_PIN[2], LOW); // ปิดการส่งคลื่นเสียง
delayMicroseconds(2);
digitalWrite(TRIG_PIN[2], HIGH); //เปิดการส่งคลื่นเสียง
delayMicroseconds(10);
digitalWrite(TRIG_PIN[2], LOW); // ปิดการส่งคลื่นเสียง

// duration = ระยะเวลาของเสียง จากultrasonic ถึงพื้น และกลับไปยัง
//          ตัวultrasonic อีกครั้ง
duration = pulseIn(ECHO_PIN[2], HIGH);

//ระยะทาง(distance) จาก ultrasonic ถึงพื้น = เวลา*ความเร็ว/2 (หาร2
//          เพราะduration มาจากultrasonic ถึงพื้น และกลับ ultrasonic)
distance = duration * 0.034 / 2;

if(distance < 8) // ถ้าระยะห่างน้อยกว่า8 ให้ไฟแดงติด(มีรถจอด)
{
digitalWrite(RED_PIN[1], HIGH);
}
else
{
digitalWrite(RED_PIN[1], LOW); // ถ้าระยะห่างมากกว่า8 ให้ไฟแดงดับ(ไม่มี มีรถ
จอด)
}
}

```

```
////////////////////////////////////
```

///ultrasonic สำหรับที่จอดรถ อันที่2 code เหมือนที่จอดแรกแต่เปลี่ยนpin เป็นของที่จอด
อันที่2

```
digitalWrite(TRIG_PIN[1], LOW);  
delayMicroseconds(2);  
digitalWrite(TRIG_PIN[1], HIGH);  
delayMicroseconds(10);  
digitalWrite(TRIG_PIN[1], LOW);
```

```
duration = pulseIn(ECHO_PIN[1], HIGH);  
distance = duration * 0.034 / 2;  
if(distance < 8)  
{  
    digitalWrite(RED_PIN[0], HIGH);  
}  
else  
{  
    digitalWrite(RED_PIN[0], LOW);  
}
```

```

////////////////////////////////////
//ultrasonicทางเข้าเพื่อเช็คว่ามีรถเข้าลานจอดกี่คัน

digitalWrite(TRIG_PIN[0], LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN[0], HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN[0], LOW);

duration = pulseIn(ECHO_PIN[0], HIGH);
distance = duration * 0.034 / 2;
if(distance < 8)//ถ้าระยะทางน้อยกว่า 8 แสดงว่ามีรถเข้า ให้count(จำนวนที่ว่างที่จอดรถ
) ลดลงหนึ่ง
{
    count--; //จำนวนที่ว่างลดลงหนึ่ง
    delay(500);
}
////////////////////////////////////
//ultrasonicทางออกเพื่อเช็คว่ามีรถออกจากลานจอดกี่คัน

digitalWrite(TRIG_PIN[3], LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN[3], HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN[3], LOW);

duration = pulseIn(ECHO_PIN[3], HIGH);
distance = duration * 0.034 / 2;

```

if(distance <8)//ถ้าระยะทางน้อยกว่า 8 แสดงว่ามีรถออก ให้count(จำนวนที่ว่างที่จอดรถ) เพิ่มขึ้นหนึ่ง

```
{  
    count++; //จำนวนที่จอดรถเพิ่มขึ้นหนึ่ง  
    delay(500);}
```

////////////////////////////////////

if(count<0)//ถ้าจำนวนที่ว่างไม่เหลือแล้วแต่ยังมีรถเข้าไป count จะน้อยกว่า0 เราก็จะให้count = 0เหมือนเดิม

```
{  
    count = 0;
```

```
}
```

if(count==0) //ถ้าcount = 0 แปลว่าที่จอดรถเต็ม ให้ ไฟสีแดงติด เชี่ยวกับ

```
{  
    digitalWrite(RED_PIN[2], HIGH); //ไฟแดงติด  
    digitalWrite(GREEN_PIN[0], LOW); //ไฟเขียวดับ  
}
```

if(count > 0)//ถ้ามีที่จอดรถให้ไฟเขียวติดไฟแดงดับ

```
{  
    digitalWrite(GREEN_PIN[0], HIGH); //ไฟเขียวติด  
    digitalWrite(RED_PIN[2], LOW); //ไฟแดงดับ  
}
```

```
delay(250);
```

```
}
```