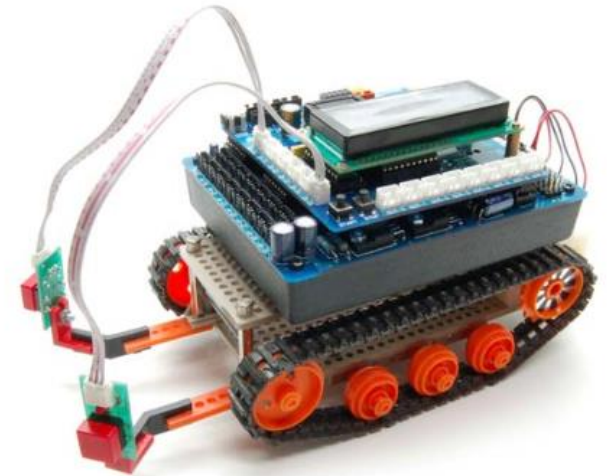
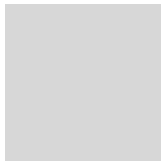
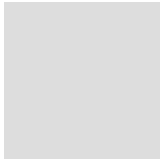




# Robo-11 Attacker



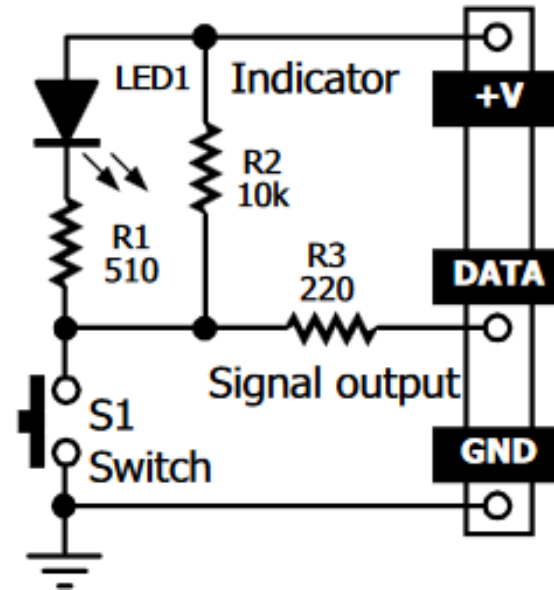
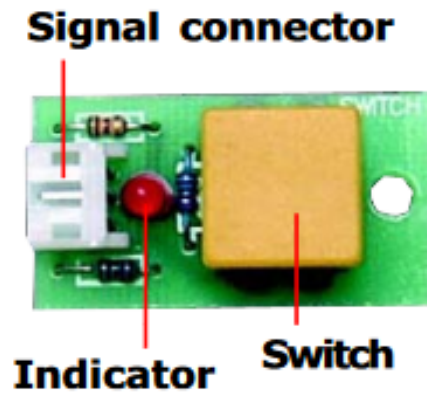
# วัตถุประสงค์



- เข้าใจการทำงานของเซนเซอร์แบบปุ่มกด
- สามารถสร้างโครงสร้างหุ่นยนต์แบบโจมตี
- สามารถเขียนโปรแกรมให้กับหุ่นยนต์แบบโจมตีได้



# ความรู้เกี่ยวกับแผงวงจรสวิตช์



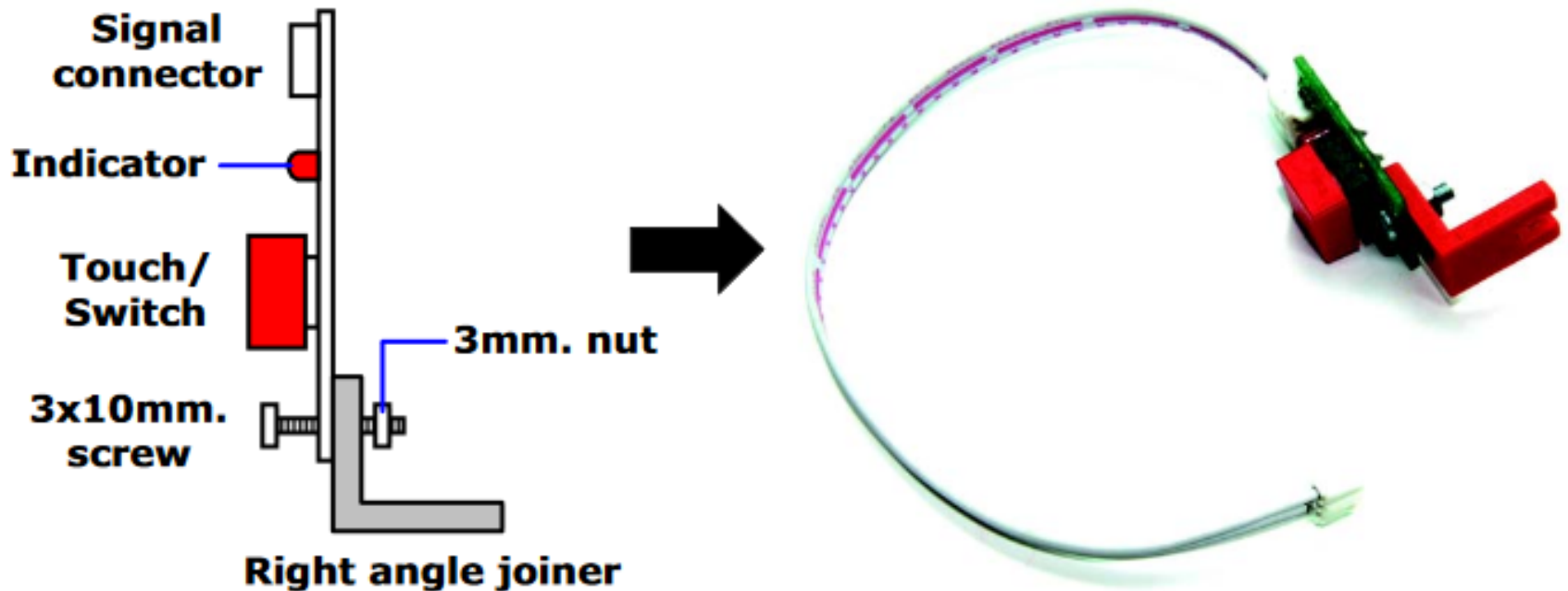
- ขณะไม่กดสวิตช์ ให้ผลลอจิกเป็น “1”
- ขณะกดสวิตช์ ให้ผลลอจิกเป็น “0” และ ไฟ LED ติดสว่าง
- เป็นดิจิทัลคอล ต่อช่อง DI-0 ถึง DI-8



# การสร้าง



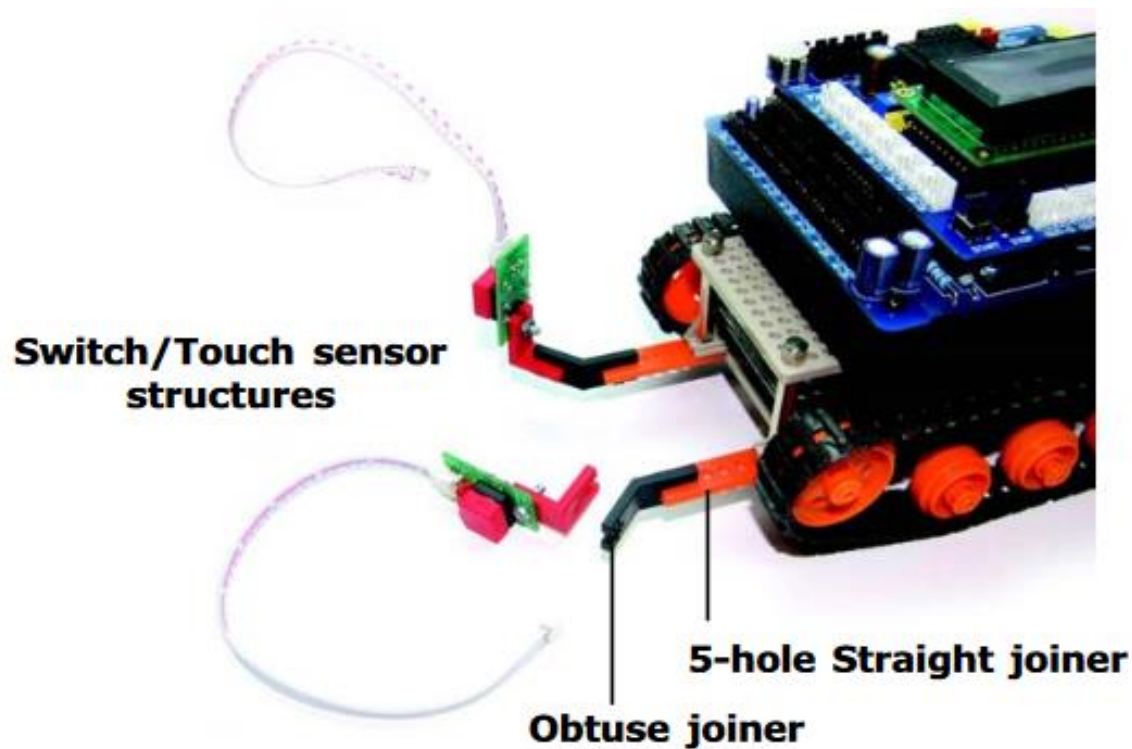
- นำชิ้นต่อมุมฉากมายึดเข้ากับแผงวงจรสวิตช์ โดยใช้สกรู ทำขึ้นมา 2 ชุด



# การสร้าง



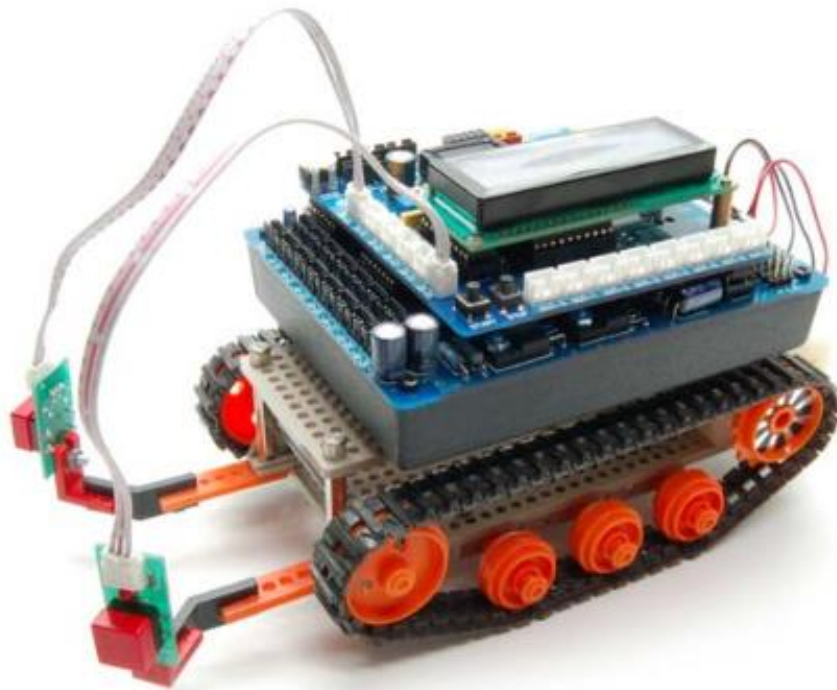
- ต่อชิ้นต่อมุ่มบ้านเข้ากับปลายของแท่งต่อ 5 รู ทางด้านหน้าของหุ่นยนต์ จากนั้นนำแผงวงจรสวิทช์ที่ติดตั้งชิ้นต่อมุ่มจาก มาต่อเข้าที่ปลายของชิ้นต่อมุ่มบ้าน ทำทั้ง 2 ข้าง



# การสร้าง



- ต่อสายสัญญาณจากแผงวงจรสวิตซ์ทางซ้ายเข้าที่จุดต่อ DI-15
- และต่อสายสัญญาณจากแผงวงจรสวิตซ์ทางขวาเข้าที่จุดต่อ DI-10



# การเขียนโปรแกรม



```
#define pow 70
void main()
{
    int obj_left,obj_right;
    printf("Press Start!\n");
    start_press();
    printf("Robot Move...\n");
    while(!stop_button())
    {
        obj_left = digital(15);
        obj_right = digital(10);
```

← กำหนดพลังงานเท่ากับ **70**

← เริ่มต้นฟังก์ชันหลัก

← เปิดปีกกาเริ่ม

← ประกาศตัวแปรเซ็นเซอร์ซ้ายและขวา

← แสดงข้อความให้กดปุ่ม **Start**

← ฟังก์ชันปุ่มเริ่มต้น

← แสดงข้อความหุ่นยนต์เคลื่อนที่

← ให้หุ่นทำงานตลอดจนกว่าจะกดหยุด

← เปิดคำสั่ง **While**

← กำหนดให้เซ็นเซอร์เท่ากับ **DI-15**

← กำหนดให้เซ็นเซอร์เท่ากับ **DI-10**



# การเขียนโปรแกรม



```
if(obj_left==0 && obj_right==0) ← ถ้าไม่มีการกดสวิตช์ทั้งสองข้าง
{
    run_fd(0.1); ← ให้หุ่นยนต์เดินหน้าไป 0.1 วินาที
}
else if(obj_left==0 && obj_right==1) ← ถ้าสวิตช์ด้านขวาถูกกด
{
    run_bk(0.5); ← ให้หุ่นถอยหลัง 0.5 วินาที
    turn_left(0.5); ← ตามด้วยเลี้ยวซ้าย 0.5 วินาที
}
```





# การเขียนโปรแกรม



```
else if(obj_left==1 && obj_right==0) ← ถ้าสวิตช์ด้านซ้ายถูกกด
{
    run_bk(0.5); ← ให้หุ่นถอยหลังไป 0.5 วินาที
    turn_right(0.5); ← ตามด้วยเลี้ยวขวา 0.5 วินาที
}
else if(obj_left==1 && obj_right==1) ← ถ้าสวิตช์ด้านขวาและซ้ายถูกกด
{
    run_bk(0.5); ← ให้หุ่นถอยหลัง 0.5 วินาที
    turn_right(1.0); ← ตามด้วยเลี้ยวขวา 0.5 วินาที
}
```



# การเขียนโปรแกรม



```
}  
    ao(); ← สั่งให้หุ่นหยุดการทำงานทุกอย่าง  
    beep(); ← ให้มีเสียงบีบหนึ่งครั้ง  
    printf("Stop...\n"); ← แสดงข้อความว่า หยุด  
}  
void turn_left(float spin_time) ← เริ่มฟังก์ชันเลี้ยวซ้าย  
{  
    motor(0,-pow); ← ให้มอเตอร์ซ้าย ถอยหลัง  
    motor(1,pow); ← ให้มอเตอร์ขวา เดินหน้า  
}
```



# การเขียนโปรแกรม



```
void turn_right(float spin_time) ← เริ่มฟังก์ชันเลี้ยวขวา
{
  motor(0,pow); ← ให้มอเตอร์ซ้ายเดินหน้า
  motor(1,-pow); ← ให้มอเตอร์ขวาถอยหลัง
}

void run_fd(float delay) ← เริ่มฟังก์ชันเดินหน้า
{
  motor(0,pow); ← ให้มอเตอร์ซ้ายเดินหน้า
  motor(1,pow); ← ให้มอเตอร์ขวาเดินหน้า
}
```



# การเขียนโปรแกรม



```
void run_bk(float delay) ← เริ่มฟังก์ชันถอยหลัง
{
  motor(0,-pow); ← ให้มอเตอร์ซ้ายถอยหลัง
  motor(1,-pow); ← ให้มอเตอร์ขวาถอยหลัง
}
```



# ขั้นตอนการทำงาน



- เขียนโปรแกรมที่ A2-1 แล้วดาวน์โหลดไปยังหุ่นยนต์
- นำหุ่นยนต์วางบนพื้นที่ทดสอบ จากนั้นเป็นสวิทช์ที่จอแสดงข้อความ Press Start
- กดสวิทช์ Start บนบอร์ด





**Thank You!**

