

Giao Tiếp Với Cảm Biến

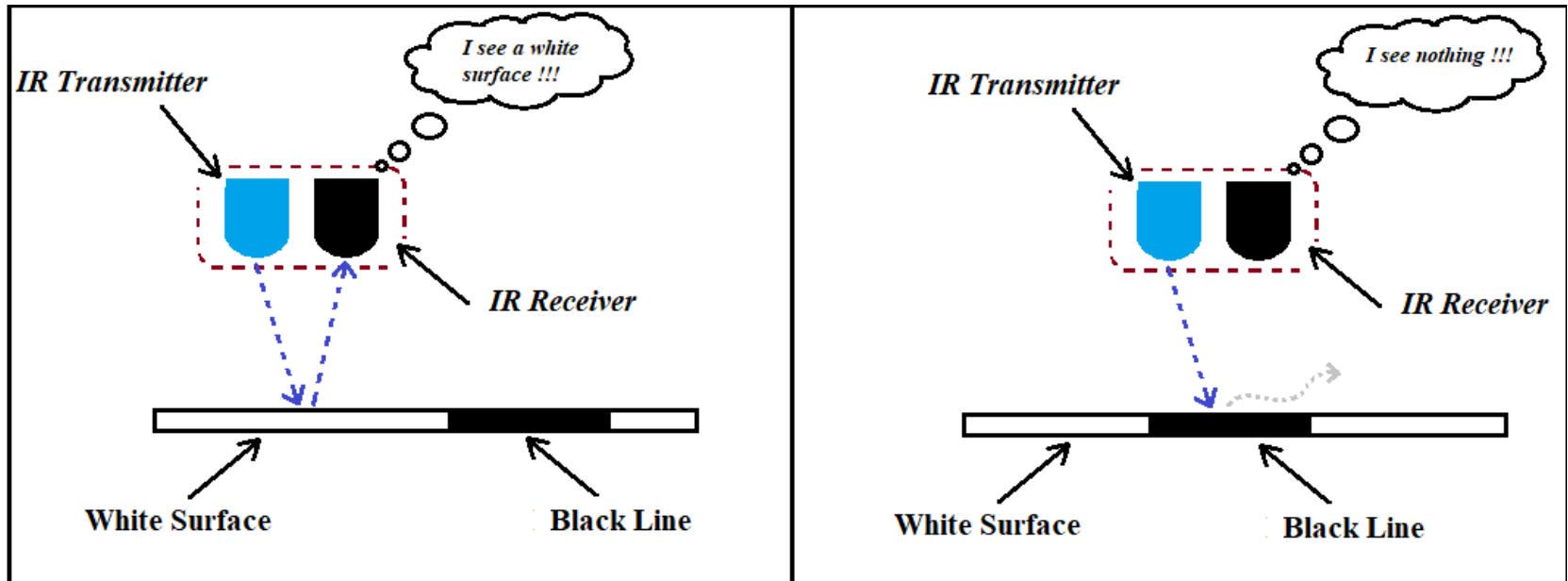
SArduino Training 2019 cho THCS
Saigon Institute of Technology



Nội dung

- I. Cảm biến dò đường thanh 5 led (Line Sensor)
- II. Cảm biến khoảng cách (UltraSonic Sensor)
- III. Servo
- IV. Bài tập

I. Cảm biến dò đường thanh 5 led (Line Sensor)



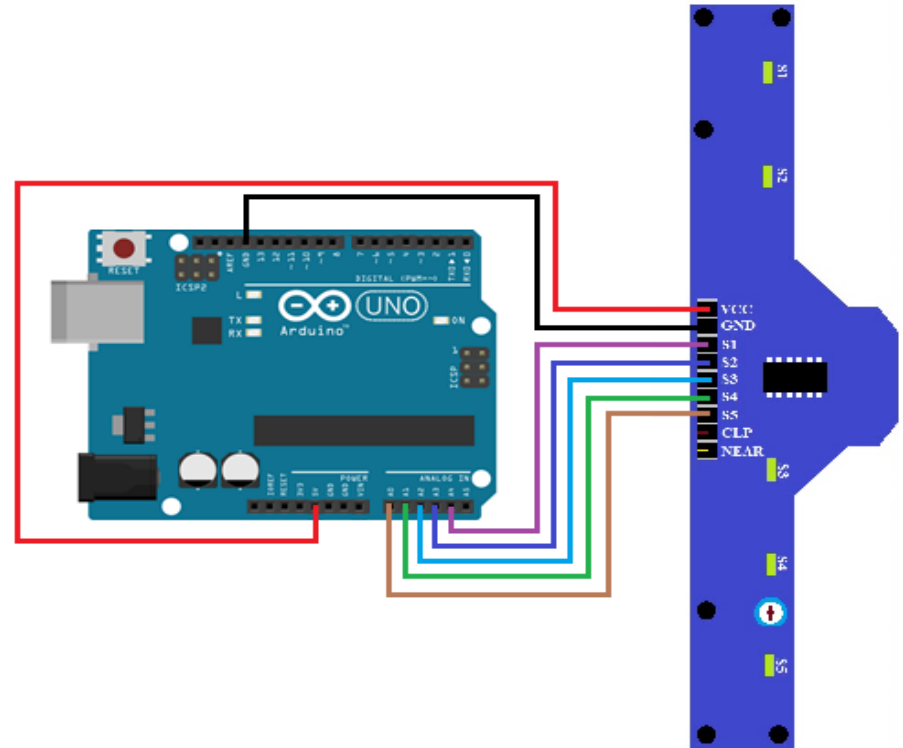
Do tính chất hấp thụ tia hồng ngoại của bề mặt màu đen so với các bề mặt khác nên tia phản xạ cũng khác biệt vì vậy mà phân biệt được đâu là bề mặt đen, đâu không phải là bề mặt đen.

Thực hành

- Yêu cầu:
Viết chương trình đọc giá trị của cảm biến dò đường tại cảm biến số 3 và ghi giá trị của cảm biến này ra cổng Serial trên mạch Arduino
- Ghi chú:
 - Đèn led của cảm biến gặp vùng địa hình màu tối: giá trị cảm biến đọc được $value = 0$
 - Đèn led của cảm biến gặp vùng địa hình màu sáng: giá trị cảm biến đọc được $value = 1$

Sơ đồ đi dây

Cảm biến dò line	Mạch arduino
SS1	A4
SS2	A3
SS3	A2
SS4	A1
SS5	A0
VCC5	5V
GND	GND



Chương trình mẫu - PASCAL

```
uses STArduino;
```

```
const ls = A2;
```

```
procedure setup;
```

```
begin
```

```
    serialBegin(9600);
```

```
    pinMode(ls, INPUT);
```

```
end;
```

```
procedure loop;
```

```
var value: longint;
```

```
begin
```

```
    value := digitalRead(ls);
```

```
    serialPrintString('Value: ');
```

```
    serialPrintlnLong(value);
```

```
    delay(1000);
```

```
end;
```

```
begin
```

```
    setup;
```

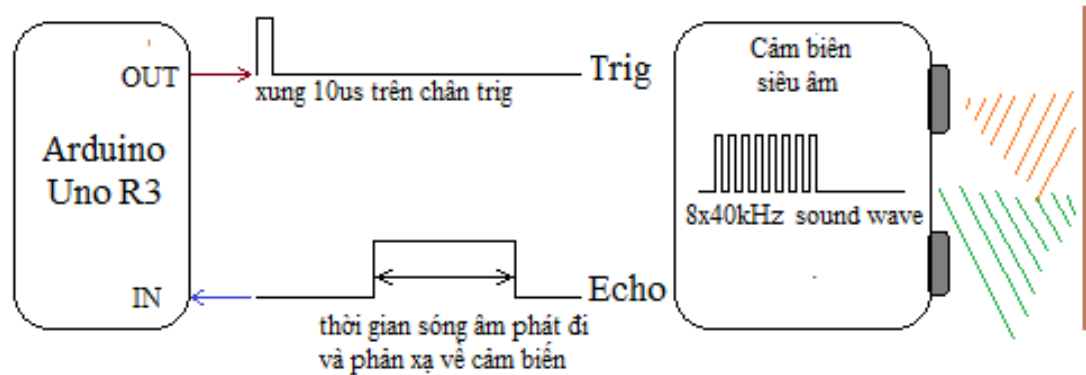
```
    loop;
```

```
end.
```

Chương trình mẫu – C++

```
#define ls A2
void setup() {
    Serial.begin(9600);
    pinMode(ls, INPUT);
}
void loop() {
    int value = digitalRead(ls);
    Serial.print("Value: ");
    Serial.println(value);
    delay(2000);
}
```

II. Cảm biến khoảng cách (UltraSonic Sensor)



- Khi bắt đầu Arduino sẽ phát 1 xung vào chân Trig có độ rộng là 10us.
- Sau khi có xung kích vào chân Trig thì cực phát của cảm biến sẽ phát ra 8 xung của sóng âm thanh.
- Ngay sau đó chân Echo sẽ kéo lên cao và đợi sóng âm thanh phản xạ lại. khi gặp sóng phản xạ thì pin Echo sẽ được kéo xuống thấp.
- Thời gian chân Echo ở mức cao sẽ chính là thời gian sóng âm thanh đi và về trên quãng đường từ cảm biến tới vật cản

Thực hành

- **Yêu cầu:**

Viết chương trình đọc giá trị của cảm biến khoảng cách và ghi giá trị của cảm biến ra cổng Serial trên mạch Arduino

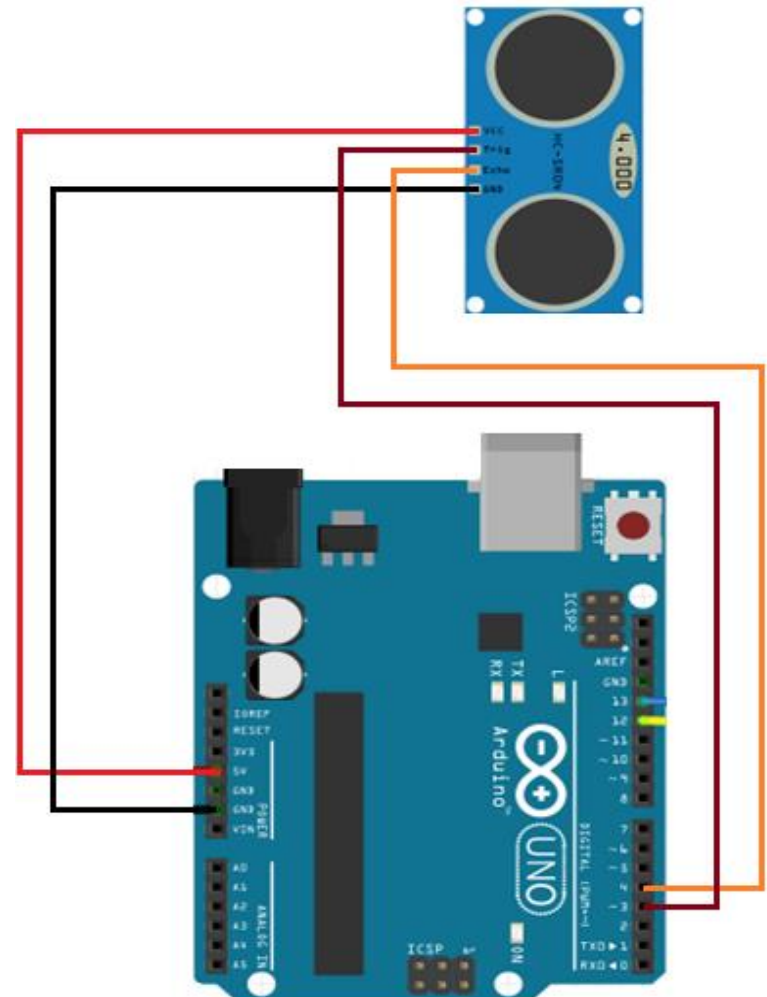
- **Ghi chú:**

- Giá trị trả về của cảm biến khoảng cách là khoảng thời gian (microseconds) tính từ lúc cảm biến phát ra sóng siêu âm đến lúc nhận lại sóng phản xạ.
- Tốc độ của âm thanh trong không khí là 340 m/s (hằng số vật lý), tương đương với 29,412 ms/cm ($10^6 / (340 \cdot 100)$). Khi đã tính được thời gian, ta sẽ chia cho 29,412 để được quãng đường, sau đó chia tiếp cho 2 để được khoảng cách từ cảm biến đến vật cản:

$$\text{Khoảng cách} = \text{microseconds} / 29 / 2$$

Sơ đồ đi dây

Cảm biến khoảng cách	Mạch arduino
VCC	5V
Trig	3
Echo	4
GND	GND



Chương trình mẫu - PASCAL

```
uses STArduino;
const
    trigPin = 3;
    echoPin = 4;

function microsecondsToCentimeters (microseconds: longint): real;
begin
    microsecondsToCentimeters := microseconds/29/2;
end;

procedure setup;
begin
    serialBegin(9600);
    pinMode(trigPin, OUTPUT); // chân trig sẽ phát tín hiệu
    pinMode(echoPin, INPUT); // chân echo sẽ nhận tín hiệu
end;
```

Chương trình mẫu - PASCAL (tt)

```
procedure loop;  
var duration: longint;  
    cm: real;  
begin  
    digitalWrite(trigPin, LOW); // tắt chân trig  
    delayMicroseconds(2);  
    digitalWrite(trigPin, HIGH); // phát xung từ chân trig  
    delayMicroseconds(10); // xung có độ dài 10 micro giây  
    digitalWrite(trigPin, LOW); // tắt chân trig  
    duration := pulseIn(echoPin, HIGH);  
    // Tính khoảng cách đến vật  
    cm := microsecondsToCentimeters(duration);  
    serialPrintString('Khoang cach (cm): ');  
    serialPrintlnReal(cm);  
    delay(500);  
end;  
  
begin  
    setup;  
    loop;  
end.
```

Chương trình mẫu – C++

```
#define trigPin 3
#define echoPin 4

double microsecondsToCentimeters(long microseconds) {
    return microseconds/29/2;
}

void setup() {
    Serial.begin(9600);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
}
```

Chương trình mẫu – C++ (tt)

```
void loop() {
    digitalWrite(trigPin, LOW); // tắt chân trig
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH); // phát xung từ chân trig
    delayMicroseconds(10); // xung có độ dài 10 micro giây
    digitalWrite(trigPin, LOW); // tắt chân trig
    long duration = pulseIn(echoPin, HIGH);

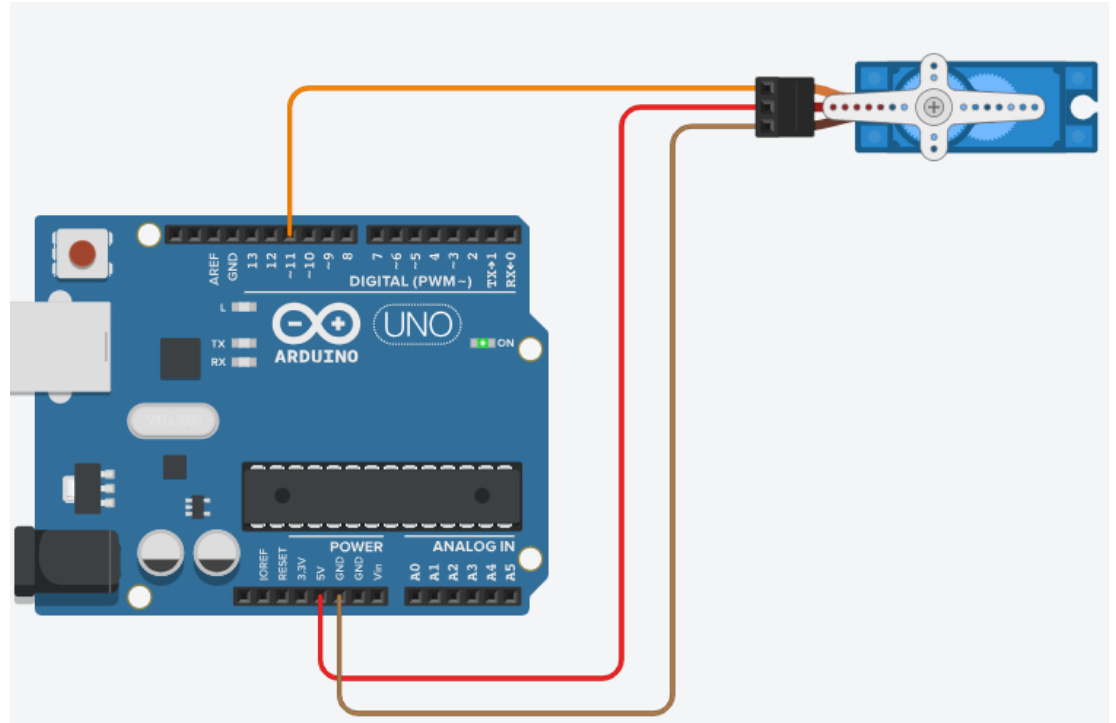
    // Tính khoảng cách đến vật
    double cm = microsecondsToCentimeters(duration);
    Serial.print("Khoang cach (cm): ");
    Serial.println(cm);
    delay(500);
}
```

III. Servo

- Yêu cầu:
 - Viết chương trình điều khiển servo, sao cho servo quay từ góc 0° đến 180° , dừng lại 2 giây và quay ngược trở về góc 0° .

Sơ đồ đi dây: Servo

Servo	Arduino
Gnd	GND
Vcc	5V
Signal	11



Chương trình mẫu – PASCAL

```
uses STArduino, STServo;
```

```
const signalPIN = 11;
```

```
var sv: SServo;
```

```
procedure setup;
```

```
begin
```

```
    pinMode(signalPIN, OUTPUT);
```

```
    servoAttach(sv, signalPIN);
```

```
    servoWrite(sv, 0);
```

```
end;
```

```
procedure loop;
```

```
begin
```

```
    servoWrite(sv, 180);
```

```
    delay(2000);
```

```
    servoWrite(sv, 0);
```

```
    delay(2000);
```

```
end;
```

```
begin
```

```
    setup;
```

```
    loop;
```

```
end.
```

Chương trình mẫu – C++

```
#include <Servo.h>           // Thư viện điều khiển servo
#define signalPIN 11
Servo myServo;             // Khai báo đối tượng servo

void setup() {
    // Cài đặt servo cho chân signalPIN
    myServo.attach(signalPIN);
    myServo.write(0); // Quay servo về góc 0 độ
}

void loop() {
    myServo.write(180); // Quay servo về góc 180 độ
    delay(2000);
    myServo.write(0); // Quay servo về góc 0 độ
    delay(2000);
}
```

IV. Bài tập

- **Cảm biến dò đường**

- Viết chương trình điều khiển đọc tất cả giá trị của cảm biến dò đường và ghi các giá trị của cảm biến ra cổng Serial trên mạch Arduino.

- **Cảm biến khoảng cách**

- Viết chương trình điều khiển đọc giá trị của cảm biến khoảng cách và ghi giá trị của cảm biến ra cổng Serial trên mạch Arduino nếu giá trị khoảng cách nằm trong khoảng 5 cm đến 1 mét.

IV. Bài tập (tt)

• Tổng hợp

- Viết chương trình điều khiển xe chạy tới trước theo đường màu đen với tốc độ tối đa cho tới khi cảm biến số 3 không nhận được màu đen thì dừng lại.
- Viết chương trình điều khiển xe chạy tới trước với tốc độ tối đa cho tới khi cách vật cản 5 cm thì dừng lại.
- Viết chương trình điều khiển xe đi theo đường màu đen trên địa hình cho trước, xe dừng lại nếu gặp vật cản cách 20 cm, nếu không còn vật cản cách 20 cm thì xe tiếp tục di chuyển theo đường màu đen.