

リモコンでサーボモーターを動かそう!

リモコンを使ってサーボモーターを制御する

このレッスンで身につける力

- ブレッドボードにサーボモーターと赤外線受信モジュールを使った回路を作ることが出来る
 - サンプルコードを実行できる
 - サンプルコードを改造して他のリモコンでもモーターを動かすことができる
-

ミッションの準備

- Osoyoo UNO Board x1
- 赤外線受信機 x1
- リモートコントローラー x1
- SG90サーボモーター x1
- ブレッドボード x1
- ジャンパー線
- USBケーブル x1
- パソコン x1

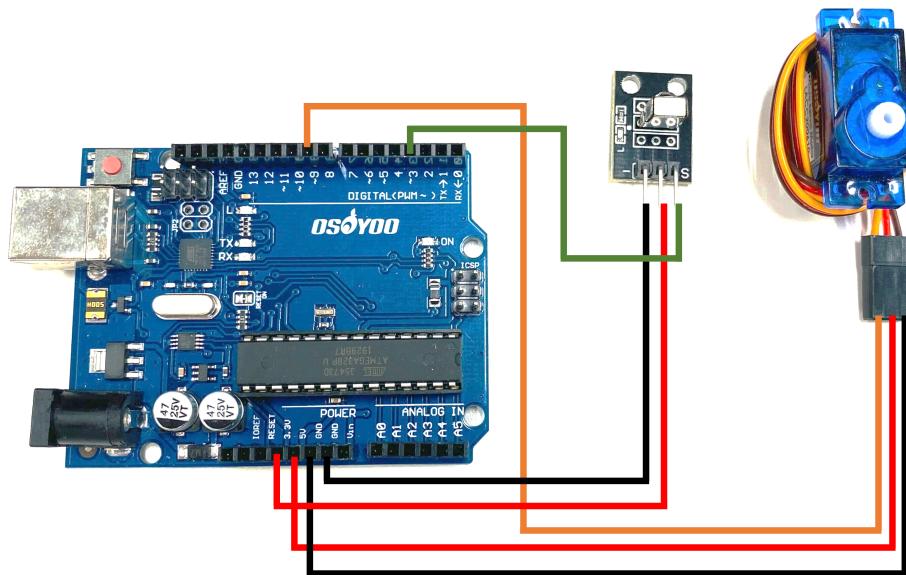
0.ハードウェアを用意しよう

- Osoyoo UNO Board(Arduino UNO rev.3と完全互換) x 1
 - USBケーブル x 1
 - パソコン x 1
-

ミッションチャレンジ

赤外線受信機とサーボモーターをArduinoにつなごう!

配線図:



- 回路が作れたらチェック!

リモコンからの信号をシリアルモニタに表示しよう!

スケッチに以下のコードをコピー&ペーストして、スケッチを実行してみよう。

```
#include <IRremote.h> // IRRemote.hの関数を使用する
const int irReceiverPin = 3; //受信モジュールのSIGはpin2
IRrecv irrecv(irReceiverPin); //IRrecvタイプの変数を作成します
decode_results results; // 結果

void setup()
{
    Serial.begin(9600); //シリアルを初期化し、ボーレートは9600に設定する
    irrecv.enableIRIn(); // 赤外線受信機モジュールを有効にする
}

void loop()
{
    if (irrecv.decode(&results)) //赤外線受信機モジュールの受信データ
    {
        Serial.print("irCode: "); //irCode : "の値を送信する
        Serial.print(results.value, HEX); //シリアルに値を出力する
        Serial.print(", bits: "); //bitsを送信する
        Serial.println(results.bits); //bitsを結果に出力する
        irrecv.resume(); // 次の値を受取る
    }
    delay(600); //600ミリ秒待機
}
```

上のプログラムをコピーして実行し、シリアルモニタを開こう！リモコンを押すと何が起こるかな？

```
16:52:13.354 -> irCode: FF38C7, bits: 32
16:52:13.972 -> irCode: FF38C7, bits: 32
16:52:14.573 -> irCode: FF38C7, bits: 32
16:52:16.375 -> irCode: FF38C7, bits: 32
16:52:17.577 -> irCode: FF18E7, bits: 32
16:52:18.179 -> irCode: FF18E7, bits: 32
16:52:44.627 -> irCode: FF18E7, bits: 32
16:52:45.200 -> irCode: FF18E7, bits: 32
16:52:46.404 -> irCode: FF4AB5, bits: 32
16:52:47.005 -> irCode: FF4AB5, bits: 32
16:52:48.207 -> irCode: FF18E7, bits: 32
16:52:48.809 -> irCode: FF18E7, bits: 32
16:52:57.844 -> irCode: FF18E7, bits: 32
16:52:58.424 -> irCode: FF18E7, bits: 32
16:52:59.059 -> irCode: FF18E7, bits: 32
```

- シリアルモニタに上の画像のような表示が出たらチェック！

リモコンでサーボモーターを操作しよう！

ロボットのタイヤを動かしているモーターは何回転でもできるかわりに、決まった角度に動かすことはできないよね。

今回動かす「サーボモーター」は、0度から180度までしか動かないかわりに、決まった角度に動かすことができるんだ！

ArduinoIDEを開き、ファイル→名前を付けて保存をクリックして、「name_lesson_16_2」という名前で保存しましょう。

スケッチに以下のコードをコピー&ペーストして、スケッチを実行してみよう。

```
#include <IRremote.h>           // IRremoteライブラリをアルドウイーノライブラリにコピー
                               // する必要があります
#include <Servo.h>
#define plus 0xFF18E7    // 時計回りのボタン
#define minus 0xFF4AB5   // 反時計回りのボタン

int RECV_PIN = 3;             // 赤外線受信機のピン
Servo servo;
int val;                      // 回転角度
bool cwRotation, ccwRotation; // 回転の状態

IRrecv irrecv(RECV_PIN);     // リモコン受信機の初期化

decode_results results;

void setup()
{
  Serial.begin(9600);
  irrecv.enableIRIn(); // 受信機を起動する
  servo.attach(9);    // サーボピン
}

void loop()
{
  if (irrecv.decode(&results)) {
    Serial.println(results.value, HEX);
    irrecv.resume(); // Receive the next value
  }
}
```

```

if (results.value == plus)
{
    cwRotation = !cwRotation;           //回転角度の値を切り替えます
    ccwRotation = false;                //これ以上回転しません
}

if (results.value == minus)
{
    ccwRotation = !ccwRotation;
    cwRotation = false;                //回転角度の値を切り替えます
}

if (cwRotation && (val != 175)) {
    val++;                            //連動ボタン用
}

if (ccwRotation && (val != 0)) {
    val--;                            //カウンター連動ボタン用
}

servo.write(val);
delay(20);                         //回転速度
}

```

リモコンの▲/▼ボタンを押してみよう!

同じボタンを2回押すと回転が止まります.

- モーターが動くことが確認出来たらチェック!

数字ボタンを押したら決まった角度に動くように改造しよう!

ArduinoIDEを開き、ファイル→名前を付けて保存をクリックして、「name_lesson_16_3」という名前で保存しましょう。

スケッチに以下のコードをコピー&ペーストして、スケッチを実行してみよう。

```

#include <IRremote.h>           //IRremoteライブラリをアルドゥイーノライブラリにコピー
                                //する必要があります
#include <Servo.h>

#define RIGHT_ARROW   0xFF5AA5 //時計回りに回転する
#define LEFT_ARROW    0xFF10EF //反時計回りに回転する
#define SELECT_BUTTON 0xFF38C7 //サーボの回転位置が中心
#define BUTTON_0 0xFF9867 //ボタン0～9を押すと、決まった位置に移動します
#define BUTTON_1 0xFFA25D // 20度ずつ回転
#define BUTTON_2 0xFF629D
#define BUTTON_3 0xFFE21D
#define BUTTON_4 0xFF22DD
#define BUTTON_5 0xFF02FD
#define BUTTON_6 0xFFC23D
#define BUTTON_7 0FFE01F
#define BUTTON_8 0xFFA857

```

```
#define BUTTON_9 0xFF906F

int RECV_PIN = 3;           //赤外線受信機のピン
int16_t pos;               // サーボ位置を保存する変数を設定する
Servo servo;
IRrecv irrecv(RECV_PIN);
decode_results results;

void setup()
{
    Serial.begin(9600);
    irrecv.enableIRIn(); // 受信機を起動する
    servo.attach(9);    //サーボピン
    pos = 90;           // 中間点90度から開始
    servo.write(pos);   // 初期位置を設定
}

void loop()
{
    if (irrecv.decode(&results)) {
        irrecv.resume();
        switch (results.value) { //ボタンに応じてサーボを動かす
            case LEFT_ARROW:   pos = min(180, pos + 5); break;
            case RIGHT_ARROW:  pos = max(0, pos - 5); break;
            case SELECT_BUTTON: pos = 90; break;
            case BUTTON_0:      pos = 0 * 20; break;
            case BUTTON_1:      pos = 1 * 20; break;
            case BUTTON_2:      pos = 2 * 20; break;
            case BUTTON_3:      pos = 3 * 20; break;
            case BUTTON_4:      pos = 4 * 20; break;
            case BUTTON_5:      pos = 5 * 20; break;
            case BUTTON_6:      pos = 6 * 20; break;
            case BUTTON_7:      pos = 7 * 20; break;
            case BUTTON_8:      pos = 8 * 20; break;
            case BUTTON_9:      pos = 9 * 20; break;
        }
        servo.write(pos);
    }
}
```

他の（プロジェクトやテレビ等の）リモコンでも試してみよう！

- 数字ボタンでサーボモーターが動いたらチェック！
- 左右の矢印でサーボモーターの位置を細かく変更出来たらチェック！

まとめ

- ・ 赤外線受信モジュールを使うためのライブラリはIRremote.h
- ・ サーボモーターを使うためのライブラリはServo.h

出来たことをチェックしよう

- ブレッドボードにサーボモーターと赤外線受信モジュールを使った回路を作ることが出来る
- サンプルコードを実行できる
- サンプルコードを改造して他のリモコンでもモーターを動かすことができる