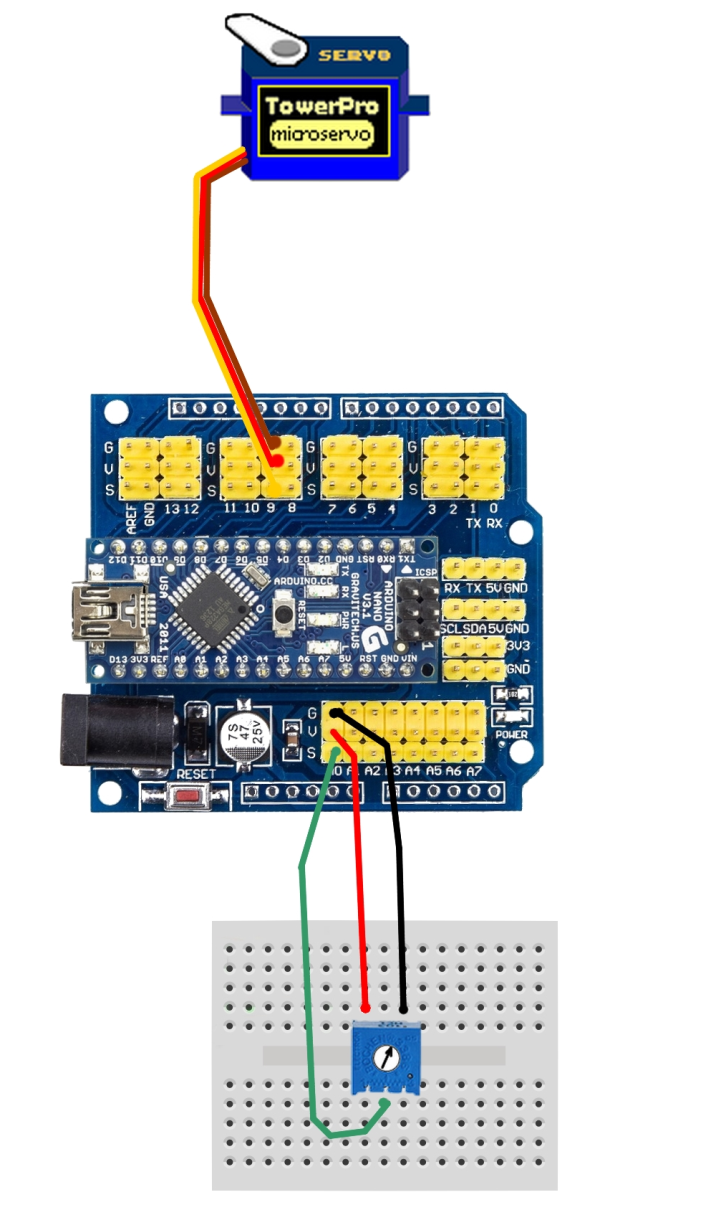
**Ejercicio 6: servosPot**

En este ejercicio vamos a intentar mover los 180º del servo motor a nuestro gusto, a través de un potenciómetro conectado a una entrada analógica de nuestro arduino

Circuito:



El potenciómetro debe ser conectado de manera que los dos pines exteriores estén conectado a alimentación (+5V) y a GND, el pin central lo conectaremos a la entrada analógica “A0” en el Arduino.

En este ejercicio también vamos a incluir una librería para hacernos más fácil el interactuar con un dispositivo, en este caso el servo.

Con un código tan sencillo haremos que el servo se mueva según movamos el potenciómetro de modo que al estar el potenciómetro en el límite inferior de su recorrido el servo estará en la posición de 0º, si centramos el potenciómetro el servo estará en la posición de 90º y si llevamos el potenciómetro a su límite superior el servo alcanzará la posición de 180º.

**Code:**

**// Controlando la posición de un servo usando un potenciómetro (resistencia variable)**

Cargamos la librería <Servo.h> , le al potenciómetro un pin analógico (A0)

**#include <Servo.h>**

Creamos el objeto myservo, asignamos pines y creamos variables

**Servo myservo; // crea el objeto “myservo”**

**int potpin = A0; // Asigna al pin analógico A0 a potenciómetro**

**int val; // variable para leer el valor del potenciómetro.**

En el setup asignamos la variable "myservo" al pin (9) del arduino

**void setup() {**

**myservo.attach(9); // Asigna el objeto “myservo” al pin (9)**

**}**

Leeremos el valor del potenciometro, lo escalamos de 0, 1023 a 0, 180 y le enviamos al servo la posición

**void loop() {**

**val = analogRead(potpin); // Lee el valor del potenciómetro (entre 0 y 1023)**

**val = map(val, 0, 1023, 0, 180); // escala o mapea el valor de val (valor entre 0 y 180)**

**myservo.write(val); // envía a “myservo” la poscion de “val” escalada entre 0 y 180.**

**delay(15); // esperamos 15 ms a que el servo vaya a su nueva posición.**

**}**

Imaginemos que tenemos hecho nuestro Sketch, lo subimos a la placa y, a pesar de estar correctamente escrito (no da errores el compilador), no hace lo que nosotros queremos. El error puede estar en la forma en que trabajamos con los valores de forma interna y eso no lo podemos ver a menos que los mostremos por el monitor serie.

Otro caso puede ser que, una vez realizado nuestro montaje, queramos que el servo comience su recorrido desde una posición determinada y no sabemos cual es. Entonces podemos utilizar el monitor serie para calibrar nuestro servo y su recorrido.

Para ello activaremos el puerto serie incluyendo en el setup() la línea:

|  |  |
| --- | --- |
|  | Serial.begin(9600); |

Esta instrucción inicializará la comunicación con el ordenador a través del puerto serie.

Luego iremos al final del Sketch, justo a la última línea, donde está el comando delay(15) y antes de ella insertaremos el comando que dará salida a los datos que veremos en el monitor serie:

|  |
| --- |
| Serial.println(val); |

Por último, y para darle tiempo al monitor serie a mostrar los datos sin que estos bailen a toda velocidad cambiaremos el valor del comando delay() a 500ms. De esta forma nuestro Sketch moverá el servo y nos mostrará la posición en la que se encuentra cada medio segundo, así podemos dejar el servo en la posición exacta que queramos viendo el número contenido dentro de la variable val.