

Práctica 7. Módulo Joystick analógico (KY-023) con Arduino UNO

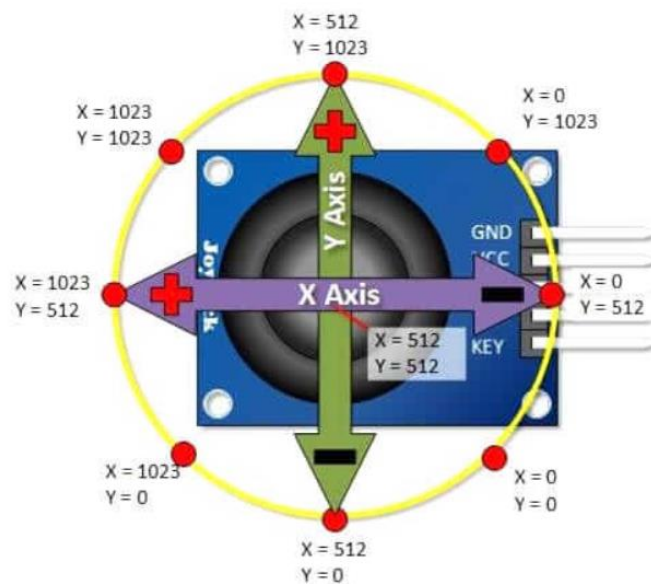
En esta práctica de implementación, consiste en utilizar el módulo Joystick para encender LEDs mediante PWM y obtener una indicación visual del movimiento de la palanca y cierre del interruptor incorporado.

El Módulo KY-023 Sensor JoyStick es un dispositivo electromecánico consta de dos potenciómetros en un ángulo de 90 grados, lo que se requiere de 2 pines analógicos para realizar la interfaz con cualquier tarjeta de desarrollo: Arduino



Este elemento te permite controlar y manejar determinados aparatos electrónicos. Normalmente se utilizan para proyectos robóticos en el cual se necesitan para la movilidad analógica de las articulaciones de un brazo robótico. El Módulo Joystick, es más utilizado para proyectos de robótica y control de dispositivos RF(Radio Frecuencia).

El módulo KY-023 cuenta con un posicionamiento de eje dado por valores entre 0 a 1023 y conforme lo movamos su valor ira variando y dando las coordenadas de cada eje, este valor lo podemos imprimir en el monitor serial en el IDE de Arduino.



Objetivo

Utilizar el módulo JoyStick para encender LEDs mediante PWM y obtener una indicacion visual del movimiento de la palanca.

Equipo y Materiales

1 Arduino UNO

1 Protoboard

1 Módulo JoyStick KY-023

2 LED rojo

2 LED verde

1 LED amarillo

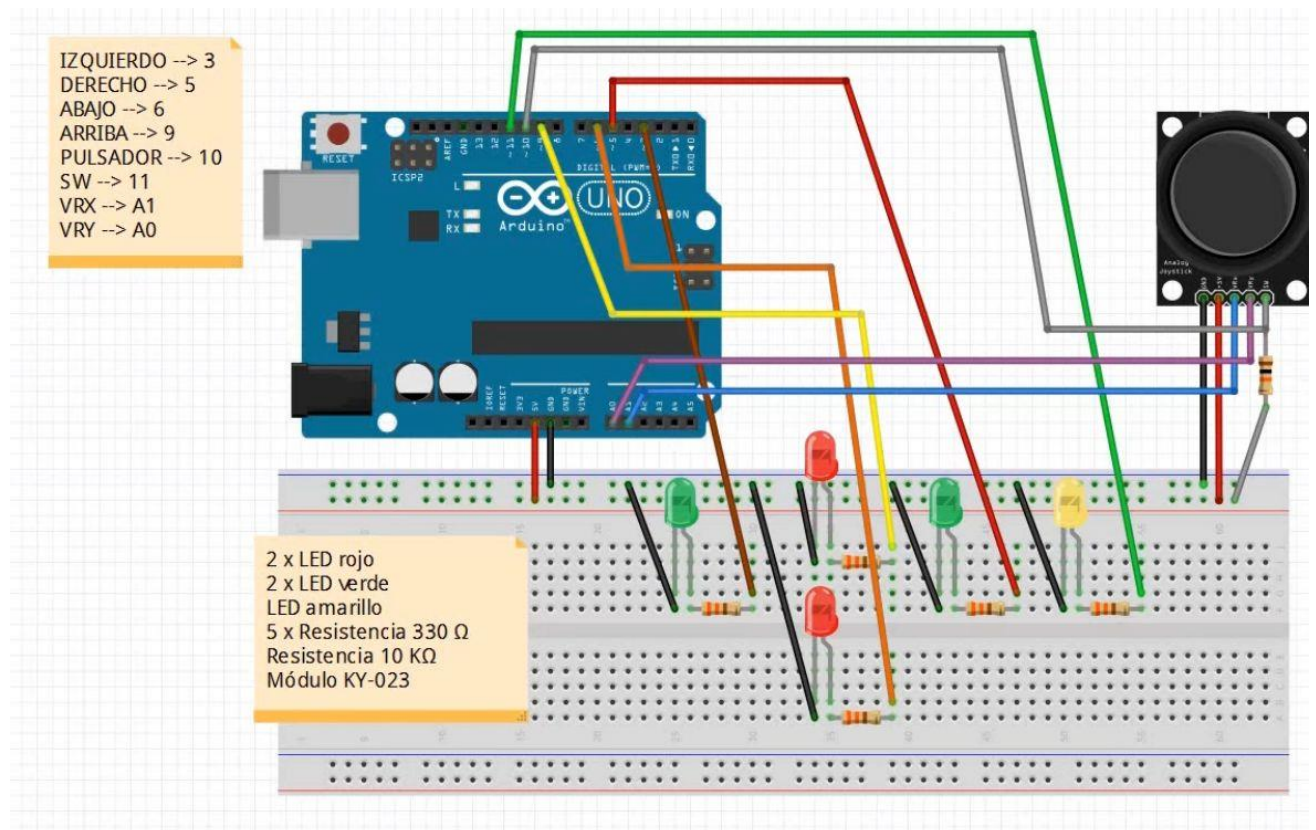
5 Resistencias de 330 o 220

1 Resistencia de 10K

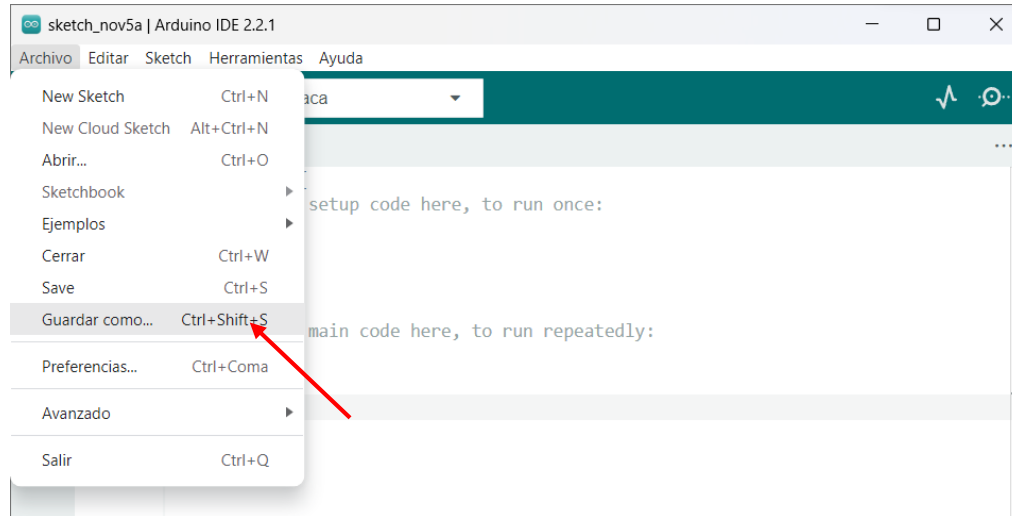
IDE Arduino

Procedimiento

PASO 1. Realice el circuito con sus componentes físicos. A continuación, se muestra cómo deben de realizarse la comunicación física entre el modulo, los componentes y el Arduino.



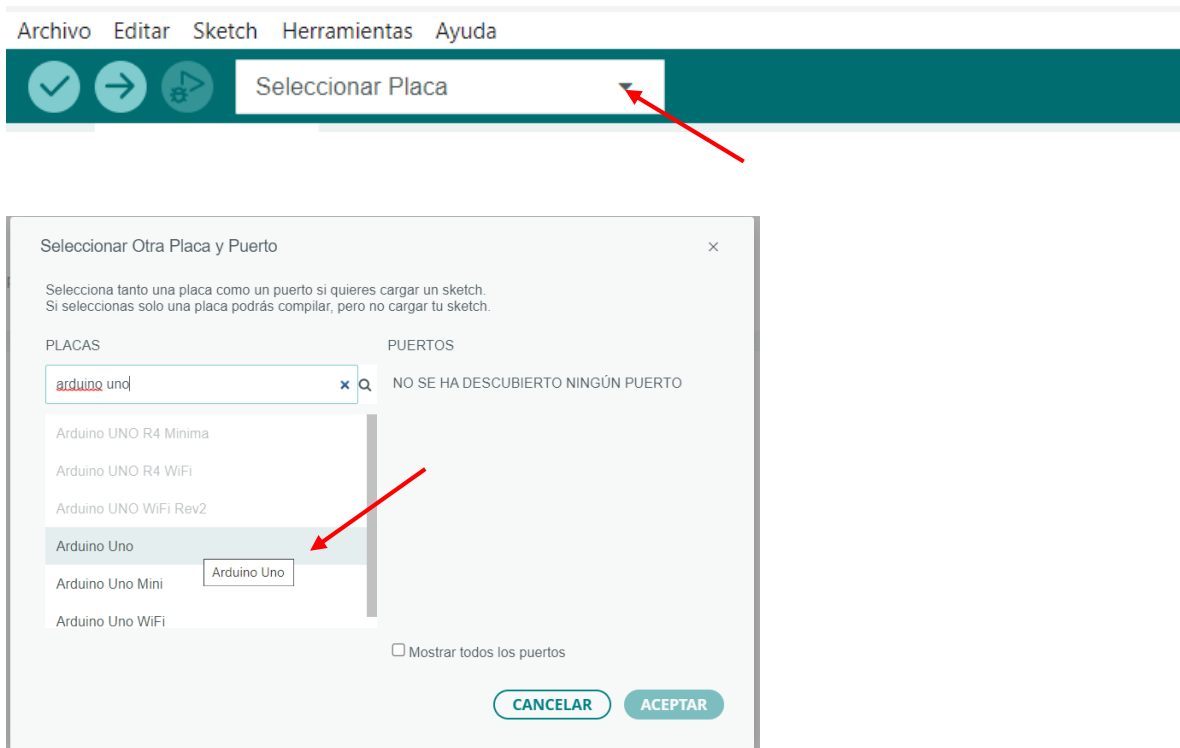
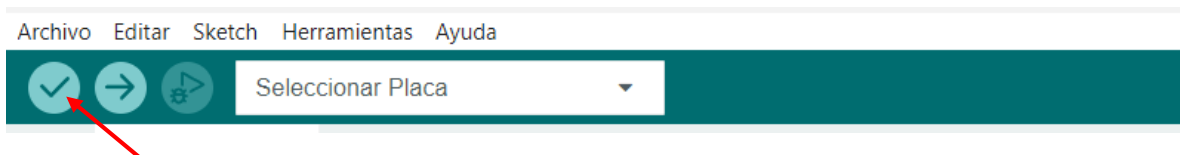
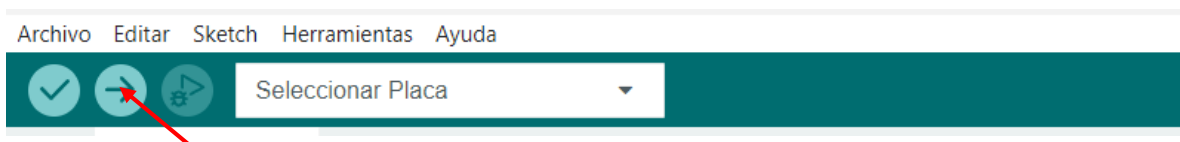
PASO 2. Diríjase al IDE Arduino. Siempre al iniciar deberá de nombrar su proyecto. En la parte superior de click sobre “Archivo” y en “guardar como”, borre el nombre predefinido y sustitúyalo por “JoyStick_KY-023.ino”



PASO 3. Una vez guardado el archivo, borre el contenido que le aparece y sustitúyalo por el siguiente código.

```
JoyStick_KY-023.ino
1  /*
2   Programa que utiliza el modulo joystick KY-023 para encender LEDs mediante PWM
3   y obtener una indicacion visual del movimiento de la palanca y cierre del
4   interruptor incorporado.
5  */
6
7  int X;          // variable para almacenar valor leído del eje x
8  int Y;          // variable para almacenar valor leído del eje y
9  int LED_IZQUIERDO = 3;  // LED izquierdo a pin digital 3
10 int LED_DERECHO = 5;    // LED derecho a pin digital 5
11 int LED_ABAJO = 6;     // LED de abajo a pin digital 6
12 int LED_ARRIBA = 9;    // LED de arriba a pin digital 9
13 int PULSADOR = 10;     // pulsador incorporado pin digital 10
14 int LED_SW = 11;       // LED de pulsador a pin digital 11
15 int SW;             // variable para almacenar valor leído del pulsador
16
17 void setup(){
18   pinMode(LED_IZQUIERDO, OUTPUT); // LED como salida
19   pinMode(LED_DERECHO, OUTPUT);  // LED como salida
20   pinMode(LED_ABAJO, OUTPUT);    // LED como salida
21   pinMode(LED_ARRIBA, OUTPUT);   // LED como salida
22   pinMode(LED_SW, OUTPUT);       // LED como salida
23   pinMode(PULSADOR, INPUT);      // pulsador como entrada
24   // entradas analogicas no requieren inicializacion
25 }
```

```
27 void loop(){
28
29   X = analogRead(A0);    // lectura de valor de eje x
30   Y = analogRead(A1);    // lectura de valor de eje x
31   SW = digitalRead(PULSADOR); // lectura de valor de pulsador
32   if (X >= 0 && X < 480){    // si X esta en la zona izquierda
33     | analogWrite(LED_IZQUIERDO, map(X, 0, 480, 255, 0)); // brillo LED proporcional
34   } else {
35     | analogWrite(LED_IZQUIERDO, 0);           // X en zona de reposo, apaga LED
36   }
37   if (X > 520 && X <= 1023){    // si X esta en la zona derecha
38     | analogWrite(LED_DERECHO, map(X, 520, 1023, 0, 255)); // brillo LED proporcional
39   } else {
40     | analogWrite(LED_DERECHO, 0);           // X en zona de reposo, apaga LED
41   }
42   if (Y >= 0 && Y < 480){    // si Y esta en la zona de abajo
43     | analogWrite(LED_ABAJO, map(Y, 0, 480, 255, 0)); // brillo LED proporcional
44   } else {
45     | analogWrite(LED_ABAJO, 0);           // Y en zona de reposo, apaga LED
46   }
47   if (Y > 520 && Y <= 1023){    // si Y esta en la zona de arriba
48     | analogWrite(LED_ARRIBA, map(Y, 520, 1023, 0, 255)); // brillo LED proporcional
49   } else {
50     | analogWrite(LED_ARRIBA, 0);           // Y en zona de reposo, apaga LED
51   }
52
53   digitalWrite(LED_SW, !SW); // escribe en LED valor opuesto al leído del pulsador
54
55 }
```

PASO 4. Seleccione la placa que está utilizando**PASO 5.** Verifique que su programa no tenga errores.**PASO 6.** Conecte el cable del Arduino a este y al puerto USB de su computadora.**PASO 7.** Cargue el programa

CUESTIONARIO

1. ¿Qué función tiene este módulo?
2. ¿Qué proyecto harías con este módulo? (Al menos 3 ideas)
3. Busca en internet otro proyecto simple con este módulo y realízalo.