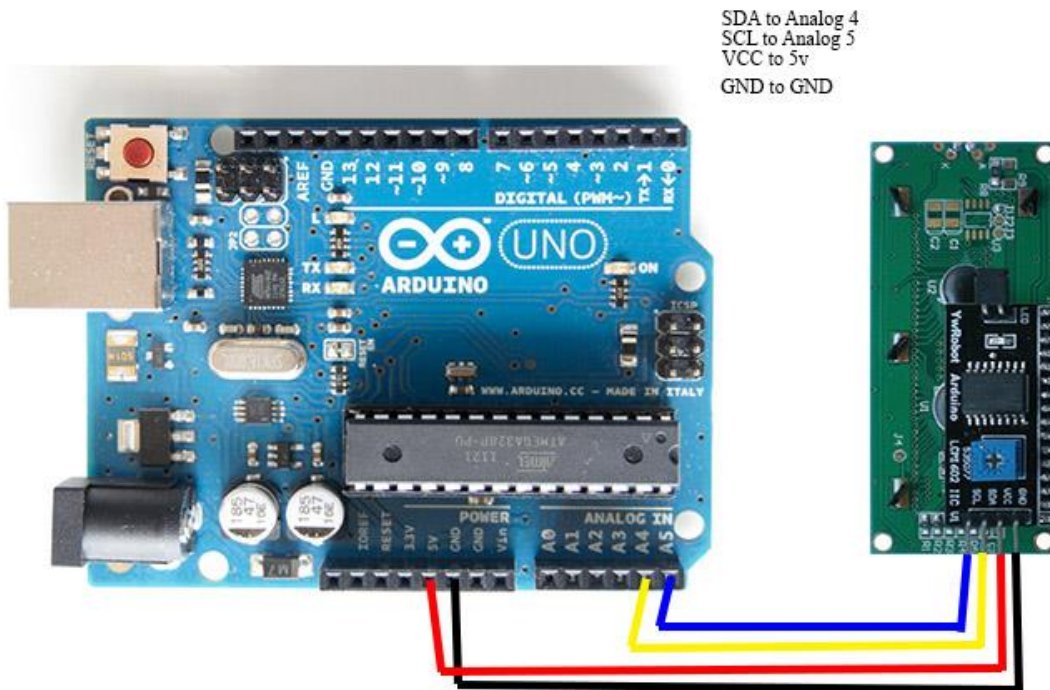


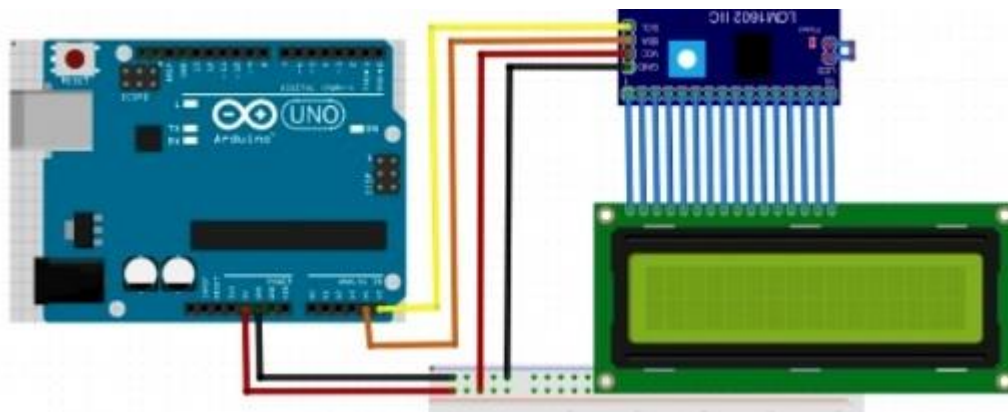
Taller de Arduino

Pantalla LCD con controlador I2C.	2
Menú de opciones.	3
Convertidor ADC.	3
Sensor de temperatura LM35.	4
Joystick analógico.	5
Servomotor.	6
Servomotor y joystick analógico.	8
Sensor ultrasónico.	9
PWM.....	10

Pantalla LCD con controlador I2C.



Conexión de controlador I2C con Arduino.



Conexión de controlador I2C con Arduino y pantalla LCD.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,20,4);
void setup(){
  lcd.init();
  lcd.backlight();
  lcd.print("Hello, world!");
}

void loop(){
  lcd.setCursor(0, 1);
  lcd.print(millis() / 1000);
}
```

Menú de opciones.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
  Serial.begin(9600);
  Serial.flush();
  lcd.begin(16, 2);
  lcd.print("hello, world!");
}

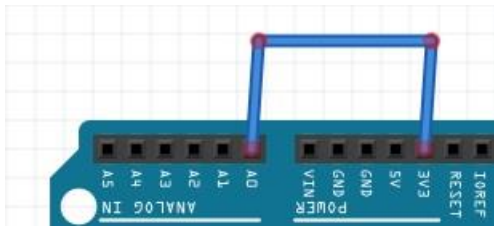
void loop() {
  lcd.setCursor(0, 1);
  lcd.print(millis() / 1000);
  switch(get_tecla()){
    case '0':lcd.blink();Serial.println("Opcion 0");break;
    case '1':lcd.noBlink();Serial.println("Opcion 0");break;
    default:Serial.println("Tecla incorrecta");
  }
}
```

```
char get_tecla(){
  while(Serial.available() == 0);
  return Serial.read();
}
```

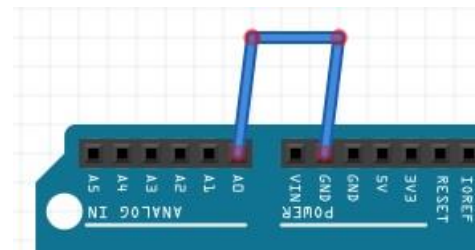
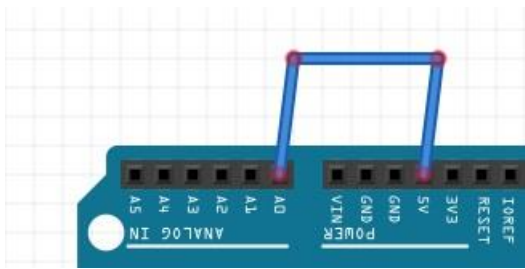
Convertidor ADC.

```
int canal=0;
void setup()
{
  Serial.begin(9600);
  Serial.flush();
}
void loop()
{
  int lectura=analogRead(canal);
  Serial.print("lectura= ");
  Serial.print(lectura);
  Serial.print(" ");
  Serial.println(lectura, HEX);
  float vin=(5.0/1023.0)*lectura;
  Serial.print("vin= ");
  Serial.print(vin);
  Serial.print(" ");
  Serial.println(vin, hex);
  delay(1000);
}
```

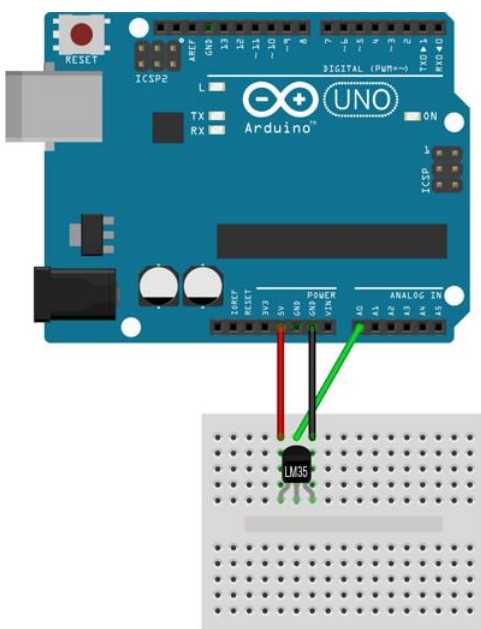
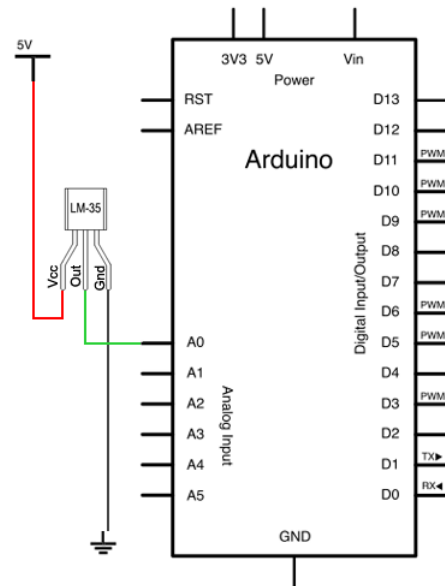
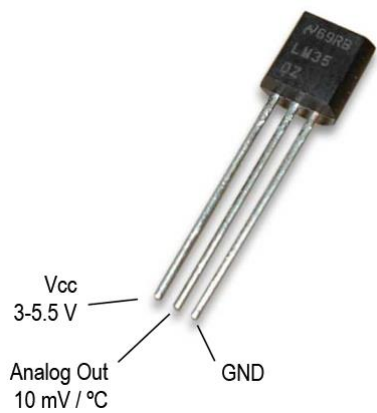
Conecte el canal 0 del ADC a 3.3 V. Abra el monitor serie y reporte los resultados.



Posteriormente conecte a GND (tierra o 0 V) y 5 V el canal analógico 0 y observe las lecturas que aparecen en el monitor serie.



Sensor de temperatura LM35.

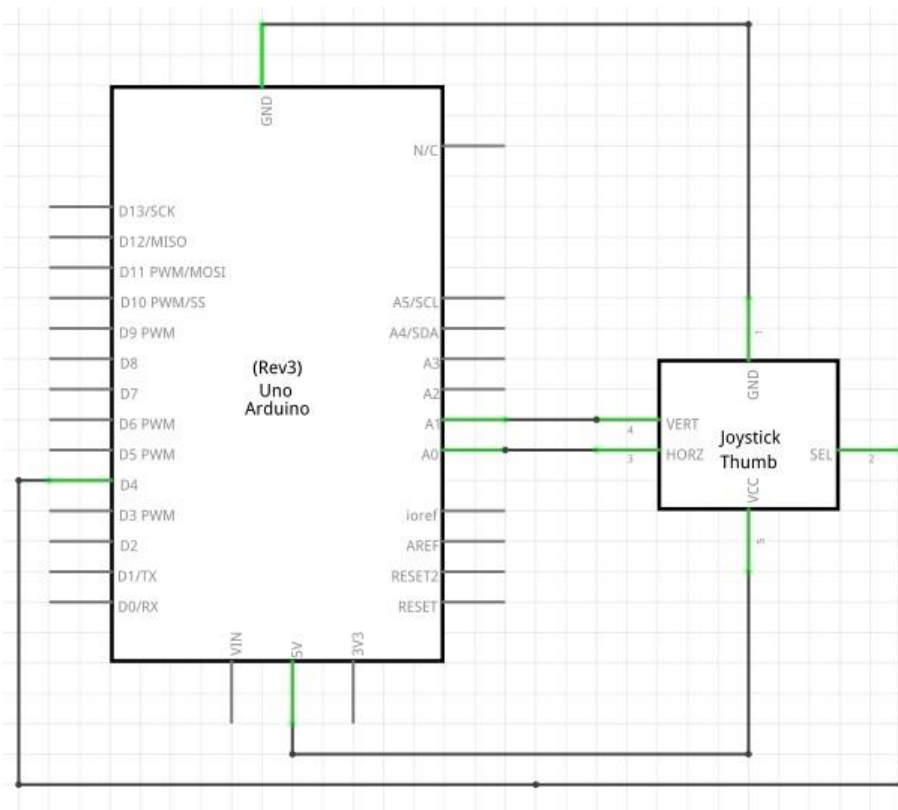


```
void setup() {  
  Serial.begin(9600);  
  Serial.flush();  
}  
void loop() {  
  int lectura = analogRead(0);  
  float Vin = 5.0 * lectura / 1023.0;  
  float temp = Vin / 0.01;  
  Serial.print("Lectura= ");  
  Serial.print(lectura);  
  Serial.print("  Vin= ");  
  Serial.print(Vin);  
  Serial.print("  Temp= ");  
  Serial.println(temp);  
  delay(1000);  
}
```

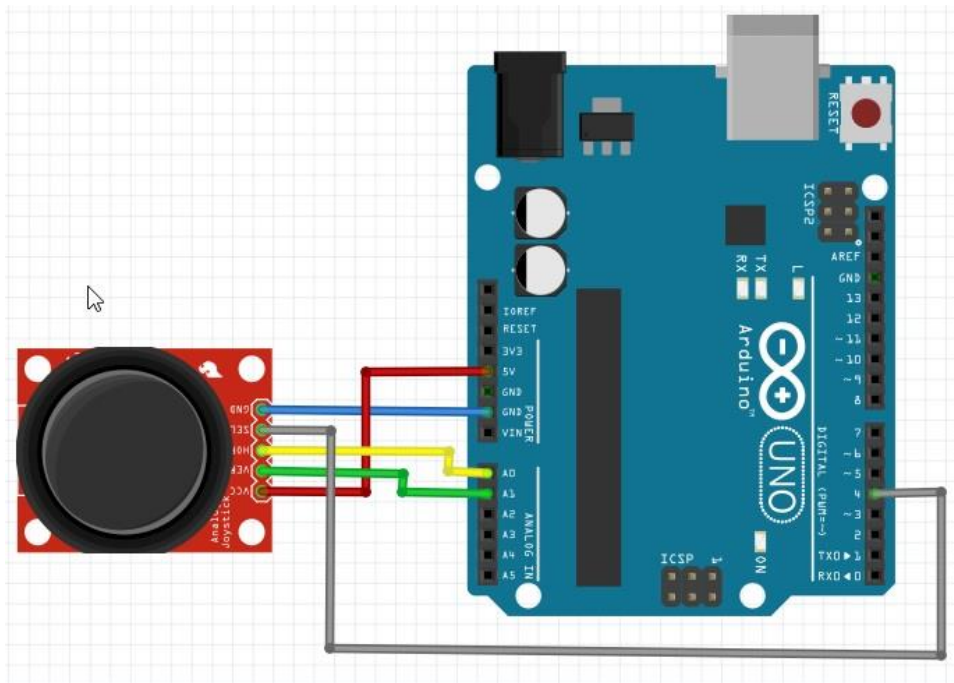
Modificar el programa de la actividad 2 por el siguiente código, compilarlo y en caso de errores corregirlos, salvar el código, cargarlo al Arduino y abrir el monitor serie para observar los resultados. Observar las diferencias.

```
void setup() {  
  Serial.begin(9600);  
  Serial.flush();  
  analogReference(INTERNAL);  
}  
void loop() {  
  int lectura = analogRead(0);  
  float Vin = 1.1 * lectura / 1023.0;  
  float temp = Vin / 0.01;  
  Serial.print("Lectura= ");  
  Serial.print(lectura);  
  Serial.print("  Vin= ");  
  Serial.print(Vin);  
  Serial.print("  Temp= ");  
  Serial.println(temp);  
  delay(1000);  
}
```

Joystick analógico.



Observar cuidadosamente el joystick comprado ya que las terminales pueden variar con respecto la figura siguiente.



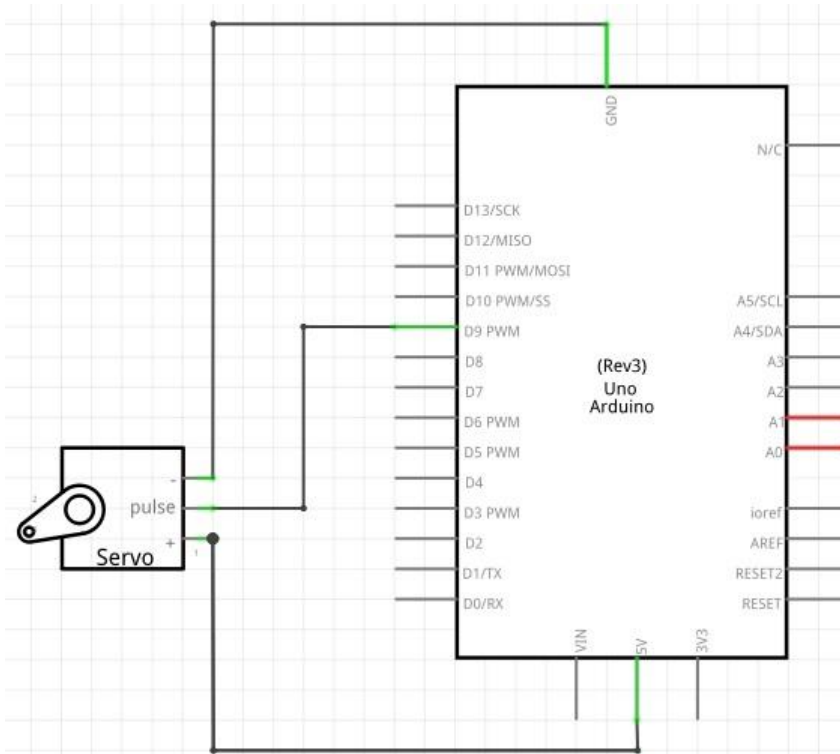
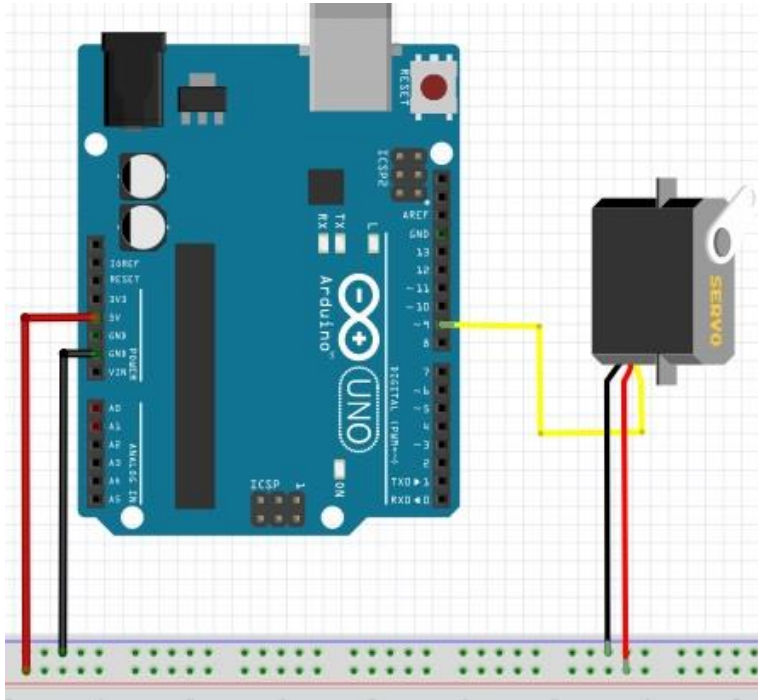
```
int Jx = 0;
int Jy = 1;
int valx, valy;
void setup() {
  Serial.begin(9600);
  Serial.flush();
}
```

```
void loop() {
  valx = analogRead(Jx);
  valy = analogRead(Jy);
  Serial.print("Jx = ");
  Serial.print(valx);
  Serial.print(" Jy = ");
  Serial.println(valy);
  delay(1000);
}
```

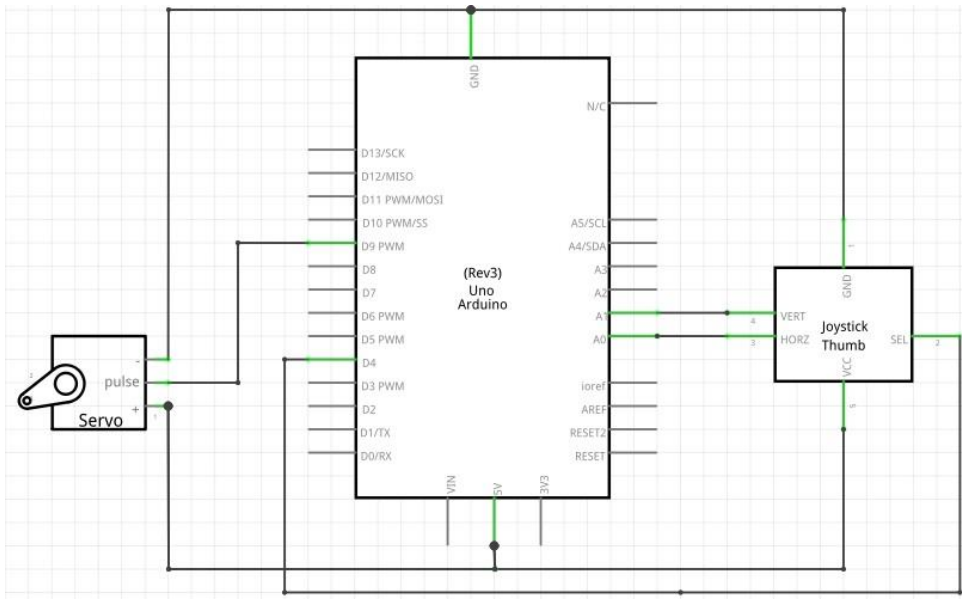
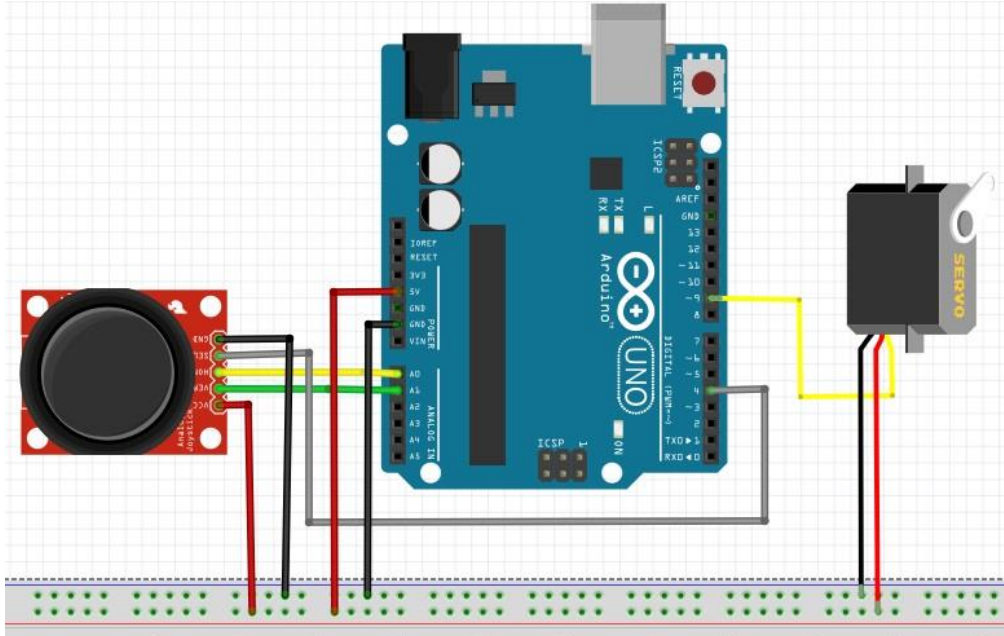
Servomotor.

```
#include <Servo.h>
Servo myservo;
int potpin = 0;
int val;
void setup() {
  myservo.attach(9);
  Serial.begin(9600);
  Serial.flush();
}
```

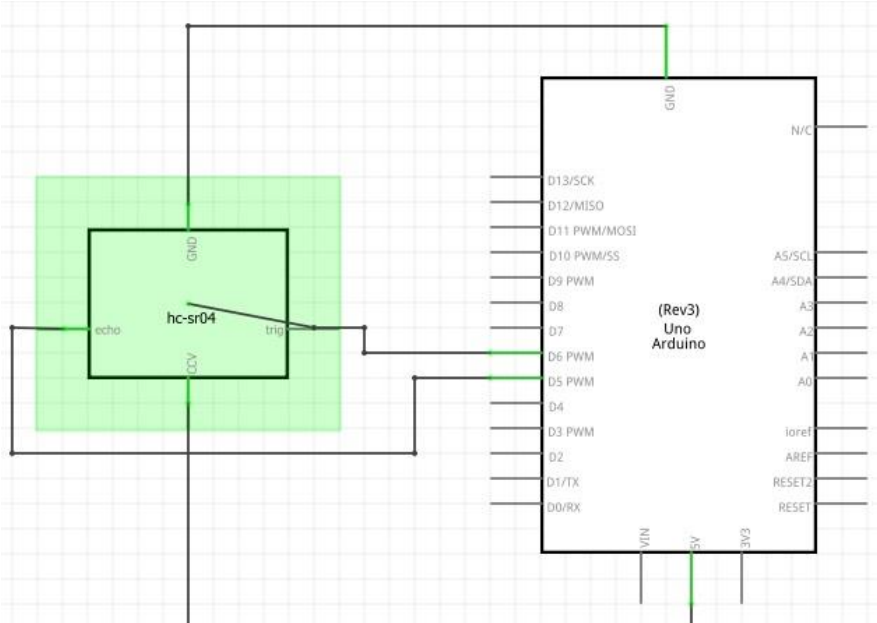
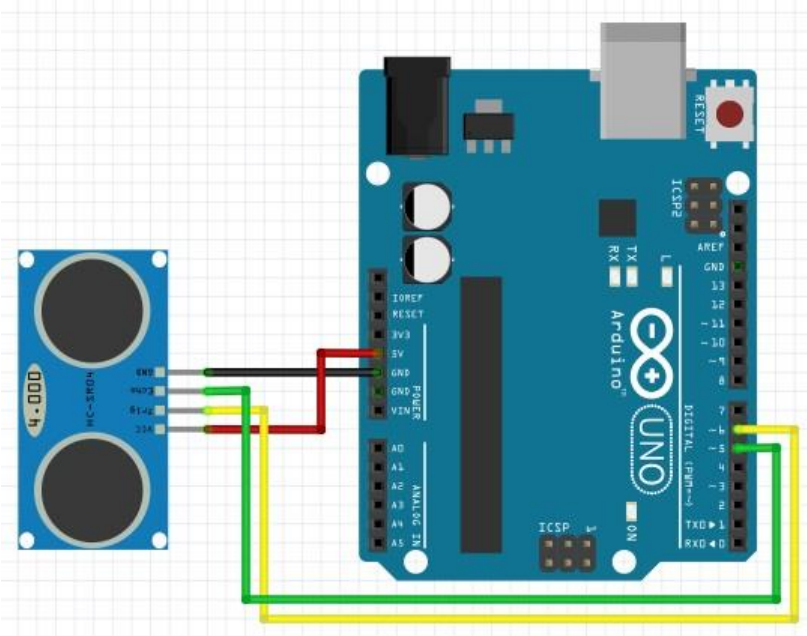
```
void loop() {
  //val = analogRead(potpin);
  Serial.println("cero");
  for(val=0;val<=1023;val++){
    int valor = map(val, 0, 1023, 0, 180);
    myservo.write(valor);
    delay(50);
  }
  Serial.println("180");
  delay(1000);
}
```



Servomotor y joystick analógico.



Sensor ultrasónico.



```

int eco=5, disparo=6;
void setup() {
  pinMode(eco, INPUT);
  pinMode(disparo, OUTPUT);
  Serial.begin(9600);
  Serial.flush();
}

```

```

void loop() {
  Serial.print("distancia = ");
  Serial.print(dist());
  Serial.println(" cm");
  delay(1000);
}

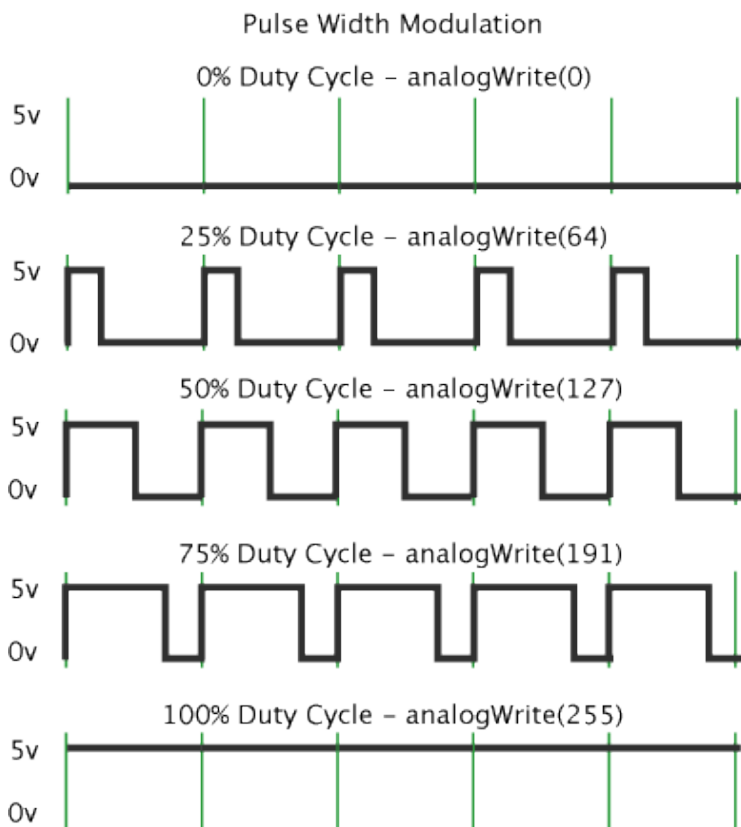
```

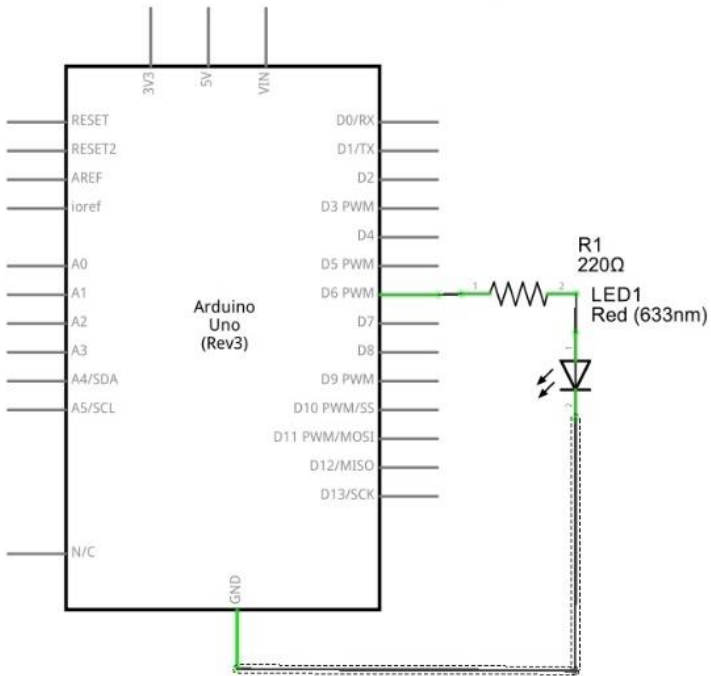
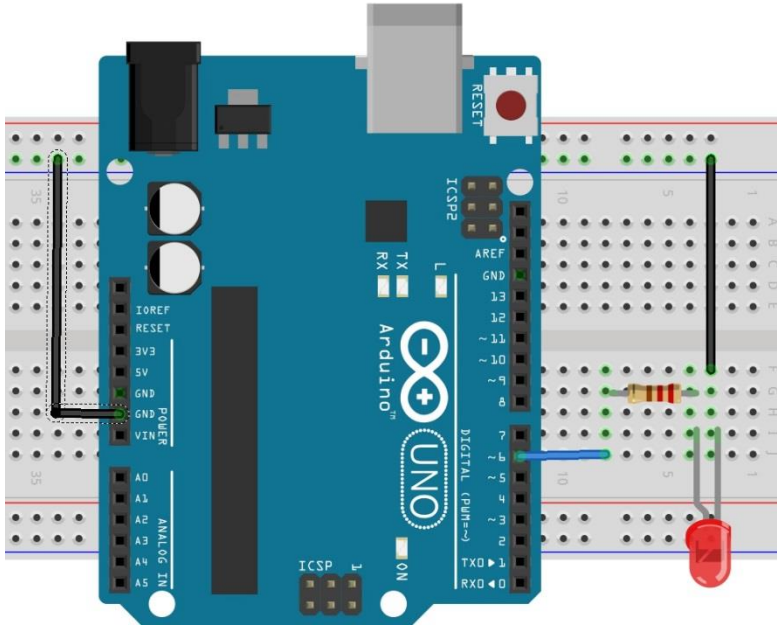
```

float dist(){
  digitalWrite(disparo,LOW);
  delayMicroseconds(4);
  digitalWrite(disparo,HIGH);
  delayMicroseconds(10);
  digitalWrite(disparo,LOW);
  float distancia=342.2*pulseIn(eco,HIGH)/10000.0/2.0;
  return distancia;
}

```

PWM.





```

#define TIEMPO 30
void setup() {
  Serial.begin(9600);
  Serial.flush();
}
void loop() {
  unsigned int i;
  Serial.println("Prendiendo LED");
  for (i = 0 ; i <= 255; i++) {
    analogWrite(6, i);
    delay(TIEMPO);
  }
  Serial.println("Apagando LED");
  for (i = 254 ; i >= 1; i++) {
    analogWrite(6, i);
    delay(TIEMPO);
  }
}

```

```

#define MAX 255
#define MIN 0
#define sensi 10
int valor=125;
int salida=6;
void setup() {
  Serial.begin(9600);
  Serial.flush();
  analogWrite(salida,valor);
}
void loop() {
  if(Serial.available()>0)
  {
    switch(Serial.read()) {
      case 'Q':
      case 'q': (valor<(sensi-1)) ? valor=0 : valor--sensi;
                Serial.println(valor);
                break;
      case 'W':
      case 'w': (valor>(255-sensi)) ? valor=255 : valor+=sensi;
                Serial.println(valor);
                break;
    }
  }
  analogWrite(salida,valor);
}

```