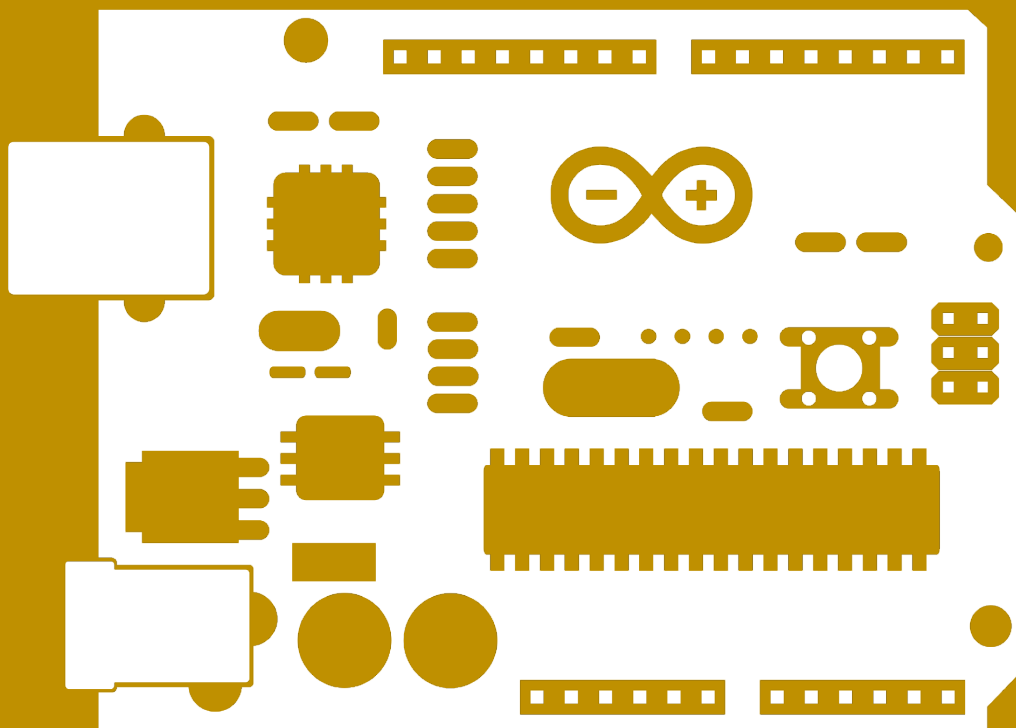




HUERTO INTELIGENTE

Curso 2024-2025
Provincia de Burgos

PROYECTO DE INNOVACIÓN EDUCATIVA
Código Escuela 4.0



MINISTERIO
DE EDUCACIÓN, FORMACIÓN PROFESIONAL
Y DEPORTES



Programa financiado por el Ministerio de Educación, Formación Profesional y Deportes

PARTE I. Conocimientos previos



1	Como conectar LCD a arduino.....	1
1.1	Con potenciometro	1
1.1.1	Conexión de la placa.....	1
1.1.2	Código en IDE Arduino.....	2
1.2	Sin potenciometro.....	4
2	Como instalar el complemento en Excel.....	5
2.1	Como conectar Arduino a Excel	6
2.2	Como hacer que arduino muestre datos en el monitor serie (y por lo tanto en nuestro excel)	7
3	Conceptos básicos sobre sensores	10
3.1	Sensor de humedad del suelo	10
3.1.1	Descripción	10
3.1.2	Aplicaciones.....	10
3.1.3	Características	10
3.1.4	Ventajas.....	10
3.1.5	Características técnicas del sensor	11
3.2	Sensor DHT11(Humedad y Temperatura)	11
3.2.1	Descripción	11
3.2.2	Características técnicas del sensor	11
3.3	Sensor de Nivel del agua	11
3.3.1	Descripción	11
3.3.2	Características técnicas del sensor	12

PARTE II. Proyectos



4	Huerto inteligente NIVEL INICIACIÓN	13
4.1	Diagrama de conexión.....	13
4.2	Códigos Fuente	13
5	Huerto inteligente NIVEL MEDIO.....	15
5.1	Conexión de sensores.....	15
6	Huerto inteligente NIVEL AVANZADO	18

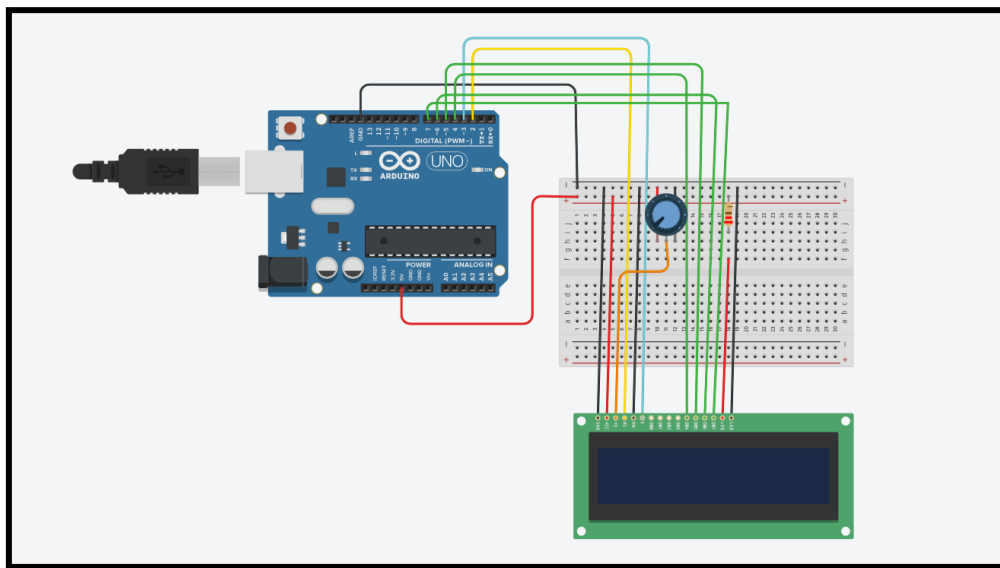
6.1	Conexión de sensores.....	18
6.2	Ejemplo de sensor de humedad de a tierra y bomba de agua.....	20

PARTE I. Conocimientos previos

1 COMO CONECTAR LCD A ARDUINO

1.1 CON POTENCIOMETRO

1.1.1 Conexión de la placa



A tierra:

- VSS
- RW
- K

A 5v

- VDD
- A (mediante resistencia de 220 Ω)

A pines digitales (estos pines, hay que indicarlos en ArduinoBlocks)

- RS
- E
- D4
- D5
- D6
- D7

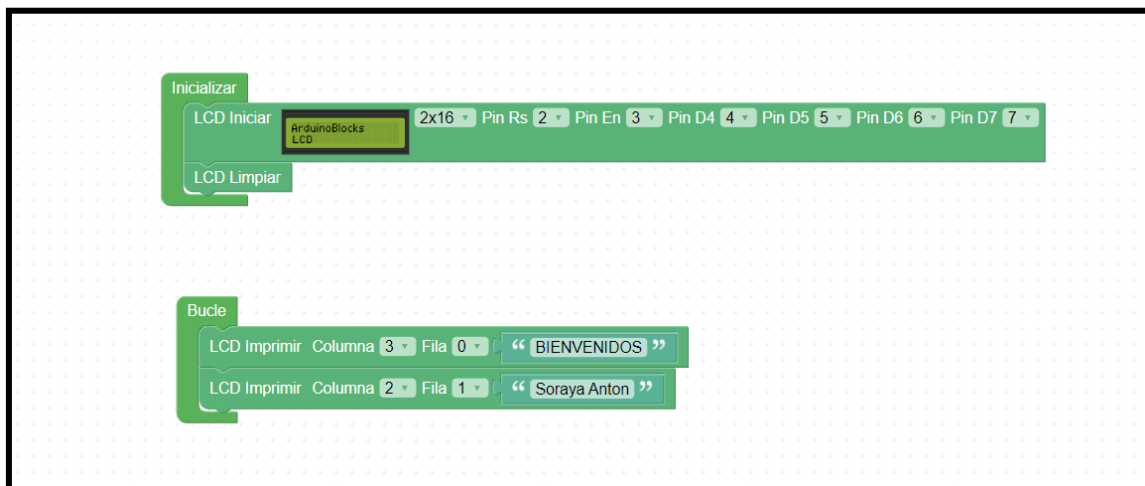
A potenciómetro (o resistencia de 1K Ω)

- V0

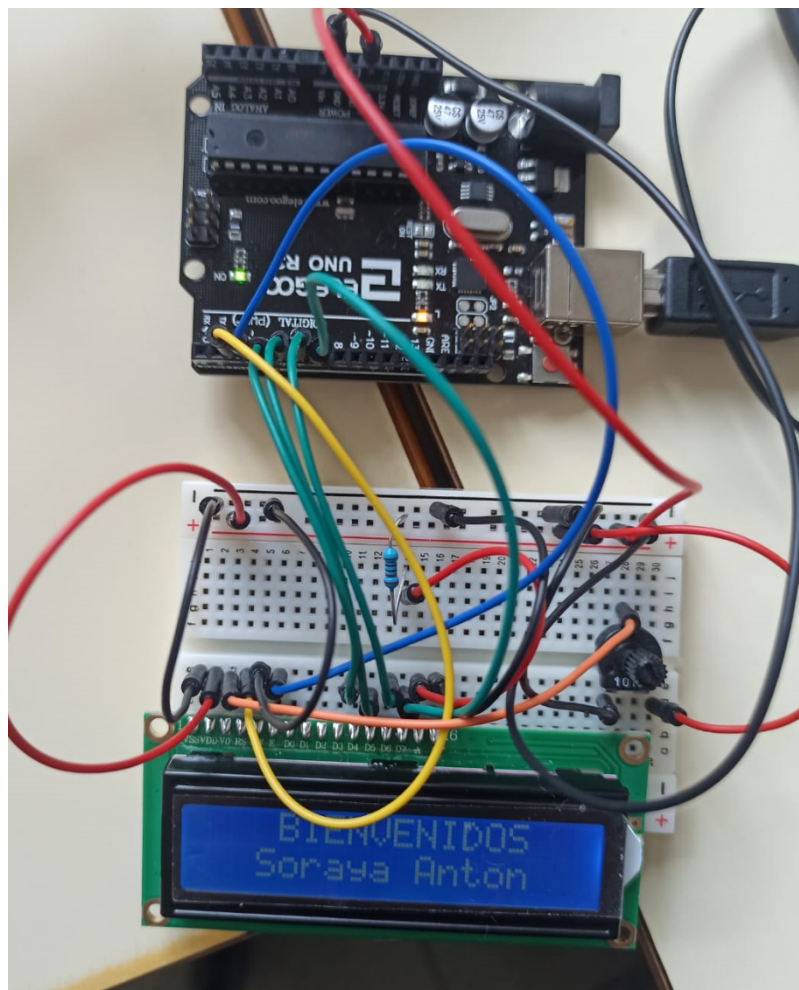
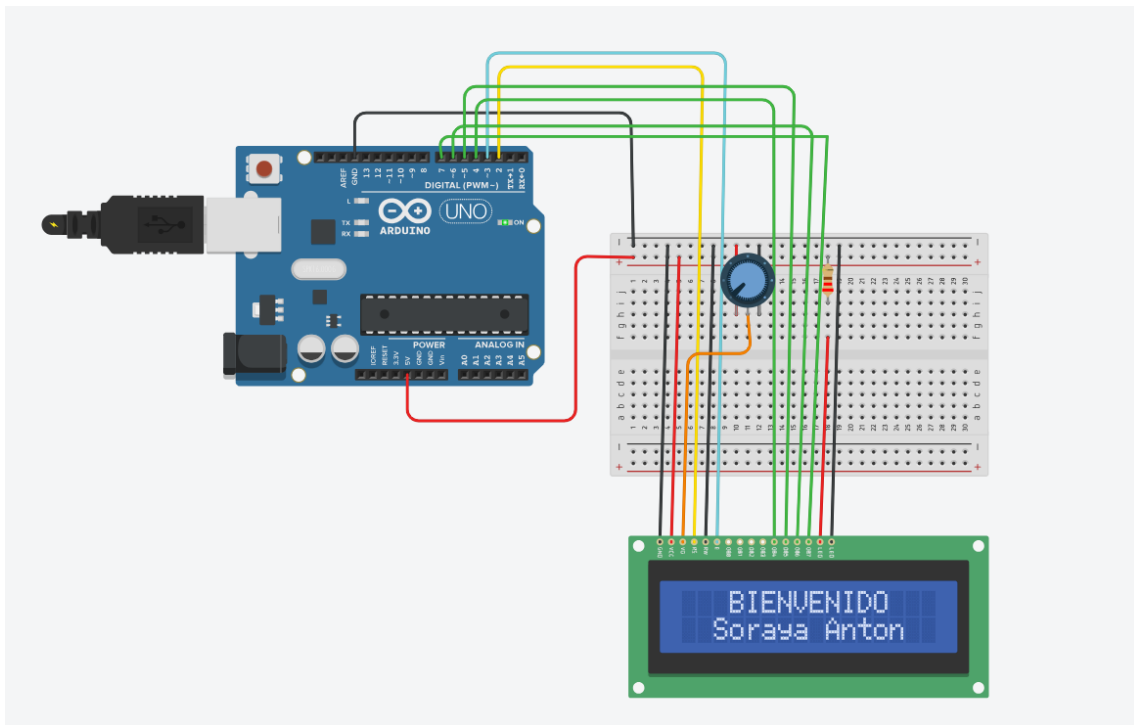
1.1.2 Código en IDE Arduino

```
1  #include <LiquidCrystal.h>
2
3  LiquidCrystal lcd(2,3,4,5,6,7); //Nombramos nuestra LCD con el nombre lcd
4
5  void setup()
6  {
7      lcd.clear();
8      lcd.begin(16,2); //Inicializamos la pantalla y le decimo que es del tamaño 16 x 2
9  }
10
11 void loop()
12 {
13
14     lcd.setCursor(3,0); //Donde empieza a escribir
15     lcd.print("BIENVENIDO");
16     lcd.setCursor(2,1); //Donde empieza a escribir
17     lcd.print("Soraya Anton");
18
19
20 }
```

Bloque en ARDUINOBLOCKS

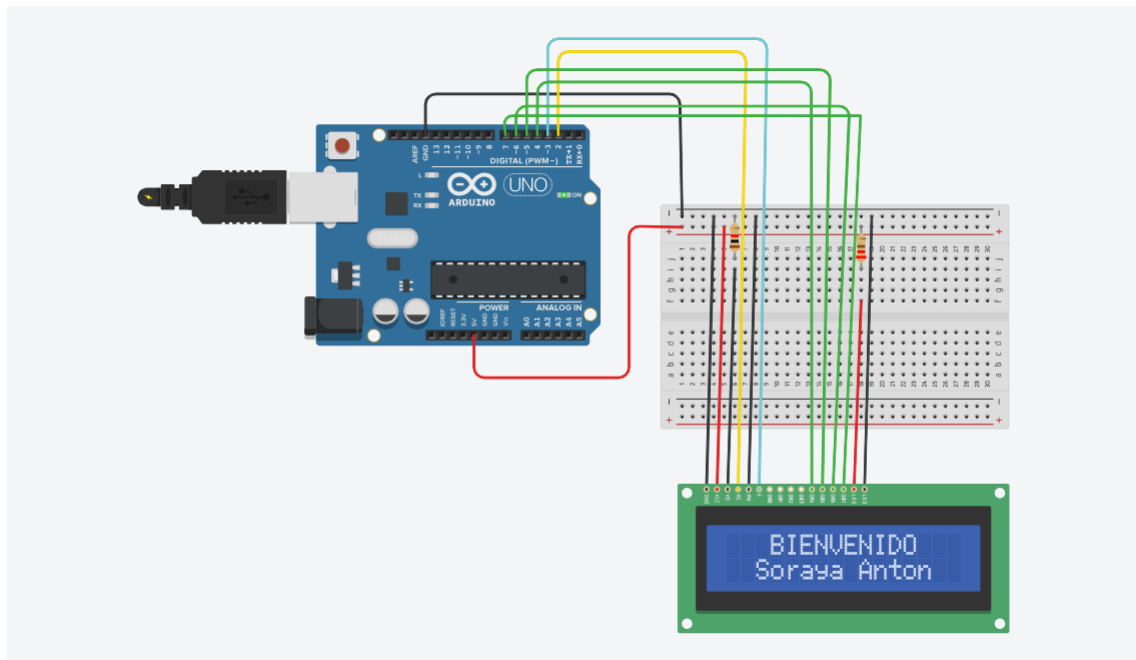


Resultado



1.2 SIN POTENCIÓMETRO

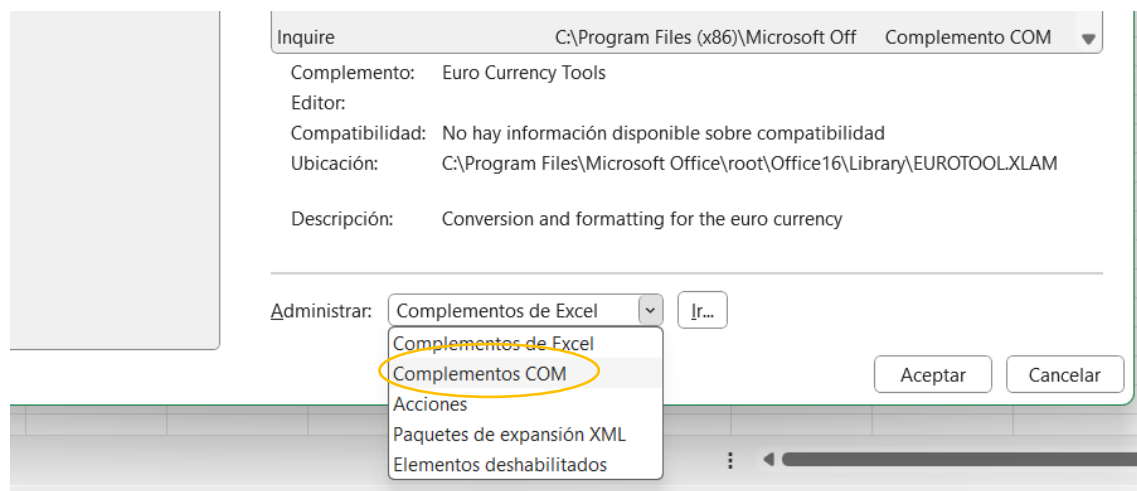
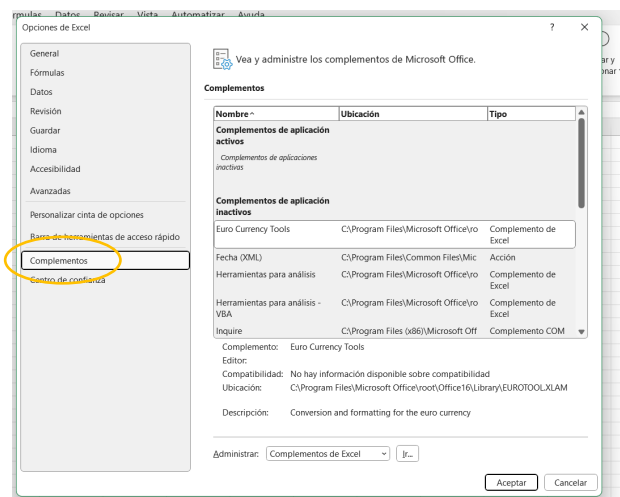
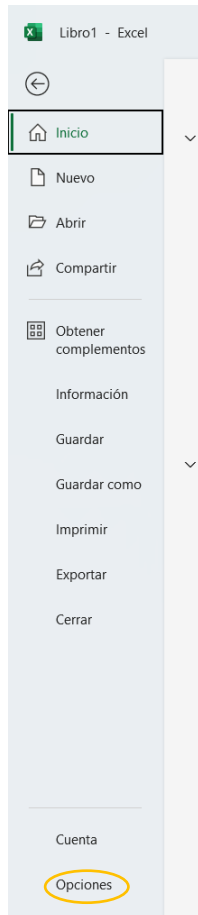
El montaje es igual, excepto que en vez de un potenciómetro conectamos, a ese pin (V0) una resistencia de 1KΩ a tierra.



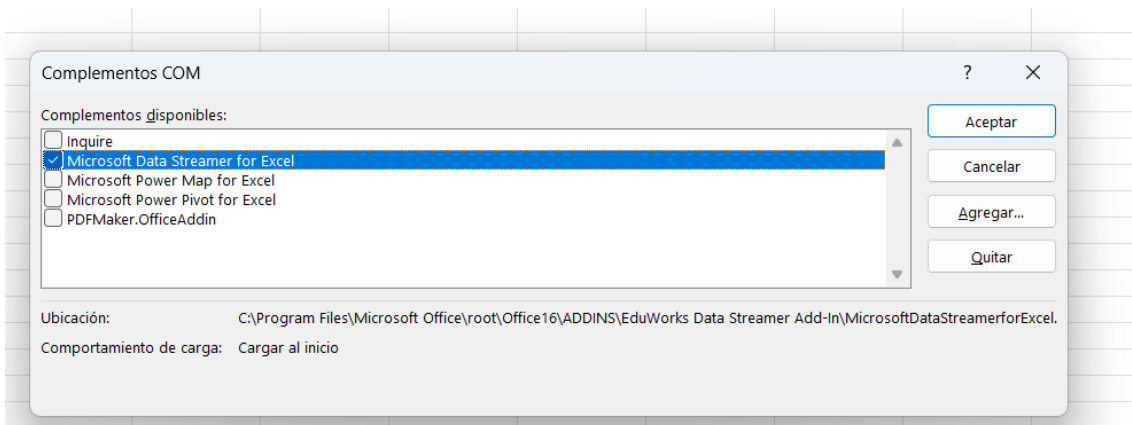
La programación no varía

2 COMO INSTALAR EL COMPLEMENTE EN EXCEL

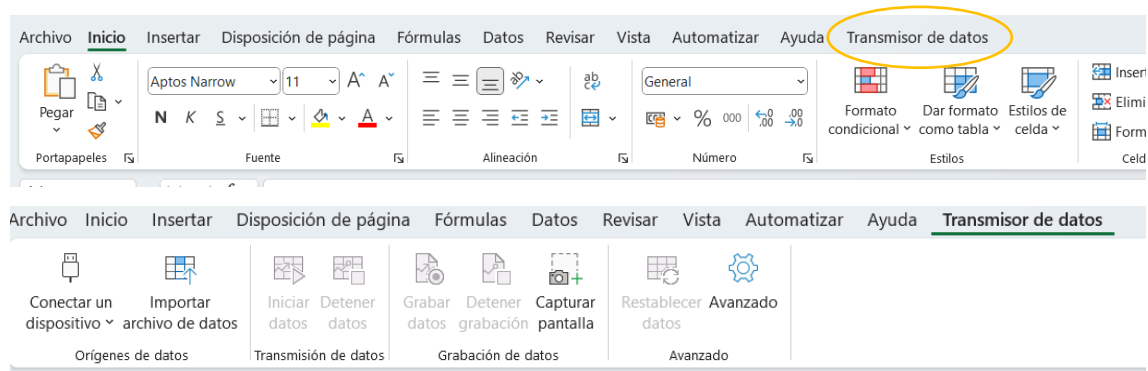
Archivo >> opciones >> Complementos >> Complementos COM >> Microsoft Data Streamer for Excel



Marcamos complementos COM y damos a IR



Aparecerá una nueva opción en nuestra cinta de Word



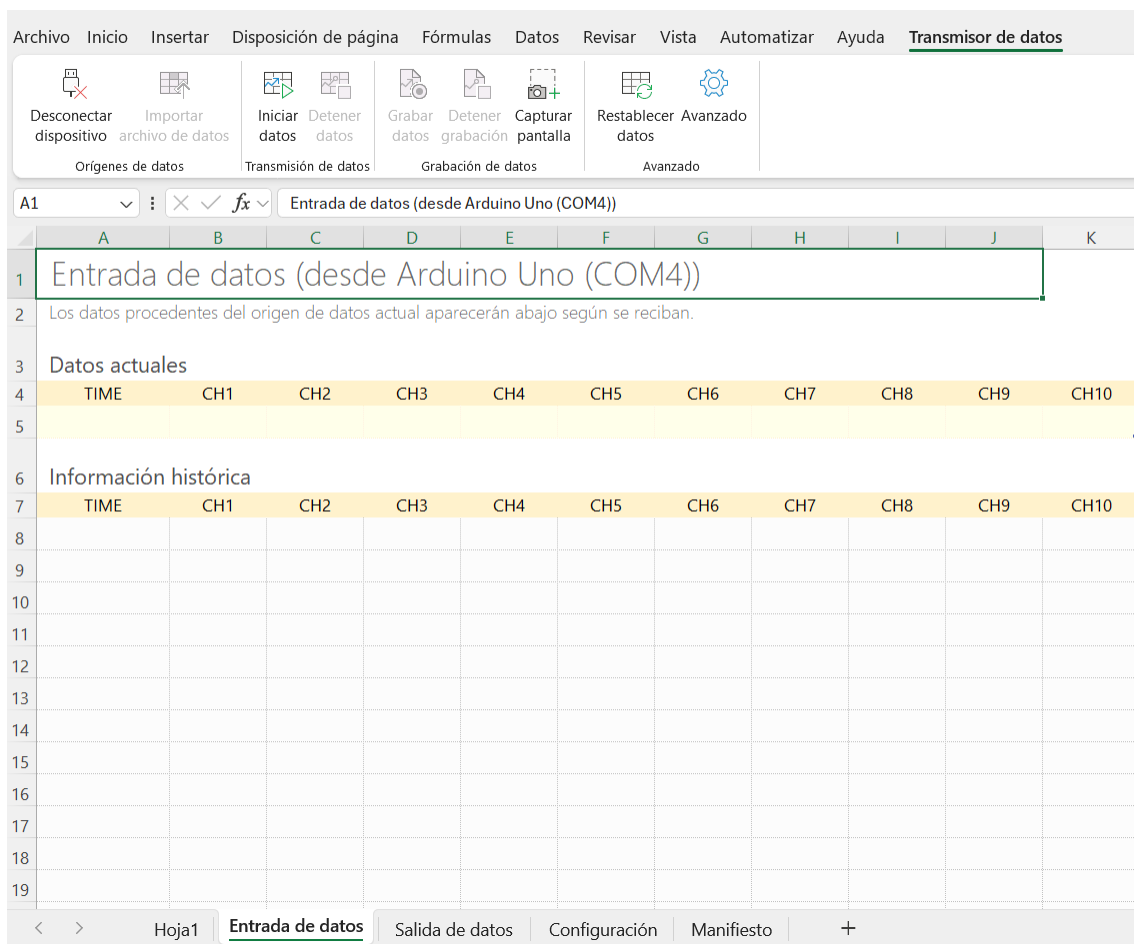
Ya tenemos el complemento instalado y listo para usar.

2.1 COMO CONECTAR ARDUINO A EXCEL

Una vez hemos cargado nuestro programa en Arduino, abrimos Excel, le damos a *conectar dispositivo*, cuando conecte con nuestra placa le daremos a iniciar datos para que empiece a volcar los datos.

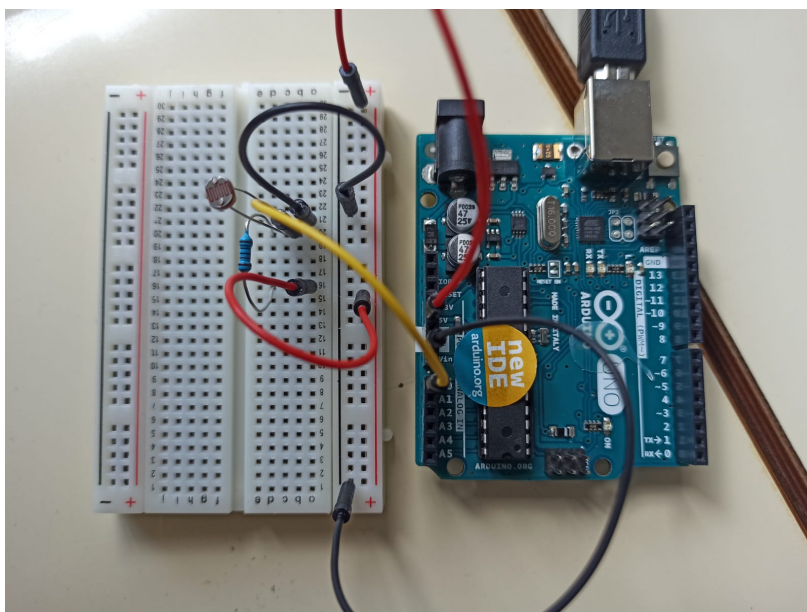
Por defecto nos sale una pantalla tipo, con 10 canales (columnas) que son los diferentes datos que va a leer y 15 filas. Esto se puede modificar en la pestaña **configuración**.

<https://www.youtube.com/watch?v=YTZxYgRMDal>

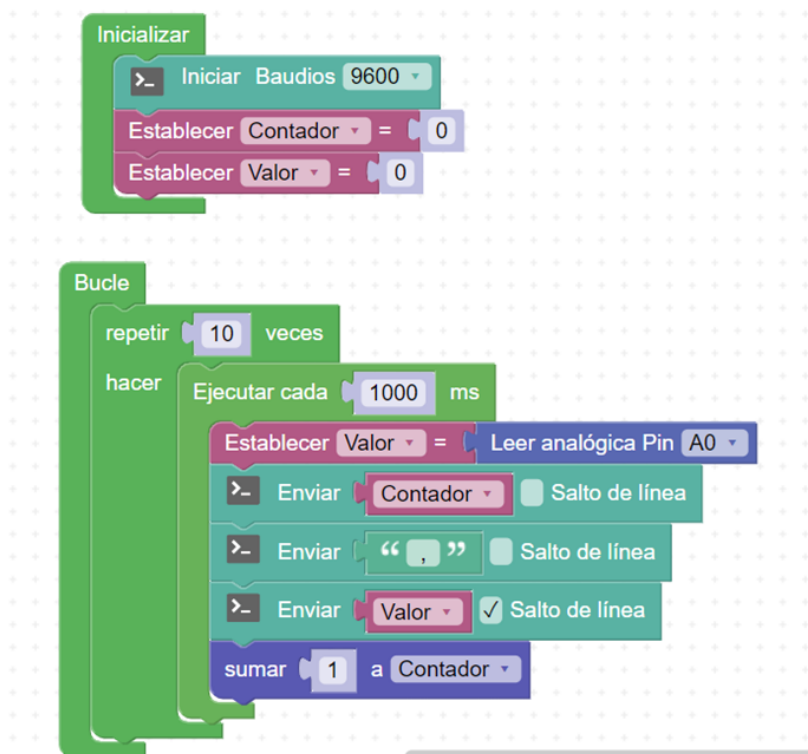


2.2 COMO HACER QUE ARDUINO MUESTRE DATOS EN EL MONITOR SERIE (Y POR LO TANTO EN NUESTRO EXCEL)

He creado una práctica que mide la resistencia de una fotorresistencia.

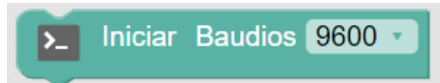


He creado el código con ArduinoBlocks

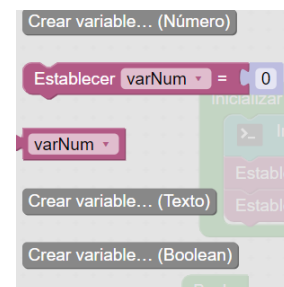


En inicializar (void_setup):

Hemos iniciado el puerto serie:



He creado dos variables, una que será el contador que en cada bucle aumentara una unidad (contador) y otra variable que será lo que lea el sensor (Valor) en VARIIVLES, CREAR VARIABLE. Y las he inicializado a 0.



En bucle:

He creado un bucle que se repite 10 veces y que se ejecuta cada segundo (1000 ms) donde ahora mi variable Valor tenga el valor de la lectura de la fotorresistencia que la tengo conectada a la salida analógica 0 (A0).

Y lo mando imprimir por el monitor serie.

He creado una variable llamada contador, que es el tiempo y que en cada bucle suma una unidad. Lo que aparece en Excel en la primera columna será el contador y en la segunda columna el valor de la fotorresistencia.

A la hora de que lo lea Excel, el código tiene que estar separado por comas, cada uno de los elementos separados por comas será una columna de la hoja, cuando hay salto de línea empieza a escribir en una línea nueva.



Importante: para que salga bien en Excel ,es que cada variable que quieres incluir en una columna tiene que estar separado por comas, en el ejemplo anterior te sale en una columna CONTADOR y en otra columna VALOR.

	A	B	C	D	E	F	G	
3	Datos actuales							
4	TIME	CH1	CH2	CH3	CH4	CH5	CH6	C
5	12:39:22,97	14	94					
6	Información histórica							
7	TIME	CH1	CH2	CH3	CH4	CH5	CH6	C
8	12:39:22,97	14	94					
9	12:39:21,96	13	93					
10	12:39:20,97	12	93					
11	12:39:19,97	11	94					
12	12:39:18,97	10	94					
13	12:39:17,97	9	94					
14	12:39:16,97	8	93					
15	12:39:15,96	7	93					
16	12:39:14,96	6	94					
17	12:39:13,95	5	94					
18	12:39:12,95	4	93					
19	12:39:11,96	3	93					
20	12:39:10,95	2	94					
21	12:39:09,94	1	94					
22	12:39:08,94	0	93					

3 CONCEPTOS BÁSICOS SOBRE SENSORES

El entorno Arduino es una plataforma electrónica formada por un conjunto de **hardware y software libre**, de código abierto.

En la parte de hardware, tenemos principalmente una **placa**, a la que pueden darse una serie de instrucciones a través de un lenguaje de programación propio.

ocasión cada línea numérica está conectada de modo horizontal, por ejemplo: f1, g1, h1, i1 y j1 están conectados entre sí, pero no conectados con la línea 2, ni con a1, b1, c1, d1 y e1. No obstante, podemos poner un cable entre dos puntos cualquiera consiguiendo así la conexión

3.1 SENSOR DE HUMEDAD DEL SUELO

3.1.1 Descripción

Este sensor de humedad del suelo es un dispositivo sencillo pero efectivo diseñado para medir el contenido de humedad en el suelo. Su funcionamiento es simple: cuando el suelo está seco, el valor analógico que emite el sensor disminuye; por el contrario, cuando hay suficiente humedad, el valor aumenta.

3.1.2 Aplicaciones

Una de las aplicaciones más útiles de este sensor es en sistemas de riego automático. Al integrarlo con un controlador Arduino, puedes crear un sistema inteligente que detecte cuándo tus plantas necesitan agua, evitando que se marchiten cuando no estás en casa.

3.1.3 Características

- ☐ **Fácil de usar:** Ideal para proyectos de jardinería y agricultura urbana.
- ☐ **Compatible con Arduino:** Se puede conectar fácilmente a cualquier placa Arduino para programar respuestas automáticas.
- ☐ **Detección precisa:** Permite obtener lecturas confiables sobre la humedad del suelo.

3.1.4 Ventajas

- ☐ **Cuidado automatizado:** Mantiene tus plantas saludables al regarlas automáticamente cuando lo necesitan.
- ☐ **Ahorro de agua:** Optimiza el uso del agua al regar solo cuando es necesario.

- **Interactividad:** Permite a los usuarios interactuar con su jardín de manera más eficiente.

En resumen, el módulo del sensor de humedad del suelo es una excelente opción si necesitas monitorear la humedad de tu suelo en proyectos de jardinería, convirtiendo tu espacio verde en un lugar más inteligente y cómodo.

3.1.5 Características técnicas del sensor

1. Voltaje de la fuente de alimentación: 3,3 V o 5 V
2. Corriente de trabajo: ≤ 20 mA
3. Voltaje de salida: 0-2,3 V (cuando el sensor está totalmente sumergido en
4. agua, el voltaje será 2.3V)
5. Cuanto mayor sea la humedad, mayor será el voltaje de salida.
6. Tipo de sensor: salida analógica
7. Interfaz: Pin1-senal, Pin2-GND, Pin3-VCC

3.2 SENSOR DHT11(HUMEDAD Y TEMPERATURA)

3.2.1 Descripción

El sensor DHT11 es un sensor pequeño y sencillo de utilizar. Nos permite obtener información tanto de la temperatura cómo de la humedad del aire que lo rodea utilizando un único pin digital.

3.2.2 Características técnicas del sensor

1. Voltaje de la fuente de alimentación: 3,3 V o 5 V
2. Corriente de trabajo: ≤ 20 mA
3. Señal de salida: Digital
4. Rango de temperatura: de 0 °C a 50 °C
5. Rango de Humedad: de 20% RH a 90% RH

3.3 SENSOR DE NIVEL DEL AGUA

3.3.1 Descripción

El sensor de nivel de agua está diseñado para detectar el agua, puede ser ampliamente utilizado para detectar la precipitación, el nivel de agua en una cavidad, o incluso la fuga de líquido.

El sensor de nivel de agua funciona mediante una serie de trazos de cobre conectados a tierra intercalados con trazos de detección. Estas trazas funcionan como una resistencia variable. La presencia de agua en el sensor provoca un cortocircuito entre estas trazas, lo que permite detectar su presencia. Convierte el nivel de agua en una salida analógica que puede ser procesada directamente por un microcontrolador. El valor de la señal depende del nivel de inmersión del sensor.

3.3.2 Características técnicas del sensor

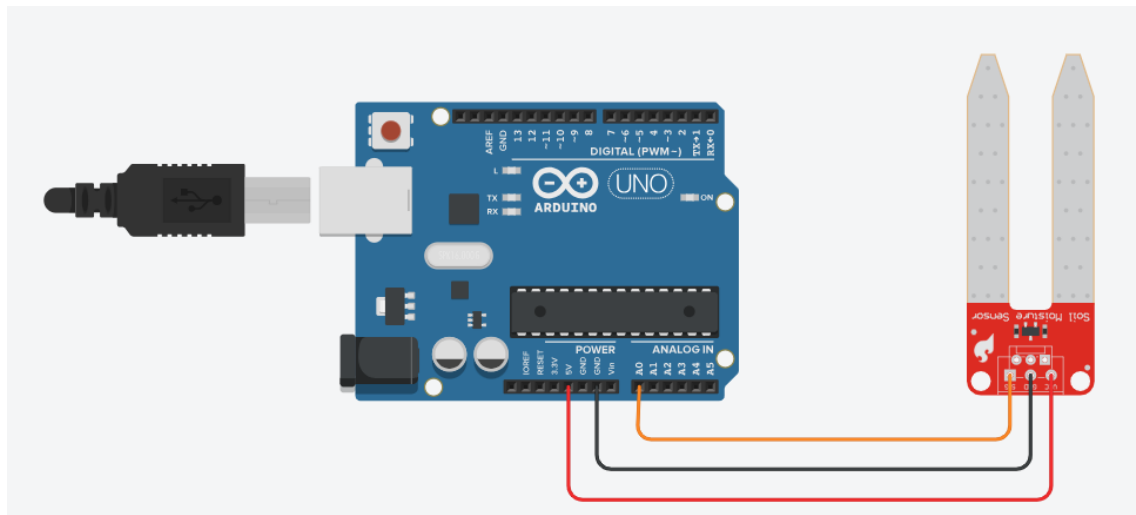
1. Voltaje de la fuente de alimentación: 3,3 V o 5 V
2. Corriente de trabajo: ≤ 20 mA
3. Señal de salida: Analógica.
4. Temperatura de funcionamiento: de 10 °C a 30 °C
5. Rango de funcionamiento: de 0% a 90%

PARTE II. Proyectos



4 HUERTO INTELIGENTE NIVEL INICIACIÓN

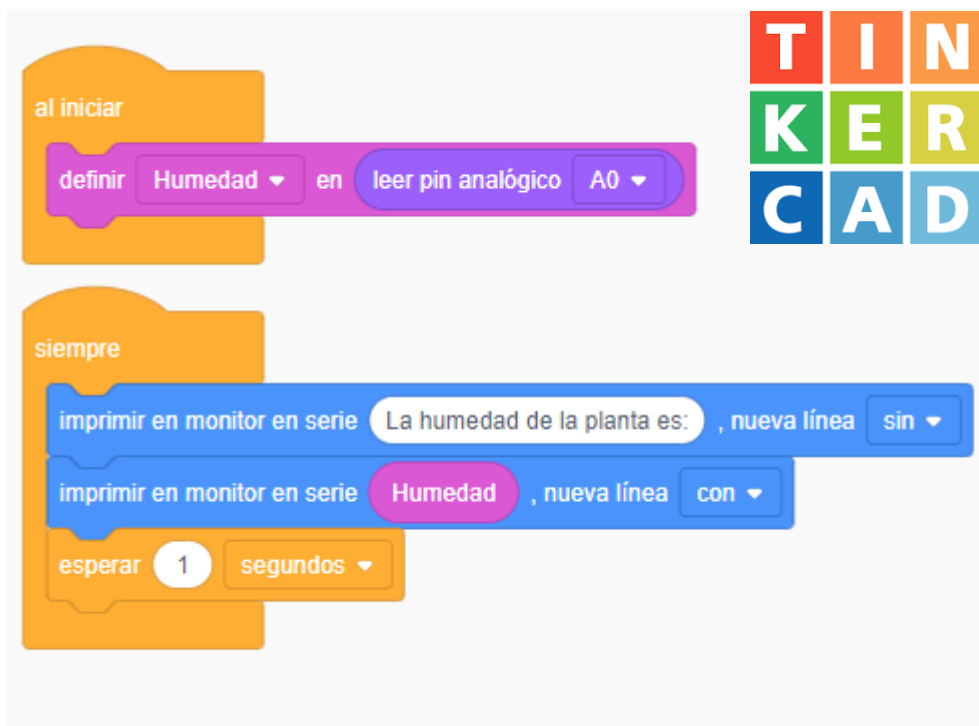
4.1 DIAGRAMA DE CONEXIÓN



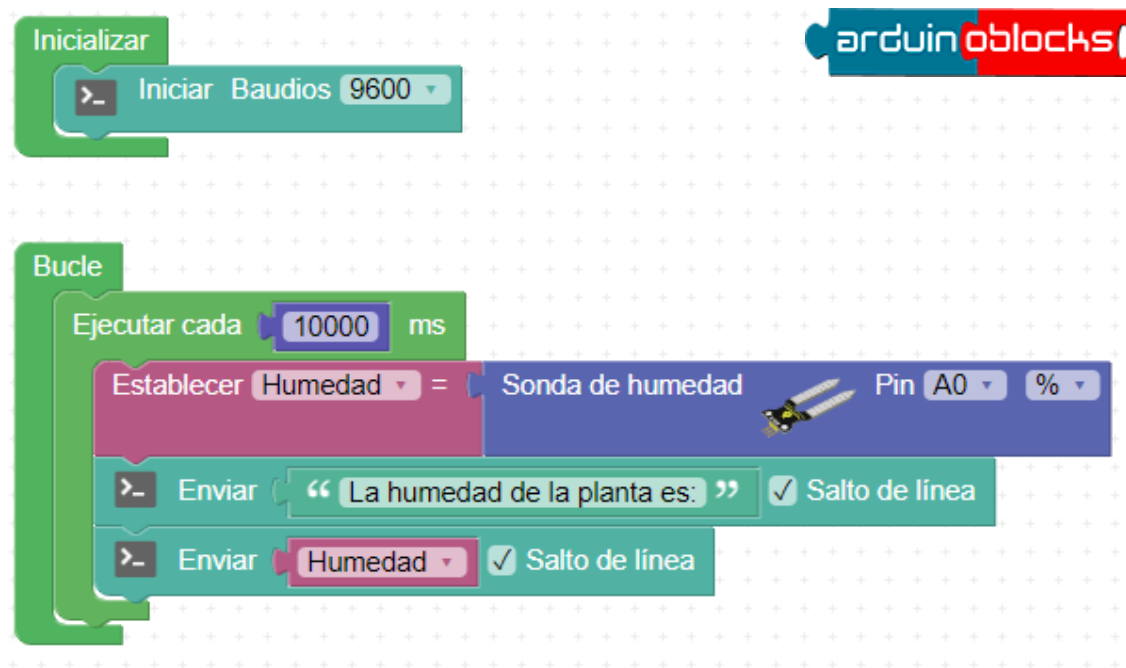
Conecte el pin S (Señal) del sensor al puerto analógico A0 de la placa Arduino, conecte el pin negativo al puerto GND, el pin positivo al puerto de 5V.

4.2 CÓDIGOS FUENTE

Código fuente de Tinkercad



Código Fuente de ArduinoBlocks



Código Fuente en C++ Arduino IDE

```
// C++ code
//
int Humedad = 0; // Declaramos la variable humedad

void setup()
{
  pinMode(A0, INPUT); // Activar el pin analógico 0
  Serial.begin(9600);

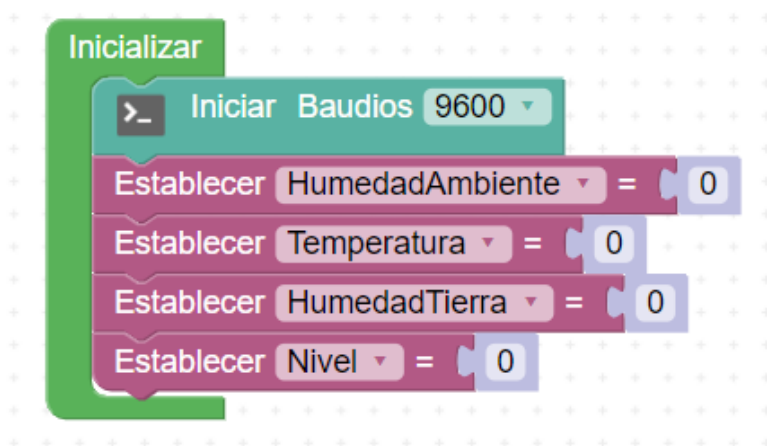
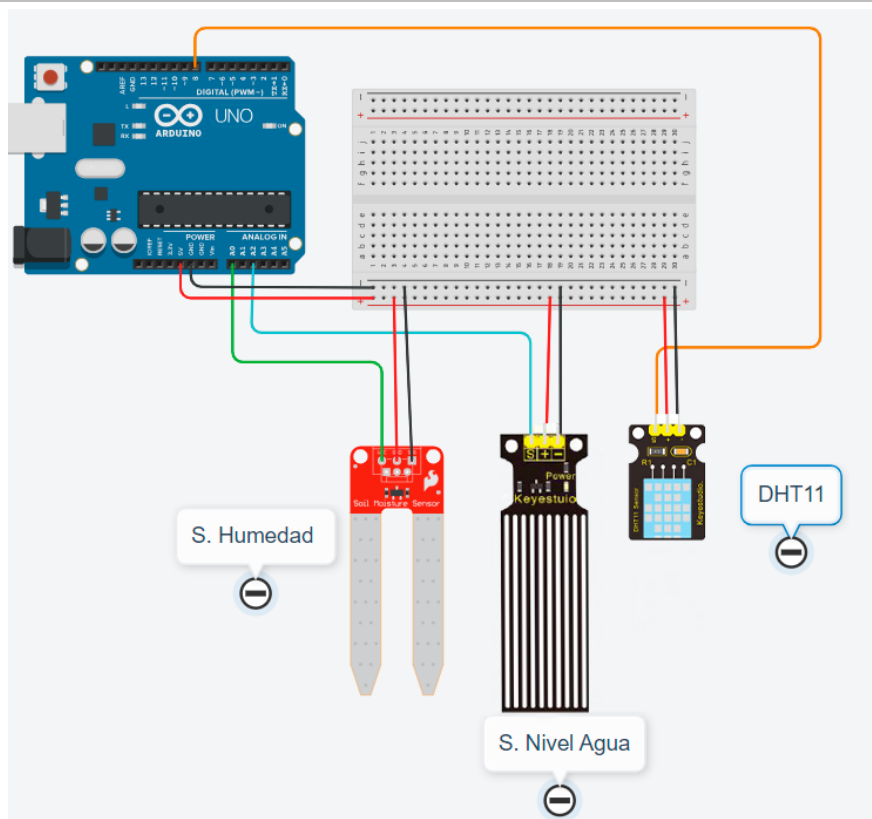
  Humedad = analogRead(A0); // El valor de la variable humedad es el
  valor que da el sensor
}

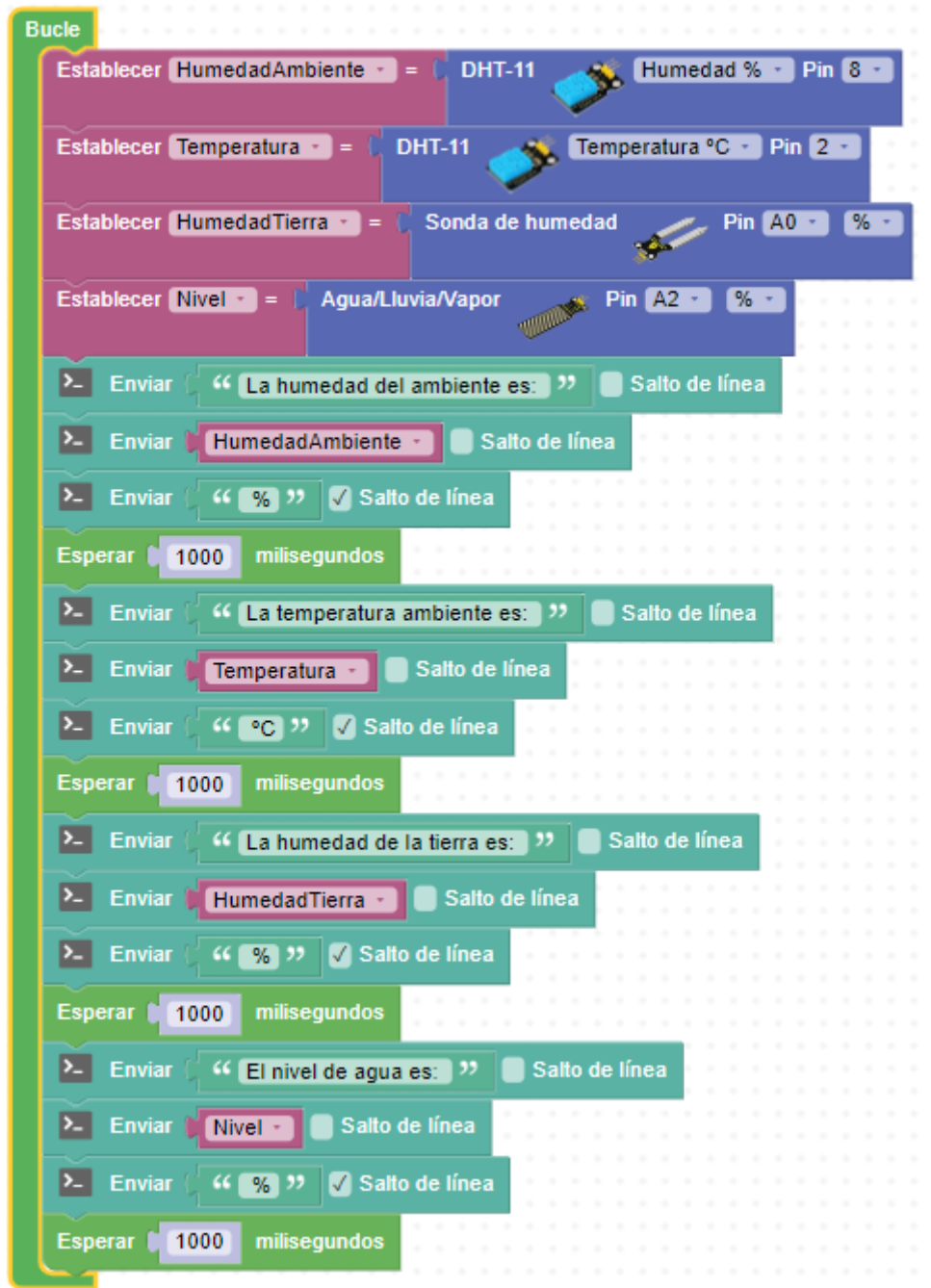
void loop()
{
  Serial.print("La humedad de la planta es:"); // escribir frase
  Serial.println(Humedad); // mostrar en el monitor de serie el valor de
  humedad
  delay(1000); // Wait for 1000 millisecond(s)
}
```



5 HUERTO INTELIGENTE NIVEL MEDIO

5.1 CONEXIÓN DE SENSORES





```

double HumedadAmbiente;
double Temperatura;
double HumedadTierra;
double Nivel;
DHT dht8(8,DHT11);
DHT dht2(2,DHT11);

void setup()
{
    pinMode(8, INPUT);
    pinMode(2, INPUT);
    pinMode(A0, INPUT);
    pinMode(A2, INPUT);

    Serial.begin(9600);
    Serial.flush();
    while(Serial.available()>0)Serial.read();

    dht8.begin();
    dht2.begin();

    HumedadAmbiente = 0;
    Temperatura = 0;
    HumedadTierra = 0;
    Nivel = 0;
}

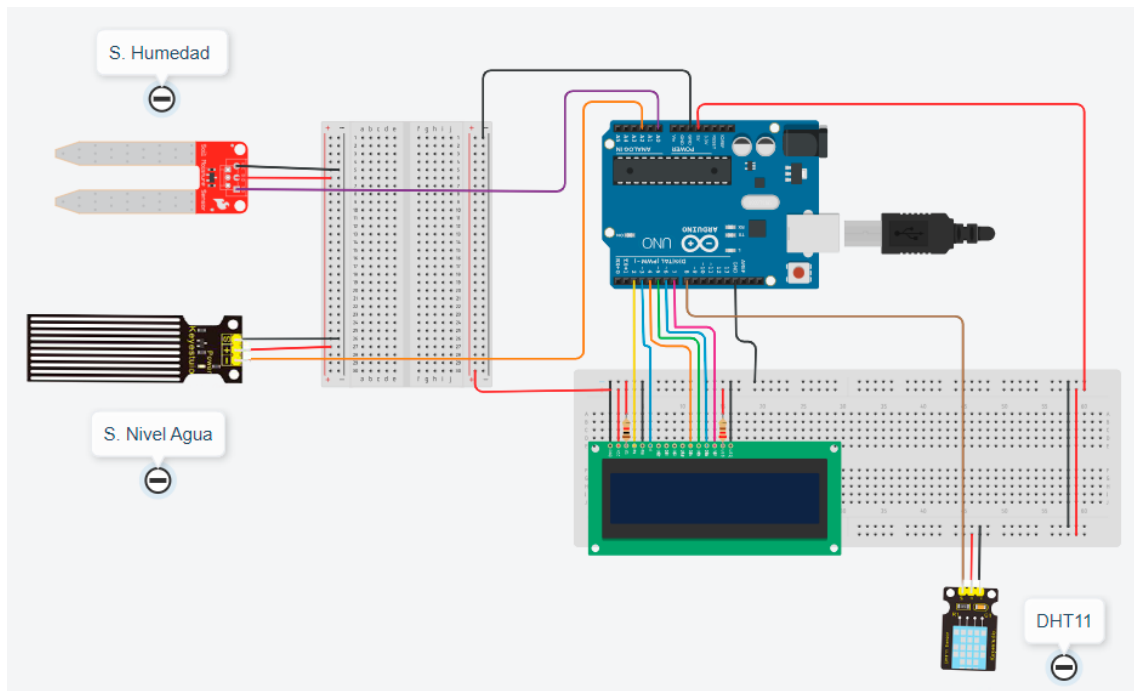
void loop()
{
    HumedadAmbiente = dht8.readHumidity();
    Temperatura = dht2.readTemperature();
    HumedadTierra = map(analogRead(A0),0,1023,0,100);
    Nivel = map(analogRead(A2),0,1023,0,100);
    Serial.print(String("La humedad del ambiente es: "));
    Serial.print(HumedadAmbiente);
    Serial.println(String(" %"));
    delay(1000);
    Serial.print(String("La temperatura ambiente es: "));
    Serial.print(Temperatura);
    Serial.println(String(" °C"));
    delay(1000);
    Serial.print(String("La humedad de la tierra es: "));
    Serial.print(HumedadTierra);
    Serial.println(String(" %"));
    delay(1000);
    Serial.print(String("El nivel de agua es: "));
    Serial.print(Nivel);
    Serial.println(String(" %"));
    delay(1000);
}

```

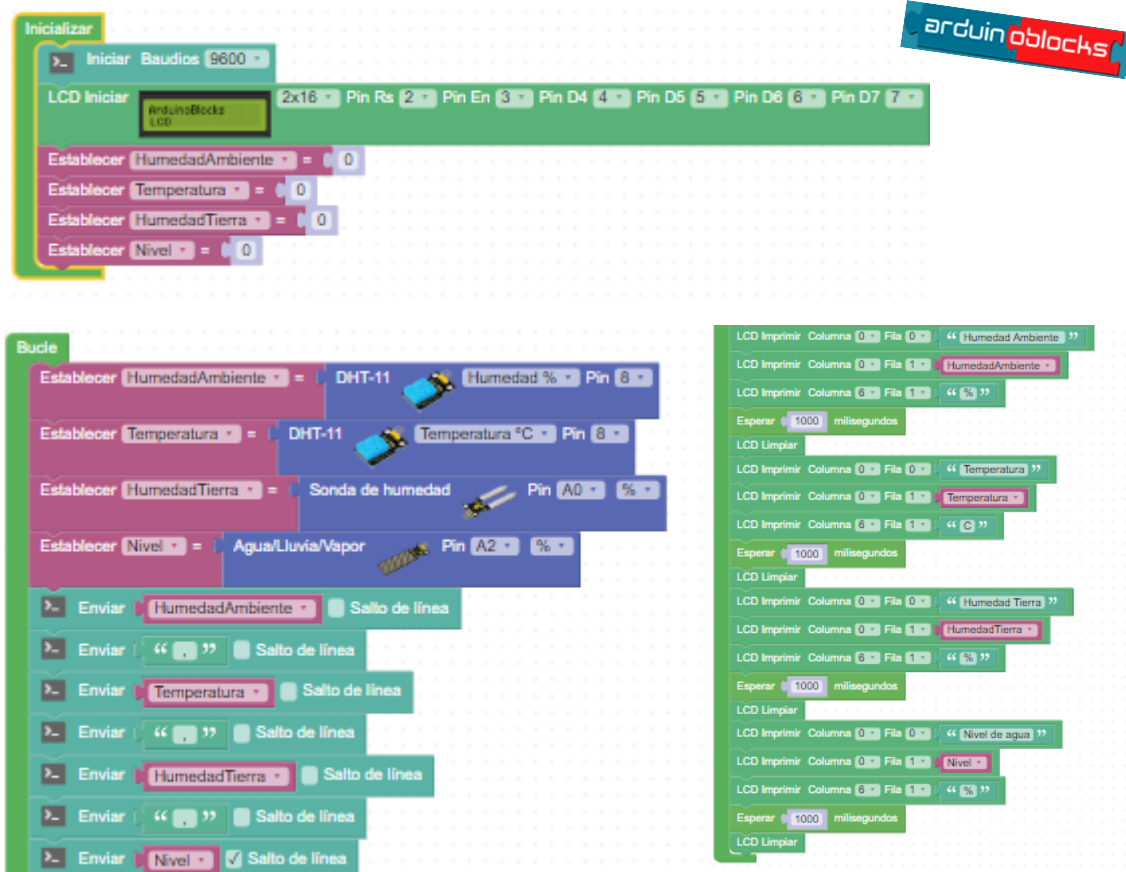


6 HUERTO INTELIGENTE NIVEL AVANZADO

6.1 CONEXIÓN DE SENSORES



Estas prácticas no se pueden simular con Tinkercad, ya que, excepto el sensor de humedad de la tierra, los demás sensores no existen.





Arduino IDE

```
include <LiquidCrystal.h>
#include "ABlocks_DHT.h"

double HumedadAmbiente;
double Temperatura;
double HumedadTierra;
double Nivel;
LiquidCrystal lcd(2,3,4,5,6,7);

DHT dht8(8,DHT11);

void setup()
{
  pinMode(8, INPUT);
  pinMode(A0, INPUT);
  pinMode(A2, INPUT);

  Serial.begin(9600);
  Serial.flush();

  while(Serial.available()>0)Serial.r
    ead();

  lcd.begin(16, 2);

  dht8.begin();

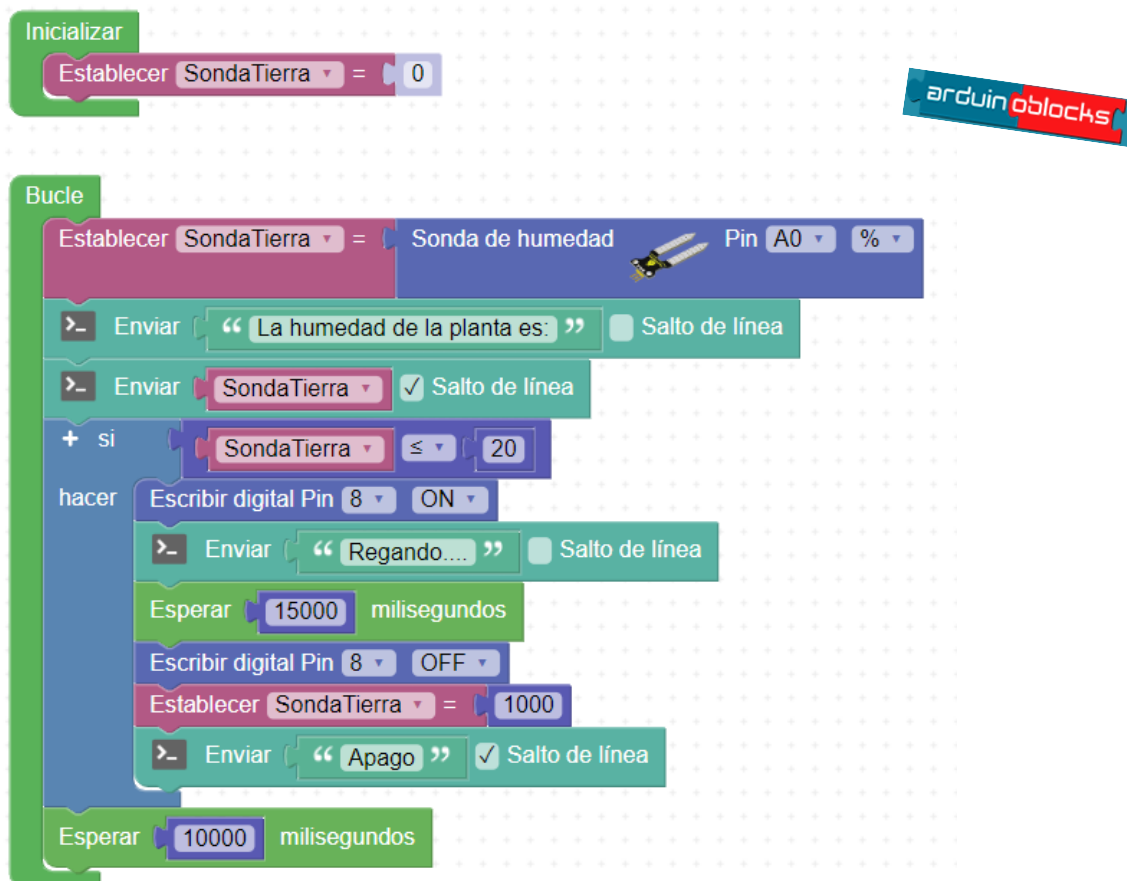
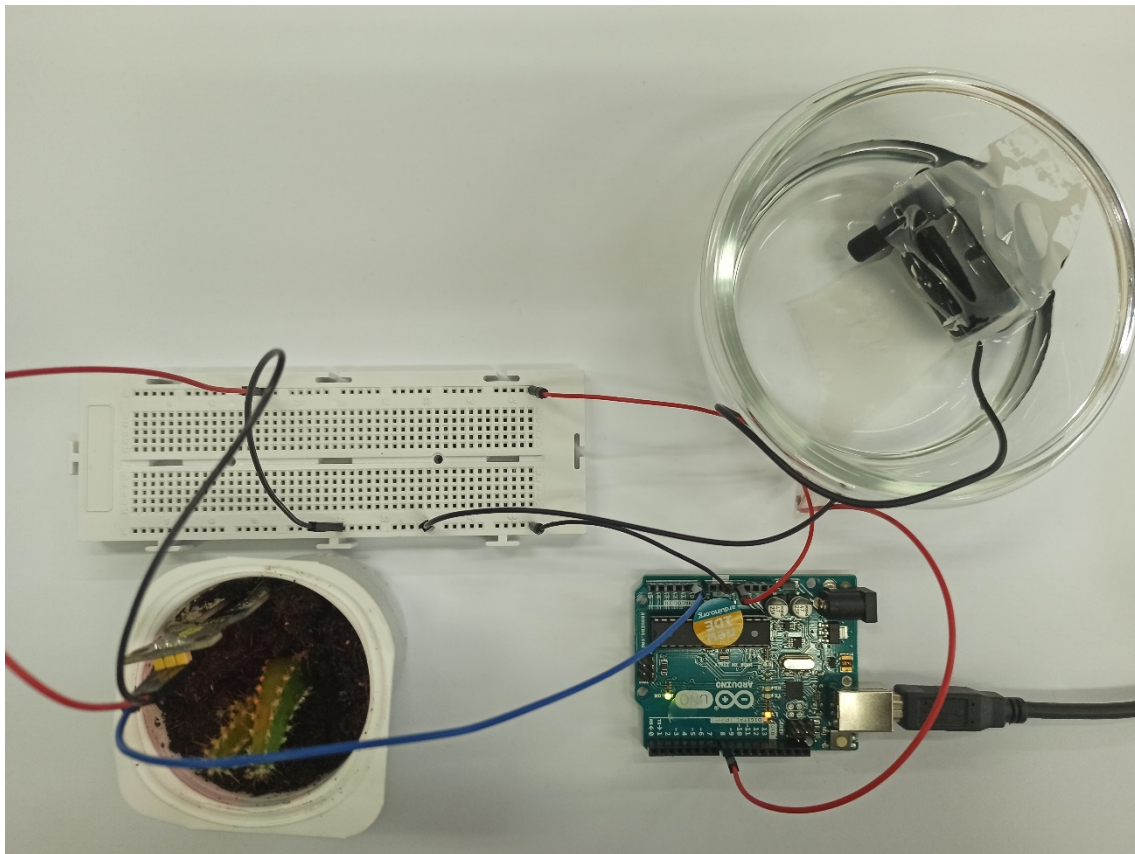
  HumedadAmbiente = 0;
  Temperatura = 0;
  HumedadTierra = 0;
  Nivel = 0;
}

void loop()
{
  HumedadAmbiente =
    dht8.readHumidity();
  Temperatura =
    dht8.readTemperature();
  HumedadTierra =
    map(analogRead(A0),0,1023,0,100);
    Nivel =
    map(analogRead(A2),0,1023,0,100);
  lcd.print(Temperatura);
  lcd.setCursor(6, 1);
  lcd.print(String("C"));
  delay(1000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(String("Humedad
    Tierra"));

  Serial.print(HumedadAmbiente);
  Serial.print(String(" , "));
  Serial.print(Temperatura);
  Serial.print(String(" , "));
  Serial.print(HumedadTierra);
  Serial.print(String(" , "));
  Serial.println(Nivel);
  lcd.setCursor(0, 0);
  lcd.print(String("Humedad Ambiente
    "));
  lcd.setCursor(0, 1);
  lcd.print(HumedadAmbiente);
  lcd.setCursor(6, 1);
  lcd.print(String("%"));
  delay(1000);
  lcd.clear();
  lcd.setCursor(0, 0);

  lcd.print(String("Temperatura"));
  lcd.setCursor(0, 1);
  lcd.setCursor(0, 1);
  lcd.print(HumedadTierra);
  lcd.setCursor(6, 1);
  lcd.print(String("%"));
  delay(1000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(String("Nivel de
    agua"));
  lcd.setCursor(0, 1);
  lcd.print(Nivel);
  lcd.setCursor(6, 1);
  lcd.print(String("%"));
  delay(1000);
  lcd.clear();
}
```

6.2 EJEMPLO DE SENSOR DE HUMEDAD DE A TIERRA Y BOMBA DE AGUA



```

double SondaTierra;

void setup()
{
    pinMode(A0, INPUT);
    pinMode(8, OUTPUT);

    Serial.begin(9600);
    Serial.flush();
    while(Serial.available()>0)Serial.read();

    SondaTierra = 0;
}

void loop()
{
    SondaTierra = map(analogRead(A0),0,1023,0,100);
    Serial.print(String("La humedad de la planta es:"));
    Serial.println(SondaTierra);
    if ((SondaTierra <= 20)) {
        digitalWrite(8, HIGH);
        Serial.print(String("Regando...."));
        delay(15000);
        digitalWrite(8, LOW);
        SondaTierra = 1000;
        Serial.println(String("Apago"));
    }

    delay(10000);
}

```

