Laborator 12. EcoFarm

Proiect realizat și documentat de:

- Ivan Sara Ioana, clasa a X-a D, specializarea matematică-informatică
- Voicu Natașa Elena, clasa a XI-a D, specializarea matematică-informatică

Cuprins

1.Introducere 2

Descrierea proiectului 2

Obiectivele proiectului 2

2.Componente necesare 3

3. Mod de lucru 11

a) Parametrii de lucru 11

Parametrii și specificații ale plăcii de dezvoltare ESP32 11

Asamblarea scheletului 11

Imagini din timpul asamblării 13

Cablare (Wiring) 14

4.Funcțiile componentelor 16

a) Sistem de control al luminii 16

1) Aprinderea LED-ului la apăsarea butonului 16

2) Aprinderea LED-ului la detectarea întunericului 18

Senzor de mișcare cu alarmă 20

Deschiderea cutiei prin servo motor 23

Detectarea temperaturii și umidității + afișare 26

Sistem de irigație inteligent 30

6.Concluzii 33

7.Referințe bibliografice 34

1.Introducere

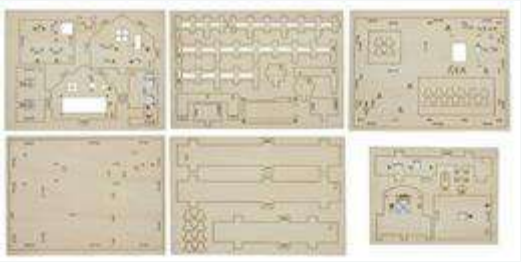
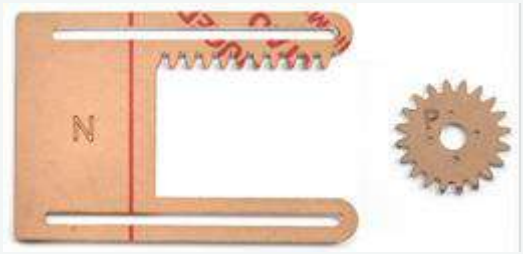
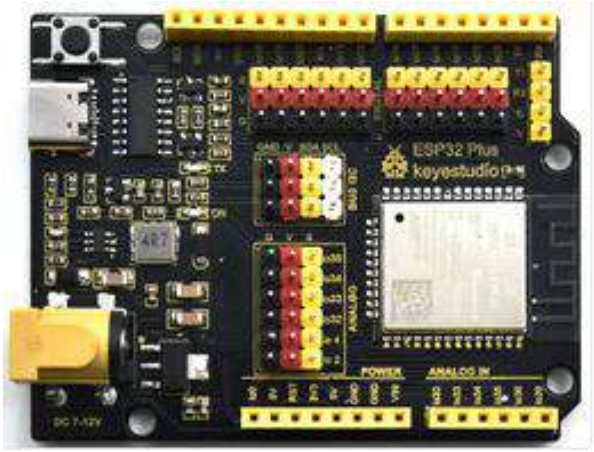
Descrierea proiectului

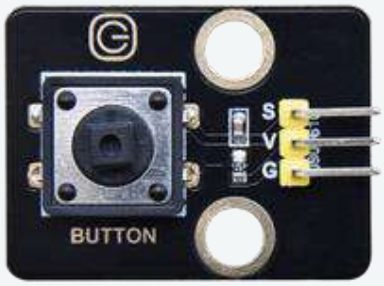
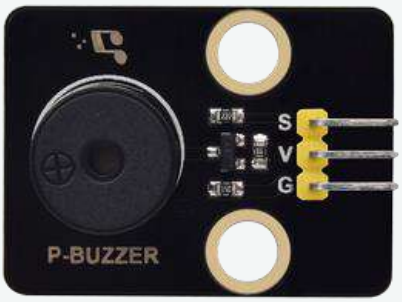
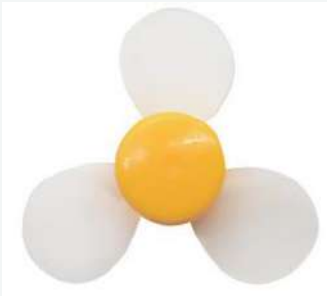
Bazat pe ESP32 Internet of Things, acest kit Smart Farm integrează mai mulți senzori, realizând manipularea automată și wireless, precum și managementul intelectual. Este caracterizat ca fiind ***cuprinzător***, datorită gamei variate de senzori pe care îi conține, ***inovator***, cultivă inovația tehnologică și capacitatea de rezolvare a problemelor din agricultură și ***interactiv***, dispunând de o multitudine de cazuri și practici experimentale.

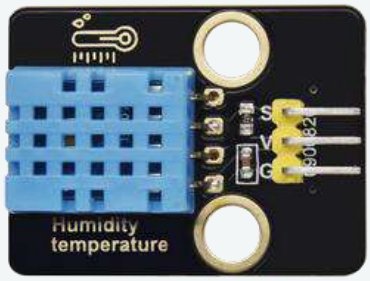
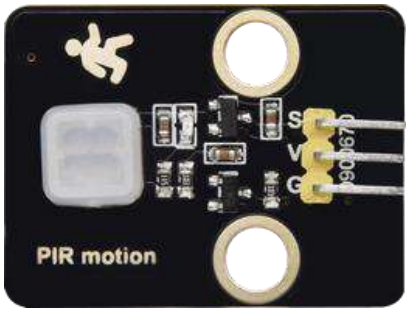
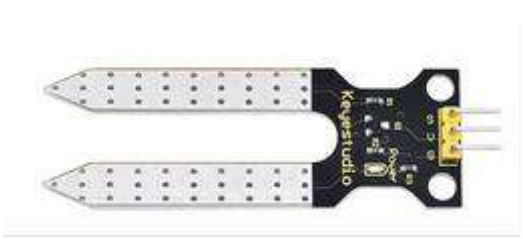
Obiectivele proiectului

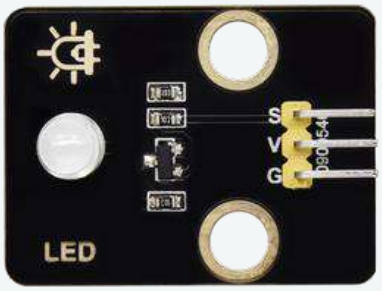
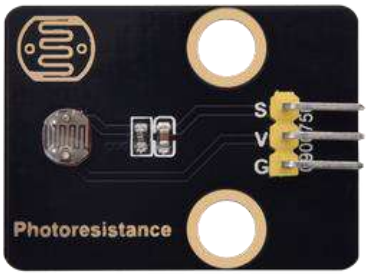
- stăpânirea și dobândirea cunoștințelor diverșilor senzori și a comunicării fără fir (wireless);
- înțelegerea în profunzime a funcționării fiecărui senzor și cum colectează și procesează datele;
- stimulează entuziasmul de învățare prin punerea în practică a elementelor de senzoristică în timpul realizării unui design practic și ușor de manevrat.

2.Componente necesare

Nr.	Imagine	Denumire / Descriere	Cantitate
1		Placă din lemn de tijol	6 diferite
2		Placă din lemn de tijol cu modele	1
3		Placă acrilică (o placă învelită într-un plastic transparent, cu rezistență sporită și ușor de lucrat cu adezivi sau diferiți solvenți)	1
4		Placă de dezvoltare ESP32 PLUS(placă de circuite imprimată cu un microcontroler sau un microprocesor montat pe ea, cu diferiți pini metalici)	1

Nr.	Imagine	Denumire / Descriere	Cantitate
5		Modul Buton (un buton efectiv conectat la placa de dezvoltare)	1
6		Capac buton (atașat modulului buton pentru ca utilizatorul să poată apasa pe el)	1
7		Buzzer pasiv (un device care generează un sunet atunci când un curent electric trece prin el)	1
8		130 Motor (un motor mic care, dacă primește un voltaj cuprins între 3V și 6V, funcționează învârtind elementul destinat din extremitatea acestuia)	1
9		Ventilator	1

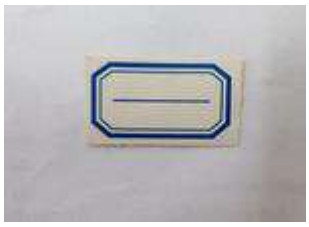
Nr.	Imagine	Denumire / Descriere	Cantitate
10		Senzor de abur (un senzor analog; atunci când umiditatea de pe suprafața acesuia crește, tensiunea electrică care iese va crește)	1
11		Senzor de temperatură și umiditate DHT11 (un device care transformă temperatura și umiditatea în impulsuri electrice pentru a putea măsura valorile acestora)	1
12		Senzor de mișcare PIR (utilizat în detectoarele de mișcare, cum ar fi dispozitivele de iluminare cu declanșare automată și sistemele de protecție care măsoară dispozitivele care emit lumină infraroșie în câmpul lor vizual)	1
13		Modul releu 5V (un comutator automat care este utilizat în mod obișnuit într-un circuit de control automat și pentru a controla un curent ridicat folosind un semnal de curent scăzut)	1
14		Senzor de umiditate a solului (măsoară și aproximează cantitatea apei din sol)	1

Nr.	Imagine	Denumire / Descriere	Cantitate
15		Senzor de nivel al apei (măsoară nivelul apei dintr-un recipient)	1
16		Modulul cu ultrasunete SR01 V3 (detectează obiecte folosind unde ultrasonice, care sunt transmise și recepționate cu un singur disc piezoelectric)	1
17		Modulul LED alb (aprinde/ stinge un LED alb la conectarea la ARDUINO)	1
18		Fotorezistor (indică prezența/absența luminii și/sau măsoară intensitatea luminii)	1
19		Servo 9G 180° (un motor care se poate roti la un anumit unghi în funcție de semnalul primit)	1

Nr.	Imagine	Denumire / Descriere	Cantitate
20		Ecran LCD I2C 1602 (un tip de afișaj cu ecran plat care utilizează cristale lichide în forma sa principală de funcționare)	1
21		Suport pentru baterii AA cu 6 sloturi	1
22		Pompă de apă DC 3V cu sârmă de 15 cm pentru panou (transportă apă dintr-un loc în altul, utilizând un fir de sârmă cu pini/ conectori la ambele capete)	1
23		Cablu USB TIP-C	1

Nr.	Imagine	Denumire / Descriere	Cantitate
24		Șurub autofiletant M1.4*6MM	4
25		Contrapiuliță M3	4
26		Șurub cu cap rotund M4*8MM	18
27		Șurub cu cap rotund M3*6MM	8
28		Șurub cu cap rotund M3*10MM	16
29		Șurub cu cap rotund M2*12MM	2
30		Piuliță M4	18
31		Piulita M3	12

Nr.	Imagine	Denumire / Descriere	Cantitate
32		Piuliță M2	2
33		Stâlp din cupru cu trecere dublă M3*10MM	4
34		Fire DuPont 3P FF de 15 cm (conectate, 26 AWG, negru-roșu-galben) (firele DuPont sunt folosite pentru a conecta hardware cum ar fi senzori, plăci Arduino și breadboard-uri)	4
35		Fire DuPont 3P FF de 20 cm (conectate, 26 AWG, negru-roșu-galben)	5
36		Fire DuPont 3P FF de 25 cm (conectate, 26 AWG, negru-roșu-galben)	1
37		Fire DuPont 20cm 4P FF (conectate, 26AWG, negru-roșu-albastru-verde)	1
38		Fire DuPont 4P FF de 20 cm (divizat, 26 AWG, negru-verde-albastru-roșu)	1
39		Fire DuPont 20cm 4P FF (divizat, 26AWG, negru-roșu-albastru-verde)	1
40		Fire DuPont MF 15CM 40P	2
41		Șurubelniță Phillips 3.0*40MM	1

Nr.	Imagine	Denumire / Descriere	Cantitate
42		Șurubelniță cu crestat	1
43		Cutie de plastic (recipiente folosite la pompa de apă unde va fi depozitată apa)	2
44		Conductă pentru pompă de apă (6 cm)	1
45		Panou solar (colectează energie regenerabilă curată sub formă de lumină solară și transformă această lumină în energie electrică care poate fi apoi folosită pentru a furniza energie pentru sarcini electrice)	1
46		Autocolante	1
47		Cheie încrucișată (este utilizată pentru slăbirea și strângerea piulițelor)	1

3. Mod de lucru

a) Parametrii de lucru

Tensiune de lucru: 5V

Tensiune baterie: 9V

Curent maxim de ieșire: ≤1.5A

Consum maxim de energie: ≤10W

Principiul de funcționare WiFi: 2.4G

Protocol Bluetooth: v4.2 Standard complet

Temperatura de lucru: -10°C ~ +65°C

Parametrii și specificații ale plăcii de dezvoltare ESP32

Tensiune: 3.3V-5V

Ieșire curentă: 1,2 A (maximum)

Putere maxima de iesire: 10W

Temperatura de lucru: -10℃~50℃

Dimensiune: 69*54*14.5mm

Greutate: 25,5 g

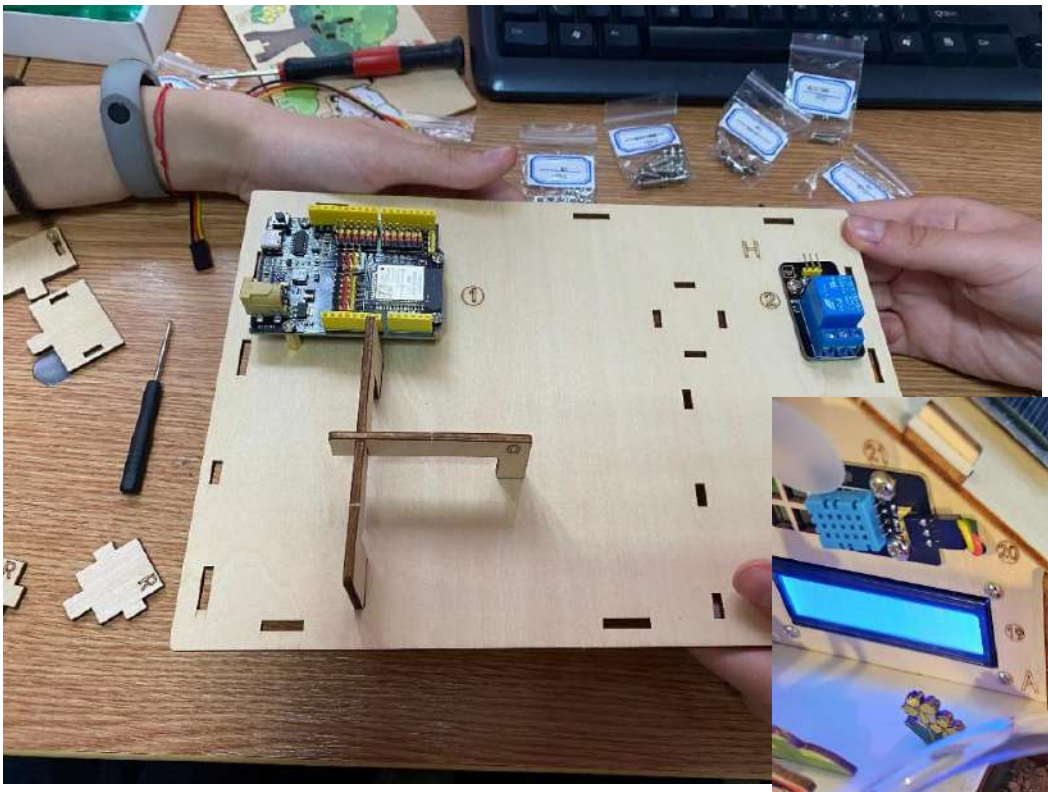
Asamblarea scheletului

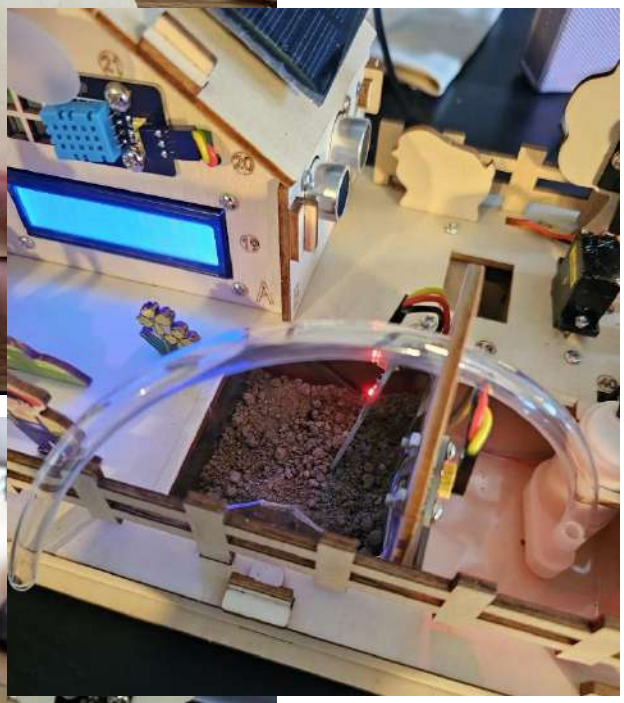
Asamblarea scheletului fermei inteligente se face urmând cu atenție fiecare indicație oferită în tutorialul pus la dispoziție de dezvoltator pe site-ul oficial https://docs.keyestudio.com/projects/KS0567/en/latest/wiki/Arduino/arduino.html#assembling .

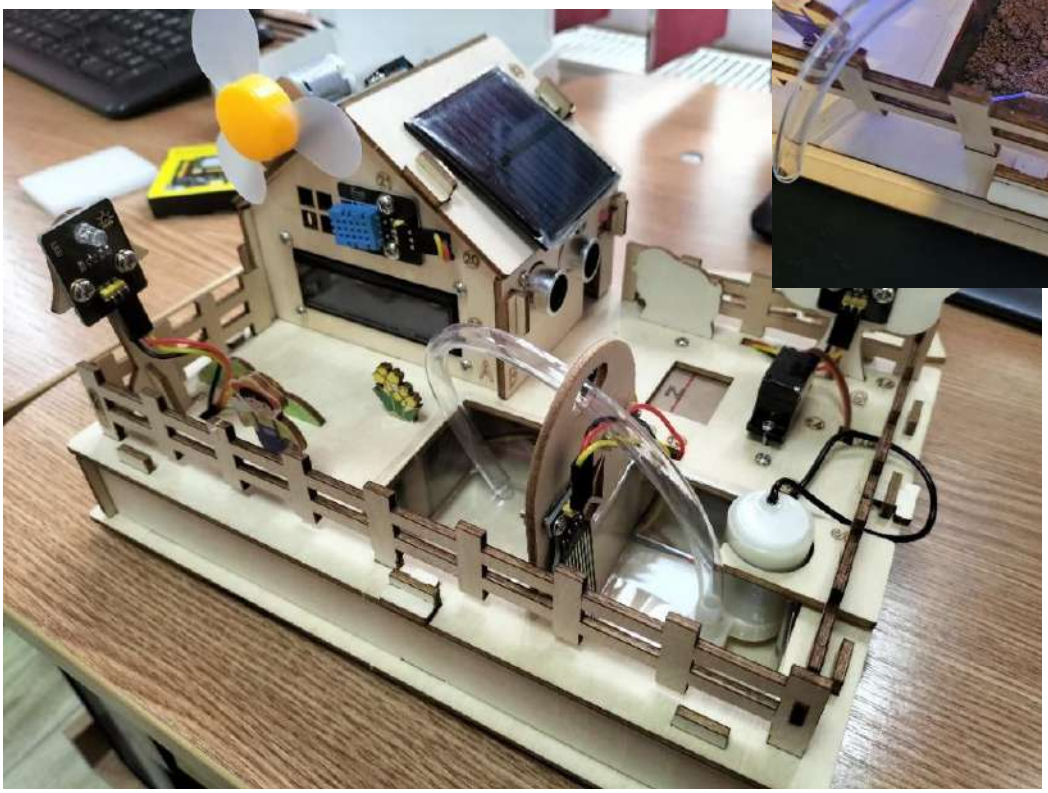
La deschiderea kitului asigurați-vă ca niciun element nu lipsește consultând lista componentelor din interiorul cutiei sau lista atașată mai sus. Elementele, scoase din cutie, vin ambalate individual, astfel:

Evitați amestecarea piulițelor sau a șurubelelor între ele sau expunerea lor pe o suprafață de pe care le puteți pierde ușor, acestea având dimensiuni mici. Recomandare: plasați-le într-un recipient din care le puteți lua ușor și folosiți instrumentele din kit (șurubelnițele, cheia încrucișată) care sunt destinate acestor tipuri de șuruburi.

Imagini din timpul asamblării







Cablare (Wiring)

Elemente necesare:

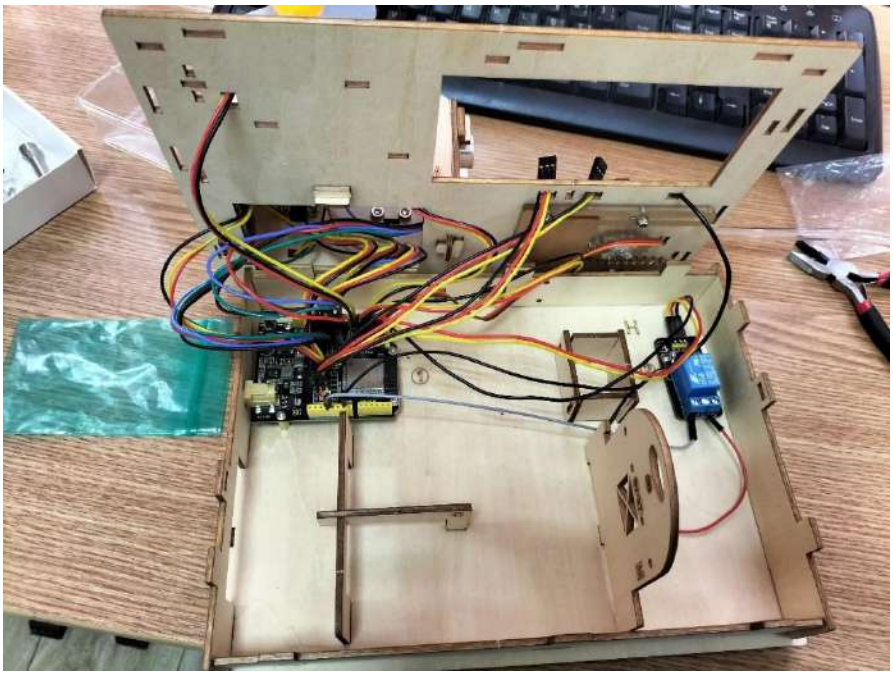
Nr.	Componente	Firele	Pinuri pentru placa ESP32
1	Ventilator	4 pini, **împărțit** negru-roșu-albastru-verde	io18(IN-) | io19(IN+)
2	Senzor de mișcare PIR	3 pini 15 cm	io23
3	Buton	3 pini 15 cm	io5
4	Modulul cu ultrasunete	4 pini, **împărțit** negru-verde-albastru-roșu	D12(TRIG) D13(ECHO)
5	LCD 1602	4 pini, **conectat**	I2C
6	Senzor de temperatură și umiditate	3 pini 20 cm	io17
7	Senzor de abur	3 pini 15 cm	io35
8	Fotorezistor	3 pini 15 cm	io34
9	Servo	–	io26
10	Buzzer	3 pini 20 cm	io16
11	LED	3 pini 20 cm	io27
12	Senzor de nivel al apei	3 pini 25 cm	io33
13	Senzor de umiditate a solului	3 pini 20 cm	io32
14	Pompa de apa	3 pini 20 cm	io25

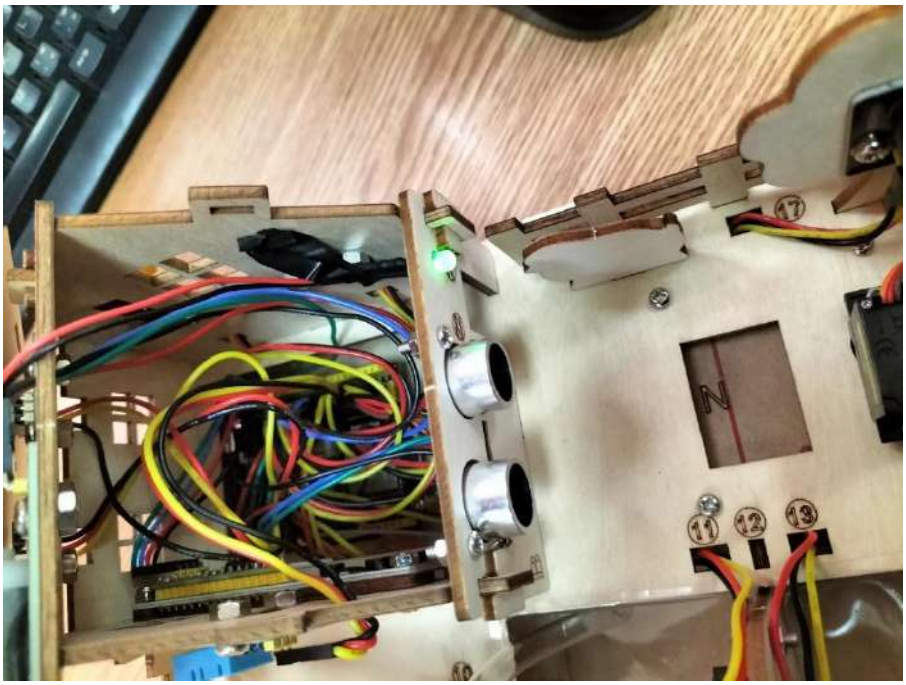
Pentru această parte este esențial să fie executată corect deoarece dă „viață” fermei smart. Conectând fiecare fir atașat senzorilor de placa de dezvoltare în

locurile indicate, senzorii vor primi curentul electric necesar funcționării lor, astfel vor face ce sunt programați să facă.

Pentru cuplarea fiecărui senzor la placă, în tutorial, la secțiunea dedicată acestui tip special de asamblare, este desenată o diagramă de circuit, indicând locul unde trebuie cuplat firul și ce componente sunt necesare pentru realizarea acesteia.

La final, toate cablurile cuplate din interiorul fermei vor arăta în felul următor:





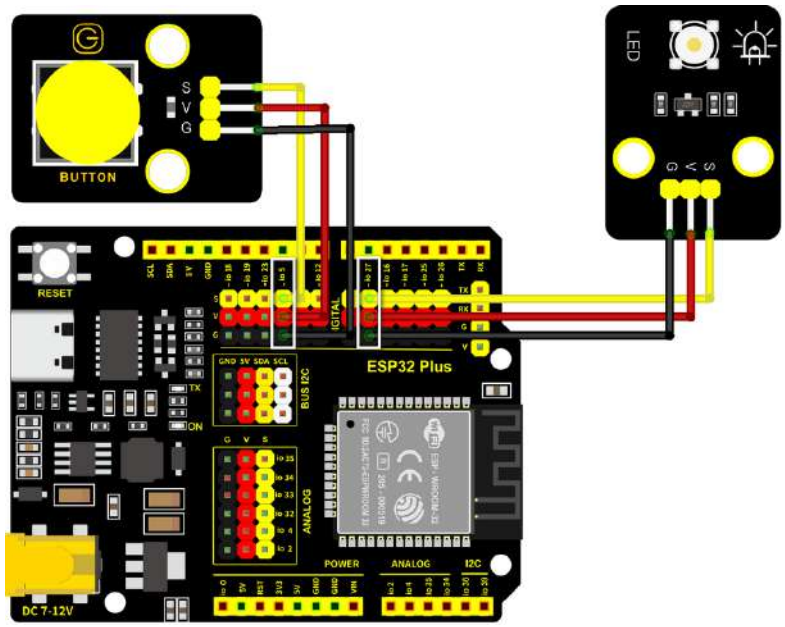
4.Funcțiile componentelor

- Software utilizat: ***Arduino IDE***

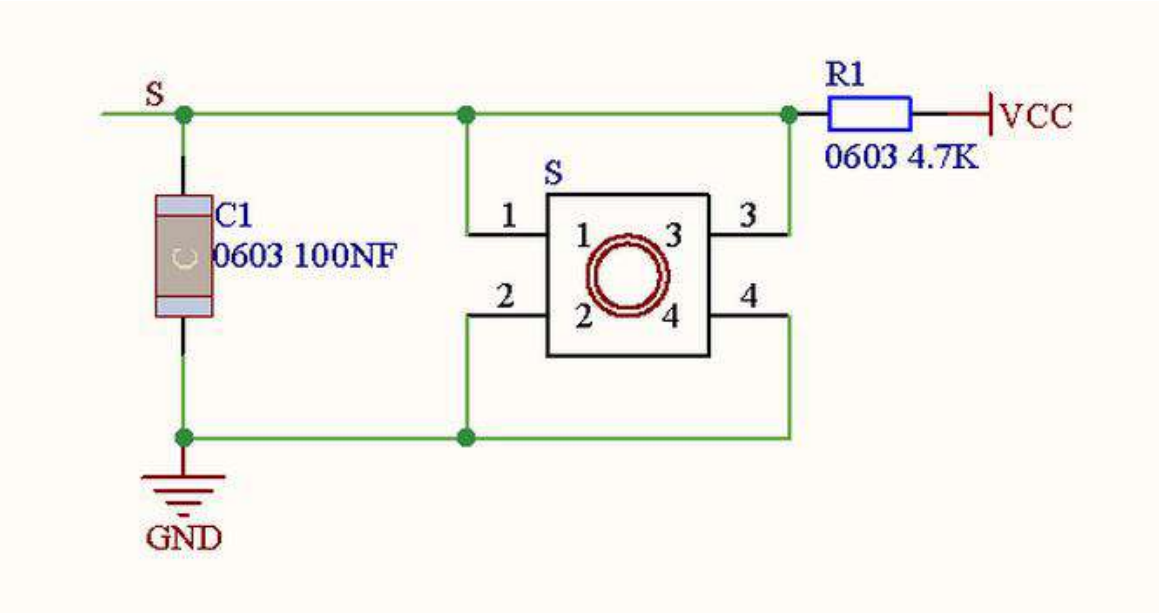
a) Sistem de control al luminii

1) Aprinderea LED-ului la apăsarea butonului

- ***Diagrama circuitului***



- ***Schemă electrică***



- ***Parametrii***

  Tensiune: 3-5 V

  Curent: ≤1.1 mA

  Putere: ≤5.5 mW

- ***Programare***

```
#define led 27
#define ButtonPin 5
int value = 0;

void setup(){
  pinMode(led, OUTPUT);
  Serial.begin(9600); //initialize serial port and set baud rate to 9600
  pinMode(ButtonPin, INPUT);
}

void ledONandOFF(){
  int ReadValue = digitalRead(ButtonPin);  // Read button value

  if (ReadValue == 0) {  // Button pressed
    delay(10);  // Debounce
    if (digitalRead(ButtonPin) == 0) {  // Confirm button press after
debounce
      value = !value;  // Toggle the value
      // If value is true, turn on the LED, otherwise keep it off
      if (value) {
        digitalWrite(led, HIGH);
        Serial.println("LED ON");
      } else {
        digitalWrite(led, LOW);
      }
    }
```

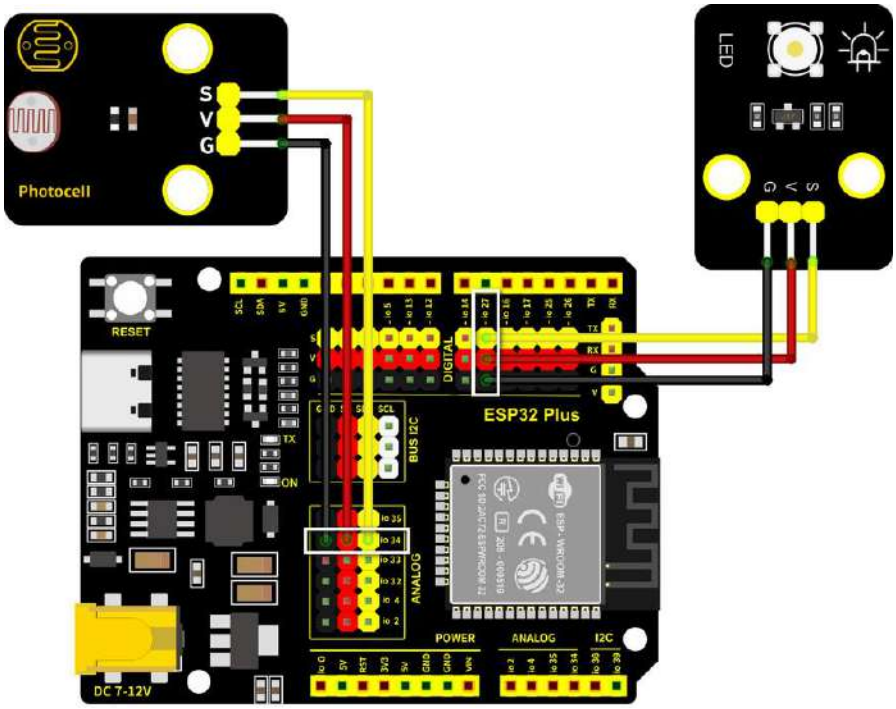
```
      while (digitalRead(ButtonPin) == 0);   // Wait until button is
released
  }
}

void loop(){
      ledONandOFF();
}
```

- ***Rezultat***

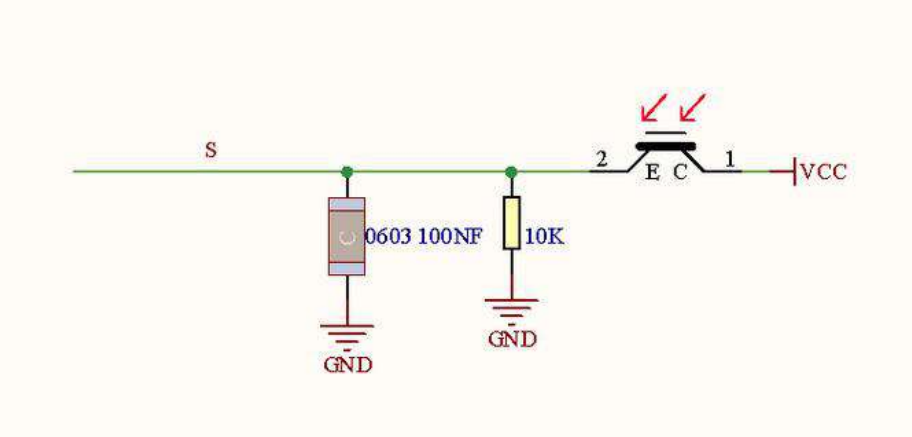
Putem citi starea pinului de intrare digitală programând pentru a detecta dacă butonul este apăsat. Astfel, diverse aplicații interactive pot fi realizate prin intermediul unui modul de buton, cum ar fi pornirea/oprirea unui LED sau ajustarea luminozității afișajului.

2) Aprinderea LED-ului la detectarea întunericului

- ***Diagrama circuitului***



- ***Schemă electrică***



- ***Programare***

```
#define led 27
#define PhotocellPin 34

void setup(){
  pinMode(led, OUTPUT);
  Serial.begin(9600);
  pinMode(PhotocellPin, INPUT);
}
void ledONandOFF(){
   int  ReadValuePhotocell  =  analogRead(PhotocellPin);   //  Read photoresistor value
  Serial.print("Photocell value: ");
  Serial.println(ReadValuePhotocell);

  if (ReadValuePhotocell < 800) {  // If light level is low and button has toggled LED on
    digitalWrite(led, HIGH);  // Keep LED on
    Serial.println("LED ON");
  } else if (!value) {
    digitalWrite(led, LOW);  // Turn off LED if button has toggled it off
    Serial.println("LED OFF");
```

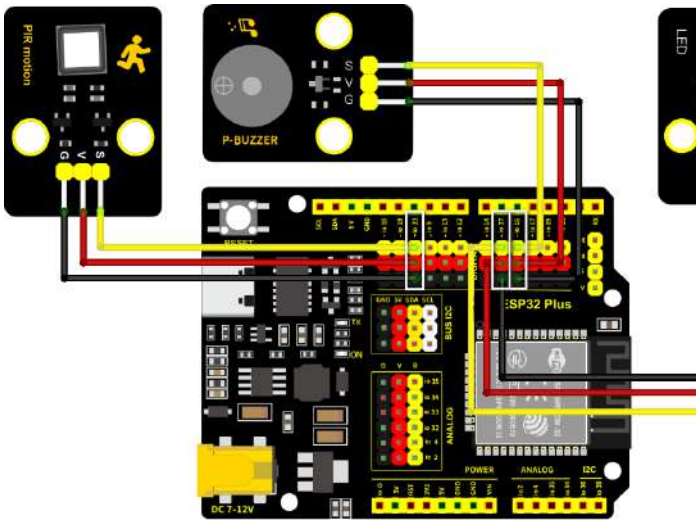
```
  }

  delay(100);  // Short delay to stabilize readings
}
void loop(){
      ledONandOFF();
}
```

- ***Parametrii***
  Tensiune: 3~5V
  Curent: 0,2mA
  Putere: 1mW
  Valoare maximă a spectrului: 540nm
  Rezistență la lumină (10 lux): 5~10KΩ
  Rezistență la întuneric: 0,5MΩ
- ***Rezultat***

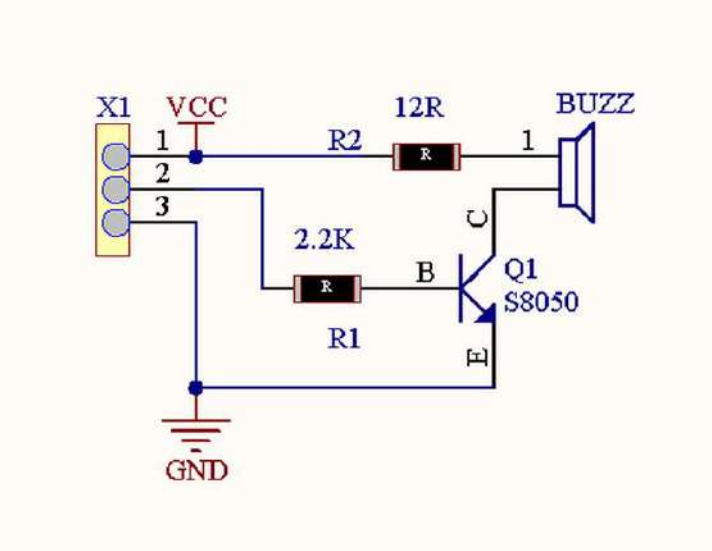
Când fotorezistorul detectează că luminozitatea ambientală este mai mică decât valoarea setată, LED-ul se aprinde. Pe de altă parte, dacă intensitatea luminii ambientale este mai mare decât valoarea setată, fotorezistorul va trimite un semnal diferit pentru a opri LED-ul.

Senzor de mișcare cu alarmă

- ***Diagrama circuitului***



- ***Schemă electrică***



- ***Programare***

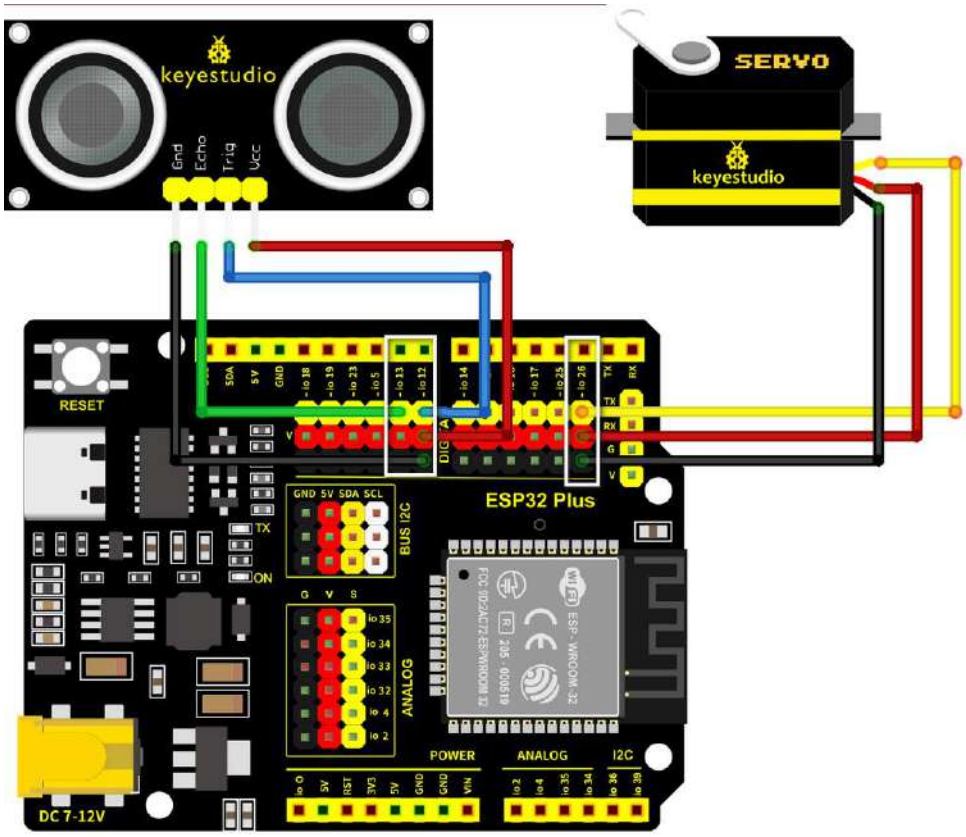
```
#define PyroelectricPIN 23
#define BuzzerPin 16

void setup(){
  pinMode(PhotocellPin, INPUT);
  pinMode(PyroelectricPIN, INPUT);
  pinMode(BuzzerPin, OUTPUT);
}

void nuTeMisca(){
  int ReadValuePyroelectric = digitalRead(PyroelectricPIN);
  if(ReadValuePyroelectric){
    Serial.println("Someone");
    tone(BuzzerPin,532);           //do --C2
    delay(100);
    tone(BuzzerPin,587);           //re --D3
    delay(100);
    tone(BuzzerPin,659);           //mi --E3
    delay(100);
```

```
    //Alarm
    for(int i = 200; i<=1000; i+=10){
     tone(BuzzerPin,i);
     delay(10);
    }
     //Alarm
    for(int i = 1000; i>=200; i-=10){
     tone(BuzzerPin,i);
     delay(10);
    }
     noTone(BuzzerPin);
  }
  else{
    Serial.println("No one");
  }
  delay(100);
}

void loop() {
  nuTeMisca();
}
```

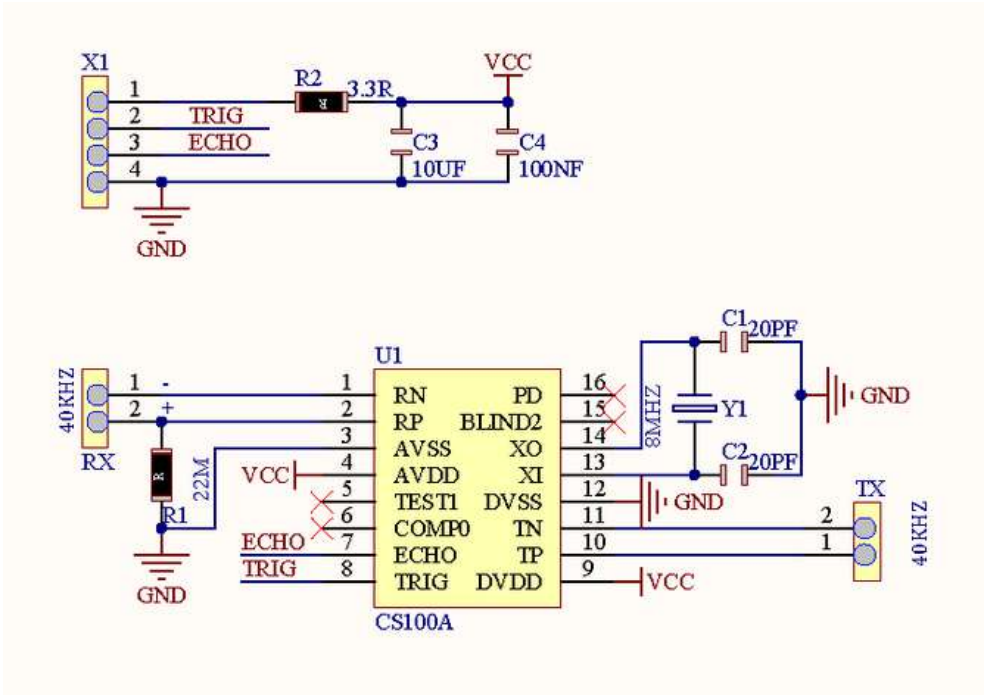
- ***Parametrii***
  Tensiune: 3~5V
  Curent: ≤5mA
  Putere: ≤25mW
- ***Rezultat***
  Când detectează o mișcare, soneria emite un semnal de alarmă.

Deschiderea cutiei prin servo motor

- ***Diagrama circuitului***



- ***Schemă electrică***



- ***Programare***

```
Servo myservo;
int pos = 0;
int servoPin = 26;

#define TrigPin 12
#define EchoPin 13
float duration,distance;

void setup(){
  myservo.attach(servoPin);    // attaches the servo on pin 26 to the servo object
  myservo.write(180);

  pinMode(TrigPin,OUTPUT);  //set trig pin to output mode
  pinMode(EchoPin,INPUT);   //set echo pin to input mode
}

float getDistance() {
  digitalWrite(TrigPin,LOW);
  delayMicroseconds(2);
  digitalWrite(TrigPin,HIGH);
  delayMicroseconds(10);  //Trigger the trig pin via a high level lasting at least 10us
  digitalWrite(TrigPin,LOW);
   duration = pulseIn(EchoPin, HIGH); //the time of high level at echo pin
   distance = (duration * 0.0343) / 2;//convert into distance(cm)
  delay(50);

  Serial.println(distance);
```

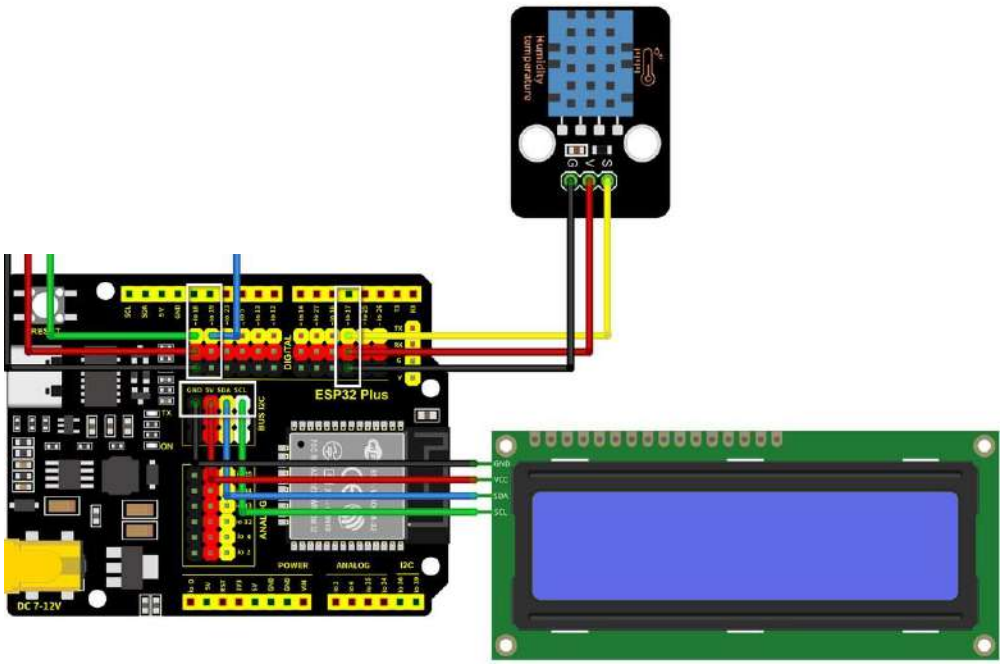
```
  return distance;
}

void servo(){
  float distanta = getDistance();
  Serial.print("Distance: ");
  Serial.println(distanta);
  //When the distance is detected within 2~7cm, open the feeding box. Or else, close.
  if (distanta >= 2 && 7 >= distanta) {
    //Servo rotates to 80° to open the box
    for (pos = 180; pos >= 80; pos -= 1) // goes from 80 degrees to 0 degrees
      myservo.write(pos);    // tell servo to go to position in variable 'pos'
    delay(6000);
    myservo.write(180);
  }
}

void loop() {
  servo();
}
```

- ***Rezultat***

Modulul ultrasonic detectează dacă animalele se află în zona de hrănire, iar Servo motorul deschide automat cutia de hrană pentru păsări.

Detectarea temperaturii și umidității + afișare

- ***Diagrama circuitului***



- ***Programare***

```
//Initialize LCD 1602, 0x27 is I2C address
LiquidCrystal_I2C lcd(0x27,16,2);

#define DHTPIN 17      // Pin where the DHT11 is connected
#define DHTTYPE DHT11  // DHT 11
DHT dht(DHTPIN, DHTTYPE);
// Fast calculate the Dew Point, its speed is 5 times of dewPoint()
void setup(){
  //Initialize LCD
  lcd.init();
  // Turn the (optional) backlight off/on
  lcd.backlight();
```

```
  //lcd.noBacklight();
  //Set the position o dcursor
  Lcd.setCursor(0, 0);
  Dht.begin();
}

double dewPointFast(double celsius, double humidity)
{
        double a = 17.271;
        double b = 237.7;
        double temp = (a * celsius) / (b + celsius) + log(humidity/100);
        double Td = (b * temp) / (a - temp);
        return Td;
}

//Dew Point. The air is saturated and dews are produced under this
temperature.
double dewPoint(double celsius, double humidity)
{
        double A0= 373.15/(273.15 + celsius);
        double SUM = -7.90298 * (A0-1);
        SUM += 5.02808 * log10(A0);
        SUM += -1.3816e-7 * (pow(10, (11.344*(1-1/A0)))-1) ;
        SUM += 8.1328e-3 * (pow(10,(-3.49149*(A0-1)))-1) ;
        SUM += log10(1013.246);
        double VP = pow(10, SUM-3) * humidity;
        double T = log(VP/0.61078);   // temp var
        return (241.88 * T) / (17.558-T);
}

double Kelvin(double celsius)
{
```

```
        return celsius + 273.15;
} //Convert Celsius degree to Kelvins

double Fahrenheit(double celsius)
{
        return 1.8 * celsius + 32;
}    //Convert Celsius degree to Fahrenheit degree

void lcdPrint(float humidity, float temperature){
  // First Screen: Humidity and Temperature in °C
  lcd.clear();
  lcd.print("Hum(%): ");
  lcd.print(humidity, 0);  // 0 decimal places
  lcd.setCursor(0, 1);
  lcd.print("Temp(C): ");
  lcd.print(temperature, 1);  // 1 decimal place
  delay(5000);  // Display for 2 seconds

  // Second Screen: Temperature in °F and Kelvin
  lcd.clear();
  lcd.print("Temp(F): ");
  lcd.print(Fahrenheit(temperature), 0);  // 0 decimal places
  lcd.setCursor(0, 1);
  lcd.print("Temp(K): ");
  lcd.print(Kelvin(temperature), 0);  // 0 decimal places
  delay(5000);  // Display for 2 seconds

  // Third Screen: Dew Point and Dew PointFast in °C
  lcd.clear();
  lcd.print("DewPt(C): ");
  lcd.print(dewPoint(temperature, humidity), 1);  // 1 decimal place
  lcd.setCursor(0, 1);
```

```
  lcd.print("DewPtF(C): ");
  lcd.print(dewPointFast(temperature, humidity), 1);  // 1 decimal place
  delay(5000);  // Display for 2 seconds
}

void temperaturaSiUmiditate() {
    Serial.println("\n");

    float humidity = dht.readHumidity();
    float temperature = dht.readTemperature();

    // Check if any reads failed and exit early (to try again).
    if (isnan(humidity) || isnan(temperature)) {
        lcd.println("Failed to read from DHT sensor!");
        return;
    }

    lcdPrint(humidity, temperature);

  if (temperature >= 29) {
    //Turn left
    ledcWrite(1, 100);
    ledcWrite(3, 0);
  }else{
    //Stop
    delay(3000);
    ledcWrite(1, 0);
    ledcWrite(3, 0);
    delay(200);
  }
}
```

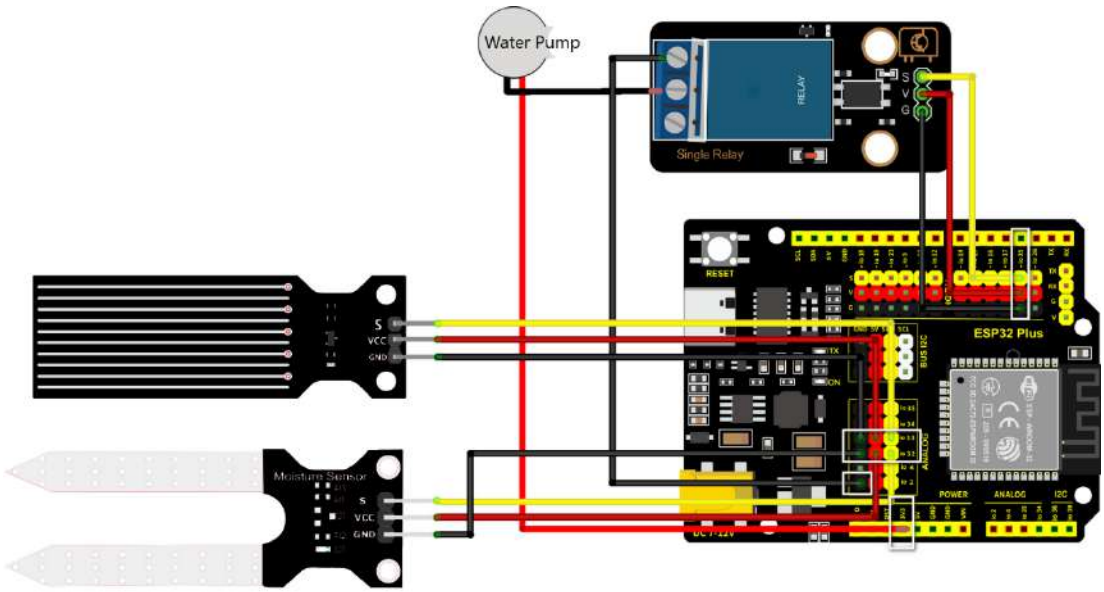
```
void loop() {
  temperaturaSiUmiditate();
}
```

- ***Rezultat***

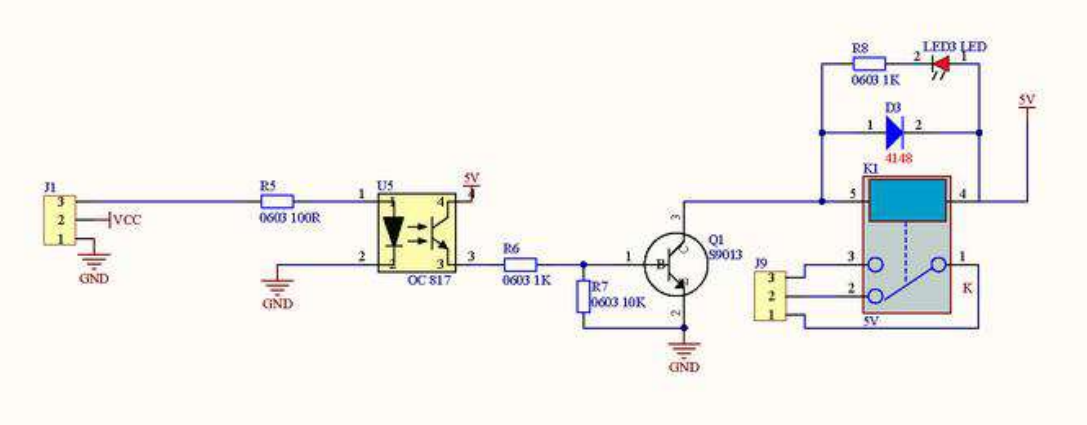
Senzorul de temperatură și umiditate DHT11 emite semnale digitale. Acesta aplică principii de achiziție și conversie a semnalelor analogice, precum și tehnologia de detectare a temperaturii și umidității, oferind astfel stabilitate pe termen lung. În acest context, citim valoarea senzorului de temperatură și umiditate DHT11 prin comunicare, iar valorile vor fi afișate pe LCD. În ansamblu, este un sistem practic care ajută fermierii să monitorizeze și să controleze starea în timp real pentru a îmbunătăți eficiența producției.

Sistem de irigație inteligent

- ***Diagrama circuitului***



- ***Schemă electrică***



- ***Programare***

```
#define SoilHumidityPin 32
#define WaterLevelPin 33
#define RelayPin 25
LiquidCrystal_I2C lcd(0x27,16,2);

void setup(){
//Initialize LCD
  lcd.init();
  // Turn the (optional) backlight off/on
  lcd.backlight();
  //lcd.noBacklight();
  //Set the position o dcursor
  Lcd.setCursor(0, 0);

  pinMode(SoilHumidityPin,INPUT);
  pinMode(WaterLevelPin,INPUT);
  pinMode(RelayPin,OUTPUT);
}

float pamant(){
  int ReadValueSoilHumidity = analogRead(SoilHumidityPin);
```

```
  return ReadValueSoilHumidity;
}

float apa(){
  int ReadValueWaterLevel = analogRead(WaterLevelPin);
  return ReadValueWaterLevel;
}

void udaPamantul(){
  if(pamant()<200) {
    if (apa()>0)
    {
      digitalWrite(RelayPin,HIGH);
      delay(400);//irrigation delay
      digitalWrite(RelayPin,LOW);
      delay(700);
    }
  }
lcd.clear();
  lcd.print("Soil Hum: ");
  lcd.print(pamant());  // 1 decimal place
  lcd.setCursor(0, 1);
  lcd.print("Water: ");
  lcd.print(apa());  // 1 decimal place
  delay(5000);  // Display for 2 seconds
}

void loop() {
  udaPamantul();
}
```

- ***Parametrii***
  Tensiune de alimentare: 5V
  Curent static: 2mA
  Tensiune maximă de contact: 250VAC/30VDC
  Curent maxim: 10A
- ***Rezultat***

Dacă solul este foarte uscat, releul va porni pentru a controla pompa de apă și a iriga plantele; iar dacă nivelul apei este prea scăzut, pompa de apă nu va putea funcționa. Acestea sunt afișate pe ecranul LCD. În acest fel, udarea plantelor și controlul nivelului apei sunt automatizate, ceea ce crește eficiența producției și reduce timpul și eforturile operațiunilor manuale.

6.Concluzii

În concluzie, realizarea acestei ferme inteligente este un mijloc bun pentru ca un elev să dobândească cateva calități benefie pentru un elev care visează să devină un viitor programator datorită părții de codare care va pune bazele unui noi limbaj de programare, ușor de înțeles și exersat de un copil pasionat.

Totuși, partea de asamblare a componentelor poate atrage interesul copiilor pasionați de arhitectură sau inginerie și le sporește încrederea în sine cu cât reușesc să realizeze cu exactitate instrucțiunile oferite. Chiar și creativitatea le este stimulată până în punctul final al experimentului.

Partea de cablare atrage toți entuziaștii de electronică prin circuitul de fire, senzorii cu funcții interesante și conectarea lor la circuitul principal.

Așadar, kitul Smart Farm va entuziasma elevii care îl vor construi în timp ce vor învăța diferite lucruri din domeniile precizate mai sus și vor dobândi competențe folositoare pe viitor indiferent de domeniul în care vor activa.

7.Referințe bibliografice

- https://docs.keyestudio.com/projects/KS0567/en/latest/wiki/index.html#
- https://en.wikipedia.org/wiki/Main_Page
- https://docs.keyestudio.com/projects/KS0567/en/latest/wiki/Arduino/arduino.html#