

IoT in Agriculture

Greenhouse model

Μπάτης Θεοφάνης

*επιβλέπων καθηγητής:
Ψάννης Κωνσταντίνος*

ΠΜΣ τμήμα Εφαρμοσμένης Πληροφορικής



Αντικείμενο εργασίας

- ✓ Δημιουργία θερμοκηπίου σε μακέτα
- ✓ Τοποθέτηση δοχείου νερού
- ✓ Τοποθέτηση δοχείων με χώμα
- ✓ Ενσωμάτωση ηλεκτρονικών συσκευών και αισθητήρων στη μακέτα για αυτοματοποιημένες εργασίες

Γιατί επέλεξα το θέμα

- ✓ Ασχολία με αγροτικές εργασίες σε μικρότερη ηλικία
- ✓ Ευκολία εργασιών
- ✓ Εξοικονόμηση χρόνου
- ✓ Καλύτερη εποπτεία



Σπουδαιότητα θέματος

Αύξηση ζήτησης πρώτων υλών πρωτογενούς τομέα
→ αποδοτικότερες παραγωγικές διαδικασίες στις
γεωργικές καλλιέργειες

Αντιμετώπιση παραγόντων:

- Καιρικές συνθήκες
- Ζιζάνια
- Συνθήκες εδάφους
- Εποπτεία εκτάσεων



IoT (Internet of Things)

Σύνδεση πολλών ηλεκτρονικών συσκευών με το (δια)δίκτυο με σκοπό την ενσωμάτωση της τεχνολογίας σε τομείς που δεν έχουν εφαρμοστεί.

Δημιουργία ενός δικτύου μέσα στο οποίο πολλές συσκευές επικοινωνούν και ανταλλάσσουν δεδομένα για την καλύτερη διαχείριση μιας κατάστασης.

Λίγα λόγια για...

Θερμοκήπιο:

- ✓ Όλοι γνωρίζουμε και έχουμε δει τι είναι
- ✓ Διάφορα σχέδια και υλικά κατασκευής

Μικροελεγκτές:

- ✓ Τους έχουμε παντού γύρω μας
- ✓ Τι είναι?



Μικτοελεγκτές

Είναι τα μικρά «πραγματάκια» που επιτυγχάνουν την ενσωμάτωση της τεχνολογίας σε διάφορους τομείς (IoT).

Πλεονεκτήματα:

- ☐ Αυτονομία
- ☐ Απλή διασύνδεση με περιφερειακά συστήματα
- ☐ Χαμηλό κόστος
- ☐ Αξιοπιστία
- ☐ Χαμηλές εκπομπές ηλεκτρομαγνητικών παρεμβολών
- ☐ Μικρό μέγεθος

Αισθητήρες

Οι ηλεκτρονικές συσκευές που μετατρέπουν σε μετρήσιμες τιμές οποιαδήποτε κατάσταση γύρω μας.

- ☐ Θερμοκρασία
- ☐ Υγρασία χώματος – αέρος
- ☐ Ήχος
- ☐ Κίνηση
- ☐ Φως

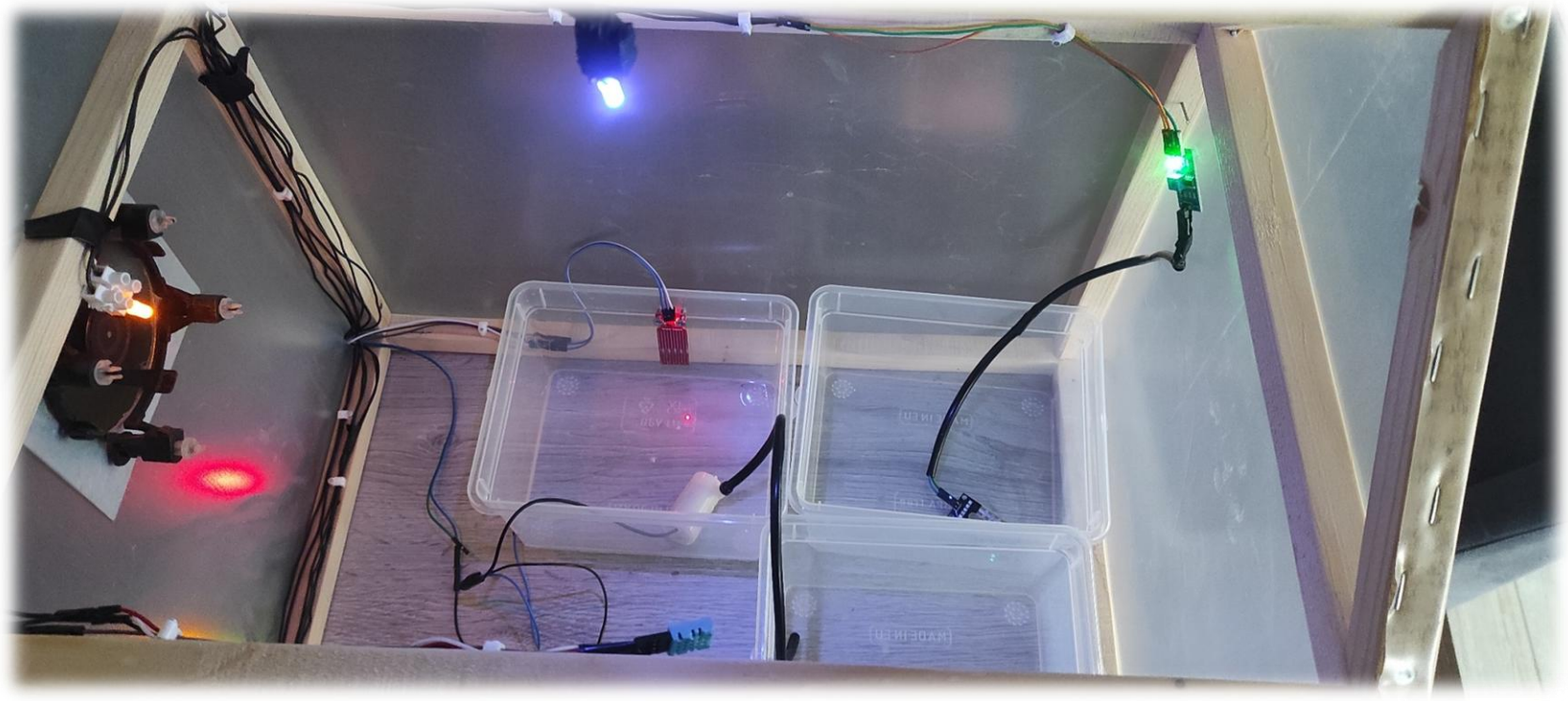
Μακέτα



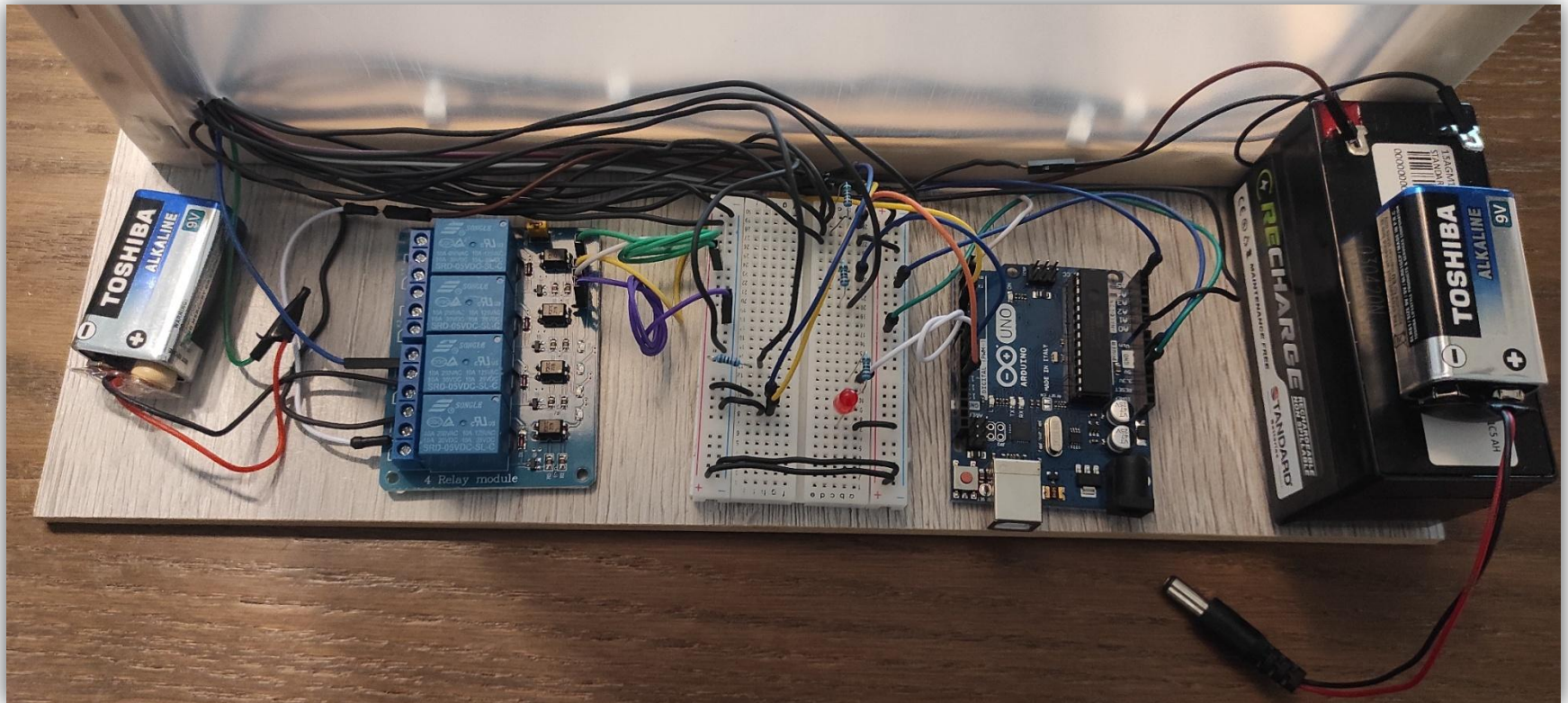
Μακέτα

- Κατασκευή από ξύλο 50x34x50 μ,π,υ
- Επένδυση από νάιλον
- 1 δοχείο νερού
- 2 δοχεία με χώμα
- Πλακέτα **ARDUINO UNO**
- **Breadboard**
- **Relay 4** καναλιών
- Αισθητήρας θερμοκρασίας
- Αισθητήρας υγρασίας χώματος
- Αισθητήρας μέτρησης στάθμης νερού
- Αντλία νερού
- Ανεμιστήρας
- **Led lights**

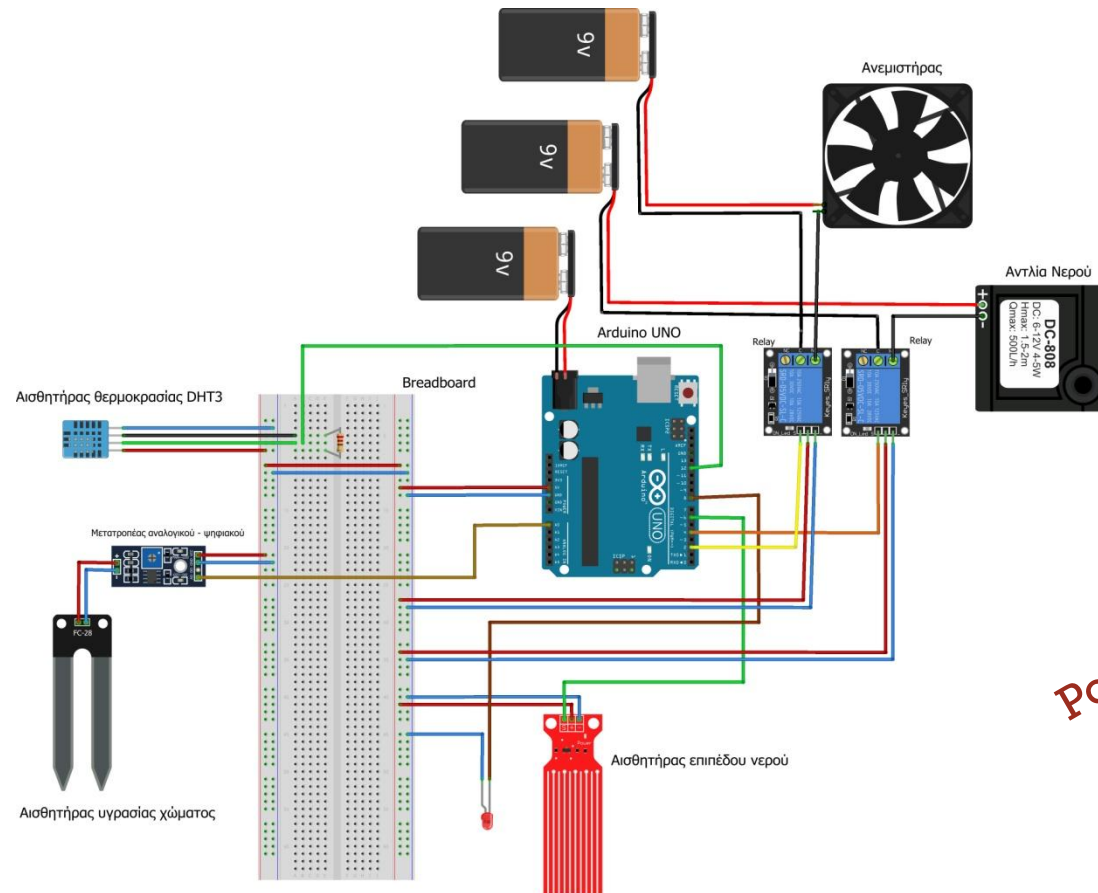
Εσωτερικό μακέτας



Συνδεσμολογία



Schematics



Powered
by
FRITZING

Πλακέτα Arduino UNO

Ο «εγκέφαλος» της μακέτας με μέγεθος μισή παλάμη.

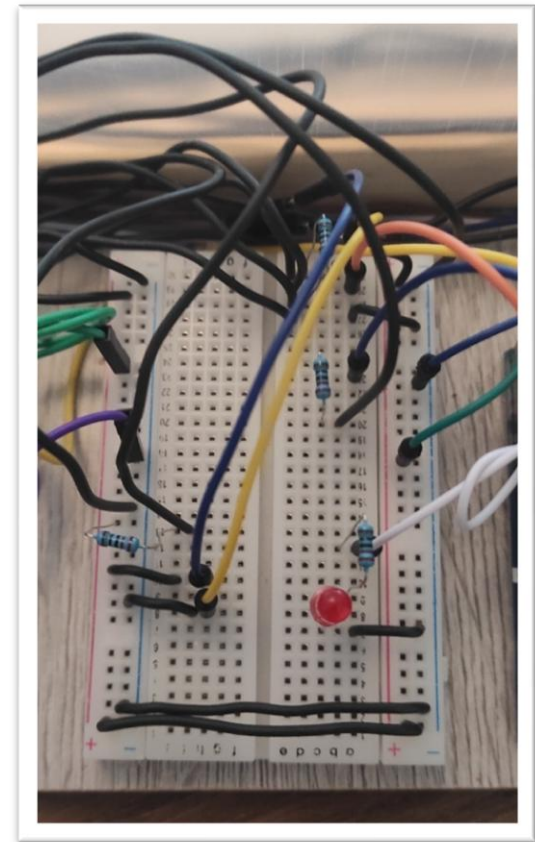
- Θύρα USB για σύνδεση σε Η/Υ
- Θύρα ρεύματος
- Τάση ρεύματος εξόδου 5V
- Τάση ρεύματος εξόδου 3.3V
- **3 Pins** γείωσης
- 14 ψηφιακά Pins εισόδου/εξόδου
- 6 αναλογικά Pins εισόδου/εξόδου



Breadboard

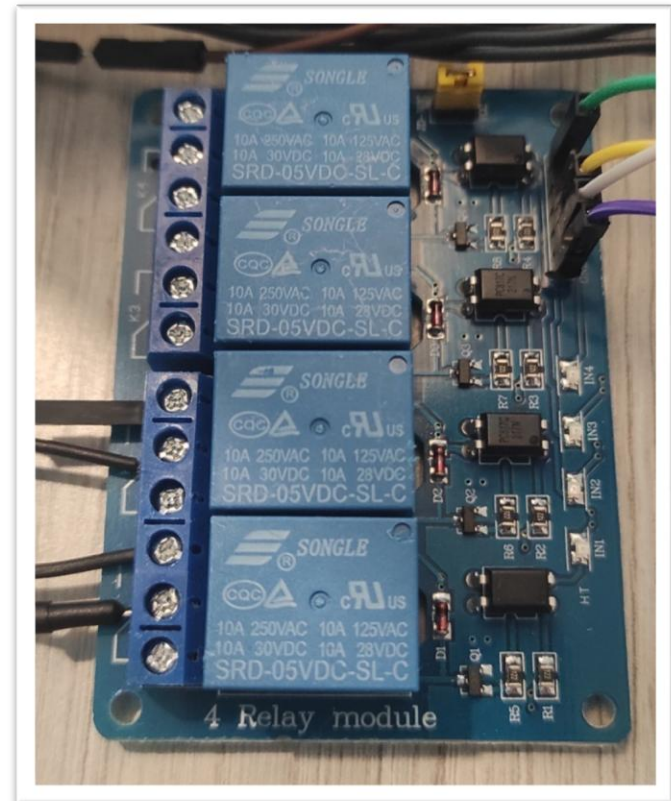
Η «επέκταση» των Pins

- Μέγεθος επίσης μισής παλάμης
- Διευκόλυνση στην σύνδεση πολλών ηλεκτρονικών εξαρτημάτων.
- Κάθε υποδοχή χαρακτηρίζεται από έναν αριθμό και ένα γράμμα (Ναυμαχία!).



Relay 4 channels

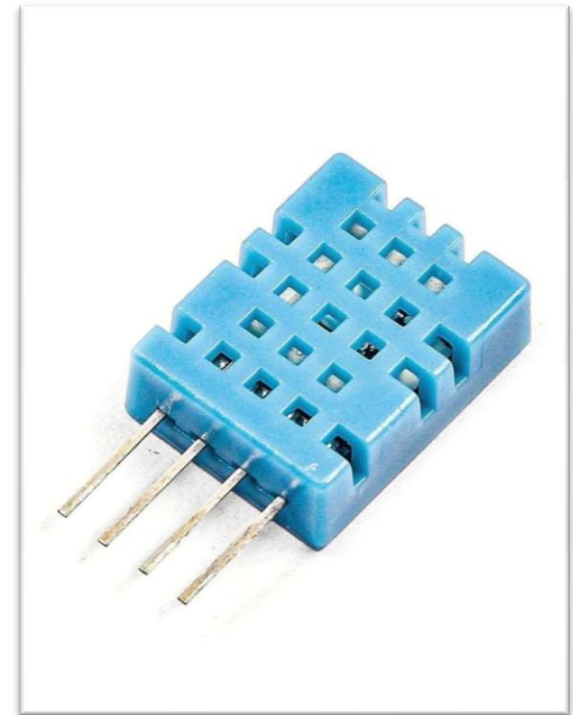
- Διαχωρισμός κυκλωμάτων
- Κάθε **relay** χρησιμοποιεί δικιά του πηγή ενέργειας.
- Η πηγή ενέργειας της πλακέτας δεν επαρκεί για όλα τα ηλεκτρονικά εξαρτήματα



Αισθητήρας θερμοκρασίας – υγρασίας αέρα

DHT11

- Μέγεθος ξύστρας
- Εύρος θερμοκρασίας 0°C- 60°C
- Εύρος υγρασίας 20%-60%
- Απόκλιση $\pm 1^{\circ}\text{C}$ και $\pm 1\%$



Αισθητήρας μέτρησης υγρασίας χώματος

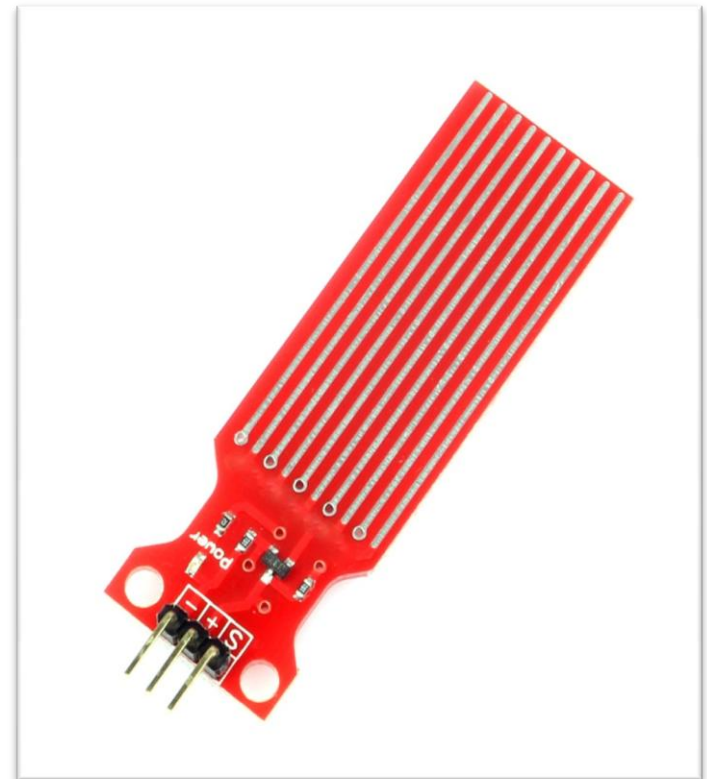
Τα «πιρούνια» μετράνε
το νερό που υπάρχει
μέσα στο χώμα.

Το δεύτερο κομμάτι
μετατρέπει τις τιμές
που πήραν τα
«πιρούνια» σε
ψηφιακές για την
καλύτερη ανάγνωση
και επεξεργασία



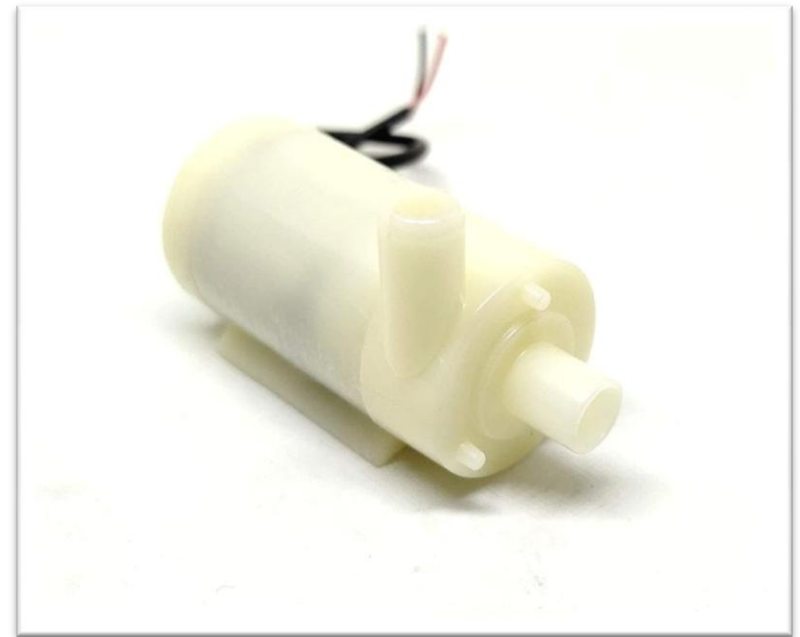
Αισθητήρας μέτρησης στάθμης νερού

Προσαρμόστηκε στο δοχείο νερού και εμφανίζει το επίπεδο στάθμης του νερού. Αν αδειάσει ή πλησιάζει να αδειάσει το νερό, αναβοσβήνει ένα λαμπάκι για υπενθύμιση.



Αντλία νερού

Λειτουργεί αυτόματα
όταν απαιτείται
άντληση νερού.
Χρησιμοποιεί δικιά του
πηγή ενέργειας καθώς
είναι 12V και δεν
επαρκεί η παροχή από
την πλακέτα.



Ανεμιστήρας

Τάση 12V επομένως
χρησιμοποιεί δικιά του
πηγή ενέργειας μέσω
του **relay**.

Ψύξη του εσωτερικού
της μακέτας

Ένδειξη λειτουργίας με
ένα **led** λαμπάκι καθώς
είναι αθόρυβο



Προγραμματισμός μέσω Arduino IDE



2 σκέλη προγραμματισμού:

void setup()

Τρέχει μία φορά κατά την έναρξη της πλακέτας.
Αρχειοποίηση λειτουργίας αισθητήρων και σειριακή ανάγνωση

void loop()

Βασική λειτουργικότητα.
Τρέχει συνέχεια (λούπα) μέχρι να τερματιστεί χειροκίνητα.

Κώδικας

```
#include "DHT.h"           // Library for DHT sensors

#define dhtPin 12           // data pin of DHT sensor connected to port 12 of Arduino Uno
#define dhtType DHT11       // DHT 11 sensor
#define fan_case 2          // data pin of fan case connected to port 4 of Arduino Uno
#define water_pump 4        // data pin of water pump connected to port 4 of Arduino Uno
#define low_water_led 8     // LED light for low water connected to port 8 of Arduino Uno
#define fan_case_led 10     // LED light for active fan case connected to port 10 of Arduino Uno
#define sensor_pin A0       // data pin of Soil Sensor connected to analog port A0 of Arduino Uno
#define respin A5           // data pin of water level sensor connected to analog port A5 of Arduino Uno

DHT dht(dhtPin, dhtType);   // Initialise the DHT library
float humidity;             // humidity
float temperature;          // temperature in degrees Celcius
int output_value;          // value of moisture
int resval = 0;             // holds the value of water level sensor
```

Δήλωση μεταβλητών πριν από την συνάρτηση **void setup()**. Επίσης, προσθέτουμε τυχόν βιβλιοθήκες που απαιτούνται για την σωστή λειτουργία των αισθητήρων

Κώδικας

```
void setup() {  
  pinMode(fan_case, OUTPUT);  
  pinMode(water_pump, OUTPUT);  
  pinMode(fan_case_led, OUTPUT);  
  pinMode(low_water_led, OUTPUT);  
  
  Serial.begin(9600);      // Initialise the serial monitor  
  dht.begin();             // start with reading the DHT sensor  
  
  Serial.println("Reading From the Soil Sensor ...");  
  delay(2000);  
}
```

Έναρξη λειτουργίας της πλακέτας και δήλωση
θυρών (είσοδος – έξοδος)

Κώδικας

```
void loop() {  
  
    humidity = dht.readHumidity();          // get the humidity from the DHT sensor  
    temperature = dht.readTemperature();    // get the temperature in degrees Celcius from the DHT sensor  
  
    // Check if all values are read correctly, if not try again and exit loop()  
    if (isnan(humidity) || isnan(temperature)) {  
        Serial.println("Reading DHT sensor failed!");  
  
        return;  
    }  
  
    Serial.print("Humidity: ");  
    Serial.print(humidity);  
    Serial.print(" %\t");  
    Serial.print("Temperature: ");  
    Serial.print(temperature);  
    Serial.print(" °C ");  
    Serial.print(" %\n");
```

Κώδικας

```
output_value = analogRead(sensor_pin);           // Read data from analog pin and store value to output_value variable
output_value = map(output_value, 550, 10, 0, 100); // map data in percentage (%)
Serial.print("Moisture : ");
Serial.print(output_value);
Serial.println("%");

//Start water pump if humidity is low
if (output_value < 0) {
    digitalWrite(water_pump, LOW);
}
else {
    digitalWrite(water_pump, HIGH);
}

//Start fan case if temperature is high
if (temperature > 30) {
    digitalWrite(fan_case, LOW);
    digitalWrite(fan_case_led, HIGH);
}
else {
    digitalWrite(fan_case, HIGH);
    digitalWrite(fan_case_led, LOW);
}
```

Κώδικας

```
resval = analogRead(respin);    // Read data from analog pin and store it to resval variable
Serial.println(resval);

if (resval <= 100) {

    Serial.println("Water Level: Empty");
    digitalWrite(low_water_led, HIGH);
    delay(100);
    digitalWrite(low_water_led, LOW);
    delay(100);
}

else if (resval > 100 && resval < 500) {
    Serial.println("Water Level: Medium");
    digitalWrite(low_water_led, LOW);
}

else if (resval >= 500) {
    Serial.println("Water Level: High");
    digitalWrite(low_water_led, LOW);
}

delay(2000);
}
```

Σειριακή ανάγνωση

```
COM4
21:51:51.272 -> Water Level: Empty
21:51:53.476 -> Humidity: 67.00 %      Temperature: 26.40 °C %
21:51:53.522 ->
21:51:53.522 -> Moisture : -72%
21:51:53.522 -> Water Level: Empty
21:51:55.788 -> Humidity: 67.00 %      Temperature: 26.40 °C %
21:51:55.788 ->
21:51:55.788 -> Moisture : -72%
21:51:55.788 -> Water Level: Empty
21:51:57.975 -> Humidity: 67.00 %      Temperature: 26.40 °C %
21:51:57.975 ->
21:51:58.022 -> Moisture : -72%
21:51:58.022 -> Water Level: Empty
21:52:00.179 -> Humidity: 67.00 %      Temperature: 26.40 °C %
21:52:00.225 ->
21:52:00.225 -> Moisture : -72%
21:52:00.272 -> Water Level: Empty
21:52:02.421 -> Humidity: 67.00 %      Temperature: 26.30 °C %
21:52:02.482 ->
21:52:02.482 -> Moisture : -72%
21:52:02.482 -> Water Level: Empty
21:52:04.689 -> Humidity: 68.00 %      Temperature: 26.40 °C %
21:52:04.736 ->
21:52:04.736 -> Moisture : -72%
21:52:04.736 -> Water Level: Empty
```

☒ Αυτόματη κύλιση ☒ Εμφάνιση χρονοσήμανσης Αλλαγή γραμμών

Μέσα από το λογισμικό **ARDUINO IDE** υπάρχει η επιλογή παρακολούθηση σειριακής ανάγνωσης.

Εμφανίζει τα δεδομένα που λαμβάνονται από τους αισθητήρες.

Βίντεο παρουσίασης μακέτας

Στον παρακάτω σύνδεσμο θα βρείτε το βίντεο που παρουσιάζεται αναλυτικά η μακέτα καθώς και οι λειτουργίες της:

<https://www.youtube.com/watch?v=1NMNb9IFMb0>



Συμπεράσματα

1. Με την ενσωμάτωση της τεχνολογίας μπορούμε να μετρήσουμε τα πάντα γύρω μας.
2. Ακόμα και με χαμηλό κόστος μπορεί η τεχνολογία να εφαρμοστεί σε μικρότερες καλλιέργειες.
3. Πρόληψη και προστασία από κινδύνους (καιρικών φαινομένων κυρίως)
4. Απομακρυσμένη εποπτεία
5. Αποδοτικότερη παραγωγή
6. Εξοικονόμηση χρόνου

Thank you!!

