

-----Fejlesztői dokumentáció-----

1) Az arduino uno kódja

Az arduino mikrokontroller kódját illetően a kódban szereplő kommentek egészen beszélések a működést illetően.

Néhány fontos dolog:

1. A föld nedvesség szenzor AirValue és WaterValue értékét használat előtt egy méréssel kell elvégezni, üresen kell mérni egyet a levegőben valamint a vízben és kiírni soilMoistureValue változót a soros monitorra.

```
//////////föld nedvesség sensor
const int AirValue = 500;    //Üresen kell mérni egyet a levegőben és helyettesíteni az értékkel
const int WaterValue = 220;  //Üresen kell mérni egyet a vízben és helyettesíteni az értékkel
```

Ezután a százalék kiszámítása mappolással történik, a soilMoistureValue mért értékéből, valamint a levegő és a víz kezdeti ellenálásából mérve történik.

```
soilMoistureValue = analogRead(A0);    //talajnedvesség

soilmoisturepercent = map(soilMoistureValue, AirValue, WaterValue, 0, 100);
```

2.A páratartalomhoz és a hőmérséklethez meg kell hívni a DHT könyvtárat (ezt előtte le kell tölteni) (#include "DHT.h"), és onnantól használhatóvá válik readHumidity().

```
#include "DHT.h"
```

3.Az LCD panel működtetéséhez le kell tölteni és meg kell hívni a szükséges könyvtárat (#include <LiquidCrystal_I2C.h>)

```
#include <LiquidCrystal_I2C.h>
```

Ezután a setcursor segítségével be tudjuk állítani a képernyő mely pontján kezdjük el kiírni a karaktereket.

```
lcd.setCursor(0,0);  
lcd.print(F("Paratart.:"));  
lcd.setCursor(10,0);
```

4. A felszín nedvességmérője egészen egyszerű módon működik, megméri hogy van-e ellenállás vagy nincs, és ez alapján ad vissza egy LOW illetve HIGH jelet, ezekre kell feltételt írni.

```
//felszín nedvesség  
String felsz;  
if(digitalRead(capteur_D) == LOW)  
{  
    felsz = "nedves";  
  
    lcd.setCursor(0,3);  
    lcd.print("talaj nedves");  
}  
  
else  
{  
    felsz = "szaraz";  
  
    lcd.setCursor(0,3);  
    lcd.print("talaj szaraz");  
},
```

A vízpumpa egészen addig fog működni amíg a talajnedvesség el nem éri a 70%-ot, természetesen az érték változtatásával ezt a határt el tudjuk tolni.

```
// Ha talajnedvesség alacsony bekapcsol a vízpumpa  
while (soilmoisturepercent<70){  
    digitalWrite(relay, HIGH);  
    soilMoistureValue = analogRead(A0);  
    soilmoisturepercent = map(soilMoistureValue, AirValue, WaterValue, 0, 100);  
}
```

Legutolsó lépésben pedig 3 mért adatot (páratartalom, hőmérséklet és talajnedvesség) összefűzzük és kiírjuk a soros monitorra hogy átadhassuk az ESP-01S eszköznek, majd 5 másodpercet várunk.

```
Serial.begin(9600);  
String uzenet = String(h) + "," + String(t) + "," + String(soilmoisturepercent);  
  
Serial.print(uzenet);  
Serial.print('\n');
```

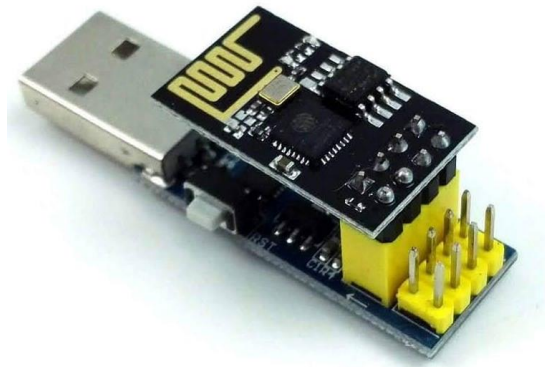
Serial monitor:

```
33.00,24.40,81  
33.00,23.70,81
```

2) Az ESP-01S kódja

Legelőször hozzá kell adni magát az eszközt az arduino IDE-hez hogy tudjuk programozni. Ehhez mellékelek egy tutorialt (<https://iotcircuitHub.com/esp8266-programming-arduino/>).

Az eszköz programozása úgy néz ki hogyha nincs ehhez egy pár dolláros kis eszközünk (esp01 usb programmer, alább látható) hogy az alábbi módon kell összekötnünk az arduinot és az ESP modult:



ESP-01S		Arduino
3v3	---	3.3V
RST	---	GND
EN	---	3.3V
TX	---	TX
RX	---	RX
ISO0	---	GND
ISO2	---	
GND	---	GND

Fel kell tölteni először az arduino-ra a felprogramozo_arduino.ino file, ezután engedni fogja az tx-rx lábakon a feltöltést az esp modulra.

A sorrend pedig az hogy be kell dugni először a GND-be az ISO0 pint, elkezdünk felölteni, és amikor connecting..... állapotban van az eszköz, 0.5 másodpercre be kell dugni a GND-be az EN pint és gyorsan kihúzni, ezt a lépést lehet kétszer is meg kell ismételni, ezután látni fogjuk hogy elkezdünk írni az esp eszköz memóriacímeire.

```
Sketch uses 281280 bytes (29%) of program storage space. Max: 1024000 bytes.
Global variables use 28140 bytes (34%) of dynamic memory, leave free: 255860 bytes.
esptool.py v2.7
Serial port COM3
Connecting.....
Chip is ESP8266EX
Features: WiFi
Crystal is 26MHz
```

Feltöltés után:

```
Stub running...
Configuring flash size...
Auto-detected Flash size: 1MB
Compressed 285440 bytes to 206686...
Wrote 285440 bytes (206686 compressed) at 0x00000000 in 18.2 seconds (effective 125.1 kbit/s)
Hash of data verified.
```

Az általános paraméteres átírásokat a felhasználói dokumentáció tartalmazza (SSD, password, webhook_key).

Elsőkörbe fontos megemlíteni hogy a webhook 3 paramétert tud átadni maximum (ezért is nem az összeset küldjük át a soros monitoron, csak a relatív fontosakat)

```
//átadjuk a webhooknak ami elküldi a http kérést
webhook.trigger(paratartalom,homerseklet,talajnedv);
```

A setup részben történik a wifi hálózathoz a kapcsolódás, szóval a hálózatot előtte be kell kapcsolni, a jelenlegi verzió nem tartalmazza a jel elvesztése esetén az újracsatlakozás futás közbeni lehetőségét.

```
pinMode(LED_BUILTIN, OUTPUT);  
digitalWrite(LED_BUILTIN, LOW);  
WiFi.mode(WIFI_STA);  
WiFi.disconnect();  
delay(1000);  
  
//kapcsolódás a hálózathoz  
WiFi.begin(_SSID, _PASSWORD);
```

A működése pedig úgy néz ki hogy amint a bufferbe bájtok kerünek a soros monitorról, karakterenként beolvassuk a küldött sort, és amint egy new line ('\n') karaktert észlelünk az adást teljesnek tekintjük és az immár teljes mértékben beolvasott sort szegmentáljuk egy előre definiált karakter alapján (.). A szétördelt stringeket egy string tömbbe tároljuk és ezután átadjuk paraméterként a webhook.trigger-nek ami elküldi a HTTP kérést az oldal irányába.

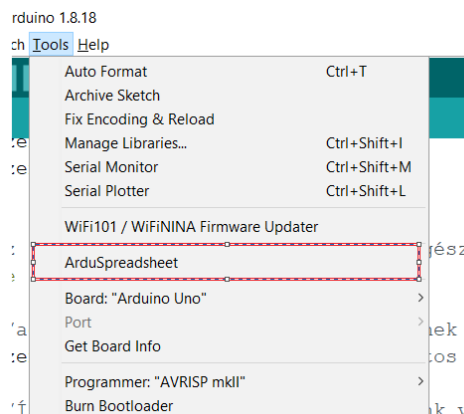
3) Arduspreadsheet kezelése

Először le kell tölteni és hozzá kell adni az Arduspreadsheet kiegészítőt az Arduino IDE-hez, fontos tudni azonban hogy ez nem kompatibilis az Arduino IDE 2.0-val.

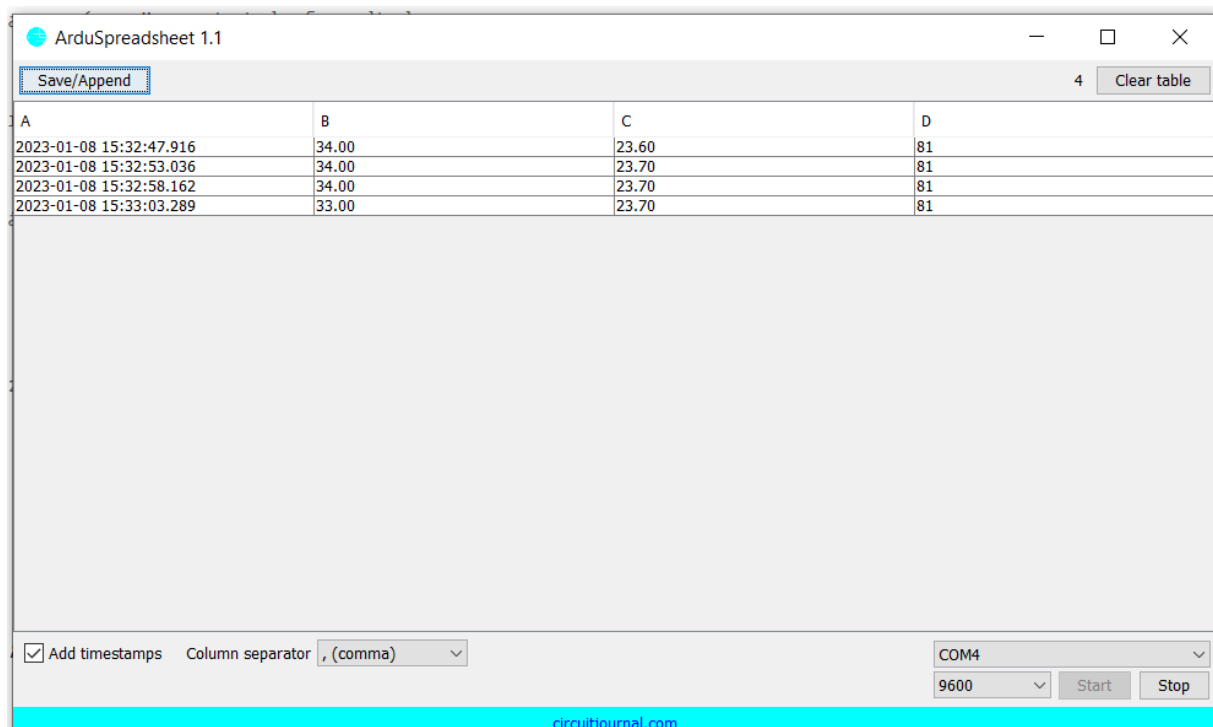
(A letöltési és telepítési tutorialhoz alább mellékeltem egy videót:

<https://www.youtube.com/watch?v=rT1CIa-ZiNo>)

Miután hozzáadtuk az alkalmazáshoz, a tools menüpont alatt elérhetővé is válik.



Miután elkezdünk adatot kiírni az soros monitorra, rá kell menni erre az újonnan megnyílt menüpontra.



Ezután a save/append gomb megnyomásával tudunk generálni egy .csv file-t. amit excel segítségével be tudunk oszlopokba olvasni.

File Home Insert Page Layout Formulas Data Review

Table Name:

Summarize with PivotTable

Remove Duplicates

Convert to Range

Insert Slicer

Export

Refresh

Properties Tools

A1

	A	B	C	D	E
1	Column1	Column2	Column3	Column4	
2	2023.01.08 15:32	34.00	23.60	81	
3	2023.01.08 15:32	34.00	23.70	81	
4	2023.01.08 15:32	34.00	23.70	81	
5	2023.01.08 15:33	33.00	23.70	81	
6	2023.01.08 15:33	34.00	23.70	81	
7	2023.01.08 15:33	34.00	23.80	81	
8	2023.01.08 15:33	33.00	23.80	81	
9	2023.01.08 15:33	33.00	23.80	81	
10	2023.01.08 15:33	33.00	23.80	81	
11	2023.01.08 15:33	33.00	23.80	81	
12	2023.01.08 15:33	33.00	23.80	81	
13	2023.01.08 15:33	33.00	23.80	80	
14	2023.01.08 15:33	33.00	23.80	81	
15					

