IOT

Trembeczki Dávid

B58TC3

Programtervező Informatikus, Levelező tagozat

A beadandó dolgozatban megvalósított projekt egyszerű: egy ESP32 mikrokontroller segítségével és egy hő és páratartalom mérő szenzorral kinyerek adatot és egy webszerverre iratom ki.

1. **Eszközök**

Ebben a projektben felhasznált eszközöket a modulshop.hu webshopból rendeltem, de ezeket be lehet szerezni más külföldi oldalról is.

* ESP32WROOM32U mikrokontroller

A képen elektronika, Elektronikus alkatrész, Áramköri elem, Számítógép-alkatrész látható

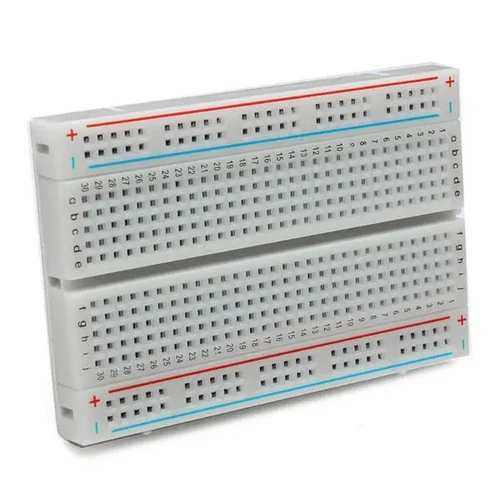
Automatikusan generált leírás

* DHT11 Hőmérséklet és páratartalom érzékelő

A képen Áramköri elem látható

Automatikusan generált leírás

* 400 lyukas breadboard, próbapanel



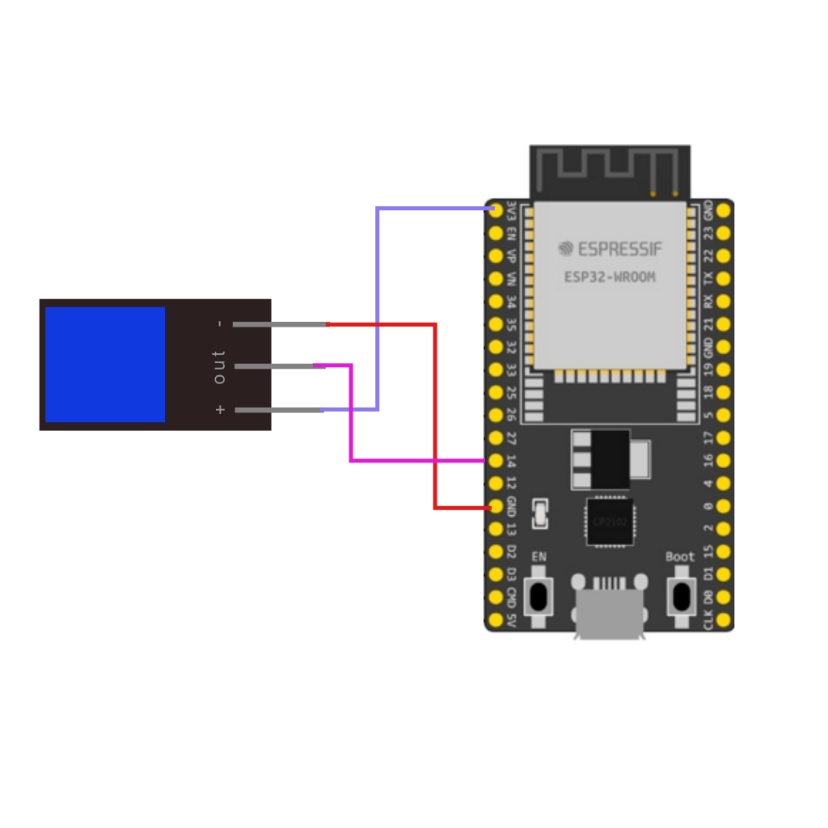
* jumper kábelek

1. **Előkészület és összeállítás**

Az összeállítás nem egyszerű mivel figyelni kell az ESP32 pin kiosztására. Ha megtaláltuk a megfelelő pineket, akkor a szenzorral össze kell kötni a kábelek és a breadboard segítségével.

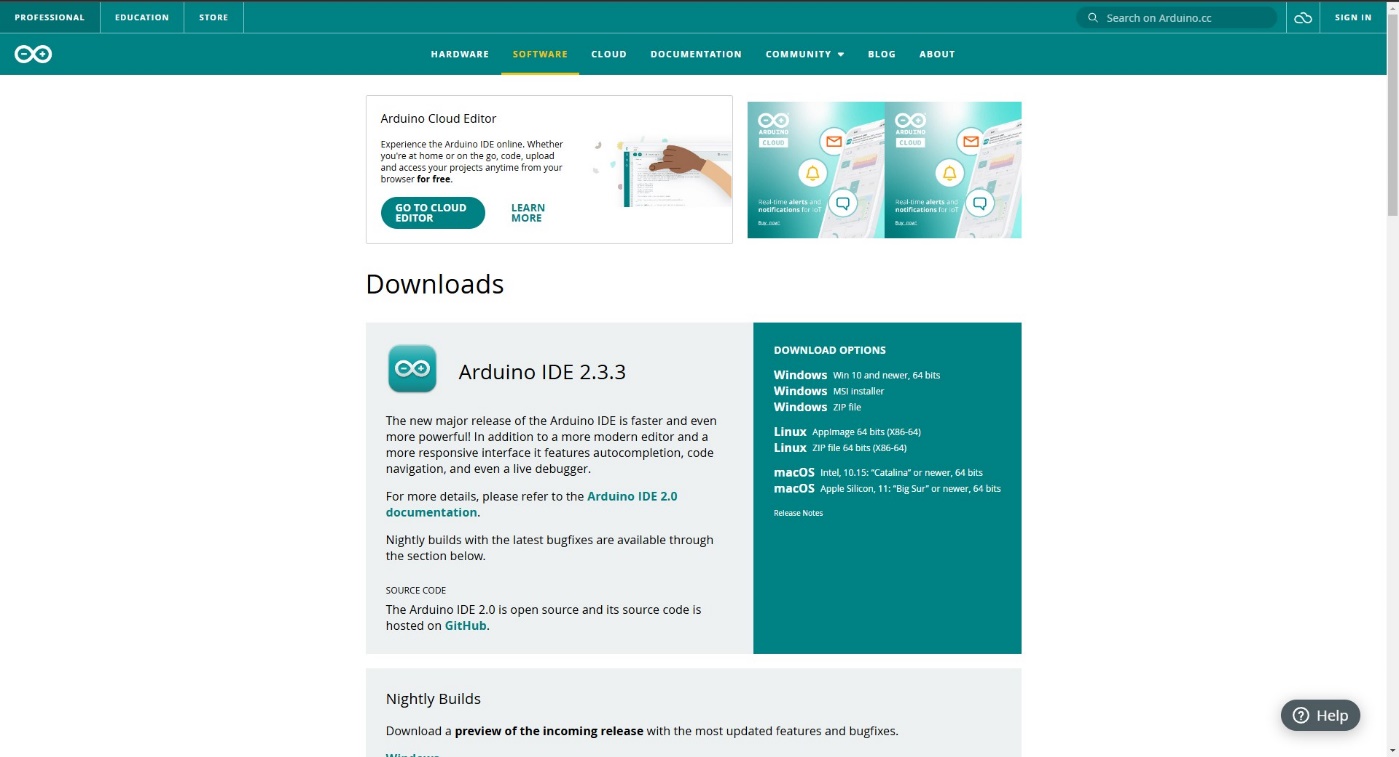
A képen szöveg, képernyőkép, tervezés látható

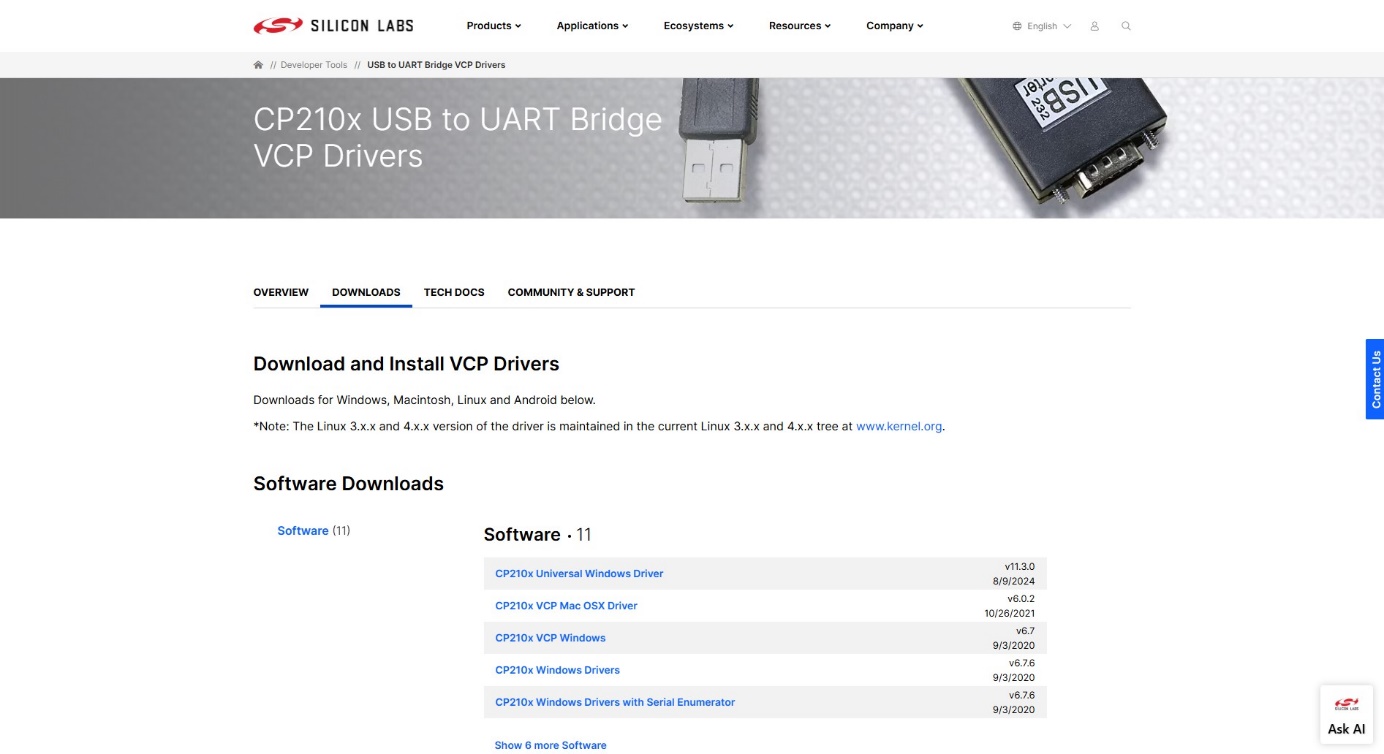
Automatikusan generált leírás

Csatlakoztattam elsőkörben anya-anya jumper kábellel az ESP32-t és a DHT11 érzékelő modult: A 3v3 pint a + megjelölésű pinre, a GND-t a – megjelölésű pinre és a GPIO14 pint pedig az „out” pinre.

1. **Kódolás**

A kódot az „Arduino IDE” nevezetű szoftverben lehet futtatni és rátölteni az ESP32-re. Ez viszont nem elég: ahhoz, hogy megfelelően funkcionáljon a mikrokontroller a CP210x drivert kellett telepíteni. Ezek után pedig a megfelelő sensor könyvtárát kell telepíteni, ami egyszerűsíti a programozó feladatát.



A képen szöveg, elektronika, képernyőkép, kütyü látható

Automatikusan generált leírás

A kód egy része saját fejlesztés viszont a WiFi server létrehozására egy már meglévő, beépített, működő kód szolgált alapul.

A Wifi server kódját a File→Examples→WiFi→SimpleWiFiServer néven lehet elérni. Itt az ssid pointer-hez meg kell adni a WiFi hálózat nevét és a \*password pointerhet pedig a jelszavát.

DHT dht(14, DHT11) függvénynél az első argumentum a felhasznált GPIO pin, a második pedig a használt szenzor pontos kódja. (Ebből a szenzor családból elérhetó a DHT21 is)

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

Itt megadjuk a temp és a humidity változók értékét, amit a dht könyvtárban lefejlesztett dht.readTemperature() és a dht.readHumidity() függvények olvasnak ki a szenzorból.

A hálózatra csatlakozás után a Serial Monitor-on kiírja a lokális IP címet, amit beírva a böngészőbe megnyílik az oldal.

A HTTP headerök használata hasznos lehet ebben az esetben: itt nem alapértelmezett az UTF-8 kódolás, így az a "Content-type:text/html; charset=utf-8" kerül beállításra. "Refresh: 2" HTTP headerrel az oldal 2 másodpercenkénti frissítését lehet elérni, így nem kell nekünk frissítgetni az oldalt, viszonylag friss adatokat kapunk.

1. **Eredmény és Összegzés**

A fejlesztés alatt több verzió és prototípus is született, majd pedig a kész kódot WiFi szerveren futtattam, amit a telefonom segítségével képernyőfotóval le is fényképeztem.A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás