

Juhász Attila
IoT eszköz alapú adatgyűjtő
rendszer

Konzulens: Krébesz Tamás István

Önálló laboratórium 2 (de ebben a félévben elkezdett téma)

A feladat leírása

- ▶ Adatgyűjtésre képes IoT rendszer megalkotása
- ▶ ESP8266 NodeMCU segítségével
- ▶ Thingspeak felhő alapú szolgáltatás igénybevételével
- ▶ Feladat kiterjesztése során: Weblap hosztolása a chip-pel
- ▶ A weblapról történő vezérlésének a megoldása

IoT rendszerekről általában

- ▶ Általános funkciójuk: csatlakoztatott- és távoli műveletek megvalósítása, előrejelző elemzések készítése, megelőző karbantartás, logisztikai megoldások segítése, erőforrás menedzsment
- ▶ Megvalósításuk:
 1. smart eszközök, kommunikációra képes elektronikus eszközök, amelyek a környezetükről gyűjtött információt helyben feldolgozzák / továbbítják.
 2. Beavatkozás a környezetbe aktuátorokkal stb.

Adatok helyben feldolgozása / továbbítása

- ▶ Big Data
- ▶ Cloud computing: az eszközökről a gyűjtött információ továbbítása szerverekre
- ▶ Fog computing: LAN hálózatokon történik a számítás, gyorsabb válaszidő
- ▶ Edge computing: az eszközön magán valósul meg a számítás, gyorsabb válaszidő, sávszélesség megtakarítása
- ▶ IaaS, SaaS, PaaS

Példák

- ▶ Smart home, épületautomatizálás
- ▶ Smart grid, smart city
- ▶ Smart healthcare
- ▶ IIoT: ipari automatizálás
- ▶ Intelligent transportation system
- ▶ ...



PC

Accessing TS diagrams via internet



Accessing the Webpage, and controlling the ESP via the page

Sending datats to the page

Controlling the ESP

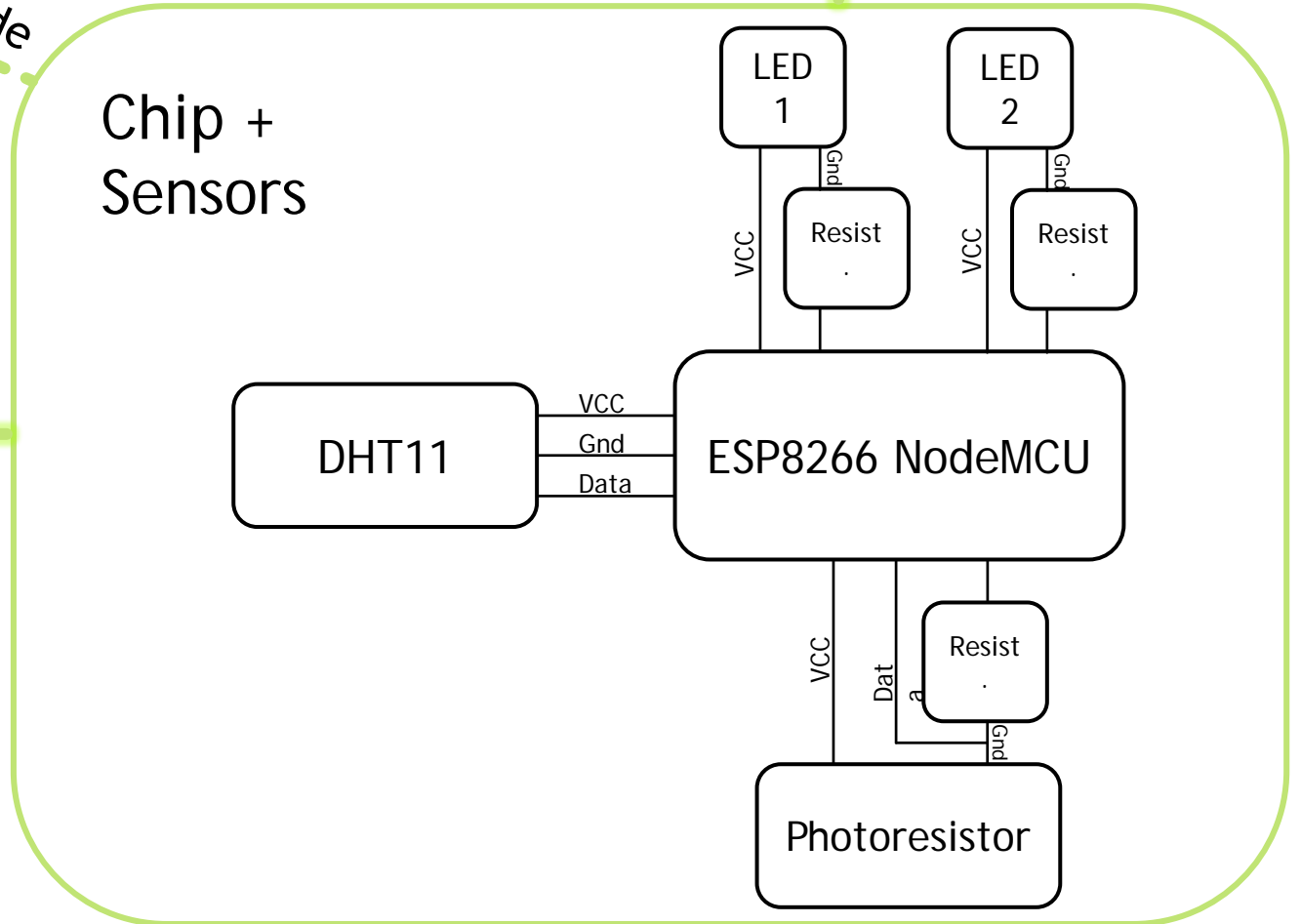


Web server

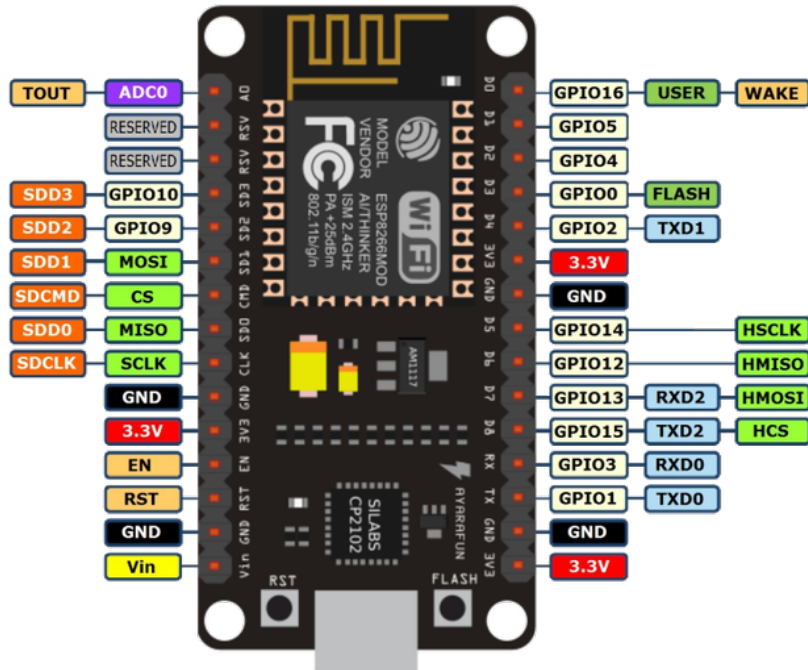
Uploading the code



Sending datats to Thingspeak



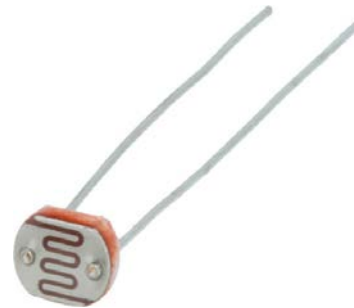
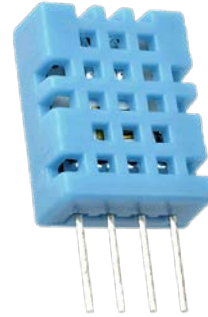
Hardver



- ▶ ESP8266 NodeMCU:
- ▶ Wi-Fi képes eszköz
- ▶ CPU: 80 Mhz, 32 bites RISC
- ▶ 128 kbyte RAM
- ▶ 4 Mbyte flash memória
- ▶ 13 GPIO láb
- ▶ SPI, I²C támogatása

Szenzorok

- ▶ Fotoellenállás:
 - ▶ A megvilágítottságtól függ az ellenállása
 - ▶ Minél nagyobb a megvilágítottság, annál kisebb az ellenállása
- ▶ DHT11:
 - ▶ Tartomány: 0-50°C, 20-90%
 - ▶ Felbontás: 16 bit
 - ▶ Pontosság: $\pm 1^\circ\text{C}$, $\pm 1\%$



Thingspeak



My Channels

[New Channel](#)

Name	Created	Updated
Humidity and Temperature Monitoring Private Public Settings Sharing API Keys Data Import / Export	2020-09-18	2020-09-18 20:01
Photoresistor Private Public Settings Sharing API Keys Data Import / Export	2020-10-15	2020-10-15 20:51

Help

Collect data in a ThingSpeak channel from another channel.

Click **New Channel** to create a new channel.

Click on the column headers in that column to filter channels with that tag.

Learn to [create channels](#) and [import](#) data.

Learn more about [ThingSpeak](#)

Examples

- [Arduino](#)
- [Arduino MKR1010](#)
- [ESP8266](#)
- [Raspberry Pi](#)
- [Netduino Plus](#)

Upgrade

Need to send more data?

Need to use ThingSpeak API?

[Upgrade](#)

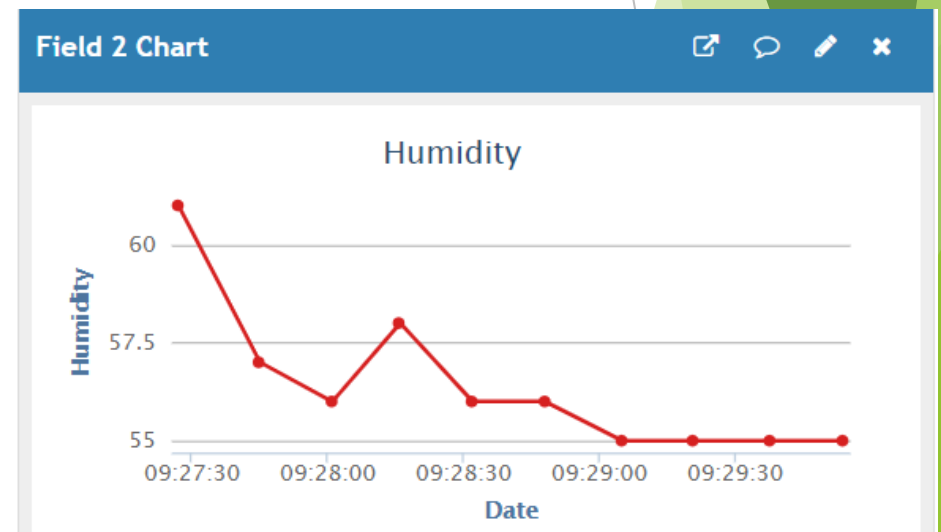
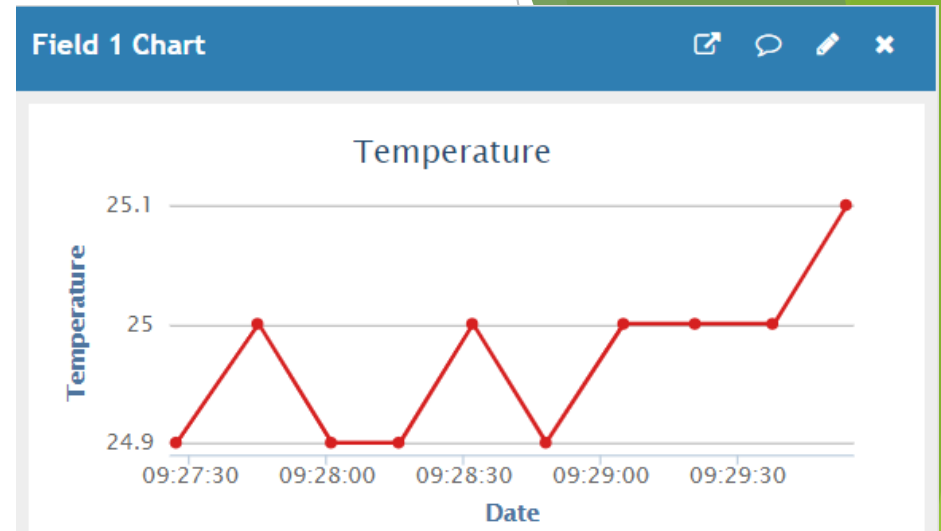
- ▶ Felhőalapú IoT fókuszú applikáció
- ▶ Diagramok kirajzolása
- ▶ További funkciók: MATLAB scriptek implementálása

Fejlesztői környezet

- ▶ Visual Studio Code:
 - ▶ Debugging
 - ▶ Linter
 - ▶ Kódkiegészítés: IntelliSense
 - ▶ Verziókövetés
 - ▶ Syntax highlight
- ▶ PlatformIO
 - ▶ Sok fejlesztői kártya választható
 - ▶ Könyvtárak kezelése egyszerűbb: letöltés, nyilvántartás

Funkcionalitás

- ▶ DHT11 és fotoellenállás csatlakoztatása az ESP8266 NodeMCU-ra
- ▶ Adatok küldése Thingspeak-re
- ▶ Diagramok rajzolása: API kulcs



Funkcionalitás

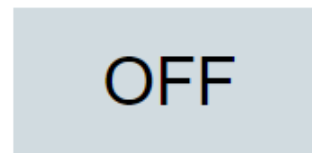
- ▶ Weblap létrehozása: HTML, CSS
- ▶ Két LED állapotának kijelzése, és vezérlése
- ▶ Fotoellenállás által mért érték 5-re normált értékének kijelzése

IoT server with ESP8266

First LED off



Second LED on



Photoresistor value, out of 5:

3.71