



M Ű E G Y E T E M 1 7 8 2

**Budapesti Műszaki és Gazdaságtudományi Egyetem**  
Villamosmérnöki és Informatikai Kar  
Elektronikai Technológia Tanszék

Kuglics Lajos Patrik

**LEVEGŐ SZENNYEZETTSÉG  
MÉRÉSE ALKALMAZOTT  
SZENZORIKAI ESZKÖZÖKKEL**

Tudományos Diákköri Konferencia Dolgozat

KONZULENS

**Dr. Géczy Attila**

BUDAPEST, 2020

# Tartalomjegyzék

<b>Összefoglaló .....</b>	<b>4</b>
<b>Abstract.....</b>	<b>5</b>
<b>1 Bevezetés .....</b>	<b>6</b>
1.1 Feladatkitűzés .....	6
1.2 Motiváció .....	6
<b>2 Irodalomkutatás.....</b>	<b>7</b>
2.1 A légszennyezés mérésének célja .....	7
2.1.1 AQI – Air Quality Index .....	8
2.1.2 COVID19.....	10
2.2 A szenzorok működése .....	11
2.2.1 Elektrokémiai szenzorok.....	11
2.2.2 Fotoionizációs detektorok.....	12
2.2.3 Optikai detektorok .....	12
2.3 Adatgyűjtő rendszerek .....	13
2.3.1 Asztali számítógép, notebook .....	13
2.3.2 Kis méretű kompakt számítógépek.....	13
2.3.3 Okostelefon.....	14
2.3.4 Mikrokontroller alapú rendszerek.....	14
2.3.5 Specializált elektronika.....	14
<b>3 Kísérletterv .....</b>	<b>16</b>
3.1 Első prototípus CCS811-gyel és SM-PWM-01C-vel .....	16
3.1.1 Használt eszközök.....	16
3.1.2 Mérési összeállítás .....	19
3.1.3 A szoftver elkészítése .....	20
3.1.4 Az adatok feldolgozása.....	22
3.1.5 Mérési eredmények laboratóriumban .....	27
3.1.6 Értékelés, folytatás.....	33
3.2 Android-kompatibilis hordozható eszköz .....	33

3.2.1 Célkitűzés.....	33
3.2.2 Felhasznált eszközök .....	34
3.2.3 Tesztelés.....	37
3.2.4 Tervezés .....	39
3.2.5 Programozás.....	42
3.2.6 Mérés .....	46
3.2.7 Összegzés.....	50
3.3 Fejlesztési tervek, piaci kitekintés .....	51
<b>4 Köszönetnyilvánítás .....</b>	<b>54</b>
<b>5 Irodalomjegyzék.....</b>	<b>55</b>

# Összefoglaló

Dolgozatom célja egy olyan, általam készített hordozható levegőminőség-mérő elektronika bemutatása, amely kis mérete, egyszerű használata, valamint okostelefonos kompatibilitása miatt alkalmas a mindennapi használatra.

A dolgozatom során egy rövid irodalomkutatás, a légszennyezettség mérésének főbb okainak, céljainak bemutatása után ismertetem a célra használt modern szenzorok fő típusait, működésüket, valamint az adataik kiolvasására és feldolgozására alkalmas lehetőségeket. Ezt követően megmutatom, hogyan készítettem el egy prototípust, valamint milyen méréseket készítettem vele, milyen nehézségekbe és problémákba ütköztem fejlesztése, használata során. A prototípust gyártási laborkörnyezetben validáltam, ezzel kapcsolatot teremtve a munkám és az Ipar4.0 elvárásainak megfelelő alkalmazott szenzorikai elvárásokra. A munkám következő szakaszában ezeket a nehézségeket, problémákat áthidalva megismertetem az olvasóval egy jól működő, enyhén kezdetleges, de viselhető és kifejezetten tanulságos információkat adó rendszer elkészítésének folyamatát, technikai megoldásait. Miután a rendszert urbánus környezetben okostelefonnal leteszteltem, kitekintést nyújtok a téma lehetőségeivel, jövőjével kapcsolatban.

Munkám során foglalkoztam C nyelvű, valamint Arduino programozással, Android szoftverfejlesztéssel, valamint elektronika tervezésével, forrasztásával is.

## **Abstract**

The goal of my work is to present a wearable air quality monitoring equipment, which is suitable for everyday use due to its small size, easy handling, and smartphone compatibility.

After showing the motives and purposes of measuring air quality, I'll present a short literature research, showing the main types of modern gas sensors used for this purpose, their means of operation, and the different options for acquiring sensor data, as well as for processing it. Later I'll present my work in making a prototype, the measurements I took with it, and the difficulties I encountered while developing and using it. I validated the prototype in a manufacturing laboratory environment, thus making a connection between my work and the standards set for applied sensorics by Industry 4.0. In the next section of this work, I'll show the development of the final system, and my solutions to tackling the problems I faced previously. In the end, my working, wearable equipment, and the instructive measurements taken using it are presented. After the system was tested in an urban environment using a smartphone, I'll show the possibilities and future development opportunities of this project.

During my work, I utilized C and Arduino programming, Android software development, and engineered electronic equipment.

# **1 Bevezetés**

## **1.1 Feladatkitűzés**

A célom egy elektronikus eszköz létrehozása volt, amely képes a levegőben található porszemcsék és gázok mérésére, ezen adatok feldolgozására, valamint továbbítására. Egy ilyen elektronika három fő részből áll. A mérést elvégző szenzorból, a szenzor adatait feldolgozó mikroprocesszorból, valamint az adatot továbbító kommunikációs modulból. Én ezt egy Arduino központú, szenzorokkal és egy Bluetooth modullal ellátott elektronikával oldottam meg.

## **1.2 Motiváció**

Életünk során a legtöbbet az egyik legtermészetesebb anyaggal érintkezünk: levegővel. Ez az anyag hordozza létünk egyik alapfeltételét, az oxigént. Azonban a levegő összetétele nem konstans, az oxigénen kívül láthatatlan kémiai és fizikai részecskék milliói vesznek minket körül. Ezek az anyagok legtöbbször semlegesek számunkra, de néhányuk kifejezetten veszélyes, egészségkárosító hatású is lehet. Az utcára lépve gyakran kipufogók füstje, föld pora, cigarettafüst szaga fogad minket. Ezen szennyező anyagok mérése elengedhetetlen segítség lehet, amennyiben képet akarunk kapni környezetünk tisztaságáról.

## 2 Irodalomkutatás

### 2.1 A légszennyezés mérésének célja

Egészségünk védelmének érdekében az egyik legfontosabb követelmény a jó minőségű levegő. A különböző égéstermékek ( $\text{CO}_x$ ,  $\text{NO}_x$ ), a szálló por, valamint a VOC-k ezt a minőséget nagyban rontják. Talán a legfontosabbak ezek közül a VOC-k: építőanyagok, háztartási eszközök, festékek mind szennyeznek vele a levegőt, amelynek következtében szem- és torokirritáció, fejfájás, szédülés lép fel [1]. Az egyik legveszélyesebb szennyező az MDF lapból készült bútorok ragasztóanyagából, valamint a bútorkokból párolgó formaldehid. Az egészségügyi határértékeket gyakran nagy mértékben meghaladó anyag többször is a figyelem középpontjába kerül [2]. Az otthonokban előforduló új bútorok a megengedett TVOC (Total VOC – összesített illékony szerves anyag) határérték 18-20-szorosát is okozhatják. Ezekből az adatokból látható, hogy a levegő szennyező anyagainak mérése nem csak ipari, de otthoni környezetben is nagyon fontos.

A kültér és beltér közötti viszonyt, valamint a legfontosabb mérési lokációkat a 2.1-es ábra tárgyalja. Ezek alapján megállapítható, hogy a kültér és beltér is fontos szempont, és a mindennapos élet lekülönbözőbb helyzeteiben is fontos lehet a levegő minőségének vizsgálata.



2.1 Ábra: Kültér és beltér – a legfontosabb lokációk mérés szempontjából

## 2.1.1 AQI – Air Quality Index

A kültéren mért légszennyezés jellemzéséhez sok helyen az AQI-t (Air Quality Index – levegőminőségi indexet) használjuk. Ennek több változata is létezik, de itt az Amerikai Környezetvédelmi Hivatal (EPA) által felállított mércét [3] fogom bemutatni, mivel ez az egyik leggyakrabban használt AQI az okos eszközökben. Ez a mérce 6 csoportra bontható: jó, elfogadható, egészségtelen az érzékeny csoportok számára, egészségtelen, fokozottan egészségtelen, veszélyes. Ezen fokozatok megállapításához a levegőben található ózon, szálló por, szén-monoxid, kén-dioxid, valamint nitrogén-dioxid mérésére van szükség. A felsorolt anyagok koncentrációjának időtartamra vett átlagának ismeretében beazonosíthatjuk a megfelelő légszennyezettségi csoportot, amelyből egy általánosított egészségügyi következtetést vonhatunk le. Ezeket a csoportokat, valamint a hozzájuk tartozó megállapításokat az alábbi ábra mutatja.

Jó ( 0-50)	Megfelelő légminőség, a légszennyezettség nagyon kevés veszélyt jelent/veszélytelen.
Elfogadható (51-100)	Elfogadható légminőség. Néhány különösen érzékeny csoport számára egészségügyi kockázatot jelenthet.
Egészségtelen az érzékenyebbek számára (101-150)	Érzékeny egyének esetén egészségügyi problémák léphetnek fel. A nagyközönségre általában nincs hatással.
Egészségtelen (151-200)	A nagyközönség egyes tagjai tüneteket mutathatnak. Az érzékeny csoportok tagjai komolyabb egészségügyi problémákat is észlelhetnek.
Fokozottan egészségtelen (201-300)	Az egészségügyi rendellenességek fellépésének kockázata mindenki számára magasabb.
Veszélyes (301-500)	Veszélyes egészségügyi problémák, a különböző betegségek fellépésének kockázata tovább nő.

2.2 Ábra: A légszennyezettségi index csoportjai, színei

Az EPA természetesen nagyobb részletességű irányelveket is megfogalmazott: ezek segítségével a különböző médiák könnyedén, pontosan jelenthetik a nagyközönség számára, hogy a jelenlegi légszennyezettség esetén mi a teendő az egyes csoportok számára. [4]



AQI kategóriák (index értékek)	Ózon (ppm)		Szálló por ( $\mu\text{g}/\text{m}^3$ )	
	8 órás időtartam	1 órás időtartam	PM <sub>2.5</sub>	PM <sub>10</sub>
			24 óra	24 óra
Egészségtelen az érzékenyebbek számára (101-150)	0.071-0.085	0.125-0.164	35.3-55.4	155-254
	Tüdőbetegségben szenvedők (pl. asztma), gyermekek, idős felnőttek, kültéren dolgozók, egyének bizonyos genetikai jellemzőkkel, valamint bizonyos tápanyagokkal szegény diétán élők számára ajánlatos a hosszú és/vagy megerőltető kültéri tevékenységek kerülése		Tüdő- és/vagy szívbetegségben szenvedők, idős felnőttek, gyermekek, valamint alacsonyabb társadalmi-gazdasági státuszú egyének számára ajánlatos a megerőltető és/vagy hosszabb időtartamú kültéri tevékenységek kerülése	
Egészségtelen (151-200)	0.088-0.105	0.165-0.204	55.5-150.4	255-354
	Tüdőbetegségben szenvedők (pl. asztma), gyermekek, idős felnőttek, kültéren dolgozók, egyének bizonyos genetikai jellemzőkkel, valamint bizonyos tápanyagokkal szegény diétán élők számára ajánlatos a hosszú és/vagy megerőltető kültéri tevékenységek kerülése; mindenki más számára ajánlatos a megerőltető és/vagy hosszú időtartamú kültéri tevékenységek korlátozása		Tüdő- és/vagy szívbetegségben szenvedők, idős felnőttek, gyermekek, valamint alacsonyabb társadalmi-gazdasági státuszú egyének számára ajánlatos a megerőltető és/vagy hosszabb időtartamú kültéri tevékenységek kerülése; mindenki más számára ajánlatos a megerőltető és/vagy hosszú időtartamú kültéri tevékenységek korlátozása	
Fokozottan egészségtelen (201-300)	0.106-0.200	0.205-0.404	150.5-250.4	355-424
	Tüdőbetegségben szenvedők (pl. asztma), gyermekek, idős felnőttek, kültéren dolgozók, egyének bizonyos genetikai jellemzőkkel, valamint bizonyos tápanyagokkal szegény diétán élők számára ajánlatos minden kültéri tevékenység kerülése; mindenki más számára ajánlatos a kültéri tevékenységek korlátozása		Tüdő- és/vagy szívbetegségben szenvedők, idős felnőttek, gyermekek, valamint alacsonyabb társadalmi-gazdasági státuszú egyének számára ajánlatos minden kültéri fizikai tevékenység kerülése; mindenki más számára ajánlatos a megerőltető és/vagy hosszú időtartamú tevékenységek korlátozása	

### 2.3 Ábra: Az EPA ajánlásainak egy részlete

Az AQI kiszámításához egy függvényt használhatunk, amely a mért érték csonkított közelítésének segítségével, valamint az AQI táblázat adataiból megadja a megfelelő indexet. Ezt minden egyes megfigyelt értékre el kell végezni, majd a legmagasabb AQI-t kell kiválasztani. Ahol 8 órás és 1 órás értékeket is lehet számolni, ott mindig a magasabbat kell kiválasztani. Fontos, hogy az AQI nem egy pillanatnyi érték: számolását mindig napi szintű mérésekre, előrejelzésekre kell alapozni.

$$I_p = \frac{I_{Hi} - I_{Lo}}{BP_{Hi} - BP_{Lo}} (C_p - BP_{Lo}) + I_{Lo}$$

### 2.4 Ábra: Az AQI kiszámításához használt függvény

- $I_p$  = a „p” szennyezőanyag AQI indexe
- $C_p$  = a p szennyezőanyag mért koncentrációja, csonkítva
- $BP_{Hi}$  = a koncentráció töréspontja, amely nagyobb vagy egyenlő mint  $C_p$
- $BP_{Lo}$  = a koncentráció töréspontja, amely kisebb vagy egyenlő mint  $C_p$
- $I_{Hi}$  = a  $BP_{Hi}$  töréspontához tartozó AQI érték
- $I_{Lo}$  = a  $BP_{Lo}$  töréspontához tartozó AQI érték

## 2.1.2 COVID19

A levegőben található részecskék kapcsán természetesen nem szabad elfeledkezni a legnagyobb globális aktualitásról: a koronavírusról, valamint annak levegőben való terjedéséről. Erről a témáról több, gyakran ellentmondásos tanulmány is készült [5] [6].Néhány kutató óvatosabb a témával kapcsolatban, míg egyesek szerint a levegő szennyezettsége és a COVID19 súlyossága, halálozási rátája szoros kapcsolatban áll. Comunian és kollégáinak munkája [7] megmutatta, hogy a szálló por önmagában elég a tüdő roncsolásához, amely gyulladásokhoz, oxigénhiányhoz vezethet, így hozzájárulva a COVID19 által okozott tünetek súlyosbításához. Azonban azt is kijelentették, hogy eredményeik finomításához sokkal több adatra, szisztematikus kutatásokra, valamint globális méretre kiterjesztett mérőhálózatra van szükség.

A legfrissebb tanulmányok (2020 október) alapján Wuhanban a járvány csúcsa alatt a PM2.5 és PM10 (2.5 és 10 mikronos átmérőjű részecskék) napi koncentrációja hasonlóságokat mutatott a vírus okozta napi halálozási aránnyal. Ezek a hasonlóságok jól megfigyelhetők a görbék időbeli változásain 55[8]. 49 másik kínai városnál is ugyan erre a konklúzióra jutottak [9]. Yongjian [10][10] ezt a kutatást kiterjesztette 120 városra, és a szennyezőanyagok általánosított lineáris modelljének felhasználásával bebizonyították, hogy a PM2.5, PM10, NO<sub>2</sub>, O<sub>3</sub> szintjei összeköthetők a fertőzések számával. Az említett szennyezőanyagok koncentrációjának 10µg/m<sup>3</sup>-os növekedése sorrendben 2.24%, 1.76%, 6.94%, valamint 4.76%-os növekedést hozott napi szinten. Wu [11] 3000, Amerikai Egyesült Államokbeli megye adataival dolgozott, amelyből hasonló trendeket mutatott ki: 1µg/m<sup>3</sup>-es PM2.5 koncentrációnövekedés 8%-os vírus miatti halálozási aránynövekedéssel járt. A téma európai szála Franciaországra, valamint a fosszilis üzemanyag alapú szállítási szektorra fókuszált. Magazzino gépi tanulás alapú kutatása [12] megmutatta, hogy egy bizonyos részecskekoncentráció elősegítheti a COVID19 terjedését, valamint növeli a légzőrendszer hajlamát a megfertőződésre.

Érdemes megemlíteni a járványügyi helyzet hatását a szennyezőanyagokra: a vírus miatti lezárások miatt sok helyen pozitívan változott a levegő minősége. A PM2.5-ös koncentráció átlagosan 31%-os csökkenést mutatott [13].

## **2.2 A szenzorok működése**

A munkában használt szenzorokat két fő részre oszthatjuk: az illékony szerves kémiai vegyületeket és a szálló részecskéket mérő eszközökre. Míg a kémiai anyagoknál az összetételük, vagy a koncentrációjuk a legfőbb tulajdonságuk, amelyet meg szeretnénk mérni, addig a szálló porok, füstök esetén inkább a méretük és azok eloszlása hordoz több információt. Ezért a kémiai anyagokat valamilyen kémiai reakció segítségével mutatjuk ki, a szálló részecskék (PM – particulate matter) pedig optikai eszközökkel mérhetőek. A gázok összetételének mérésére több mód is elterjedt, azonban itt most csak az integrált áramkörökkel, egyszerű elektronikákkal megvalósítható szenzorokról esik szó, így a gázkromatográfiában használt lángionizációs detektor, a hővezetőképességi detektor, vagy az elektronbefogásos detektor csak említés szintjén jelenik meg [14].

### **2.2.1 Elektrokémiai szenzorok**

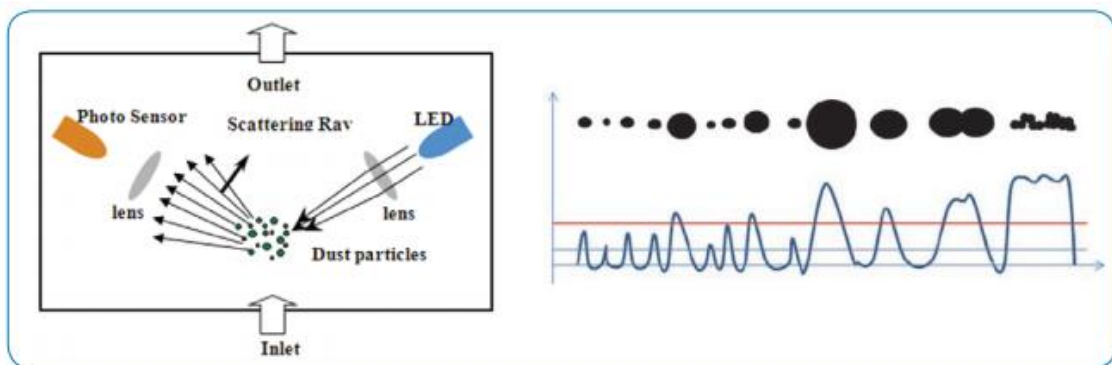
A levegő részecskéi folyadékban található elektródák segítségével mérhetőek meg. A szennyező anyag oxidációjának/redukciójának következtében mérhető áram generálódik/megváltozik az érzékelő ellenállás mértéke. Legtöbbször két elektróda segítségével mérhető egy anyag jelenléte: egy érzékelő elektróda segítségével, amelyen a kémiai folyamatok lejátszódnak, valamint egy referencia elektródával, amely az érzékelő elektróda konstans potenciáljáért felel [15]. Az ilyen típusú detektorokat legtöbbször elő kell melegíteni a mérést befolyásoló vékony H<sub>2</sub>O filmréteg miatt, valamint a specifikált hőmérsékleti tartomány elérésének érdekében [16].

## 2.2.2 Fotoionizációs detektorok

A detektorban nagy energiájú fotonok (általában UV tartomány) részekre bontják a szennyező gázok molekuláit. A kiváló elektronok elektromos áram formájában jelennek meg a kimeneten. A fotonok energiáját változtatva állítható az érzékelt gázok tartománya. A fotoionizációs detektort jó linearitás, széles érzékelési tartomány (főleg szerves anyagok), valamint akár pikogramm felbontás jellemzi, azonban a mintában jelen lévő CO<sub>2</sub> csökkenti a hatékonyságát. A lángionizációs detektor is ezeken az alapokon működik, azonban ott az ionizáló energiát fotonok helyett hő adja át.

## 2.2.3 Optikai detektorok

Az általános optikai elven működő műszerben egy párhuzamosított fényű forrás van, amelynek sugarai elé a minta kerül. A minta szennyező anyagait kétféleképpen mérhetjük: vagy a sugár és a minta vonalában elhelyezett detektorral, amely a részecskék által csökkentett intenzitású fényt méri (direkt), vagy pedig a sugár útján kívül elhelyezkedő detektorral, amely a részecskékről szóródó fényt érzékeli (light scattering). <1 µm-es részecskéknél szinte kizárólag szórt fényt mérünk. A levegőmozgásért legtöbbször egy fűtő ellenállás felel: a felmelegített levegő felfele száll, nyomáskülönbséget okozva, így a műszer belsejében folyamatos légmozgás alakul ki. Egy ilyen érzékelő látható a 2.3 ábrán.



2.5 Ábra: Az optikai detektor működése különböző méretű részecskék esetén [17]

## **2.3 Adatgyűjtő rendszerek**

A szenzorok által mért adatokat valamilyen eszköz segítségével rögzítenünk kell, hogy később kiértékelhessük őket. Erre több lehetőségünk is van: használhatunk operációs rendszerrel rendelkező számítógépet, mikrokontroller alapú rendszereket, vagy akár erre a célra specializált elektronikát is.

### **2.3.1 Asztali számítógép, notebook**

Számítási kapacitásuk ezeknek a legnagyobb, azonban nagy méretük miatt nem integrálhatóak egy kézi műszerbe. Bár a szenzoradatok kiolvasásához nem optimálisak, az eredmények előállításához, tárolásához, további munkálatokhoz rengeteg eszközzel és tárhellyel rendelkeznek.

### **2.3.2 Kis méretű kompakt számítógépek**

Ide tartozik a Raspberry Pi, valamint például a PC/104. Ezek az eszközök operációs rendszerrel rendelkeznek (Linux), kis fogyasztásuk és kompakt méretük ellenére jó számítási kapacitásuk van.

#### **2.3.2.1 Raspberry Pi**

A Raspberry Pi egy körülbelül 6x10cm-es, kártya méretű számítógép. Fő tulajdonságai az alacsony ára, kis mérete, jó számítási kapacitása, valamint a hozzá kapható rengeteg ingyenes online segédanyag. Elterjedt az oktatásban és a hobbi projektek készítői között is.

#### **2.3.2.2 PC/104**

A PC/104 egy szabványos méretű és uniform buszon működő termékcsalád. Körülbelül 9x10cm-es, egymásra helyezett, összecsavarozható panelekből áll. Robosztus felépítése, kompakt mérete miatt az űrkutatásban is használatos. Egy CPU-kártyából, és a hozzá csatolt különféle periféria modulokból áll.

### **2.3.3 Okostelefon**

Ezek az eszközök ideálisak egy ilyen projekt esetén: vezetékes és vezeték nélküli kommunikációra is alkalmasak (USB, Bluetooth, WiFi, NFC), számítási kapacitásuk magas, méretük alapján (a mai trendek ellenére) pedig kompaktnak tekinthetők. Két fő csoportjuk az Android és iOS alapú rendszerek.

### **2.3.4 Mikrokontroller alapú rendszerek**

A mikrokontroller alapú rendszerek egy központi mikrokontrollerből, valamint a hozzá kapcsolt perifériákból áll. Ilyen rendszerek az Arduino, a Silabs Gecko sorozata, vagy a LaunchPad. Ezek az eszközök nagyon hasonlóak a kártyaszámítógépekhez, azonban nem operációs rendszert futtatnak. Egyre több ilyen termék támogatja az USB-n keresztüli programozást, gyakran azonban SPI-on, UART-on lehetséges csak a kommunikáció.

#### **2.3.4.1 Arduino**

Az Arduino egy open-source platform, amely könnyen használható hardverből és magas szintű szoftver-könyvtárakból áll. Eredete az olaszországi IDII nevű akadémiai programhoz köthető, ahol a tanulók könnyen használható elektronikus eszközöket szerettek volna létrehozni [18]. A mai Arduino elterjedt a hobbiprojektek készítői, valamint az elektronikát tanulók körében könnyű használhatósága és kedvező ára miatt. A hardverek fő eleméül az Atmel gyártó ATmega mikrokontroller termékcsoportja szolgál. Közös jellemzőjük a nagy kivezetésszám, sok GPIO láb, valamint pár integrált A/D és D/A átalakító. A PCB-n előre illesztett LED-eket és nyomógombokat helyeznek el a könnyű használat érdekében.

### **2.3.5 Specializált elektronika**

A „könnyebben” olvasható (pl. feszültség szint, ellenállás) szenzoradatokat egy órajel segítségével mikrokontroller nélküli rendszerben is tárolhatjuk. Ezek azonban bonyolult, komplex rendszerek, valamint egyetlen szenzorhoz illeszthetőek. SPI-on, UART-on, vagy egyéb kommunikációs interfészen dolgozó szenzorokat nem

illeszthetünk hozzá. A mikrovezérlők alacsony ára miatt alkalmazásuk már nem tekinthető költséghatékony megoldásnak.

## 3 Kísérletterv

### 3.1 Első prototípus CCS811-gyel és SM-PWM-01C-vel

Első célkitűzésként egy olyan eszközt szerettem volna készíteni, amely képes kvalitatív információt nyújtani egy helyiség levegőjéről. Ehhez szükségem volt szenzorokra, a szenzoradatokat feldolgozó hardverre, a feldolgozott adatokat fogadó eszközre, a programozáshoz szükséges szoftverre, valamint az elektromos és/vagy mechanikai kapcsolatot kialakító kiegészítőkre.

#### 3.1.1 Használt eszközök

##### 3.1.1.1 Szenzorok

A mérésekhez szálló részecskék, valamint gázok érzékelésére 2 szenzort alkalmaztam. Ezek főbb tulajdonságait az alábbi táblázat tartalmazza:

3.1 Táblázat: Az első prototípusban használt szenzorok

Gyártó	Termék	Mért összetevő	Érzékenység (por/gáz)	Detektor típusa	Kimenet
Amphenol	SM-PWM-01C	por	>1 $\mu\text{m}$ – 0.01-3000 $\mu\text{g}/\text{m}^3$	fényszóródás	PWM (x2)
Adafruit	CCS811	VOC, eCO <sub>2</sub> <sup>[1]</sup>	eCO <sub>2</sub> :400-8192 ppm TVOC: 0-1187 ppm	elektrokémiai	I <sup>2</sup> C

---

<sup>1</sup> A CCS811 adatlapja szerint az eCO<sub>2</sub> a mért TVOC-ból számított becslés a levegő CO<sub>2</sub> tartalmára, amennyiben a TVOC főleg emberi leheletből származik.



Itt pedig azok a szenzorok szerepelnek, amelyek még alkalmasak lettek volna a feladatra, azonban áruk/komplexitásuk miatt nem kerültek kipróbálásra:

**3.2 Táblázat: A prototípushoz figyelembe vett szenzorok**

Gyártó	Termék	Mért összetevő	Érzékenység (por/gáz)	Kimenet
Amphenol	MiCS-4514	CO, NO <sub>2</sub> , NH <sub>3</sub> , CH <sub>4</sub> stb.	változó <sup>[2]</sup>	ellenállás (x2)
Omron	B5W-LD0101-1	por	>0,5 µm - 1µg/m <sup>3</sup>	feszültségimpulzus (x2)
Bosch	BME680	pára/nyomás/hőm./ VOC	ΔR – 0.08%	I <sup>2</sup> C, SPI
Sharp	GP2Y1010AU0F	por	0-1 mg/m <sup>3</sup>	feszültség

A táblázatokon jól látható, hogy az egy csoportba tartozó szenzorok között a lényegi különbség az érzékenység és a kimenet típusa. A B5W-LD0101-1 és a MiCS-4514 2-2 kimenettel rendelkezik. Az előbbinél ez külön kimenetet jelent a PM2.5 és a PM10 osztályok számára. Utóbbinál az oxidáló és a redukáló gázok számára van 1-1 kimenet. Ez a plusz információ sokat segíthet a levegő minőségének értékeléséhez, mivel egy adott szállópor koncentrációnál a magasabb PM2.5 tömegszázalék sokkal egészségtelebber, mint a magas PM10 arány, a redukáló és oxidáló gázokat megkülönböztetve pedig pontosabb becsléseket tehetünk a levegő összetételére a hiányzó mérések ellenére.

---

<sup>2</sup> A szenzor minden összetevőre más tartományban érzékeny, lásd:

[https://eu.mouser.com/datasheet/2/18/0278\\_Datasheet-MiCS-4514-rev-16-1144833.pdf](https://eu.mouser.com/datasheet/2/18/0278_Datasheet-MiCS-4514-rev-16-1144833.pdf)

### **3.1.1.2 Szenzoradat-feldolgozó eszköz**

A szenzoradatok feldolgozását egy Arduino Nano végezte. Ez egy 5V-os tápfeszültségű, 16MHz-es, 22 digitális I/O lábbal rendelkező panel. Az I/O lábakból 8 használható analóg bemenetként, 6 pedig hardveres PWM-re képes. Az Arduino központi magját képező Atmega328P-t USB interfészen programozhatjuk a hozzá illesztett USB<->soros kommunikáció chip segítségével, valamint SPI programozásra is van lehetőség.

### **3.1.1.3 Adatrögzítő eszköz, kommunikáció, tápellátás**

Az adatok rögzítésére egy Windows operációs rendszerrel rendelkező notebook-ot használtam. Az eszköz méretéből adódóan relatíve könnyen tudtam azt a különböző mérések helyszínére vinni, azonban súlya és sérülékenysége meglehetősen sok elővigyázatosságot követelt meg. A tápellátást, valamint az Arduino Nano és az operációs rendszer közötti kapcsolatot egy USB Mini-B – USB type A típusú kábelen folyó soros kommunikáció valósította meg.

### **3.1.1.4 Programozó felület**

A programozáshoz az Arduino hivatalos szoftverét, az Arduino Software 1.8.10-es verzióját használtam. Ez a program rengeteg könyvtárral, driverrel, és példaprogrammal van felszerelve a gyors és kényelmes kódolás érdekében. A beépített soros port monitor és plotter segítségével pedig könnyen ellenőrizhető a Nano által kiértékelt szenzoradat szöveges és/vagy grafikus felületen is.

Egy eszköz programozásához elég a portot, valamint a panel típusát kiválasztani, a használt könyvtárak ezek alapján automatikusan kiválasztják az utasításainak megfelelő regisztereket, értékeket. Ennek köszönhetően a legtöbb alkalmazáshoz elég a beépített, magas szintű funkciókat használni, nincs szükség az AVR részletes ismeretére.

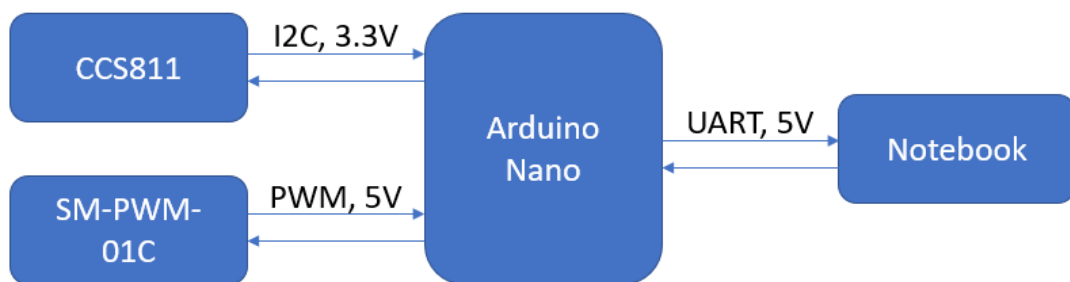
### **3.1.1.5 Egyéb eszközök**

A prototípus összeszereléséhez a legfontosabb eszköz az úgynevezett breadboard volt. Ez egy olyan panel, amelyen forrasztás nélkül, könnyen hozható létre

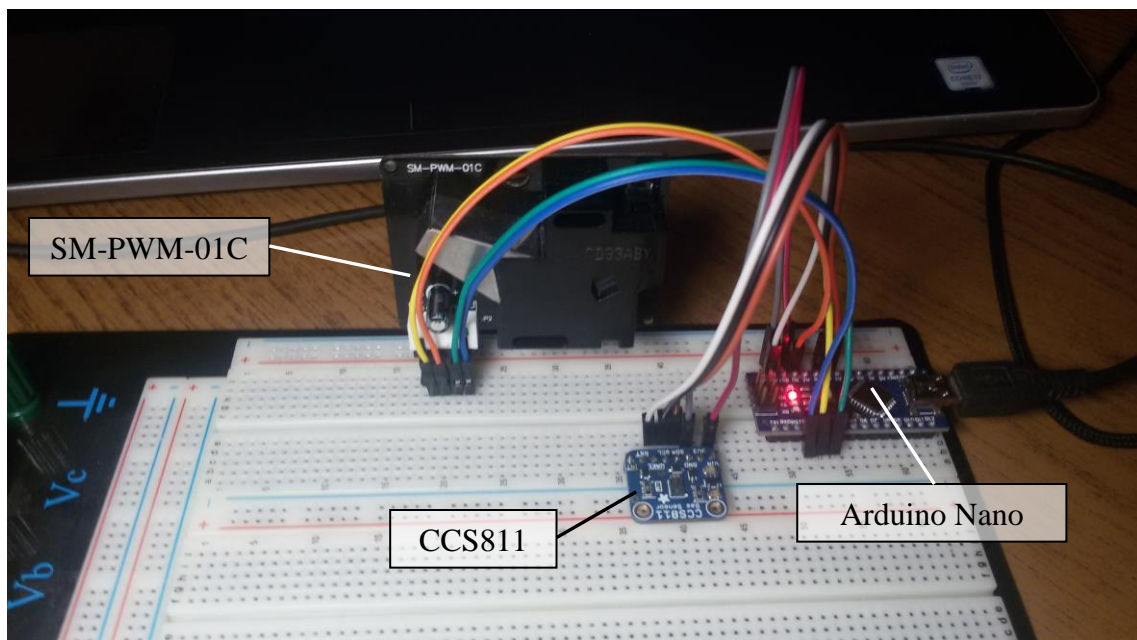
elektromos összeköttetés. A breadboardhoz természetesen még különböző hosszúságú illesztőkábeleket is használtam.

### 3.1.2 Mérési összeállítás

Az elektronikát egy breadboardon állítottam össze, így gyorsan tudtam rajta kisebb-nagyobb változtatásokat eszközölni, valamint így nem volt szükség se NYHL-re, se forrasztásra. A mérési összeállítás terve és megvalósítása a 3.1-es, valamint a 3.2-es ábrán látható.



3.1 Ábra: A mérési elrendezés terve



3.2 Ábra: A megvalósítás

Mindkét eszköz az 5V lábról kapja a tápellátást, a földjük a GND-hez csatlakozik. A CCS811 I<sup>2</sup>C-n kommunikál, az Arduino Nano dokumentációja szerint ehhez a protokollhoz az A4 (SDA) és az A5 (SCL) láb tartozik. Az SM-PWM-01C esetében digitális jelként kezelve a PW modulációt a Nano D2 és D3 lábát használtam. A CCS811 esetén még egy  $\sim$ WAKE jelet is földre kellett húzni, hogy az eszköz működjön. Az SM-PWM-01C helyes működéséhez el kellett zárni az oldalán található nyílást, amelyet a könnyű tisztíthatóság érdekében hagytak nyitva a gyártás során. Ehhez egy fekete szigetelő szalagot használtam, a fényvisszaverődés megakadályozása érdekében.

### 3.1.3 A szoftver elkészítése

Az eszköz működéséhez két fő feladatot kellett ellátnia a Nano-nak: az adatok begyűjtése a szenzoroktól, és ezek formázott, könnyen felhasználható továbbküldése. A CCS811 utasításait egy könyvtárban elérhetővé tették, így a megfelelő header bevonásával magas szintű nyelven könnyen programozható a szenzor [19].

Alább a legfontosabb kódrészleteket, funkciójukat, működésüket magyarázom el.

```
void setup() {
  Serial.begin(115200);
  if(!ccs.begin()){
    Serial.println("Kommunikacios hiba!");
    while(1);
  }

  Timer1.initialize(250);
  Timer1.attachInterrupt(timerIsr);

  while(!ccs.available());
  float temp = ccs.calculateTemperature();
  float reference = 21.0;
  ccs.setTempOffset(temp - reference);
}
```

Első lépésben a soros kommunikációt állítottam be. Itt a lehető leggyorsabb átvitelre törekedtem, mivel a 4kHz-es mintavételezéshez szükség volt a magas kommunikációs sebességre. Ezt 115200 Baud sebességre állítottam be, itt elegendő

gyorsaság mellett még teljesen stabilan működött. A 'ccs.begin()' funkcióval inicializáltuk a szenzort, hiba esetén kiléptünk. Ezután megadtuk a referencia hőmérsékletet, mivel a CCS811 csak relatív hőmérsékletmérésre képes.

```
for (int i=0;i<4;i++) {
    ccs.readData();
    tmp += ccs.calculateTemperature();
    co2 += ccs.getCO2();
    voc += ccs.getTVOC();
    delay(1000);
}
```

A CCS811 másodpercenként feljegyezte a 3 mért változót, majd 4 mérésből átlagolt, így csökkentve az esetleges rendellenes, kiugró értékek befolyását a kiértékelésnél.

```
void timerIsr() {
    bool pin2 = PIND2;
    bool pin3 = PIND3;
    Serial.print(pin2);
    Serial.print(",");
    Serial.println(pin3);
}
```

Az SM-PWM-01C 4kHz-es mintavételezéséhez egy timer-t állítottam be, így a két szenzor adatait két külön szálon tudtam elintézni. Mivel az Arduino beépített függvényei egyszerű használhatóságuk ellenére lassúak, ezért a gyakori futás miatt kénytelen voltam egy kevés AVR kódot használni. Az időzítéshez egy előre megírt könyvtárat használtam fel, a TimerOne nevű open-source projektből [20].

A mért adatokat végül soros porton küldtem tovább a számítógép felé.

```
cli();
Serial.print("T;");
unsigned long timestamp = millis()/1000;
Serial.print(timestamp);
Serial.print(";");
Serial.print(co2);
Serial.print(";");
Serial.print(voc);
Serial.print(";");
Serial.println(tmp);
sei();
```

A 'cli()' és 'sei()' utasítások az interrupt tiltását és engedélyezését takarják. Ezek nélkül, ha futás közben az UART-kommunikáció során egy interrupt érkezik, a két szenzor adatai összekeveredhetnek és/vagy kerethiba léphet fel a kommunikációban. Ezért feltétlen szükséges volt az adatblokk védelme. Enélkül az alábbi és ehhez hasonló hibák jelentek meg a soros kommunikációban:

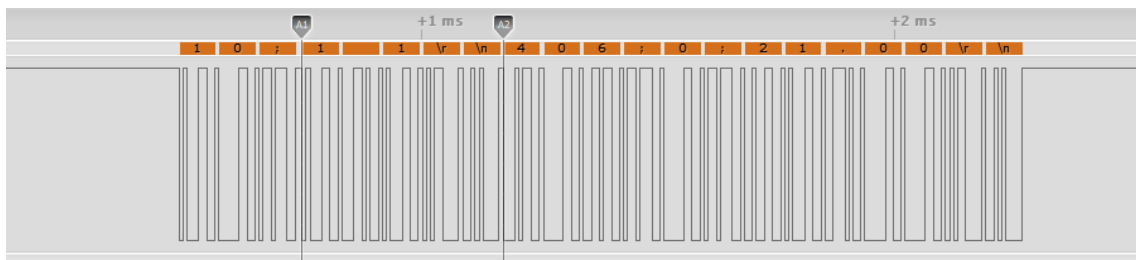
Elvárt adatok: 10;406;0;21.00(\r\n)

1;1(\r\n)

Valójában: 10;1;1(\r\n)

406;0;21.00(\r\n)

Ezt a hibát a 3.3-mas ábrán láthatjuk: az A1 és A2 jelzők közé zárt rész az SM-PWM-01C adatai, míg a többi a CCS811-hez tartozik.



**3.3 Ábra: Hiba a soros kommunikációban**

Mivel a két szenzor adatait külön szeretném venni, ezért a CCS811 adatai elé egy 'T' karaktert helyeztem, így a feldolgozás során könnyebben tudtam azokat azonosítani, kiválasztani.

### **3.1.4 Az adatok feldolgozása**

#### **3.1.4.1 Az adathalmaz szétbontása**

A legelső feladat az érkező adatok fogadása PC-n. Windows operációs rendszeren ezt a Parancssorból könnyen megoldhatjuk:

```
type COMx >> adatok.txt (x a port száma)
```

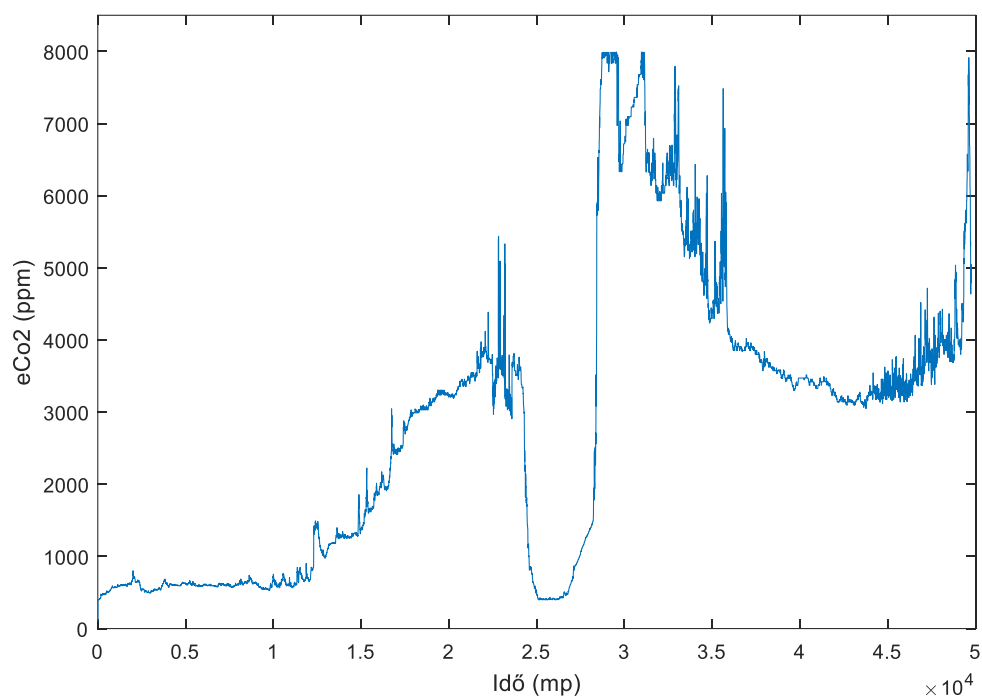
Az elkészült 'adatok.txt' fájlt szét kell válogatnunk. A fő segítségünk a sorkezdő T karakter. Először minden ilyen sort egy külön fájlba írunk, majd azokat a sorokat mentjük ki, amelyek nem tartalmazzák. Parancssorban:

```
findstr T adatok.txt >> ccsadat.txt  
findstr /v T adatok.txt >> smadat.txt
```

Ezzel a két sorral gyorsan és egyszerűen kinyerhetőek a kívánt adatok. Az én esetemben ezzel a módszerrel egy 546 Mb-os szöveges fájlt (amely körülbelül 14 óra alatt keletkezett a másodpercenkénti ~4000 beérkező adat hatására) körülbelül 1 perc alatt szétbontottam. Erre a feladatra egyetlen ismertebb szövegszerkesztő sem volt már képes.

#### **3.1.4.2 Az adatok kirajzolása**

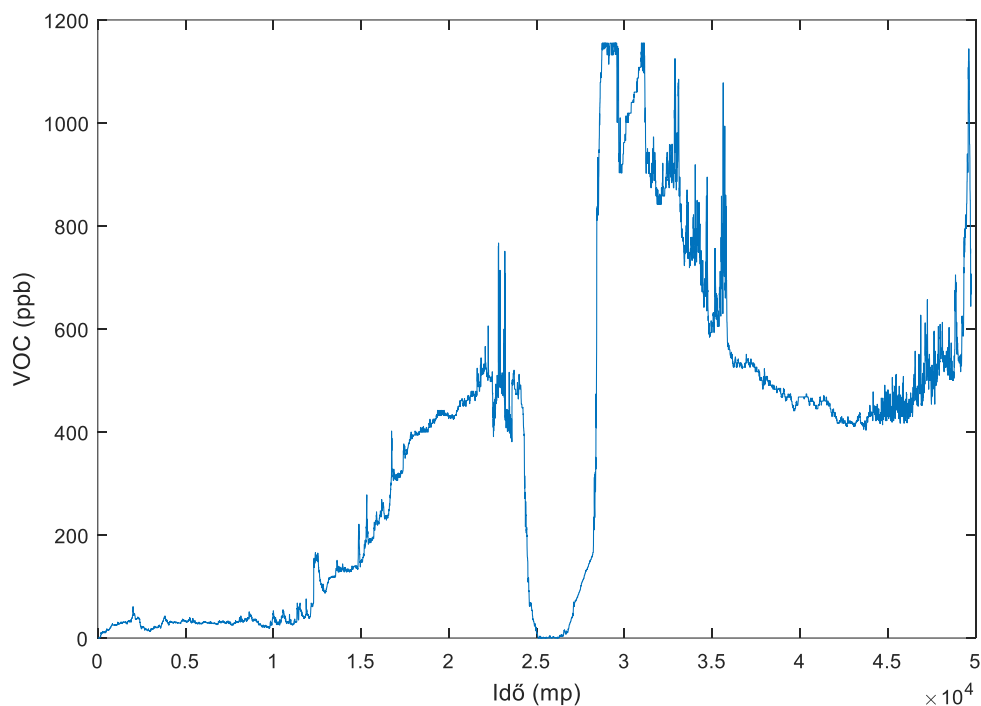
A vizuális megjelenítéshez MATLAB-ot használtam. Itt pár parancs ismeretében könnyen hozhatunk létre elegáns ábrákat. A programot elindítva az 'Import Data' gombra kattintva a forma megadásával vektorokba rendezve megkapjuk a mért értékeket. Ezután a 'plot(x,y)' függvényt használva, megkapjuk a grafikus ábrázolást. A VOC, valamint az ebből számolt eCO<sub>2</sub> értékek a 3.4-es és 3.5-ös ábrán láthatóak.



**3.4 Ábra: eCO<sub>2</sub> alakulása egy szobában 14 óra alatt (18 óra – 08 óra)**



VOC:



**3.5 Ábra: VOC szint egy szobában 14 óra alatt**

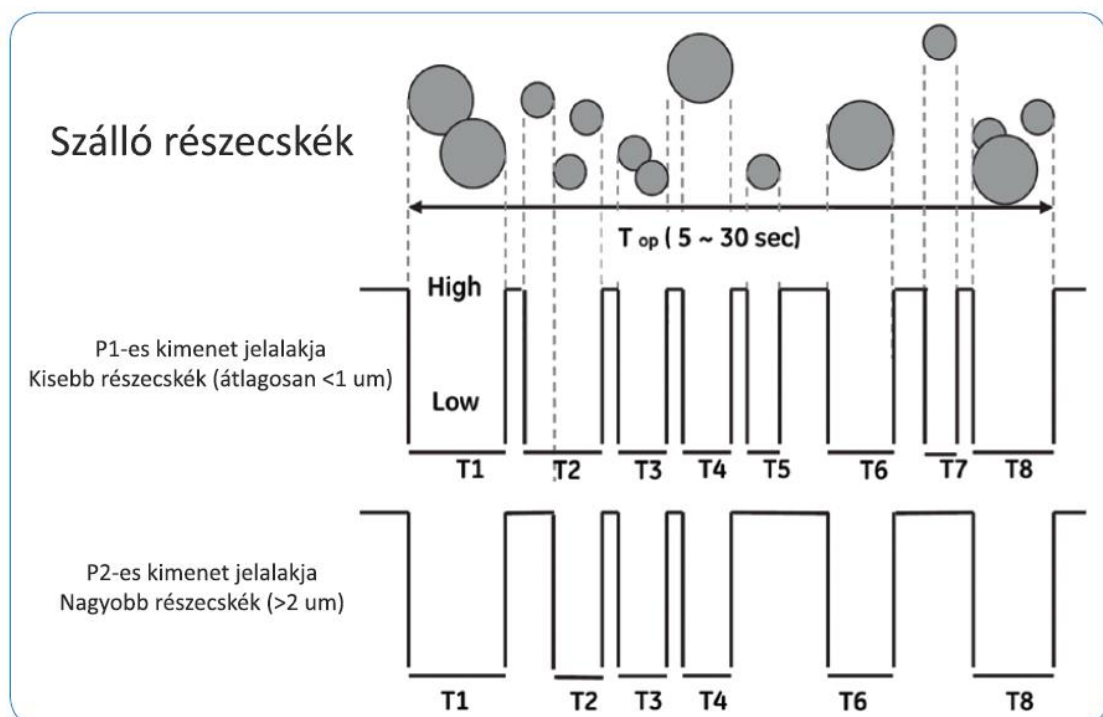
Jól látható, hogy a levegő minőségének romlása után egy ablaknyitás (2.5-ös időjelzés) milyen drasztikusan lecsökkenti a VOC-k mértékét a levegőben. A bezárása 4, a szobában tartózkodó személy esetén azonban ugyan ilyen hirtelen megnöveli ezt a szintet. Ez a mérés éjszaka készült, a légmozgás hiánya, valamint a teljesen bezárt ablakok okozhatják a hirtelen megugrást (körülbelül 10 perc alatt). Az érzékelő csak 1187 ppb-ig tud mérni, ezért itt a valódi szintet nem láthatjuk. A hirtelen változások valószínűleg torzítják a mérési eredményeket, ezért ezen tényezők miatt ez a mérés nem számít mérvadónak, viszont a történésekkel konzisztens, így minőségi megállapításokat tehetünk segítségével.

Az SM-PWM-01C értékeinek kirajzolása előtt ellenőriztem, hogy az Arduino mintavételezési frekvenciája elegendő-e a kívánt feladatra. Ehhez egy 24MHz-es, 8 csatornás Saleae USB logikai analizátort használtam 2MS/s beállításban. A két eszköz között nem találtam eltérést, ahogyan azt a 3.6-os ábra is mutatja.



3.6 Ábra: Az Arduino Nano és egy logikai analizátor mérése

Az adatlap szerint a szenzor két kimenete (egy kimenet a kisebb, egy a nagyobbak számára) alapján lehetséges megállapítani, hogy mekkora porszem hatására adott jelet az eszköz. Ezt a 3.7-es ábra vizualizálja.



3.7 Ábra: A porszemcsék méretének megkülönböztetése

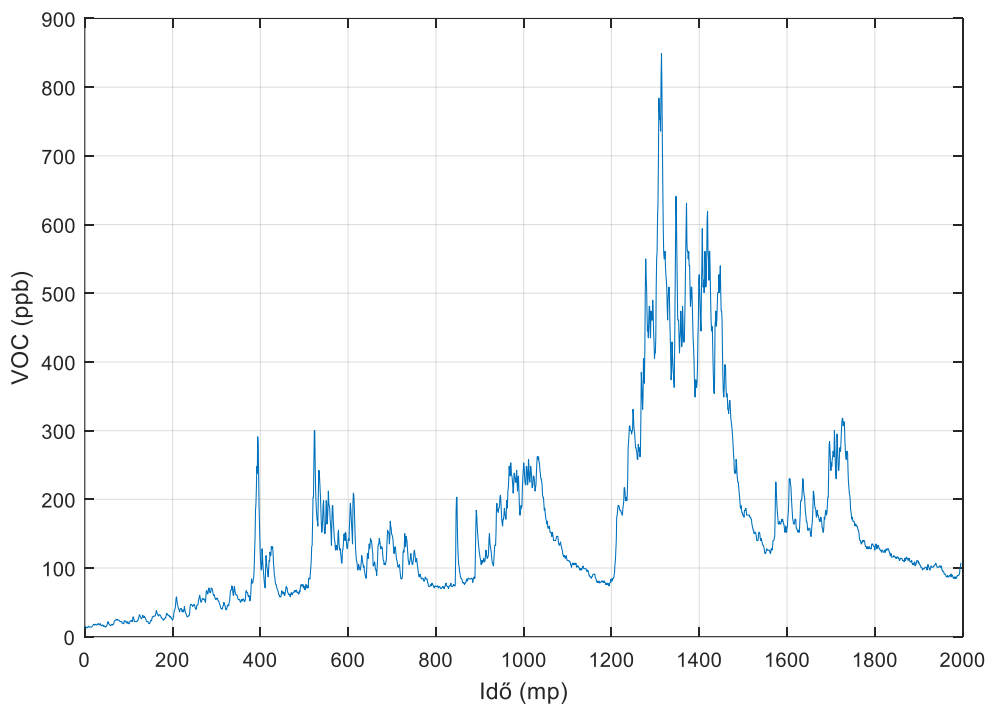
Ezzel ellentétben többször is előfordult, hogy a kisebb részecskéket érzékelő detektor nem jelzett, míg az érzékenyebb (tehát kicsi és nagy szemcséket is érzékelő) detektor észlelt valamit. Ez miatt a P2-P1 = 'kicsi részecske' számítás nem mindig működik.

### **3.1.5 Mérési eredmények laboratóriumban**

A CCS811 teszteléséhez a V1 alagsori laboratóriumban egy kis méretű gőzfázisú kemence, valamint egy szelektív forrasztó közelében végeztem méréseket azok működtetése közben. A továbbiakban csak a mért VOC értékek grafikus ábrázolását fogom feltüntetni, mivel az eszköz által számolt eCO<sub>2</sub> értékek csak emberi leheletről származó VOC-k esetén tekinthetők mérhető adatnak.

#### **3.1.5.1 Gőzfázisú forrasztó kemence**

Elsőként a gőzfázisú forrasztó kemencénél mértem: a szenzor a kemence tetején, annak nyílásától körülbelül 10 centiméterrel helyezkedett el. A mérés elején a bekapcsolt kemence felfűtési szakaszának, majd a felfűtés után a nyílást takaró ajtó nyitásának hatását szerettem volna feltérképezni. A 3.8-as ábrán látható a teljes folyamat.

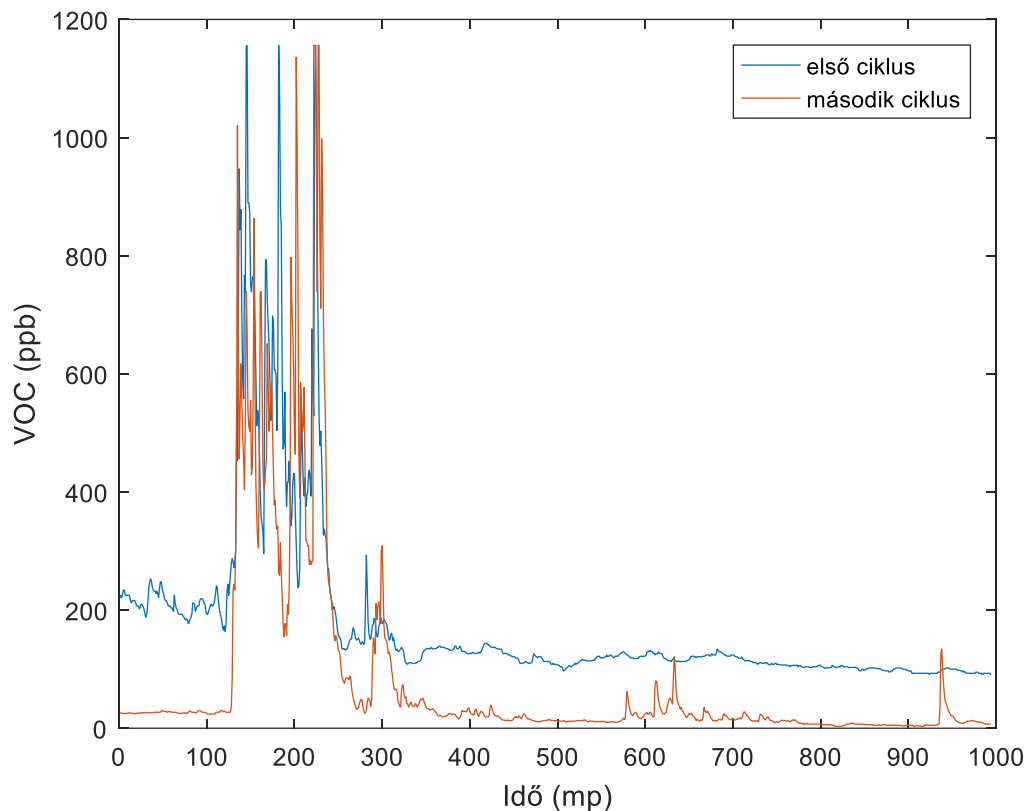


**3.8 Ábra: Mérés kis méretű gőzfázisú forrasztó közelében**

Jól látható, hogy a mért VOC lassan emelkedett a fűtés közben, majd a 400 másodperces jelnél az ajtó nyitására körülbelül 300 ppb-re ugrik fel. Ez az ajtónyitás az 520, 900, és 1700-as időpillanatokban is megfigyelhető. Az 1200-tól 1500-ig terjedő csúcsot a forrasztási térből direkt a szenorra eresztett Galden gőz okozta.

### **3.1.5.2 Szelektív forrasztó kemence**

Itt két teljes forrasztási ciklust is ki tudtam mérni. A szenzor a kemence szélén, a forrasztott NYHL-el körülbelül egy szinten helyezkedett el, annak szélétől 5-10 centiméterre. A szenzor és a munkaterület között a berendezés fém fala helyezkedett el, a borítófedél viszont nem volt ráhúzva a kemencére.

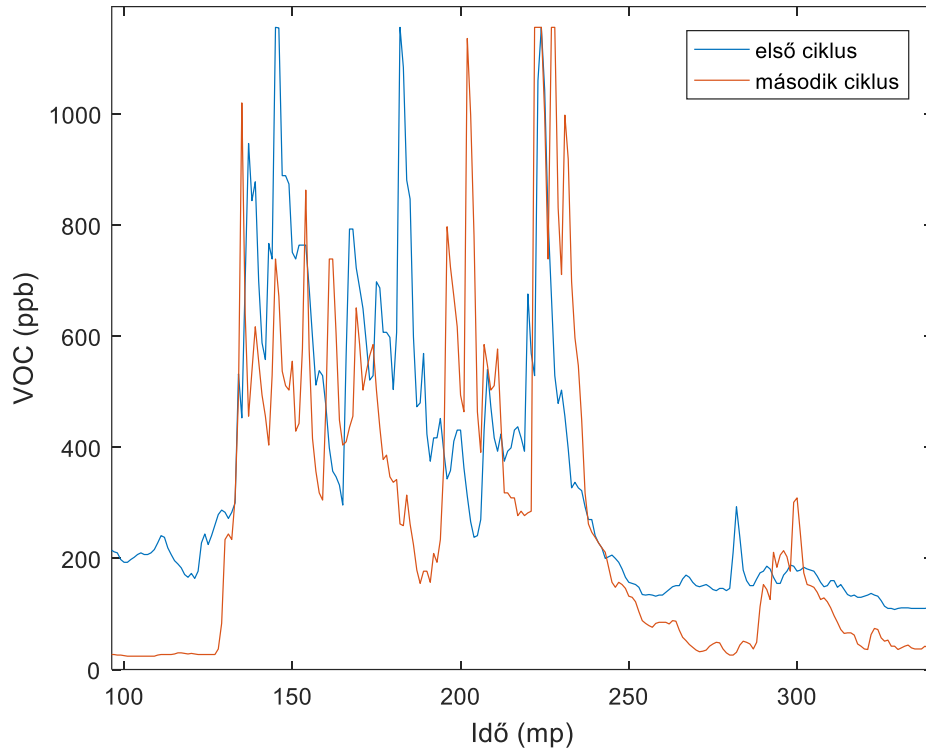


**3.9 Ábra: Szелеktív hullámforrasztó kemence VOC kibocsátása**

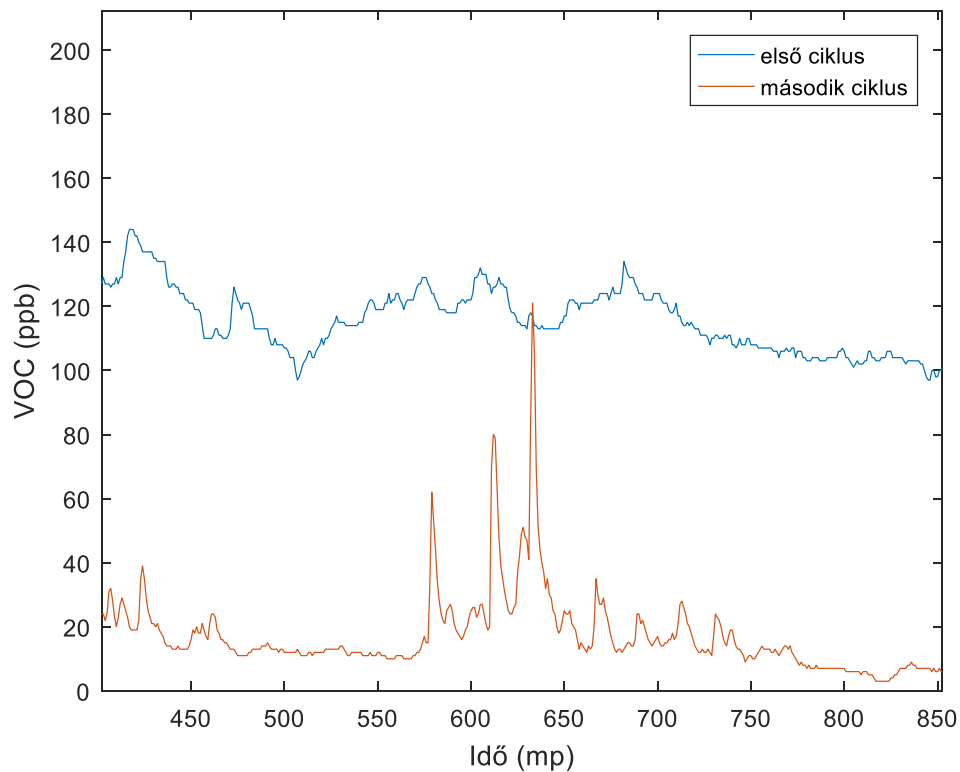
A 3.9-es ábrán látható, hogy a két ciklus jellegre nagyon hasonló, számottevő eltérést az első ciklus 200 ppb-s ofszetje mutat. Ezt az eszköz kontakthibából eredő újraindulása utáni önkalibrálási szakasz kihagyása okozta. A ciklus első 100 másodpercében a NYHL behelyezése, a folyamat előkészítése zajlott. Ezután a 120-250 másodperces intervallumban a folyasztószer felvitele ment végbe, amelynek légszennyező hatása telítésbe vitte a szenzort ( $VOC_{max} = 1187$  ppb). Ez a részlet a 3.10-es ábrán figyelhető meg jobb felbontásban.

Ebben a két percben a munkaterület környékén zavaró szagokat alig lehetett érezni, a megnövekedett szennyezőanyagokat csak a szenzor érzékelt igazán. Ez egyrészt kellemesebb munkaterületet eredményezett, ugyanakkor az egészségkárosító szerek szagtalansága aggályokat vethet fel, mivel az operátor így nem veszi észre, ha azokat nagy mennyiségben lélegzi be, az érezhető jelek hiányát hamis biztonságként

értelmezheti. Ez a kisebb felfedezés rámutat arra, hogy légzőszerveink érdekében mennyire fontos lehet egy kompakt, akár ruhába is illeszthető szenzor kifejlesztése.



**3.10 Ábra: A folyaszószér felvitelének szakasza**



**3.11 Ábra: A távolság hatása az érzékelt gázokra**

A forrasztás alatt a korábbiakhoz képest kevés VOC távozott a munkatérből. Ennek ellenére a mért értékeken látható volt, hogy a forrasztás mikor zajlott a szenzorhoz közel, és mikor haladt a távolabbi oldal felé. A 3.11-es ábrán a második ciklus esetén ez jól kivehető, az első ciklusban azonban a nem kívánatos ofszet miatt ezek a kis részletek elvesztek a zajban.

### **3.1.5.3 Kézi marógép**

Az Amphenol SM-PWM-01C porérzékelő tesztelését a laboratórium NYHL-fűrő és maró szobájában végeztem. Itt 3 eszköz működésének porkibocsátását próbáltam feltérképezni: egy automata marógép, egy automata fűrőgép, valamint egy kézi marógép. A két automata gép vákuumos elszívással rendelkezett, működésük során a szenzor nem tudott érdemi változást mutatni a levegő portartalmában. Ez azt is

mutatja, hogy a berendezések valószínűleg (a mért tartományban) emberre káros port nem bocsátanak ki. A kézi marógépben is van elszívás, azonban sűrített levegős tisztítása során már sikerült látványosabb mérést készíteni, amelyet a 3.12-es ábrán láthatunk.



**3.12 Ábra: Kézi marógép tisztításának légszennyezése (10 mp-es időintervallumban)**

Jól látható, hogy a 4. másodpercben megkezdődött a tisztítás, mindkét csatorna „fésűs” lett. A felső jelalak a kisebb, az alsó a nagyobb részecskéket mutatja (aktív '0'). A kisebb részecskék kimenete az idő ~37%-ban, a nagyobb részecskék kimenete pedig az idő 33%-ában volt aktív. Ez azt sejteti, hogy a gépben lerakódott por inkább  $2 \mu\text{m}$ -nél kisebb átmérőjű részecskékből áll.



### **3.1.6 Értékelés, folytatás**

A programozás, az elektromos összeköttetés mind könnyű feladatnak bizonyult. A mérések előtt, a szenzorok próbálgatása során azonban egy kontaktushiba miatt több óra mérési eredményt vesztettem el, valamint a számítógépem elhasználódott USB-portja miatt a szelektív hullámforrasztó első ciklusának mérése is pontatlan volt.

A kapott mérési eredmények, görbék összességében tanulságosnak, hasznosnak bizonyultak, azonban szerettem volna egy olyan eszközt elkészíteni, amely képes ezeket az adatokat azonnal, valós időben kimutatni. Ehhez egy újabb verzióba kezdtem, amelynek fő célja a hordozhatóság, valamint a légszennyezettségre való gyors reagálás lehetőségének megadása.

## **3.2 Android-kompatibilis hordozható eszköz**

### **3.2.1 Célkitűzés**

A második, fejlettebb verzió tervezésénél három tulajdonságot tartottam a legfontosabbnak:

- hordozhatóság: a kész eszköz vezeték nélkül is képes legyen a működésre
- aktualitás: a szenzor által mért adatok feldolgozása, kijelzése valós időben történjen, az egészséget károsító környezetben tartózkodásról ne csak egy későbbi időpontban szerezzünk tudomást
- praktikusság: az eszköz használata legyen egyszerű, a káros környezetre pedig automatikusan, felügyelet nélkül figyelmeztessen

Ezen célok eléréséhez szükségem volt egy olyan, kijelzővel és/vagy egyéb interfésszel rendelkező eszközre, amely képes vezeték nélküli kommunikációra, valamint beépített akkumulátorral rendelkezik. Erre a tökéletes, könnyen hozzáférhető megoldás az okostelefon volt.

## 3.2.2 Felhasznált eszközök

### 3.2.2.1 Szenzorok

Az előző projektben mutatott teljesítménye, valamint apró mérete alapján a CCS811-et elengedhetetlen erőforrásnak tartottam a második konstrukció elkészítéséhez. Az SM-PWM-01C szenzor azonban erőforrás-igényes kimenete, szabálytalan mérései (kis részecskék és nagy részecskék kimenetének inkohereciája) miatt kikerült a választható szenzorok listájából. Helyette az alábbi szenzort választottam:

Gyártó	Termék	Mért összetevő	Érzékenység (por/gáz)	Detektor típusa	Kimenet
Winsen	ZPH01	por	>1 µm	fényszóródás	PWM/UART

Ez a szenzor UART és PWM kimenettel is rendelkezik. Ha a MOD lába magas impedanciás állapotban van, akkor egy másodperces periódusidővel a levegőben található koncentrációval arányos kitöltési tényezőjű PWM jelet ad ki (aktív '0'). Amennyiben ezt a MOD lábát földre húzzuk, akkor másodpercenként egyszer, 8 adatbájt és 1 stopbyte formátumban 9600 Baud sebességgel küldi ki a mért adatokat a 3.13-mas ábrán látható módon.

	0	1	2	3	4	5	6	7	8
Név:	Start bájt	Detekció típusának kódja	Egység (alacsony pulzus aránya)	Alacsony pulzus arányának egész része	Alacsony pulzus arányának tört része	-	Üzem mód	-	Ellenőrző bájt
Értéktartomány:	0xFF	0x18	0x00	0x00-0x63	0x00-0x63	0x00	0x01	0x00	0x00-0xFF

3.13 Ábra: A ZPH01-es szenzor kimeneti formátuma az adatlap alapján

### **3.2.2.2 Szenzoradat-feldolgozó hardver**

A szenzoradatok feldolgozását most is egy Arduino Nano végezte. Bár terveztem egy alacsonyabb szintű, AVR alapú, mikrokontrolleres rendszer használatát, a programozáshoz, hibakereséshez nem álltak rendelkezésemre megfelelő eszközök, így ezt a tervet elvettem.

### **3.2.2.3 Kommunikációs modul**

A szenzoradatok feldolgozó és az adatokat gyűjtő eszköz közötti adatátvitelhez a Bluetooth vezeték nélküli technológiát választottam. Ennek a lebonyolításához egy HC-05-ös, UART-on kommunikáló Bluetooth modult használtam. Ennek a modulnak az előnye a könnyű használhatósága, a sok dokumentáció és példaprogram, valamint az alacsony ár. A modult AT parancsokkal könnyedén konfigurálhatjuk.

### **3.2.2.4 Adatgyűjtő, felhasználói interfész**

Az adatok kijelzésére, tárolására Android alapú okostelefonomat választottam. Ez egy Redmi márkájú Note 7-es modell. Előnye, hogy az Androidra írt programokat kis változtatásokkal, előre látó tervezéssel szinte minden Android operációs rendszert futtató eszközre fel tudjuk telepíteni.

### **3.2.2.5 Programozó felületek**

Az Arduino Nano programozását most is a hivatalos fejlesztőkörnyezetben végeztem el.

Az Android programozását az elterjedt JavaScript nyelv ismeretének hiányában az MIT által kifejlesztett, szabadon használható MIT App Inventor-ral végeztem [21]. Ez egy könnyen használható, magas szintű, grafikus interfésszel rendelkező rendszer, amelyben a telefonra telepített „MIT AI2 Companion” program segítségével valós időben lehet a szoftver változásait ellenőrizni.

### **3.2.2.6 Tervezőprogram**

Mivel mindenképpen szerettem volna egy tartósabb, kontakthibáktól mentes hardvert készíteni, ezért NYÁK-ot is terveztem. Ehhez az Autodesk Eagle 9.5-ös,

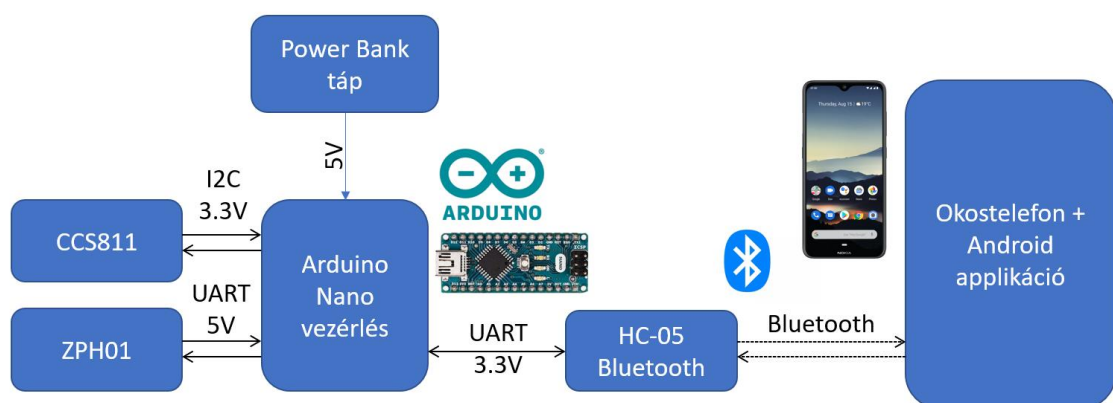
tanulói verzióját használtam. Ebben az ingyenes verzióban minden funkcionalitás benne van, csupán a panel méretét korlátozza. Választásomat korábbi tapasztalataim, a program előzetes ismerete alapján tettem.

### 3.2.2.7 Tápellátás

Az áramkör tápellátásához egy 2200 mAh-s powerbank telepet használtam. Ez az eszköz 5V/1A kimenettel rendelkezik, valamint beépített töltő- és védőáramkörökkel van felszerelve. Kis súlya, mérete, valamint funkcionalitása miatt ideális eszköznek bizonyult.

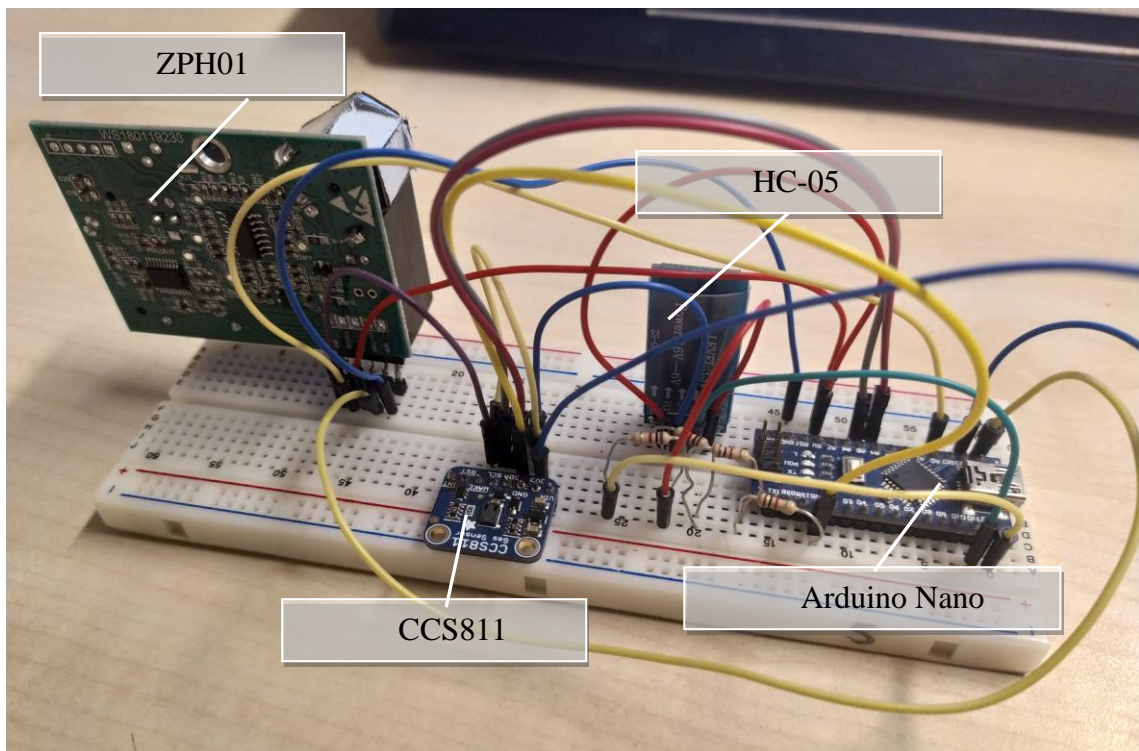
### 3.2.2.8 Kiegészítő eszközök

A második verziót már panelre terveztem, de ennek ellenére működőképességét most is breadboard-on teszteltem, a 3.14-es ábra szerinti összeállításban. Ehhez szükségem volt még vezetékekre is. A végleges, panelre szerelt verzióhoz egy telefontartót is felhasználtam, amelynek segítségével relatíve egyszerűen hordozhatóvá vált az eszköz.



3.14 Ábra: A második verzió blokkvázlata

### 3.2.3 Tesztelés



3.15 Ábra: A második verzió breadboard-on

Az elektronika helyes működését először prototípus-panelen teszteltem. A ZPH01 MOD lábának földre húzásakor az eszköz UART kimenetre kapcsol. Ezt a soros portot az Arduino A0 (TXD) és D13 (RXD) lábára kötöttem. A HC-05-ös modul 5V-ról működik, azonban adatlábai 3.3V-os feszültségszinten működnek. A feszültségszintek betartása, a modul védelme érdekében a modul RX lábára érkező jelet egy feszültségosztóval csökkentettem.

#### 3.2.3.1 Arduino szoftver

A Bluetooth modul és a ZPH01 is soros kommunikációval működnek. Ahhoz, hogy a debuggolást számítógépről is követhessem, 3 soros kommunikációs portra volt szükségem. Ehhez az Arduino SoftwareSerial könyvtárát használtam, amelynek segítségével könnyen újabb csatornákat deklarálhatunk. Ez a könyvtár az AVR megszakításait kihasználva soros kommunikációt tud lebonyolítani azokon a pineken

is, amelyeket hardveresen nem erre terveztek. A csatornákat 2-2 soros kóddal hozhatjuk létre:

```
SoftwareSerial blue(11, 12);      //a bluetooth modul csatornája
SoftwareSerial dust(14,13);      //a porérzékelő csatornája
blue.begin(9600);
dust.begin(9600);
```

A csatornák inicializálása után a már megismert „print()” és „read()” függvényeket használva kommunikálhatunk a modulokkal.

A következő feladat a szenzoradatok olvasásának időzítése. A CCS811 adatait másodpercenként egyszer tudjuk olvasni, egy általunk választott időpontban. A porérzékelő adatai bármikor érkehetnek, ezért egy „polling” típusú függvényt készítettem:

```
unsigned long prevtime = 0;
void loop() {
    if (dust.available()) { ... ZPH01 szenzor adatainak feldolgozása
    }
    if (millis()-prevtime > 1000) { ... CCS811 olvasás, adatküldés BT-n
    ...
    prevtime = millis();
    }
}
```

A ZPH01 adatainak olvasásához egy tömböt készítettem, majd a beérkező adatbájtokat ebbe a tömbbe tároltam. A kimeneti formátuma alapján (3.13-mas ábra) a 3-mas és 4-es indexű bájtok a legfontosabbak számunkra. Az üzenet elején található 0xFF kezdő bájt segítségével könnyen el tudtam helyezni a beérkező adatokat.

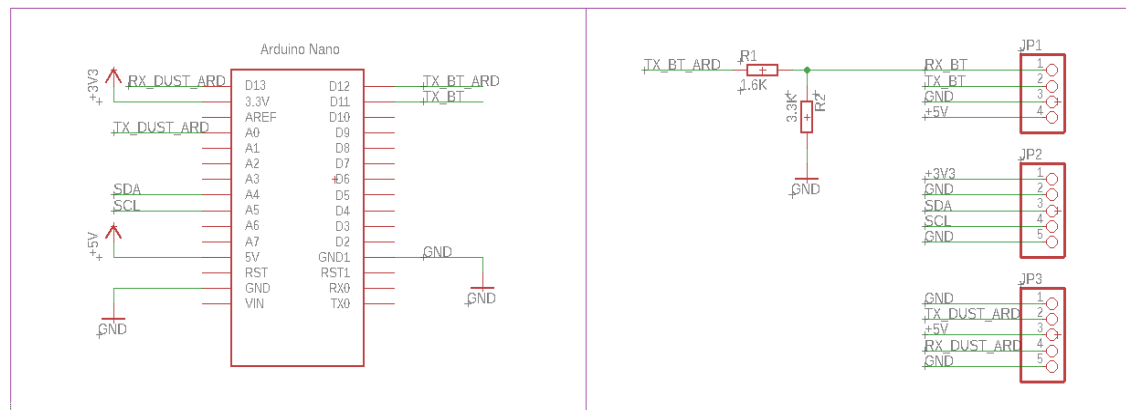
Az egységes adatforma érdekében a kiküldött adatokat mindig xxxx|xxxx|xx|xx (co2|voc|por egész|por tized) formátumra hoztam. Ez az okostelefonon futó alkalmazás szempontjából volt fontos.

Miután meggyőződtem a helyes működésről, újabb célnak a stabil, hordozható verzió létrehozását tűztem ki. A tápellátás tervezéséhez megmértem a fogyasztást, amely körülbelül 200-230 mA között mozgott, a Bluetooth kapcsolattól függően (csatlakoztatva 200 mA, csatlakoztatás nélkül 230 mA – ez a csatlakozásra váró periódusban bekövetkező folyamatos címhirdetés miatt lehetséges).

## 3.2.4 Tervezés

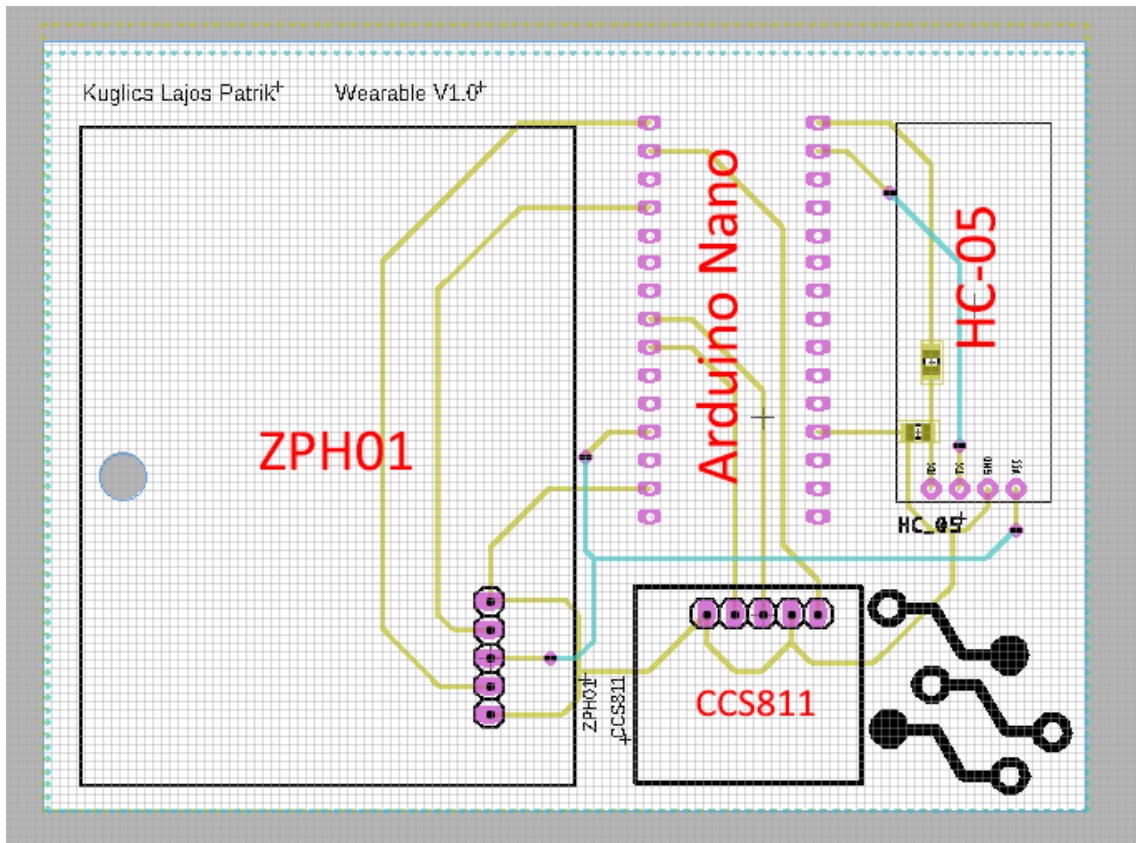
### 3.2.4.1 Nyákterv elkészítése

A NYÁK tervezésének első lépése a kapcsolási rajz elkészítése volt. Mivel a felhasznált modulok egyikéhez sem találtam könnyen használható, naprakész könyvtárat, ezért a csatlakozási pontjaikat szabványos, 2.54 mm-es lábkiosztású tűksorokkal helyettesítettem. A tervezésnél nem volt semmilyen bonyolult feladatom: a modulok beépített tápegységei, védelmi áramkörei, a tápellátás stabilitása miatt sem tápszűréssel, sem egyéb védelemmel nem kellett foglalkoznom, így a 3.15-ös ábrán látható kapcsolási rajzot kaptam.



3.16 Ábra: a kapcsolási rajz

Ezután az alkatrészek elhelyezése következett. Itt figyelmem kellett, hogy a porérzékelő álló, függőleges helyzetben tud csak jól mérni. A Bluetooth modul antennáját a panel szélére pozicionáltam, így a lehető legkevesebb zavar mellett működhet. Az alkatrészek elhelyezéséhez megmértem a méreteiket, majd ezek alapján téglalap alakú jelölők segítségével elhelyeztem őket. A modulokat egy oldalra terveztem, hogy minél laposabb, egyenletesebb elektronikát kapjak. Az alsó oldalra csak huzalozás, földkitöltés, valamint két apró ellenállás került. A kész nyákterv a 3.16-os ábrán látható.



3.17 Ábra: a nyákterv

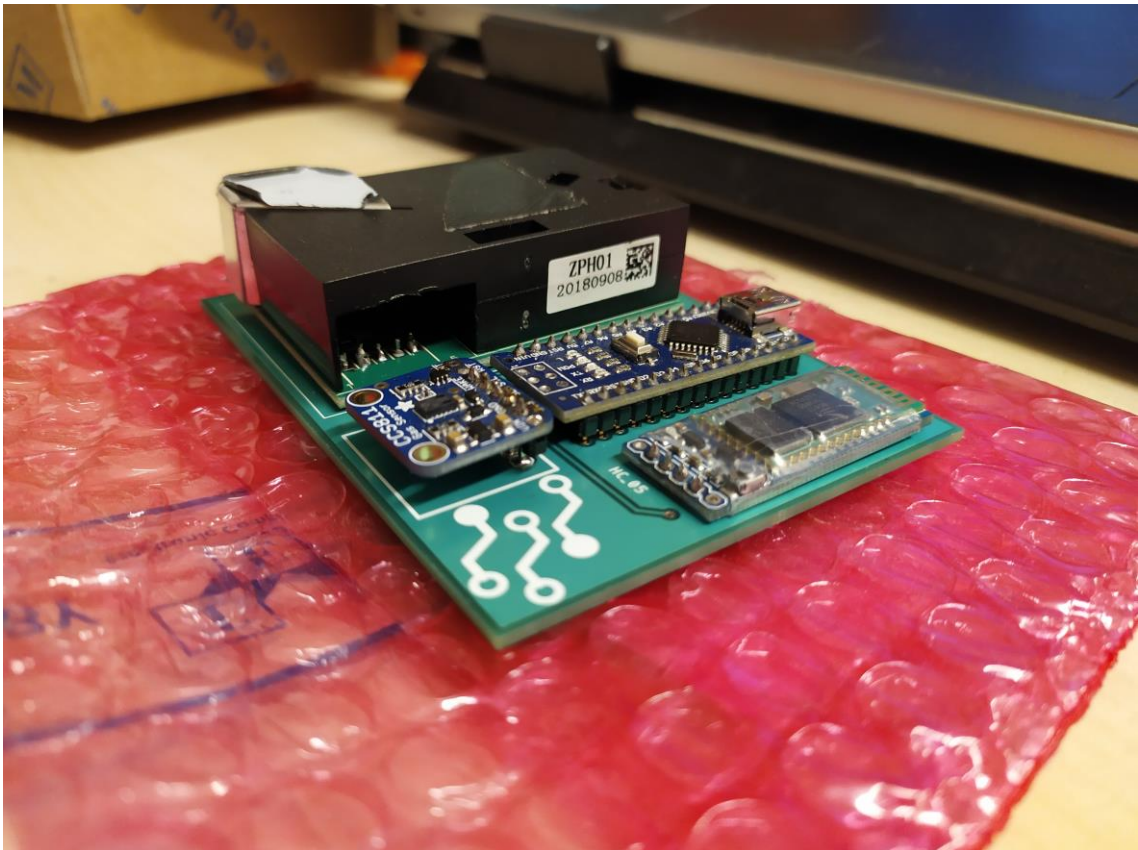
### 3.2.4.2 Forrasztás

A panel elkészülte után az alkatrészeket kézzel forrasztottam be. A CCS811-tyet emeltem a panel síkjából, hogy a későbbiekben a tartójában legyen körülötte légmozgás. A ZPH01 beszereléséhez le kellett cserélnem a rajta található 90°-os csatlakozót. Ez egy óvatos művelet volt, amelyet szinte hibátlanul sikerült elvégezni, csupán a műanyag ház minimális sérülésével járt, amely a 3.17-es ábrán a CCS811 feletti részen figyelhető meg.

Az elektronika tesztelését először labortápról végeztem. A feszültséget lassan növeltem az eszköz bemenetén, figyelve a fogyasztást. Ez az elővigyázatosság rengeteg problémát hozott: a bluetooth modul, számomra ismeretlen okból így nem kapcsolt be, csak ha a feszültséget sokkal gyorsabban kapcsoltam rá. Ezt valószínűleg a bemenetén található tápegység alacsony bemeneti feszültség ellen védő áramköre okozhatta.

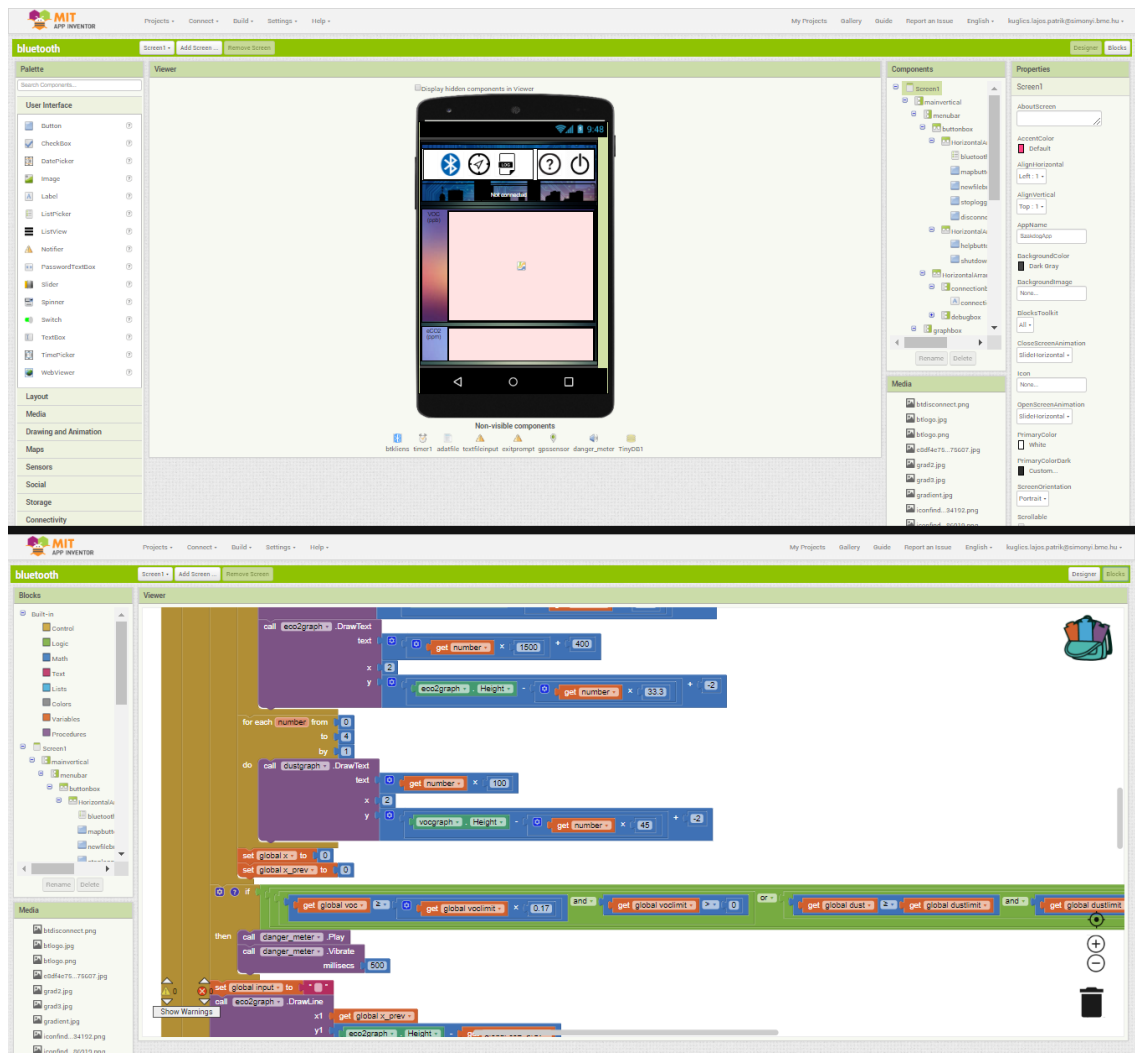


Szerencsére ez a probléma az éles működés során nem lép fel, hiszen ott a tápellátás kapcsolása mindig gyorsan végbemegy.



**3.18 Ábra: A kész elektronika**

## 3.2.5 Programozás



3.19 Ábra: Az MIT App Inventor felülete

Az Android programot az MIT App Inventor felhőszolgáltatásával készítettem. Ebben a szerkesztőben a 3.18-as ábrán látható grafikus felületeken végezhetjük az UI tervezését, valamint a programkód megírását is. A kódolásnál nincs szükség egyedi függvények írására, a háttérben futó folyamatok részletes ismeretére: előre megírt megszakításokhoz írhatunk tetszőleges lépéssorozatokat magas szintű, jól átlátható, dokumentált nyelven.

Az alkalmazásom 3 fő részből áll: a főoldalból, a „Help/Settings” és a „Map” ablakokból. Az ablakok közötti adatátvitelért egy lokális TinyDB adatbázis a felelős.

### 3.2.5.1 Főoldal

A 3.19-es ábrán látható főoldalon található gombok segítségével csatlakozhatunk az elektronikához, itt indíthatunk adatnaplózást, valamint itt figyelhetjük meg a szenzoroktól kapott adatokat külön-külön grafikonokon. A képernyő tetején található menüsorból irányíthatjuk a programot, léphetünk át másik ablakokba, valamint itt léphetünk ki az alkalmazásból.

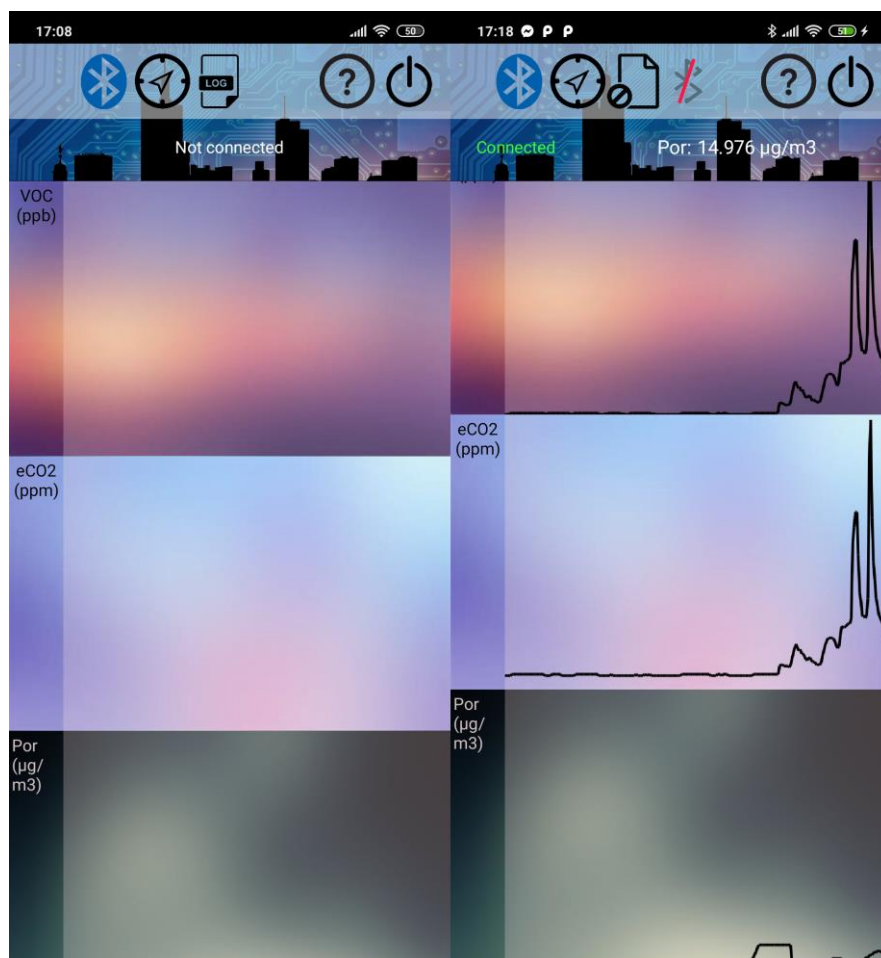
A Bluetooth ikonra kattintva feljön a párosított eszközök listája, ahonnan kapcsolódhatunk a szenzorhoz, majd a tőle kapott adatokat a grafikonra rajzolja az alkalmazás. A „Log” gomb megnyomása, és a fájlnev megadása után a helyi tárhelyre naplózhatjuk a kapott értékeket. Ha kapcsolódtunk a szenzorhoz, vagy naplózást indítunk, megjelenik az adott aktivitást megszakító gomb is. A kapcsolat bontására szolgáló gombot hosszan nyomva megtekinthetjük a beérkező adatokat eredeti formátumukban: ez a kis rejtett funkció elengedhetetlen segítséget nyújtott a kommunikációs problémák elhárításában.

A Bluetooth kapcsolat felépítésekor először mindig kiürítjük a bejövő adatokat, majd az új adatokat egy karakterláncba mentjük el. Ezt a változót a küldő oldalon beépített elválasztó karakter („|”) segítségével feldaraboljuk, majd egy tömbbe rakjuk. A tömb elemeit minden ciklusban a grafikonokra rajzoljuk. Amennyiben a naplózás engedélyezve van, a karakterláncot a telefon tárhelyére mentjük a helyadatokkal (opcionális), valamint a pontos idővel együtt.

A grafikonra rajzolásnál figyelniünk kell, hogy az Y irány fordítottan működik, ezért a felrajzolni kívánt pontot mindig a grafikon magasságából kell kivonni, így kapjuk meg a megfelelő koordinátáit. A mérési pontok rajzolásához mindig az előző mérési pontból húzunk egy vonalat, majd a jelenlegi adatokat elmentjük, így azokat a következő vonalhoz felhasználhatjuk.

A grafikonokra kapcsolódás után felrajzoljuk a skálát, valamint az „x” irányú betelítés után mindig frissítjük. A grafikonok szélesebbek a képernyőnél, teljes tartalmukat a grafikon oldalra húzásával lehet megtekinteni. Ennek köszönhetően egy relatíve nagyobb időtartományt tudunk megfigyelni, valamint kisebb képernyőkön is jól használható.

A felhasználó értesítésére egy hangos, rezgő értesítést is implementáltam. Amennyiben a VOC vagy por koncentráció meghaladja a beállított határértéket, a telefon jelzi azt. A jelzések gyakoriságát, valamint a határértékeket a „Help/Settings” oldalon konfigurálhatjuk.



3.20 Ábra: Az alkalmazás kezdőképernyője

### 3.2.5.2 Map ablak

A „Map” ablakban bekapcsolhatjuk a helyadatok naplózását, valamint megtekinthetjük az eszköz által mért jelenlegi pozíciónkat, amennyiben a telefon beállításában engedélyezzük a helymeghatározást.

A térképhez az ingyenes, nyílt forráskódú OpenStreetMap modult használtam fel. Az OpenStreetMap projektet az OSM Foundation tartja karban. A térkép betöltéséhez internetkapcsolat szükséges. Itt lehetőség lett volna a Google Maps API (application programming interface) használata is egy beépített HTML keretben, azonban a bonyolultabb funkciók eléréséért fizetni kell, így inkább maradtam az OSM használatánál.

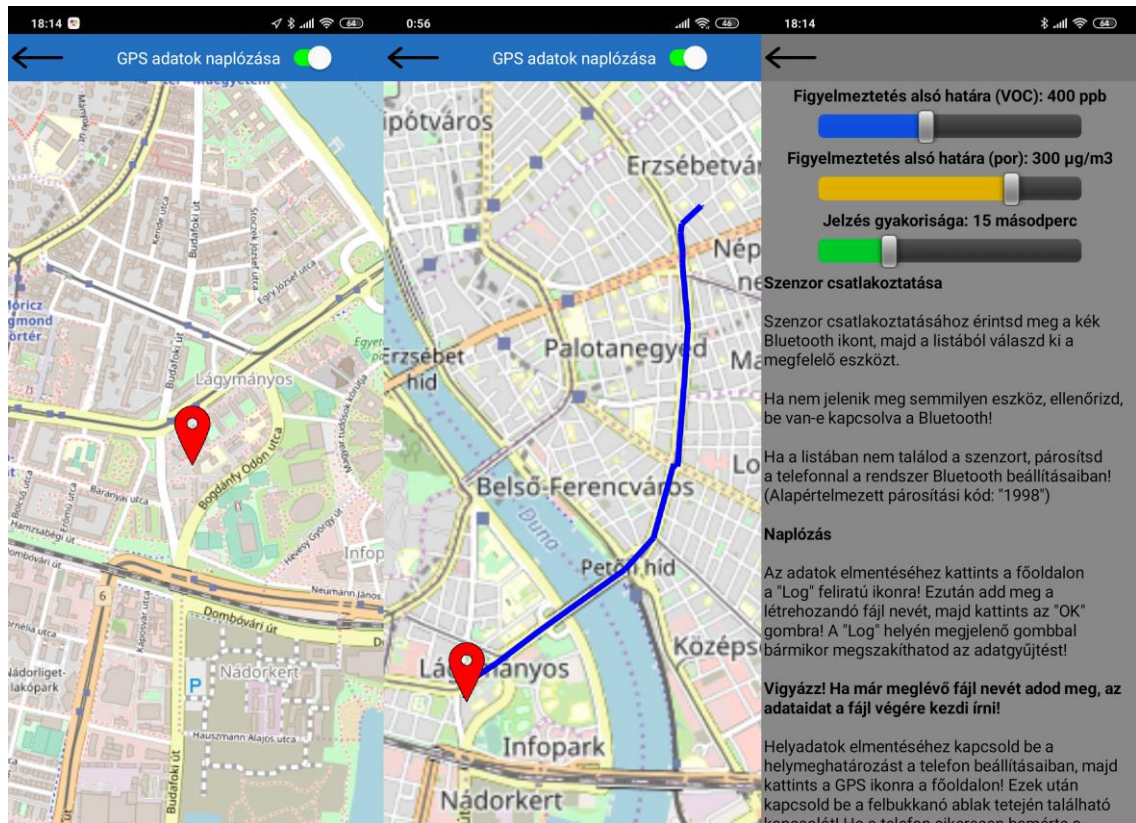
A Google Maps integrálása, valamint a Google szerverein végzett számítások sok, digitális adataikat féltő ember számára érzékeny területnek számíthatnak. Ugyanakkor esztétikailag, a térkép naprakészsége szempontjából egyértelműen jobb megoldást nyújt, mint az OSM használata. Egy ilyen döntést egy eladásra szánt termék esetében mindenképpen megfelelően mérlegelni kell. Ugyanakkor a Google Maps API-ját monetizációs célból átalakította a közelmúltban a Google [22], ezért az alkalmazásnak a pénzügyi háttérét meg kell teremteni.

A háttérben futó helymeghatározó szolgáltatásnak köszönhetően a térkép 10 másodpercenként frissíti a pozíciónkat. Használat közben az alkalmazás egy GeoJSON formátumú karakterláncba menti a GPS által meghatározott magasság és szélesség párokat. Ezeket a pontokat egy „LineString” modulba töltjük, amely a térképen vonalakkal összeköti a pontokat, így a felhasználó mindig vissza tudja nézni, merre járt az alkalmazás indítása óta. Ez a funkció a 3.21-es ábra középső képernyőjén látható.

### 3.2.5.3 Help/Settings

A „Help/Settings” ablakban elolvashatjuk az alkalmazás használatához szükséges tudnivalókat, útmutatókat, a mért összetevők rövid ismertetőjét, valamint beállíthatjuk a figyelmeztető jelzés időzítését, határértékeit. Ha valamelyik határértéket 0-ra állítjuk, az a komponens nem fogja aktiválni a figyelmeztetést. Az időzítő 0-ra

állítás teljesen kikapcsolja a figyelmeztetést. Ezeknek a beállításoknak köszönhetően a felhasználó könnyedén a saját ízlésére szabhatja az értesítéseket. Ez különösen hasznos, ha valaki például zárt térben szeretne értesítéseket kapni a magas porkoncentrációról, azonban a VOC miatt mindig jeleznie az eszköz.



3.21 Ábra: A Map és a Help/Settings ablakok

## 3.2.6 Mérés

### 3.2.6.1 Összeszerelés

A hordozható konstrukció elkészítéséhez vettem egy szövet alapú, fólia fedővel rendelkező telefontartót, amelyet általában sportoláshoz használnak. A fólián kivágtam a szenzoroknak szükséges felületeket. A powerbank mérete miatt éppen, hogy csak belefért a tartóba, ezért a tartó oldalát, és az USB csatlakozó feletti fóliát is ki kellett vágnom a 3.21-es ábrán látható módon. A felesleges USB kábelt a pánt alá rejtettem. A

végeredmény egy stabil, az elektronika mozgását jól megakadályozó, könnyű, viselhető eszköz lett.



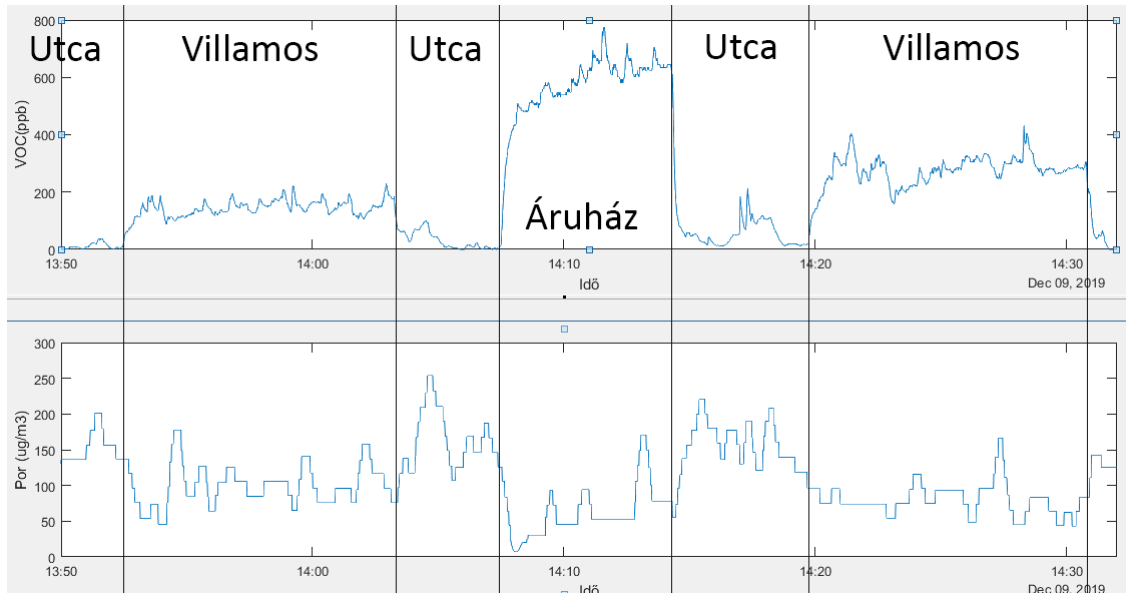
**3.22 Ábra: Az összeszerelt eszköz**



**3.23 Ábra: A berendezés felvéve**

### 3.2.6.2 Próbák a városban

Az első mérést a 4-es és 6-os villamos vonalán végeztem. Az elektronikát a karomra rögzítettem, a 3.22-es ábrán látható módon. Úticélom egy Dohány utcai üzlet volt. A Schönherz kollégiumtól indultam, majd a Blaha Lujza térig utaztam. Itt leszálltam, az üzletig sétáltam. Visszafele ugyanezt az útvonalat választottam.



3.24 Ábra: Az első próbán mért eredmények

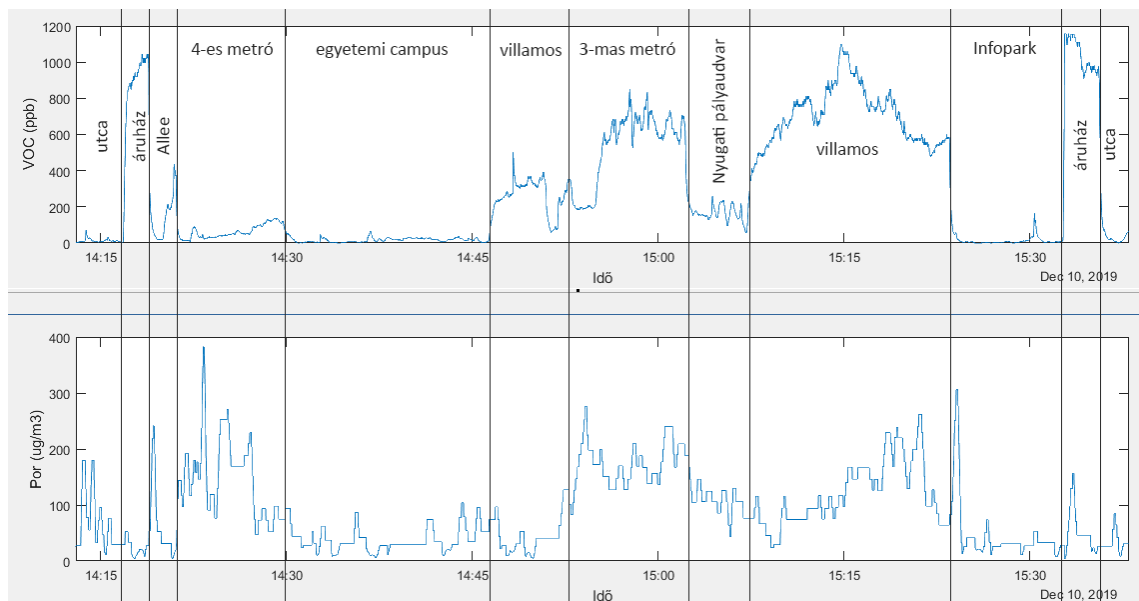
A mért adatok megjelenítéséhez most is Matlab-ot használtam. A .txt kiterjesztésű szövegfájlok adatait a („,|”) delimiter segítségével importáltam, majd a por százalékos adatait 20.8-al megszorozva megkaptam a  $\mu\text{g}/\text{m}^3$ -es értéküket. Ezeket a plot() függvénnyel rajzoltam ki, így kaptam a 3.23-mas és 3.24-es ábrákat. Az ábrákon látható segítő vonalakat és feliratokat ezután egy képszerkesztő szoftverrel szerkesztettem rá.

A grafikonokon jól látható, hogy az utcán alacsony a VOC koncentráció, a por azonban itt mindig megugrik. A villamosra szállva kis mértékben lecsökkent a szálló por koncentrációja, a VOC szint viszont megnövekedett. A villamoson az ajtó mellett utazva a telefonra érkező adatokat figyelve láttam, hogy mindig az ajtónyitáskor csökken a VOC szint, a légmozgás miatt a por koncentrációja azonban megnövekedett.



Az üzletbe lépve azonnal éreztem, hogy jót tenne egy szellőztetés, ezt a VOC görbén látható hirtelen ugrás is bizonyítja.

Visszafele a villamoson magasabb VOC szinteket mértem, amire egy magyarázatot találtam: a belváros fele tartó villamoson (az odaút során) kevesebben utaztak, valamint a végállomásnál jobban ki tudott szellőzni, a belvárosból kifele azonban sokkal többen voltak, valamint a levegő is telítettebb volt az addig megtett út (végállomástól számítva) miatt.



**3.25 Ábra: Eredmények egy második út során**

A második tesztre egy hosszabb úton került sor. Egy nemrégiben közzétett [23], a londoni metróhálózatban elvégzett kísérletsorozat mintájára szerettem volna megnézni, mennyire rossz a levegő a budapesti metrómegállóiban. Míg a londoni metrónál helyenként 400-800  $\mu\text{g}/\text{m}^3$ -es PM<sub>2.5</sub> koncentráció volt, addig a 4-es és 3-mas metró megállóiban ezek az értékek legtöbbször 200  $\mu\text{g}/\text{m}^3$  alatt maradtak. Ez alól a kivételeket a metrószerelvény érkezésekor keletkezett légáramlat okozta, amely mindkét szekciónál hirtelen nagy kiugrást okozott a mérésben. Ennek fő okai a metró szelével érkező rengeteg, mechanikai kopásból származó apró fém részecske, valamint a légmozgás által felkapott, az utasok által levitt porszemcsék lehetnek.

A két város metrómegállóinak levegője közötti különbséget minden bizonnyal több, egymástól nem teljesen független tény okozza: Londonban a magas terhelés miatt nem engedhetik meg maguknak, hogy a metróhálózatot felújítsák, mivel az magas kieséssel, a tömegközlekedés megbénulásával járna. Emiatt az öreg, sokat használt, gyenge szellőztetéssel rendelkező metróalagutakban, felszerelésekben könnyen összegyűlik a mechanikus elemek kopásából származó fémpor, amelynek mennyisége a nagy leterheltség miatt fokozatosan növekszik. Ezzel ellentétben a budapesti metróhálózat újabb, jobb a szellőztetése, valamint az alacsonyabb terhelés miatt jobb karbantartással, kisebb szennyezéssel működik.

Ennek ellenére mindkét mérésből látható, hogy a tömegközlekedést előnyben részesítők sajnos jól mérhető, az utcai viszonyokhoz képest nagy mennyiségű szálló részecske koncentrációnak vannak kitéve. Ahogy az előbb említett tanulmány a föld alatti PM2.5-ről írja, nagyon fontos, hogy ezeket a veszélyforrásokat jobban kivizsgáljuk, elhárítsuk.

Mivel a porkoncentráció sok tényezőtől függhet (például időjárás, évszak, napszak), ezért fontos hangsúlyoznom, hogy ezek a mérések csupán az arányokat hivatottak bemutatni, a pontos, átlagos koncentrációk megadásához akár több száz mérésre is szükség lehet, ennél pontosabb, drágább eszközök segítségével.

### **3.2.7 Összegzés**

A ZPH01-es szenzor jobban működött, mint elődje, azonban enyhén lassú válaszideje, „darabos” kimenete miatt még van tér továbbfejlesztésre. A CCS811 viszont alacsony fogyasztásával, gyors válaszidejével több leendő projektben is biztos helyet szerzett magának.

Mindent figyelembe véve sikerült a terveknek megfelelő végeredményhez jutnom: az elektronika könnyű, egyszerűen használható, az Android alkalmazás pedig véleményem szerint esztétikusra sikerült. A beépített jelző és naplózó funkciók jól működnek, az elektronika megfelelő „csomagolásával”, az alkalmazás polírozásával akár egy eladható termék-párosként is megállnák a helyüket. A legnagyobb bajt talán a

magas fogyasztás okozza (a porérzékelő egyedül 150mA-t is fogyaszthat akár), de egy kompaktabb, hatékonyabb, esetleg saját fejlesztésű szenzorral talán ezt is ki lehetne küszöbölni.

### 3.3 Fejlesztési tervek, piaci kitekintés

Az integrált szenzoráramkörök árának rohamos csökkenésével egyre többen engedhetik meg maguknak, hogy azokat tartalmazó eszközöket vásároljanak. Ezt, valamint az okostelefon-használat magas szintjét, és az egészségtudatosság erősödését figyelembe véve magától adódik, hogy egy viselhető, ergonomikus eszközt viszonylag könnyen el lehet adni.

Még a nyugati országokban sincs ennek kultúrája, csupán egy-két nevesebb hordozható szenzor létezik, mint például a Flow [24], vagy az Atmotube [25]. Ezeknek közös jellemzője, hogy nem a hagyományos értelemben vett „wearable” kategóriába tartoznak, hanem pántokkal lehet őket hátitáskára, egyéb helyre erősíteni. Ennek ellenére több nevesebb híroldal is foglalkozott velük, például a Wired, a Bloomberg, vagy a Phys.org. Ebből látható, hogy bár a „mainstream” közösség még nem igazán ismeri ezeket a lehetőségeket, a témára érzékeny emberek miatt biztosan lenne kereslet egy, az általam elkészített elektronikához hasonló termékre.

A közeljövőben mindenképpen szeretném kétfelé választani a projektet: egy CCS811 alapú, hajlékony, vagy ruhába építhető, szinte észrevétlenül viselhető szenzor, amelyet azok számára optimalizálnék, akik több időt töltenek el zárt terekben, valamint egy, a fent említett termékekhez hasonló porérzékelő elektronikát, amely a kültéren tartózkodók, esetleg sportolók számára lenne kialakítva. A por koncentrációjának mérése körülményesebb: fontos a légmozgás iránya, valamint a mérés jellegéből adódóan az elektronika is nagyobb méretű, nagyobb fogyasztással rendelkezik, ezért a meglévő termékektől túl sokban nem tudok eltérni alternatív technológiák hiányában.

Ezen felül még opcionális fejlesztési irány lehet egy stacionárius, robusztusabb kivitelű állomás, amely mind kül- és beltéren elhelyezhető. Ezzel kültéren urbánus lokációk, állomások, aluljárók, forgalmasabb utcák levegőjét lehetne monitorozni,

beltéren pedig boltok, közintézmények (pl. iskolák, orvosi rendelők), munkahelyek légminőségéről adhatna értékes információkat. Közintézmények, közterületek esetén állami projekt keretében lehetne ezt a rendszert létrehozni, míg magánterületen, boltokban, hasonló területeken a céges társadalmi felelősségvállalás lehetne a fő mozgatórugó a terjeszkedésre. Ezekkel az eszközökkel könnyen megmondható, hogy mennyire szellőzik jól egy bevásárlóközpont, mikor van szükség egy tanterem szellőztetésére, vagy mennyire tiszta egy park levegője. Megfelelő kültéri lefedettség esetén pedig zónákra lehet bontani a települést, és az adatok feldolgozását követően átfogó képet lehet adni a terület légminőségéről. Ennek segítségével nem csak a telepítés helyén használható adatokat kapunk, hanem naprakész információt adhatunk a lakosságnak az AQI-ról, valamint rengeteg adatot szerezhetünk egy pontos előrejelző modell elkészítéséhez, kutatások elősegítéséhez.

Az Android alkalmazásom, mint első saját készítésű app, motiváló tényezőként hatott rám, és elégedett voltam a keretrendszerrel is, ezért nem szeretném lecserélni, inkább funkcióinak kibővítését tűzöm ki célul. Hosszú távon azonban az App Inventor által nyújtott lehetőségek biztosan nem lesznek kielégítőek, így a későbbiekben szeretném megtanulni a Kotlin nyelven történő Android programozást is, amivel egy igazán elegáns, letisztult felületet tudnék készíteni funkcióbeli korlátozások nélkül.



3.26 ábra: Zóna szerinti felosztás szemléltetése

A közösségi adatgyűjtés (angol: „crowdsourcing”) eléréséhez mindenképpen szükséges egy szerver, amely kommunikál az egyes eszközökkel, a beérkező adatokat feldolgozza, megjeleníti. Mivel a mozgó szenzorok kapcsolatára az egyetlen reális, valós idejű kapcsolat a mobilinternet-hálózat használata, ezért nagyon fontos, hogy az adatok átvitele minimális adathasználattal járjon: ehhez a feladathoz tökéletes választás az MQTT protokoll (Message Queuing Telemetry Transport), amely alacsony sávszélesség-használatra, valamint sok-az-egyhez/sok-a-sokhoz kapcsolatok működtetésére specializálódott [26]. Az utolsó ábra egy lehetséges közösségi felhasználást mutat be, amely során látszik, hogy a környezetből nyert információk alapján zónákra lehet osztani az urbánus környezetet. Ez alapján terveim szerint először a kampuszt lehetne felderíteni, később pedig akár kerületi-városi vizsgálatra is lehetőség nyílhatna.

## **4 Köszönetnyilvánítás**

Szeretnék köszönetet mondani az Elektronikai Technológia Tanszéknek, amiért elkészítették nekem a végső eszközhöz elengedhetetlen nyomtatott áramköri lapot.

## 5 Irodalomjegyzék

- [1] Kataoka, Hiroyuki & Ohashi, Yasuhiro & Mamiya, Tomoko & Nami, Kaori & Saito, Keita & Ohcho, Kurie & Takigawa, Tomoko. (2012). *Indoor Air Monitoring of Volatile Organic Compounds and Evaluation of Their Emission from Various Building Materials and Common Products by Gas Chromatography-Mass Spectrometry*; 10.5772/31659.
- [2] Miles Brignall: 'Our MDF furniture brought toxic fumes into our home'; <https://www.theguardian.com/money/2019/feb/09/mdf-furniture-toxic-fumes-formaldehyde> (elérés: 2019.12.01.)
- [3] <https://www.airnow.gov/aqi/aqi-basics/>
- [4] <https://www.airnow.gov/sites/default/files/2020-05/aqi-technical-assistance-document-sept2018.pdf>
- [5] Yongjian Zhu, Jingui Xie, Fengming Huang, Liqing Cao, *Association between short-term exposure to air pollution and COVID-19 infection: Evidence from China, Science of The Total Environment, Volume 727, 20 July 2020, 138704, 10.1016/j.scitotenv.2020.138704*
- [6] E. Bontempi, *First data analysis about possible COVID-19 virus airborne diffusion due to air Particulate matter (PM): The case of Lombardy (Italy), Environmental Research Volume 186, July 2020, 109639, 10.1016/j.envres.2020.109639*
- [7] Silvia Comunian, Dario Dongo, Chiara Milani, Paola Palestini, *Air Pollution and COVID-19: The Role of Particulate Matter in the Spread and Increase of COVID-19's Morbidity and Mortality Int. J. Environ. Res. Public Health 2020, 17(12), 4487; https://doi.org/10.3390/ijerph17124487*
- [8] Ye Yao, Jinhua Pan, Zhixi Liu, Xia Meng, Weidong Wang, Haidong Kan, Weibing Wang, *Temporal association between particulate matter pollution and case fatality rate of COVID-19 in Wuhan, Environmental Research, Volume 189, October 2020, 109941*
- [9] Ye Yao, Jinhua Pan, Weidong Wang, Zhixi Liu, Haidong Kan, Yang Quiu, Xia Meng, Weibing Wang, *Association of particulate matter pollution and case fatality rate of COVID-19 in 49 Chinese cities, Science of The Total Environment, Volume 741, 1 November 2020, 140396*

- [10] Z. Yongjian, X. Jingu, H. Fengming, C. Liqing, *Association between short-term exposure to air pollution and COVID-19 infection: Evidence from China*, *Science of the Total Environment* (2020)
- [11] Wu, X., Nethery, R.C., Sabath, B.M., Braun, D., Dominici, F., 2020. *Exposure to air pollution and COVID-19 mortality in the United States*. medRxiv.
- [12] Cosimo Magazzino, Marco Mele, Nicolas Schneider, *The relationship between air pollution and COVID-19-related deaths: An application to three French cities*, *Applied Energy* Volume 279, 1 December 2020, 115835
- [13] Zander S. Venter, Kristin Aunan, Sourangsu Chowdhury, Jos Lelieveld, *COVID-19 lockdowns cause global air pollution declines*, *PNAS* August 11, 2020 117 (32) 18984-18990; first published July 28, 2020; <https://doi.org/10.1073/pnas.2006853117>
- [14] *ELTE EKOL: A gázkromatográfia alapjai*; <http://technologia.chem.elte.hu/hu/zoldkemialabor/GCalap.pdf> (elérés: 2019. 12. 01.)
- [15] <http://www.membrapor.ch/electrochemical-gas-sensors/> (elérés: 2019.12.01)
- [16] C.K.Laird, I.Verhappen (2010); *Instrumentation Reference Book (Fourth Edition)*, Chapter 25 - Chemical Analysis: Gas Analysis; Pages 401-428
- [17] [https://www.mouser.com/catalog/specsheets/Amphenol\\_12202018\\_AAS-916-139A-Telaire-SM-UART04L-AppNote-102018-web.pdf](https://www.mouser.com/catalog/specsheets/Amphenol_12202018_AAS-916-139A-Telaire-SM-UART04L-AppNote-102018-web.pdf) (elérés: 2019. 04. 01.)
- [18] <https://en.wikipedia.org/wiki/Arduino> (elérés: 2019.12.01.)
- [19] Dean Miller: Overview; <https://learn.adafruit.com/adafruit-ccs811-air-quality-sensor> (elérés: 2019.12.01.)
- [20] <https://github.com/PaulStoffregen/TimerOne> (elérés: 2019.12.01.)
- [21] <http://appinventor.mit.edu/> (elérés: 2019.12.01)
- [22] Mike Gingerich: *Google Maps Api moving to paid service*; <https://www.digitalhill.com/blog/google-maps-api-moving-to-paid-service/> (elérés: 2019.12.12)
- [23] J.D. Smith, B.M. Barratt, G.W. Fuller, F.J. Kelly, M. Loxham, E. Nicolosi, M. Priestman, A.H. Tremper, D.C. Green (2019). *PM2.5 on the London Underground*; *Environment International*, Volume 134, 2020
- [24] <https://plumelabs.com/en/flow/> (elérés: 2019.12.02)
- [25] <https://atmotube.de/?view=en> (elérés: 2019.12.02)



[26] <https://mqtt.org/>