



DIPLOMATERV-FELADAT

Emri Balázs (MSG2UV)

szigorló villamosmérnök hallgató részére

Intelligens termosztát tervezése

A feladat az otthon automatizálás egy részfeladatának, a fűtés vezérlésének modernizált megvalósítása. A modern otthonban a lakásban található eszközök távolról interneten keresztül is elérhetők. A fűtés vezérlésénél például már a régebbi típusú termosztátnál is beállíthatjuk a hét minden napján több időpontban a kívánt hőmérsékletet, amivel spórolni lehet az energiával. Azonban ezeknél a körülmények megváltozására nem tudunk távolból reagálni. Ha kiderül, hogy több vagy kevesebb napig leszünk távol, mint előre gondoltuk, a régi eszközöknél távolból nem tudunk módosítani a programon. Azonban távkapcsolattal rendelkező eszközöknél ez is megoldható. A távkapcsolat legrugalmasabb lehetőségeket biztosító módja az internet. Megfelelő védelemmel ellátott szerver segítségével az otthonunkhoz tartozó honlapon keresztül kényelmesen elvégezhető az átállítást akár okostelefonról is. A feladat során egy központi szeveren keresztül az internetre csatlakoztatható termosztát fejlesztése a cél. A tervezés során figyelni kell a versenyképességre mind a használhatóság, kezelői felület, mind a költség szempontjából. A felhasználói élmény javítására, az eszköz optikai kézmozdulatérzékeléssel, a kapacitív touch gombokkal és egy kisméretű színes grafikus kijelzővel kell, hogy rendelkezzen.

1. Végezzen irodalomkutatást az otthonautomatizálás témakörben, elsősorban az IoT (Internet of Things) területre koncentrálv! Nézzon utána, hogy jelenleg milyen, a megtervezendőhöz hasonló eszközök kaphatók és hasonlítsa össze azokat ill. a megtervezendő eszközt, tudás és ár szempontjából.
2. Készítse el a modul részletes specifikációját!
3. Készítse el a modul hardver rendszertervét!
4. Készítse el a modul szoftver rendszertervét!
5. Készítse el az eszköz deszkamodelljét vagy egyszerűsített prototípusát és írja meg a működtető program legfontosabb részeit!
6. Valósítson meg internetes kommunikációt az elkészített eszközzel!

Tanszéki konzulens: Dr. Benesóczky Zoltán, mestertanár

Külső konzulens: Martonicz Dániel

Budapest, 2015. március 12.

.....
Dr. Jobbágy Ákos
tanszékvezető



Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Méréstechnika és Információs Rendszerek Tanszék

Emri Balázs

INTELLIGENS TERMOSZTÁT TERVEZÉSE

KONZULENS

Dr. Benesóczky Zoltán

BUDAPEST, 2015

Tartalomjegyzék

Összefoglaló	8
Abstract.....	9
1. Bevezető	11
2. Internet of things.....	12
2.1 Intelligens ház	12
2.1.1 Az intelligens ház fogalma	12
2.1.2 Kommunikációs lehetőségek az eszközök között.....	12
2.2 Otthonautomatizálás internetre kötött eszközök segítségével	17
2.3 Más intelligens funkcionalitással rendelkező termosztátok.....	19
2.3.1 Honeywell RTH959OWF	19
2.4.2 NEST	20
2.4 Saját termék elhelyezése az alternatívák között.....	21
2.4.1 Machine learning – gépi tanulás	21
2.4.2 Ember-gép interfész	22
2.4.3 Piaci ár	22
3. A modul részletes specifikációja.....	24
3.1 Az eszköz által megvalósított funkciók	24
3.2 Beviteli eszközök	24
3.3 Kimeneti eszközök a felhasználó irányába	24
3.4 Szabályzás megvalósítása	25
3.5 Hőmérséklet mérése.....	26
3.6 Internetes kommunikáció.....	26
3.7 Eszköz működési feszültségtartománya	26
4. Hardveres megvalósítás.....	27
4.1 A hardver blokkvázlata	27
4.2 EMW3165 Wi-Fi modul	28
4.2.1 Az STM32F411 mikrokontroller	29
4.2.2 Broadcom Wifi chip.....	29
4.3 Hőmérséklet és páratartalom mérése	30
4.4 Si1143 - optikai kézmozdulat érzékelés	31
4.5 Cypress CY8CMBR2044 kapacitív touch kontroller	34

4.6	Kijelző – KD026FM	35
4.7	FT800 grafikus chip	36
4.8	Power Supply tervezés	37
4.9	Kapcsoló relék	38
4.10	A nyomtatott áramkör tervezése	39
5.	Szoftveres megvalósítása	41
5.1	Beágyazott operációs rendszer.....	41
5.1.1	A FreeRTOS beágyazott operációs rendszer	41
5.1.2	A taszkok állapotai.....	42
5.1.3	Létrehozott taszkok és kapcsolat közöttük	42
5.1.4	Interrupt szubrutinok.....	45
5.2	Optikai kézmozdulat érzékelést megvalósító szoftvermodul	46
5.2.1	A mérés elve	47
5.2.2	A szoftvermodult megvalósító fontosabb függvények	50
5.3	Kijelzőt vezérlő szoftvermodul.....	50
5.3.1	SPI.....	51
5.3.2	FT800 inicializálása	51
5.3.3	Képek megjelenítése a kijelző	52
5.3.4	További grafikai objektumok megjelenítése a kijelző	53
5.3.5	A szoftvermodult megvalósító fontosabb függvények	55
5.4	Az internetes kommunikációt megvalósító szoftvermodul	55
5.4.1	TCP/IP	55
5.4.2	Wifi	56
5.4.3	A szoftvermodult megvalósító fontosabb függvények	57
5.5	Fejlesztést támogató szoftverek	57
5.5.1	SourceTree / BitBucket.....	57
5.5.2	WICED-SDK fejlesztőkörnyezet.....	57
6.	Szerver oldali program.....	59
7.	Elkészült munka értékelése és továbbfejlesztési lehetőségek.....	61
7.1	Eddig elkészült munka.....	61
7.1.1	A szoftver tesztelése	62
7.2	Továbbfejlesztési lehetőségek	63
8.	Köszönetnyilvánítás	64
9.	Ábrajegyzék.....	65

10.	Irodalomjegyzék.....	67
------------	-----------------------------	-----------

HALLGATÓI NYILATKOZAT

Alulírott **Emri Balázs**, szigorló hallgató kijelentem, hogy ezt a diplomatervet meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző(k), cím, angol és magyar nyelvű tartalmi kivonat, készítés éve, konzulens(ek) neve) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálózatán keresztül (vagy hitelesített felhasználók számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik. Dékáni engedéllyel titkosított diplomatervek esetén a dolgozat szövege csak 3 év eltelte után válik hozzáférhetővé.

Kelt: Budapest, 2015. 12. 13.

.....
Emri Balázs

Összefoglaló

Diplomatervem első részében ismertetem az irodalomkutatást az otthonautomatizálás témakörében, elsősorban az internetre csatlakoztatott eszközök terén. Bemutatom, hogy milyen lehetőségek adódnak számunkra, ha házunkat úgynevezett okosházzá alakítjuk ki, milyen előnyökkel jár ez a mindennapi élet szempontjából. A feladatom az okosház fejlesztésében egy intelligens termosztát tervezése. Mivel más eszközök is vannak már ilyen téren, törekszem a versenyképességre és az árra. Ár és funkciók terén összehasonlítást végzek más eszközökkel. Ezt követően ismertetem, milyen funkciókat kell megvalósítania az általam tervezett eszköznek. Ismertetem az intelligens termosztát fejlesztéséhez felhasznált chipeket, illetve az ezek által nyújtott funkciókat. Ismertetem a mikrokontroller vezérlőprogramját és elkészítésének eszközeit. Bemutatom az internetes kommunikáció megvalósításának eszközét és a programot. Az eszköz vezérlőprogramja, a PC-n futó program, a kapcsolási rajzok és a NYHL terve a diplomaterv mellékleteként található meg.

Abstract

In the first part of the thesis I present literature research on the topic of home automation, particularly in the field of devices connected to the Internet. I present what opportunities arise for us if our house is formed by so-called smart house, the benefits of this in terms of everyday life. My task of a smart home is development of an intelligent thermostat. Since there are already other devices such field, striving for competitiveness and price. Price comparisons are made in different functions and tools. Then, I present what functions should my thermostate realize. I describe the development of chips used in smart thermostat and the features provided by them. I will present the tools and the preparation of the microcontroller firmware. Introduce the Internet communication tool for the implementation of the program. The device control program, the PC-side program, the schematic diagrams and the PCB design are uploaded to the Thesis's attachment.

1. Bevezető

Manapság egyre elterjedtebbek a világon mindenhol az úgynevezett „okos házak”. Az okos otthonok kényelmes környezetet teremtenek a családok részére, mindemellett biztonságossá és energiatakarékosná teszik a házat. Például ha téli pihenésből jön haza a család, a termosztátot beállíthatjuk az okos telefonról is, hogy hazaérve kellemes hőmérséklet fogadjon. Nyáron a zárt árnyékolók megakadályozzák a helyiségek felmelegedését, nincs szükség a klíma egész napos használatára. Éjszaka pedig a redőnyök, vagy a tetőtéri ablak szellőztetési pozíciója lehetővé teszi a hűvös levegő beáramlását. Az érzékelőknek köszönhetően a ház automatikusan működik. Ha hirtelen vihar kerekedik, a napellenző visszahúzódik és az ablakok bezáródnak. A természetes és a mesterséges világítás tökéletes összhangban variálható az ott lakók igényeit figyelembe véve létrehozhat előre beprogramozott forgatókönyveket, például nyáron, hétközben reggel 7:00 órakor a redőnyök felhúzódhatnak, az ablakok résnyire kinyílnak, a kávégép bekapcsol, megszólal a kedvenc rádióadónk stb. Mindez automatikusan. Ilyen az okos otthon, figyelembe veszi az ott lakók igényeit, könnyen programozható és bővíthető, kényelmet, biztonságot és energiamegtakarítást nyújt.

2. Internet of things

2.1 Intelligens ház

2.1.1 Az intelligens ház fogalma

Az „intelligens ház” igénye - amely reagál a lakók igényeire és elképzeléseire a világítás, hőmérséklet beállításával - már a huszadik század első felében megjelent. Az új elektronikus technológiák és ezek integrációjának a régebbi, hagyományos építési technológiákkal köszönhetően az intelligens otthon végre egy valós lehetőséggé vált.

Az intelligens ház, intelligens épület jelentése nem az, hogy az épület valóban mesterséges intelligenciaként gondolkodna. Sokkal inkább az, hogy egy komplex számítógépes rendszernek hála képes érzékelni környezetét és a belső állapotokat, és a ház különféle elektromos rendszereit úgy irányítani, hogy azok a lehető leghatékonyabban működjenek.

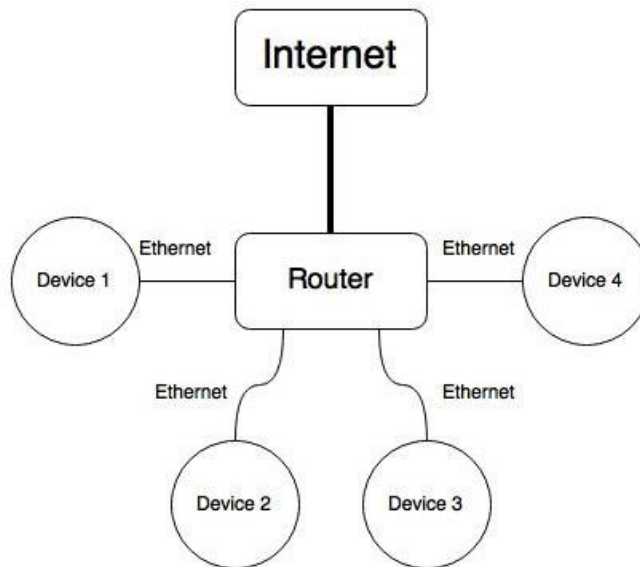
2.1.2 Kommunikációs lehetőségek az eszközök között

Fontos szempont egy okos ház rendszer kialakításánál az eszközök közötti kommunikáció. Ebből a szempontból több lehetőséget is megvizsgáltam és ennek megfelelően választottam ki az én feladatomhoz megfelelő technológiát.

- **Ethernet:**

Az Ethernet¹ név alatt általában az 1982-ben Xerox Intel által kiadott lokális hálózati technológiát értjük. A később megjelent a 802.3u szabvány a gyors Ethernet, ami 100Mbit sebességre képes.

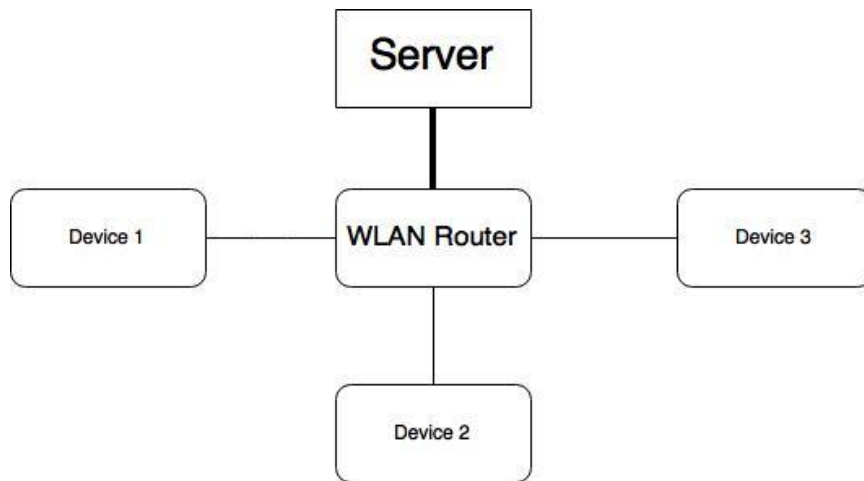
Az intelligens ház kialakítása szempontjából az Ethernet használata akkor valós lehetőség, ha a kábelezés nem okoz gondot, esetleg a ház építésekor előre gondoltak erre és a megfelelő kábeleket lefektették.



2-1. ábra Az Ethernettel megvalósított hálózat lehetséges topológiája

- Wi-Fi

A PC-vel kapcsolatosan már jól ismert Wireless LAN megoldást vizsgáltam meg, megfelel-e a vele szemben támasztott elvárásaimnak. A 802.11 szabványon alapuló, más néven WiFi (Wireless Fidelity) nagy sebességre képes (már átlagosnak számít a 300Mb/s). A vele kialakított hálózatok csillagpontosak, de persze más protokollt is meg lehet vele valósítani, de alapvetően pont-pont kapcsolatra lettek kialakítva. A csillagpont központja egy router, mely a bejövő kéréseket a megfelelő célpontnak továbbítja. Ez centralizált rendszerekhez megfelelő lehet, de figyelembe kell venni azt, hogy egy átlagos, kereskedelmi forgalomban kapható router is csak nyolc egységet tud kiszolgálni egyszerre. Ez egy háztartásban lévő PC-k számára elegendő lehet, de például hatvan eszköz számára már nem. Az eszközök hatótávolsága megfelelő egy átlagos házban belüli összeköttetésre. Nem beszélve arról, hogy elemes táplálásról, WLAN fogyasztása esetében nem igen beszélhetünk. Például egy távirányító, ami az idejének nagy részét egy töltőn tölti, nem igazán jó megoldás, praktikusság és energiatakarékosság szempontjából sem. A választási lehetőségek közül a WiFi-re esett a választásom, mivel ez ma már minden háztartásban elterjedt és megfelelő hatótávolságot biztosít. Az elemes tápellátás már a kezdetektől fogva nem szerepelt a specifikációmban, így a fogyasztásból eredő hátrányával nem kellett foglalkoznom.



2-2. ábraWiFi hálózat lehetséges topológiája

- Bluetooth:

A rövidhullámú rádiófrekvenciás kommunikációk másik széles körben elterjedt megoldása a Bluetooth technológia. A mai mobiltelefonok már ezen képesség nélkül nem is kerülnek forgalomba. A rövidtávú (időben és térben is) személyes hálózatok kialakításának legismertebb formája. Az Ericsson hozta létre és szabványosította 1994-ben. Alapvetően az RS-232 vezeték nélküli kiváltására lett kifejlesztve, ám pár év után olyan sikeres lett, hogy már nem volt elég ez az egyszerű protokoll kialakítás. A WiFi-hez hasonlóan, a 2.4GHz-es sávot használja, itt kapott 79 darab sávot, 1MHz-enként. A kezdetek óta, a modulációját is más alapokra kellett helyezni, így lett GFSK-ból, p/4-DQPSK és 8DPSK a 2.0-ás verzióban. A 2.0 + EDR verziójú IC-k 3Mb/s, a 3.0 + HS verziójúak pedig 24Mb/s sebességre képesek. Ez jóval több mint amire nekem szükségem van a feladatom megvalósításához. A fejlődése során szükségessé vált, többek között, a biztonságos kapcsolat kialakítása is, tehát a felépülő kapcsolatok mára már titkosítva működnek. A Class 1 minősítésű modulok maximálisan, nyílt területen 100 méter hatótávolságúak. A megoldásomban ez a távolság megfelelő lehetne, de sajnos az épületen belüli falak gátat szabnak a terjedésnek, nagymértékben rontva a jelminőséget. A Bluetooth topológiája szintén pont-pont megoldású, tehát ez esetben is egy komoly vezeték nélküli központ kialakítása lett volna a feladat, ami képes kezelni az eszközöket a háztartásban. Bonyolult lett volna így megoldani a feladatot, habár minden egységnél közvetlen kapcsolatot lehetett volna kialakítani például egy mobiltelefon segítségével. Ehhez viszont egy külön szoftver installálására is szükség lett volna, ami nagyban nehezítette volna fejlesztést és a kompatibilitást. A Bluetooth IC-k fogyasztása,

Class 1 esetében akár 100mW is lehet. Ez már kedvezőbb, mint a WiFi esetében volt. Tovább lépve egy olyan megoldást kerestem, ami idejének nagy részét energiatakarékos üzemmódban tudja tölteni, és a hatótávolságát sem akadályozzák az épületek falai.

A Bluetooth legközelebbi versenytársa az infravörös technológia, hiszen mindkét megoldás elsősorban az adatátvitelre koncentrál. Vannak azonban olyan helyzetek, amelyekben a Bluetooth jobb megoldás az infravörösnél, és van, amikor ez fordítva igaz. Mivel a Bluetooth képes szilárd tárgyakon áthatolni, és a piconeten belül maximális mozgási szabadságot tesz lehetővé, olyan adatátviteli lehetőségeket nyújt, amikre az infravörös technológia csak nehezen vagy egyáltalán nem képes. A Bluetooth technológiával a hangkommunikáció is lehetséges, a minden irányban létrehozható átvitelnek köszönhetően pedig, amint a mobiltelefon a számítógép hatósugarán belül ér, megkezdődhet az adatok szinkronizálása. Rengeteg adatgyűjtő és szenzor hálózat alkalmazza ezt a szabványt és nagyfokú támogatás övezi a készülékgyártók felől.

- **Zigbee:**

A Zigbee hálózatok működése a vezeték nélküli ad-hoc és szenzor hálózatok filozófiáján alapul. Az ad-hoc hálózatok nem rendelkeznek központi adminisztrációval, így elkerülhetővé válik egy fontos csomópont kiesése okozta összeomlás. Legjellemzőbb tulajdonságuk a többugrásos kommunikáció. Ezáltal a szenzorok érzékelése jóval nagyobb területre terjeszthető ki, mint a hagyományos vezeték nélküli hálózati esetekben, ahol az összes csomópontnak közvetlenül rá kell látnia a bázisállomásra. Az egyes csomópontok adott rádió-hatósugárral bírnak és mivel nem biztos, hogy a bázisállomás hatósugarában vannak egy speciális útvonalválasztó algoritmus segítségével, a többi csomóponton keresztül, érik el a továbbítandó üzenetek célpontját. Így esetünkben lehetővé válik, hogy egy távoli csomópont a saját mérési adatait a többi csomópont segítségével átküldhesse a bázisállomásra. A hálózati elemek tetszőleges időpontban kapcsolódhatnak be illetve ki a hálózati közösségbe. A topológia változására a hálózat automatikusan képes reagálni.

Egyre felkapottabbak azok a rádiós szenzor hálózatok, amelyek a ZigBee szabványon alapulnak. A ZigBee Alliance által létrehozott filozófián alapuló eszközök a célja, hogy egy alacsony költségű, flexibilis, alacsony fogyasztású, kis adatmennyiségű,

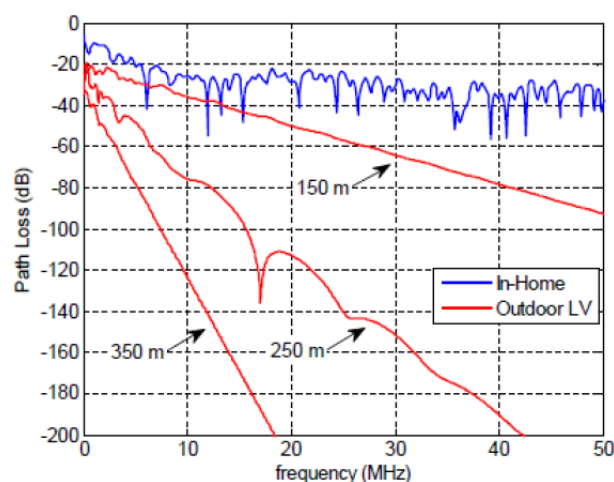
otthoni önszerveződő hálózatot tudunk a segítségével megvalósítani. Ezen eszközök több rétegű szenzorhálózati struktúrát használnak.

- Power Line Communication – Kommunikáció villamos hálózaton

A PLC² (Power Line Communication) technológia alatt az energiaátvitelre tervezett elektromos elosztóhálózaton történő adattovábbítást értjük, ami rendszerint valamilyen nagyfrekvenciás (pár kHz-től több tíz MHz-ig) jellel valósul meg. A technológia egyik legnagyobb előnye, hogy nem kell fizikailag új kommunikációs csatornát kiépíteni, hiszen a már meglévő villamosenergia-átviteli vezetékeket használhatjuk. Mivel ezeket nem adatátvitelre tervezték (nem árnyékoltak, nem illesztettek, stb.), ezért a PLC legnagyobb hátránya, illetve elterjedésének legnagyobb akadálya a technológia érzékenysége.

PLC technológia felosztása:

- Extrémén keskeny sávú (Extremely Narrow Band – ENB) PLC – nagyon alacsony adatsebesség (bps-os) kiterjedt hálózatokhoz
- Keskeny sávú (Narrow Band – NB) PLC – alacsony adatsebesség (<1Mbps) és keskeny spektrum
- Szélessávú (Broad Band – BB) PLC – nagy adatsebesség (>10Mbps) és széles spektrum

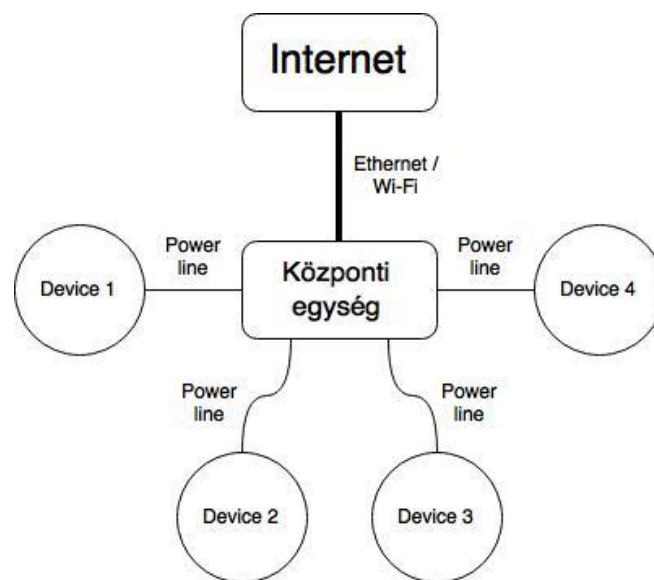


2-3. ábra Szakaszcsillapítás a frekvencia függvényében különböző vezetékhozzsok esetében³

Otthoni alkalmazás során alacsony a frekvenciatartományok közötti csillapítás-eltérés, ez széles spektrumhasználatot tesz lehetővé. Ez a kedvező szakaszcsillapítás

magasabb frekvenciák esetén a rövid vezetékhoossznak köszönhető. Kisfeszültségű szabadvezetékeken már jelentős csillapítással kell számolni, így hosszabb távolságra magasabb frekvenciák (és nagy sáv szélesség) nehezen oldható meg.

Az okosház megvalósítása szempontjából egy valós lehetőséget kínál egy házban belüli összeköttetésre. Ethernetnel szembeni előnye, hogy nem szükséges új kábeleket lefektetni és használható olyan helyen is ahol a rádióhullámok nem jól terjednek (pl. falak között). Ebben az esetben egy központi internetre csatlakozó modul valósíthatná meg az átjárást a házban belül csatlakoztatott eszközök és az internet/központi szerver között.



2-4. ábra PLC-vel megvalósított hálózat egy lehetséges topológiája

2.2 Otthonautomatizálás internetre kötött eszközök segítségével

Az elmúlt évtizedekben a legnagyobb változást hozó technológiai újítás az internet volt. Elképesztő ütemben nő a csatlakoztatott eszközök száma, ma már a fejlett államok szinte minden lakójának zsebében megtalálható egy olyan eszköz, amivel kapcsolódnak a világméretű hálózatra. Ráadásul nem csak a pusztán számok nőttek, a

csatlakoztatott eszközök is egyre több félék lettek – mobiltelefonok, televíziók, hűtőszekrények, biztonsági kamerák, és a felsorolás még hosszan folytatható lenne. Felgyorsult világunkban egyre több tevékenységet kell egyre gyorsabban, hatékonyabban elvégezni. Így van ez a ház körül is. Nem véletlen, hogy a legtöbb ember szereti a legtöbb feladatot automatizálni, különösen a rendszeres, monoton, korábban fizikai munkát igénylő, és nem utolsó sorban nem túl izgalmas tevékenységeket. Ez persze sokszor részben lustaságból fakad, részben „játék”, de gondoljunk bele, mennyi mindent kezdhünk a felszabaduló időben: több figyelmet fordíthatunk családunkra, gyermekeinkre vagy hobbinkra. Úgy tűnik, beteljesülhet az Ericsson jóslata az 50 milliárd hálózatra csatlakozó eszközről 2020-ig, és miszerint „minden csatlakozni fog az internetre, ami hasznos, ha csatlakozik”.

Az Internet of Things fogalom arra az átalakulásra utal, hogy az interneten túlsúlyba kerülnek a gép által felvitt adatok, ellentétben az ember által készített tartalommal– ehhez persze szükséges az adatgyűjtés problémájának megoldása illetve az eszközös széles körű internetre csatlakoztatása.

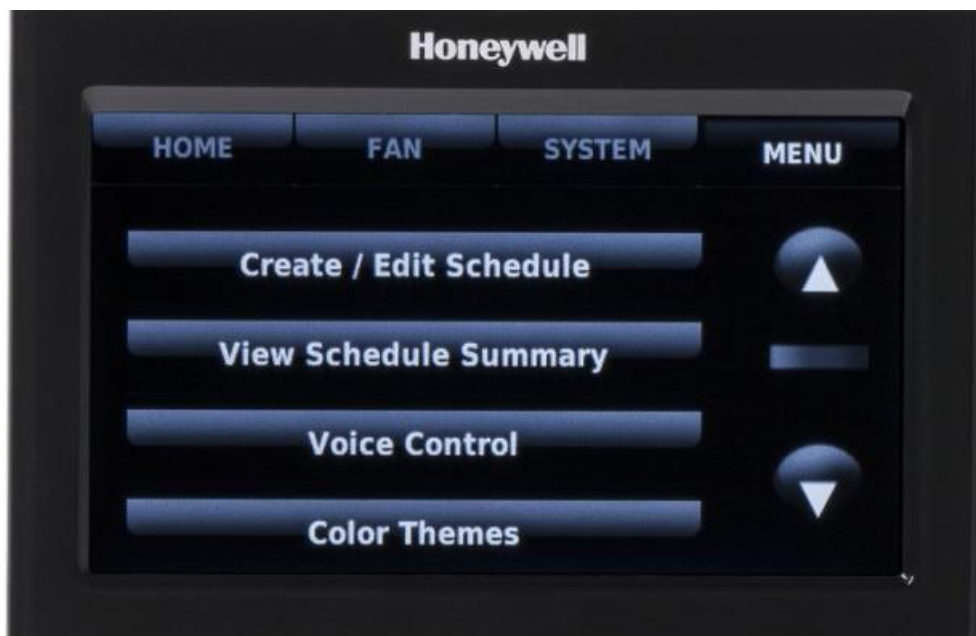
Ezek a folyamatok különös jelentőséggel bírnak az otthonautomatizálás területére nézve. Nem mehetünk el persze amellett, hogy ezek a technológiák felvetik a magánszféra védelmének problémáját és más problémákat is– de úgy gondolom, hogy idővel ezek meg lesznek oldva, a kérdés csak az, tudunk-e majd mit kezdeni az így létrejött információhalmazzal. Az egyik sokat ígérő felhasználó pedig pont az otthonunkat átszövő technológiai hálót rendszerbe szervező automatizációs megoldások köre. A házban található eszközök internetre csatlakoztatása nem csak irányíthatóság szempontjából fontos, hisz a csatlakoztatott eszközök nem feltétlenül, mint buta végpontok ülnek a rézdrót végén – képesek hozzáférni az interneten található információhalmazhoz, felhasználva azt működésükhöz és az emberrel való kommunikációjuk során, valamint akár cselekedni és képesek a felhasználó helyett. Például a hűtőszekrény megrendeli a terméket helyettünk vagy a termosztátunk tudja mikor jövünk haza a munkából és ennek megfelelően szabályozza a hőmérsékletet.

2.3 Más intelligens funkcionalitással rendelkező termosztátok

Diplomatervemnek ebben a fejezetében megvizsgálom a már meglévő, piacon kapható eszközöket vizsgálom meg, a következő fejezetben pedig bemutatom az én megoldásom miben tér el vagy egyezik meg. Az eszközök személyes kipróbálására nem adódott lehetőségem, internetes fórumokra és cikkekre támaszkodtam az alternatívák áttekintésekor.

2.3.1 Honeywell RTH959OWF⁴

Az RTH959OWF a Honeywell cég által fejlesztett intelligens termosztát.



2-5. ábra Honeywell RTH959OWF

- Beviteli eszközök: hangvezérlés és érintőképernyő
- 300\$-os ár
- Távolról is irányítható PC-ről vagy okostelefonról
- Az internetes kapcsolatot WiFi-n keresztül éri el

Egy programozható termosztát kényelmesebbé és energiatakarékosabbá teheti otthonunkat, azzal, hogy automatikusan szabályozza a hőfokot. Például magasabbra állítja a légkondicionálót, amíg alszunk, és kikapcsolja, ha elmegyünk otthonról. Ezek

az előnyök azonban csak akkor használhatók, ha a termosztát könnyen beállítható. A Honeywell új programozható termosztátja is ilyen: egyszerre felhasználóbarát, és innovatív.

A Honeywell 300 dolláros RTH959OWF modellje távolról is irányítható, okostelefonon vagy számítógépen keresztül. Az érintőképernyőt a színekkel kontrasztossá tehetjük, használata pedig könnyen, intuitívan elsajátítható. A Wi-Fi beüzemelése és a szükséges alkalmazás letöltése sem okoz gondot. Ez a vállalat első hangvezérléses termosztátja, ez a funkciója is logikusan, megfelelően működik.

2.4.2 NEST⁵

A Nest Labs nevű céget - amely a házak és lakások fűtését automatikusan vezérlő eszközéről ismert - 2014-ben vásárolta fel a Google. Az ő termékül a Nest-termosztát, amely intelligens tanuló funkcióval is rendelkezik. Megtanulja, mikor kelnek és mikor fekszenek használói, és ehhez igazodva állítja a fűtést. Miután észleli, mikor szoktak felébredni a háziak, ehhez igazodva állítja reggelente magasabbra a hőmérsékletet. Ha elmennek otthonról, automatikusan lejjebb veszi a hőmérsékletet - vagy nyáron kikapcsolja a klímát, még ha ezt el is felejtették megtenni a lakók. A fűtés és hűtés szabályozása okostelefonról is lehetséges a távolból. A gyártó szerint csupán azzal, hogy automatikusan leveszi a fűtést vagy a klímát, amikor arra nincs szükség, húsz százalékkal képes csökkenteni a költségeket.

- A meglévő 1db termosztát lecserélésével biztosítja a kazán indítását, és leállítását (ki/be kapcsoló kontakt jel)
- Csak fűtési üzemmódot kezel
- A kapcsolt relé kimenetek száma 1 db, tehát kazánnak tud csak indítójelet adni
- Egységenként, azaz lakásonként /szintenként 1 db termosztáttal számol
- Az internetes kapcsolatot WiFi-n keresztül éri el
- Öntanuló szoftverrel rögzíti a felhasználó által beállított hőmérsékleti értékeket, és egy bizonyos tanulási idő eltelte után magától állítja be a hőmérsékletet

- A be- és kikapcsolások között eltelt időből számol körülbelüli energiafogyasztást
- Saját mobil applikáció segítségével telefonról is elérhető a termosztát
- 249\$-os ár (Európában 179£)
- Véleményem szerint jó design



2-6. ábra Nest-termosztát és az okostelefon alkalmazás

A leírásban benne vannak az előnyei és hátrányai is. Mindenképp hátrány, hogy csak egy kazánnak tud jelzést adni és nem lehet minden szobába telepíteni, de mindenképpen egy profi eszközzel van dolgunk vele egy nagy háttérű cégtől.

2.4 Saját termék elhelyezése az alternatívák között

Az általam tervezett eszköz fűtési és hűtési funkciókat egyaránt képes kezelni.

2.4.1 Machine learning – gépi tanulás

A gépi tanulás a Mesterséges Intelligencia (MI) egyik ága, olyan rendszerekkel foglalkoznak, melyek tanulni képesek, azaz tapasztalatokból tudást generálnak. A gyakorlatban ez azt jelenti, hogy a rendszer példa adatok, minták alapján képes önállóan, vagy emberi segítséggel szabályszerűségeket/szabályokat felismerni/meghatározni. A rendszer tehát nem csupán betanulja „kívülről” mintákat, hanem képes ezek alapján olyan általánosításra, ami alapján – a tanulási szakasz végeztével – ismeretlen adatokra vonatkozólag is „helyes” döntéseket tud hozni.

Erre az általános célra sokféle algoritmus áll rendelkezésre, amiket a konkrét feladattól függően érdemes kiválasztani.

Az eszköz a továbbfejlesztése során szintén képes lesz a felhasználó szokásait lekövetni a gépi tanulás segítségével. Ennek megvalósítására a szerveren kell futtatni azt az algoritmust, amely ezt a gép tanulást elvégzi, lekezeli és a megfelelő utasításokat elküldi az interneten az általam tervezett termosztátnak. Ilyen felhasználói szokás az, amikor a felhasználó minden nap ugyanakkor megy munkába, ekkor az eszköznek le kell vennie a fűtést és mire visszatérne addigra újra befűteni a házat. Mivel ez egy erőforrás igényesebb feladat lenne, ezért ennek a programnak a futtatása mindenképp a szerveroldalon célszerű futtatni, nem pedig a mikrokontrollerrel rendelkező eszközömon.

2.4.2 Ember-gép interfész

Kapacitív touch és az optikai kézmozdulat érzékelés együttese egy olyan dolog, amely felkeltheti a felhasználó érdeklődését az eszköz iránt. Bár először kicsit felesleges funkcióknak tűnnek ezek, gondoljunk bele, ha valaki meglátja a barátjánál ezt a fajta beviteli környezetet, lehet neki is igénye lesz rá. Véleményem szerint egész márkák épülnek a user-experience javító funkciókra, amelyek feleslegesnek is tűnhetnek, de mégis igénye van a felhasználónak rá (pl. újlennyomat érzékelés a telefonokban).

2.4.3 Piaci ár

A terméket a piacra szánjuk, ezért fontos elvégezni az eszköz költségeinek a kiszámítását.

Alkatrész	Ár (USD)
Kapacitív touch szenzor	1,1
Grafikus chip	4,1
Mikrokontroller + Flash memória + Wi-Fi chip	7,8
Kijelző	7
Kapcsoló relék (2db)	2x1,5
Gesture szenzor + hőmérséklet mérő szenzor	3+3,6

Táp IC	0,8
4 rétegű NYHL gyártása (kínai ár – nagyobb darabszám)	2,5

2-7. ábra Az eszköz alkatrészeinek költségei

Ezek a számok összesen 33 USD. Ehhez jönnek még:

- összeszerelés költsége
- műanyag 'ház' költsége
- support fenntartásának költsége
- elromlott eszközök javítása / cseréje
- dobozolási költség
- egyéb költségek.

Egy terméket, hogy nyereségesek legyünk legalább az előállítási költség kétszereséért kell adni. Ha 50 USD árral számolunk, akkor a legolcsóbb eladási költség 100 USD. Ez azt hiszem, a korábbi fejezetben bemutatott termosztátokkal összehasonlítva teljesen korrekt, olcsóbb a termékünk.

3. A modul részletes specifikációja

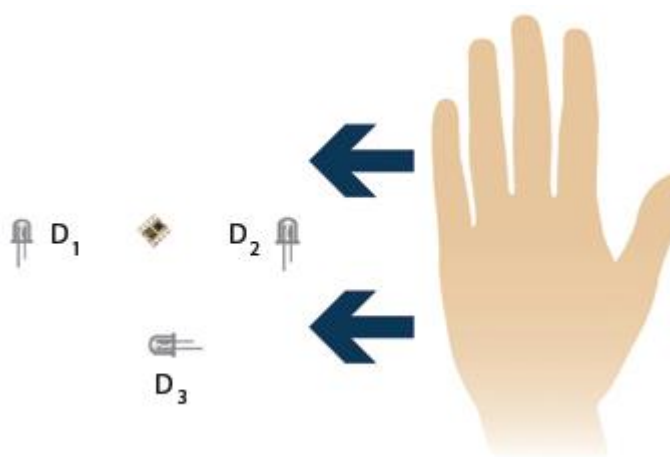
3.1 Az eszköz által megvalósított funkciók

Az általam tervezett eszköznek egy átlagos termosztáttól eltérően többféle funkciót kell megvalósítania. Mivel eladható terméket szeretnék, ezért a felhasználói élmény nagy hangsúlyt kap.

3.2 Beviteli eszközök

Mindenképpen előnyös valamilyen –egy egyszerű mechanikus nyomógombtól különlegesebb- beviteli felület. Ezért választottam napjainkban egyre divatosabb kapacitív touch gombokat és optikai kézmozdulat érzékelést. Ez a technológia sokkal érzékenyebb, mint a rezisztív touch. A kapacitív touch gombokat a hőmérséklet állítására használtam.

Az optikai kézmozdulat érzékelés a jól ismert „swipe” funkcióhoz szükséges. Ez jobbra-balra és fel-le érzékelést valósít meg nálam. A kézmodulat érzékelést az eszközönnél a menüben való navigálásra használtam (ebbe beletartozik a hőmérséklet beállítása is).



3-1. ábra Swipe mozdulat balra

3.3 Kimeneti eszközök a felhasználó irányába

A felhasználó irányába egy színes – képek megjelenítésére is alkalmas – kijelzőnek kellett választani. A termosztáthoz kimeneti eszköznek egy

folyadékkristályos kijelzőt választottam. Egy 2.6 inch-es kijelző biztosít a felhasználói felületet, amelyen a kinti és benti hőmérséklet egyaránt megjeleníthető, illetve az időjáráshoz tartózkodó képet is meg tudunk rajta jeleníteni, ha szeretnénk.

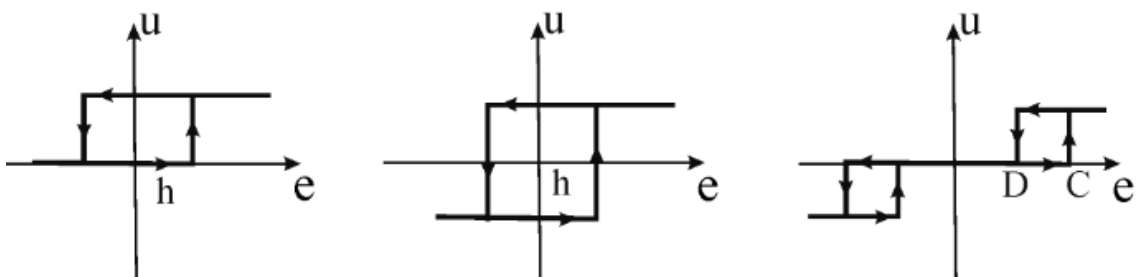
3.4 Szabályzás megvalósítása

Az alapvető termosztát funkciójaként az eszköz tartalmaz kétállású szabályzót valósítottam meg. A kétállású szabályzók nem folytonos kimenettel rendelkeznek, a kimenet két diszkrét értéket vehet fel. Alkalmazásukat egyszerű struktúrájuk és megvalósíthatóságuk, kis költségük motiválja. Lassú folyamatok esetében kielégítő szabályozást biztosítanak. Ez annak tulajdonítható, hogy a relé kimenetüknek köszönhetően a beavatkozó a teljes energiát rövid idő alatt beleviszi a folyamatba.



3-2. ábra Szabályozási rendszer többállású szabályzóval

A kétállású szabályozó a hiba függvényében két fix értéket képes kiadni ($u=0$ vagy $u=1$), ennek megfelelően a beavatkozó szerv is két üzemmódban működhet a kiadott vezérlőjel függvényében. A túl gyors kapcsolás elkerülésére a szabályozót (h szélességű) hiszterézissel módosíthatjuk.⁶



3-3. ábra Többállású hiszterézises szabályzók statikus átvitele

A kimeneti jelet relén keresztül továbbítja vezetéken az eszköz a kazán felé. A relék árama volt a fő szempont a kiválasztásuknál, legalább 0.5 Amperes áramot

képesnek kell lenniük. Az eszközt úgy terveztem meg, hogy későbbi fejlesztési lehetőségként a hűtés vezérlése is meg legyen oldva ugyanígy, ha szükség lesz rá.

3.5 Hőmérséklet mérése

A termosztát funkciója a hőmérséklet mérése. Ezzel a benti hőmérsékletet tudjuk mérni, a kinti hőmérsékletet az internetes szerverről tudjuk lekérdezni. A mérést 0.3 °C pontossággal kell végezni. Tartománya 10-45 °C.

3.6 Internetes kommunikáció

A termosztát egy szerveren keresztül fog az internetre csatlakozni. A szerveren tárolhatóak lesznek a felhasználó által beállított hőmérséklet adatai, időjárási adatokat kérdezhetünk le.

A kommunikáció követelményei között TCP/IP kapcsolat szerepelt és meghatározott keretekben küldeni az adatokat.

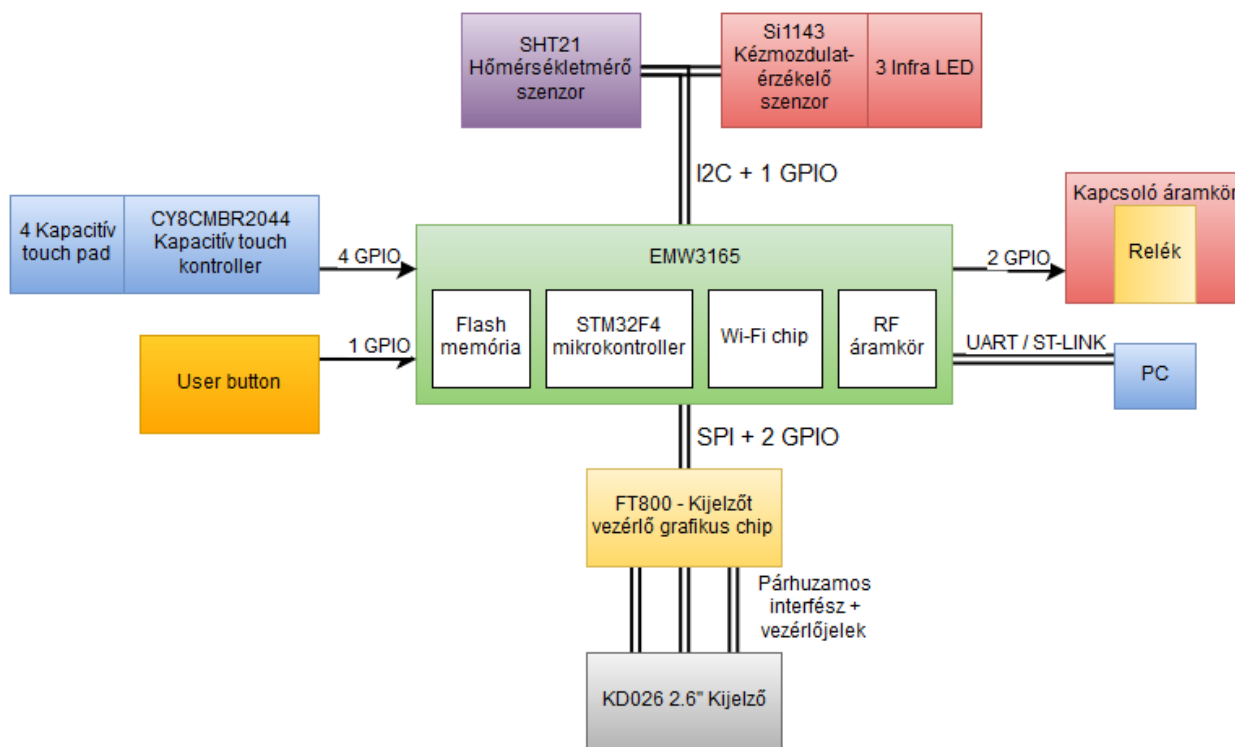
A funkció megvalósításához egy WiFi-s chippel és egy STM32 mikrokontrollerrel rendelkező eszközt vásároltam, kedvező árral rendelkezik.

3.7 Eszköz működési feszültségtartománya

A tápellátást úgy kellett tervezni, hogy 5-12V feszültségtartományon működőképes legyen. Ez egy normál hálózati adapter kimeneti feszültségének felel meg.

4. Hardveres megvalósítás

4.1 A hardver blokkvázlata



4-1. ábra A hardver blokkvázlata

A mikrokontroller SPI buszon keresztül kapcsolódik a kijelzőt vezérlő grafikus chip-hez. A grafikus chip-ben tárolható kép, így nem kell minden egyes alkalommal kiküldeni a soros buszon, a képre pedig gombok rajzolhatóak.

A grafikus chip párhuzamos buszon csatlakozik az eszközhöz, ez legalább 60 Hz-es képfrissítési sebességet biztosít.

Az eszközre tettem egy felhasználói gombot, ami az eszközön – összeszerelés után – csak tűvel lehet benyomni, ennek a funkciója még nincs specifikálva, valószínűleg valamilyen reset funkciót fog megvalósítani.

I²C buszon kapcsolódik az eszköz a szenzorokhoz, amelyek különböző címeken érhetőek el. A busz megfelelő sebességet biztosít az adatok kiolvasásához.

A számítógéppel történő programozáshoz ST-LINK/V2 debugger-t / programozót használtam, ennek a csatlakozóját is.

A számítógépes adatkapcsolathoz egy UART perifériát is felhasználtam, amit egy egyszerű UART – USB átalakítóval használtam is a gépemen.

4 GPIO lábon csatlakozik a kapacitív touch kontroller a mikrokontrollerhez – értelemszerűen a 4 vezeték egy-egy pad érintésekor alacsony feszültség szintre vált.

2 GPIO lábon történik a kapcsoló relék vezérlése. Egyikkel a fűtés vezérlése történik, a másikkal a hűtés vezérlése fog történni.

4.2 EMW3165 Wi-Fi modul

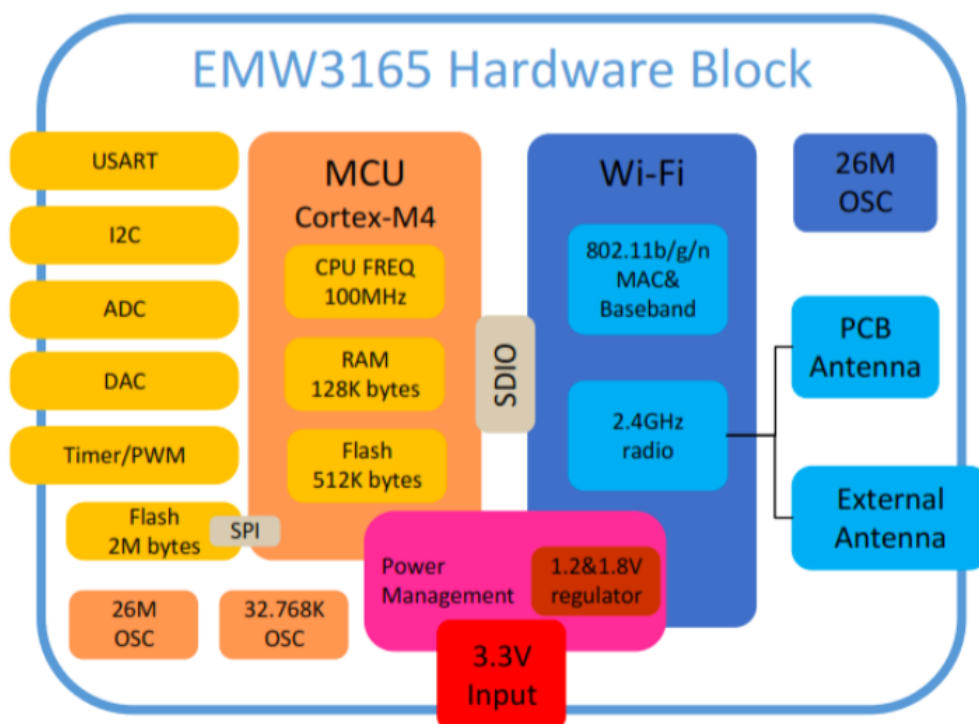
EMW3165⁷ egy alacsony energiaigényű beágyazott Wi-Fi modul, amely magában foglal egy WLAN chipet és egy STM32F411CE 32 bites mikrokontrollert.



4-2. ábra EMW3166

A modul jó választásnak bizonyult, hiszen a mikrokontroller megvalósítja a szükséges perifériákat és a Wi-Fi chip vezérlését, továbbá beépített oszcillátorok is vannak az eszközön, 2 Megabájtos-os SPI-on a mikrokontrollerre csatlakoztatott flash memória és PCB antenna.

Megjegyzem: azért is előnyös ennek a modul CE jelöléssel ellátott, így könnyebben kereskedelmi forgalomba kerülhet a termék, ha ugyanis nekünk kellene (a rádiófrekvenciás alkalmazás miatt) engedélyeztetni az eszközt az nem lenne elég olcsó.



4-3. ábra EMW3165 blokkvázlata

4.2.1 Az STM32F411 mikrokontroller

Az STM32F411 100MHz-es Cortex®-M4 maggal ellátott mikrokontroller. Rendelkezik az összes szükséges perifériával (I2C, SPI, GPIO, SDIO, USART).

4.2.2 Broadcom Wifi chip

A chip által támogatott szabványok:

Szabvány	Moduláció	Működési frekvencia (GHz)	Sebesség (maximális/jellemző) (Mbit/s)	Hatótávolság (kültéren/beltéren) (m)
802.11b	DSSS/ CCK	2,4	11/4,3	140/38
802.11g	OFDM	2,4	54/19	140/38
802,11n	MIMO-OFDM	2,4	600/47	250/70

A szabványok közül az eszköz teszteléséhez a 802.11g szabványt használtam, amely széleskörűen elterjedt.

4.3 Hőmérséklet és páratartalom mérése

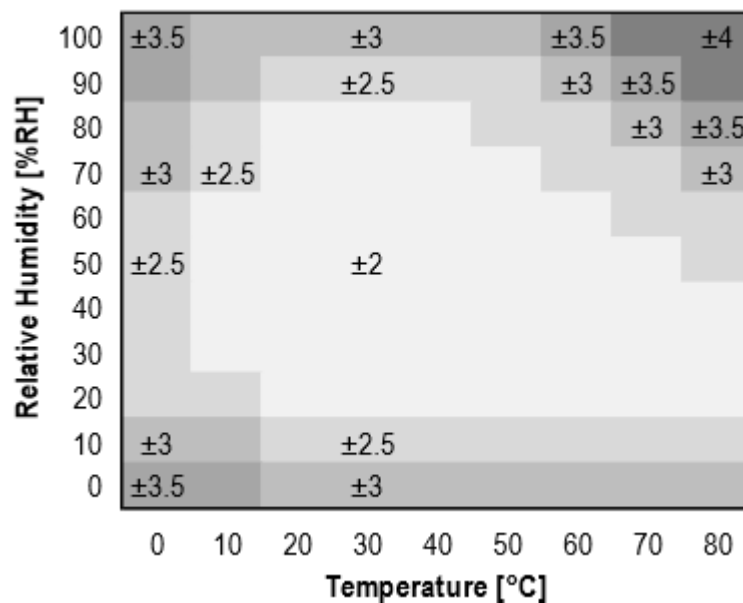
A hőmérséklet és páratartalom mérésére a Sensirion SHT21⁸ chipet választottam. Tulajdonságai:

- I²C interfész
- Nincs szükség kalibrációra
- Kicsi áramfelvétel
- -40°-tól +125°C-ig képes hőmérsékletet mérni
- 0-100%-ig képes relatív páratartalmat mérni.

A feladathoz kellő pontossággal rendelkezik a chip.



4-4. ábra SHT21 hőmérséklet és páratartalom szenzor



4-5. ábra A szenzor pontossága %-ban, a hőmérséklet és páratartalom függvényében

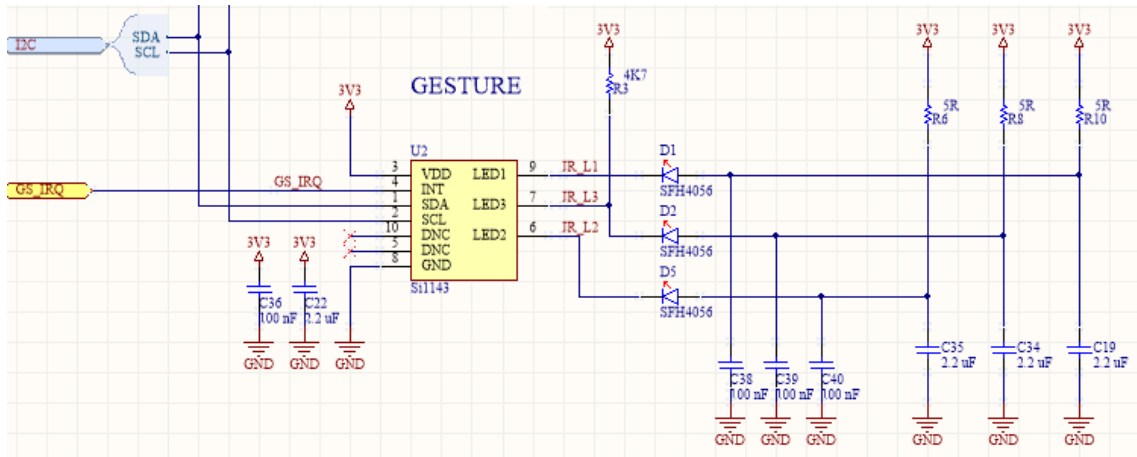
4.4 Si1143 - optikai kézmozdulat érzékelés

Az optikai kézmozdulat érzékelést a Silicon Labs Si1143⁹ chip valósítja meg 3 db SFH4056 típusú infravörös LED-del. Az érzékelés elve az, hogy a chip érzékeli az infravörös fényt, ezt mérhető adattá konvertálja, így különböző időszelletekben meghajtva a LED-eket és visszamérve a kezünkről visszavert infravörösfény-értékeket, a különbségükből meghatározható a pozíció. Mivel csak „swipe” mozdulatokra lesz használva, így a kéz pozícióinak a különbségei a számunkra lényeges adatok.

A szenzor ki- és bemenetei:

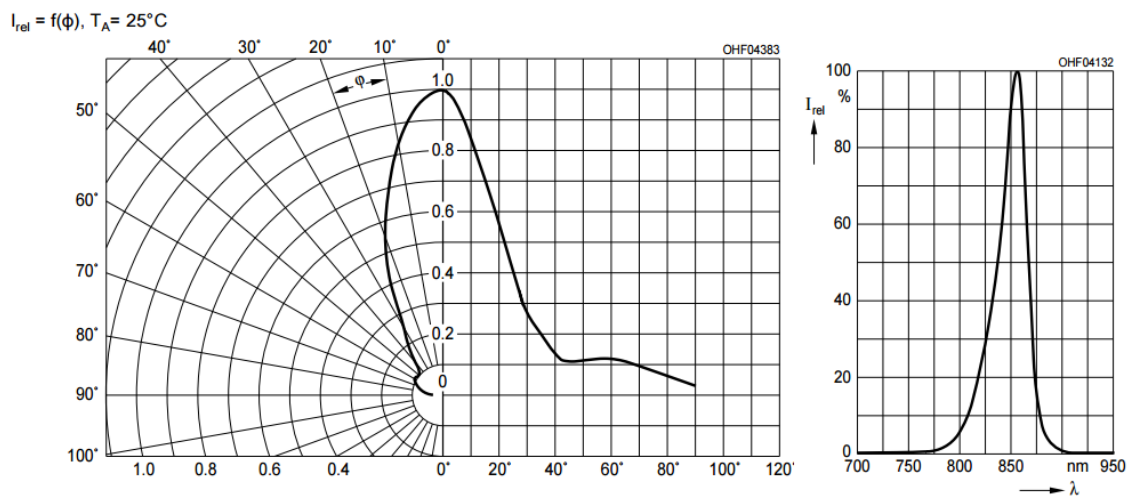
- VCC (3.3 V)
- GND
- SCL (I²C órajel vezeték)
- SDA (I²C adat vezeték)
- IRQ (Interrupt Request)
- 3 kimenet a LED-ek meghajtására.

A szenzor kapcsolási rajza látható az alábbi ábrán. A speciális infravörös LED-ek 22 Ohm-os ellenálláson keresztül csatlakoznak a tápfeszültségre (bár lehet a kapcsolási rajzon az érték megtévesztő).



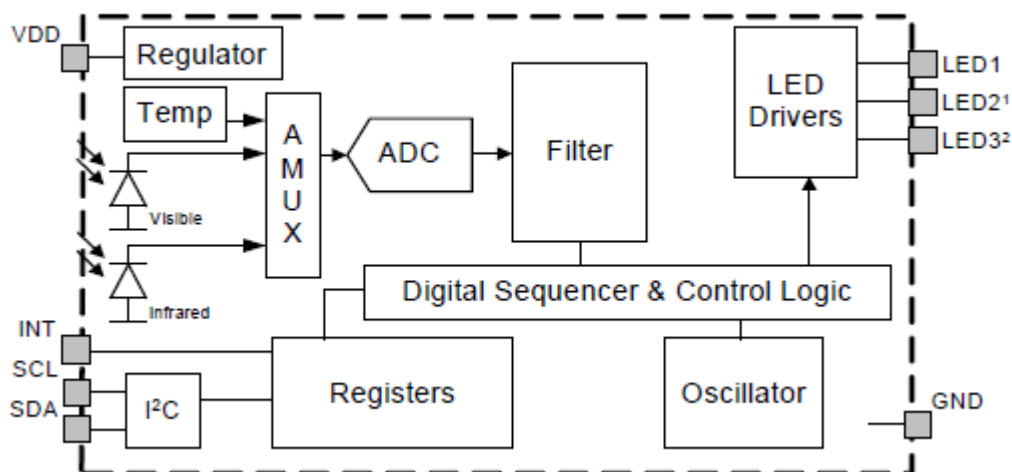
4-6. ábra Si1143 kapcsolási rajza

A LED-eknek speciális infravörös LED-eket választottam, kibocsátási hullámhossztartományuk és iránykarakteristikájuk az alábbi:



4-7. ábra Az infravörös LED-ek karakterisztikái

Hullámhosszban jól lefedik az infravörös tartományt, sugárzási szögben pedig megfelelő a működés szempontjából mivel inkább merőleges irányba összpontosítja a fényt, így nincs túl nagy veszteség.

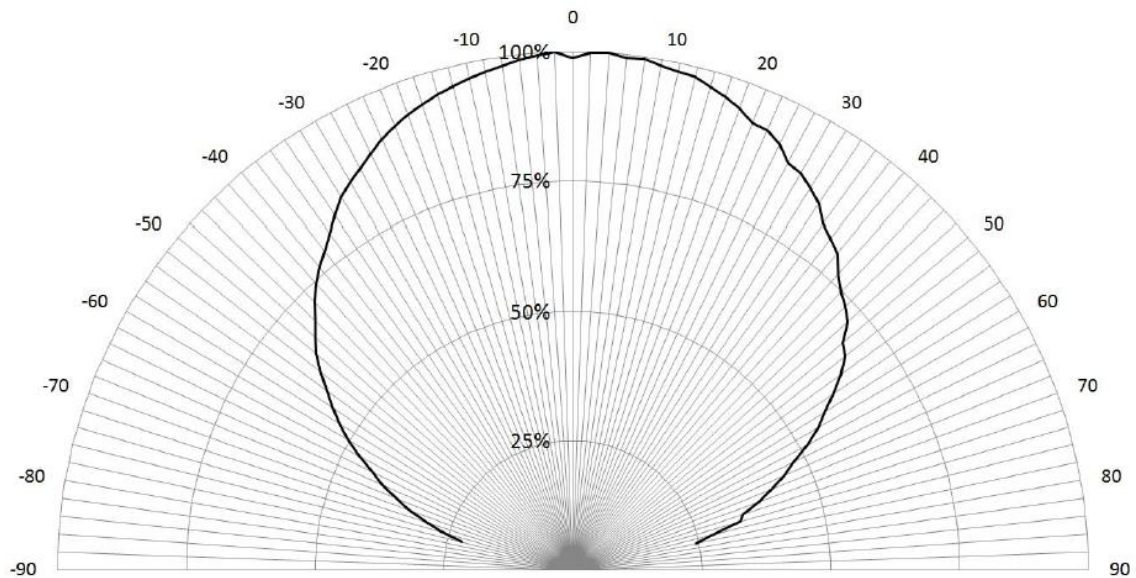


4-8. ábra Si1143 blokkvázlata

A LED-ek meghajtását egy digitális logika vezérli, a külső infravörös fényt és látható fény értékek egy analóg multiplexer ADC-en méri az eszköz. A chip képes külön-külön méréseket lebonyolítani mindhárom LED-del, illetve képes csak a környezetet mérni (látható és infravörös tartományt). A chip alkalmas 20-50 cm távolságban lévő tárgyakról visszaverődő fény mérésének a lebonyolítására. A chipben 3 fotodióda van:

- látható fényre érzékeny
- egy nagy infravörös fényre érzékeny
- egy kisebb, infravörös fényre érzékeny

Én a méréseim során a nagy, infravörös fényre érzékeny diódát használtam. A fotodiódák iránykarakterisztikája:



4-9. ábra Nagyobb, infravörös fényre érzékeny dióda vertikális karakterisztikája

4.5 Cypress CY8CMBR2044 kapacitív touch kontroller

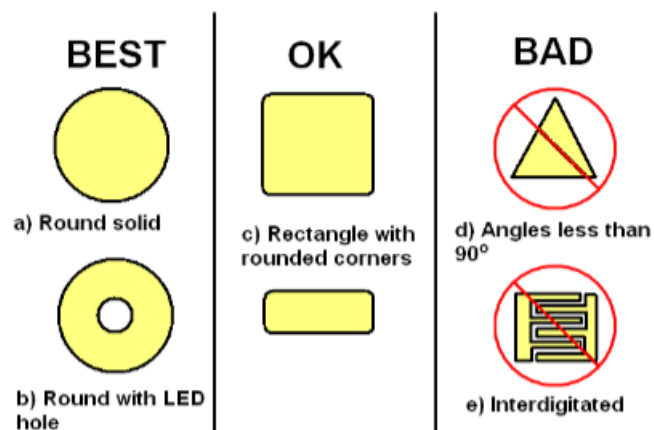
Kapacitív touch vezérlőnek a Cypress CY8CMBR2044-t¹⁰ választottam, mert:

- képes 4 kapacitív touch gombot kezelni
- nem szükséges szoftveres felprogramozás
- 4 GPIO-val könnyen kezelhető mikrokontroller oldaláról
- alacsony áramfelvétele van.



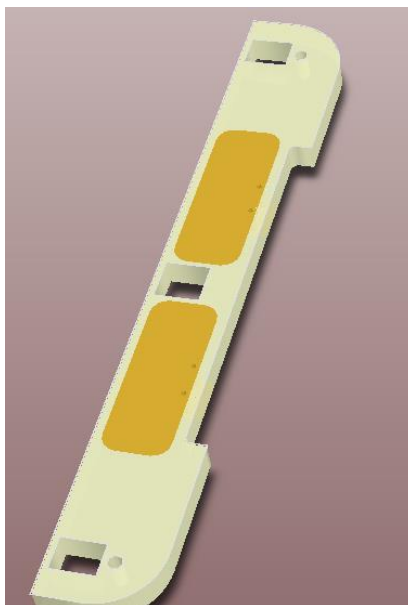
4-10. ábra Cypress CY8CMBR2044

A kapacitív touch gombokhoz a layout-on megkötések tartoznak.



4-11. ábra Kapacitív touch pad-ek

Az ábrán látható típusok közül én a c) típust választottam. Ezeket a pad-eket egy külön NYHL-re terveztem, mivel a kijelzővel lesznek egy síkban.



4-12. ábra A megtervezett kapacitív touchpad-ek

4.6 Kijelző – KD026FM

Kijelzőnek a KD026FM¹¹ kijelzőt választottam. Tulajdonságai:

- SPI és DBI (párhuzamos) interfész
- 240x320 pixel felbontás
- 39.6 x 52.8 mm méret (2.6 inch átmérő)
- 262k szín
- -20 – +70°C működési hőmérséklet

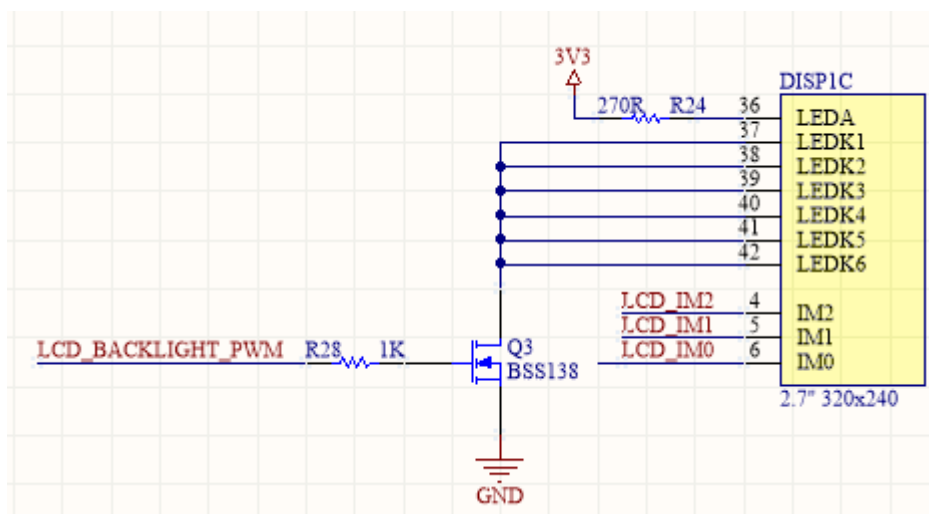
- alacsony ár.

Az interfészek közül a párhuzamos RGB interfészt használok FT800 grafikus chippel, ez gyorsabb megjelenítést tesz lehetővé, mint a SPI és mikrokontroller párosítás, ráadásul a számításokat nem a mikrokontrollernek kell elvégezni a processzoridőt pazarolva.



4-13. ábra LCD Display

A kijelzőben lévő LED-eket egy PWM jellel vezérem, aminek a kitöltési tényezőjének a változtatásával lehetséges a kijelző fényerejének állítása.



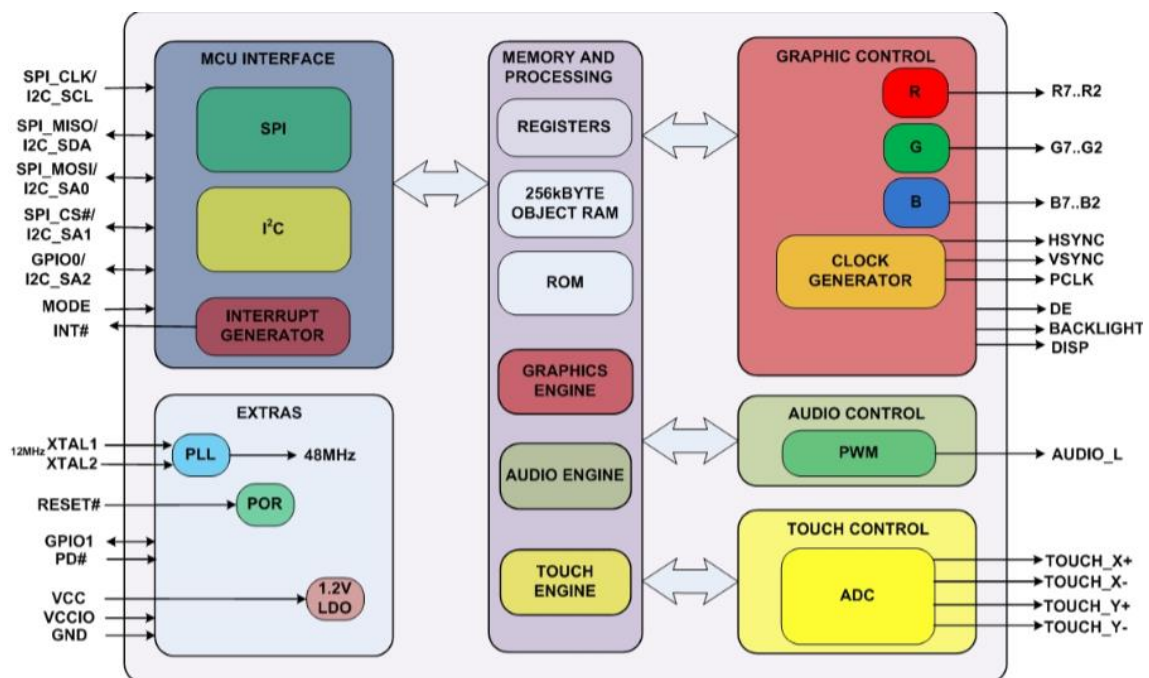
4-14. ábra LCD LED-jeinek PWM vezérlése

4.7 FT800 grafikus chip

A kijelző meghajtásához egy grafikus chip-et használtam, így lehetővé vált komolyabb grafikus alkalmazás készítése is. A lehetőségek közül az FT800¹²-ra esett a választásom, amely:

- grafikus kontrollert tartalmaz

- audio feldolgozásra képes
- rezisztív touch kontrollerrel rendelkezik
- Embedded Video Engine (EVE)
- beépített grafikus műveletek
- RGB párhuzamos kimenet
- HSYNC VSYNC kezelése
- támogatja a 480x272 (WQVGA) és a 320x240 (QVGA) felbontást
- képernyő engedélyezés (Display Enable) kimenet
- külső RAM nem szükséges hozzá
- SPI/I2C perifériával rendelkezik a host mikrokontroller felé (ezek közül az SPI-t használtam az elérhető nagyobb sebesség miatt)

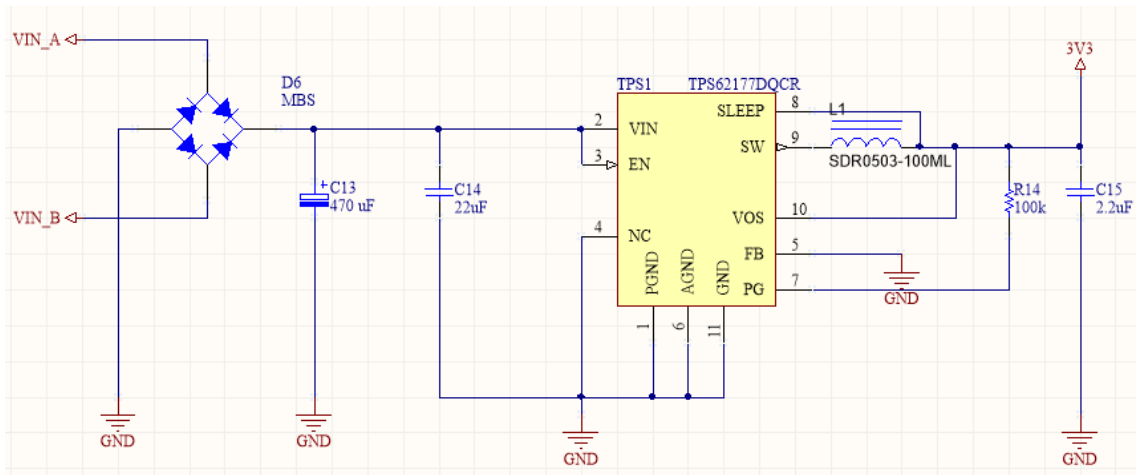


4-15. ábra Az FT800 blokkvázlata

4.8 Power Supply tervezés

A táp előállítására TPS62177DQCR IC-t választottam. Bemeneti feszültségtartománya 4.5-28V. Kapcsolóüzemű, step down converter, a jó hatásfok

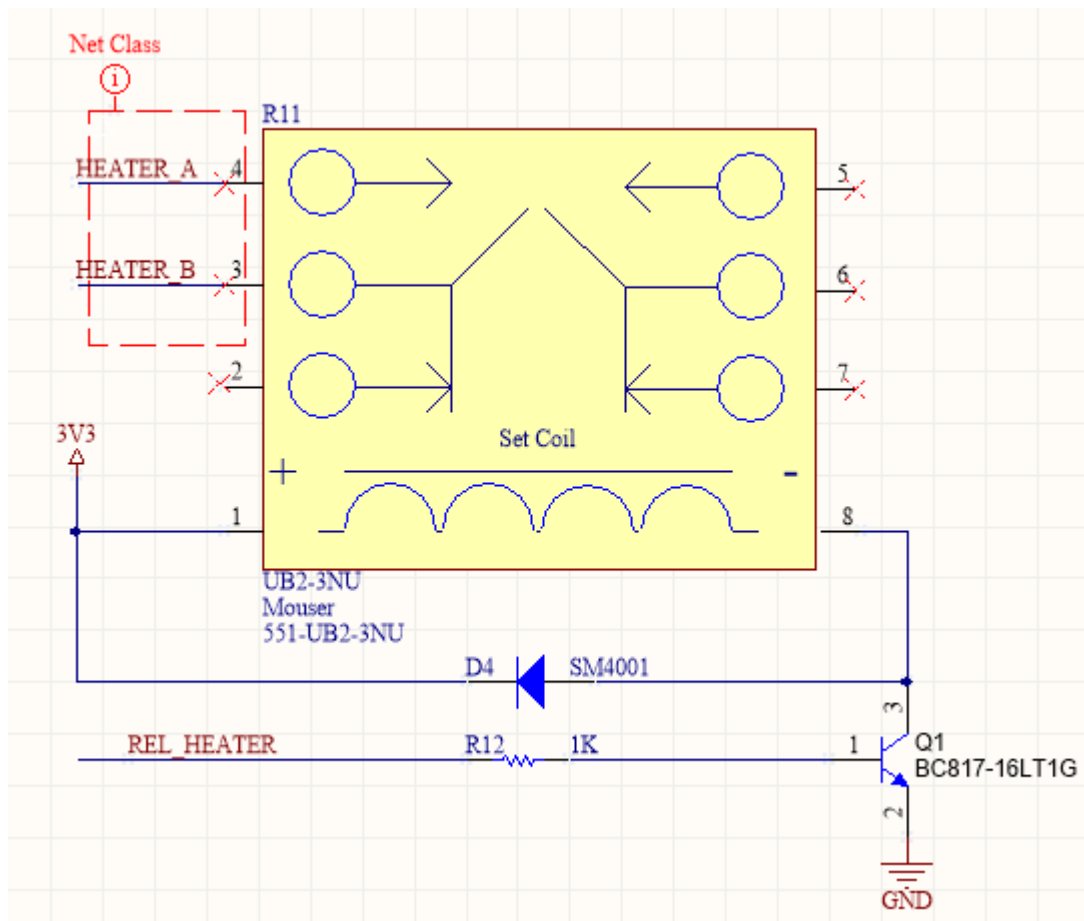
vége. A szabályozókörében szereplő alkatrészeket úgy választottam meg, hogy a kimeneti feszültség 3.3V legyen. A bemenetre egy nagyméretű 470 uF-os elektrolit kondenzátort raktam, hogy a hálózati zavarokat megfelelően szűrje. Kimeneti árama 0.5A ami megfelelően képes ellátni az eszközt.



4-16. ábra Az eszköz tápellátása

4.9 Kapcsoló relék

Az eszköz a fűtési rendszerrel kétállapotú kapcsoló reléekkel érintkezik. Ehhez a feladathoz a NEC gyártmányú UB2-3NU¹³ típusú reléket választottam. 1 Amperes áramával teljesíti a specifikációban leírtakat.



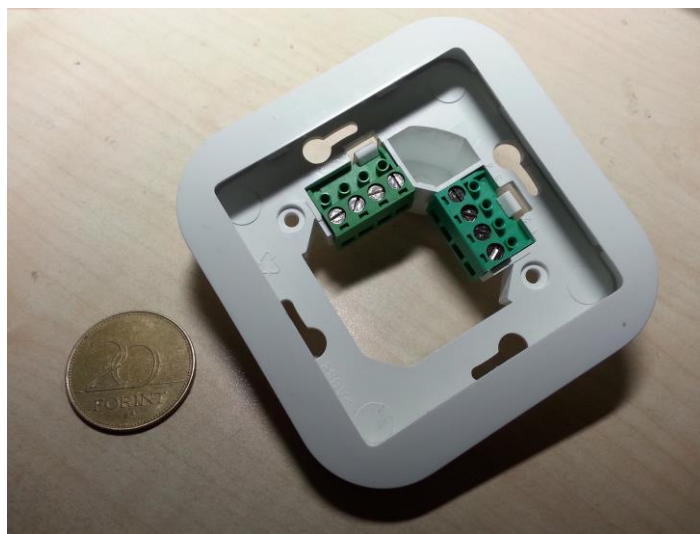
4-17. ábra A fűtést vezérlő relék kapcsolása

4.10 A nyomtatott áramkör tervezése

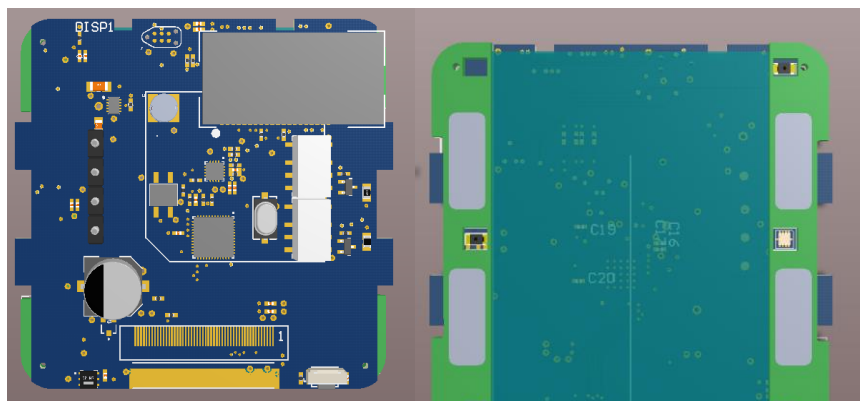
Az áramkör tervezését Altium Designer programban végeztem. A tervezés az alábbi feladatokat foglalta magába:

- Alkatrész könyvtárak elkészítése, amibe beletartoznak:
 - Alkatrészek kapcsolási rajz szimbólumainak elkészítése.
 - Footprint-ek megrajzolás.
- Kapcsolási rajz készítése
 - Hierarchikus tervezés: egy Top Level modul és hozzá kapcsolódó kisebb modulok. Ez az feladatnál nagy segítségemre volt, mivel ilyen szintű kapcsolási rajzot nehéz lenne átlátni.
- Layout tervezés:

- a Layout tervezés nem szerepelt a feladatkiírásban, de mivel folytatni akarom a jövőben is a projectet, a hardver megtervezése volt a következő feladat.
- A huzalozást 4 rétegen oldottam meg. A két belső réteg GND és 3V3 kitöltésűre készítettem.
- A huzalozásnál figyelembe kellett venni a kapacitív touch vonalak elhelyezését
- A NYHL tervét a diplomaterv mellékletébe töltöttem fel



4-18. ábra Az eszköz falra szerelhető doboza



4-19. ábra A megtervezett eszköz 3D modellje Altium Designer programban

5. Szoftveres megvalósítása

Egy ilyen feladatnál célszerűnek láttam egy beágyazott operációs rendszert használni. A szoftver különböző szoftvermodulokból áll össze. A különböző taszkok ezeket a modulokat használják a funkciók megvalósításához. Először a beágyazott operációs rendszert ismertetem, majd a benne létrehozott taszkokat. Ezután kitérek arra, hogyan valósítottam meg a szenzorok és a vezérlés szoftvermoduljait.

5.1 Beágyazott operációs rendszer

5.1.1 A FreeRTOS¹⁴ beágyazott operációs rendszer

A beágyazott operációs rendszerek piacán számos rendszer forráskóddal együtt elérhető - ilyen például a μ C/OS-II kernel - amelyek az iparban már bizonyítottak az évek során, ezért széles körben alkalmazzák. Megbízható, jól támogatott rendszert ipari alkalmazásra már lényegesen nehezebb találni, ha az ingyenes felhasználhatóság is fontos szempont. A FreeRTOS egy ingyenesen elérhető, nyílt forráskódú, valós idejű operációs rendszer (pontosabban kernel), amelyet kifejezetten beágyazott rendszerekben történő felhasználásra fejlesztenek. A FreeRTOS sem teljesítményében, sem funkcióiban nem marad el fizetős versenytársaitól köszönhetően a tapasztalt fejlesztő csapatának, számos processzorra elérhető és ingyenessége miatt várhatóan egyre többen fogják használni.

FreeRTOS kernelt valós idejű alkalmazásokhoz tervezték, felépítése a korábbi értelmezések szerint monolitikusnak tekinthető. A kernel egy preemptív prioritásos ütemezővel rendelkezik, a prioritási szinteket és az egyes szinteken létrehozott folyamatok számát kizárólag a hardver korlátozza. A modern operációs rendszerek ütemezése tipikusan ilyen. A rendszer jól skálázható és igen kompakt, a kernelből származó tárgykód mérete a felhasznált funkcióktól függően 4 Kbyte és 9 Kbyte között mozog. A forráskód nyelve a C és az Assembly, a legtöbb napjainkban használt architektúrára elérhető, amelyekhez számos demonstrációs célú példaalkalmazást mellékelnek. Az alábbiakban a fontosabb jellemzőket összefoglalva:

- valós idejű preemptív prioritásos ütemezés
- adott prioritási szinten korlátlan számú folyamat

- skálázhatóság
- 27 hivatalosan támogatott architektúra
- C nyelvű forráskód
- példaalkalmazások és részletes dokumentáció
- a legtöbb architektúrához ingyenesen elérhető fejlesztőeszközök
- különböző licencek, alapvetően ingyenes felhasználhatóság ipari célokra.

5.1.2 A taszkok állapotai

Ütemezés szempontjából a taszkok a felsorolt négy állapot valamelyikében lehetnek, szabályos állapotátmentet kizárólag újraütemezés, eseményre várakozás, illetve explicit felfüggesztés eredménye lehet.

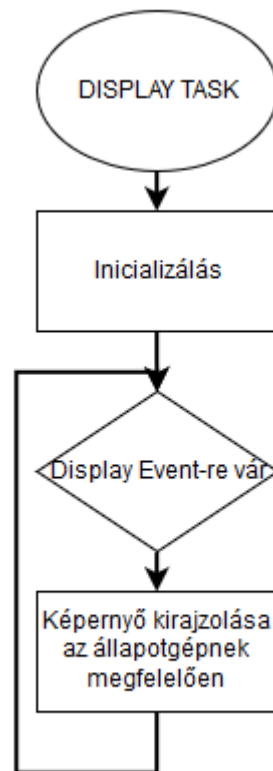
- Running: az aktuális futó taszkhoz tartozó állapot, egy taszk csak ebben az állapotában terheli a processzort
- Ready: a taszk futásra kész, a létrehozásuk után minden taszknak ez az alapértelmezett állapota
- Blocked: blokkolt állapotban akkor van a taszk, ha valamilyen esemény bekövetkezésére várakozik, ilyen esemény lehet például egy késleltetés, vagy várakozás egy szemaforra
- Suspended: a taszk futtatása explicit felfüggesztésre került egy `vTaskSuspend` hívás hatására, a felfüggesztésből a `vTaskResume` hívás segítségével lehet a taszkot újra futásra kész állapotba juttatni

5.1.3 Létrehozott taszkok és kapcsolat közöttük

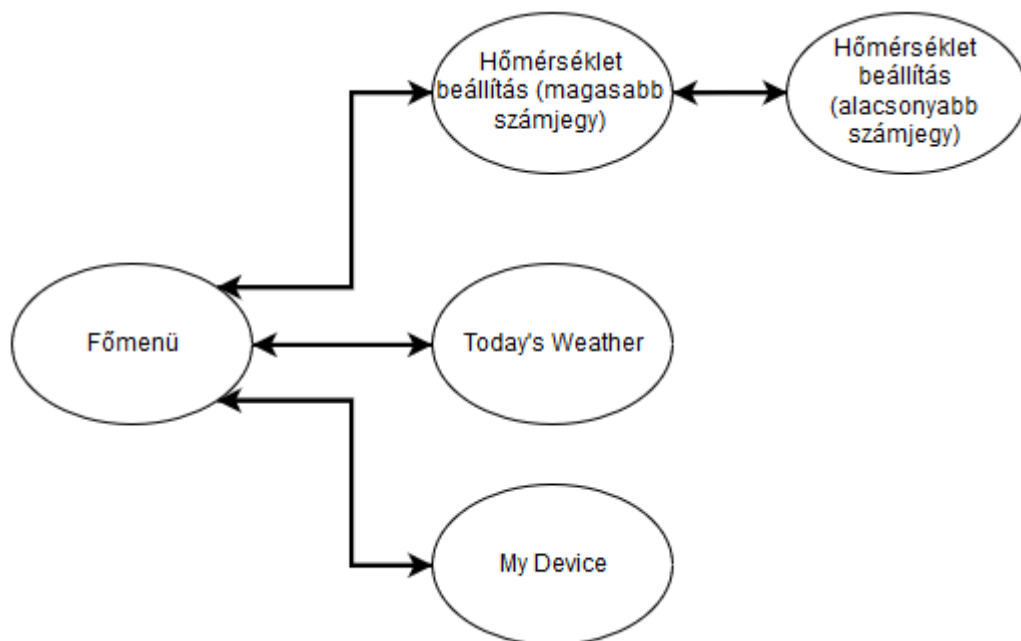
5.1.3.1 Display taszk

A Display taszk a kijelző frissítéséért és kirajzolásáért felel. A taszk az inicializálás után egy eseményre várakozik (`xEventGroupWaitBits(...)`). Ezt az eseményt a Gesture taszk és egy –a kapacitív touchpad-eket kezelő ISR- tudja aktiválni.

Az esemény bekövetkezése után a képernyőt frissíti, majd újra várakozik az eseményre.



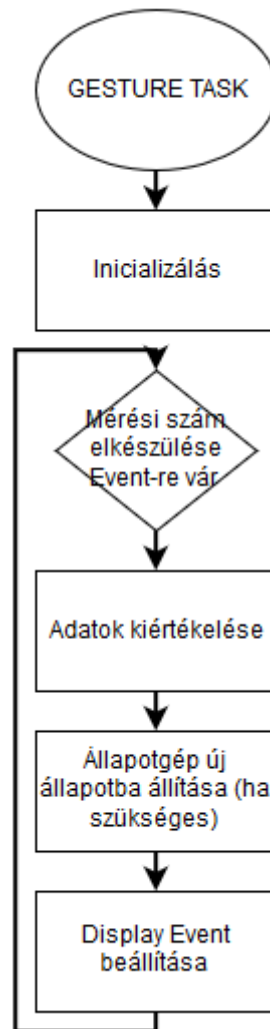
5-1. ábra A Display taszk folyamatábrája



5-2. ábra Menük és állapotátmeneteik

5.1.3.2 Gesture taszk

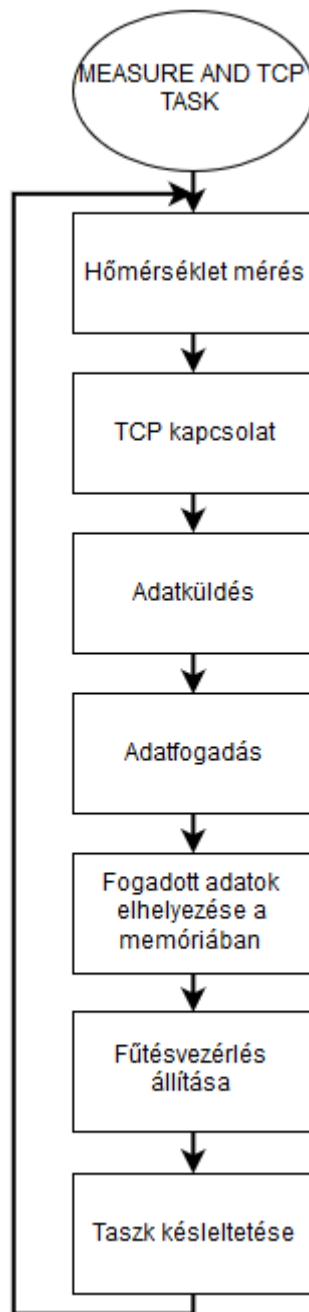
A Gesture taszk a kézmozdulat érzékelés kiértékeléséért és ennek megfelelően az állapotgép frissítéséért felel. Ez a taszk is egy eseményre várakozik, mégpedig a kellő mérési szám (alapesetben 16) elérésére.



5-3. ábra Gesture taszk folyamatábrája

5.1.3.3 Measurement és Communication taszk

Ez a taszk felel a hőmérséklet mérésért és továbbításáért hálózaton keresztül. Itt történik a fűtésvezérlés is, egyszerű if-else utasítással.



5-4. ábra A Measurement és Communication taszk folyamatábrája

5.1.4 Interrupt szubrutinok

A megszakítások közül a timer interruptot és a GPIO (external) interruptokat használtam. Előbbit az optikai kézmozdulat érzékelést megvalósító szoftvermodul használja, utóbbiakat pedig az Si1143 (mérés elkészülte) és a CY8MBR2044 (valaki megérintette a kapacitív touch pad-et) szenzorok tudnak megszakítást generálni.

5.2 Optikai kézmozdulat érzékelést megvalósító szoftvermodul

Az Si1143 optikai kézmodulat-érzékelésért felelős chip 5 különböző állapotban képes működni:

- Kikapcsolt állapot
- Inicializálási állapot: tápfeszültség rákapcsolása után rövid ideig tart míg inicializálja magát a chip
- Standby mód
- Kikényszerített mérés: egy mérés paranccsal elvégez egyszeri mérést, a mérés eredményét a megfelelő regiszterekbe tölti
- Automata üzemmód: előre beállított időközönként (100us- ~10s) méréseket végez, a mérések elkészülte után ezeket az IRQ lábának a nullába állításával jelzi, hogy van kiolvasandó adat. Amennyiben mi az adatot nem olvassuk ki, a következő méréssel felülírja. A mérések között az eszköz sleep módba megy át.

A programom a fent említett módok közül természetesen az automata üzemmódra helyeztem ki.

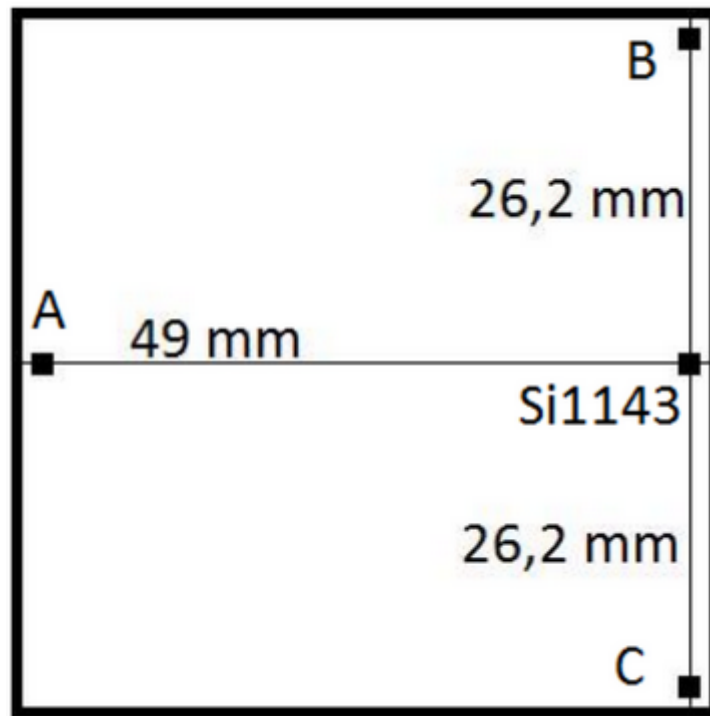
Az eszköz többféle mérésre programozható fel:

- környezeti látható fény mérése
- környezeti infravörös fény mérése
- 3 db proximity sense („közelség érzékelése”): ezzel érzékeli, hogy hol van az előtte lévő tárgy, az infravörös LED-ek visszaverődései alapján
- belső hőmérséklet mérése

Mindegyik proximity sense (PS) mérésnél beállítható mely LED-eket használja. Én úgy konfiguráltam be a chipet, hogy mindegyik PS mérésnél az egyik-egyik LED-et hajtja meg és így végezzen mérést. A méréseknél beállítható még az integrálási idő, hogy mennyi ideig mérjen.

5.2.1 A mérés elve

A kézmozdulatok érzékeléséhez és feldolgozásához célszerű a kézmozdulatot. 3 LED segítségével 2 dimenzióban tudjuk érzékelni a mozgásokat. Ez az eszköz nyomtatott áramköri lapjával párhuzamos sík. Legyen a függőleges irány az y irány, a vízszintes az x.



5-5. ábra A LED-ek (A-B-C) és a szenzor elhelyezkedése a panelen

Az y irányt könnyen megkaphatjuk. Mivel az 'A' LED az $y=0$ helyen van ezért ez kiesik a méréseink során. A különböző méréseket PS-sel jelölöm. Így az y koordináta az alábbi módon számítható ki: $y = PSB - PSC$.

Hasonló megfontolásokkal (közelítéssel) kiszámítható az x koordináta is: $x = PSA - [(PSB + PSC)/2]$.

Az eszköz előtti teret a megadott koordináták szerint.

Bal - Fent	Közép – Fent	Jobb – Fent
Bal - Közép	Közép - Közép	Jobb – Közép
Bal - Lent	Közép - Lent	Jobb - Lent

5-6. ábra A kijelző előtti tér 2D felosztása

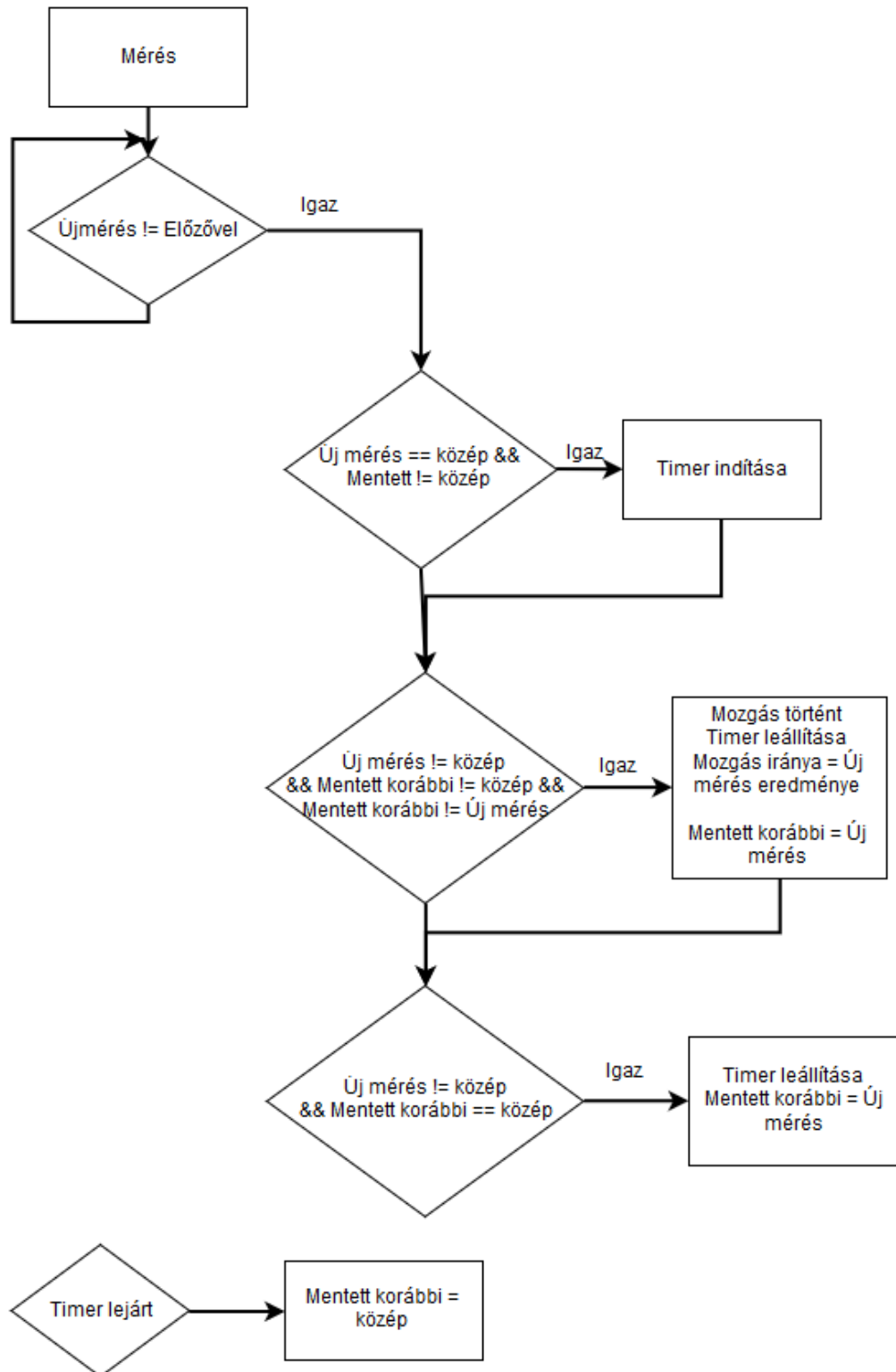
Bal - Fent	Közép – Fent	Jobb – Fent
Bal - Közép	Közép - Közép	Jobb – Közép
Bal - Lent	Közép - Lent	Jobb - Lent

5-7. ábra Egy jobbra irányuló „swipe” mozdulat

A mozgás irányának a meghatározásához 3 pozíciót tárolok el:

- új mérés által adott pozíciót
- előző mérés által adott pozíciót
- egy elmentett korábbi pozíciót, amit meghatározott eseményekkor felülírok.

Ezekből az alábbi algoritmussal határozom meg a mozgás irányát:



5-8. ábra Mozgás irányát meghatározó algoritmus folyamatábrája

Ezt az eljárást külön-külön futtatom az X és Y tengelyen végzett mozgás meghatározására. Az érzékelés úgy kívánja meg, hogy az átlós mozgás esetén ne

történjen változás, tehát például kizárja egymást az, hogy egy balra történt swipe mozdulat történt egyszerre egy felfele irányuló swipe mozdulattal.

5.2.2 A szoftvermodult megvalósító fontosabb függvények

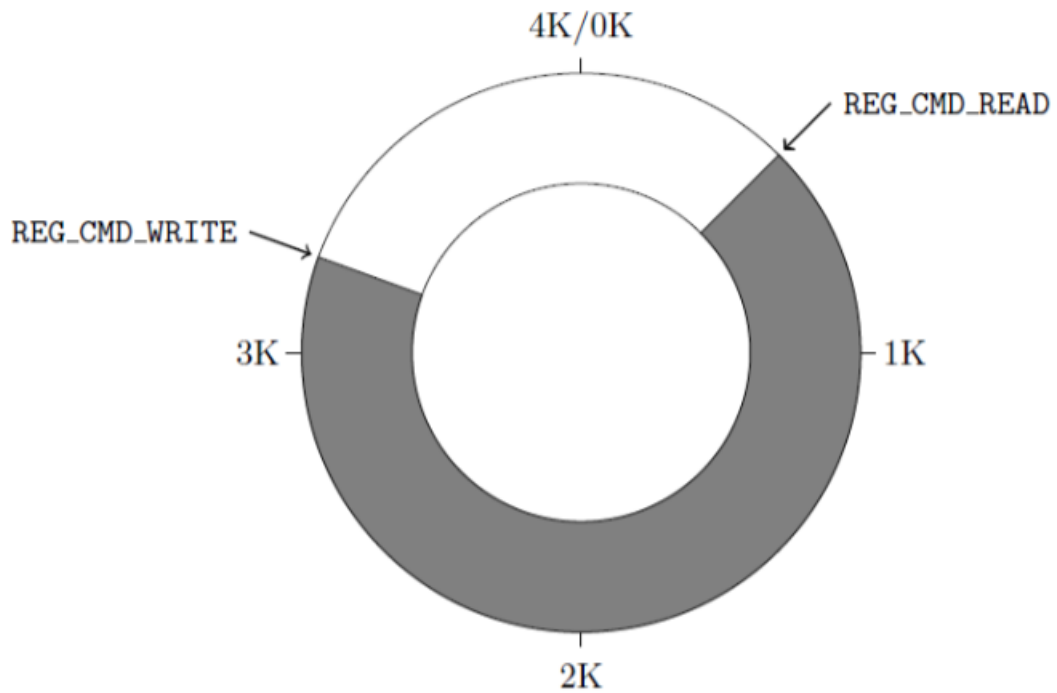
A programomban megírt lényegesebb függvények a következők:

- `char read_reg(char regaddress)` - regiszter olvasás
- `void write_reg(char regaddress, char data)` – regiszter írás
- `extern void sensors_init(void)` – szenzor inicializálása
- `extern position_t evaluate(void)` – mért értékekből pozíció kiszámítása
- `extern movement_t evaluate_mov_x (position_t xnewposition)` – pozícióból 'x' irányú mozgás kiszámítása
- `extern movement_t evaluate_mov_y (position_t xnewposition);` - pozícióból 'y' irányú mozgás kiszámítása

A program elején egy konstansban meg lehet adni hány mérést átlagoljon, méréseim alapján célszerű ezt a számot 20-50 közötti értékre beállítani. A méréseket 10ms gyorsasággal végeztem. Egyes alkalmazásoknál a későbbiekben indokolt lehet 100 us-ig lemenni.

5.3 Kijelzőt vezérlő szoftvermodul

A kijelző vezérléséhez használt FT800-as chipet SPI buszon keresztül lehet vezérelni. A kijelző 256 kilobájtos grafikus RAM-mal és a co-processor egy 4 kilobájtos FIFO jellegű RAM-mal, ahova parancsokat lehet beírni. Az ide beírt parancsokat a chip cirkulárisan hajtja végre, a grafikus RAM-ot én bitmap-ek tárolására használtam.



5-9. ábra A co-processor RAM felépítése

5.3.1 SPI

A kijelző vezérléséhez használt FT800-as chipet SPI buszon keresztül lehet vezérelni. Az általam konfigurált SPI beállításai:

- Órajel 25 MHz (az FT800 inicializálásakor csak 6,25 MHz, a mikrokontrollerrel elérhető legnagyobb SPI órajel 50 MHz, de az FT800 csak 30 MHz-et tud maximálisan, a 50 MHz-es órajelet pedig kettes osztókkal lehet leosztani).
- Szoftveres chip select
- Master mód
- Duplex mód
- First bit MSB

5.3.2 FT800 inicializálása

A chip inicializálását a hivatalos programming guide alapján végeztem. Mivel a programot először WQVGA (480x272) felbontásra írtam meg (a fejlesztői kit miatt) ezért a program konfigurálható WQVGA és QVGA (320x240) felbontásra is.

5.3.3 Képek megjelenítése a kijelző

Fontos volt a kijelzés szempontjából, hogy képeket is meg tudjon jeleníteni a kijelző. A képeket tömörített formátumban küldöm ki a kijelzőnek. Ennek a tömörített formátumnak a neve zlib, amelyet az FTDI által kínált ingyenes konvertáló programmal elő lehet állítani bármilyen jpg vagy png kiterjesztésű fájlból. Ezeket a tömörített állományokat a mikrokontroller flash-jében tárolom, innen küldöm ki 5-öt FT800 felé, amely bitmap-ként letárolja a grafikus RAM-jában és onnan kirajzolja.

A bitmap legnépszerűbb pixeles képformátum. A tömörítetlen bitmap fájlokban az egyes információk pixelenként, azaz képpontonként vannak tárolva. Gyakorlatilag minden egyes képpontról csak a szín van eltárolva, mivel nincs is más lényeges adat. Ez a szín lehet 1, 4, 8, 16, 24 vagy 32 bites. A tömörítetlen formátum hátránya, hogy a kép mérete nagy a tömörített formátumokhoz képest.

A tömörített fájlok mérete (320x240 felbontású kép esetén) körülbelül 22 kilobájt. A kijelző többfajta bitmap formátumot támogat, ezek közül én az RGB332 formátumot választottam, mivel látható, hogy már ez is elég jó képminőséget biztosít a feladathoz. Ebben az esetben 1 bájt 1 pixelnek felel meg.

Bit	7	6	5	4	3	2	1	0
	blue1	blue0	green2	green1	green0	red2	red1	red0



5-10. ábra A kijelző által támogatott bitmap formátumok

A képek kiküldéséhez az FT800 felé CMD_INFLATE parancsot használtam, aminek a felépítése a következő:

+0	CMD_INFLATE(0xffffffff22)
+4	Cél pointer, jellemzően a grafikus RAM adott területe
+8	Bájt 0.
+9	Bájt 1.
..	..
+n	Bájt n.

Egyszerre maximum 8 kilobájtnyi adatot küldhetünk át így, tehát ha egy zlib fájl nagyobb, azt csomagokra kell osztani.

A kép megjelenítéséhez az alábbi parancsok szükségesek:

- BEGIN(BITMAP) – bitmap kirajzolásának a kezdete
- BITMAP_SOURCE – a bitmap a grafikus RAM mely részében található
- BITMAP_LAYOUT – bitmap típusának a megadása
- BITMAP_SIZE – bitmap méretének beállításai
- VERTEX2F – megjelenítési pozíció beállítása
- DISPLAY – megjelenítési parancs

5.3.4 További grafikai objektumok megjelenítése a kijelző

Az FT800 lehetőséget ad további grafikai objektumok kirajzolására, anélkül, hogy azokat képként kellene továbbítani a chip felé. Ilyen grafikai objektumok például:

- Feliratok (gombok)
- Szövegek
- Grafikus órák
- Görgetősávok stb.



5-11. ábra Példa feliratok kirajzolására

Ezek közül az objektumok közül a feliratok kirajzolását használtam a leginkább.

A parancs felépítése a következő:

+0	CMD_BUTTON(0xffffffff0d)
+4	X pozíció
+6	Y pozíció
+8	W szélesség
+10	H magasság
+12	Font - betűtípus
+14	Options – felirat megjelenési típusa
+16	String 1. bájt
+17	String 2. bájt
..	..
+n	String lezáró '\0'

A gomb megjelenítéséhez az alábbi parancsokat használtam:

- CMD_DLSTART – egy rajzolósi lista kezdete
- CMD_FG_COLOR – betűszín beállítása - opcionális
- CMD_GRAD_COLOR – háttérszín beállítása -opcionális

- CMD_BUTTON – felirat rajzolása
- DL_END –rajzolási lista vége
- DL_DISPLAY – megjelenítés

5.3.5 A szoftvermodult megvalósító fontosabb függvények

- void FT800_Init(void); - FT800 inicializálása
- void Menu(void); - menü kirajzolása a képernyőre
- void DrawImage(void); - kép kirajzolása a képernyőre.
- void DrawButton(unsigned short x, unsigned short y, unsigned short wide, unsigned short height, unsigned short font, unsigned short options, char* s); - felirat kirajzolása a képernyőre a megadott pozícióba, megadott mérettel és megadott string-gel

5.4 Az internetes kommunikációt megvalósító szoftvermodul

5.4.1 TCP/IP

Internetes kommunikációra a rendelkezésre álló lehetőségek közül a TCP/IP kapcsolatot választottam. A protokoll kliens – szerver alapú kommunikációt tesz lehetővé.

A TCP/IP felépítése a rétegződési elven alapul, minden egyes réteg egy jól definiált feladatot végez el, és a rétegek egymás között szolgálatelérési pontokon keresztül kommunikálnak. Minden réteg csak a vele szomszédos réteggel képes kommunikálni, mivel ezek egymásra épülne.

Az általam tereztett eszköz kliensként funkcionál, számítógépen pedig egy szerver programot futtatok. A kliensnek szüksége van a szerver IP címére és a Port címre, amelyen a szerver program. Port-nak 50007-et választottam – mivel ezt nem használja más program, a kollégiumi IP címem pedig fix.

A szerver – kliens kapcsolat működése értelmében a kliens adatot küld a szervernek amire a szerver válaszol. Egy fix adatkeretbe foglaltam a küldendő és fogadott adatokat az alábbiak szerint (T = hőmérséklet):

1. byte	2. byte	3. byte	4. byte	5. byte	6. byte	7. byte
0xAA nem használt	mért T felső byte	mért T alsó byte	0x00 nem használt	beállított T	beállított dT (hiszterézis)	0x00 nem használt

5-12. ábra Szerver felé küldendő adatkeret a kientől

1. byte	2. byte	3. byte	4. byte	5. byte	6. byte	7. byte
0xBB nem használt	külső T ASCII első számjegy	külső T ASCII második számjegy	0x00 nem használt	beállított T	beállított dT (hiszterézis)	0x00 nem használt

5-13. ábra Kliens felé küldendő adatkeret a szervertől

5.4.2 Wifi

A chip Wi-Fi kapcsolaton keresztül kapcsolódik a hálózatra. Erre a feladatra a mobilinternettel rendelkező telefonomat használtam Access Point-nak. A kapcsolathoz a titkosításához a ma leggyakrabban használt WPA2-t használtam

Mobil Wi-Fi AP beállítása

Hálózat neve (SSID)

☐ Saját eszköz elrejtése

Biztonság

Jelszó

Legalább 8 karakterből álló jelszót írjon be

☐ Jelszó megjelenítése
☐ Speciális opciók megjelenítése

Mégse **Mentés**

5-14. ábra Access Point beállítása

5.4.3 A szoftvermodult megvalósító fontosabb függvények

- `void tcp_client(char* Frame_to_Send, char* Frame_to_Receive )` – keret küldése és fogadása TCP kapcsolaton, kliensként
- `wiced_network_up_default( &interface, &device_init_ip_settings );` - hálózati interfész inicializálása
- `wiced_tcp_create_socket( &tcp_client_socket, interface )` – socket létrehozása

5.5 Fejlesztést támogató szoftverek

A szoftver fejezet végén kitérnék még a fejlesztőkörnyezetre és a fejlesztést támogató szoftverekre. Mindenképpen jól jött egy verziókövető alkalmazás és jól bevált fejlesztőkörnyezet

5.1.1 SourceTree / BitBucket

Bitbucket egy web-alapú tárhely szolgáltatás olyan projektekre, ahol többen tudnak együtt dolgozni. A feladathoz verziókövetésre / issue-k írására használtam, így a külső konzulensem figyelemmel tudta kísérni a projectemet és a felmerülő problémákat.

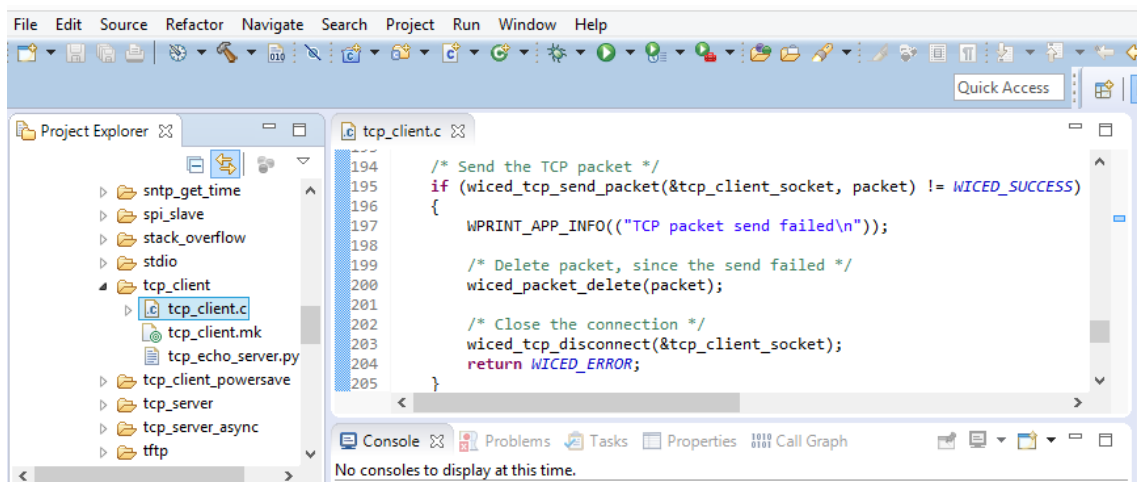
A SourceTree egy kliens volt ehhez a web-alapú szolgáltatáshoz, amivel a verziókövetéshez szükséges műveleteket Commit / CheckOut stb. használtam.

5.5.2 WICED-SDK fejlesztőkörnyezet

Az EMW3165 gyártója egy Eclipse alapú fejlesztőkörnyezetet kínál az eszköz programozására, amelyben számos előre megírt könyvtárat találunk, így nem kell foglalkoznunk az alacsonyabb szinteken lévő driver-ek megírásával, azokat könnyen felhasználhatjuk a programunkban. Ilyen előre megírt könyvtárak például:

- TCP/IP kliens illetve szerver
- HTTP kliens
- UDP transmitter / receiver

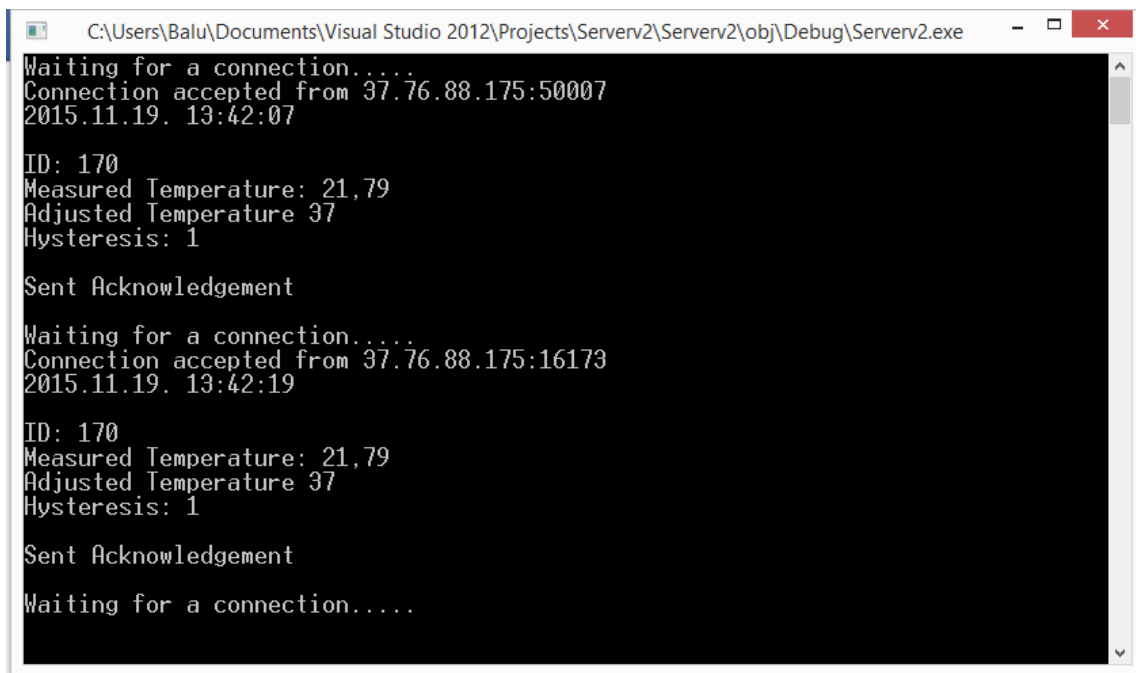
Az Eclipse-hez hasonlóan megtalálható benne a debug funkció, illetve program letöltése az eszközre.



5-15. ábra WICED-SDK fejlesztőkörnyezet

6. Szerver oldali program

A szerveroldali programot C# nyelven írtam Visual Studio 2012-ben. Célja elsősorban a kommunikáció működésének demonstrálása, illetve alapvető adatok kiírása. Egy egyszerű terminál ablakban fut, végtelen ciklusban. Fogadja a 5.1.3.3 pontban leírt kereteket és ennek megfelelően kiírja a paramétereket és visszaküldi a programban megadott adatokat a küldési keretnek megfelelően.



```
C:\Users\Balu\Documents\Visual Studio 2012\Projects\Serverv2\Serverv2\obj\Debug\Serverv2.exe
Waiting for a connection.....
Connection accepted from 37.76.88.175:50007
2015.11.19. 13:42:07

ID: 170
Measured Temperature: 21,79
Adjusted Temperature 37
Hysteresis: 1

Sent Acknowledgement

Waiting for a connection.....
Connection accepted from 37.76.88.175:16173
2015.11.19. 13:42:19

ID: 170
Measured Temperature: 21,79
Adjusted Temperature 37
Hysteresis: 1

Sent Acknowledgement

Waiting for a connection.....
```

6-1. ábra A szerver oldali program konzolos felülete



6-2. ábra A szerver oldali program folyamatábrája

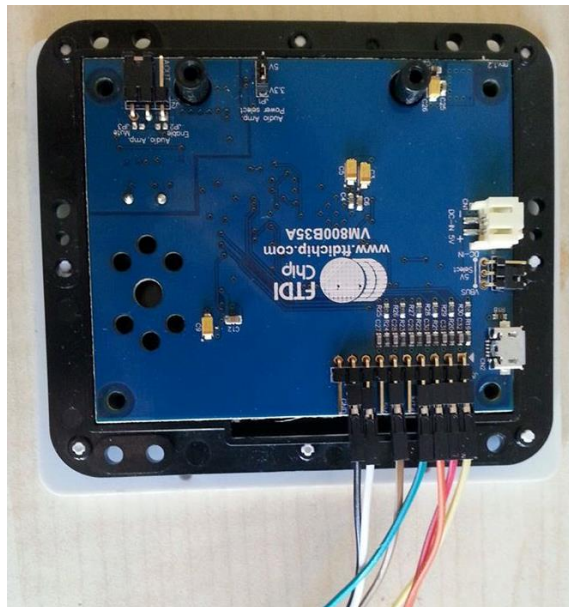
7. Elkészült munka értékelése és továbbfejlesztési lehetőségek

7.1 Eddig elkészült munka

Hardveres fejlesztés terén eljutottam odáig, hogy a NYHL terveket elküldjük gyártásba. Ezen anyagi okokból nem Magyarországon készülnek (ittthon az eszköz NYHL-ének legyártása kb. 40 ezer Forintot tenne ki, mindez Kínában 15 ezer, bár a postával számolni kell, de 5 db esetén megéri).

A fejlesztett programot az eszköz deszkamodelljén fejlesztettem és teszteltem.

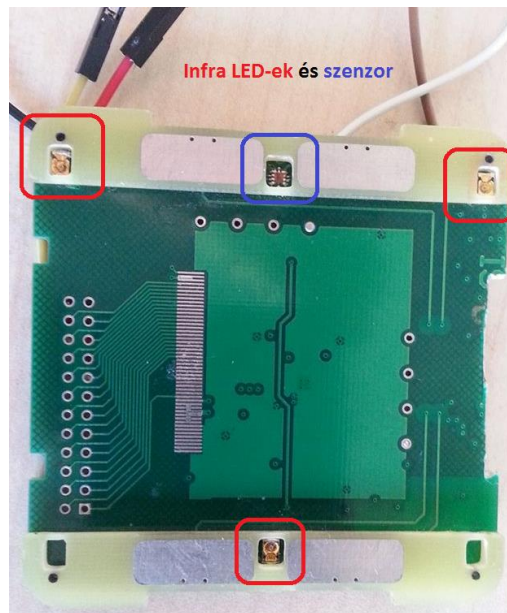
A grafikus processzorhoz kapható egy fejlesztői kit kijelzővel, azon fejlesztettem a programot.



7-1. ábra FT800 Fejlesztői kit kijelzővel

Az intelligens termosztáthoz készült egy korábbi verzió is, amin az Si1143 szenzoron, a hőmérsékletmérő- szenzoron és a kapcsoló relén kívül minden másra volt tervezve, de csak az előbb említett alkatrészek lettek beszerelve a fejlesztés céljából. A LED-ek elhelyezkedése szempontjából fontos volt, hogy a korábbi verzióban a LED-ek pozíciói egyezzenek az újabb hardver tervben szereplővel, ez teljesült.

A deszkamodellt az EMW3165-tel építettem össze



7-2. ábra Infra LED-ek és az érzékelő szenzor



7-3. ábra EMW3165

7.1.1 A szoftver tesztelése

A fejlesztői-kártyákon fejlesztettem és ezeken teszteltem. Várhatóan a jövőben elkészülő hardveren is megfelelően fog futni.

7.2 Továbbfejlesztési lehetőségek

A NYHL legyártása után meg szeretnénk építeni a termosztát prototípusát. A fejlesztett szoftvert az egybeépített eszközön ki kell próbálni. Tesztelni kell, hogy megfelelően működik-e együtt a fűtési rendszerrel. Sorozatgyártás előtt az eszköz megbízhatósága is a vizsgálat tárgyát képezi.

Egy komoly szerver oldali program fejlesztése nem volt része a diplomatervemnek, de mindenképpen a következő lépések egyike lesz. Ez már készülóban van, de nem én fejlesztem. A megfelelő időjáráshoz tartozó képeket is a szerverről szeretnénk leküldeni az eszköznek.

Fontos kérdés a hálózati biztonság, amit mind az eszközön, a hálózaton és a szerveren garantálni kell. Bár nem olyan kritikus a rendszer, mint például egy autóipari fejlesztés, mégsem engedhető meg, hogy mások meghack-eljék eszközünket, manipulálják a fűtést vagy a hűtést.

8. Köszönetnyilvánítás

A diplomatervemhez nyújtott mind szakmai, mind adminisztratív segítséget köszönöm tanszéki konzulensemnek, Benesóczky Zoltánnak.

A projectért köszönettel tartozom külsős konzulensemnek Martonicz Dánielnek, mind a szakmai mind az anyagi támogatásért.

Továbbá megköszönöm családomnak, hogy egyetemi tanulmányaim alatt mindenben támogattak.

9. Ábrajegyzék

2-1. ÁBRA AZ ETHERNETTEL MEGVALÓSÍTOTT HÁLÓZAT LEHETSÉGES TOPOLOGIÁJA	13
2-2. ÁBRAWIFI HÁLÓZAT LEHETSÉGES TOPOLOGIÁJA	14
2-3. ÁBRA SZAKASZCSILLAPÍTÁS A FREKVENCIA FÜGGVÉNYÉBEN KÜLÖNBÖZŐ VEZETÉKHOSSZAK ESETÉBEN	16
2-4. ÁBRA PLC-VEL MEGVALÓSÍTOTT HÁLÓZAT EGY LEHETSÉGES TOPOLOGIÁJA	17
2-5. ÁBRA HONEYWELL RTH959OWF	19
2-6. ÁBRA NEST-TERMOZTÁT ÉS AZ OKOSTELEFON ALKALMAZÁS	21
2-7. ÁBRA AZ ESZKÖZ ALKATRÉSZEINEK KÖLTSÉGEI	23
3-1. ÁBRA SWIPE MOZDULAT BALRA.....	24
3-2. ÁBRA SZABÁLYOZÁSI RENDSZER TÖBBÁLLÁSÚ SZABÁLYOZÓVAL	25
3-3. ÁBRA TÖBBÁLLÁSÚ HISZTERÉZISES SZABÁLYZÓK STATIKUS ÁTVITELE	25
4-1. ÁBRA A HARDVER BLOKKVÁZLATA	27
4-2. ÁBRA EMW3166	28
4-3. ÁBRA EMW3165 BLOKKVÁZLATA	29
4-4. ÁBRA SHT21 HŐMÉRSÉKLET ÉS PÁRATARTALOM SENZOR.....	30
4-5. ÁBRA A SENZOR PONTOSSÁGA %-BAN, A HŐMÉRSÉKLET ÉS PÁRATARTALOM FÜGGVÉNYÉBEN....	31
4-6. ÁBRA SI1143 KAPCSOLÁSI RAJZA	32
4-7. ÁBRA AZ INFRAVÖRÖS LED-EK KARAKTERISZTIKÁI	32
4-8. ÁBRA SI1143 BLOKKVÁZLATA	33
4-9. ÁBRA NAGYOBB, INFRAVÖRÖS FÉNYRE ÉRZÉKENY DIÓDA VERTIKÁLIS KARAKTERISZTIKÁJA	34
4-10. ÁBRA CYPRESS CY8CMBR2044.....	34
4-11. ÁBRA KAPACITÍV TOUCH PAD-EK.....	35
4-12. ÁBRA A MEGTERVEZETT KAPACITÍV TOUCHPAD-EK.....	35
4-13. ÁBRA LCD DISPLAY	36
4-14. ÁBRA LCD LED-JEINEK PWM VEZÉRLÉSE	36
4-15. ÁBRA AZ FT800 BLOKKVÁZLATA	37
4-16. ÁBRA AZ ESZKÖZ TÁPELLÁTÁSA.....	38
4-17. ÁBRA A FŰTÉST VEZÉRLŐ RELÉK KAPCSOLÁSA	39
4-18. ÁBRA AZ ESZKÖZ FALRA SZERELHETŐ DOBOZA.....	40
4-19. ÁBRA A MEGTERVEZETT ESZKÖZ 3D MODELLJE ALTUM DESIGNER PROGRAMBAN	40
5-1. ÁBRA A DISPLAY TASZK FOLYAMATÁBRÁJA.....	43
5-2. ÁBRA MENÜK ÉS ÁLLAPOTÁTMENETEK.....	43
5-3. ÁBRA GESTURE TASZK FOLYAMATÁBRÁJA	44
5-4. ÁBRA A MEASUREMENT ÉS COMMUNICATION TASZK FOLYAMATÁBRÁJA	45

5-5. ÁBRA A LED-EK (A-B-C) ÉS A SZENZOR ELHELYEZKEDÉSE A PANELEN	47
5-6. ÁBRA A KIJELZŐ ELŐTTI TÉR 2D FELOSZTÁSA	48
5-7. ÁBRA EGY JOBBRA IRÁNYULÓ „SWIPE” MOZDULAT	48
5-8. ÁBRA MOZGÁS IRÁNYÁT MEGHATÁROZÓ ALGORITMUS FOLYAMATÁBRÁJA	49
5-9. ÁBRA A CO-PROCESSZOR RAM FELÉPÍTÉSE	51
5-10. ÁBRA A KIJELZŐ ÁLTAL TÁMOGATOTT BITMAP FORMÁTUMOK	52
5-11. ÁBRA PÉLDA FELIRATOK KIRAJZOLÁSÁRA.....	54
5-12. ÁBRA SZERVER FELÉ KÜLDENDŐ ADATKERET A KLIENSTŐL	56
5-13. ÁBRA KLIENS FELÉ KÜLDENDŐ ADATKERET A SZERVERTŐL	56
5-14. ÁBRA ACCESS POINT BEÁLLTÁSA	56
5-15. ÁBRA WICED-SDK FEJLESZTŐKÖRNYEZET	58
6-1. ÁBRA A SZERVER OLDALI PROGRAM KONZOLOS FELÜLETE	59
6-2. ÁBRA A SZERVER OLDALI PROGRAM FOLYAMATÁBRÁJA	60
7-1. ÁBRA FT800 FEJLESZTŐI KIT KIJELZŐVEL.....	61
7-2. ÁBRA INFRA LED-EK ÉS AZ ÉRZÉKELŐ SZENZOR.....	62
7-3. ÁBRA EMW3165	62

10. Irodalomjegyzék

¹ Andrew S. Tanenbaum: Számítógépes hálózatok, ISBN 963-545-213-6, Panem Könyvkiadó Kft., 1999

² Next Generation Narrowband (Under 500 kHz) Power Line Communications (PLC) Standards

<http://www.g3-plc.com/sites/default/files/press-releases/Next%20generation%20NB-PLC%20standards%20%28China%20Comms%202015%29.pdf>

³ Andrea M. Tonello - Advances in Power Line Communications and Application to the Smart Grid Tutorial at EUSIPCO 2012

http://www.eurasip.org/Proceedings/Eusipco/Eusipco2012/Conference/Tutorials/Tutorial%208/TUTORIAL_EUSIPCO_TONELLO.pdf

⁴ Molnár Berta: Honeywell: okos és felhasználóbarát termosztát

<http://pcworld.hu/haztartas/honeywell-okos-es-felhasznalobarat-termosztat.html>

⁵ Bankó Zsolt - Nest, avagy mi van a fészekben?

<http://ikomfort.ngbsh.hu/2014/07/nest-avagy-mi-van-a-feszekben/>

⁶ <http://www.ms.sapientia.ro/~martonl/Docs/T3%20-%20Tobballasu%20szabalyozok.pdf>

⁷ EMW3165 Datasheet:

<https://hackadaycom.files.wordpress.com/2015/07/emw3165.pdf>

⁸ SHT21 Datasheet

http://www.mouser.com/ds/2/682/Sensirion_Humidity_SHT21_Datasheet_V4-469715.pdf

⁹ Si1143 Datasheet

<https://www.silabs.com/Support%20Documents/TechnicalDocs/Si114x.pdf>

¹⁰ CY8CMBR2044 Datasheet

<http://www.cypress.com/file/122496/download.pdf>

¹¹ KD026FM Datasheet

<http://en.lcdlcd.com/res/starteklcden/pdres/201410/20141010173455601.pdf>

¹² FT800 Datasheet

http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT800.pdf

¹³ UA2/UB2 Datasheet

http://www.mouser.com/ds/2/212/KEM_R7004_UA2_UB2-540999.pdf

¹⁴ <http://www.freertos.org/>