

# FELADATKIÍRÁS

Tervezzen intelligens órát mikrokontroller alkalmazásával! A rendszernek az alábbi követelményeknek kell megfelelnie:

- A berendezés egy központi és egy (vagy több) távoli egységből álljon, amelyek között a kommunikáció vezeték nélkül legyen megoldva.
- A központi egység rendelkezzen LCD-kijelzővel, valamint egy DCF-vevő segítségével automatikusan frissítse az óra pontos idejét.
- A DCF vevőt célszerű távoli egységeként megvalósítani.
- A távoli egység legyen alkalmas különféle adatok (például dátum, pontos idő, hőmérséklet) megjelenítésére, valamint ébresztésre.
- A berendezés legyen alkalmas ismételt ébresztések egyszerű megadására, és emlékeztető üzenetek megjelenítésére a központi egység kijelzőjén.
- Számítógépről is legyen vezérelhető.



**Budapesti Műszaki és Gazdaságtudományi Egyetem**  
Villamosmérnöki és Informatikai Kar  
AUT Tanszék

Varga Dominik

**INTELLIGENS ÓRA TERVEZÉSE  
MIKROKONTROLLERREL**

PCB tervezés

KONZULENS

**Dr. Ivánctsy Szabolcs**

BUDAPEST, 2022

# Tartalomjegyzék

<b>1 Bevezetés .....</b>	<b>4</b>
<b>2 Tervezési döntések – Blokkvázlat.....</b>	<b>5</b>
2.1 HEAD .....	6
2.2 NODE .....	7
<b>3 Alkatrészkeresés, nehézségek és hasznos ötletek .....</b>	<b>10</b>
<b>4 PCB tervezés.....</b>	<b>11</b>
4.1 Tervezési előkészületek .....	11
4.2 Schematic – kapcsolási rajz .....	11
4.2.1 Mikrokontroller.....	11
4.2.2 Tápellátás és annak védelme, feszültségmérés .....	13
4.3 Layout terv .....	14
<b>5 DCF77 mérése, megértése .....</b>	<b>17</b>
5.1 DCF77.....	17
<b>6 Összegzés, jövőbeli tervek .....</b>	<b>19</b>
<b>Irodalomjegyzék.....</b>	<b>20</b>
<b>Függelék .....</b>	<b>21</b>

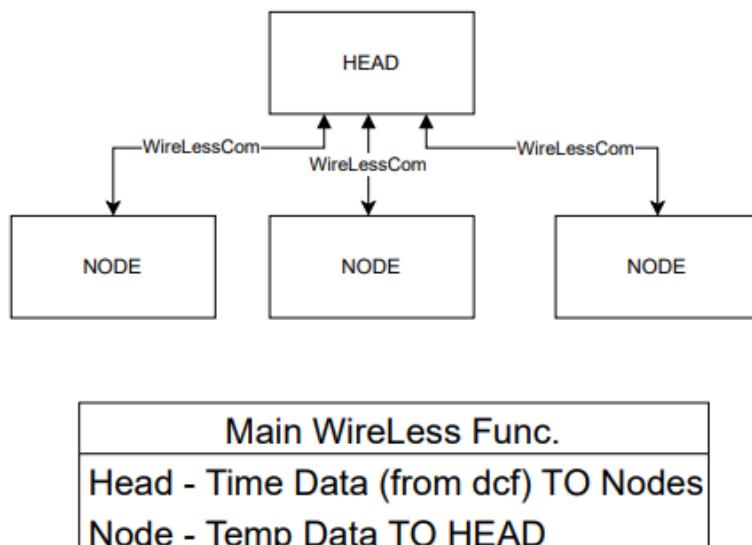
# 1 Bevezetés

Az önálló labor projekt célja két nyomtatott huzalozott lemez (későbbiekben csak nyák) tervezés és elkészítése. A vezérlő nyákunkat a project során HEAD-nek nevezzük. A HEAD-del végez kommunikációt a második nyák, mely a projekt során NODE nevet kapta, ez a nyák a falióra szerepét tölti be. A két nyákat a következő feladatokra kell felkészítenünk:

A HEAD feladatköre, hogy egy DCF-77-es vevő segítségével fogadni tudja a 77,5 kHz-en sugárzott atomóra jelét (a pontos időt). Képesnek kell lennie még időzítések beállítására saját interface-szel, illetve számítógéppel történő kommunikáció során. Ezeket az adatokat továbbítania kell a NODE-ok számára szimultán módon. Illetve a NODE-okból kinyert adatok kijelzését is meg kell valósítania.

A NODE feladatköre egy nagyobb kijelzőn az idő pontos kijelzése, az idő szinkronizációs parancsok, illetve időzítési parancsok fogadása és lekezelése, a beállított időzítésekkel a riasztás elvégzése, illetve hőmérsékleti adatok mérésére és ezek továbbítására a HEAD felé.

A nyákok közti kommunikációt vezeték nélküli módon kell végrehajtani.



1. ábra Az önálló laboratórium projekt elvi felépítése

## 2 Tervezési döntések – Blokkvázlat

A tervezés során voltak olyan döntések, melyek mind a kettő nyákat érintették. Ezekre szeretném először kitérni. Az első ilyen döntés a mikrokontroller kiválasztása, ez alapvetően különbözhetne a két nyák esetében, hiszen más feladatköröket töltenek be. Viszont a pillanatnyi gazdasági helyzet miatt (2022 chip hiány) azzal a mikrokontrollerrel kell dolgoznunk, amihez hozzá jutunk. A kezdeti tervek szerint ez egy STM32G030C8T6-os ARM Cortex-M0+ 32-bit MCU volt, viszont ez a kontroller sajnos csak 8 KB RAM-mal rendelkezik. Így a félév közepé tájékán, amikor elérhetővé vált egy hasonló kontroller nagyobb RAM-mal, ezt a modellt lecseréltem. Végül egy STM32G070RBT6-os kontrollert találtam, ez a modell hasonlóan elődjéhez egy ARM Cortex-M0+ 32-bit MCU, viszont 36 KB RAM-mal rendelkezik. Ez azért fontos tervezési döntés, mert ahogy már említettem a fentebbiekben a HEAD-nek és NODE-nak számos feladatot, „taskot” kell teljesíteni. A nagyobb RAM pedig lehetővé teszi, hogy egy beágyazott operációs rendszert használjunk erre a célra (például FreeRTOS).

A kontrollerek mellé szükséges illesztenünk egy 32.768kHz-es kristályt, amivel a belső kristályt helyettesítjük a pontosabb időmérés végett.

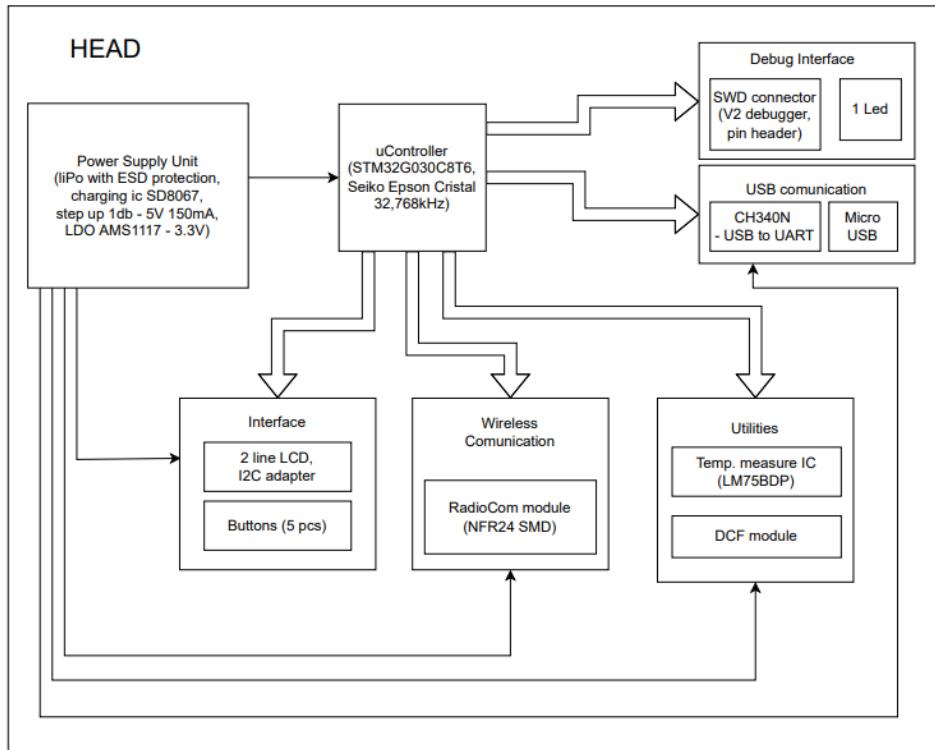
Mind a kettő nyák során az ST, SWD (Serial Wire Debug) csatlakozó típusát alakítottam ki a kontrollerek programozására, debugolására, illetve elhelyeztem egy LED-et ezek mellett.

A második fő kérdéskör pedig a HEAD és a NODE közötti kommunikáció módja. Erre a beágyazott környezeteken belül rengeteg különböző mód létezik. Ebben a projektben én egy rádió kommunikáció alapú megoldást választottam (NRF24, LoRa, CC1101, HC12 stb.). A rádió kommunikációs modulokat először leszűkítettem az NRF24-re és LoRa-ra. Az NRF24 egy 3.3V-ról működő, a kontrollerhez SPI-on csatlakozó kompakt megoldás. Létezik belőle mini változat (SMD szerelhető NRF24L01). Képes akár 700-500 méteren adatokat át küldeni. A LoRa ezzel szemben egy robusztus megoldás sok adat átvitelére fejlesztve, akár 6 km-es távolságba.

Ezek közül pedig az NRF24-et választottam mivel ennek mini változata jobban beleillik a „faliórák” (NODE-ok) esetében a projektbe. Ezen kívül úgy gondolom, hogy a LoRa robusztussága és adatátviteli távolsága nem lett volna kihasználva a projekt során.

## 2.1 HEAD

Minden nyák tervezésének első lépése egy blokkvázlat elkészítése mellyel nyomon követhetjük gondolatainkat és egyszerűbbé tehetjük a tervezés későbbi fázisait. A következő blokkvázlat a HEAD vezérlőnket fogja bemutatni, itt jól látható, hogy vékony nyíllal az áramellátást, vastaggal pedig a kommunikációs csatornákat jelöltetem:



2. ábra HEAD blokkvázlat

A HEAD Interface-hez 5 db nyomógombot és egy 2 soros LCD kijelzőt választottam. A számítógéppel történő kommunikációhoz pedig egy micro usb-t terveztem a nyákba mely képes az akkumulátorunk töltésére is (segéd IC-vel). Úgy gondoltam, ez nagyobb kihívást fog jelentni a nyáktervezés során, mint egy Wifi-s kommunikációs modul.

A HEAD áramellátásának egy Lítium-Polimer akkumulátort választottam, ezek az elemek már beépített védelemmel rendelkeznek. Ez az akkumulátor önmagában még nem elég, hiszen elő kell állítanunk belőle a megfelelő feszültségszinteket, illetve azt is számításba kell vennünk, hogy mennyi mA-re lesz szükség nyák adott részeinek áramellátásához. A HEAD során két fő feszültségszintet különbözetünk meg, az első a 3.3 V, ezen a feszültségen működik a kontrollerünk, az NRF24, és a különböző Utility

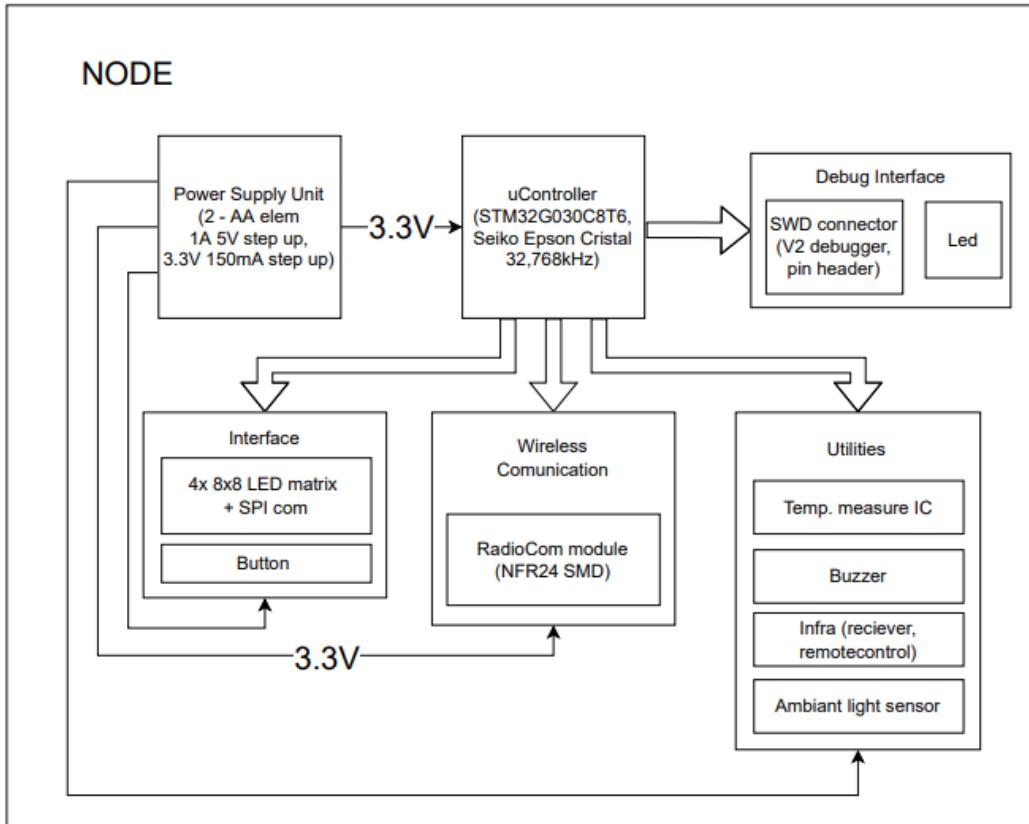
eszközeink (hőmérsékletmérő szenzor), illetve az USB kommunikációt kisorosító UART átalakító is. Ezen komponensek áramfelvételé  $\mu$ A nagyságrendű. A nyákunknak egyetlen részére kell 5V-ot előállítanunk ez pedig nem más, mint az Interface, azon belül is a 2 soros LCD kijelző áramkör. Ennek áramfelvételé mA nagyságrendű.

A 3.3V elő állítására egy LDO-t (Low Drop Out) feszültség szabályozót alkalmazunk (LDS3985 [1]), mivel LiPo akkumulátorunk névleges feszültsége 3.6V (max 4.2V, min 3.2V). Ez az IC úgy működik, hogy a fölösleges teljesítményt elfúti saját magán. Itt figyelembe kellet venni még, hogy speciális LDO-t tervezünk a nyákunkba, amelynek low drop-ja kisebb, mint 0.3V. Az általam választott LDO-nak pedig 2mV a maximum feszültségesése, így megfelel erre a célra.

Az 5V-os feszültség szint elő állításához pedig egy Step Up Convertert használunk (TPS61023DRLR [2]). Itt figyelembe kell venni az adott elemek áram felvételét, az általam választott modell tökéletesen kielégíti a LCD kijelzőket.

## 2.2 NODE

A NODE áttekintő blokkvázlata a következő, ebben az esetben is a vékony nyilak az áramellátást jelölnek, a vastagabbak pedig az adatátvitelt:



**3. ábra NODE blokkvázlat**

Ennek a nyáknak az esetében az Interface-t egy 4 panelból álló 8x8-as LED mátrixszal, illetve 5 darab nyomógombbal terveztem megvalósítani, ez elég nagy felületet biztosít az idő kijelzésére, illetve egyéb jövőbeli plusz információk, például hőmérséklet, időjárás kijelzésére is. A nyákra terveztem egy fényérzékelőt is, ennek a szerepe, hogy esténként kikapcsolja a LED világítást, ezzel energiát spórolva. Illetve a NODE egyik fontos feladata, hogy képes legyen riasztásokat leadni, ezt egy Buzzer (piezo buzzer resistor) segítségével valósítottam meg. Mivel ez a nyák „falióra”-ként működik, fontosnak tartottam egy infra szenzor elhelyezését is a nyákon. Ez lehetővé teszi a távirányítóval történő irányítást.

A NODE áramellátásának két AA-s elemet választottam. A két elem sorosan van összekötve, ezáltal a közös feszültségük 3V-ot ad ki. A nyákunkon két fő feszültségszintet különböztetünk meg, az első a 3.3 V, ezen a feszültségen működik a kontrollerünk, az NRF24, és a különböző Utility eszközeink (hőmérsékletmérő szenzor, light szenzor infra szenzor). Ezen komponensek áram felvétele  $\mu$ A nagyságrendű. A nyákunknak egyetlen

részére kell 5V-ot elő állítanunk ez pedig nem más, mint az Interface azon belül is 4 darab 8x8-as LED mátrix áramkör. Ennek áram felvétele mA nagyságrendű.

A 3.3 Volt illetve 5 Volt előállításához is ugyanazt az IC-t használtam. Ez az előző fejezetben már ismertetett Step Up Converter (TPS61023DRLR [2]). A nyákunkban a legnagyobb fogyasztók a LED mátrixaink, erre kell méreteznünk a maximum fogyasztást 5V-os esetben. Ennek pontos számításához vissza kell számolnunk egy pixel áramfelvételét és ki kell számolnunk, hogy körülbelül mennyi pixelt jelenítünk meg egyszerre. Erre a két értékre tudunk optimalizálni. Ha kiszámítottuk az áramfelvételét, a feszültségszintet és a teljesítményt akkor az IC keresésben nagy segítséget tud nyújtani a Texas Instruments oldala [3]. A számítások alapján az áramfelvétel mA nagyságrendű, így ennek megfelelően kerestem IC-t.

Szerencsére a talált IC jóval túlteljesíti a számolt áramfelvételeinket,  $I_{out}$ -ja tipikusan 1.5A, bemeneti feszültség szintje 0.5V-5.5V-ig, illetve kimeneti feszültsége 2.2V-5.5V-ig terjed [2].

### **3 Alkatrészkeresés, nehézségek és hasznos ötletek**

Az önálló laboratórium projektének alkatrészeit különböző „vendorok”-tól (beszállítóktól) szereztem be. Kiemelném, hogy a híresebb beszállítókon (Mouser, Farnel, Hestore) két fontosabb és az esetek többségében olcsóbb beszállítót is találtam. Az első a Lomex ahol a különböző passzív komponenseinket (ellenállás, kapacitás) olcsón és jó minőségben szerezhetjük be. Illetve amit még kiemelnék az Lcsc, ahol Kínából tudunk különböző olcsó IC-ket rendelni. A komponenseim nagy része innen származik. Egyetlen hátránya, hogy az adatlapok nem minden esetben elérhetők angolul. De ilyenkor könnyen használhatóak a különböző online fordítók az adatlapok fordítására.

A jövőbeli tervezések során érdemes lenne egy olyan alkatrészkeresési stratégiát követni, hogy olyan IC-ket keresünk a létfontosságú helyekre (Step Up Converter, LDO, MCU) melynek footprint-je és tulajdonságai megegyeznek (tehát ugyanazok az IC-k más gyártóktól). Ezáltal növelte az esélyét, hogy több helyről beszerezhessük az IC-ket (legyenek „backup” IC-k). Ez a módszer kicsit meghosszabbítaná a különböző IC-k keresését, de nyugodtan tudnánk a nyáktervezés későbbi állapotaiba lépni (ahol például a passzív komponensek, ellenállások, kapacitások mennyisége és típusa derül ki). Amikor pedig megrendeljük a különböző beszállítóktól ezeket az elemeket, biztosak lehetünk, hogy „backup” IC-ink egyike még raktáron van. Ez különösen fontos a 2022-es chip hiányat tekintve, mivel van, hogy nagyobb cégek nagyon gyorsan felvásárolnak adott típusú IC-ket (ezek újra gyártása pedig sok időt vesz igénybe).

## 4 PCB tervezés

A blokkvázlatok és az elvi tervezés befejezésével a nyáktervezés következő nagyobb szakaszába lépünk ez a PCB terv elkészítéséből áll. Egy PCB terv (nyákterv) négy fő elemből áll: a kapcsolási rajzból (Schematic-ból), a PCB layout-ból ( minden fizikai felépítést jellemző leírást tartalmaz), BOM-ból azaz a Bill Of Materials-ból amely egy pontos lista a komponenseinkről, végül pedig a Gerber fájlok ból, ezek a gyártási fájlok.

### 4.1 Tervezési előkészületek

Én a tervezés során Altium Designer tervező programot használtam. Ez egy piacvezető PCB tervező eCAD program. Ahhoz, hogy el tudunk benne kezdeni tervezni, létre kell hoznunk pár forrás fájlt a használni kívánt IC-kről. Ezek a Schematic rajzok és a footprint rajzok (fizikai rajz paraméterek).

Ezeket létrehozhatjuk saját magunk, vagy használhatjuk a szoftver beépített generátorait (Footprint Wizard), illetve az Altium saját belső könyvtárát (Component List) vagy online könyvtárát (Manufacturer Part List). A program széleskörű támogatottsága miatt rengeteg online oldal található, ahol fellehetjük ezt a két fájlt, ilyen például az Ultra Librarian. Ezen felül létezik egy 3D CAD fájlból történő generálási módszer is.

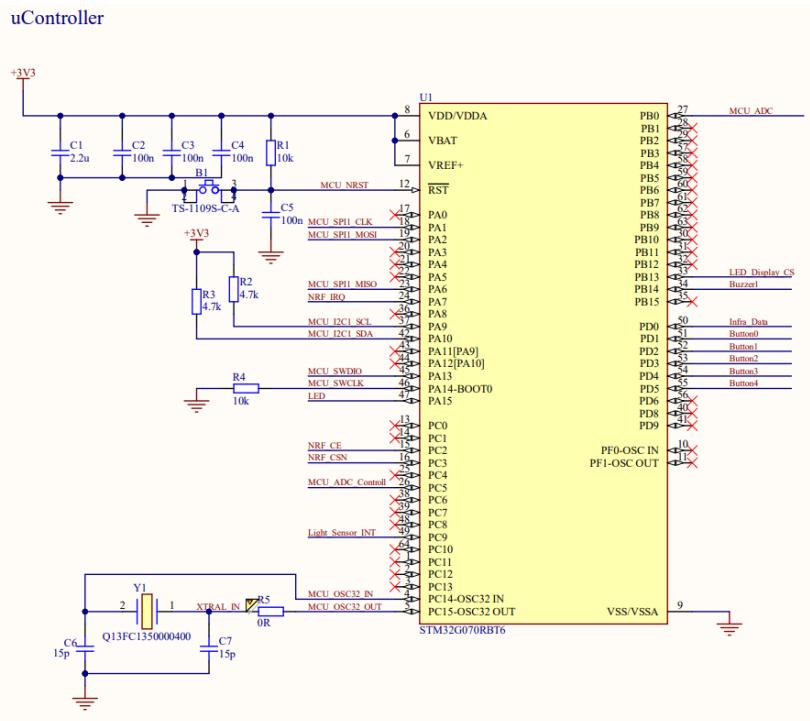
### 4.2 Schematic – kapcsolási rajz

A schematic, a nyákok kapcsolási rajza, mely a különböző IC-k adatlapjai alapján készül el. Ebben a fejezetben a schematic-jainak kapcsolatos érdekességeket szeretném bemutatni. A HEAD és NODE teljes schematic-ját a függelékben belül olvashatjuk.

#### 4.2.1 Mikrokontroller

A mikrokontrollerünk kapcsolási rajza nagyrészt (a lábkiosztás kivételével) megegyezik a két nyák esetén. Ezzel a résszel kapcsolatban két dolgot emelnék ki. Először is a de-coupling és a bulk de-coupling kapacitásokat. Ezek a képünk (4. ábra) bal fölső sarkában láthatóak. A de-coupling kapacitások általában 100nF-osak szoktak lenni és jó ökoliszabály ezekre, hogy annyit kell elhelyeznünk a nyákon, ahány különálló

táplába van a mikrokontrollerünknek. Az én esetemben 3-at raktam, de a későbbiekben rájöttem, hogy elég lett volna egy, mert a 3 láb egymás mellett található. Ezeknek a kapacitásoknak a szerepe a stabil áramellátás biztosítása, ha hirtelen a mikrokontrollernek áramra van szüksége, akkor ezekből a kapacitásokból tud nyerni. Ezeket minél közelebb kell elhelyeznünk a későbbiekben a mikrokontroller táplábaihoz. A bulk de-coupling kapacitás egy kicsit nagyobb (2.2u, 4.7u, 10u) én 2.2uF-osra választottam meg. Ezeknek a kapacitásoknak főként az a feladatkörük, hogy stabillá tegyék az áramforrásunkat. Van olyan eset is ahol, ferral bead-fel szűrik a táp bemeneteket.



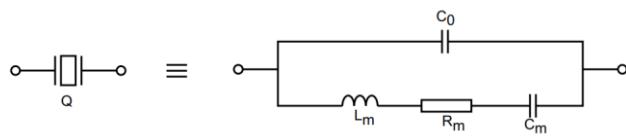
4. ábra Mikrokontroller kapcsolási rajza

A második pedig a 32.7kHz kristály illesztése a mikrokontrollerhez. Ehhez először az oszcillátor kristályunkat helyettesítjük egy RLC rezgő körrel ezt az 5. ábra mutatja. Majd ezek alapján a következő képlet segítségével illesztjük a komponenst a mikrokontrollerhez.

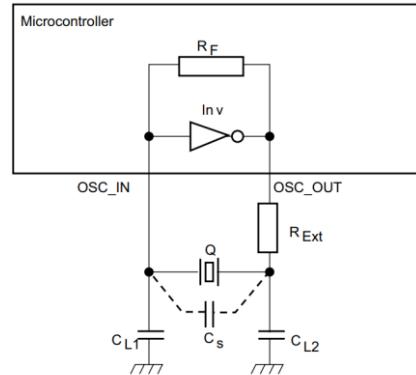
$$C_L = \frac{C_{L1} \times C_{L2}}{C_{L1} + C_{L2}} + C_s$$

A visszacsatolási ellenállásunkat először érdemes 0 Ω-os ellenállásként felvenni, ha így nem kezd el rezonálni a kristályunk, akkor a dokumentáció alapján egy nagyobb

$M\Omega$  nagyságrendű ellenállásra kell ezt cserélnünk. Az ST biztosít nekünk egy Oszcillátor tervezési dokumentációt ahol ezekről minden részletesen olvashatunk [4].



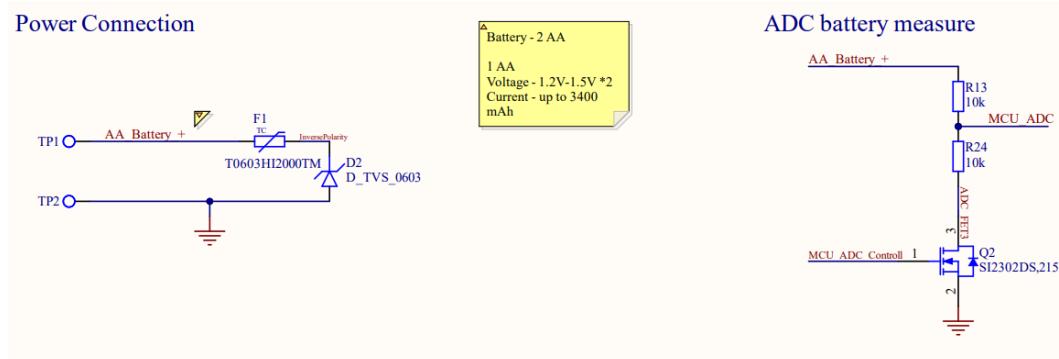
5. ábra Oszcillátor kristály RLC körrrel történő helyettesítése



6. ábra Kristály illesztése mikrokontrollerhez

#### 4.2.2 Tápellátás és annak védelme, feszültségmérés

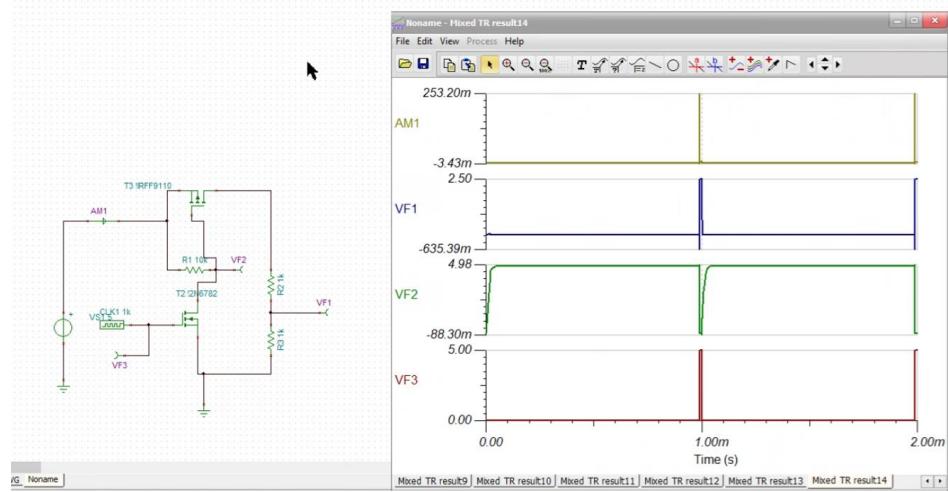
Mind a két nyák esetében elhelyeztem a táp csatlakozási részére egy biztosítékot (fuse-t) a pozitív táp vonalon, illetve egy TVS diódát a két bemenet közé, melynek katódja a pozitív tápellátás felé néz. Ez fordított polaritás ellen védi az áramkörünket. Illetve felkészítettem a nyákokat az akkumulátorunk feszültségének mérésére a következő módon:



7. ábra Fordított polaritás elleni védelem és feszültség mérés

Ez a feszültségmérési stratégia azért működik, mert a mikrokontroller analóg mérési bemenetén a magasabb feszültségek ellen diódák védik a műszert. Ha ezeket a

diódákat nem vesszük figyelembe a mérést a következő áramkörrel hajthatjuk végre (8. ábra).



8. ábra ADC mérés mikrokontrolleren

### 4.3 Layout terv

A layout terv elkészítésének egyik első lépése a nyák fizikai felépítésének meghatározása. A projekt feladat során minden a két nyákot négyrétegű nyáknak terveztem. A felső és alsó rétegek a jelrétegek a két belső pedig egy föld (GND) réteg és egy táp réteg (3.3V). Ezeket a rétegeket a programban kell konfigurálunk a PCB gyártó alapján.

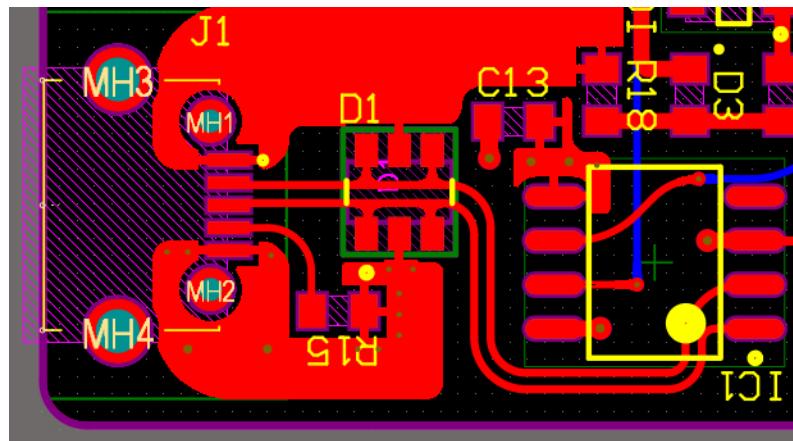
#	Name	Material	Type	Weight	Thickness	Dk	Copper	Top Ref	Bottom Ref	Width (W1)	Trace Ga...	Impe...	Devia...	Delay...
	Top Overlay		Overlay											
	Top Solder	SM-001	Solder Mask		0.0254mm	4								
	Top Surface Fini...	PbSn	Surface Finish		0.02mm									
1	L1 (Signal)	CF-004	Signal	1oz	0.035mm		Above	2 - L2 (GND)	0.27062mm	0.2032mm	90	0%	6.0471...	
	Dielectric 1	PP-017	Prepreg		0.2104mm	4.6								
2	L2 (GND)	CF-004	Signal	1/2oz	0.0152mm		Above	1 - L1 (Signal)	3 - L3 (+3V3)	0.26236mm	0.127mm		0.03%	
	Dielectric 3	Core-039	Core		1.065mm	4.6								
3	L3 (+3V3)	CF-004	Signal	1/2oz	0.0152mm		Below	2 - L2 (GND)	4 - L4 (Signal)	0.26236mm	0.127mm		0.03%	
	Dielectric 5	PP-017	Prepreg		0.2104mm	4.6								
4	L4 (Signal)	CF-004	Signal	1oz	0.035mm		Below	3 - L3 (+3V3)		0.27062mm	0.2032mm	90	0%	6.0471...
	Bottom Surface...	PbSn	Surface Finish		0.02mm									
	Bottom Solder	SM-001	Solder Mask		0.0254mm	4								
	Bottom Overlay		Overlay											

9. ábra Négy layeres felépítés és ellenállás szabályozás

Itt még egy fontos, beállítást is szemléltetek a fentebbi képen. Ez nem más, mint a „differential pair rooting”-hoz szükséges ellenállás számítás a rétegek és két vezeték között. A „differential pair rooting” lehetővé teszi, hogy két vezeték vonal húzása során a vezetékek közti ellenállás állandó legyen.

Ahhoz, hogy ez működőképes legyen a tervező környezetben kiválasztottam melyik rétegeken szeretném majd ezeket a vezetékeket húzni és megadtam a  $90\ \Omega$ -ot mint

ellenállás korlátot, illetve a 8 mil-es (0.2032mm) vezetékek közti távolságot. Az ellenállás beállítása az USB szabványból ered, a 8 mil pedig a nyákgyártó saját bevett értéke. Ezt követően az Altium-ra hagytam a többi számítást a differenciális vezetőpár húzása során.

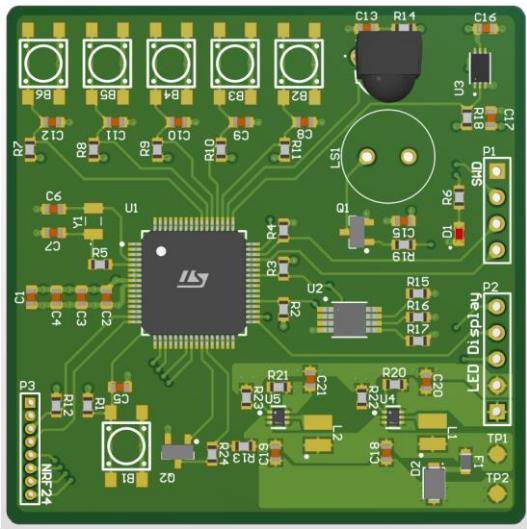


**10. ábra „Differential pair rooting” USB-től az UART átalakítóig**

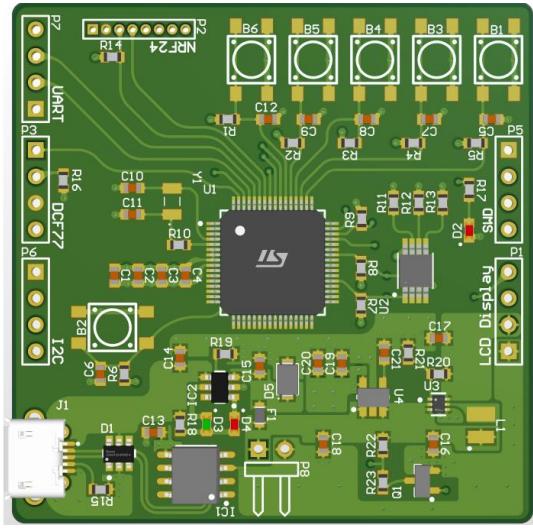
A fent látható ábrán a HEAD nyák bal alsó sarka látható. Itt található a micro USB csatlakozó (baloldal), az ESD elleni védelem (középső IC) és az UART átalakító. Ezek között az USB D+ és D- adatvonali között alkalmaztam a fent említett „differential pair rooting”-ot.

A következő két ábrán pedig a két nyákunk teljes felépítése látható (11. ábra és 12. ábra). Itt megfigyelhetjük, hogy a kialakítás során fontos tervezési döntés volt a nyákok tápellátásáért felelős részének elkülönítése a nyákokon (a NODE esetében bal alul a HEAD esetében pedig a nyák alsó fele). Itt a stabil feszültségszintek érdekében poligon kitöltési módszert alkalmaztam ( minden egyes nagyobb poligon egy különálló feszültségszintet képvisel). Ezeket a kitöltési zónákat egy későbbi fejlesztés során akár el is rejthetjük a belső táp rétegen (annak szeparálásával).

A NODE esetében egy jelentős kialakítási döntés volt, hogy a kezelő felületet (gombok, infra szenzor, light szenzor) a felhasználó számára elérhetővé kellet tenni. Így ezek a komponensek kikerültek a nyák oldalára, úgy, hogy kilógjanak a LED kijelző alól.



11. ábra NODE



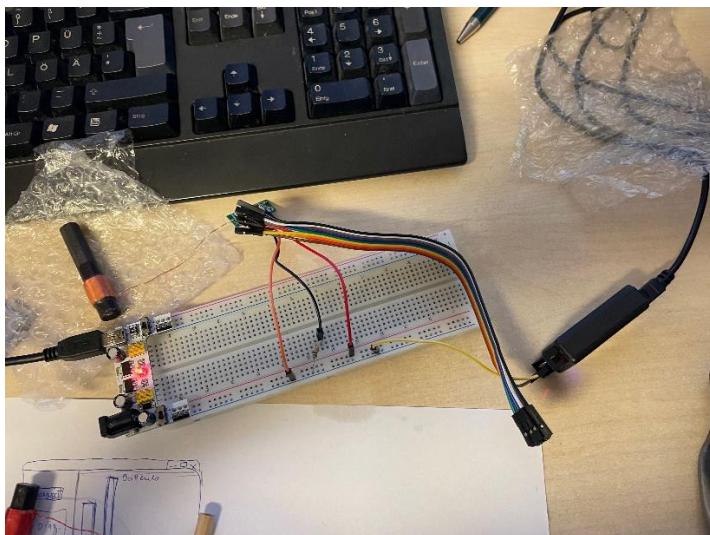
12. ábra HEAD

## 5 DCF77 mérése, megértése

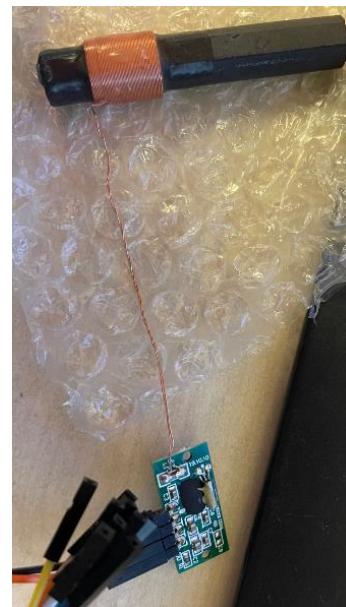
A félév vége során sajnos elég sok idő elment arra, hogy rendelt nyákok megérkezzenek (vámkezelés). Viszont ebben az időben bread board segítségével tudtam kísérletezni a különböző fő eszközeimmel. Egyik ilyen eszköz volt a DCF77-es vevő modul.

### 5.1 DCF77

A DCF77-es modul amit a projekt során használtam, képes a 77,5 kHz-en érkező jelek pontos vételére, és ezek digitális jellé történő átalakítására. A Mainhausen-i rádióállomás sugározza ki erre a frekvenciára a frankfurti atomóra pontos idejét. Ehhez hasonló modulokat Európa szerte gyakran használnak, például autókban, mikrohullámú sütőkben, órákban.



13. ábra DCF77 mérési összeállítás



14. ábra DCF77 modul

A digitális jeleink kódolva érkeznek, helyes működés esetén minden egyes másodpercben érkezik egy érvényes bit. Bekapcsolt állapotban egy ilyen vevő folyamatosan fogadja a feldolgozott jeleket, ezért ezeket a modulokat általában egy mérés időtartamáig bekapsolják, ezt követően pedig kikapsolják.

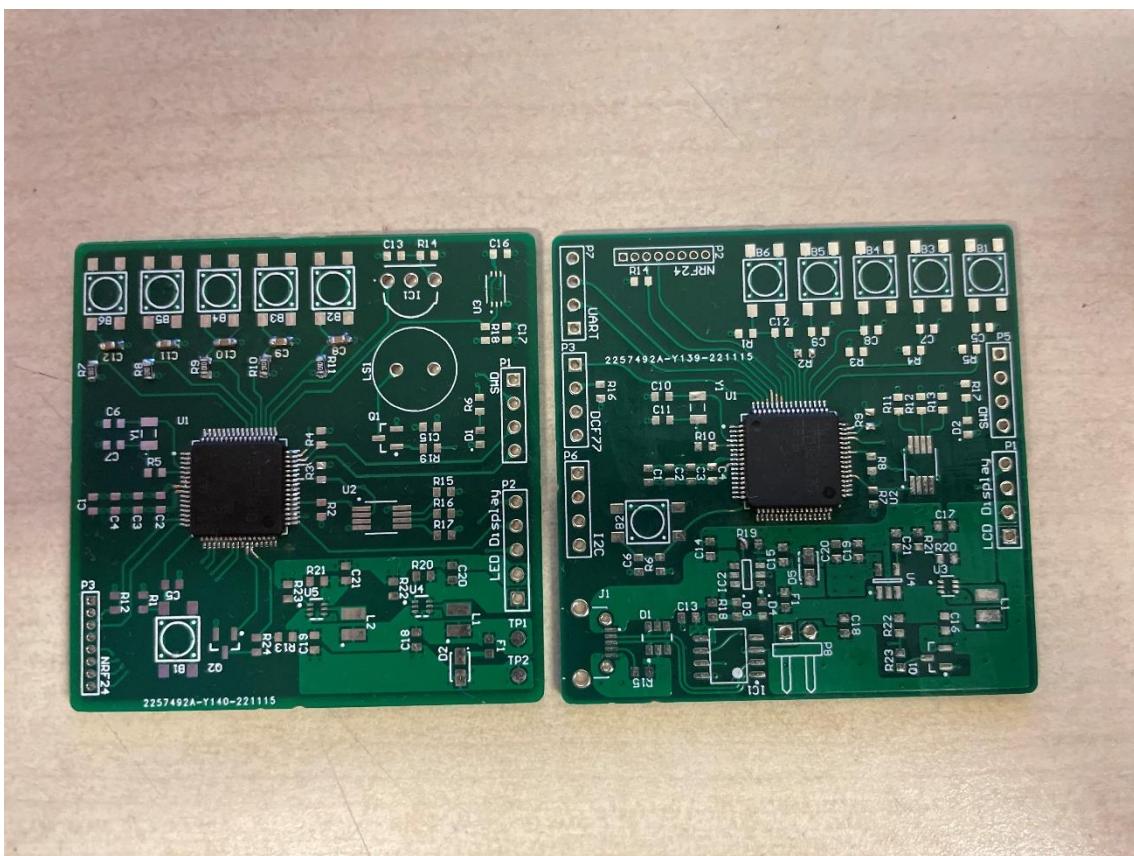
A mérési elrendezésben egy logikai analizátor segítségével vizsgáltam a bejövő jeleket, illetve két alkalommal oszcilloszkóppal is, de sajnos a méréseim során nem jött be számomra hasznosítható adat. A logikai analizátoros esetben rengeteg zavar volt a jelen, ez valószínűleg abból származott, hogy hosszabb vezetékeket használtam, és nem volt szűrő kapacitásom. Az oszcilloszkópos mérések során pedig nem jött be egyáltalán jel az eszközből. Valószínűleg a mérések hibája a Schönherz beton/vas szerkezetéből ered, emiatt már 77,5 kHz-es jelek sem jutottak be az épületbe.

A jövőben egy szabadtéri tesztelést is szeretnék végezni, illetve a HEAD-et elláttam extra kommunikációs interface-ekkel (I2C és UART), ennek segítségével pedig a későbbiekben van lehetőségem egy GPS alapú időmeghatározásra is.

## 6 Összegzés, jövőbeli tervezek

Az önálló labor projekt során elsajátítottam a nyáktervezés alapjait. A tervezés első lépcsőfokától a PCB összeszereléséig (forrasztás). Részleteibe menően megtanultam az Altium Designer használatát a két nyák tervezése során. Szerintem az önálló labor projektem fő célját elérte, és a feladat megoldásához sikeresen el tudtam készíteni mind a két nyákat.

A jövőben szeretném a két nyáknak a firmware-ét is elkészíteni. Ehhez FreeRTOS-t szeretnék használni (erre a nyákunk fel van készítve 36 KB RAM). Illetve a HEAD-et tervezés során elláttam plusz bővítési lehetőségekkel, esetleg a jövőben hozzáadhatok a projekthez egy barométert az időjárás meghatározására. Illetve ha a DCF77-es vevőt nem sikerül működő állapotba hozni, abban az esetben egy GPS alapú idő szinkronizációra is felkészítettem a nyákat. A pontos idő a műholdakon keresztül is lekérdezhető (ilyen alapon működnek például okostelefonok).

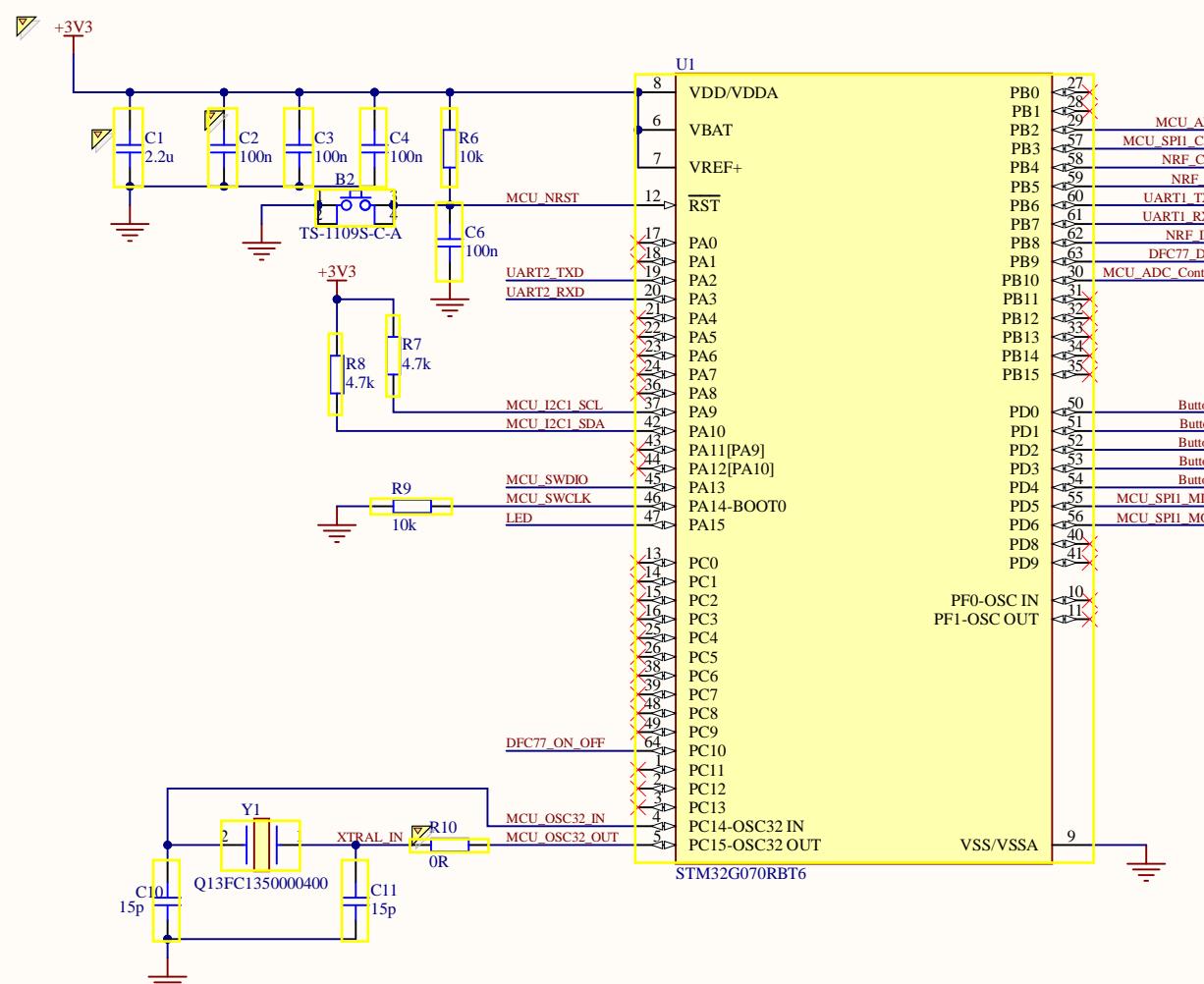
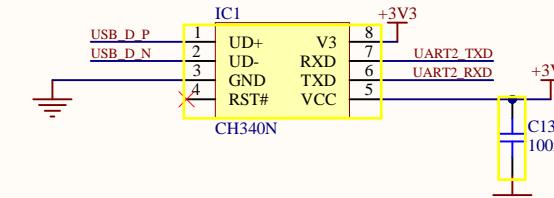
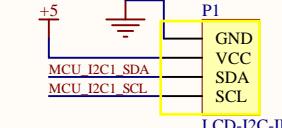
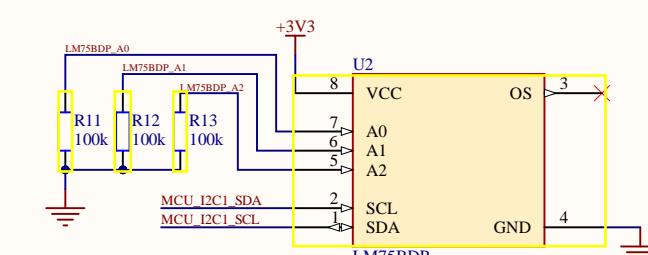
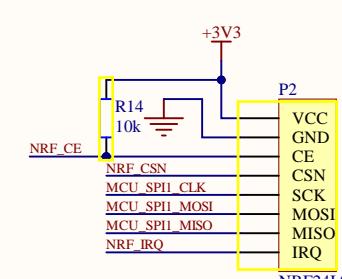
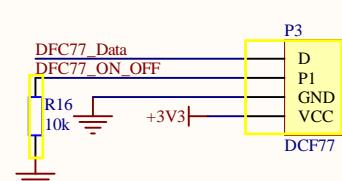
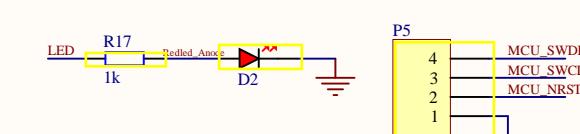
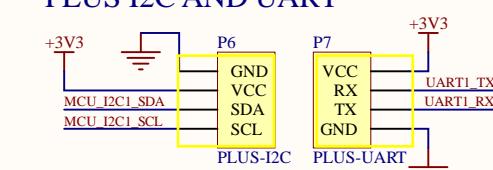


**15. ábra A két nyák az összeállítási folyamat közben**

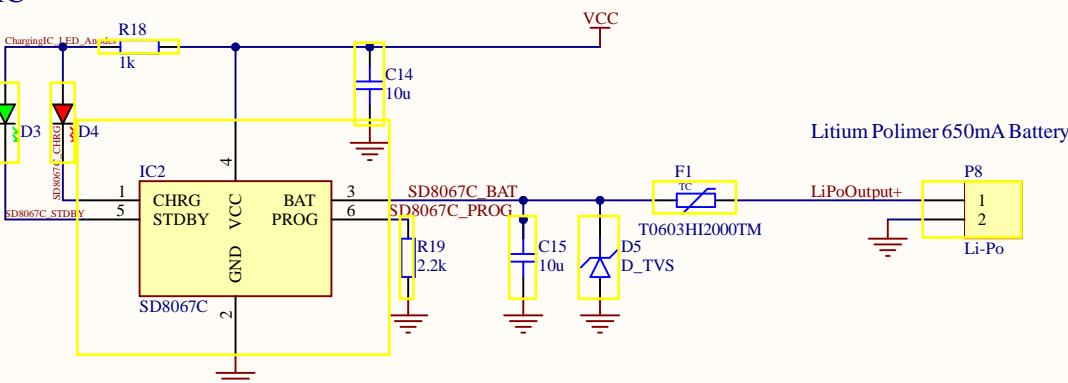
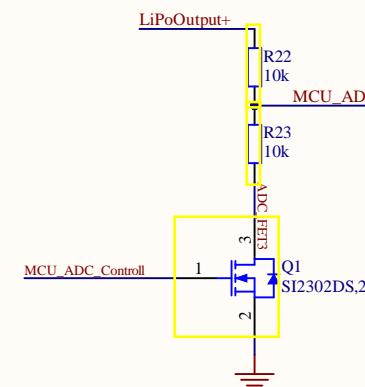
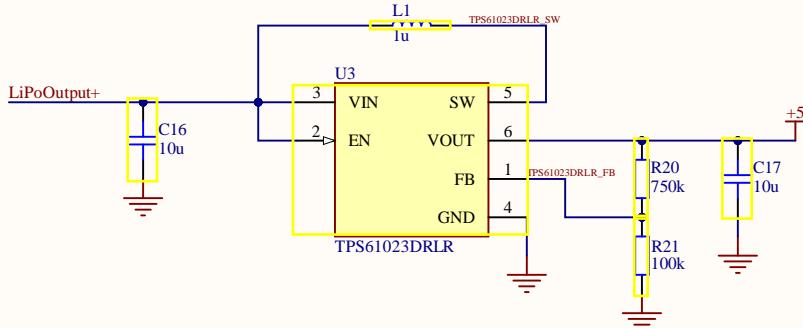
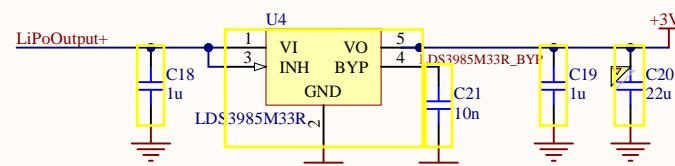
## Irodalomjegyzék

- [1] LDS3985M33R datasheet <https://www.st.com/resource/en/datasheet/lds3985.pdf> (revision 2022.12.10)
- [2] TPS61023 3.7-A Boost Converter with 0.5-V Ultra-low Input Voltage <https://www.ti.com/lit/ds/symlink/tps61023.pdf?ts=1670680001608&refurl=https%253A%252F%252Fwww.google.com%252F> (revision 2022.12.10)
- [3] Create a new DC/DC power design <https://webench.ti.com/power-designer/switching-regulator> (revision 2022.12.10)
- [4] Oscillator design guide for STM8AF/AL/S, STM32 MCUs and MPUs [https://www.st.com/resource/en/application\\_note/cd00221665-oscillator-design-guide-for-stm8afals-stm32-mcus-and-mpus-stmicroelectronics.pdf](https://www.st.com/resource/en/application_note/cd00221665-oscillator-design-guide-for-stm8afals-stm32-mcus-and-mpus-stmicroelectronics.pdf) (revision 2022.12.11)

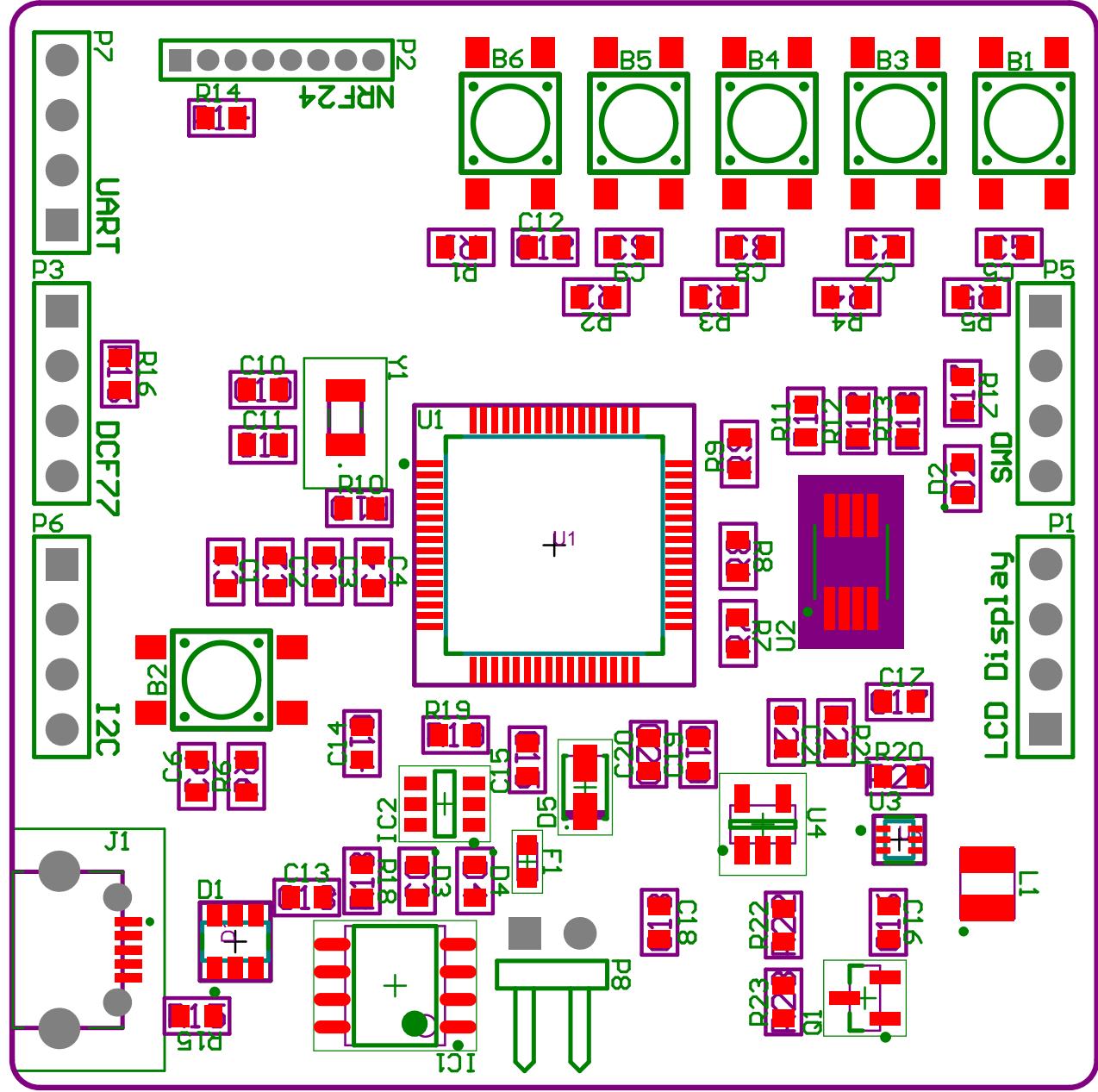
## Függelék

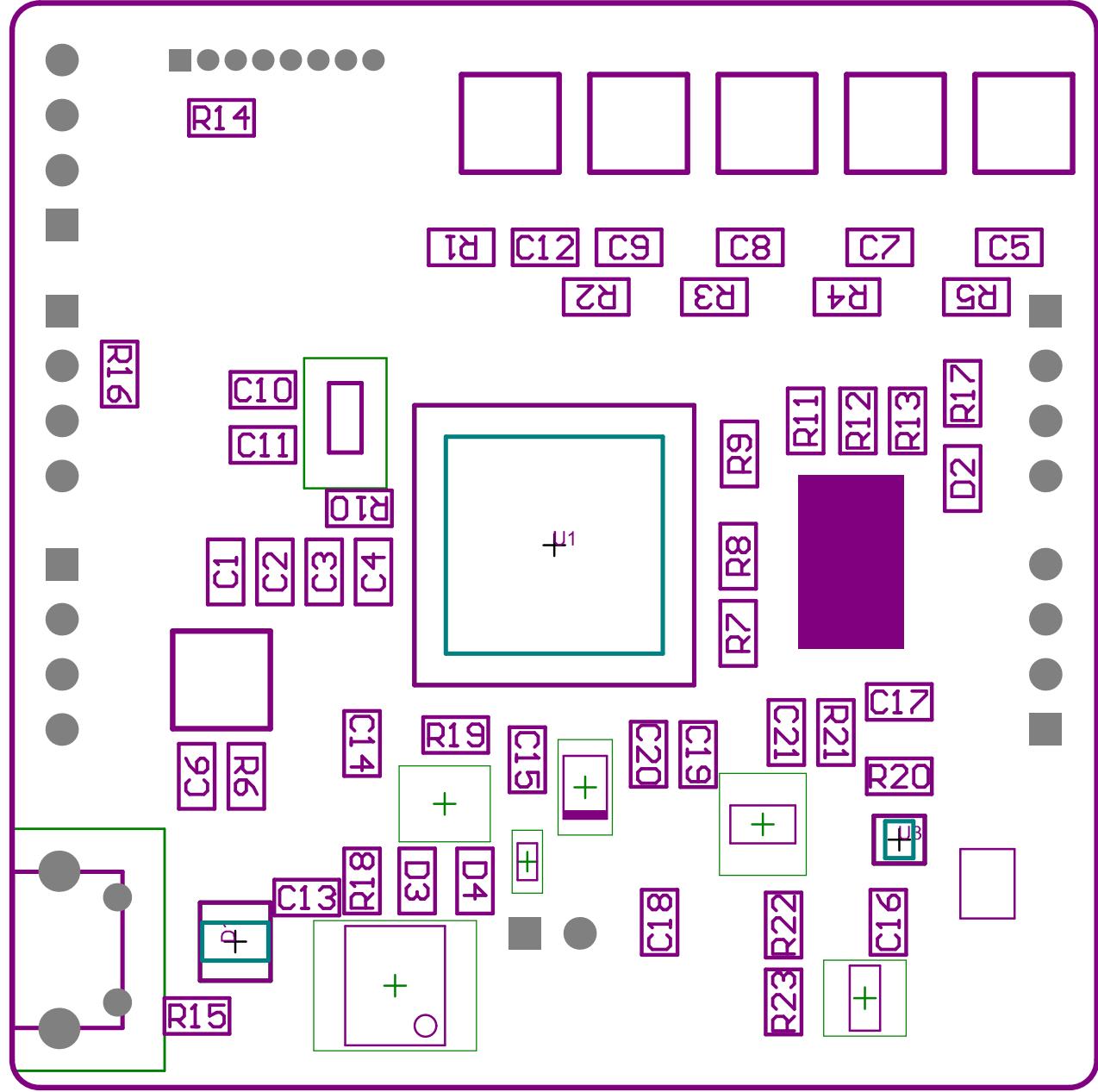
**uController****USB connection + to UART + ESD protection IC****LCD Interface****Temp sensor****NRF24****DCF77****SWD****PLUS I2C AND UART**

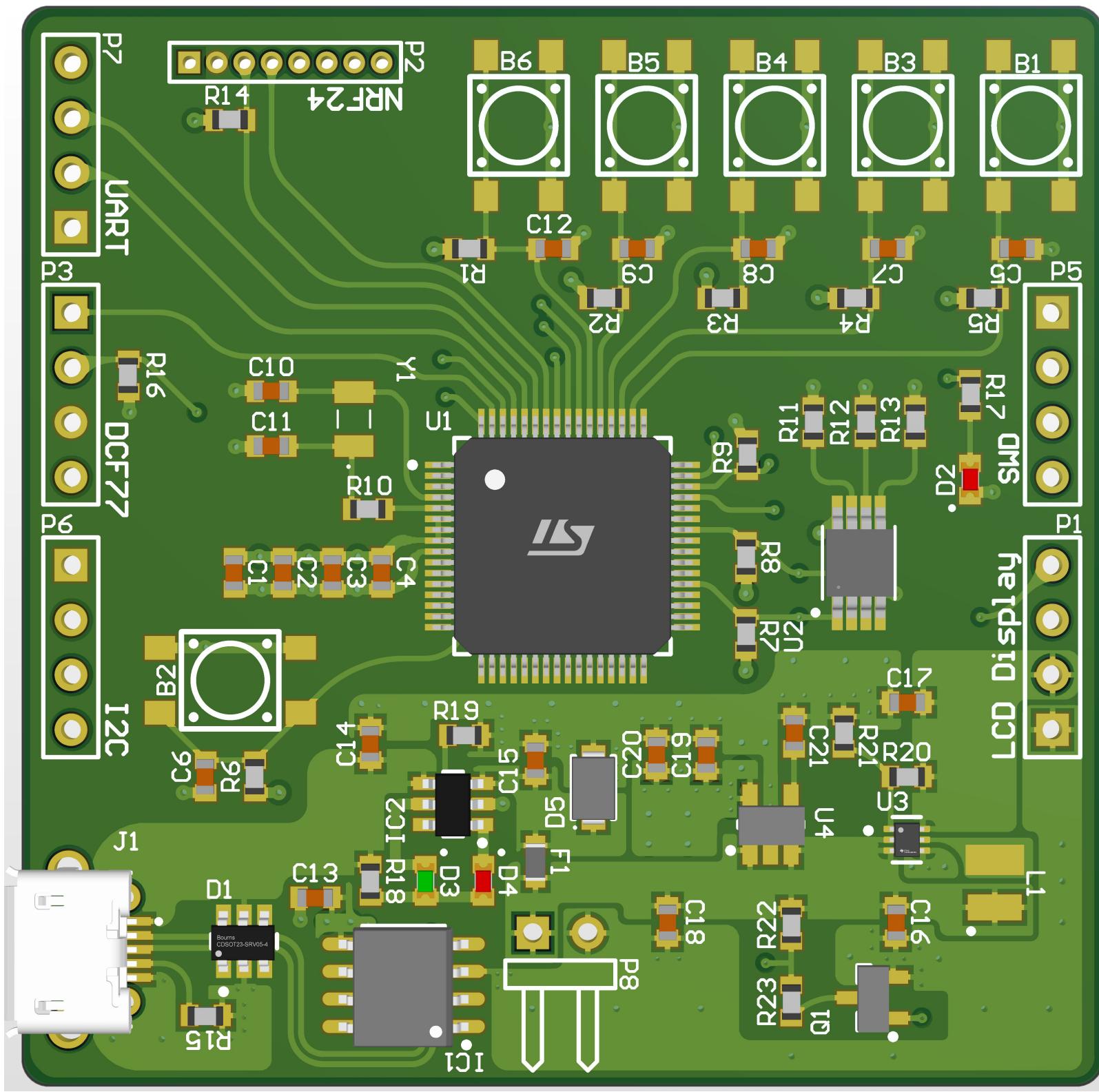
Title		
Size	Number	Revision
A3		
Date: 12.11.2022	Sheet of	
File: D:\PCB\...\uController_and_communication	1	Dominik Varga

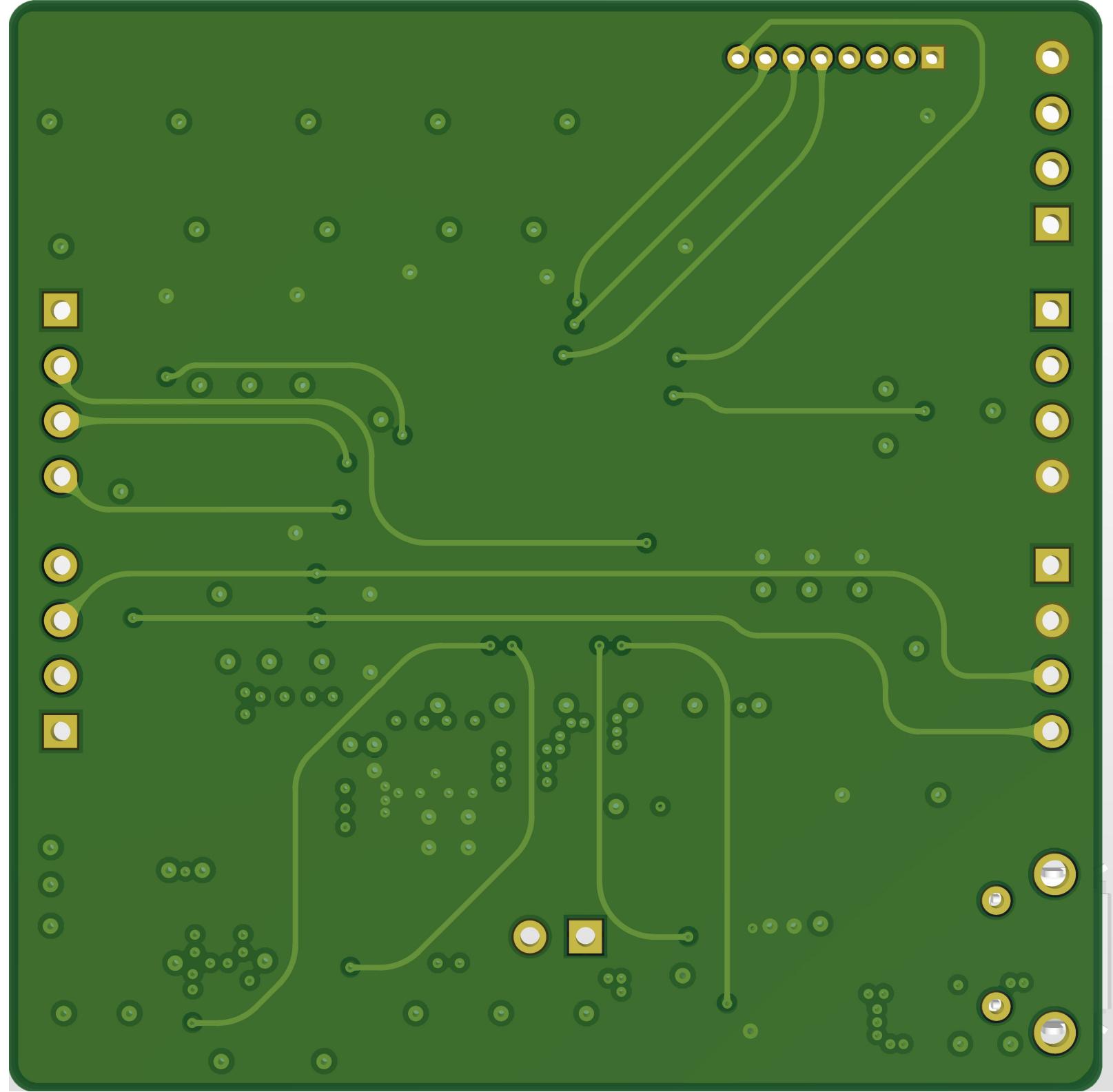
**Charging IC****ADC Lipo measure****Step Up Converter up to 5V****LDO - 3.3V low power 250mA**

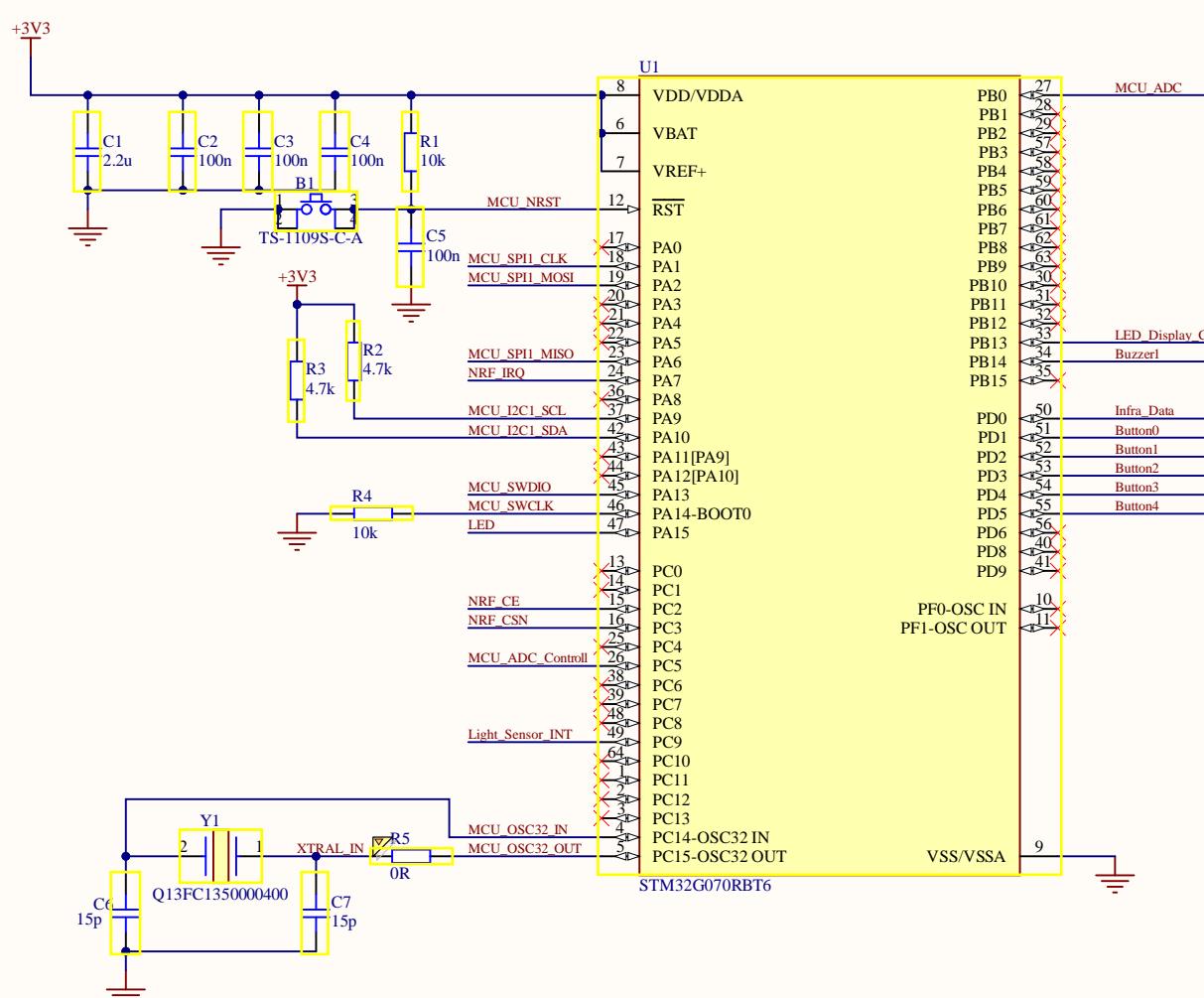
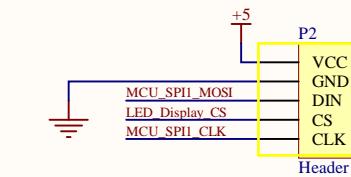
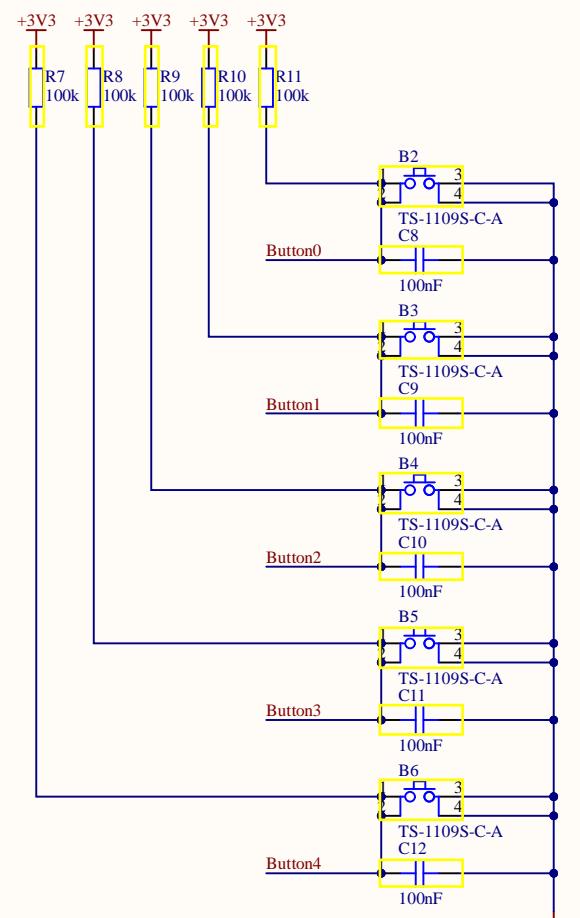
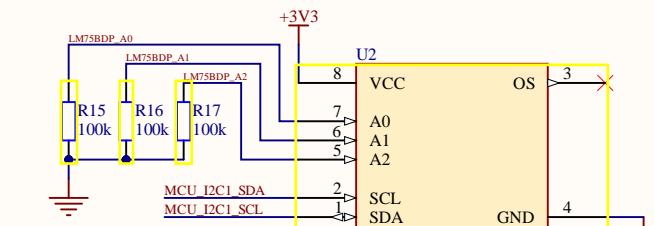
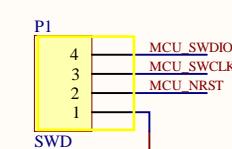
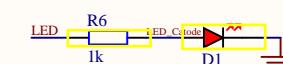
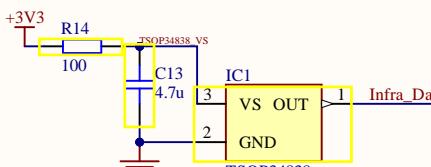
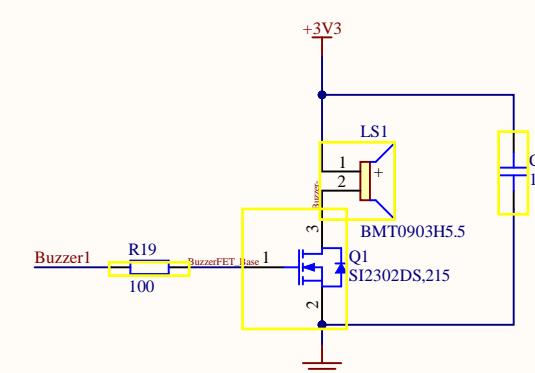
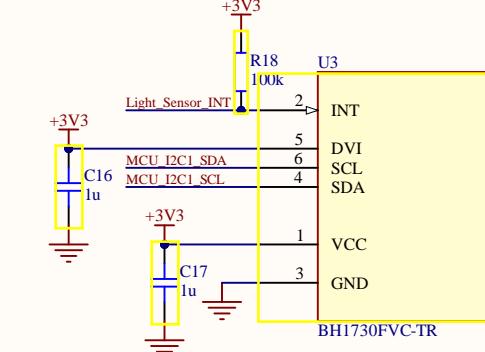
Title <b>Intelligent Clock HEAD</b>		
Size A4	Number	Revision
Date: 12.11.2022		Sheet of
File: D:\PCB\firsttry\Power.SchDoc		Drawn By: Dominik Varga





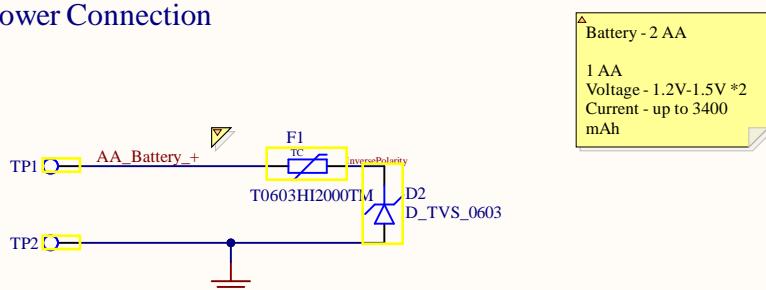




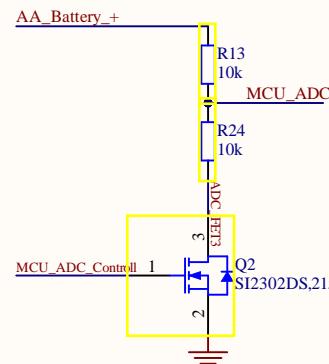
**uController****Interface****Controll Interface****Temp sensor****SWD****Infra reciever****Buzzer****Ambiant light sensor****Intelligent Clock NODE**

Size	Number	Revision
A3		
Date: 12.11.2022	Sheet of	
File: D:\PCB\...\uController_Utils.SchDoc		Drawn By: Dominik Varga

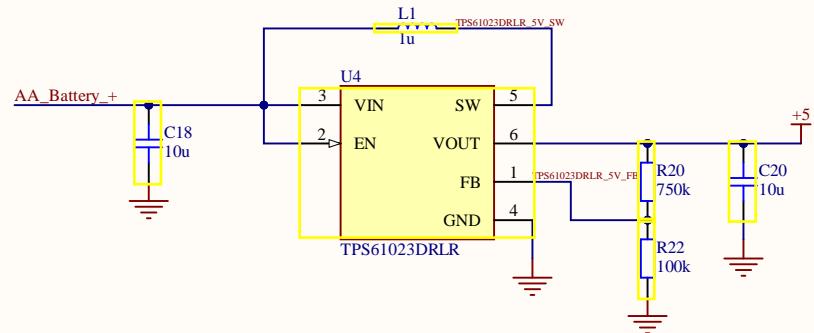
### A Power Connection



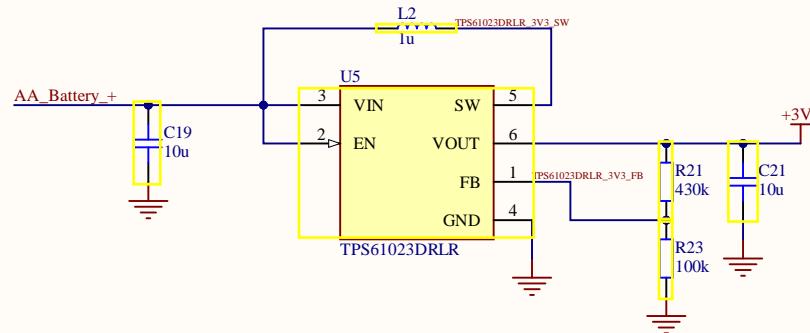
### A ADC battery measure



### C Step Up Converter up to 5V



### C Step Up Converter up to 3V3



Title <b>Intelligent Clock NODE</b>		
Size A4	Number	Revision
Date: 12. 11. 2022		Sheet of
File: D:\PCB\..\Power_Supply_Unit.SchDoc		Drawn By: Dominik Varga

