

*BMSZC Bolyai János
Műszaki Szakgimnáziuma És
Kollégiuma*

1134 Budapest Váci út 21

A pályamunka címe:

ÉBRESZTŐÓRA

Készítette: Deli Tamás

Felkészítő tanár: Bodor Gyula

Tartalomjegyzék

1	Tervezés	2-9
1.1	Konkrét funkciók kiválasztása	3
1.2	A vezérlő kiválasztása a funkciók megvalósításához	3
1.3	Vázlat készítése	4
1.4	Breadboard-on való megvalósítás	4-5
1.5	Áramkör bevitele Eagle NYÁK tervező programba	5-8
1.6	Nyák megtervezése	8-9
2	Megvalósítás	9-12
2.1	A NYÁK elkészítése vasalós technikával	9-11
2.2	Beültetés	11-12
2.3	Áramkörök élesztése és vizsgálata	12
3	Kapcsolások magyarázata	13-23
3.1	Óra-blokk	13-19
3.2	LCD-blokk	19-22
3.3	Tápláló-blokk	22-23
4	Program	23-27
4.1	LCD vezérlő programja	23-24
4.2	Óra-blokk vezérlő programja	24-26
4.3	Programtesztelés	27
5	Doboz	27-28
5.1	A doboz elkészítése	27
5.2	Összeszerelés	28

6	Készülék tesztelése és észrevételek	28
7	Fejlesztési lehetőségek	28
8	Költségek	28-30
9	Irodalomjegyzék	30-31

A feladat leírása:

A feladat egy multimédiás ébresztőóra megalkotása, mely különböző módokon képes ébreszteni. A feladat célja ezenkívül a legfrissebb kezelési felületek alkalmazása a minél egyszerűbb felhasználáshoz.

1. Tervezés:

1.1 A konkrét funkciók kiválasztása

Az ébresztő óra ötlete a reggeli felkelés nehézségéből fakadt. Azt akartam megoldani, hogy több eszköz tegye kényelmessé és hatékonyá az ébresztés műveletét.

Ennek megfelelően az ébresztőórának képesnek kell lennie

- a) Zenét lejátszani: ez szükséges a narrátor miatt, mely a könnyebb információcserére használható ember és gép között, ezenkívül a reggeli ébresztésnél a kedvenc zenék lejátszására és alapértelmezett kellemes dallamok (madárcsicsergés, erdő hangjai) lejátszására alkalmas
- b) Hőmérsékletet mérni, így plusz információt szolgáltatva a felhasználónak
- c) Rezgéssel ébreszteni, ilyen módon javítva a felkelés valószínűségét
- d) Kijelezni az aktuális időt és a beállításhoz szükséges információkat
- e) Érintőképernyős vezérléssel bevinni az adatokat
- f) Nyomógombbal érzékelni a főbb beállítások változását
- g) Más eszközökről való beállítás lehetőségét Bluetooth vezeték-nélküli kommunikáció segítségével fenntartania
- h) Az idő pontos mérésére a tápfeszültség eltávolítása esetén is
- i) Hangot felismernie
- j) Fényhatást keltenie, az ébresztés hatékonyságának növeléséért, ezenkívül egy táv-vezérelhető lámpa szerepét is felveszi ily módon

1.2 A vezérlő kiválasztása funkciók megvalósításához

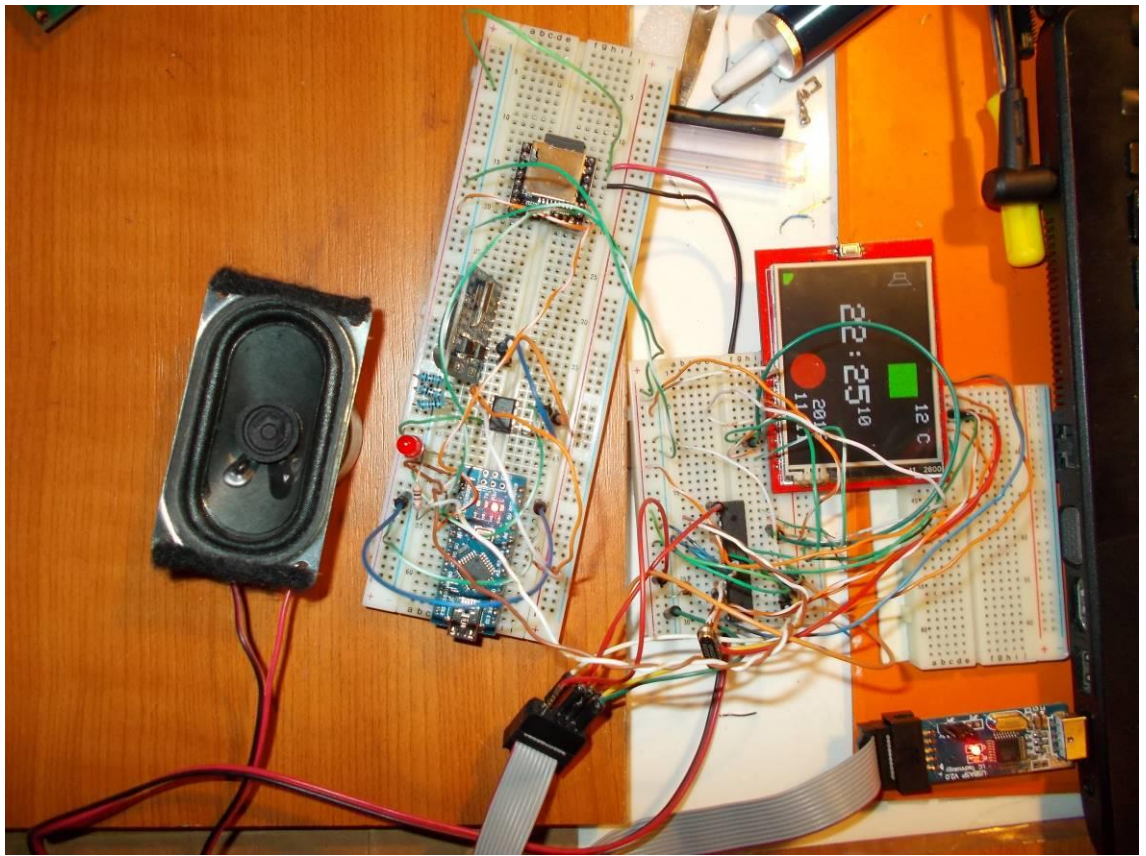
Azért választottam AVR mikrokontrollert mert van tapasztalatom és tudásom hozzá, ezenkívül kompatibilis az Arduino keretrendszerével, mely rengetek modullal képes kommunikálni, ezek között van egy olcsó érintőképernyős kijelző. A kijelző volt a következő tényező, mely miatt az AVR mikroprocesszor-család mellett döntöttem.

1.3 Vázlat készítése

A vázlat készítése során derült ki hogy a kiválasztott funkciókhoz szükséges ki- és bemenetek száma túl sok egy ATMEGA328 (28 láb) mikroprocesszorhoz. Azért nem választottam ATMEGA16 –ot (40 láb) mert a munka szerkezeti és szoftveres felépítése is leegyszerűsödött két ATMEGA328 elhasználásával, valamint a 16-os mikrovezérlő nem tökéletes az Arduino rendszeréhez.

Használhattam volna ATMEGA2560 (100 láb), vagy ATMEL ARM (144 láb) processzort is, ezek az SMD tokozás miatt nem jöhettek szóba. Az áramkört három blokkra osztottam, az egyik az LCD kijelző vezérlésével, az érintőképernyő és a nyomógombok értékeink feldolgozásával van elfoglalva, ezt ettől kezdve LCD-blokknak fogom nevezni. A második blokk (óra-blokk) a zene az mp3 fájlok lejátszását és audio-jel erősítését, a tápfeszültség-független óra kiolvasását, a rezgő motor működését, a hőmérséklet érzékelő kiolvasását, a Bluetooth kommunikációra képes eszköz működését vezérli. A harmadik blokk (táp-blokk) a tápellátást biztosítja, valamint az izzót vezérlő áramkör is itt kapott helyet.

1.4 A vázlaton elkészült kapcsolási rajzok tesztelése breadboard-on.

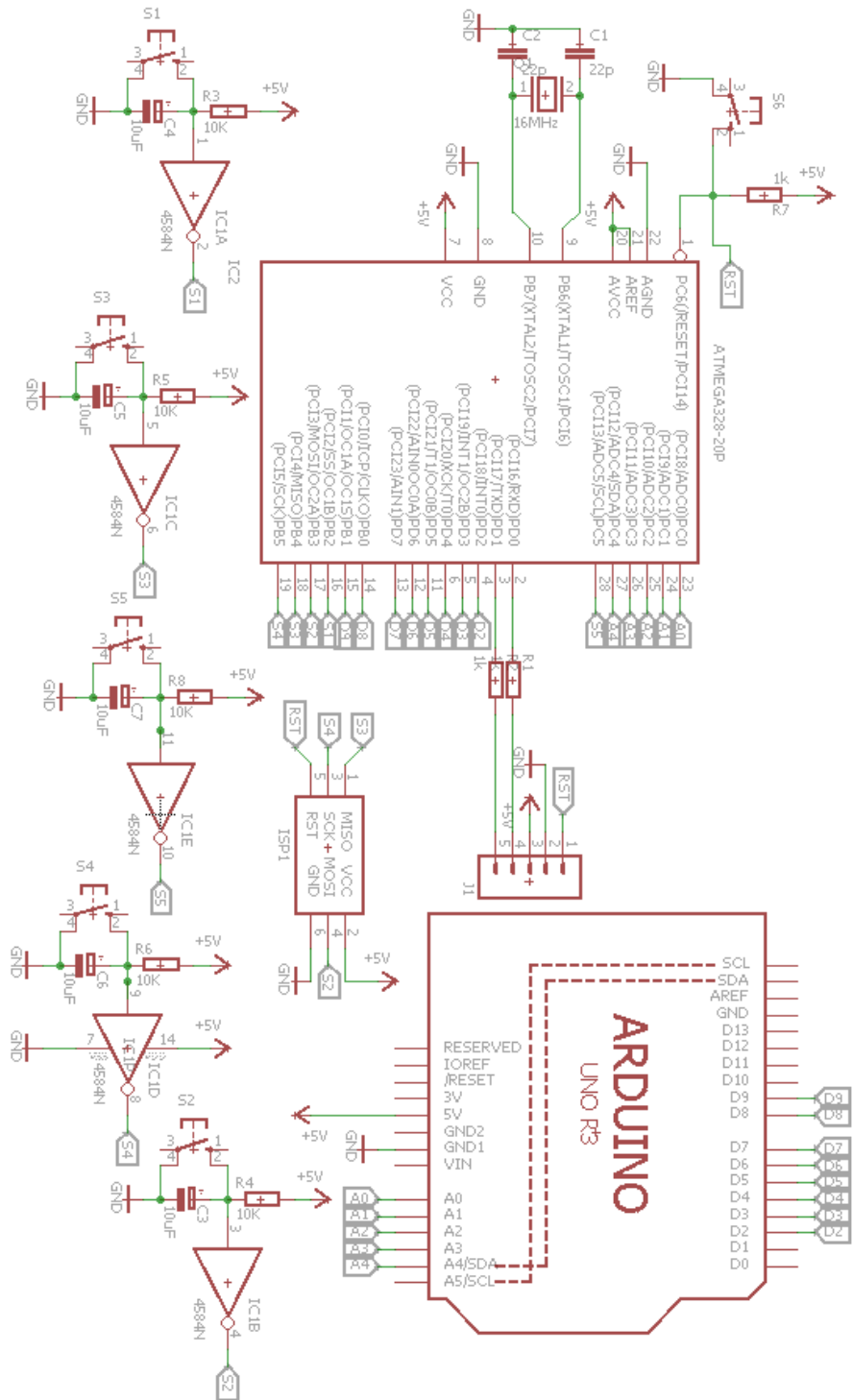


1.ábra Breadboard-on való összeszerelés

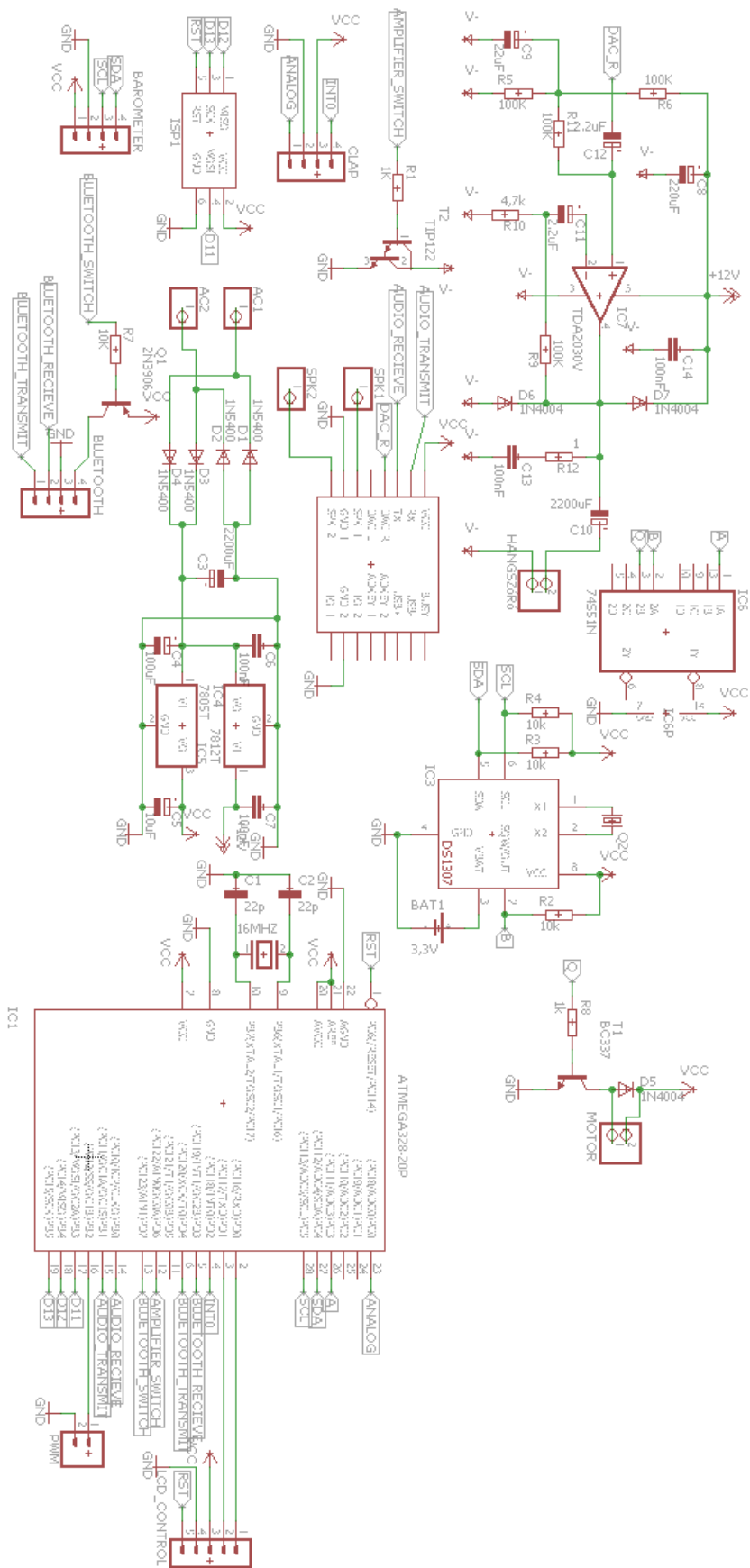
A Breadboard-os teszt során a részegységek működését megvizsgáltam, így bizonyosodván meg a kapcsolási rajz helyességéről és az alkatrészek viselkedéseinek sajátosságáról.

1.5 Áramkör kapcsolási rajzának bevitele Eagle NYÁK tervező programba az elkészített vázlat alapján (2-4.ábra).

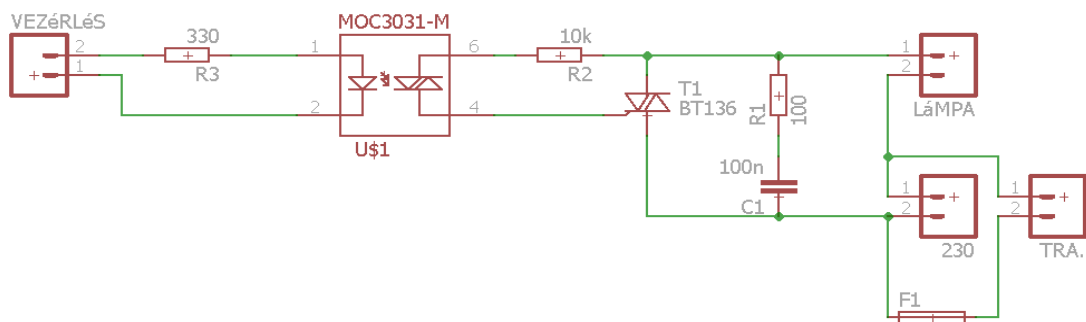
A programhoz nem tartozékként járó alkatrészek letöltése.



2.ábra Eagle kapcsolási rajz az LCD blokkhoz



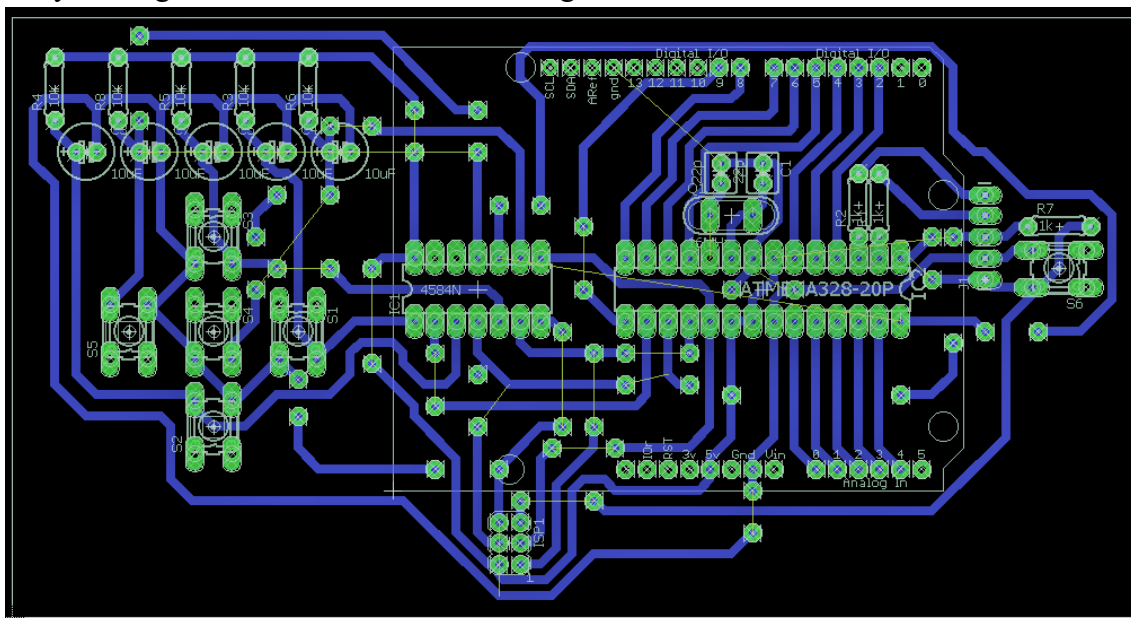
3.ábra Eagle kapcsolási rajz az óra blokkhoz



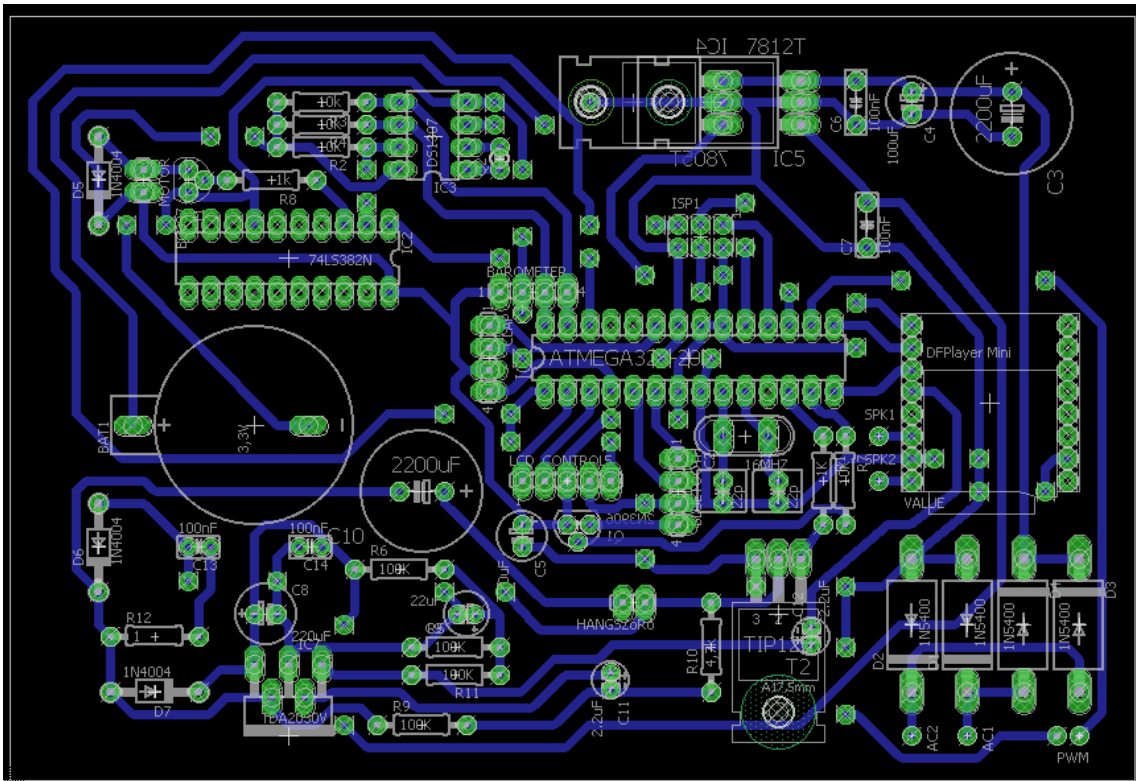
4.ábra tápláló blokk Eagle kapcsolási rajza

1.6 A NYÁK megtervezése.

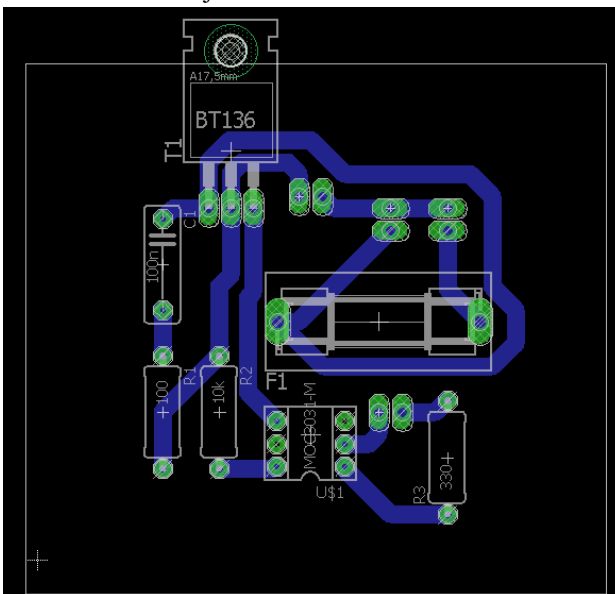
Az áramkör egyszerűsítése az alkatrészek csoportjainak és helyzetének figyelembevételével. Az áramkört egyoldalásra terveztem ezért és a bonyolultsága miatt átkötésekre van szükség.



5.ábra Eagle beültetési rajz az LCD blokkhoz



6.ábra beültetési rajz az óra blokkhoz



7.ábra beültetési rajz a tápláló blokkhoz

2. Megvalósítás:

2.1 A NYÁK elkészítése fizikailag vasalásos technikával

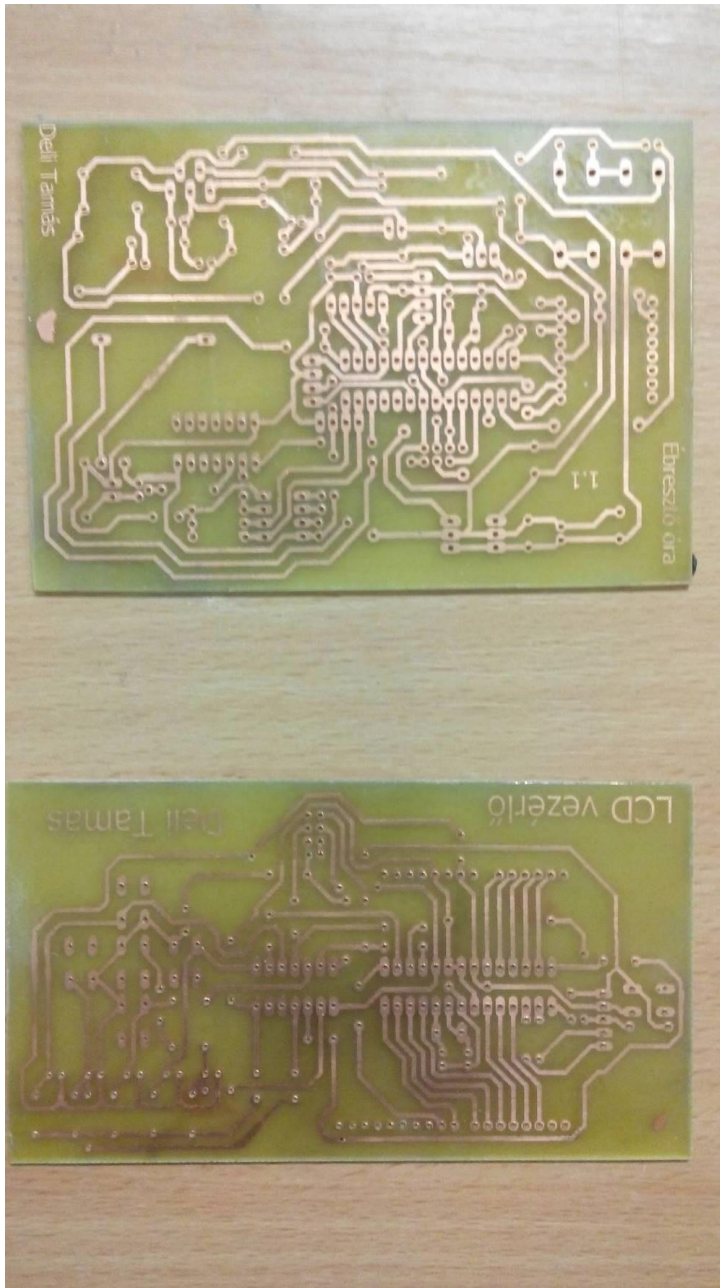
A kész NYÁK tervet lézeres nyomtatóval nyomtattam műnyomó lapra.

A vasalandó műnyomó-lapot méretre vágtam, majd előkészítettem a tiszta NYÁK lapot. Az előkészítés a lemez méretre vágásával, leradírozásával és zsírtalanításával (lakkbenzin) történik.

A vasalandó műnyomó lapot az előkészített lemezre helyezése, majd levasalása. A vasalás 10 percet vesz igénybe. Ezek után hagytam hogy víz alatt felpúposodjon a lemezen lévő papír, ekkor elkezdtem leszedni a papírt a kártyáról és a maradékot rajztűvel távolítottam el óvatosan. A vasalási hibákat lakkfilccel korrigáltam. A maratásra kész áramköri kártyát nátrium-perszulfáttal bekevert meleg buborékoztatott vízben marattam le. Mikor már csak a kívánt vezetősávok és furathelyek látszódtak, kivettem a maratóból és nitro higítóval és egy kendő segítségével eltávolítottam a lemezen maradt tintát és lakkfilcet.

A lemez lefújása forrasztási segédanyaggal majd megszáritása (5 perc).

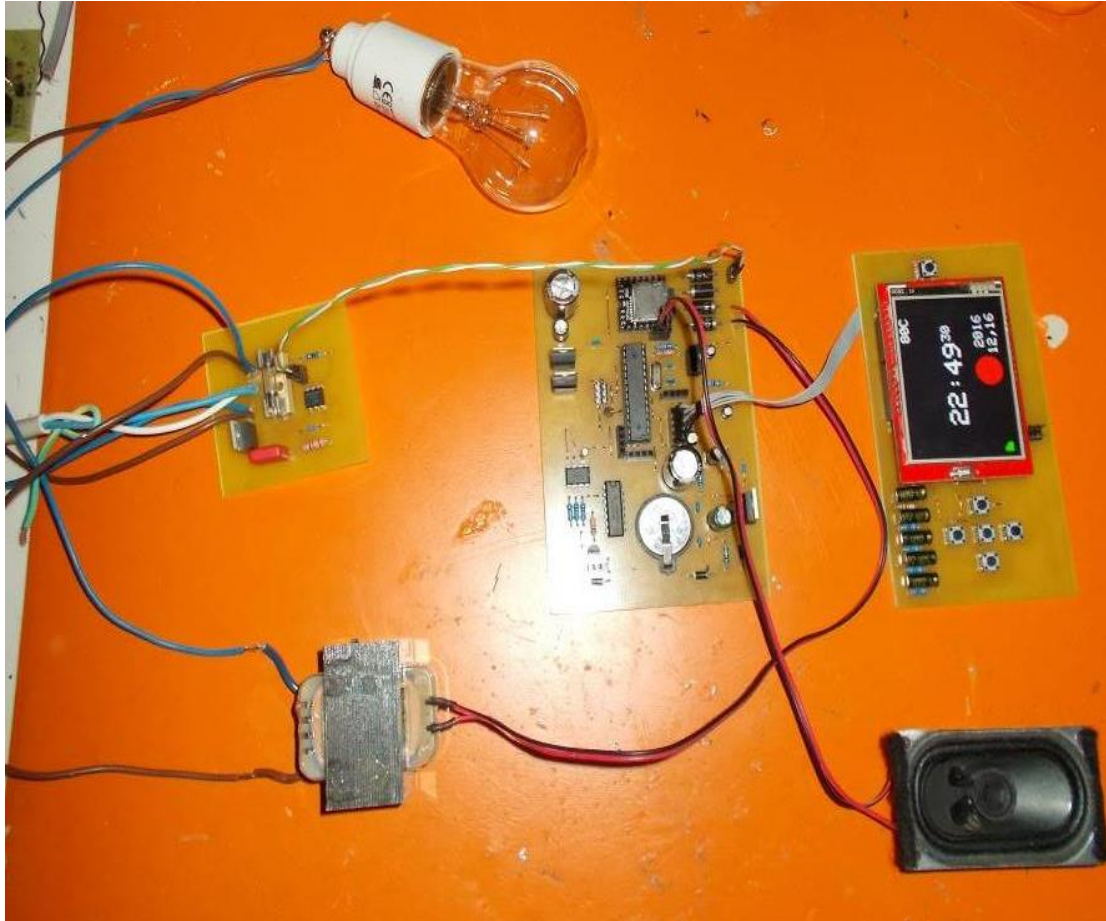
A NYÁK kifűrése 1mm –es fűró segítségével.



8.ábra kész nyáklemez kifűrt állapotban

2.2 Beültetés:

Haladás az átkötésektől kezdve magassági sorrendben a 2200uF-os kondenzátorig.



9.ábra Beültetett nyáklemezek, tesztelés

2.3 Az áramkörök élesztése:

Az élesztést mikrokontrolleres környezet nélkül végeztem el. Ellenőriztem hogy a vezetősávok között van –e kontakt és hogy tápfeszültség el jut –e a kellő pontokra.

A valós élesztésnél az összes alkatrészt beültettem és funkciótesztet végeztem. A mérés sikeres volt.

Megmértem az folyó áramokat:

Váltakozó áramú mérés:

Lámpa: 0,33A

Egyenáramú mérések:

Az áramkör alapállapotban ébresztési állapot nélkül: 0,23A

Ébresztési állapotban: 0,37A

A rezgéskeltő áramkör 80mA-t fogyaszt.

Miután kipróbáltam az áramköröket külön-külön, leteszteltem hogy megfelelően kommunikálnak –e egymással.

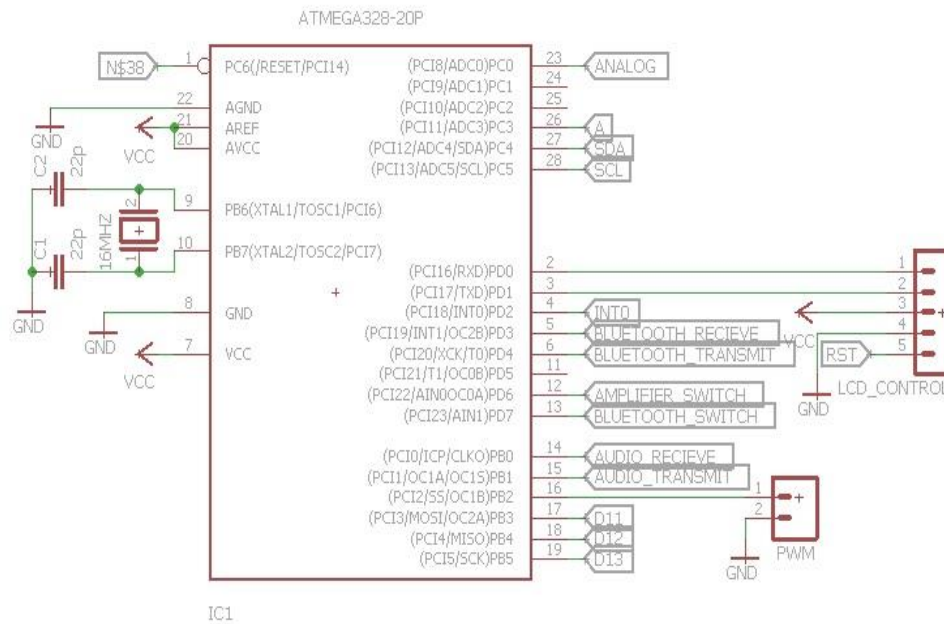
A következőkben bemutatom a három áramkörblokkot részletekre bontva.

3. Kapcsolások magyarázata:

3.1 Az óra-blokk:

3.1.1 Vezérlő

Ahhoz hogy kompatibilissé tegye az Arduinó keretrendszeréhez a mikrokontrollert 16 MHz –es órajelet kell beadnom a külső oszcillátort kezelő lábakra 22 pF-os kondenzátorok kíséretében. A könnyebb fejlesztés érdekében elláttam ISP programozó csatlakozóval a NYÁK-ot. Az óra-blokknak nincsen külön újraindító kapcsolója, ezért önálló működés során lebeg a RESET láb. A két blokk közötti kommunikációt és tápcsatlakozást egy 5 szálás szalagkábel oldottam meg. Az 5 vezeték egyik szála az LCD-blokkon lévő újraindító nyomógommbal van összekötötésben, így képes a két áramkör egyszerre újraindulni, szinkronizálódni. A szalagkábel tartalmazza a pozitív és a negatív tápfeszültséget és egy kétirányú soros kommunikációs protokollal rendelkező csatornát. A soros kommunikáció tulajdonságai 11250, nincsen paritásbit és egy stop bit van. Az ATMEGA328-as mikrovezérlő 5V feszültséggel működik. A vezérlő IC-hez közvetlenül csatlakozik egy kétpólusú csatlakozó, mely a lámpa felkapcsolásáért felelős. A vezérlő galvanikusan izolálva van a 230V-os részektől.



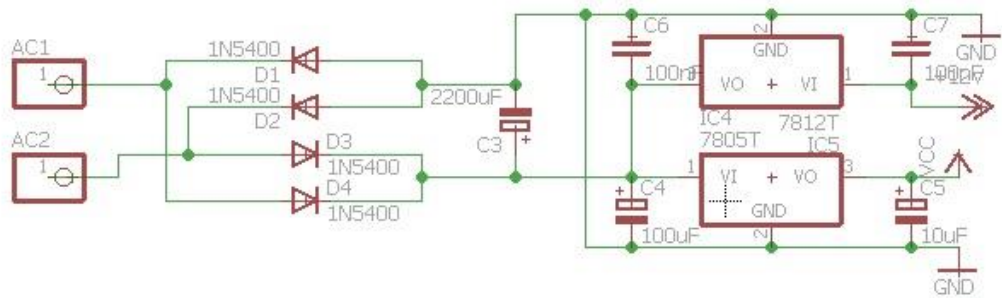
10.ábra LCD vezérlő kapcsolási rajza

3.1.2 A táplálás:

Egy 12V-os transzformátorról működik a kapcsolás.

A váltó feszültséget 16V egyen és 5 V egyen feszültséggé kell átalakítani. Ezt 4db diódával egyenirányítottam, egy 2200uF-os kondenzátorral zajmentesítettem és a

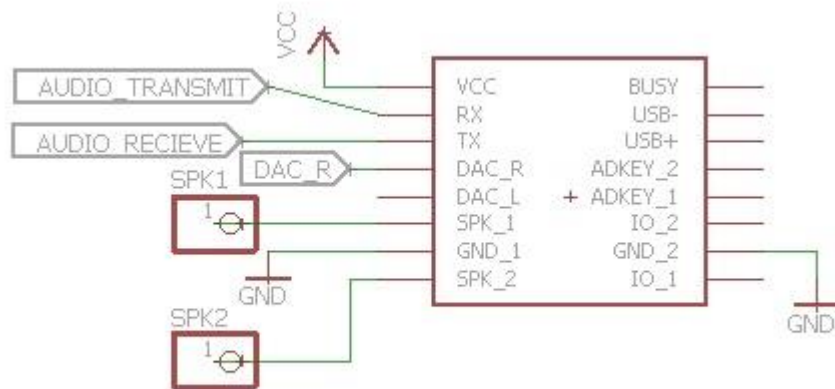
két különböző feszültségértékre megfelelő 7805 és 7812 IC-kkel stabilizáltam. A stabilizátor IC-eket elláttam kondenzátorokkal a gerjedés megakadályozása végett.



11.ábra tápláló kör kapcsolási rajza

3.1.3 MP3 lejátszó modul

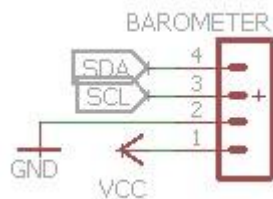
A modult 5V-ról működtetem a lejátszandó zenét egy a modulba beépített micro SD tartóba helyezett memóriakártya tartalmazza. A megfelelő hanganyagot soros kommunikációval üzeni meg a vezérlő a modulnak. A soros kommunikáció tulajdonságai: 9600 bits/s, nincs paritás bit és egy stop bit van.



12.ábra DFPlayer Mini SKU:DFR0299 modul kapcsolási rajzban való megjelenése

3.1.4 Hőmérsékletmérés:

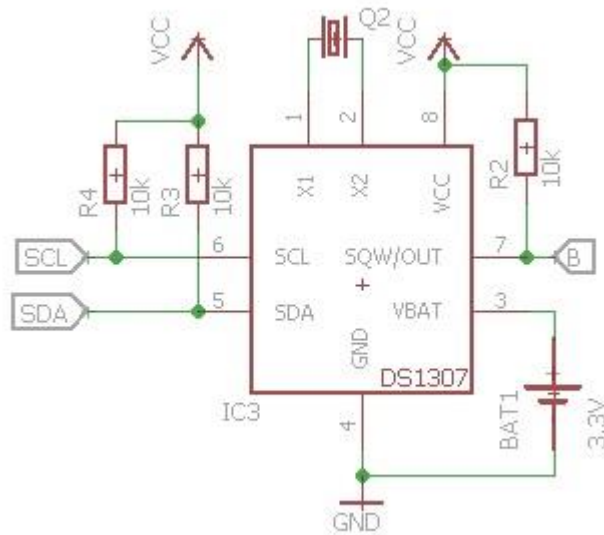
A hőmérséklet mérését egy barométer modullal oldottam meg, mivel ez már otthon megtalálható volt. A barométer modul I2C kommunikációs protokollal küldi el a hőmérsékletet tartalmazó regiszterét a vezérlőnek a vezérlő kérésére. A modul 5V-tal működik.



13.ábra Barométer csatlakozó kapcsolása

3.1.5 óra:

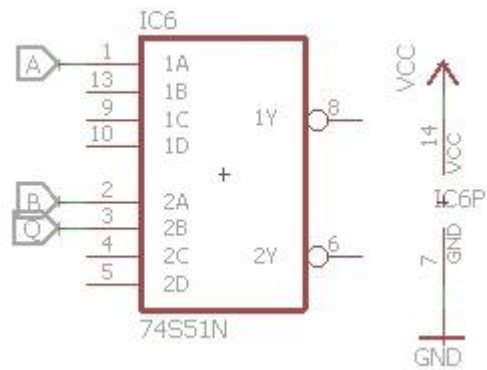
Az óra egy elem tápfeszültségét veszi igénybe a folyamatos működés érdekében mely 3,3V-os. Az óra egy 8 lábú integrált áramkör, megfelelő működéséhez szüksége van egy kvarc kristályra mely 327 KHz-es. Az IC neve DS1307, ez az IC kommunikál ugyanazon I2C csatornára kötve vezérlővel. A kommunikáció sikeressége érdekében az SCL és az SDA lábakat egy 10K ellenállás segítségével pozitív, 5V, tápfeszültségre kell kötni. A funkció IC tartalmaz egy frekvenciagenerátort melyet egy regisztere átírásával 1Hz-es generátorra lehet alakítani. Az IC-ben egy FET van melynek e kimeneti lába lebeg, ezt a megfelelő működéshez ellenálláson keresztül fel kell húzni pozitív tápfeszültségre, mint a kommunikációs lábait. A generátor használatára azért van szükség, mert ennek segítségével a program számára egyszerűbb a rezgés funkciót megoldani.



14.ábra DS1307 valós idejű óra integrál áramkör kapcsolása

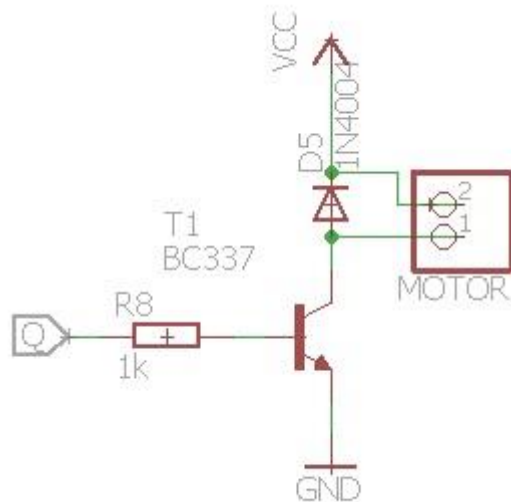
3.1.6 AND –kapu

Az előző IC-vel dolgozik szimbiózisban. Az és kapcsolat egyik változója a vezérlő egyik kimeneti lábára, a másik az frekvenciagenerátorra van kötve, így a logikai függvény másodpercenként negálódik amennyiben a vezérlő megfelelő lába bekapcsol. A megfelelő IC nem volt elérhető a tervezőprogramban, ezért a legközelebbi felépítésben hasonló integrált áramkör került be.



15.ábra És kapcsolatot megvalósító 74S08

3.1.7 Mivel a logikai IC nem bírná a vibrátor motor áramfelvételét ezért egy tranzistorral kapcsolom be a motort. Az induktív terhelés miatt egy diódát alkalmazok az áram visszafolyása elleni védelem érdekében.

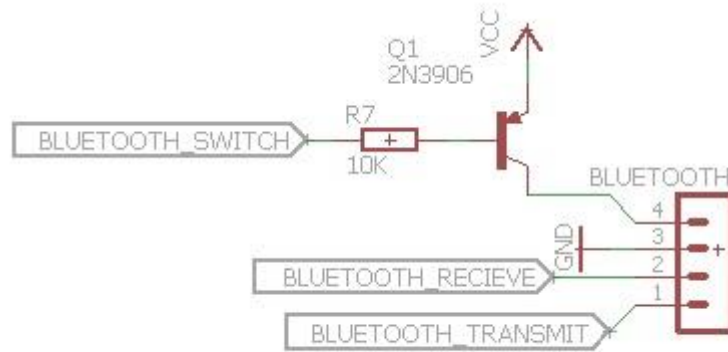


16.ábra Tranzisztoros motorkapcsoló kapcsolási rajza

3.1.8 Bluetooth:

A folyamatos sugárzás elkerülése érdekében egy tranzistorral kapcsolom ki és be a modul tápfeszültségét. A modul soros kommunikációval „üzen” a vezérlőnek a beállítási parancsról.

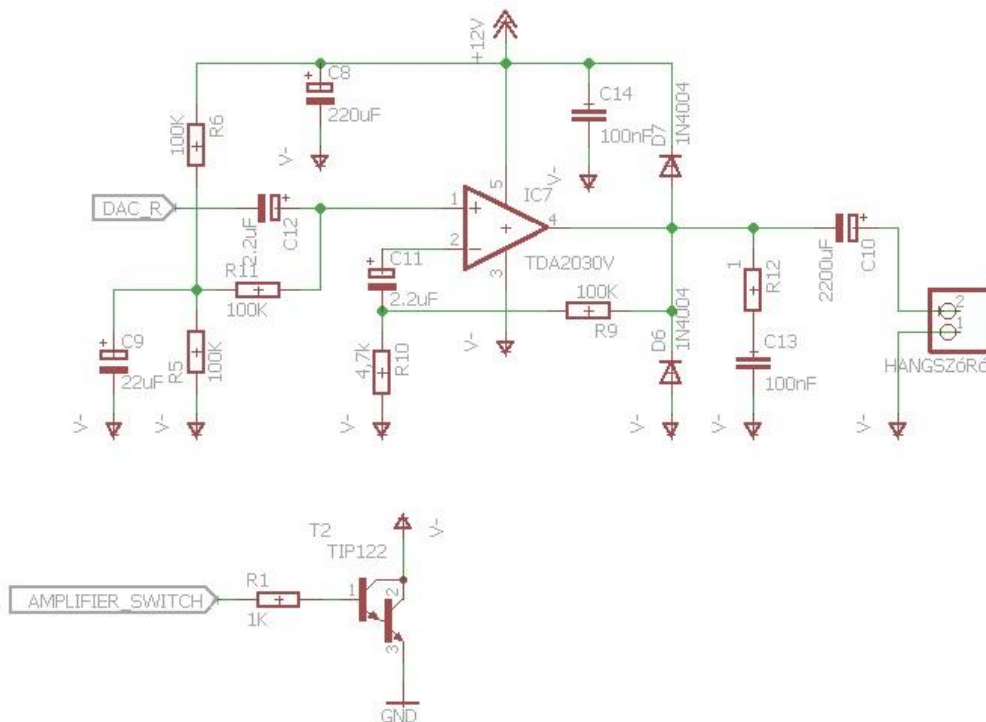
A kommunikáció tulajdonságai, 8 adatbit, 9600 bits/s , nincs paritásbit és egy stop bit van.



17.ábra Tranzisztoros bluetooth kapcsoló kapcsolási rajza

3.1.9 Erősítő:

A blokk tartalmaz egy egycsatornás TDA2030A erősítőt az audio-jel felerősítése miatt. Az erősítőt az áramtakarékoság végett egy NPN darlington (TIP122) tranzisztorral kapcsolom a vezérlő kimeneti lábával. Az erősítő 12V-ról működik.

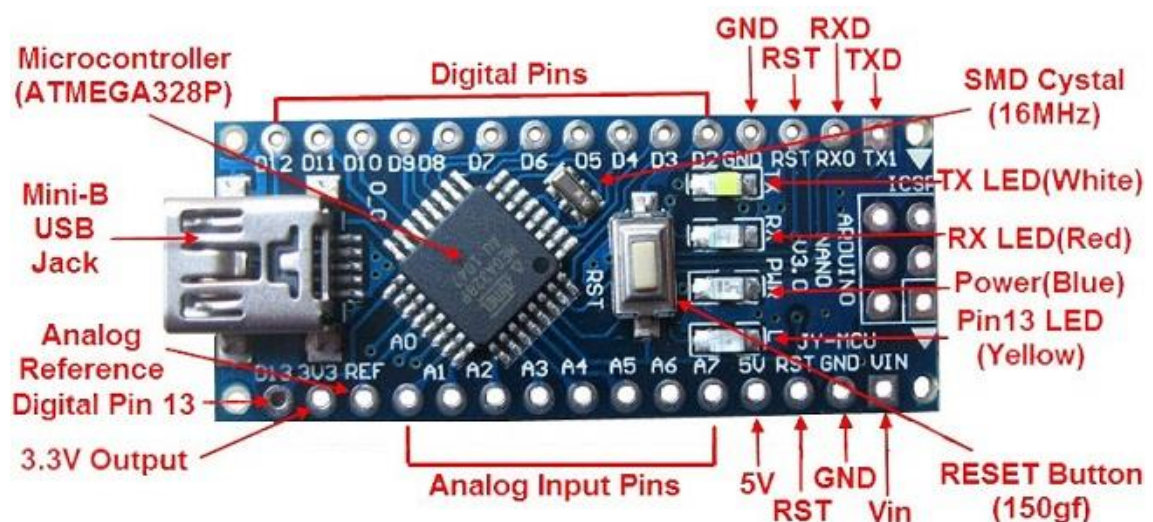


18.ábra TDA erősítő kapcsolási rajza

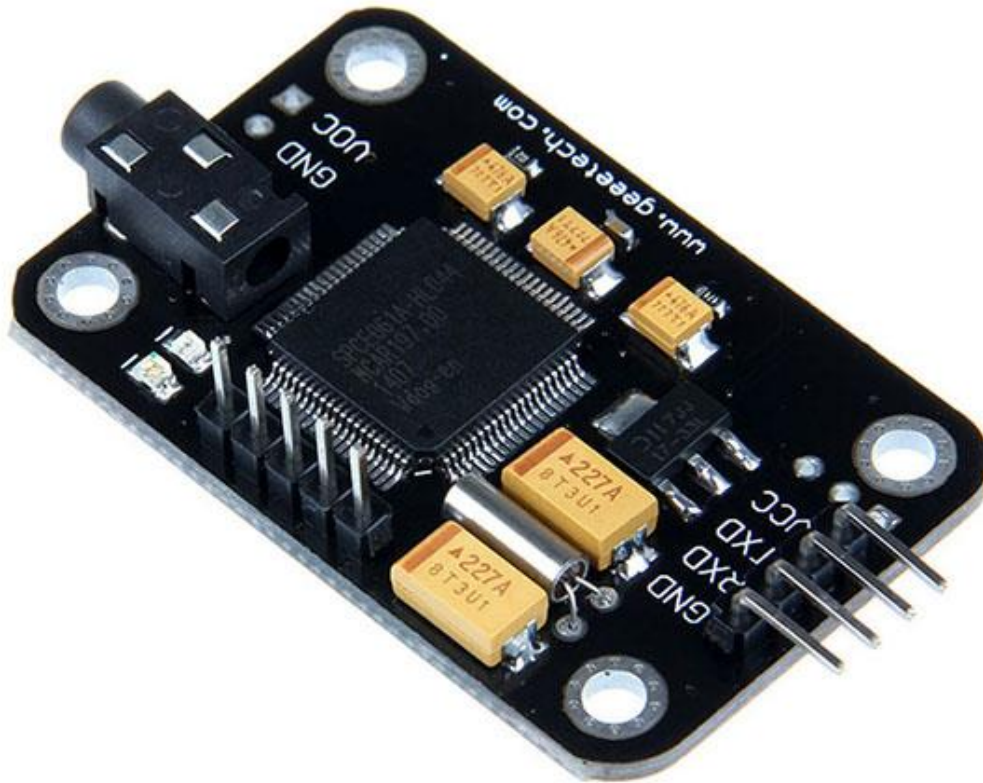
3.1.10 Beszédfelismerő modul:

Mivel a vezérlői memóriája megtelt a hangfelismerés lekezelés számára ezért, egy újabb vezérlő blokkot iktattam a már túlterhelt óra-blokk vezérlője és a hangfelismerő modul közé. Az ok nem csak a programmemória megtelése, hanem az óra-blokk vezérlőjének

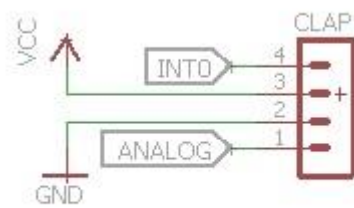
hardveres kapacitásának teljes mértékű kihasználása a kommunikációs protokollokkal működtetett eszközök részéről. Ezért egy már nem saját nyáktervvvel rendelkező vezérlő egységet, az Arduino Nano-t építettem be a dobozba. (A programot a mellékletben lehet megtalálni.) A modulra már a doboz összeszerelése előtt feltöltöttem a felismerésre szükséges hangfájlokat. A hangfájlokat az eszköz meglehetősen nehézkesen és nagy hibával tudja összehasonlítani, ezért több hangfájlt vettem fel más hangsúlyokban, más hangerősségeken és mindegyik hangfájlt egyidejűleg hasonlítja össze a „hallottakkal” így a hiba mértéke csökken. A beállító funkciókat a Nano vezérlő egységnek el kell végeznie az indulás előtt, ez körülbelül 5 másodpercbe kerül. Az indulási beállítások közé tartozik az hogy a 3 * 5 tároló-helyes bankok közötti váltás , ebben az esetben abba a bankba kell váltani amelybe felvettem kezdetben a hangfájlokat, a másik beállítás hogy milyen karakterhalmazt bocsásson a Nano vezérlőegységre a hangfelismerő modul amikor egyező szakaszokat talál. A Nano vezérlő és a hangfelismerő modul Rx és Tx lábait az előzőekhez hasonlóan összekötjük, emellett a tápvonalait is összekötjük. A vezérlő magából digitális jeleket ad ki, magát a „toggle” funkciót is elvégzi, amely azt jelenti ha az mondom hogy, „be” akkor egy kimeneti lába magas szinten lesz addig, amíg nem mondom hogy „ki” , mert ekkor alacsony szintet fog kiadni a kiválasztott digitális lábán. Ezzel a módszerrel teszem lehetővé hogy a már túlterhelt óra- blokk még le tudja kezelni ezt a funkciót, az LCD- blokk még rendelkezik bár programmemóriával, de a két vezérlő között kommunikáció ebben az esetben nem megfelelő, ezenkívül az LCD sok lábat igényel, ezek a lábak „eltakarják” a kommunikációs lábakat, emiatt ez a blokk hardveres okok miatt sem képes erre a feladatra. Az óra-blokk ezen túl könnyedén lekezeli a digitális jelet, mintha csak egy villanykapcsoló jelét kellene feldolgozni.



19.ábra Arduino Nano és lábkiosztása



20.ábra Greetech Voice Recognition Module



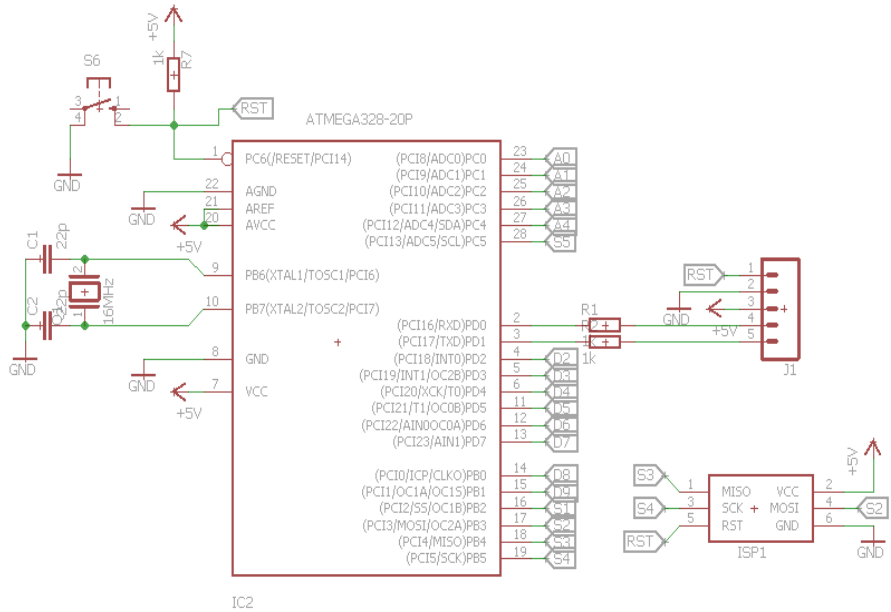
21.ábra Hangfelismerő modul

3.2 Az LCD-blokk

3.2.1 A vezérlő

A vezérlő controller itt is ATMEGA 328 és ezen a NYÁK-on is az Arduino keretrendszerhez van alakítva az áramkör. A két vezérlő kommunikációs csatornája 1k ellenállással van áramkorlátozva.

Ez a panel közvetlenül tartalmaz újraindító nyomógombot. Ezt egy 5 vezetékes szalagkábel továbbítja a tápfeszültséggel és a soros-kommunikációval együtt.



22.ábra Az LCD vezérlő áramkörének kapcsolási rajza

3.2.2 A könnyebb fejlesztés érdekében a kapcsolás tartalmaz ISP programozó csatlakozót.

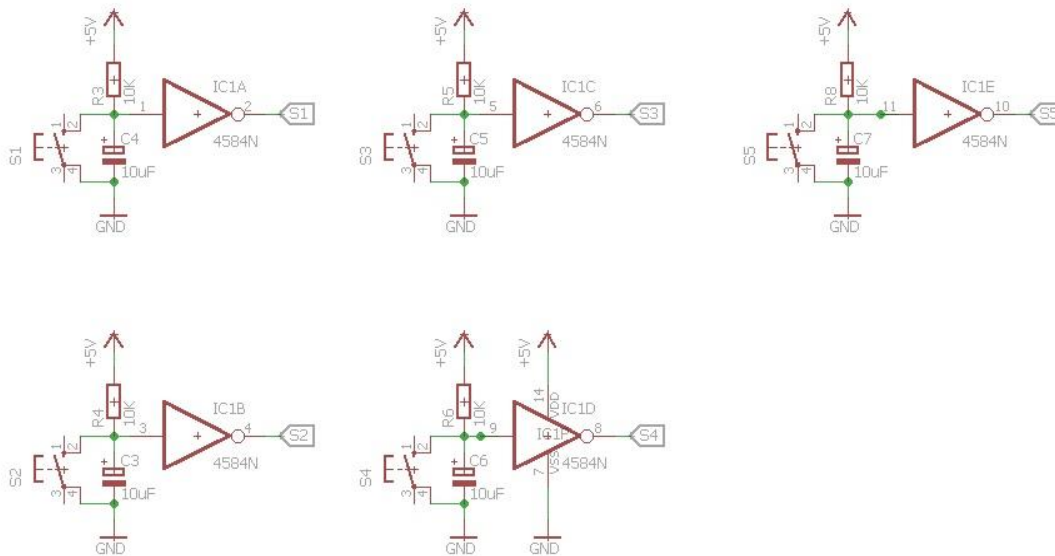


23.ábra ISP programozó csatlakozó megjelenése a kapcsolásban

3.2.3 A nyomógombok:

A pergésmentesítésre nincsenek erőforrások mivel a vezérlőt lefoglalja az érintőképernyő, a soros kommunikáció és az LCD kijelző, ezért szoftveres pergésmentesítés helyett hardveres pergésmentesítést használtam. A hardveres pergésmentesítés:

A nyomógomb egy kondenzátort tölt fel zárt állapotban és süt ki egy ellenálláson keresztül ($T=R \cdot C$). Ha a nyomógomb nem azonnal veszi fel a kellő értéket stabilan, akkor a kondenzátor nem éri el a megfelelő feszültség értéket, vagyis a nyomógombot a kiszámolt értékig kell nyomva hagyni stabilan ahhoz hogy az invertált Schmitt-trigger felkapcsolja tápfeszültségre a vezérlő bemeneti lábait.

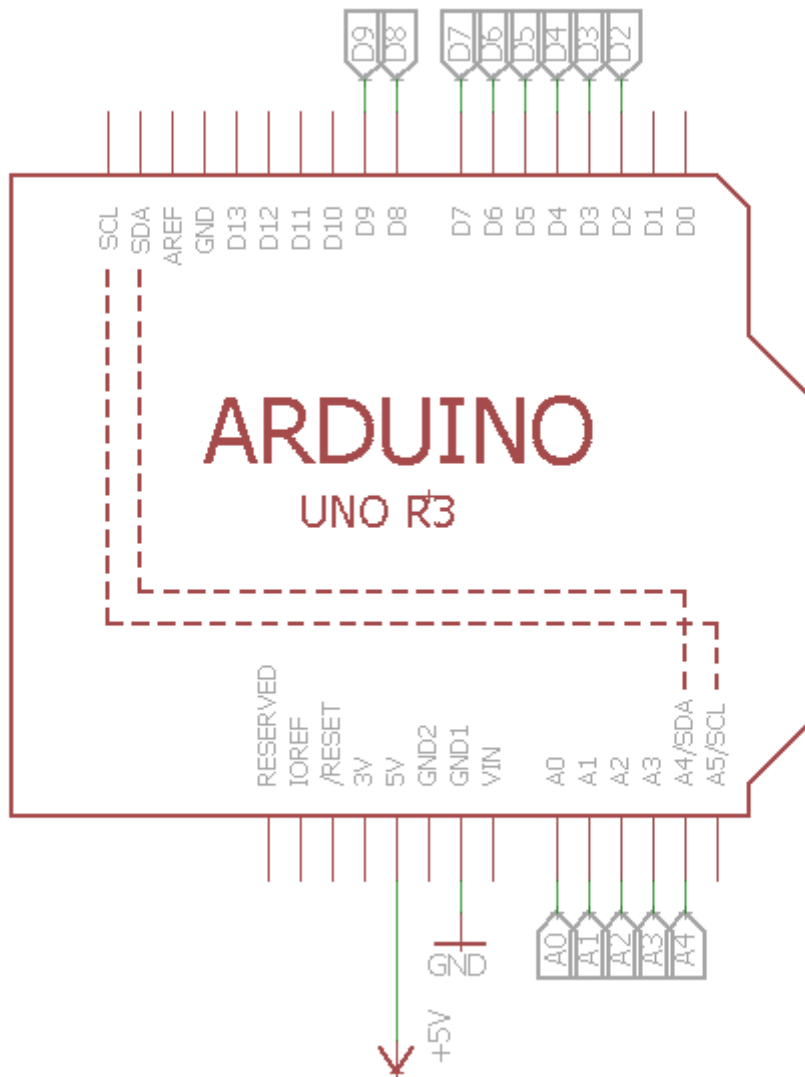


24.ábra Schmitt-triggeres nyomógomb-pergésmentesítés kapcsolási rajza

3.2.4 LCD:

Az LCD kijelző azonos lábakat használ fel az érintőképernyő lábával, ezért a programban irányváltásra van szükség az eszközök működéséhez.

Az érintőképernyő rezisztív (ellenállásalapú), feszültség értékben jelzi ki a megnyomott szegmens elhelyezkedését, ezért analóg átalakító (ADC) lábra kell helyezni a kimenetét. Mindkét eszköz 5V-ól működik. Mivel a kijelzőt Arduino Uno-ba való könnyű csatlakozásra tervezték, a tervezésnél egy ilyen eszközt tettem be, így illeszkedik pontosan az LCD lábaira az LCD-blokk.



25.ábra LCD kijelző csatlakozójának kapcsolási rajza

3.3 Tápláló-blokk

Ez a blokk tartalmazza a nagyfeszültségű részeket.

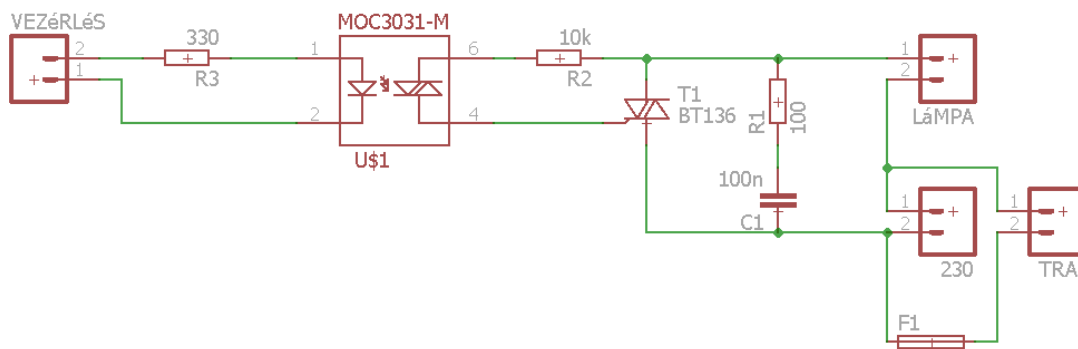
3.3.1 Lámpa:

Tirisztor és egy opto-kapu segítségével lehet egy digitális kimeneti lábbal felkapcsolni az izzót. A vezérlése az óra-blokkhoz tartozik.

3.3.2 Erre a panelre fut be a 230V, ezért itt található a transzformátor csatlakozója és az ehhez tartozó biztosíték, mely 2 amperes gyors olvadású.

A transzformátor 16 V váltakozó feszültséget állít elő, ezt a feszültséget szolgáltató vezetékek csatlakoznak az óra blokk diódás egyenirányítóhoz.

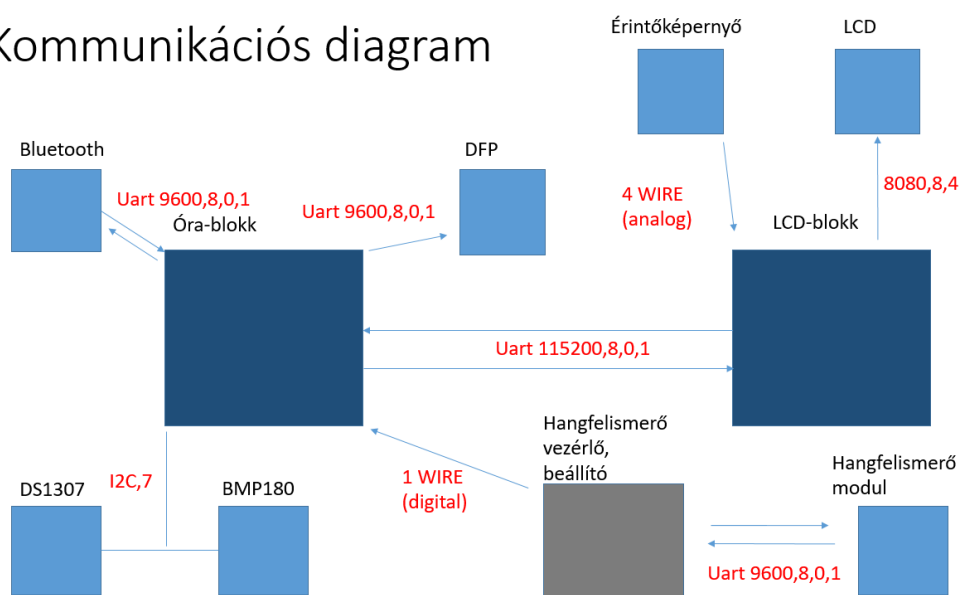
3.3.3 Az izzó ehhez a nyákhöz csatlakozik, vastag szigetelésű vezetékeken keresztül jut el a feszültség az E27 –es foglalathoz.



26.ábra Tápláló-blokk kapcsolási rajza

4. A program

Kommunikációs diagram



27.ábra Kommunikációs diagram

Az alábbi működési vázlatához és a kiválasztott funkciókhoz kell illeszkednie a programnak.

Szem előtt tartottam hogy mindkét mikrovezérlő ugyanazon fejlesztőkörnyezetben és ugyanazon programozó használatával lehessen felprogramozni, ez megkönnyítette a fejlesztést és a költségek is kisebbek voltak.

A programot még Breadboard-on kezdtem el írni, ekkor még képlékeny volt az áramkör felépítése, ekkor döntöttem el véglegesen egy erős érvel a két mikroprocesszor használatát, ez a programmemória volt. Az LCD és az érintőképernyő vezérlése ekkor 65% . Jelenleg a programmemória egészét használom 32 256 bájtból.

1.1 Az LCD vezérlő programja:

Használt könyvtárak meghívása.

Változók deklarációja.

Definiálás a könnyebb átláthatóság miatt.

A lábak irányának, alapértékeinek és a kommunikációk tulajdonságainak beállítása.

Az EEPROM-ban elmentett értékek betöltése a változóba.

A kommunikációk elindítása, emellett a már változóba betöltött értékek áttöltése az óra blokk vezérlőjébe.

Egy végtelenített ciklusban a következő feladatokat hajtja végre a vezérlő

Ha küldött az óra blokk vezérlője idő, dátum és beállítási adatokat akkor ezek betöltése a változóba és elmentése az EEPROM-ba a következő módon. Az óra blokktól két féle karakterlánc (String) érkezik:

- I. a szinkronizációs, mely tartalmazza az időt, dátumot és a jelen hőmérsékletet
- II. beállító karakterlánc, ez az óra működésének főbb paramétereit tartalmazza

A karakterlánc egyes „helyiértékein” különböző bináris és decimális változók vannak. Ezeknek a helye állandó, ez a két helyiértékű decimális számoknál melyek 0-tól 99 -ig változhatnak a kommunikáció mindkét felén 10 hozzáadásával és kivonásával oldható meg. A karakterlánc 19 karakteres és tartalmaz a megfelelő működést és helyiértékeket biztosító karaktereket (3-at) is, ha ezek a karakterek nem a megfelelő helyen vannak az információ nem helyes, ergo nem fogja a változó értékeibe írni a vezérlő.

Ha van új bejövő információ, akkor feldolgozza és megy tovább a következő műveletekre, ha nincs szintén úgy. A bejövő információk utáni művelet az érintőképernyő tengelyeinek és érintési nyomásának kiolvasása, melyhez meg kell fordítani az eddig LCD-t vezérlő lábakat bemenetre. A program keretekre van osztva (11). Az adott keretben történt műveletek határozzák meg, hogy melyik keret lesz a következő. Ezzel van megoldva a menürendszer. Az erőforrásigények csökkentése végett ha értékek beállításával foglalkozó keretet jelenít meg az LCD kijelző akkor az óra blokktól jövő információkat az LCD vezérlő figyelmen kívül hagyja. A beállítások során használhatóak a nyomógombok és az érintőképernyő is, kivétel az elhanyagolható beállítások a programmemória kapacitása miatt. A keretekben az LCD képernyőre való kirajzolás zajlik és a nyomógombok állapotának ellenőrzése. Az érintőképernyő érzékelése a nyomás alapján különül el a véletlenszerű és a határozott érintésektől. Összevonásra kerültek azok a kódrészletek amelyek egynél többször vannak meghívva.

1.2 Az óra blokk programja

Ez a vezérlő a programmemória 50% -át használja fel.

A program lefutása a következők szerint történik:

Használt könyvtárak meghívása.

Változók deklaráció.

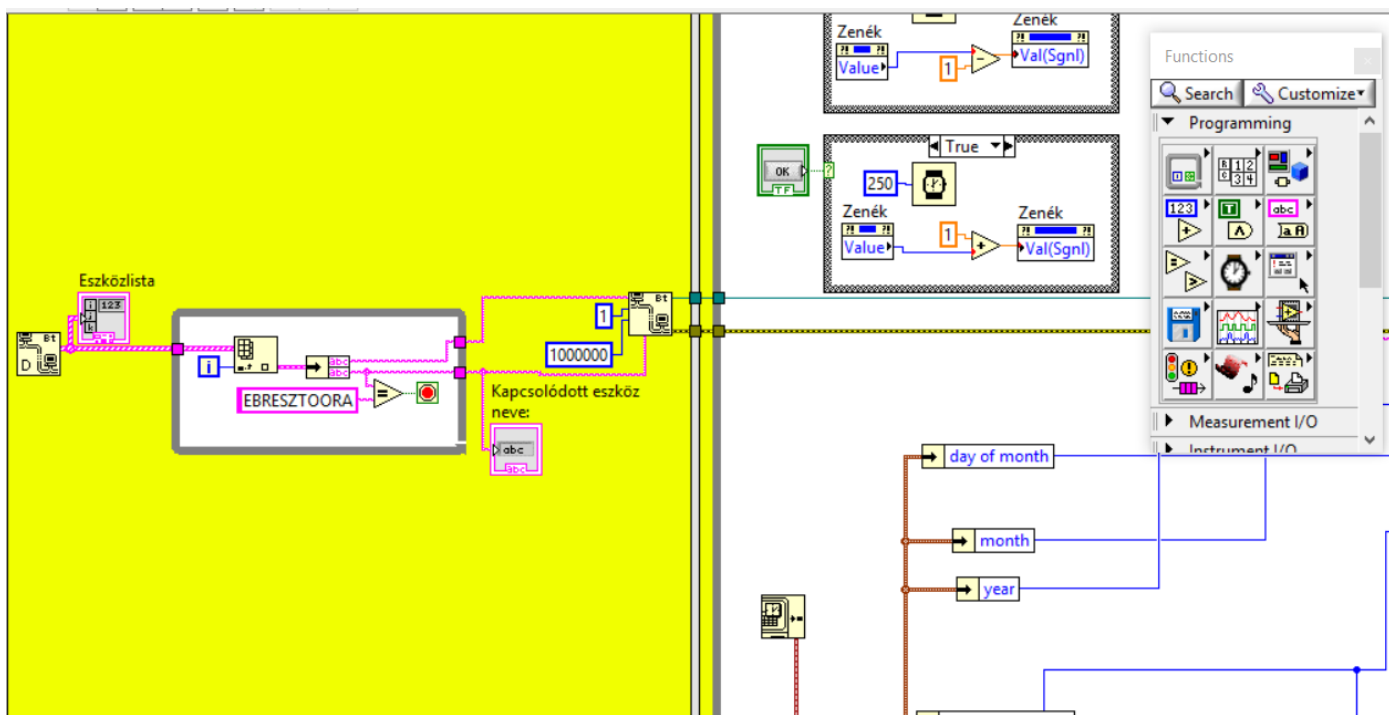
A lábak irányának, alapértékeinek és a kommunikációk tulajdonságainak beállítása.

A vezérlő lábak irányának és alapértékeinek meghatározása.

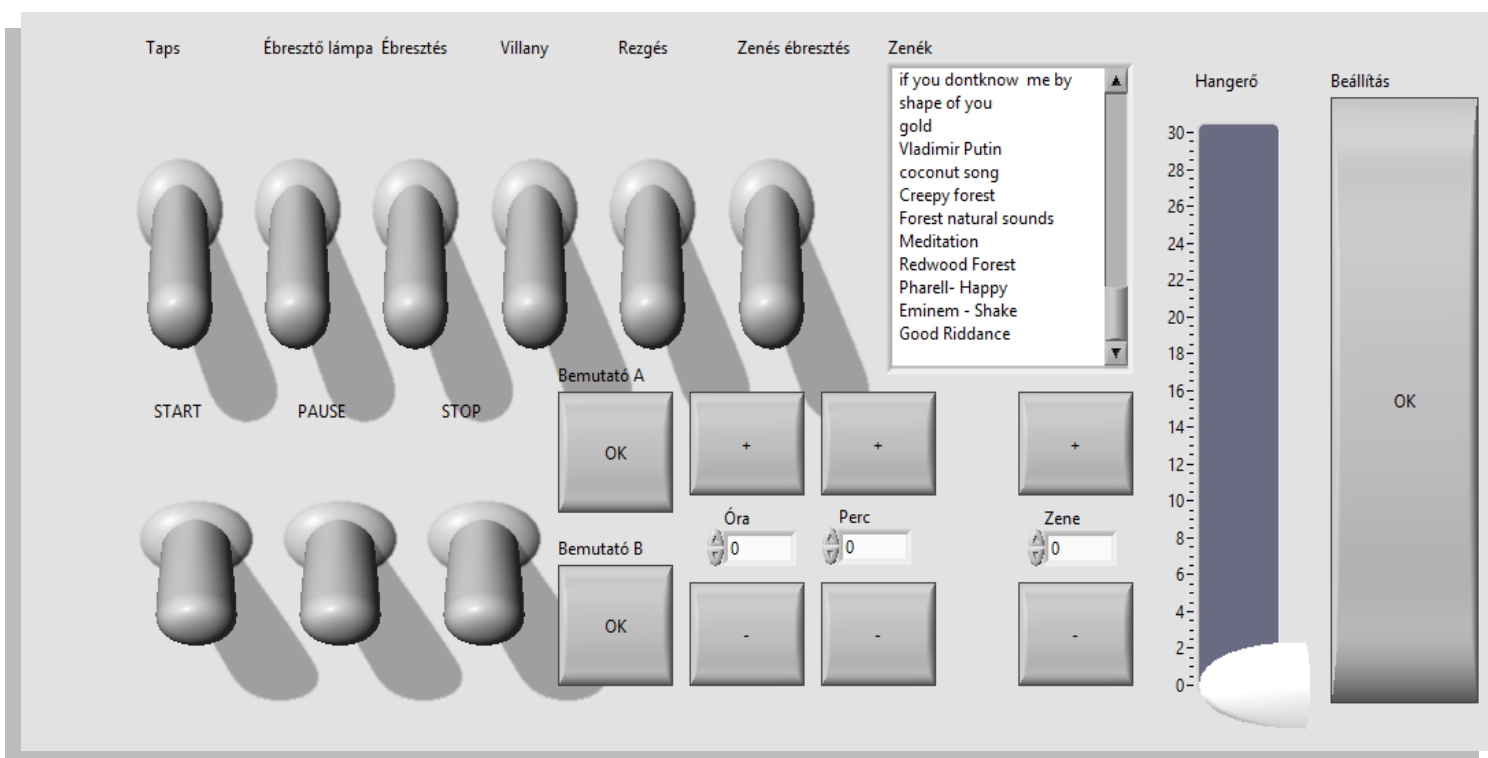
A kommunikációk elindítása. LCD-blokk, Bluetooth –modul, Barométer modul, mp3-interface modul, DS1307-es idő modul.

A végtelenített ciklus megkezdése:

A végtelenített ciklusban lefutásonként beállítja a vezérlő beállítja az idő mérésére alkalmazott IC 1Hz-es órajel-generátorának frekvenciáját. A ciklus ezenkívül a Bluetooth és az LCD –bloktól érkező információkat is kész fogadni. Az LCD-blokk vezérlőjétől 9 féle információ érkezik ezek közül kiemelném az idő beállítására szolgáló karakterláncot és az ébresztés paramétereit beállító karakterláncot. A karakterláncot ugyanazon algoritmussal bontom fel mint az LCD-blokk vezérlésében, viszont itt a karakterláncok 12 karakter hosszúak és négy karakteres típus specifikus ellenőrző adatrészletet tartalmaznak, melyek az adatok hitelességéért felelősek az zavarokkal szemben. A Bluetooth-modultól 35 karakteres 4 ellenőrzőkarakteres karakterlánc érkezik. Ezt az LCD –blokk kommunikációja után fut le a végtelenített ciklusban. A karakterek száma azért ilyen nagy mivel az összes felhasználó által beállítható paraméter értékét tartalmazza. Amint adatot kap egy eszköztől az integrál áramkör, beállítja a saját változóit és küldi tovább az LCD-blokkhoz. Így valósul meg az információcsere a két vezérlő között, változás esetén soros kommunikáción keresztül frissít a rendszer, ezért két programszállal egyszerűsödött le a programozás. A Bluetooth-modul a vezérlővel 9600 bits/s -mal kommunikál, míg számítógéppel, com –porton keresztül. A számítógépes kezelőfelületet Labview fejlesztőkörnyezetben „rajzoltam” meg. Az írt program, windows környezetben működik és android, vagy egyéb operációs rendszereken Bluetooth Soros kommunikációs program letöltése után a paraméterek manuális beírásával utasítható az ébresztőóra. (program a mellékletben: ebresztoora.vi)



28.ábra Betekintés a Labview programozási nyelvbe és a programomba



29.ábra A kezelőfelület Bluetooth eszközön

Amennyiben az előzőeken végigment a mikroprocesszor a jelenlegi időt és a beállított ébresztési időt hasonlítja össze a program. Ha egyenlőek, akkor a beállított ébresztési lehetőségekkel fog élni az áramkör.

A programok a mellékletben szerepelnek.

4.3 A program tesztelése

Az összes funkció kipróbálása a használati környezet leszimulálásával. A környezeti változókat a teszteléshez igazítva ellenőriztem a program stabil és helyes működését.

5. Doboz

5.1 Doboz készítése

A dobozt 2 mm-es alumíniumlemezből készítettem el.

A lépések a következők voltak:

- a) tervezés, számolás
- b) anyag kiválasztása és bejelölése jelölő tolómérő segítségével
- c) anyag levágása és hajlítása
- d) sorjázás tűreszelővel
- e) munka ellenőrzése
- f) éjjeli lámpa szétszerelése, dobozhoz igazítása
- g) csavar és perforációhoz szükséges lyukak megfúrása
- h) mivel ez még nem egy zárt doboz kellett oldallemezeket is vágni, ezt az előzőeknek megfelelően jelölés, vágás, hajlítás, sorjázás, ellenőrzés műveletek sorrendje szerint végeztem el



30.ábra Az alumínium doboz peroráció, ablakok és külső tapéta nélkül

5.2 Összeszerelés:

Az áramköri kártyákat, távtartókat, kontaktusokat, transzformátort, kapcsolót és lámpát a helyére csavaroztam
M3-as csavarokat és csavaranyákat használtam.



31.ábra Az összeszerelt eszköz

6. A készülék tesztelése, és észrevételek

Felhasználható funkciók megfelelő működésének ellenőrzése.

Az észrevételeim a következők:

- A feszültség stabilizátor a hűtés nélküli használatra nem megfelelő 11V feszültség-esés (Udrop) és 0,5A áramfelvétel mellett, ezért hűtésre van szükség, melyet az IC nyákból való kiszerelésével és a fémdobozhoz illesztésével javítottam ki. A csillámlemez használata szükséges, mivel a doboz a védőföldeléshez van kapcsolva biztonságvédelmi szempontok miatt.
- az SD kártya bár a doboz jelenlegi könnyen hozzáférhető pontján van, viszont kerülhetett volna közvetlen, a doboz oldallemezeinek szétszerelésével nem járó helyre is.

7. Fejlesztési lehetőségek:

- Többféle parancsszó felismerése a lámpa ki és bekapcsolásán kívül
- A 3 vezérlő egységet egy ARM Energia processzorra
- A lámpa fényerejének szabályozhatósága, és felcserélése LED világításra
- A doboz kompaktabbá tétele
- Az eszköz narrációs funkciói terjedjenek ki az idő, időjárás, beállított események felolvasásáig
- Naptár integrálása
- A rezgő ébresztési lehetőség kifejlesztése

8. Költségek:

Alkatrészek	Ár [Ft]
DS1307	300
32,768KHz	40
8*10KΩ	24
2*Hüvelysor	200
2*Tűsor	200
TIP122	85
4*100KΩ	8
4,7KΩ	5
2*2200μF	100
22μF	8
3*100nF	18
2*1N4004	7
220μF	34
2*2.2μF	16
5*1KΩ	6
4*1N5408	60
2*10μF	12
100μF	6
2*7805	140
7812	70
2N3906	20
4*22pF	32

2*16MHz	64
2*ATMEGA328	1040
BC337	8
1N4007	9
7408	175
DFP modul	680
LCD	1200
6*nyomógomb	138
CD40106	85
motor (rezgés)	280
BMP180	1400
gomb elem	210
MOC3021	120
SN2030	90
TDA2030A	160
BT136	110
biztosíték	190
biztosíték foglalat	80
220Ω	1
Hangfelismerő modul	6000
Lámpa	1500
	Összesen: 14931

9. Irodalomjegyzék:

<https://www.arduino.cc/en/Tutorial/ArduinoToBreadboard>

<https://forum.arduino.cc/index.php?topic=366304.0>

<https://www.olimex.com/Products/Components/RF/BLUETOOTH-SERIAL-HC-06/resources/hc06.pdf>
https://www.openhacks.com/uploads/productos/dfplayer_mini_-_robot_wiki.pdf
http://pdf.datasheetcatalog.com/datasheet_pdf/contek-microelectronics/TDA2030A.pdf
<http://www.eng.utah.edu/~cs5780/debouncing.pdf>
<https://soldnerd.com/2014/11/12/switch-debouncing-using-74hc14/>
<https://learn.sparkfun.com/tutorials/bmp180-barometric-pressure-sensor-hookup->
<http://bildr.org/2011/03/high-power-control-with-arduino-and-tip120/>
<http://electronics.stackexchange.com/questions/58374/ds1307-breakout-board>
<https://datasheets.maximintegrated.com/en/ds/DS1307.pdf>
<http://dlnware.com/i2c>
<https://www.arduino.cc/en/Reference/SoftwareSerial>
<http://www.edaboard.com/thread265546.html>
<http://www.avrfreaks.net/forum/7805-so-hot>
<https://www.sparkfun.com/datasheets/Components/LM7805.pdf>
<http://www.instructables.com/id/AT-command-mode-of-HC-05-Bluetooth-module/>
http://www.geeetech.com/wiki/index.php/Arduino_Voice_Recognition_Module
http://www.geeetech.com/wiki/images/6/69/Voice_Recognize_manual.pdf