

DIPLOMATERVEZÉSI FELADAT

Schuszter André

Villamosmérnök hallgató részére

Multifunkciós egyfázisú inverter tervezése

Napjainkban a félvezető technika és az akkumulátorok térhódításának köszönhetően egyre inkább előtérbe kerülnek a különböző energiaforrásokról üzemeltetett egyfázisú inverterek. Az invertert tápláló energiaforrás a legtöbb esetben vagy akkumulátor vagy napelem, míg az inverter felhasználási területét tekintve lehet hálózatra kapcsolt vagy szigetüzemi működésű. Előbbiek általában teljesítmény szabályozott üzemben a DC forrásból származó teljesítményt táplálják vissza a hálózatba, utóbbiak a különböző kereskedelmi forgalomban kapható eszközök számára biztosítanak szabályozott feszültséget és frekvenciát. A diplomamunka során egy DC forrásról üzemeltett galvanikusan leválasztott invertert kell megvalósítani, amely hálózatra kapcsolt üzemben a DC forrás által szolgáltatott teljesítményt táplálja a hálózatba, míg szigetüzemben a rendelkezésre álló egyenfeszültségből változtatható frekvenciás és változtatható amplitúdójú feszültséget állít elő.

A bevezetőben leírt specifikációnak megfelelően a hallgató feladata egy, a fenti kritériumoknak megfelelő inverter tervezése a következőknek megfelelően:

- Az áramkör műszaki paraméterei a következők:

Bemeneti feszültségtartomány:	12-25 V _{dc}
Kimeneti feszültségtartomány szigetüzemben:	0-230 V _{ac}
Kimeneti névleges teljesítmény:	150 W
- Ismerje meg a galvanikus csatolású feszültségnövelő kapcsolás működését!
- Ismerje meg az egyfázisú hídkapcsolású inverter működését!
- Méretezze a kapcsolás főáramköri és vezérlőáramköri elemeit!
- Készítse el a konverterek beágyazott vezérlő szoftverét!
- Készítsen HMI felületet a jellemző paraméterek megjelenítésére és beállítására!
- Élessze és értékelje az elkészített áramkör működését!

Tanszéki konzulens: Dr. Balogh Attila

Budapest, 2017. március 6.

Dr. Charaf Hassan
egyetemi tanár
tanszékvezető



Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Automatizálási és Alkalmazott Informatikai Tanszék

Schuszter André

Multifunkciós egyfázisú inverter tervezése

Konzulens: **Dr. Balogh Attila**

Budapest, 2017

Tartalomjegyzék

Összefoglaló	
Abstract.....	
1	Napelem
2	Váltóirányítók(inverterek)
2	Tervezési peremfeltételek
3	Hardver elemek kiválasztása
3.1	Előtöltés
3.2	Boost konverter
3.3	Mikrokontroller
3.4	Felhasználói felület
3.5	Mérések
3.6	Szűrő tervezése
3.7	Transzformátor
3.8	Sznuubber kondenzátor
3.9	Félvezetők
3.10	Nyomtatott Huzalozású Lemez
3.11	Inverterek
4	Beágyazott Szoftver
4.1	Inicializálás
4.2	Boost konverter vezérlése
4.3	LCD kijelző kezelése
4.4	ADC kezelése
5.4	DMA használata

HALLGATÓI NYILATKOZAT

Alulírott, Schuszter André, szigorló hallgató kijelentem, hogy ezt a szakdolgozatot 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, 2017. 11. 12.

Schuszter André

Összefoglaló

Napjainkban a hagyományos energiakészletek csökkenésével, és a környezettudatos szemlélet elterjedésével, egyre nagyobb igény mutatkozik az alternatív energiaforrások felhasználására. Az elkövetkező években különösen nagy növekedés várható, a napelemek által megtermelt energia mennyiségéből.

Mivel a napelemek bizonyos időszakokban több energiát termelnek, mint amennyire a felhasználónak szüksége lenne, így a „felesleges” energiát visszatáplálhatja a hálózatba. A diplomatervem témája a napelem által előállított DC feszültségből AC feszültséggé váltó áramkör tervezése

Az első részben bemutatom a tervezés hardveres peremfeltételeit. Ismertetem az alkatrészek kiválasztásának szempontjait, a kapcsolási rajz egyes részleteire is kitérek, valamint a huzalozást mutatom be.

A második részben a vezérlést megvalósító szoftverről írok. Bemutatom az inverterek, feszültségnövelő áramkör vezérlését, a ki- és bemenetek kezelését, a feszültség és árammérés megvalósításának módját, valamint a felhasználói felület kezelését.

A legvégén összegzésként az elkészült áramkört értékelem, tesztelem.

Abstract

In our days with the decrease of primary stocks and environmentally friendly lifestyle, there is a need to use secondary energy sources. In the foreseeable future a big growth is expected in the number of solar cells, in the amount of energy, which can be produced by the solar cells.

Sometimes the solar cells produce more energy than the user need, in this case the user will be an energy supplier for the energy network. In my thesis I will plan an iverter, which can convert the direct current produced by the solar cell into alternate current, which has the same amplitude and frequency as int he energy network is available.

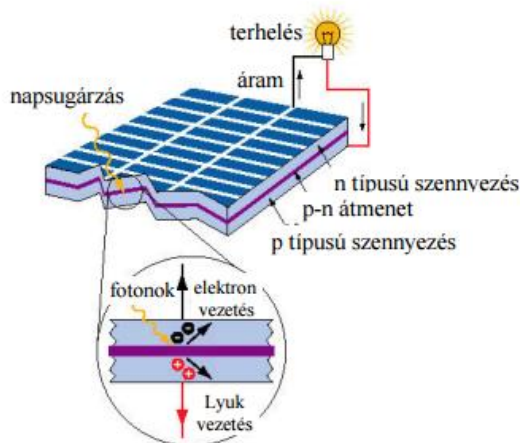
Int he first part of my thesis I show how a DC/DC converter and inverter works; what were the boundary conditions to choose the hardware parts; I present detailed the most important parts of the schematic; I demonstrate how the printed circuit board was planned.

Int he second part I present the software, which I wrote to control the inverters and the boost converte. Besides this, I present the user interface, and what kind of parameters can the user adjust, how the inputs and outputs were set, how current and voltage is measured.

At the end as a conclusion i will test the completed device.

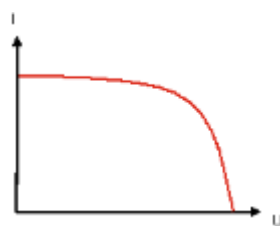
Napelem

A napelem egy szilárdtest eszköz, amely az elektromágneses sugárzást elektromos energiává alakítja. Az energiaátalakítás során a sugárzás elnyelésével mozgásképes töltött részecskék keletkeznek, amelyet a beépített elektromos tér rendezett mozgásra kényszerít, áram fog folyni (1. ábra).



1. ábra: napelem sematikus vázlata [1]

Alapanyag szerint megkülönböztetjük a napelemeket. Az esetek többségében mono- illetve polikristályos napelemek készülnek, esetenként gallium-arzenid vegyületen alapuló napelemekkel is találkozhatunk. A napelemekből kivehető teljesítmény egyaránt függ a beesési szögtől, és a megvilágítás intenzitásától.



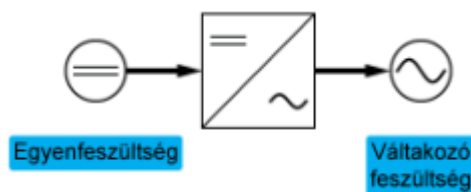
1. ábra: napelem feszültség-áram karakterisztikája

Az 1. ábrán egy napelem jellegzetes feszültség-áram grafikonja látható. A kivehető legnagyobb teljesítmény (= görbe alá berajzolható legnagyobb területű grafikon) Maximal Power Point-ban (MPP) érhető el.

A napelemek széles körben való elterjedését eddig hátráltatta a magas áruk, amely a drága előállítási költségükből adódik. A csökkenő fosszilis energiakészletekkel, és a technológia fejlődésével, egyre nagyobb igény várható a napelemek felhasználása iránt.

Váltóirányítók (inverterek)

A váltóirányítók, vagy más néven inverterek egyenfeszültséget vagy áramot alakítanak váltakozó feszültséggé vagy árammá. Nagyon fontos teljesítményelektronikai áramkörök egy sor fontos alkalmazással. Jól tudott, hogy a fogyasztói villamos hálózat, legyen egy vagy háromfázisos, a szállított feszültség effektív értékével és a feszültség állandó frekvenciájával jellemezhető. Másrésről egy sor gyakorlati alkalmazás (pl. aszinkron gépek fordulatszám szabályozása, speciális szervomotor vezérlések, indukciós hevítések, villamos vontatások, stb.) megkívánják a betápláló feszültség frekvenciájának és effektív értékének a változtatását és bizonyos esetekben a standard 1, 3 fázistól való eltérést is. Ennek a feladatnak az elektromechanikus megvalósítása igen költséges és már elavult a gyenge hatásfok és az állandó karbantartás szükségessége miatt. A teljesítményelektronikai megoldások mára már teljesen egyeduralkodóvá váltak az alkalmazások többségében, aminek köszönhetően a szakirodalom is rendkívül gazdag és széles körű ezen a speciális területen. Az egyenáram váltakozó árammá való átalakítása többféleképpen megvalósítható. Egy inverter képes terhelés oldali külső természetes kommutációval működni ha a terhelés mondjuk szinkron motor. Ezek az üzemmódok azonban elég speciálisak és az átalakítás csak akkor lehetséges, ha a kommutációs energia forrása a bemeneti váltakozó áramú hálózat vagy a terhelő kör szinkron motor rendelkezésre áll. Én az impulzus szélesség modulált váltóirányítót fogok tervezni, ahol kitöltési tényező változtatásával valósítom meg a váltakozó feszültséget. [1.1]



.ábra: Inverter jelölése

Tervezési peremfeltételek

Az eszköznek képesnek kell lennie működni szigetüzemben, és hálózatra is vissza kell tudni táplálni. Egyenáramot visszatáplálni nem megengedett, így a napelemek által előállított egyenáramhoz kellett egy átalakítót, invertert, terveznem, aminek kimenetén már váltakozó áramú feszültség jelenik meg. A tervezés előtt utána kellett néznie a piacon kapható napelemeknek jellegzetes paramétereinek. Kiválasztottam egy 140W névleges teljesítményűt [2]. A napelem rövidzárási árama 8.85 A szakadási feszültsége 21.6V. Továbbá szerettem volna, ha egy 12V-os ólomakumulátorról is demonstrálható lenne működése. Ahhoz, hogy az átalakító hangja ne legyen zavaró a kapcsolási frekvenciával a hallható hang tartománya (20-20000 Hz) fölé kellett mennem. Egy kapcsoló segítségével választható a szigetüzemű, vagy hálózatra csatlakozó üzemmód. Szigetüzemben a frekvencia és a kimenő váltakozó feszültség nagysága változtatható egy potenciométerrel változtatható. A kimenő paraméterek aktuális értékei egy LCD kijelzőről olvashatók le.

Némi túlméretezéssel adódtak a tervezési peremfeltételek:

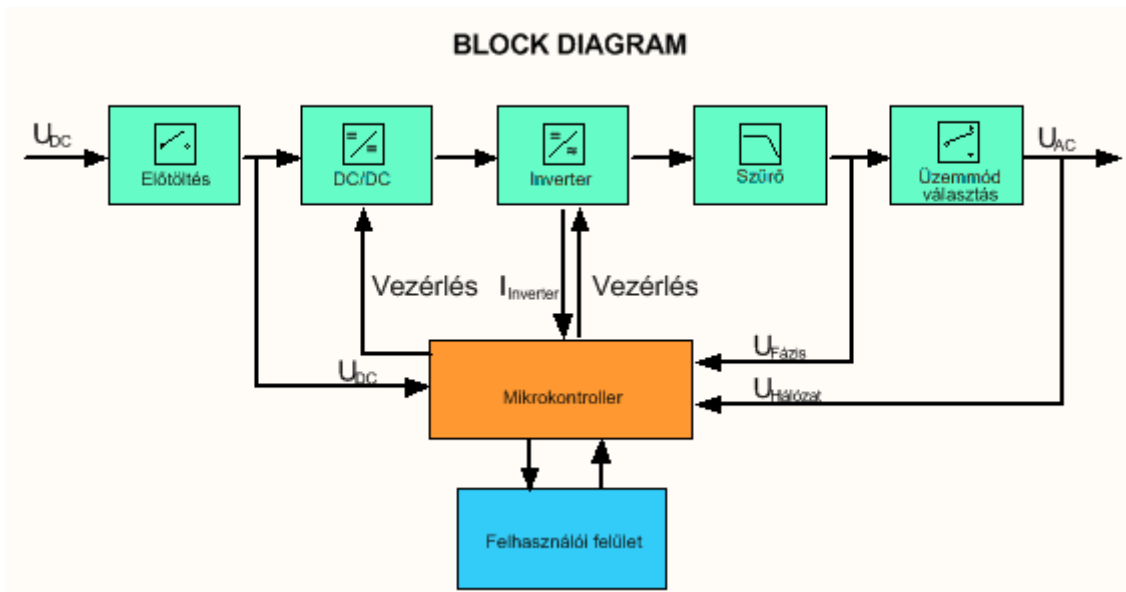
Bemenő egyenfeszültség: 12-25V

Kimenő váltakozó-feszültség: 0-250V

Névleges teljesítmény: 150W

Kapcsolási frekvencia: 25 kHz

Hardver elemek kiválasztása



.ábra: A kapcsolás blokkdiagrammja

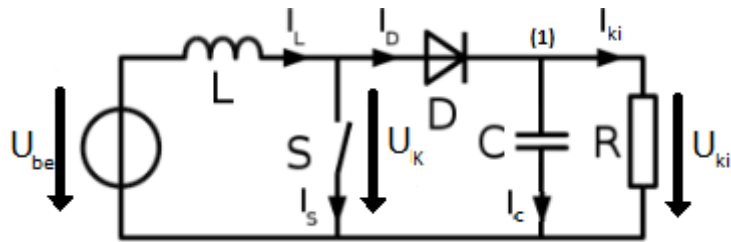
A .ábrán látható a kapcsolás blokkdiagrammja. A bejövő egyenfeszültség először egy előtöltőn halad át, ami megakadályozza, hogy a kondenzátorokra hirtelen nagy feszültség jusson, és ezáltal nagy áramok induljanak meg. Ezután következik egy DC/DC átalakító, ami a bejövő feszültséget egy adott szintre (25V) növeli, amit az inverter váltófeszültséggé alakít át. Az inverter kimenetére egy szűrő csatlakozik, amely a kapcsolófrekvenciás zajokat szűri, illetve a jelet szinusszá átlagolja. Az átalakító vagy szigetüzemben működik, vagy hálózatra táplál vissza. A mikrokontroller vezérli a DC/DC átalakítót, illetve az invertert, valamint meghatározott pontokban méri a feszültséget, áramot. Ezen mért értékek kijelzése a felhasználói felületen történnek, illetve a felhasználó beállításait is a mikrokontroller kezeli.

Előtöltés:

Az előtöltésért egy NTC ellenállás a felelős. Az NTC egy negatív hőmérséklet együtthatójú ellenállás, ami azt jelenti, hogy a szokványos ellenállásokkal ellentétben az NTC ellenállásának értéke csökken a hőmérséklet növekedésével. Bekapcsoláskor áramkorlátozó szerepe van, tehát nem tudnak nagy áramok megindulni a bekapcsolásból adódó feszültségkülönbség miatt. A bekapcsolás után a rajta disszipálódó teljesítmény hő formájában jelenik meg, emiatt ellenállása, így áramkorlátozó szerepe is csökken. Az NTC-vel párhuzamosan kötve található egy relé is amely, az előtöltés után meghúz ezáltal rövidre zárja az NTC-t, nem disszipálódik rajta több hő.

Boost konverter:

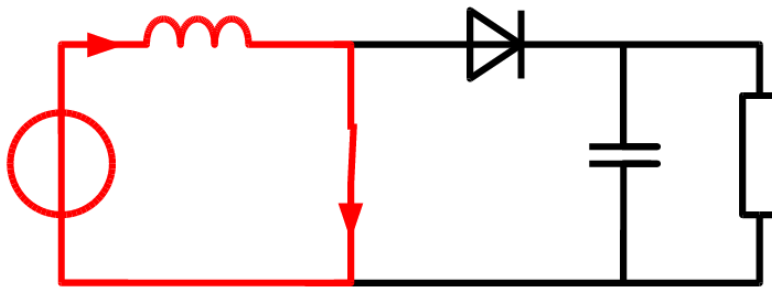
A bemenetre jutó feszültségszint tartománya 12-25V, amit egységesen 25V-ra akarok felboostolni, így az inverter vezérlése egyszerűbb lesz. A boost konverter a kapcsolóüzemű tápegységek csoportjába tartozik, legalább két félvezetőt tartalmaz, és egy energiatároló elemet (tekercs, kondenzátor). A feszültség hullámosság csökkentése érdekében a kimenetre egy szűrő kondenzátort helyezünk.



.ábra: Boost konverter felépítése

Működése:

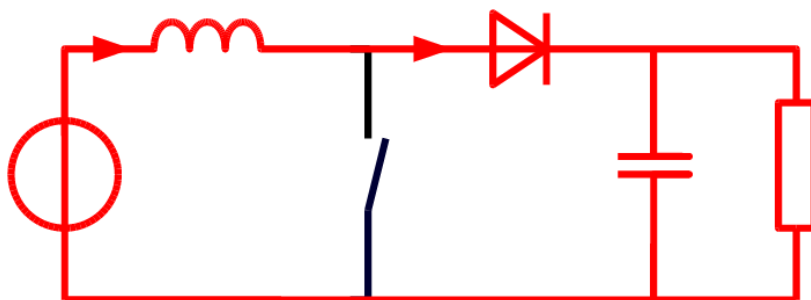
A kapcsoló zárt (.ábra) állásában a rövidzár miatt az áram az óramutató járásával megegyező irányban folyik keresztül a tekercsen, amiben az energia tárolódik. A tekercsen keresztül folyó pozitív áram hatására a tekercs feszültsége növekszik.



.ábra: Áram iránya a kapcsoló zárt állásában

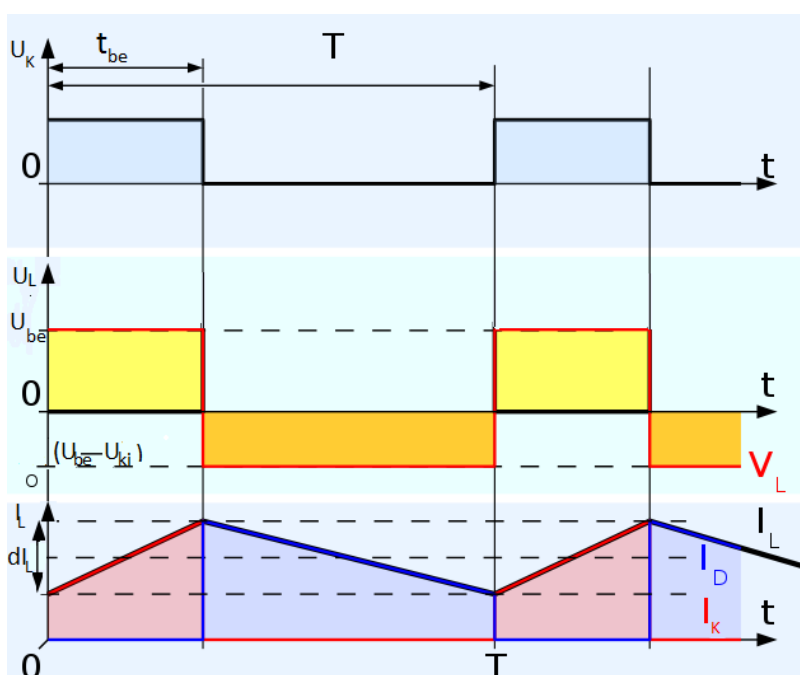
Amikor az impulzusszélesség vezérelt kapcsoló kinyit a magas kimeneti feszültség hatására megváltozik a tekercs polaritása (negatív feszültség), és az árama csökkenni kezd.

Amennyiben elég gyors a kapcsolás a tekercs árama nem csökken nullára, folyamatos vezetésben marad. Abban az időszakban, amikor a kapcsoló zárt állásban van, a kimeneti feszültséget a kondenzátor tartja.



.ábra: Áram iránya a kapcsoló zárt állásában

A boost konverterben folyamatos vezetés esetén fellépő egy periódus alatti feszültségek, áramok grafikonja az alábbi ábrán látható.



.ábra: Az induktivitás feszültsége, árama

Az ábrán a be- és kikapcsolás alatti tekercsfeszültség-idő szorzat megegyezik egymással (a grafikonon a narancs- és citromsárga színű terület).

$$U_{be} * t_{be} = (U_{ki} - U_{be}) * (T - t_{be})$$

$$U_{ki} = U_{be} * \frac{T}{T - t_{be}} = U_{be} * \frac{1}{1 - d}$$

A feszültségnövelő áramkör (.ábra) (1) pontjára a csomóponti törvényt felírva, megkapjuk, hogy a dióda áramának átlaga megegyezik a kimeneten mérhető árammal:

$$i_D = i_c + i_{ki}$$

Az áramok átlagát nézve is fennáll az egyenlőség

$$i_{DAV} = i_{cAV} + i_{kiAV}$$

A kondenzátor áramának középértéke 0, hiszen feszültsége nem „mászik el”, így:

$$i_{DAV} = i_{kiAV}$$

Mivel a kimenet feszültsége állandó, így árama is az ($i_{ki} = \frac{U_{ki}}{R}$), tehát pillanatértéke megegyezik átlagértékével, ebből tehát az következik, hogy:

$$i_{DAV} = i_{ki}$$

A szükséges induktivitás a következőképpen határozható meg:

$$\Delta I_L = \frac{U_{be}}{L} * t_{be} = \frac{U_{be} - U_{ds}}{L} * (T - t_{be})$$

$$L \geq \frac{U_{be}}{\Delta I_L} * t_{be}$$

A félvezetők méretezéséhez szükséges adatok:

Maximális feszültségük:

$$U_{K,csúcs} = U_{D,csúcs} = U_{Ki,csúcs}$$

Maximális áramuk:

$$i_{K,csúcs} = i_{D,csúcs} = i_{LAV} + \frac{\Delta i_L}{2} = \frac{1}{1-d} i_{ki} + \frac{U_{be} * t_{be}}{2L}$$

A szükséges kapacitás pedig:

$$i_c = C * \frac{dU_c}{dt} = C * \frac{\Delta U_c}{\Delta t} = C * \frac{\Delta U_{ki}}{\Delta t}$$

A kapcsoló zárt állapotában a kondenzátor árama megegyezik a kimenő árammal, hiszen a diódán nem folyik áram.

$$\Delta U_{ki} = \frac{i_c * \Delta t}{C} = \frac{i_{ki} * t_{be}}{C}$$

Tehát a kapacitás:

$$C \geq \frac{i_{ki} * t_{be}}{\Delta U_{ki}}$$

A fenti képletek alapján a konverter méretezése:

$$f_{sw} = 25 \text{ kHz}$$

$$U_{ki} = 25 \text{ V}$$

$$U_{be,min} = 12 \text{ V}$$

$$\Delta U_{ki,min} = U_{ki} * 5\% = 12V * 0.05 = 0.6 \text{ V}$$

$$U_{ki} = U_{be} * \frac{T}{T - t_{be}} = U_{be} * \frac{1}{1 - d}$$

$$d_{max} = 1 - \frac{U_{be,min}}{U_{ki}} = 1 - \frac{12}{25} = 0,52$$

$$t_{be,max} = d_{max} * \frac{1}{f_{SW}} = 0,52 * \frac{1}{25000Hz} = 20.8 \mu s$$

$$i_{ki} = \frac{P}{U_{be}} = \frac{150W}{25V} = 6 A$$

$$i_{LAV} = \frac{1}{1 - d} * i_{ki} = \frac{1}{0.48} * 6A = 12.5 A$$

$$\Delta i_L = 25\% * i_{LAV} = 0.25 * 12.5A = 3.125 A$$

A kondenzátor:

$$C \geq \frac{i_{ki} * t_{be}}{\Delta U_{ki,min}} = \frac{6 A * 20.8 \mu s}{0.6 V} = 208 \mu F$$

Az induktivitás:

$$L \geq \frac{U_{be}}{\Delta i_L} * t_{be,max} = \frac{12 V}{3.125 A} * 20.8 \mu s = 79.8 \mu H$$

Mivel nem találtam olyan induktivitást, amely legalább 80μH-s és a szükséges áramot szállítani tudná, ezért nekem kellett egyet méreteznem. Először meg kellett határoznom a szükséges minimális vaskeresztmetszet, ablakkeresztmetszet szorzatát, hogy ki tudjak egy vasmagot választani:

$$I_{LRMS} = \sqrt{I_{LAV}^2 + \left(\frac{\Delta I_L}{2 * \sqrt{3}}\right)^2} = \sqrt{12.5 A^2 + \left(\frac{3.125A}{2 * \sqrt{3}}\right)^2} = 12.53A$$

$$K_i = \frac{I_{Lmax}}{I_{LRMS}} = \frac{I_{LAV} + \frac{\Delta I_L}{2}}{I_{LRMS}} = \frac{12.5 + \frac{3.125}{2}}{12.53A} = 1.12$$

$$(A_V A_A)_{min} = \frac{L * I_{Lmax} * I_{LRMS}}{K_t * j_{max} * B_{max}} = \frac{80\mu H * 14.06A * 12.53A}{0.75 * 4 * 10^6 \frac{A}{mm^2} * 0.3T} = 1.566 cm^4$$

A_V: vaskeresztmetszet [cm⁴]

A_A: ablakkeresztmetszet [cm⁴]

A tanszéken rendelkezésre állt egy pormag (Micrometals T141-40), amely a következő paraméterekkel rendelkezett:

$$A_V A_A = 2.66 cm^4$$

$$A_L = 60 nH/N^2$$

$$N = \sqrt{\frac{L}{A_L}} = \sqrt{\frac{80\mu H}{60 \frac{nH}{N^2}}} = 36.51 \rightarrow 37 \text{ menet}$$

A_L : menetenkénti induktivitás [nH]

N: menetszám

A vezeték-keresztmetszet a következő képlet szerint számítható:

$$q = \frac{I_{RMS}}{j_{max}} = \frac{12.53A}{4 \frac{A}{mm^2}} = 3.13mm^2$$

q: A szükséges vezeték-keresztmetszet

Mikrokontroller:

A mikrokontroller kiválasztásánál meghatározó szempont volt, hogy már rendelkeztem némi tapasztalattal az STMicroelectronics mikrokontrollerek körében, így a gyártó termékei közül választottam a feladatra legalkalmasabbat. Mivel a kontroller beforrasztása kézzel történt, így a legkisebb, de még elegendő lábszámú eszköz kikeresése volt a cél. A szükséges kivezetések összeszámlálása után tovább szűkült szóba jöhető eszközök köre a 48 lábszámú mikrovezérlőkre. Továbbá elvárás volt, hogy rendelkezzen legalább 4 PWM csatornával és 9 analóg mérés fogadására alkalmas pinnel. A feladatnál nem volt fontos szempont az ultra alacsony energiafogyasztás, de a nagyteljesítményű mikrokontrollerek sem voltak szükségesek, így a 32 bites kategóriában az F0, F1, F3-as család bizonyult megfelelő választásnak. A keresési feltételek beállítása után olyan vezérlőt választottam ami rendelkezik lebegőpontos egységgel, az STM32F303CBT7 mikrovezérlőt.

A vezérlő főbb adatai:

- ARM 32-bit Cortex-M4 CPU
- Maximum 72 Mhz-es órajelfrekvencia
- 4-32 Mhz-es kristály használható
- 40 KHz-es belső oszcillátor
- Kommunikációs interfészek (CAN, I2C, SPI, USART, USB)
- 2-3.6V tápellátás
- 0-3.6V konverziós tartomány
- ADC felbontás: 12bit
- Program memória: 128 KB

- RAM memória: 40 KB
- Csomagolás: LQFP
- Timerek száma: 9x16 bit, 1x32 bit
- ADC csatornák száma: 15 csatorna
- I/O csatornák száma: 37 I/O

Felhasználói felület:

A felhasználó számára ki kell jelezni az áramkör egyes pontjain mérhető feszültséget, áramerősséget, illetve a felhasználó által beállított értékeket. Ehhez 2X16-os alfanumerikus kijelző alkalmatlan lenne, ezért egy nagyobb kijelzővel rendelkező, több információt megjeleníteni képes grafikus kijelzőt (128x64 DOT LIGHT:LED 64128QCCBW-3LP) választottam.



.ábra: A grafikus LCD

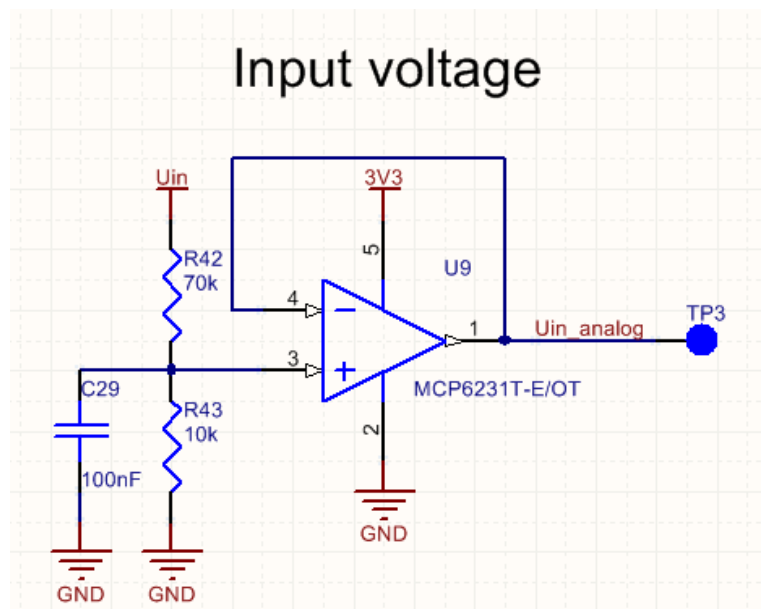
A felhasználó által beállítható a feszültség és a frekvencia szigetüzemben. Ez egy-egy potenciométerrel valósítottam meg, amelynek analóg kimenőjelét úgyszintén a mikrovezérlővel dolgozom fel.

Mérések:

A feszültség mérése a mikrovezérlő analóg bemenetével történik. Mivel a mikrokontrollerünk 3.3V-ról működik, így a magasabb feszültségeket át kell konvertálnunk a 0-3.3V-os tartományba. A legcélravezetőbb megoldás erre, egy ellenállásosztó használata, amelyben a nagyobb pontosság érdekében 1%-os tűrésű ellenállásokat alkalmaztam, és egy követőerősítőt tettem utánuk, hogy a további fokozatok ellenállása ne befolyásolja a

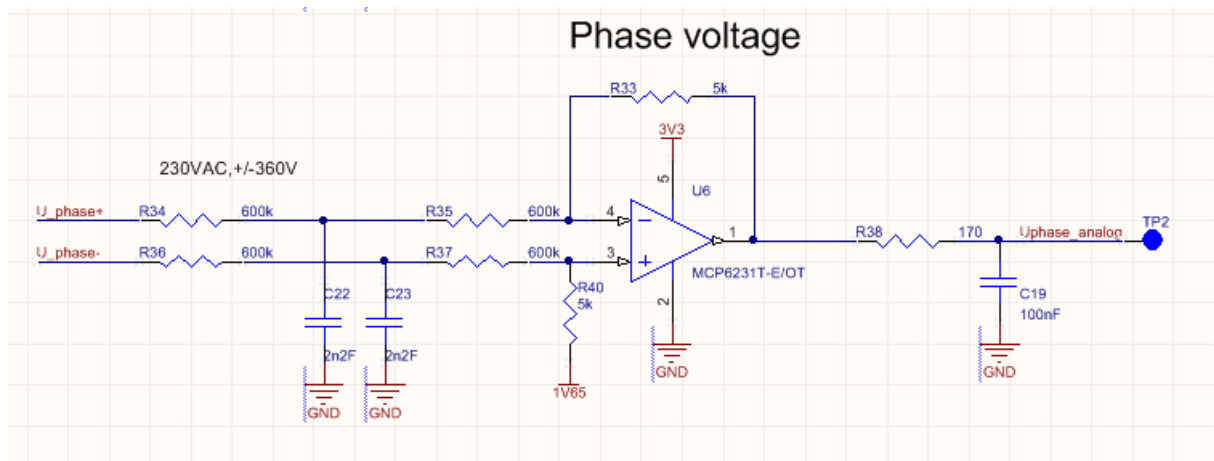
feszültség osztását. Továbbá egy 100nF-os kondenzátort is tettem az alsó ellenálláshoz a zavarfeszültség szűrésére. Ezzel a módszerrel csak egyenfeszültséget tudok mérni, mint például a bemenő feszültséget, amely 12-25V lehet, így az osztás ebben az esetben:

$$U_{in_analog} = U_{in} * \frac{R_{42}}{R_{42} + R_{43}} = 25 * \frac{10k}{70k + 10k} = 3.125V$$



.ábra: Bemenő feszültség konvertálása a 3.3V-os tartományba

A váltakozó feszültség a fél periódusa alatt negatív feszültségű, így ezt nem lehet egy egyszerű ellenállásosztóval a 0-3.3V- os tartományba konvertálni. Itt differenciálisan kell mérni és a váltakozó feszültség referenciaszintjét a mérési tartomány közepére (1.65V) helyezni. Az 1.65V-ot a tápfeszültség felére történő leosztásával értem el, és ezzel toltam el a műveleti erősítő referenciaszintjét. Az erősítő után elhelyeztem egy aluláteresztő szűrőt a nagyfrekvenciás zajok szűrésére. A fázisfeszültség mérésénél a 230V-os effektív értékű feszültség csúcsértéke 325V. A hálózati feszültség-ingadozást is rászámolva a +/-360V-ot kellett 0-3.3V tartományba átkonvertálnom. A 720V-os tartomány erősítés után, $720 * \frac{5k}{600k+600k} = 3V$, tehát a referenciaszint 1.65V-os eltolásával a fázisfeszültség a 0.15-3.15V-os tartományba konvertálódott.

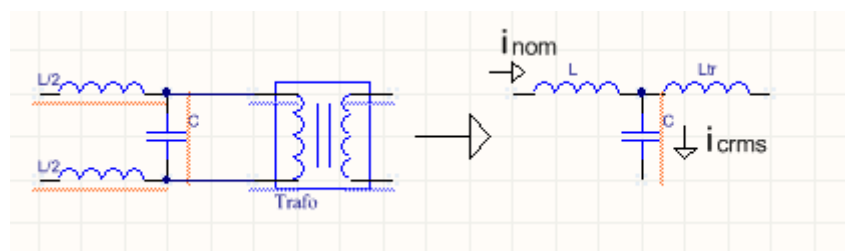


.ábra: differenciális feszültségmérés

Szükségem van még az áram mérésére. Egyrészt a Boost konverter fojtójának áramára van szükségem a szabályozáshoz, amely mindig pozitív, így ezt egy kis ellenálláson átfolyatva, az ellenállás feszültségének méréséből meg tudom határozni a rajta keresztül folyó áramot. Mivel ennek a söntnek nagy a közös módusú erősítése, így feszültségét sönterősítővel mérem. Az inverter áramának mérésénél nem használhatom ezt a mérést, hiszen ott kétirányban is folyik áram, így oda egy árammérő IC-re van szükségem ami a +/- 15 A-es tartományban alkalmas a mérésre. Ide az Allegro Microsystems terméke az ACS709LLFTR-20BB-T jelentett megoldást, ami az áramértékkal arányos feszültséget ad ki egy pinjén, így ezt az analóg feszültségértéket kell szoftveresen áramértékké átszámolnom.

Szűrő:

Az inverter kimenete nem „tisztá” szinusz jel, ezért ezt átlagolni, szűrni kell, hiszen a kimenő jel tartalmazza a kapcsolási frekvenciás komponenseket is. Szűrőnek egy LC szűrőt használok, amelynek méretezéséhez az alkatrészek értékeit a következő egyenletek segítségével lehet meghatározni:



.ábra: Helyettesítőkép

Az ábrán látható a szűrő és a transzformátor együttese alkotott elrendezés helyettesítőképe.

A kondenzátor meghatározásához feltételezem, hogy a kondenzátoron folyó áram effektív értéke 1%-a lesz a névleges áramnak így:

$$I_{crms} = 0.01 * I_{névl}$$

A névleges áram a trafó primerjén :

$$I_{névl} = \frac{P}{U_{primer}} = \frac{150W}{12V} = 12.5A$$

$$I_{crms} = 0.01 * 12.5A = 0.125A$$

A szükséges kondenzátor értéke:

$$C \cong \frac{I_{crms}}{\omega * U_{primer}} = \frac{0.125A}{2 * \pi * 50Hz * 12V} = 33\mu F$$

Mivel ekkora kapacitású kondenzátor kedvező árban nem kapható, így kisebb kondenzátort választottam: $C_{szűrő} = 22 \mu F$

Az induktivitás kiválasztásánál a következő egyenleteket vettem figyelembe:

$$\omega * L_{tr} = \frac{U_{névl}^2}{S_{névl}} * \frac{\epsilon}{100}$$

$$L_{tr} = \frac{230^2}{160} * \frac{1}{2 * \pi * 50} * \frac{3}{100} = 31.6mH$$

A szűrő rezonancia frekvenciája legyen legalább egy dekáddal kisebb, mint a félvezetők kapcsolási frekvenciája (25 kHz), a megfelelő szűrés érdekében, illetve ez a frekvencia ne legyen egyenlő az 50Hz-es szinusz páratlan felharmonikusaival.

$$f_{rez} \leq \frac{1}{2 * \pi * \sqrt{L * C}}$$

A soros rezonancia frekvencia sem lehet az 50Hz páratlan számú többszöröse, hiszen akkor rezonancia léphetne fel:

$$f_{sor} = \frac{1}{2 * \pi * \sqrt{\left(\frac{L_{tr} * L}{L_{tr} + L}\right) * C}}$$

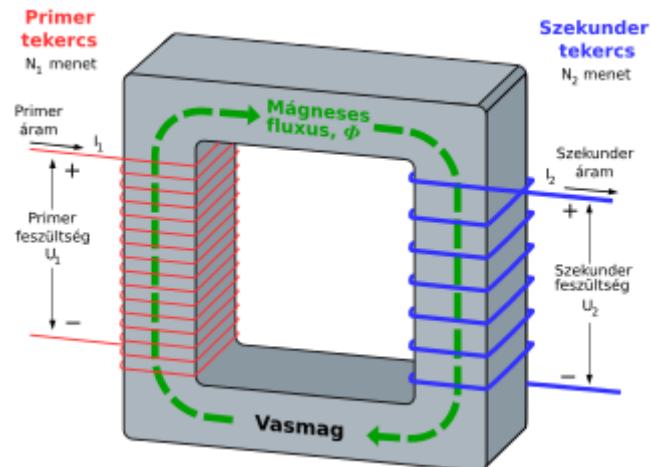
Valamint a trafó induktivitása és a kapacitás alkotta kör frekvenciája sem lehet se az 50Hz, se a 25kHz páratlan számú többszöröse:

$$f_{rez} = \frac{1}{2 * \pi * \sqrt{L_{tr} * C}}$$

Az egyenletek megoldása után a már meghatározott kapacitás mellé az induktivitás értéke $L > 184.2 \mu\text{H}$ lett. Mivel kettőt építék be az áramkörbe egy induktivitás értékét pont a felére kell választani: $L_{szűrő} = 100 \mu\text{H}$. Ezt a fojtót a Boost konverter $80 \mu\text{H}$ fojtójához hasonlóan én készítettem el. Ugyanaz az $A_V * A_A$ keresztmetszetű pormag megfelelő volt azonban, a nagyobb induktivitás miatt 41 menetszám kellett.

Transzformátor:

A 230V váltakozó feszültség előállítása kétféleképpen történhet. Egyrészt lehet, hogy felboostolom a feszültséget $\sqrt{2} * 230\text{V} = 326\text{V}$ -ra a DC/DC átalakítóval és inverterrel pedig előállítom belőle a váltakozó feszültséget. Ebben az esetben a félvezetők meglehetősen drágák lettek volna, így helyette azt a megoldást választottam, hogy a bemeneti feszültséget 25V-ra boostoltam és az inverterrel egy 12V-os effektív értékű váltakozó feszültséget állítottam elő, majd egy hálózati transzformátorral előállt 230V-os effektív értékű szinusz jel a kimeneten. Mivel a névleges teljesítmény 150W, így ettől nagyobb 160VA-es teljesítményű transzformátort választottam. A transzformátorok közül egy toroidot választottam amelynek primerfeszültsége 12V szekunderfeszültsége 230V.



.ábra: Egyfázisú transzformátor felépítése

Sznobber kondenzátorok:

A vezetéknek a hosszával arányosan nő az induktivitása. Mivel áram folyik keresztül rajta, az induktivitás egy bizonyos energiát is tárol. Abban az esetben amikor félvezetők nyitva vannak, megszakad az áramkör. Ekkor a vezetékek által alkotott induktivitásban tárolt energia áttöltődik a snubber kondenzátorba. A szükséges kondenzátor a következő képlettel határozható meg (a vezeték becsült induktivitásértéke: $L_{vez}=10 \mu H$, 5%-os feszültség-hullámosságot megengedve):

$$\frac{1}{2} * L_{vez} * (i_{Boost,ki})^2 = \frac{1}{2} * C_{snubber} * (U_{Boost,ki} + \Delta U_{Boost,ki})^2$$

$$C_{snubber} = \frac{10 \mu H * (6 A)^2}{(25 V + 0.05 * 25 V)^2} = 575 nH$$

Félvezetők:

Félvezetőnek választható FET és IGBT is. Alacsonyabb kapcsolási frekvencián a FET energiavesztése kisebb. Én kapcsolási frekvenciának 25 kHz-et választottam, ami nem esik már a hallható tartományba. Ezen a frekvencián már az IGBT-k vesztesége a relatívan kisebb. Gate meghajtók segítségével a FET-ek kapcsolási vesztesége jelentősen csökkenthető, így én ezt a megoldást választottam. További feltétel volt a félvezető kiválasztásánál, hogy a már felboostolt feszültséget (25V) a két sorba kötött inverter elviselje (a FET Source-Drain feszültsége nagyobb legyen, mint 12.5V), valamint folyamatosan át tudjon rajtuk folyni a boost konverter kimeneti árama, 6A. A FET Gatejére elhelyezendő ellenállás a következőképpen számolható ki:

$$i_{gate} = \frac{Q_{g,total}}{t_{be}} = \frac{8nC}{20ns} = 0.4A$$

$$R_{gate} = \frac{U_M}{i_{gate}} = \frac{5 V}{0.4A} = 12.5 \Omega$$

A félvezetők meghajtása a LT1158CSW#PBF nevű eszközzel történt, ami egy félhíd meghajtó, két FET meghajtására képes, őket ellenütemben vezérli. Az eszköz előnye, hogy holtidőt is beiktat az egyik FET kikapcsolása, másik bekapcsolása közé, így nem kell tartani a félvezetők összegyújtásától. A meghajtó a mikrovezérlő felől a félhíd vezérlésére szolgáló PWM jelet kapja, amit egy 3300Ω ellenálláson keresztül a földre húzok, hogy bekapcsoláskor a

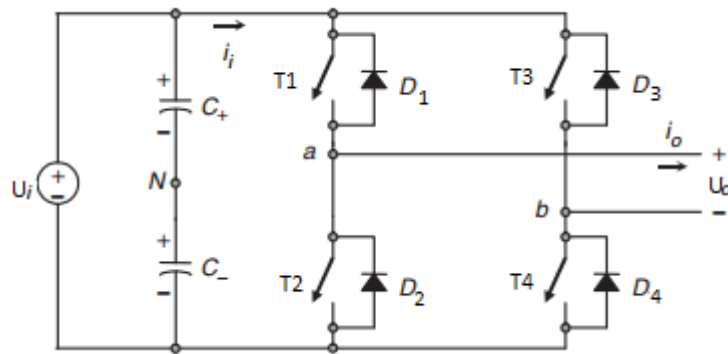
félvezetők nehogy vezérelt állapotba kerüljenek. Hiba esetén a meghajtó a /Fault kimenetén jelez a mikrovezérlőnek.

Nyomtatott huzalozású lemez:

A NYÁK tervezésénél egy lapra terveztem tenni a főkör, illetve vezérlőköri alkatrészeket is. Mivel a főkörben nagyobb áramok folynak, így a nagyobb zavarfeszültség miatt jól elkülönítettem ezt a részt a vezérlőkörtől. A két részhez külön földkitöltést (referenciaszintet) használtam, amelyet a NYÁK egy pontján összekötöttem. A NYÁK gyártása az egyetemen történt, így az ő gyártási peremfeltételeiket kellett figyelembe vennem. A huzal vastagsága $35\mu\text{m}$ a huzalszélessének legalább 0.2mm -nek a furatátmérőnek legalább 0.3mm -nek kell lennie. A vezetékek között is kell egy bizonyos szigetelési távolságot hagyni, ami a gyártó előírása szerint legalább 0.2mm , ezt a távolságot kellett tartanom a telefólia és vezetékek között is. Ez a szigetelési távolság a 230V -os feszültségű vezetékeknél megnő, ott 3mm -es szigetelési távolságot használtam. A NYÁK alsó oldalára nem tettem alkatrészt, így stabilan asztalra helyezhető. Az alkalmazandó huzal szélessége a főkör esetén szélesebb, hiszen ott található a teljesítményelektronikai rész, így ott nagyobb áramok folynak. Az inverterek kimenetén 12V -os csúcsértékű feszültség mérhető, így a 150W -os felső határteljesítmény leadásakor 17.5A is folyhat, így az inverterek kimenetén 7.5mm széles huzalokat használtam, így kisebb lett az ellenállásuk, kevésbé melegednek. Az alkatrészek elhelyezésénél egyrészt szempont volt, hogy a logikai részre kerüljenek a mérőerősítők, kijelző, valamint a mikrokontroller. A csatlakozókat (sorkapocs) a NYÁK szélére helyeztem, valamint a kis induktivitás miatt az alkatrészeket a lehető legközelebb helyeztem el egymáshoz. Mivel az alkatrészek többségéhez találtam 3D modellt, így mechanikai elhelyezkedésüket is ellenőrizni tudtam a 3 dimenziós ábrán. Szempont volt, hogy a kijelző alá ne nagyon kerüljenek alkatrészek, hiszen élesztés során nehezen tudnám mérni a feszültségüket.

Inverter:

A .ábrán látható egy egyfázisú teljes híd kapcsolású inverter. Az inverter $S1+$ és $S1-$, illetve $S2+$, $S2-$ kapcsoló nem lehet egyszerre zárva, hiszen az rövidre zárná a tápfeszültséget és a földet.



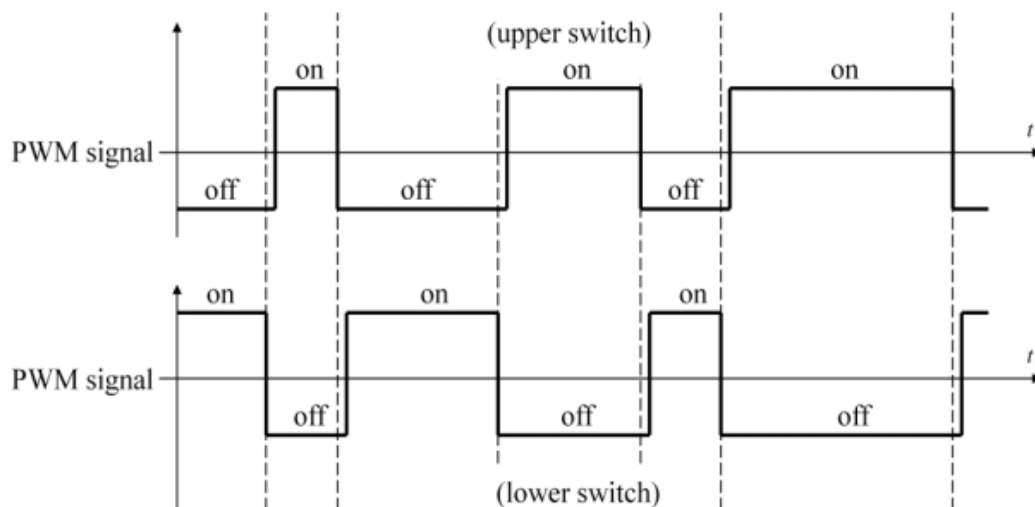
.ábra: Egyfázisú teljes híd inverter

A kapcsolók állását tekintve a következő állapotokat különböztetjük meg:

Állapot	V_o
T_1, T_4 :be, T_2, T_3 :ki	U_i
T_2, T_3 :be, T_1, T_4 :ki	$-U_i$
T_1, T_3 :be, T_2, T_4 :ki	0
T_2, T_4 :be, T_1, T_3 :ki	0

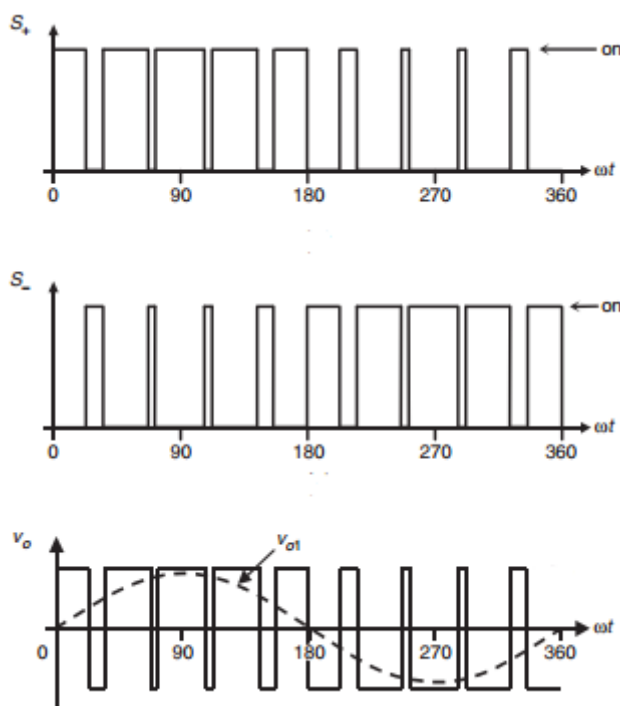
.ábra: Kimenet a kapcsolók állása szerint

Az inverterek vezérlésénél elkerülendő az az eset, amikor egy hídág két kapcsolója is zárva van. Habár rövid ideig képesek a félvezetők eldisszipálni az energiát, de hamar a félvezető végleges meghibásodásához vezet. Ennek elkerülésére holtidőt iktatunk a félvezetők bekapcsolásakor (lásd .ábra). Egy olyan FET meghajtót választottam, ami hardveres úton már beállítja a holtidőt, így erre nem kell külön figyelnem a szoftverben.



ábra: Késleltetés a PWM jel felfutó élénél

A 2-vel feljebb levő ábrán bemutattam, hogy az inverterek kapcsolása szerint kiadható a bemeneti feszültség, illetve a bemeneti feszültség fordított polaritással ($-U_i$). A PWM kitöltési tényezőjének változtatásával, és egy szűrő alkalmazásával előállítható a szinuszos váltakozó feszültség.



ábra: Az inverter kimenetén megjelenő jelalak

A ábrán jól látható, hogy a kívánt szinusz pozitív félperiódusa alatt a pozitív kimeneti feszültséget előállító kapcsolók vannak zárva (S_{1+} , S_{2-}), míg a negatív félperiódus alatt (S_{1-} , S_{2+}) bekapcsolt állapota dominál. Az is észrevehető, hogy amikor a szinusz pillanatértéke közelíti az amplitudóját, akkor legnagyobb az alkalmazott kitöltési tényező.

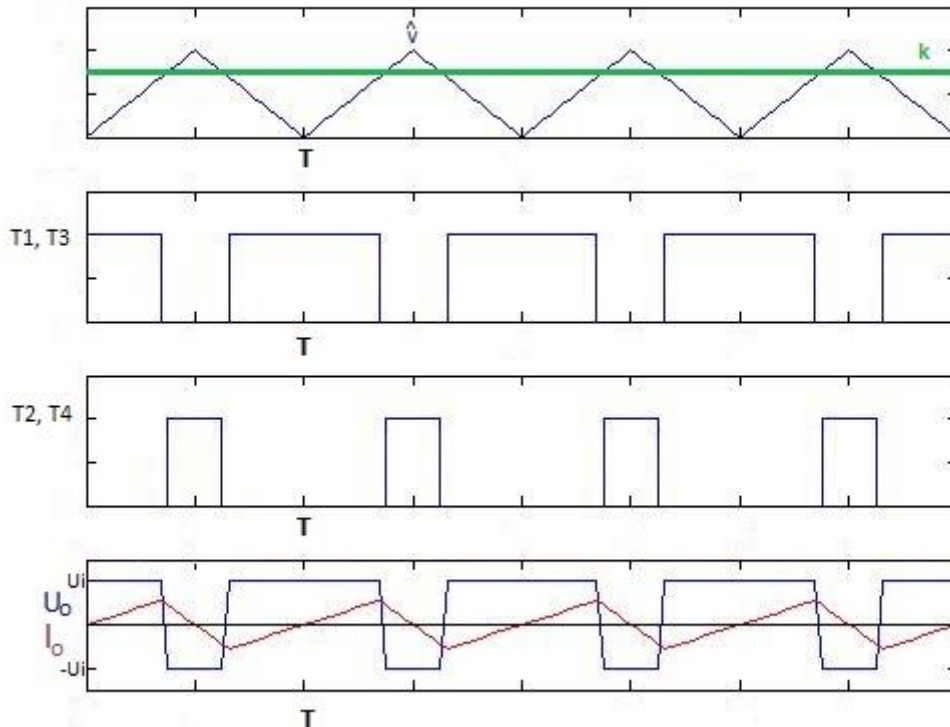
Inverterek vezérlése:

Tanulmányaim során kétfajta vezérléssel ismerkedtem meg:

Ellenütemű vezérlés:

A háromszög alakú PWM számláló kimeneti értékét egy vezérlőértékkel (v) hasonlítjuk össze, és az alapján határozzuk meg a kapcsolók állapotát. Ha a vezérlőjel értéke

nagyobb, mint a PWM aktuális értéke, akkor az T_1 kapcsoló záródik. A hídág alsó kapcsolója (T_2) pontosan a felső negáltja, míg a másik hídágban $T_4=T_1, T_3=/(T_4)$.



Ennél a vezérlési módnál a következőképpen alakul az áramhullámosság:

A kimeneti feszültség:

$$U_o = \frac{2 * U_i}{PWM_{csúcs}} * v - U_i = U_i * \left(\frac{2 * v}{PWM_{csúcs}} - 1 \right)$$

$$\frac{v}{PWM_{csúcs}} = \frac{\frac{U_o}{U_i} + 1}{2}$$

Az idő, amikor a T_1, T_4 kapcsolók záródnak:

$$t_1 = \frac{v}{PWM_{csúcs}} * T_{sw}$$

Az áramváltozás, amikor csökken az áram:

$$\Delta i = \frac{t_1 * (U_o - U_i)}{L}$$

Az előző egyenleteket felhasználva:

$$\Delta i = \frac{T_{SW} * \frac{\frac{U_i}{U_o} + 1}{2}}{L} * (U_o - U_i) = \frac{T_{SW} * U_o}{2 * L} * \left(1 - \left(\frac{U_i}{U_o}\right)^2\right)$$

A kimeneti áram lüktetése akkor lesz a legnagyobb, amikor az inverter kapcsain mérhető feszültség éppen nulla.

Ennek értéke:

$$\Delta i = \frac{T_{SW} * U_{dc}}{2 * L}$$

T_{SW} : kapcsolási periódusidő

U_i : inverter kapcsain mérhető feszültség

U_o : a szűrőkör kapcsain mérhető feszültség

L : szűrőkör induktivitása

Eltolásos vezérlés:

Ebben az esetben két vezérlőjelet is használok. Az egyik vezérlőjel neve legyen v_1 a másiké v_2 . A kapcsolók vezérlése a következőképpen történik:

T_1 -t akkor zárom, ha a v_1 vezérlőjel értéke nagyobb, mint a komparálójel (k), valamint T_4 -t akkor, ha a v_2 vezérlőjel nagyobb, mint a komparálójel. Mivel egy hídág tranzisztorai nem lehetnek egyszerre zárva, így S_1 - vezérlése ellentétes T_1 -al, valamint T_3 vezérlése ellentétes T_4 -al. A két vezérlőjel közül a v_1 szabadon választható, v_2 értéke a következőképpen adódik:

$$v_2 = PWM_{csúcs} - v_1.$$

Az áram hullámossága itt a következőképpen alakul:

A v_1 -re vonatkozó kitöltési tényező itt is megegyezik az ellenütemű vezérlésnél látottal:

$$\frac{v_1}{PWM_{csúcs}} = \frac{\frac{U_i}{U_o} + 1}{2}$$

Az áramhullámosság csökkenő áramra felírva:

$$\Delta i = \frac{U_o - U_i}{L} * \left(\frac{t_1 - t_3}{2}\right); \quad t_3 = T - t_1$$

A kitöltési tényezőre felírt egyenletet is felhasználva:

$$\Delta i = U_o * \frac{1 - \frac{U_i}{U_o}}{L} * T_{sw} * \left(\frac{\frac{v_1}{PWM_{csúcs}} - (1 - \frac{v_1}{PWM_{csúcs}})}{2}\right)$$

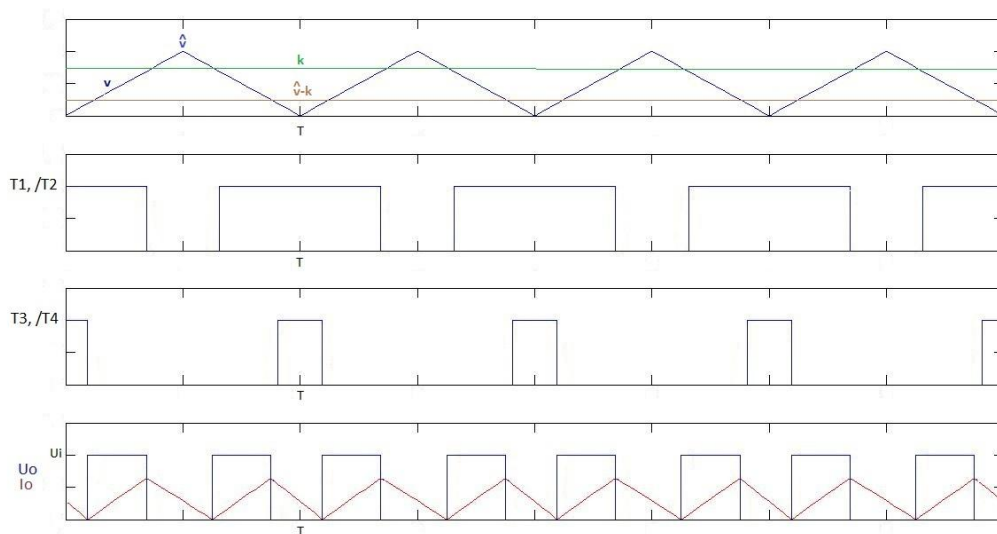
$$\Delta i = \frac{U_o * T_{sw}}{2 * L} * \left(1 - \frac{U_i^2}{U_o^2}\right) * \frac{U_i}{U_o}$$

Legnagyobb az áramhullámosság, ha a bemenőfeszültség (U_i) a kimenőfeszültség (U_o) fele:

Ennek értéke:

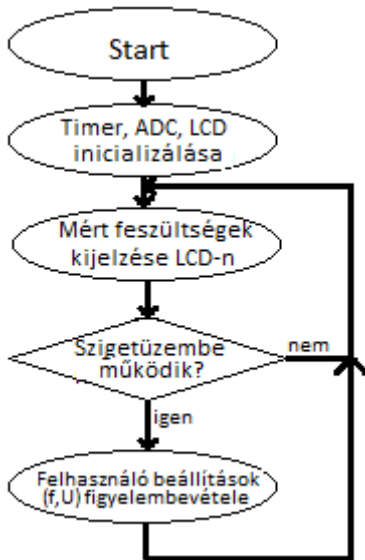
$$\Delta i = \frac{U_o}{8 * L} * T_{sw}$$

Tehát negyedakkora, mint az ellenütemű vezérlésnél volt, így ezt a vezérlési módot választottam az inverterek vezérlésénél.



Beágyazott szoftver

A beágyazott szoftvert Eclipse fejlesztőkörnyezetben írtam. A szoftvernek meg kellett valósítania: a grafikus LCD kijelző vezérlését, a szinkron Boost konverter FET-jeit 25 kHz frekvenciával a kiszámolt kitöltési tényezővel kapcsolgatni, a mért analóg értékeket feldolgozni, váltóáramot előállító invertereket vezérelni.



Inicializálás:

Először a GPIO pineket kellett beállítanom. A pinek egy csoportját kimenetként, míg másokat bemenetként kellett konfigurálni. Itt beállítható a meghajtás módja (Open drain, Push-Pull), illetve belső aktív fel- és elhúzás is beállítható. A pinek működési sebességére is három üzemmód választható: gyors, lassú, közepes. Egyes pineken PWM jelet várjuk őket alternatív funkcióban használjuk, a megfelelő timert „hozzákötjük”. Az analóg jelek fogadására szolgáló pineken is jellezük, hogy analóg jellel dolgozunk, beállítjuk a felbontást is.

Boost konverter:

A Boost konverter megvalósításához a 25kHz-es impulzusszélesség modulált jelhez meg kellett határoznom a kitöltési tényezőt (d) . Ehhez mérnem kellett a konverter bemenetén és kimenetén levő analóg feszültségértékeket. Az analóg jelek méréséhez ennek megfelelően kellett konfigurálnom a pineket:

```
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AN;
```

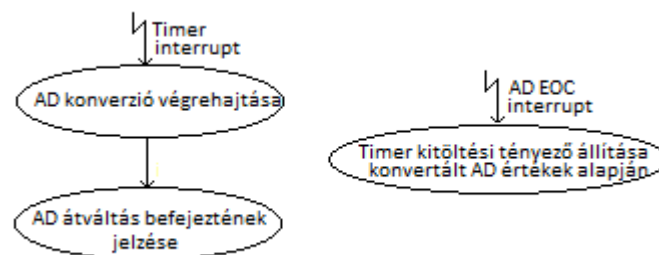
Valamint konfigurálnom kellett az ADC-t (ADC_Configuration()) , és egy ADC értékét kiolvasó függvényt is meg kellett írnom (readADC1()).

Az ADC mellett a PWM-hez konfigurálnom kellett egy timert:

```
void InitializeTimer3()
{
    RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM3, ENABLE);

    TIM_TimeBaseInitTypeDef timerInitStructure;
    timerInitStructure.TIM_Prescaler = 80;
    timerInitStructure.TIM_CounterMode = TIM_CounterMode_Up;
    timerInitStructure.TIM_Period = 4000;
    timerInitStructure.TIM_ClockDivision = TIM_CKD_DIV1;
    timerInitStructure.TIM_RepetitionCounter = 0;
    TIM_TimeBaseInit(TIM3, &timerInitStructure);
    TIM_Cmd(TIM3, ENABLE);
}
```

A program blokkvázlata az alábbi ábrán látható:



.ábra: Booster vezérlési folyamat ábra

Minden timer periódus végén jön egy interrupt, amely AD konverzióra szólítja fel a bejövő feszültséget mérő ADC-t. Az átváltás befejeztekor a bejövő feszültségből kiszámolja a timernek beállítandó kitöltési értéket, hogy a Booster kimenetén a bejövő feszültségtől függetlenül 25V-os feszültség jusson az inverterekre.

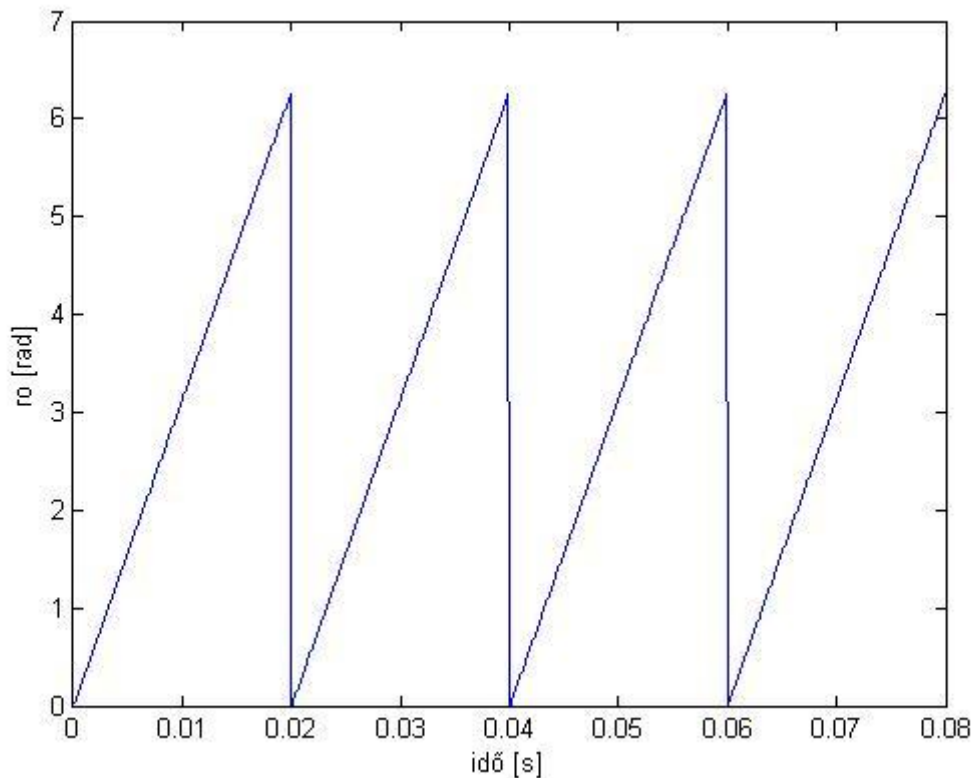
Inverterek vezérlése:

Az inverterek vezérlése a 3-as timer két csatornájáról történik. Eltolásos vezérlést alkalmazok, így az egyik csatornára x komparálási szintet állítok, a másik csatornának a

komparálási szint a periódus-x lesz. A szinuszos jelhez folyamatosan számolnom kell a kitöltési tényezőt. Ehhez egy referenciajelet használok és minden egyes timer interrupt érkezésekor növelem a fázisszöget (.ábra). Az invereket vezérlő timer 25kHz-el éri el végértékét, 40 μ s-onként megszakításokat generál. Nekem egy 50Hz-es frekvenciájú referenciajelre van szükségem, vagyis 20ms-onként nő 2π -t a fázisszög.

$$\frac{20 \text{ ms}}{2\pi} = \frac{40 \mu\text{s}}{\Delta\rho}$$

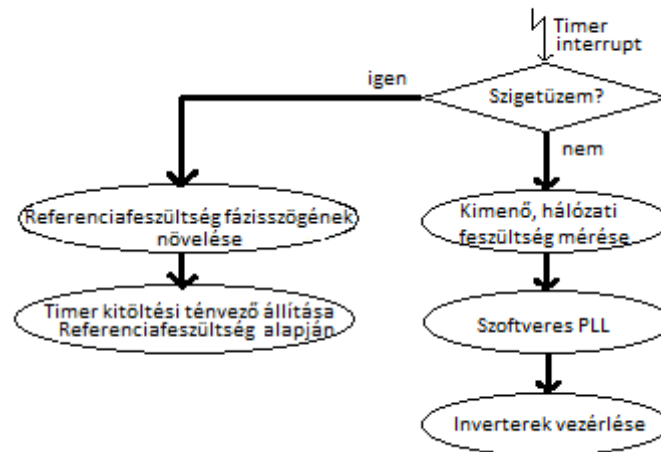
$$\Delta\rho = \frac{4\pi}{1000}$$



.ábra: Referenciajel fázisszögének változása időben

Tehát minden egyes timer interruptban megnövelem a referenciajel fázisszögét a $\Delta\rho$ értékével, majd ha elérte a 2π -t nullázom. Így a referenciajelem $U_{ki,ref} = 12 * \sqrt{2} * \sin(\rho)$ lesz. Azért, hogy a kondenzátorokban ne folyjon nagy áram a bekapcsolás utáni első másodpercben az amplitúdót fokozatosan felfuttatom. A timereknek adandó kitöltési

tényezőt pedig a $d = \frac{U_{ki,ref}}{U_{ki,Booster}}$ képlet alapján számítom.



.ábra: Inverterek vezérlésének folyamat ábrája

LCD kijelző:

Az LCD kijelző egy 64 soros 128 oszlopos pontmátrixból áll, amelynek minden egyes pontja külön vezérelhető. A pontok vezérlését két chip végzi, így az egyes elküldött utasításoknál a CS1, CS2 vonalon jelezni kell, hogy az adott parancsot melyik chipnek küldjük. Használathoz nem elegendő a chipek feszültség alá helyezése, hanem mindkét chipnek el kell küldeni a bekapcsol parancsot, amelyet a következő függvény végez:

```

void LCD_On()
{
    LCD_Command(CS, 0x3F);
}
  
```

A fenti függvény meghívja az LCD_Command függvényt mindkét chipre. Az LCD command a bemeneti paraméterként megadott chip select (cs) értéke alapján kiválasztja a megfelelő chipet, majd a kapott parancs hexadecimális értékét az adatvonalakra teszi (itt a B port alsó bitjeire). A D/I vonalon jelzi, hogy utasítás következik, valamint az enable vonal logikai magas szintre emelésével jut érvényre az utasítás. Legvégül a chip select, enable vonalakat is logikai alacsony szintre állítja. Az utasítás érvényre jutásához nem elegendő a az enable vonalat magas szintre emelni, ott is kell tartani egy meghatározott ideig, ezért oda késleltetés került.

```

//LCD parancs küldése
void LCD_Command(uint8_t cs, char command)
{
    uint16_t temp;
    if(cs==1)    GPIO_WriteBit(GPIOB, LCD_CS1, Bit_SET); // a cs paraméter
    alapján megfelelő chipek engedélyezése
    if(cs==2)    GPIO_WriteBit(GPIOB, LCD_CS2, Bit_SET);
    if(cs==0)    GPIO_WriteBit(GPIOB, LCD_CS1 | LCD_CS2, Bit_SET);
}
  
```

```

        GPIO_WriteBit(GPIOB, LCD_CS1 | LCD_CS2, Bit_SET);           //mindkét
chip engedélyezése
        temp=GPIOB->ODR & 0xFF00;                                   //parancsbyte
        kerüljön a DB0..DB7-re
        GPIOB->ODR=temp|command;
        GPIO_WriteBit(GPIOB, LCD_DI, Bit_RESET);                   // parancs következik
        GPIO_WriteBit(GPIOB, LCD_EN, Bit_SET);                     //EN=1
        //delay
        GPIO_WriteBit(GPIOB, LCD_EN, Bit_RESET);                   //EN=0
        //delay
        GPIO_WriteBit(GPIOB, LCD_CS1 | LCD_CS2, Bit_RESET);       //chip
engedélyezések visszavonása
    }

```

Az adatküldő függvény (LCD_Data) teljesen megegyezik az LCD_Command függvénnyel, annyi különbséggel, hogy ott D/I nem utasítást, hanem adatot jelzünk.

Az LCD_Line, LCD_Column függvénnyel kiválaszthatjuk, hogy a kijelző melyik sorát, oszlopát akarjuk vezérelni. Ha a kijelző egy oszlopába írtunk akkor az oszlopszámláló értékét növeli, így a következő kiírt adat a már eggyel megnövelt sorszámú oszlopba kerül.

A kijelző 64 soros, amit feloszt 8 lapra, így egy lapra 8 sor kerül, ami megegyezik az adatvezetékek számával. Tehát célszerű a kijelzőt a lapfelosztás szerint használni, így 8 sornyi szöveg írható a kijelzőre. A kijelzőn egy karaktert 5 oszlop szélesen ábrázolok. A karakterekhez készült egy tömb, ami tartalmazza az összes karakter ábrázolásához szükséges bájtokat (5 oszlop széles, így 5 bájt kell egy karakter ábrázolásához), amiket a fonts nevű character tömbben tárolom- egy sorba 5 adatot írtam, hiszen egy karakterhez 5 bájt tartozik:

```

char fonts[] = {
    0x00, 0x00, 0x00, 0x00, 0x00, // SPACE    0
    0x00, 0x00, 0x5F, 0x00, 0x00, // !        1
    0x00, 0x03, 0x00, 0x03, 0x00, // "        2
    0x14, 0x3E, 0x14, 0x3E, 0x14, // #        3
    0x24, 0x2A, 0x7F, 0x2A, 0x12, // $        4
    0x43, 0x33, 0x08, 0x66, 0x61, // %        5
    0x36, 0x49, 0x55, 0x22, 0x50, // &        6
    0x00, 0x05, 0x03, 0x00, 0x00, // '        7
    0x00, 0x1C, 0x22, 0x41, 0x00, // (        8
    0x00, 0x41, 0x22, 0x1C, 0x00, // )        9
    0x14, 0x08, 0x3E, 0x08, 0x14, // *       10
    0x08, 0x08, 0x3E, 0x08, 0x08, // +       11
    0x00, 0x50, 0x30, 0x00, 0x00, // ,       12
    0x08, 0x08, 0x08, 0x08, 0x08, // -       13
    0x00, 0x60, 0x60, 0x00, 0x00, // .       14
    0x20, 0x10, 0x08, 0x04, 0x02, // /       15
    0x3E, 0x51, 0x49, 0x45, 0x3E, // 0       16
    0x00, 0x04, 0x02, 0x7F, 0x00, // 1       17
    0x42, 0x61, 0x51, 0x49, 0x46, // 2       18
    0x22, 0x41, 0x49, 0x49, 0x36, // 3       19

```


...

A sztringeket mindig a lezáró nulla ('\0') karakterig iratom ki, minden egyes karakterre meghívom a LCD_Character függvényt, ami bemenő paraméterként megkap egy mutatót az aktuális karakterre, valamint egy mutatót az aktuális oszlopsorszáma- azért kell, hogy mutató legyen az oszlopsorszáma, hogy a character függvény lefutása után is megmaradjon az általunk is beállítható oszlopsorszáma értéke. A character függvény egy nagy switch-case szerkezet, hiszen az adott karakter alapján eldönti, a fonts nevű karaktertömb hanyadik elemeit kell a kijelzőre írunk:

```
void LCD_Character(char character, char page, char *column)           //egyes
karakterekhez tartozó fonts kezdő oszlop a képernyőn : *column
{
    switch(character){
        case 'a':LCD_Font(page,*column,65*5); break;
```

Tehát, ha éppen egy 'a' karaktert akarunk kiírni, akkor meghívódik az LCD_Font nevű függvény, amelynek harmadik paramétere a fonts nevű karaktertömbben az 'a' karakterhez tartozó index (az 5-ös szorzótényező, azért van ott, mert egy karaktert 5 oszlop szélesen iratok ki).

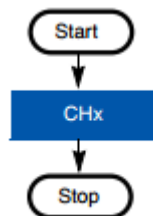
Az LCD_Font függvény először kiad az aktuális lap meghatározásához egy utasítást, majd egy 5-ször lefutó ciklus következik, ami az oszlopszámláló értékétől függően kiválasztja, hogy melyik chipre vonatkozik a kiírandó adat, valamint, ha oszlopszámláló értéke nagyobb, mint 64, akkor az oszlopszámot virtuálisan csökkenti, hiszen a 65. oszlop adata, a második chip első oszlopába kell, hogy kerüljön.

ADC kezelése:

Mivel szükségünk van különböző feszültségértékek mérésére, így nélkülözhetetlen az AD átalakító használata. A bemeneti feszültséget mintavételezi, és nagyságát egy digitális számmal fejezi ki, ezáltal az analóg jel időben és értékben is diszkrét lesz. Az STM32 AD konvertere egy eléggé komplex periféria. Kezelése rugalmas, így sokféle feladatot is képes ellátni. Az általam használt AD átalakító, a fokozatos közelítés (szukcesszív approximáció) elvét használja. Ennek a módszernek előnye a magas felbontás, közepes mintavételezés mellett.

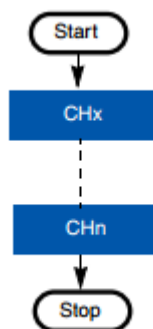
ADC két átváltási módot támogat: szokásos és beillesztett. A szokásos az, ami a többi mikrokontrollernél is megszokott- mindegyik csatorna ugyanazon az adatregiszteren osztozik. Többfajta átváltási mód létezik:

Egy csatorna, egyszeri átváltás **ábra**: Ez a legegyszerűbb ADC mód, egy csatorna egyszeri átváltását eredményezi, aztán leáll.



.ábra: Egy csatorna, egyszeri átváltása

Több csatorna (scan), egyszeri átváltás **ábra**: Ebben a módban több csatornát mintavételezünk, ahol csatornánként beállítható a mintavételezési idő, illetve a csatornák közötti átváltási sorrend.



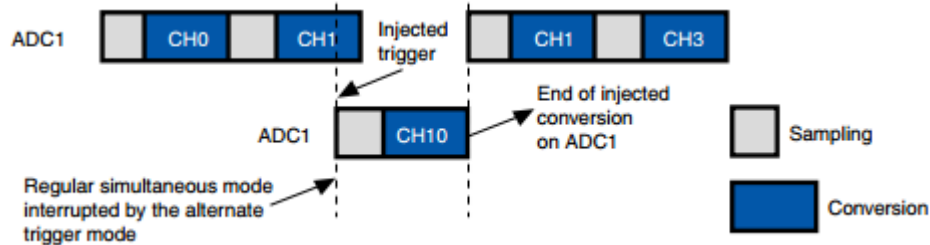
.ábra: Több csatorna, egyszeri átváltása

Az egyszeri átváltások mellett létezik

Az eredmény 16 bites regiszterben tárolódik. Beállítható a konverzió felbontása (maximum 12 bit), illetve, hogy a hasznos bitek a regiszter felső, vagy alsó felébe kerüljenek (a nem használt biteket kinullázza).

Az egyszeri átváltások mellett létezik a folyamatos átváltás is, amikor az ADC a háttérben fut a CPU közbelépése nélkül. DMA használatával a konvertált értékek a memóriába íródnak, anélkül, hogy a CPU-t működésében feltartóztatnánk.

Harmadik lehetőség a beillesztett csoport mód (injected group mode), ahol a szokásos átváltási mód, egy szoftveres, vagy külső trigger hatására megszakítódik, és a magasabb prioritású csatorna(k) átváltása történik.

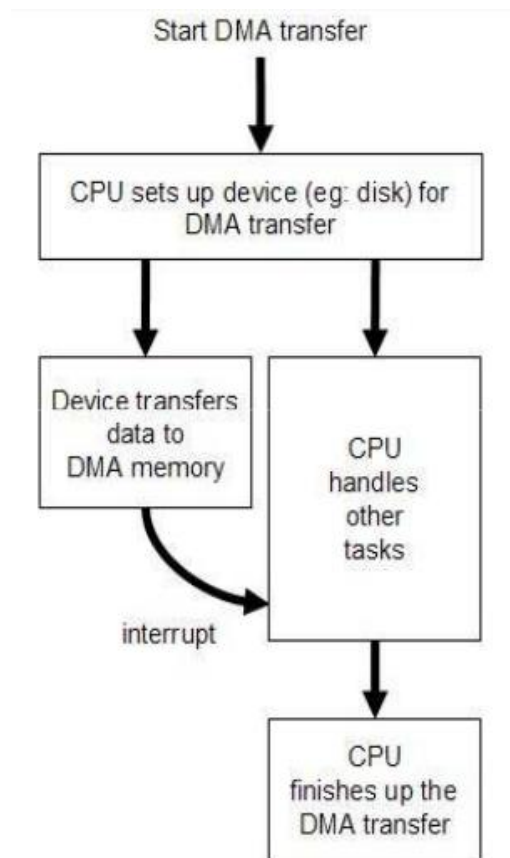


.ábra: Beillesztett csoport mód [4]

Az összes ADC módot nem részleteztem, csak a számomra relevánsokat. Én az alkalmazásomban a folyamatos átváltást alkalmaztam DMA átvitelrel, mivel az interruptos átváltás esetén nem tudtam volna megkülönböztetni, hogy az ADC melyik csatornájának értéke váltódott át.

DMA használata:

A rugalmas DMA (direct memory access) képes a memória-memória, periféria-memória, memória-periféria közötti adatcserét lebonyolítani. A DMA alkalmazása csökkentheti a processzor által kiszorgálandó megszakítások számát, továbbá magát a hardware-t is picit esszerűsítheti, hiszen nem kell minden ADC-hez önálló FIFO-t rendelni. A DMA alkalmazása két busz ciklust igényel: periféria olvasása, memória írása. A DMA vezérlő semmilyen feldolgozást nem végez az adatokon. Az alábbi **ábrán** látható, hogy a DMA használatának előnye, vagyis, hogy képes tehermentesítenei a processzort.



. ábra: A DMA párhuzamos működése a processzorral [5]

A DMA konfigurálásakor beállítható, hogy mekkora adathosszúságot kezeljen. Ez lehet bájt (8 bit), félszó (16 bit), szó (32 bit). Esetemben az ADC felbontása 12 bit, ami 2 bájtba fér bele, így a félszónyi adathosszt választottam. Cirkuláris címezés azért kellett, mert mindig ugyanarra a memóriacímre akartam írni a beolvasott ADC értékét, így egy változó íródik folyamatosan felül.

Hivatkozás:

[1] http://fft.gau.hu/fizika/FIZIKA2/1011/lev/Napelem_karakterisztika_merese.pdf

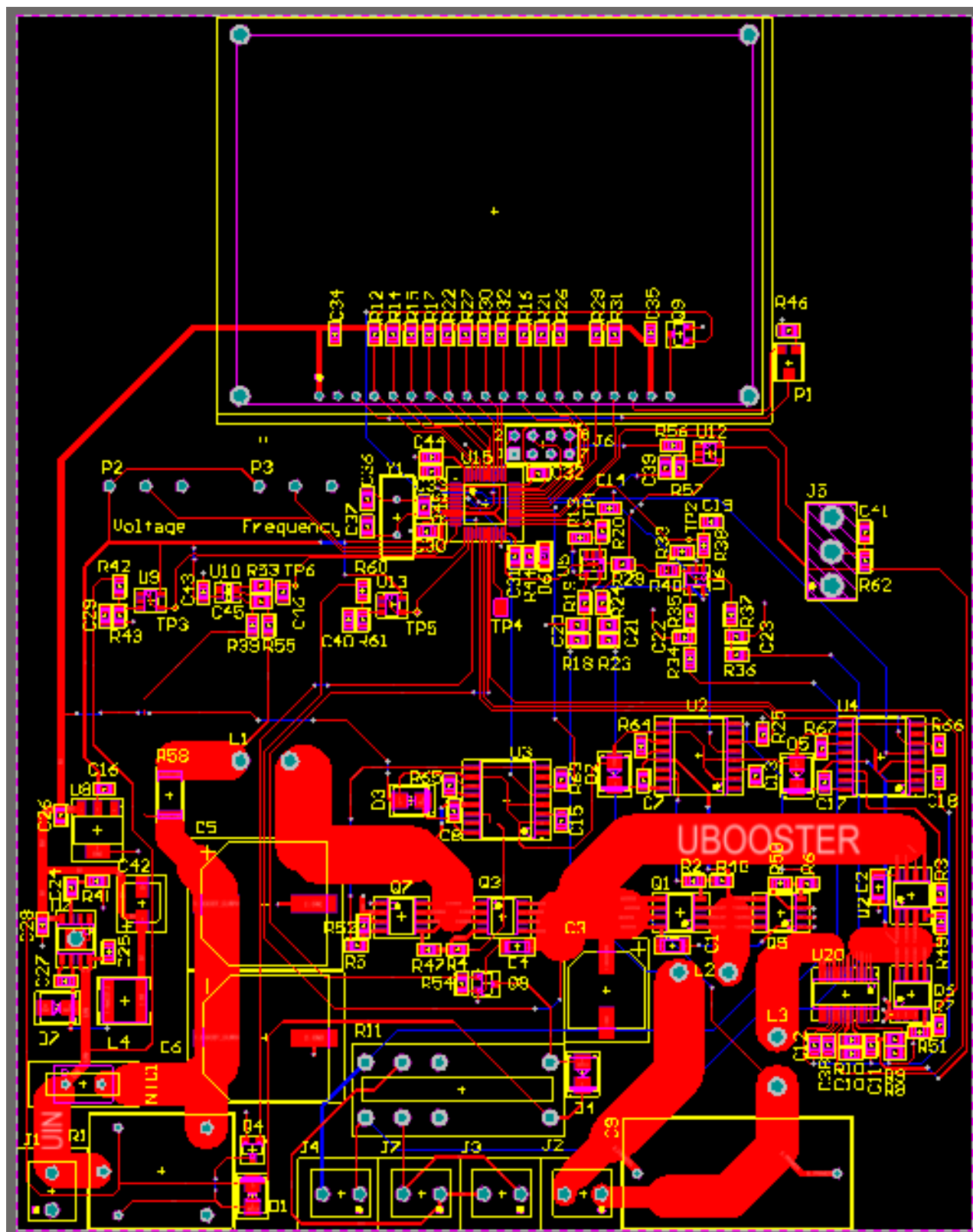
[1.1] http://moodle.autolab.uni-pannon.hu/Mecha_tananyag/korszeru_teljesitmeny_elektronika/ch08.html

[2] <https://www.victronenergy.com/upload/documents/Datasheet-BlueSolar-Polycrystalline-Panels-EN.pdf>

[3] http://www.st.com/content/ccc/resource/technical/document/reference_manual/fa/06/c7/15/3b/2a/4c/f1/DM00041563.pdf/files/DM00041563.pdf/jcr:content/translations/en.DM00041563.pdf

[4] http://www.st.com/content/ccc/resource/technical/document/application_note/c4/63/a9/f4/ae/f2/48/5d/CD00258017.pdf/files/CD00258017.pdf/jcr:content/translations/en.CD00258017.pdf

[5] https://www.google.hu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0ahUKEwiHg7nr087TAhUBvhQKHVrYC2QQFggqMAE&url=https%3A%2F%2Fwww.mit.bme.hu%2Fsyste%2Ffiles%2Foktatas%2Ftargyak%2F7336%2FNagyteljMikrov_07_NVIC_DMA_v04.pdf&usg=AFQjCNEOtWG84HebhQLILqylRPOw1QU4AQ&sig2=Wk2kpB8hI2hRINQcfpZwJg



PCB földkitöltés nélkül

