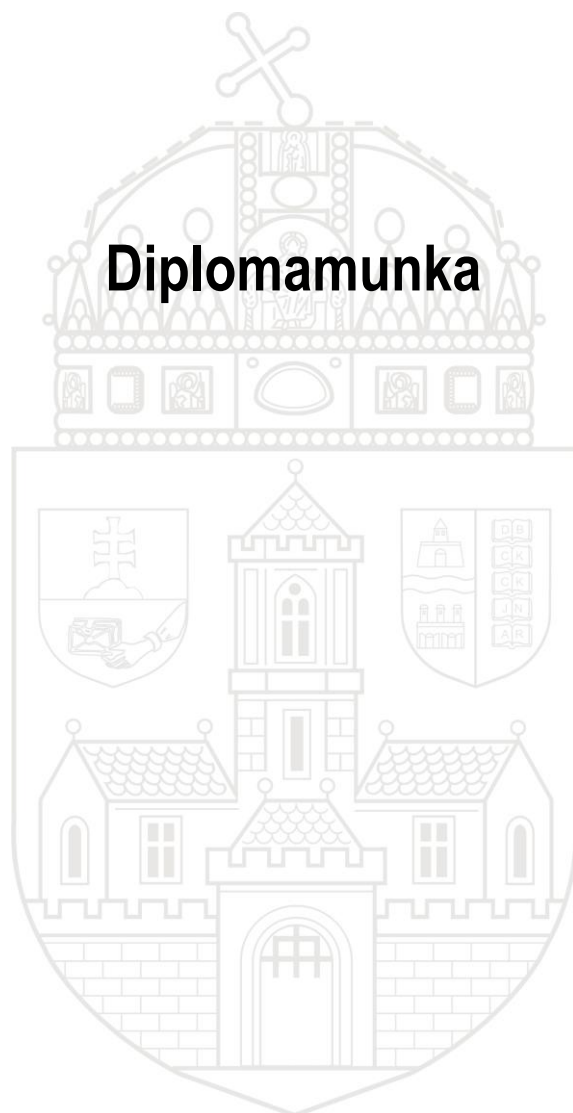




ÓBUDAI EGYETEM

**Kandó Kálmán Villamosmérnöki Kar
Automatika Intézet**



**OE-KVK
2021**

**Geiszt Dániel
P7B1JO
T006073/FI12904/K**



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Óbudai Egyetem
Kandó Kálmán Villamosmérnöki Kar
Kandó Kálmán Villamosmérnöki Kar

DIPLOMAMUNKA FELADATLAP

Hallgató neve: GeisztDániel

Diplomamunka száma: SZDP7B1J02020

Törzskönyvi száma: T006073/F112904/K

Neptun kódja: P7B1J0

Szak: villamosmérnöki

Specializáció: Villamosenergia-átalakítók, -tárolók és irányítástechnika specializáció

A dolgozat címe: Léptet motoros hajtások vizsgálata/tervezése

A dolgozat címe angolul: Examination/planning of stepper motor drives

A feladat részletezése: A léptet motorok felépítésük szerinti csoportosítás, a változó reluktanciájú, az állandómágneses és hibrid kivitelek jellemzése, összehasonlítása. A különböző vezérlési módok áttekintése. Az egész lépéses, fél lépéses és mikrolépéses maghajtási módok bemutatása. Egy léptet motor vezérlő áramkör tervezése, implementálása.

Intézményi konzulens neve: Dr. Számel László

A kiadott téma elévülési határideje: 2023. május 15.

Beadási határidő : 2021.05.15.

A diplomamunka: Nem titkos.

Kiadva: Budapest, 2021.03.16.

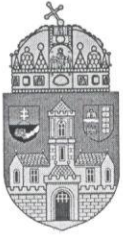
P.H

.....
Intézetigazgató

A dolgozatot beadásra alkalmasnak találom: 2021. május 15.

.....
belső konzulens

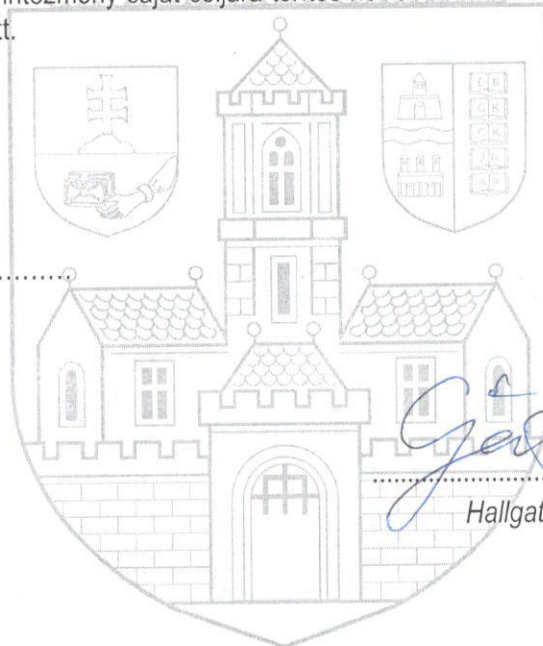
.....
külső konzulens



HALLGATÓI NYILATKOZAT

Alulírott hallgató kijelentem, hogy a diplomamunka saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült diplomamunkában található eredményeket az egyetem és a feladatot kiíró intézmény saját céljára térítés nélkül felhasználhatja, a titkosításra vonatkozó esetleges megkötések mellett.

Budapest, 2021. 05. 15.



[Handwritten signature]

Hallgató aláírása

ÖNÁLLÓ PROJEKT / SZAKDOLGOZAT / DIPLOMATERV KONZULTÁCIÓS NAPLÓ

LEADÁSI HATÁRIDŐ:
AZ AKTUÁLIS FÉLÉV SZORGALMI IDŐSZAKÁNAK UTOLSÓ NAPJA

| | | | |
|---|---|---|---|
| Hallgató neve: <div style="border: 1px solid black; padding: 2px;">Geiszt Dániel</div> | NEPTUN kód: <div style="border: 1px solid black; padding: 2px;">P7B1JO</div> | Képzés: <div style="border: 1px solid black; padding: 2px;"><input type="checkbox"/> BSC <input checked="" type="checkbox"/> MSC</div> | Tagozat: <div style="border: 1px solid black; padding: 2px;"><input type="checkbox"/> Nappali <input checked="" type="checkbox"/> Levelező <input type="checkbox"/> Távoktatás</div> |
|---|---|---|---|

| | |
|--|---|
| Telefon: <div style="border: 1px solid black; padding: 2px;">06-20/244-2922</div> | E-mail cím: <div style="border: 1px solid black; padding: 2px;">geisztdaniel@gmail.com</div> |
|--|---|

| | |
|--|--|
| Egyetemi konzulens: <div style="border: 1px solid black; padding: 2px;">Dr. Számel László</div> | Külső konzulens: <i>(nem szükséges kitölteni)</i> <div style="border: 1px solid black; padding: 2px;">-----</div> |
|--|--|

Tantárgy:

Önálló projekt I. | Önálló projekt II. | Szakdolgozat | Diplomaterv I. | Diplomaterv II.

Téma megnevezése:

Léptetőmotoros hajtások vizsgálata/tervezése

Kérjük, hogy az adatokat elektronikusan vagy nyomtatott nagy betűkkel töltsse ki!

| Alk. | Dátum | Tartalom | Aláírás |
|------|-------------|----------------------|---------|
| 1. | 2021.02.25. | Feladat megbeszélése | |
| 2. | 2021.03.12. | Feladat megbeszélése | |
| 3. | 2021.03.29. | Elméleti háttér | |
| 4. | 2021.04.26. | Összefoglalás | |
| 5. | 2021.05.08. | Ellenőrzés | |

A Konzultációs naplót összesen 5 alkalommal az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel, illetve mindkét konzulenssel alá kell íratatni a beadás előtt.

A hallgató a tantárgyat sikeresen teljesítette.

Önálló projekt és Diplomaterv tantárgyak esetén kitöltendő:

Érdemjegye:

Védés várható ideje (év és hónap):

Szakdolgozat és Diplomaterv II. tantárgy esetén kitöltendő:

Budapest,

.....
Egyetemi konzulens

.....
Külső konzulens

Ezt a lapot NEM KELL a Szakdolgozatba / Diplomatervbe bekötni.

Tartalmi kivonat

Diplomamunkámban a léptetőmotoros hajtás vizsgálatával foglalkozom. A dolgozatom első részében a léptetőmotorok általános jellemzése kerül bemutatásra. Ebben a fejezetben áttekintem a léptetőmotor típusokat, illetve részletesen ismertetem a vezérlési módokat. Ezek után az áramkorlátozás elvének bemutatásával folytatom a dolgozatomat.

A munkám középpontjában egy léptetőmotor vezérlő áramkör tervezése volt az elsajátított ismereteknek megfelelően. Az Atmel cég 8-bites ATmega8 nevű mikrokontrollerét és egy L298-as IC-t használtam erre a célra. A mikrokontroller programjának részletes áttekintése után a felhasznált eszközök kerülnek bemutatásra.

A diplomamunkámat a megvalósított vezérlő elemzésével, következtetések összegzésével zárom.

Abstract

This thesis describes the concept of the stepper motor drivers. In the first part of the thesis a general description is provided about the stepper motors. In this chapter the types of the stepper motors are reviewed and the working methods are described in a detailed form. After that the thesis continues with the principles of the current limitation.

The key focus of the thesis is the designing of a stepper motor driver in accordance with the knowledge acquired. For that purpose I have used the 8-bit ATmega8 microcontroller made by the Atmel company and a L298 IC. After a detailed overview of the microcontroller program, the devices used are presented.

The thesis is closed with the analysis of the implemented control panel and the summary of the conclusions made.

Tartalomjegyzék

| | |
|--|-----------|
| 1 Bevezetés | 5 |
| 2 Léptetőmotorok jellemzése..... | 6 |
| 2.1 A léptetőmotorok rövid áttekintése..... | 6 |
| 2.2 Léptetőmotorok típusai..... | 7 |
| 2.2.1 Változó reluktanciájú léptetőmotorok | 8 |
| 2.2.2 Állandómágneses léptetőmotorok | 11 |
| 2.2.3 Hibrid léptetőmotorok | 12 |
| 2.3 Léptetőmotorok előnyei és hátrányai..... | 14 |
| 2.4 Fázistekercsek kapcsolásának lehetőségei..... | 15 |
| 2.4.1 Unipoláris..... | 15 |
| 2.4.2 Bipoláris..... | 16 |
| 2.5 Meghajtási üzemmódok | 17 |
| 2.5.1 Egészlépéses üzemmód (Full Step)..... | 17 |
| 2.5.2 Fél lépéses üzemmód (Half Step)..... | 19 |
| 2.5.3 Mikrolépéses üzemmód (Micro Step)..... | 21 |
| 2.6 Áramkorlátozás | 22 |
| 2.6.1 Az alapok | 22 |
| 2.6.2 Az áramkorlátozás módszerei..... | 25 |
| 3 Léptetőmotor vezérlésének megvalósítása | 28 |
| 3.1 Léptetőmotor vezérlő IC: L297 | 28 |
| 3.2 Kéthidas motorvezérlő IC: L298 | 29 |
| 3.3 Vezérlés (nyílt hurkú)..... | 30 |
| 3.4 Zárt hurkú szabályozás | 32 |
| 4 Implementálás..... | 34 |
| 4.1 PWM | 34 |
| 4.2 AD átalakító..... | 35 |
| 4.3 Szabályozó..... | 36 |
| 4.4 Lépésmódok..... | 37 |
| 4.4.1 Mikrolépés..... | 37 |
| 4.4.2 Teljes és fél lépéses mód..... | 38 |
| 5 A hardver felépítése..... | 40 |

| | |
|---|-----------|
| 5.1 Tápegység..... | 40 |
| 5.2 ATmega8 mikrokontroller..... | 41 |
| 5.3 L298 | 42 |
| 5.4 Árammérés és műveleti erősítő | 43 |
| 5.4.1 Műveleti erősítő..... | 43 |
| 5.4.2 Aluláteresztő szűrő..... | 45 |
| 6 A megvalósított vezérlő | 47 |
| 7 Összefoglalás | 50 |
| Függelék..... | 51 |
| Irodalomjegyzék, források..... | 55 |

1 Bevezetés

Napjainkban az ipari automatizálás egyre népszerűbbé válik. Nem csak az iparban, hanem a kisebb vállalkozások körében is nagyobb lett az igény az automatizálás iránt. A kisvállalkozások nem feltétlenül igényelnek olyan hatalmas pontossággal ellátott technológiát, mint az automatizált ipar, viszont megkövetelnek egy bizonyos pontosságot úgy, hogy a költségeket mindemellett minimalizálni tudják. Ezért van az, hogy az iparban általánosan használt egyenáramú (DC)-motorok mellett egyre nagyobb teret hódít magának a léptetőmotor.

Az elmúlt években a digitális elektronika hatalmas méretű növekedése közvetett módon befolyásolta a léptetőmotorok technológiai fejlődését. Ennek az az oka, hogy a léptetőmotorokat bármely digitális vezérlővel könnyedén összekapcsolhatjuk. Számos területen alkalmazzák őket, például nyomtatókban, folyamatvezérlő rendszerekben és robotokban [1].

2 Léptetőmotorok jellemzése

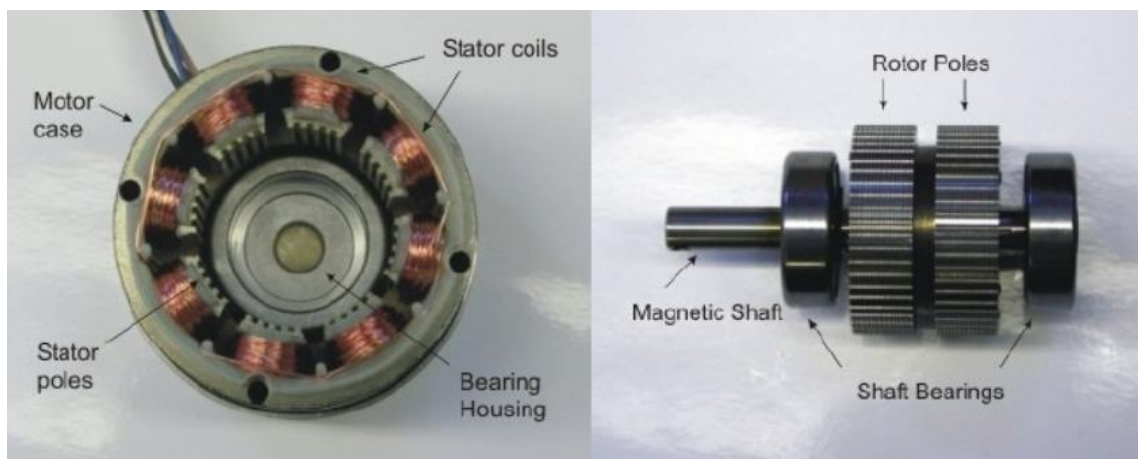
2.1 A léptetőmotorok rövid áttekintése

A léptetőmotor, mint általában minden más motor, álló (stator)- és forgórészből (rotor) áll. A léptetőmotor abban tér el a többi motortól, hogy csak az állórésze tekercselt. Szénkefét egyáltalán nem tartalmaz, így kopó alkatrész hiányában megbízható, élettartama nagyrészt a csapágy minőségétől függ.

A motor úgy hajtható meg, hogy az állórész tekercseit egymás után impulzusokkal vezéreljük. Ha a gerjesztést átkapcsoljuk az egyik tekercsről a mellette levőre, akkor a rotor egy bizonyos szögben elfordul. Ennek az elfordulásnak a nagyságát nevezzük lépésszögnek.

Összességében az mondható el, hogy a léptetőmotor egy elektromechanikai eszköz, mely a kapott elektromos impulzusokat alakítja át a rotor pozícionálásával. Ezeknek az impulzusoknak a frekvenciájával befolyásolható a léptetőmotor fordulatszáma. Ez azt jelenti, hogy a bemeneti jelek frekvenciájának változtatásával a léptetőmotor kívánt fordulatszáma elérhető.

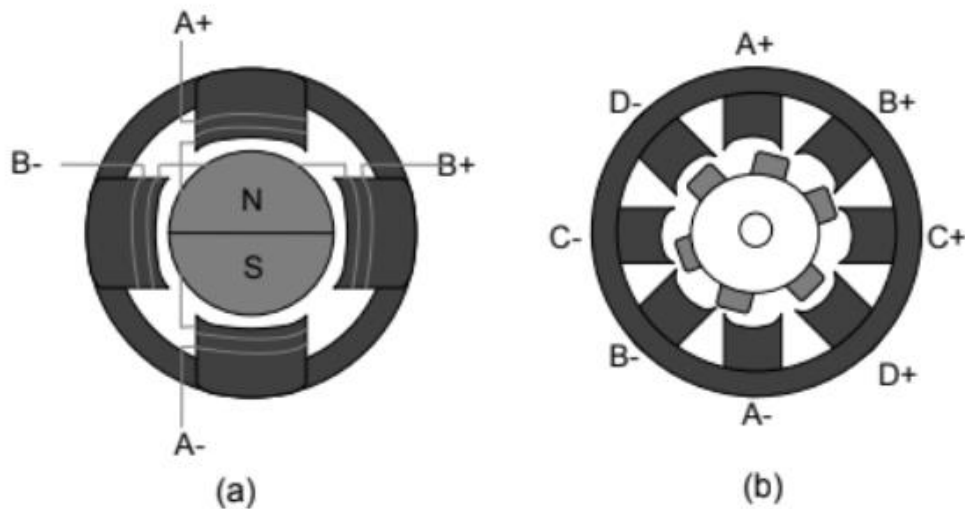
A léptetőmotorokat impulzusgenerátorok vagy mikrokontrollerek vezérlik [2]. Az egyes mozgások annyira pontosan beállíthatóak, hogy visszajelző használatára sincsen szükség [3]. Ezért tudnak nyílt hurkú rendszerben is hatékonyan működni. Ennek előnye, hogy a generált hibák nem adódnak össze, amitől a lépéspontosság javul.



1. ábra: Egy léptetőmotor belső szerkezete [4]

A fenti ábra egy léptetőmotor belső szerkezetének általános kialakítását mutatja meg. A bal oldali képen a motor belseje látható, oldalt az állórész tekercseivel, melyek kiálló pólusokon helyezkednek el. A jobb oldali képen a rotor látható a két ellentétes polaritású fogazott résszel, melyek a tengelyre vannak rápréselve a közöttük lévő állandó mágnessel. Ennek a mágnesnek a polarizációja miatt az egyik fogazott rész északi, míg a másik déli pólust kap. A tengelyen még két csapágy látható, melyeknek szerepe a pontosabb forgó mozgás biztosítása.

A léptetőmotorok árát nagymértékben a méretük, felépítésük, és a pontosságuk határozza meg.



2. ábra: Léptetőmotor álló és forgórésze [5]

- (a) Kétfázisú állandó mágneses (PM) léptetőmotor, és
- (b) Négyfázisú változó reluktanciájú (VR) léptetőmotor

2.2 Léptetőmotorok típusai

A léptetőmotorok működési elve nagyon egyszerű, mivel a forgórész pozíciója rögzített szögfordulattal állítható be úgy, hogy a gerjesztést a tekercsek között kapcsolgatják. Ez az alapkonceptiója mindegyik léptetőmotor-fajtának. A felépítésük és működési elvük alapján azonban a léptetőmotoroknak három különböző típusa van.

Ezek a:

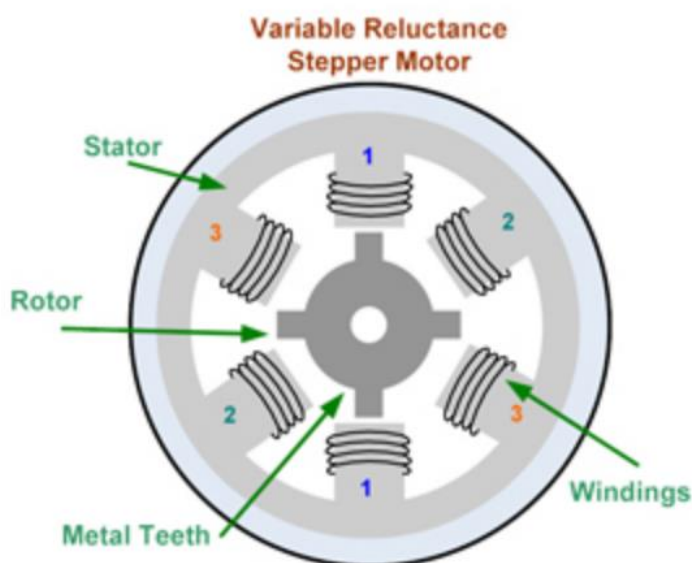
- i. Változó reluktanciájú léptetőmotorok
- ii. Állandómágneses léptetőmotorok
- iii. Hibrid léptetőmotorok

2.2.1 Változó reluktanciájú léptetőmotorok

A változó reluktanciájú léptetőmotorok a legegyszerűbb felépítésű léptetőmotorok, fogoztak mind az álló-, mind a forgórészen. Lényeges tulajdonságuk, hogy tekercselés csak az állórészükön található. Állandó mágneset nem tartalmaznak, a forgórészük lágy mágneses anyagból készül [6].

A működési elve a következőképpen határozható meg: működés közben elektromágnesek jönnek létre, melyek a hozzájuk legközelebb eső forgórész-fogakat magukhoz húzzák, így idézve elő a forgatást [7]. Megfelelő kialakítással tetszőleges irányba forgathatóvá válik a motor.

A következő ábra egy változó reluktanciájú léptetőmotort mutat be:



3. ábra: Változó reluktanciájú léptetőmotor [8]

A 3. ábrán a forgórész négy foga látható, illetve a háromfázisú állórész. Változó reluktanciájú léptetőmotoroknál minimum három fázisra van szükség ahhoz, hogy szimmetrikus felépítésű gépet tudjunk létrehozni [7].

Két fázis esetén csak egy meghatározott irányba képes a motor forgó mozgást végezni, mégpedig akkor, ha a forgórésze aszimmetrikus felépítésű. Fontos, hogy a tengelyre ható erők eredője zérus legyen, illetve, hogy mágneselesen szimmetrikus elrendezést kapjunk. Ez úgy érhető el, hogy minden fázishoz páros számú fog tartozik [7].

A léptetőmotor használatához ismernünk kell a teljes körbeforduláshoz szükséges lépésszámot (S). Ez a következőképpen számítható:

$$S = Z_r \cdot m^* , \quad (1)$$

Ahol

Z_r : a forgórész fogszám,

m^* : a fázisszám.

Ha a teljes körbefordulást (360°) elosztjuk a lépésszámmal, akkor megkapjuk az egy lépésének a szögét. Ezt jelölje: α .

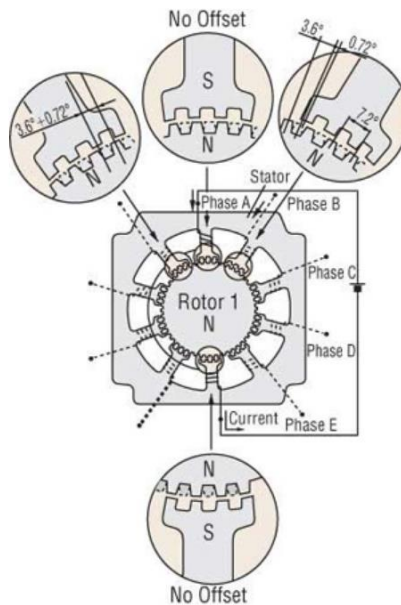
$$\alpha = \frac{360^\circ}{S} , \quad (2)$$

A 3. ábrán szereplő léptetőmotor esetén:

$$S = 12 , \quad (3)$$

$$\alpha = 30^\circ , \quad (4)$$

Alapvetően a léptetőmotorok alacsonyabb lépésszöggel rendelkeznek, mint 30° . Ahhoz, hogy a pozicionálás pontosabb legyen, minél kisebb lépésszög szükséges. Számos megoldás létezik a lépésszám növelésére, vagyis a lépésszög csökkentésére. Az egyik legegyszerűbb módszer a forgórész fogszámának növelése. Nem szabad túl kicsi fogakat alkalmazni, mivel azok negatívan befolyásolhatják a motor nyomatékát. Erre a problémára kínál megoldást a fogsokszorozás, melynek lényege, hogy nem csak a forgórészen, hanem az állórészen is megsokszorozzák a fogakat. Így a nyomaték is (részben) megőrizhető. A fogsokszorozást illusztrálja a 4. ábra.



4. ábra: Fogsokszorozás [9]

2.2.1.1 A változó reluktanciájú léptetőmotor fordulatszám [6]

A szinkronmotorok fordulatszám a következő képlet felhasználásával határozható meg:

$$n = \frac{60f_1}{p}, \quad (5)$$

Ahol:

n : a szinkron fordulatszám,

f_1 : az alapösszetevő frekvenciája,

p : a póluspárok száma.

Váltakozó reluktanciájú motorokra hasonló összefüggés írható fel:

$$n = \frac{60f_0}{S} = \frac{60f_1}{Z_r}, \quad (6)$$

Ahol:

n : a motor fordulatszám,

f_0 : a léptetés frekvenciája,

f_1 : az állórész fázisainak a táplálási frekvenciája,

S: egy teljes kör megtételéhez szükséges lépések száma,

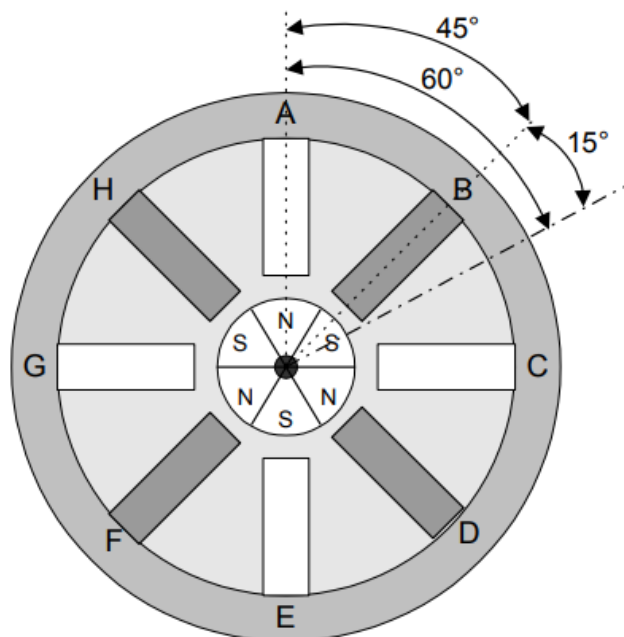
Z_r : a forgórész fogainak száma.

Ezen egyenletekből az a következtetés vonható le, hogy a változó reluktanciájú léptető motorokat szinkron motoroknak tekinthetjük.

2.2.2 Állandómágneses léptetőmotorok

Az állandómágneses léptetőmotorok felépítése csupán részben különbözik a változó reluktanciájú léptetőmotorok felépítésétől. A két típusú motor állórésze teljes mértékben megegyezik (abban az esetben, ha nem beszélünk fogsokszorozásról). Mindkettő típusú motor esetében a kiálló pólusokon helyezkednek el a tekercsek. A forgórészük viszont különböző. Állandómágneses léptetőmotoroknál állandómágneses forgórészt alkalmaznak. A többpólusú mágnes miatt figyelembe kell venni, hogy a tekercseken milyen irányú áram folyik. Az állandómágneses léptetőmotorok nagyobb tehetetlenséggel, és ennél fogva lassabb gyorsulással rendelkeznek, mint a változó reluktanciájú léptetőmotorok [10].

Egy tipikus állórész-elrendezést mutat be a következő ábra:



5. ábra: Állandómágneses léptetőmotor tipikus állórész-elrendezése [11]

A rotor 6 pólusú, az egyes pólusok 60° -ra helyezkednek el egymástól. Az állórész mind a 8 pólusa a mellette levővel 45° -os szöget zár be. Jelen esetben 2 tekercs van, az egyik az A, C, E, és G pólusokat köti össze, míg a másik a B, D, F, és H-t. Tételezzük fel, hogy egy bizonyos irányban áram folyik az első tekercsen, és a polaritások:

A és C: déli pólus

E és G: északi pólus

A forgórész elrendezése ennek megfelelően az ábráról leolvasható. Ahhoz, hogy a forgórész például 15° -ban elmozduljon az óramutató járásával megegyező irányban, az első tekercs táplálását meg kell szüntetni, és a második számú tekercset kell gerjeszteni. B, D, F és H a második tekercshez kapcsolódnak a következő módon:

B és D: déli pólus

F és H: északi pólus

Ezután ismét az első tekercset kell gerjeszteni fordított polaritással, ezáltal A és C polaritása északira változik, E és G pólusoknak pedig délire. Ebből kifolyólag a forgórész 15° -os lépést tesz az óra járásával megegyező irányba. Ebben a konstrukcióban 15° -os lépésszög érhető el. A lépésszám azonban könnyen növelhető ún. körmös pólusú kialakítással, vagy finomlépéses üzem alkalmazásával [7].

A fent ismertetett motort állandómágneses léptetőmotorok nevezzük, mivel a forgórésze állandómágnes. Jellemzőjük az olcsó ár, és a $7,5$ - 15 fokig terjedő lépésszög [11].

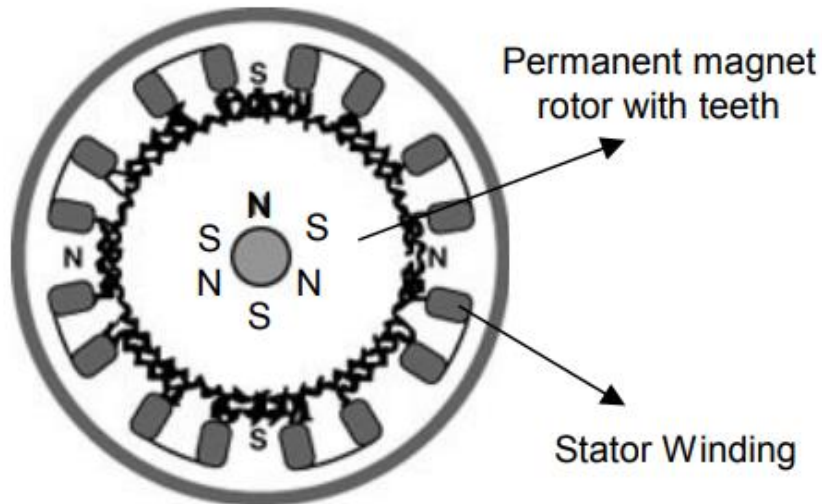
2.2.3 Hibrid léptetőmotorok

A hibrid léptetőmotorok speciális kialakításúak. Azért nevezzük őket így, mert egyesítik a változó reluktanciájú és az állandómágneses léptetőmotorok pozitív tulajdonságait.

Jellemző rájuk a gyors, pontos mozgás relatív nagy nyomaték-megmaradással. Felbontásuk nagy, a legszélesebb körben alkalmazott kétfázisú bipoláris motorok $1,8^\circ$ -os lépésszöggel rendelkeznek. A legnagyobb nyomatékot körülbelül 1000

lépés/másodpercnél adja le. Természetesen léteznek ettől eltérő fordulatszámú, speciális kialakítású hibrid léptetőmotorok is.

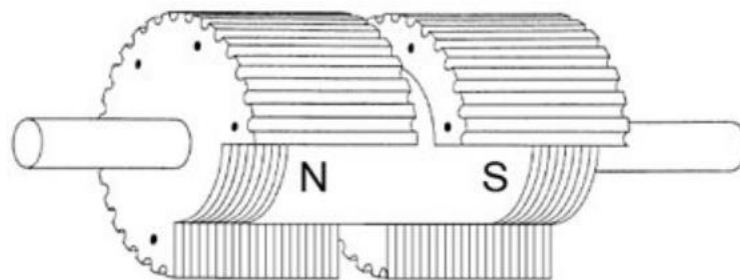
A 6. ábrán egy hibrid léptetőmotor keresztmetszeti képe látható.



6. ábra: Hibrid léptetőmotor keresztmetszeti képe [11]

Az állórészen 8 kiálló pólus van, fázisonként négy. Látható, hogy ezek több fogból álló fogcsoportot tartalmaznak. A rotor fogszáma is soknak mondható, így a lépésszám is magas lesz.

Az alábbiakban részletesen áttekintem a forgórész felépítését (7. ábra). A tengely antimágneses, melyre egy állandómágnes gyűrű van húzva. A gyűrűre húznak még két fogazott ferromágneses sapkát. A bal oldali sapka északi, a jobb oldali sapka déli pólusú lesz az állandómágnes miatt. A két sapka egymáshoz képest fél fognyival el van forgatva az azonos irányú nyomaték hatása végett.



7. ábra: Hibrid léptetőmotor forgórésze [12]

Az iparban és kisteljesítményű hajtások esetén a hibrid léptetőmotorok a legelterjedtebbek.

2.3 Léptetőmotorok előnyei és hátrányai

A léptetőmotor alkalmazásának előnyei:

- A bemeneti impulzus arányos a motor forgásszögével.
- Teljes nyomatékkal rendelkeznek nyugalmi pozícióban.
- Pontos, és megismételhető pozicionálás. A jobb léptetőmotoroknál 3-5% hiba mutatkozik egy lépésnél, viszont ez a hiba nem halmozódik.
- Nagyon megbízhatóak, mivel nem tartalmazznak szénkeféket. A motor élettartama ezért a csapágyaktól függ csak.
- A digitális impulzusokra adott válasz lehetőséget nyújt nyílt hurkú vezérlésre, melynek következtében olcsóbban és egyszerűbben vezérelhetjük őket.
- A fordulatszámot nagyon széles skálán tudjuk változtatni, mivel ezek a bemeneti impulzusok frekvenciájával arányosak.
- Nincsenek komoly stabilitási problémák, még nyílt hurkú vezérlés esetén sem.

A léptetőmotorok hátrányai:

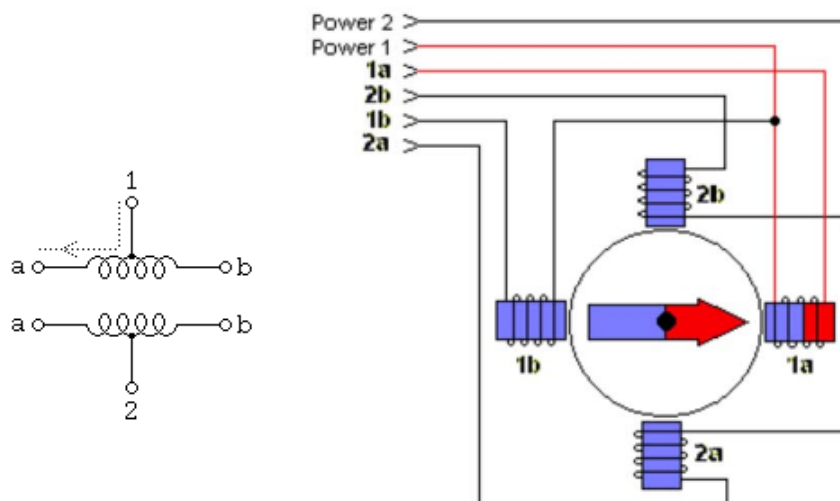
- Ha nem megfelelő a vezérlés, akkor rezonancia fordulhat elő.
- Rendkívül nagy fordulatszámmal nem könnyű működtetni.
- Nagyobb az áramfelvétele, mint egy DC-motornak.
- Kisebb nyomaték-kapacitással rendelkeznek, mint egy DC-motor.
- A többi motornál megszokott néhány ezer fordulat/perc fordulatszámnál sem működnek túl jól.

2.4 Fázistekercsek kapcsolásának lehetőségei

A fázistekercsek kapcsolásán azt kell érteni, hogy a tekercsek motoron belül hogyan vannak szétosztva, illetve hogyan vannak motoron kívülre kivezelve. A léptetőmotorok tekercselése kétféle lehet áramirány szempontjából. Megkülönböztetünk unipoláris és bipoláris léptetőmotorokat. Ebben a fejezetben ezt a két tekercselési módot fogom bemutatni.

2.4.1 Unipoláris

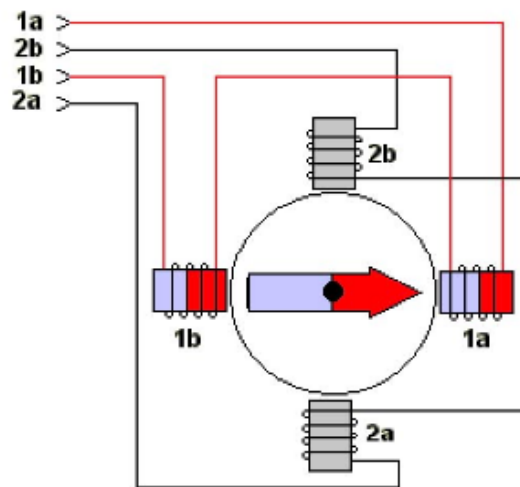
Az unipoláris léptetőmotoroknál a tekercsekben üzem közben az áram iránya változatlan marad. A tekercsüket csak ki- és bekapcsoljuk. Egy ilyen léptetőmotor leegyszerűsített megvalósítását mutatja a 10. ábra. Az összetartozó féltekercsek egymással szemben helyezkednek el, egymással össze vannak kötve. A középső nyíl jelképezi az állandó mágnesként viselkedő forgórészt. Ha a vezérlőjelek (1a, 2b, 1b, 2a) valamelyikét alacsony feszültségszintre (földre) állítjuk, akkor mindig csak abban az egy tekercsben indul meg az áram, vagyis elektromágnesként működik. Ezek alapján egy teljes kör megtételéhez az alábbi sorrendben kell a jeleket földpotenciálra kapcsolni: 1a→2a→1b→2b→1a. Ennek értelmében az óramutató járásával megegyező irányú rotorforgást kapunk. Természetesen a tekercsek kapcsolgatásának iránya egyértelműen meghatározza a forgásirányt. Jelen esetben ez az egészlépéses üzemet valósítja meg.



8. ábra: Unipoláris léptetőmotor működési elve [13][14]

2.4.2 Bipoláris

A 9. ábra feltüntetett bipoláris tekercselésnél az azonos fázishoz tartozó tekercsek egymással szemben helyezkednek el és csak a két tekercs két-két végpontja van kivezetve. Ennél a megvalósításnál az áramirány működés közben változik a tekercsekben.



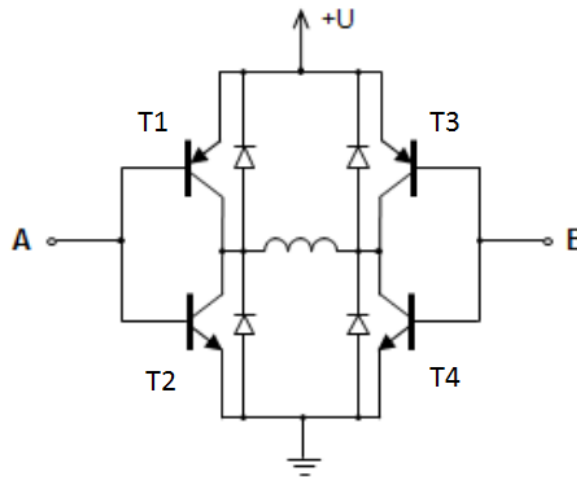
9. ábra: Bipoláris léptetőmotor működési elve [14]

Az éppen aktuálisan gerjesztett féltekercs és a vele szemben lévő párja is mindig gerjesztett állapotban van. Ezért ez a kapcsolási mód nagyobb nyomatékot eredményez, mint unipoláris esetben, mivel a forgórész egymással szemben lévő fogaira erő hat, és ezek az erők összeadódnak. Ez a tulajdonság az, amelynek köszönhetően a gyakorlatban többnyire bipoláris megoldásokat használatosak.

2.4.2.1 Bipoláris léptetőmotorok vezérlése

Ahogyan dolgozatomban már korábban említettem, a bipoláris léptetőmotor tekercsein mindkét irányban folyhat az áram. Ennek a kivitelezésére szolgál a 10. ábra bemutatott H-híd. Az ábrán a H-híd ellenállások nélküli vázlata látható. Védődiódák alkalmazására szükség van a kapcsolásban, nehogy irányváltáskor egy feszültségimpulzus tönkretegy a tranzisztorokat. A híd segítségével egyetlen tekercsen tudjuk változtatni az áram irányát. Ha az "A" jelszint alacsony, a "B" pedig magas, akkor T1 és T4 tranzisztor vezet, a másik kettő zárt állapotban marad. Ekkor a tekercsen balról

jobbra folyik az áram. Ha a két jelszintet megcseréljük, akkor T2 és T3 vezet, az áram iránya az előzőhöz képest ellentétes lesz.



10. ábra: H-híd

Bipoláris léptetőmotorok vezérlésére számos IC kapható, melyben benne vannak a kész H-hidak, nem szükséges ennek megépítésével foglalkozni.

2.5 Meghajtási üzemmódok

A következő fejezetekben részletesen bemutatásra kerülnek a különböző meghajtási üzemmódok és azok jellemzői.

2.5.1 Egészlépéses üzemmód (Full Step)

Az egészlépéses üzemmód már a nevéből adódóan is utal arra, hogy ez a hajtásmód teljes lépéseket hajt végre.

Az egészlépéses üzemmódnak két típusát különböztetjük meg :

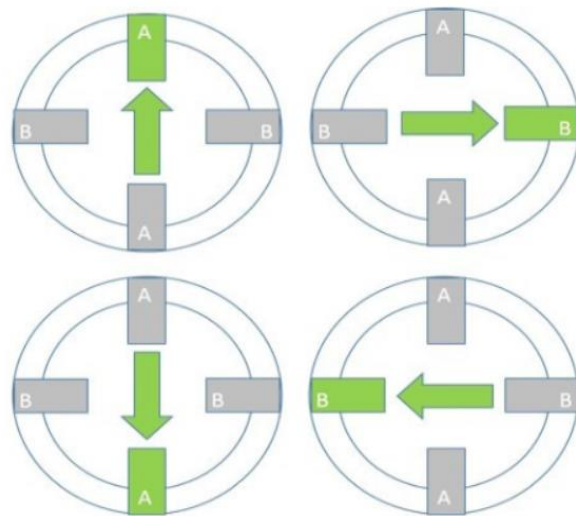
- hullámhajtás
- két fázisú gerjesztés

Hullámhajtásnál a tekercseket sorban gerjesztjük, viszont fontos megjegyezni, hogy egyszerre csak egy tekercs lehet gerjesztett. Ez egy nagyon egyszerű hajtási mód, azonban a nyomatéka csekély. Ennek oka, hogy egyszerre csak egy tekercs áll gerjesztés

alatt, lényegében az egész munkát ez végzi. Annak a sorrendjét, amiben a tekercseket gerjesztjük a lenti táblázat adja meg, majd pedig a 13. ábra szemlélteti.

1. táblázat: Hullámhajtás esetén a tekercsek gerjesztése

| Lépés | A | B | A' | B' |
|-------|---|---|----|----|
| 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 |
| 3 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 0 | 1 |



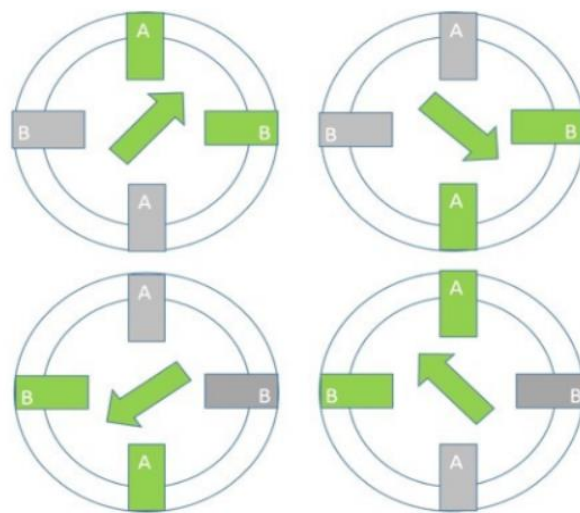
11. ábra: Lépéssorrend teljes lépés üzemmódban [15]

Egészlépéses üzemmód megvalósítható két fázis egyszerre történő gerjesztésével is. Az előző móddal szemben előnye a nagyobb nyomaték, viszont magasabb az energiaigénye is.

2. táblázat: Egészlépéses üzemmód két fázis egyidejű gerjesztésével

| Lépés | A | B | A' | B' |
|-------|---|---|----|----|
| 1 | 1 | 1 | 0 | 0 |
| 2 | 0 | 1 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 |
| 4 | 0 | 0 | 0 | 1 |

Mivel mindig egymás melletti tekercsek vannak gerjesztve, a forgórész pólusa ezek közé fog beállni. A hullámhajtásnál megismert pozíciókhoz képest pontosan egy fél fordulat az eltérése. A forgórész pozícióját a 12. ábra szemlélteti.



12. ábra: Lépéssorrend két fázis egyidejű gerjesztésénél [15]

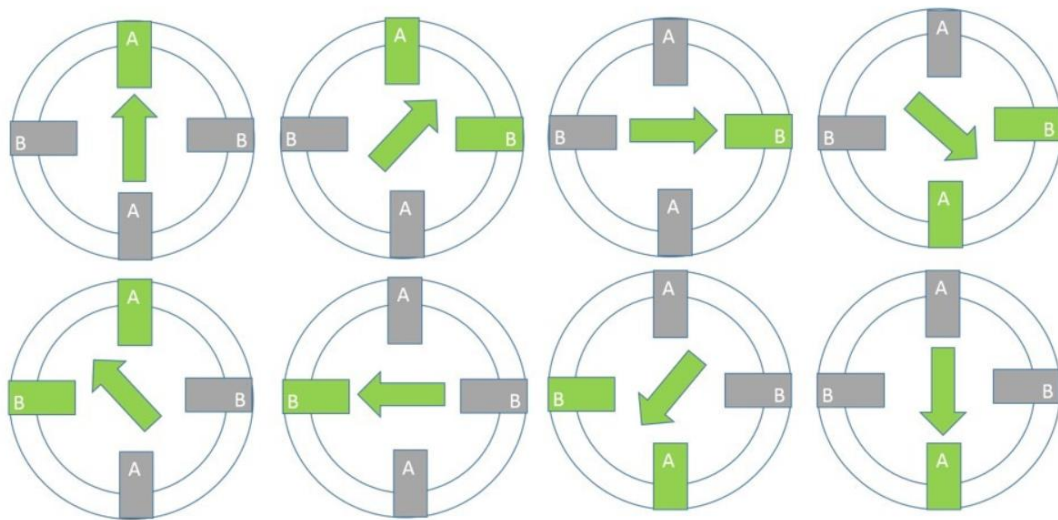
2.5.2 Féllépéses üzemmód (Half Step)

Az egészlépéses üzemmódnál bemutatott két típus kombinációja a féllépéses üzemmód. Felváltva gerjesztünk egyszerre 2 tekercset, majd egy tekercset. Ebben az üzemmódban a lépések száma a duplája, mint egészlépéses módban. Az alábbi táblázat mutatja a tekercsek gerjesztésének sorrendjét.

3. táblázat: Tekercsek gerjesztésének sorrendje féllépéses üzemmódban

| Lépés | A | B | A' | B' |
|-------|---|---|----|----|
| 1 | 1 | 0 | 0 | 0 |
| 2 | 1 | 1 | 0 | 0 |
| 3 | 0 | 1 | 0 | 0 |
| 4 | 0 | 1 | 1 | 0 |
| 5 | 0 | 0 | 1 | 0 |
| 6 | 0 | 0 | 1 | 1 |
| 7 | 0 | 0 | 0 | 1 |
| 8 | 0 | 0 | 0 | 0 |

A fenti táblázatban bemutatott gerjesztések hatására a forgórész elmozdulásának sorrendje:

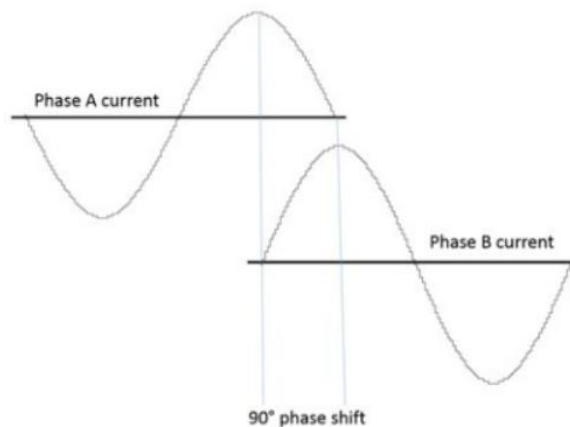


13. ábra: Féllépéses üzemmódban a forgórész pozíciója [15]

2.5.3 Mikrolépéses üzemmód (Micro Step)

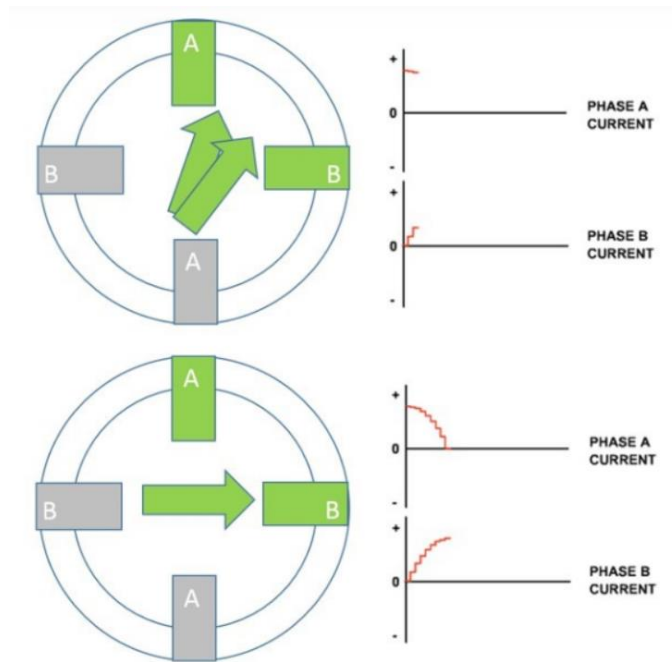
Ahogy az előző pontban tárgyalt féllépéses üzemmódnál, ebben az üzemmódban is felváltva gerjesztünk egy, vagy két tekercset. A különbség abban áll, hogy a két tekercsen a gerjesztés nem lesz egyenlő. Evvel a módszerrel a mechanikai zaj- és rezonanciaproblémák jelentős mértékben csökkenthetőek. Rendkívül nagy előnye, ami miatt népszerű a használata, hogy pontos, kisebb felbontású léptetést tesz lehetővé.

A fázisonkénti áramok jelalakjai:



14. ábra: fázisok árama mikroléptetés esetén [15]

Mikor az egyik fázison az áram maximális, vele egy időben a másik fázis árama nulla. Megfigyelhető még, hogy az áramok jelalakjai megközelítik az ideális szinuszjelet. Minél nagyobb felbontást alkalmazunk, annál jobban. A felbontás lehet például 1/50, 1/100, vagy esetleg 1/256. Utóbbi esetben már nagyon sima, szinte folytonosnak érzékelhető léptetést tudunk megvalósítani. A szinuszjelek szinkronban vannak, ezért zökkenőmentes az átmenet egy tekercselésről a másikra.



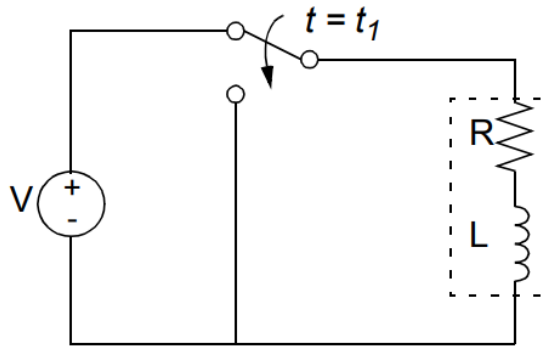
15. ábra: mikroléptetés, fázisonkénti áramokkal [15]

2.6 Áramkorlátozás

Ha a léptetőmotorra a névlegesnél nagyobb feszültséget kapcsolunk, akkor a tekercseinek árama gyorsabban fog növekedni. Ennek következtében magasabb fordulatszám és nyomaték hozható ki a motorból. Ebben a fejezetben az áramkorlátozás módszerét fejtem ki és bemutatom azt, hogy ez a művelet miképp befolyásolja a teljesítménynövekedést [16].

2.6.1 Az alapok

Az alacsony fordulatszám miatt hanyagoljuk el a pólusfeszültséget. Ebben az esetben a léptetőmotor tekercseit ugyanúgy, mint egy sima tekercset, le tudunk modellezni egy L induktivitással és egy R ellenállással.



16. ábra: Rezisztív-induktív tekercsmo­dell [16]

Az átfolyó áram nagyságát Ohm törvényével számolhatjuk ki.

$$I = \frac{U}{R}, \quad (7)$$

Ha időben váltakozó áram folyik a tekercsen, akkor az időben váltakozó mágneses mezőt hoz létre. Ez a mező minden pillanatban olyan feszültséget indukál, mely Lenz törvénye értelmében ellentétes a generátor pillanatnyi feszültségével, vagyis az áram változásával.

$$U_i(t) = L \cdot \frac{dI}{dt}, \quad (8)$$

A 16-as ábrán lévő kapcsolás egy tekercsmo­dell vázlatos kapcsolási rajzát mutatja be. A $t=0$ időpontban az átfolyó áram értéke nulla, ez a feszültségforrás bekapcsolásának pillanata. Majd $t=t_1$ időpillanatban, az áram maximális értékénél a feszültségforrás újra kikapcsolt állapotba kerül.

A tekercsben az áram exponenciálisan növekszik a maximuma eléréséig. Az időfüggvénye [16]:

$$I(t) = \frac{U}{R} \cdot \left(1 - e^{-\frac{R}{L}t} \right), \quad (9)$$

A maximális átfolyó áramot Ohm-törvénnyel határozhatjuk meg:

$$I_{MAX} = \frac{U}{R}, \quad (10)$$

A bekapcsolás időpillanatában az áramnak a meredekségét is figyelembe kell venni, ehhez a fenti egyenlet mindkét oldalának venni kell az idő szerinti első deriváltját.

$$\frac{dI(t)}{dt} = \left(\frac{U}{R} - \frac{U}{R} \cdot e^{-\frac{R}{L}t} \right) \frac{d}{dt}, \quad (11)$$

$$\frac{dI(t)}{dt} = \frac{U}{L} \cdot e^{-\frac{R}{L}t}, \quad (12)$$

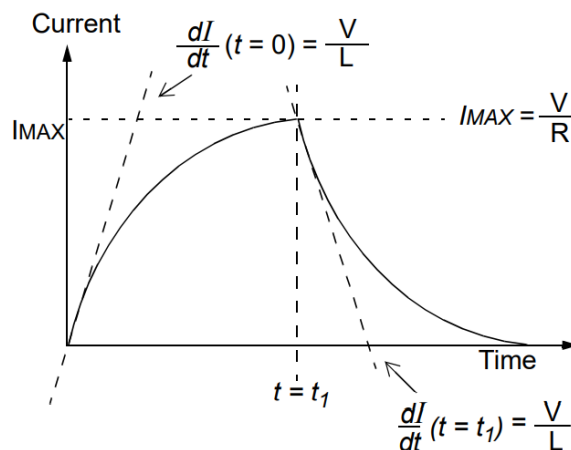
A $t=0$ esetben:

$$\frac{dI(t)}{dt} = \frac{U}{L}, \quad (13)$$

A következő vizsgálandó tartomány a t_1 , vagyis a kikapcsolás utáni szakasz. Az áramnak a csökkenése a növekedéshez hasonlóan exponenciális jellegű.

$$I(t \geq t_1) = \frac{U}{R} \cdot e^{-\frac{R}{L}t}, \quad (14)$$

Az összefüggések megértését segíti a 17. ábra.



17. ábra: Tekercs áramának változása az idő függvényében [16]

A fenti képletekből látható, hogy a tekercs áramának időbeli változása függ az ellenállás, az induktivitás és a feszültség nagyságától. Az ellenállás és az induktivitás értékei adottak a motor által, tehát csak a rákapcsolt feszültség változtatható meg. Minél nagyobb a feszültség, annál gyorsabban emelkedik az áram a maximális szint fölé. Ez túlmelegedést idéz elő, amely következtében a motor tönkremenetelét sem szabad kizárni. Szükség van tehát egy olyan megoldásra, ami elősegíti az áram gyors emelkedését a korlátozásával együtt.

2.6.2 Az áramkorlátozás módszerei

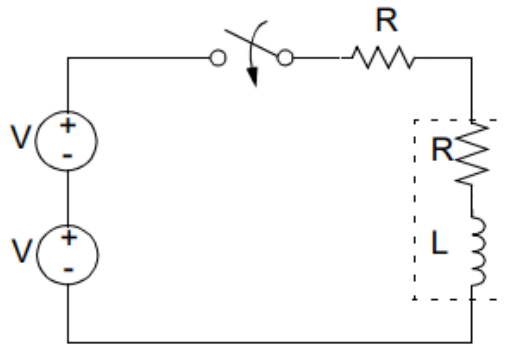
Egy lehetőség az áramkorlátozásra, ha sorosan kötünk egy ellenállást a tekercsmodellünkhöz. Ezt szemlélteti a 18. ábra. Két ugyanolyan ellenállás van sorba kötve egy induktivitással. Ezzel együtt a feszültséget is a duplájára növelhetjük, ezt szimbolizálja a két feszültséggenerátor az ábrán.

Ezek hatására a maximális áram változatlan marad, viszont az induktivitás változatlansága miatt gyorsabban nő az áram, ezáltal előbb éri el a maximális szintet.

$$I_{\text{MAX}} = \frac{2U}{2R} = \frac{U}{R}, \quad (15)$$

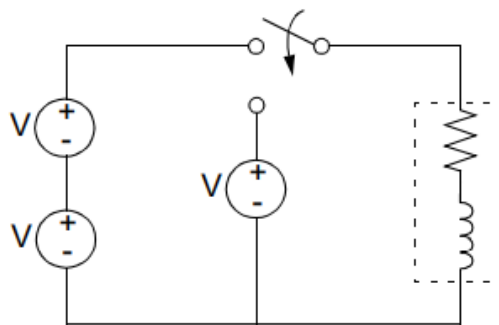
$$\frac{dI}{dt}(t=0) = \frac{2U}{L}, \quad (16)$$

A megnövekedett ellenállásérték miatt a $P=R \cdot I^2$ képlet alapján a teljesítményvesztés is emelkedik. A hozzáadott ellenállás nem korlátozza hatékonyan az áramot, mivel hővé disszipálja. További hátránya, hogy az ellenállás a feszültségtől függően rendkívül nagy, illetve költséges lehet.



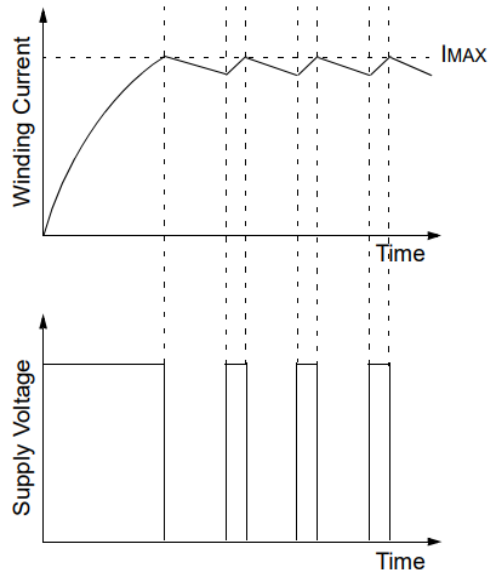
18. ábra: Tekercesmodell egy hozzáadott ellenállással [16]

Az áramkorlátozás egy hatékonyabb módja lehet, ha egy helyett két különböző feszültségforrást alkalmazunk a kapcsolásban. Az egyik feszültsége egyezzen meg a motor névleges feszültségével, a másiké legyen ennél nagyobb. Az alapgondolat az, hogy az áram gyors felfutásának idején nagyobb feszültség szintről van megtáplálva a kapcsolás, majd innen kapcsol át az alacsonyabb feszültség szintű táplálásra. Ez az összeállítás kiküszöböli a sorosan kapcsolt ellenállás hátrányait, gyorsabb áramnövekedést is eredményez. Ennek ellenére a két feszültségforrás alkalmazása nem praktikus a gyakorlatban.



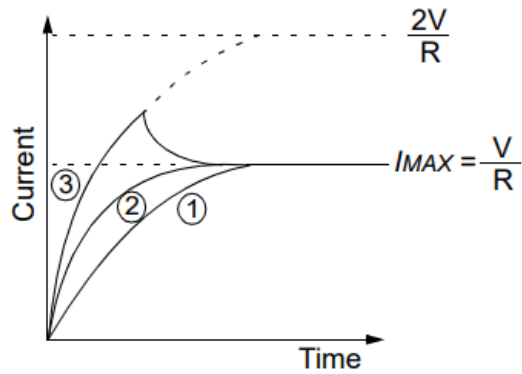
19. ábra: Tekercesmodell két különböző feszültségforrással [16]

A fent leírt módszer hatékonyságát elérhetjük mindössze egy feszültségforrással. Erre a feszültségforrásra a motor névleges feszültségénél jóval nagyobb feszültségűt kell használni. A motor tekercseire így a névleges feszültségnél nagyobb impulzusokat tudunk adni úgy, hogy a tekercsek még elbírják a bekapcsolás pillanatáig. A magas feszültségnek köszönhetően az áramnövekedés meredekebb lesz. Amint az áram eléri az általunk meghatározott értéket, a feszültséget lekapcsoljuk. Így az áram alakja a felfutásban is elég gyors, és a maximum körüli részen is közelíti az ideális állapotot.



20. ábra: Feszültségimpulzusok hatására az áram görbéje [16]

A következő ábrán a három, feljebb kifejtett módszer összehasonlítása látható:



21. ábra: A három ismertetett módszer különböző áramfelfutásai [16]

Ahol az 1-essel jelölt eset az $I(t, U, R)$ függvény, a 2-es az $I(t, 2U, 2R)$, a 3-as pedig az $I(t, 2U, R)$ függvény.

3 Léptetőmotor vezérlésének megvalósítása

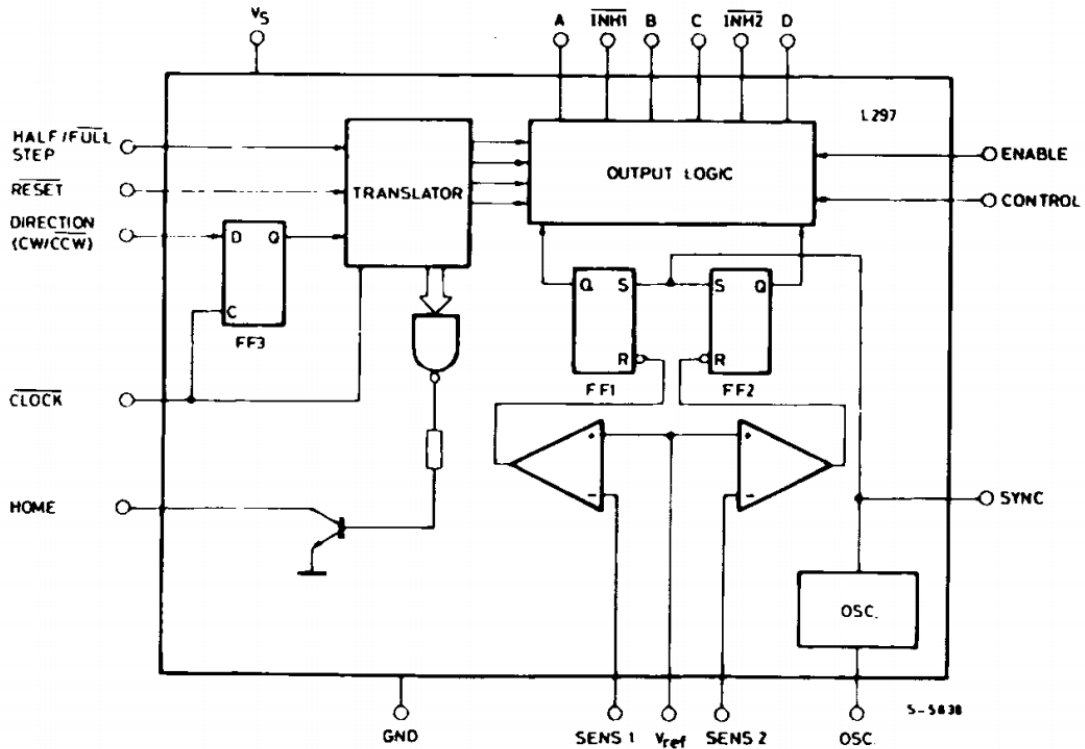
Ha saját vezérlőt szeretnénk megvalósítani, akkor a léptetőmotorok optimális vezérléséhez szükség van arra, hogy a tekercs áramait megfelelően tudjuk beállítani. Ebben a fejezetben evvel fogok foglalkozni, illetve egy vezérlő megvalósításához szükséges eszközöket fogom bemutatni.

3.1 Léptetőmotor vezérlő IC: L297

Az L297-es az STMicroelectronic terméke, széles körben alkalmazzák az L298-cal együtt léptetőmotorok hajtására. Az L297-es négyfázisú jeleket generál kétfázisú bipoláris, vagy négyfázisú unipoláris motorokhoz, melyek így hullámhajtásos, egészlépéses, vagy féllépéses üzemben képesek működni. Amit a készülék mindenképpen igényel, az az órajel (\overline{CLOCK}), az irány ($DIRECTION$) és az üzemmód ($HALF / \overline{FULL STEP}$) meghatározása. A SENS1 és SENS2 bemenetre egy-egy kis értékű ellenállást kell kötni, majd utána szabad csak rákapcsolni a motor fázistekercseire. Ezekon az ellenállásokon a tekercs áramával arányos feszültség esik, amit a SENS bemenetek érzékelnek, majd az érkező feszültségértékeket egy-egy komparátor összehasonlítja a referencifeszültséggel. Ha a mért feszültség nagyobb, mint a referencifeszültség, akkor az RS flip-flop törlődik, kimenete jelzi a kimeneti logika felé, hogy az aktív fázisok áramellátását szakítsa meg. Miután a tekercsáram a kívánt érték alá csökken, akkor a flip-flop egy órajel után ismét beáll, kimenete elküldi a jelet, hogy újra lehet aktiválni a fázisokat [17].

Az áramkorlátozást csak arra használjuk, hogy a tekercsáram gyorsan növekedjen anélkül, hogy átlépné a maximális értéket. Mikrolépéses üzemmód nem valósítható meg az L297 felhasználásával. Ezenfelül a referencifeszültséget mindig újból be kell állítanunk, ha a motor tulajdonságai megváltoznak.

Az L297-es vázlata a 22. ábra látható.



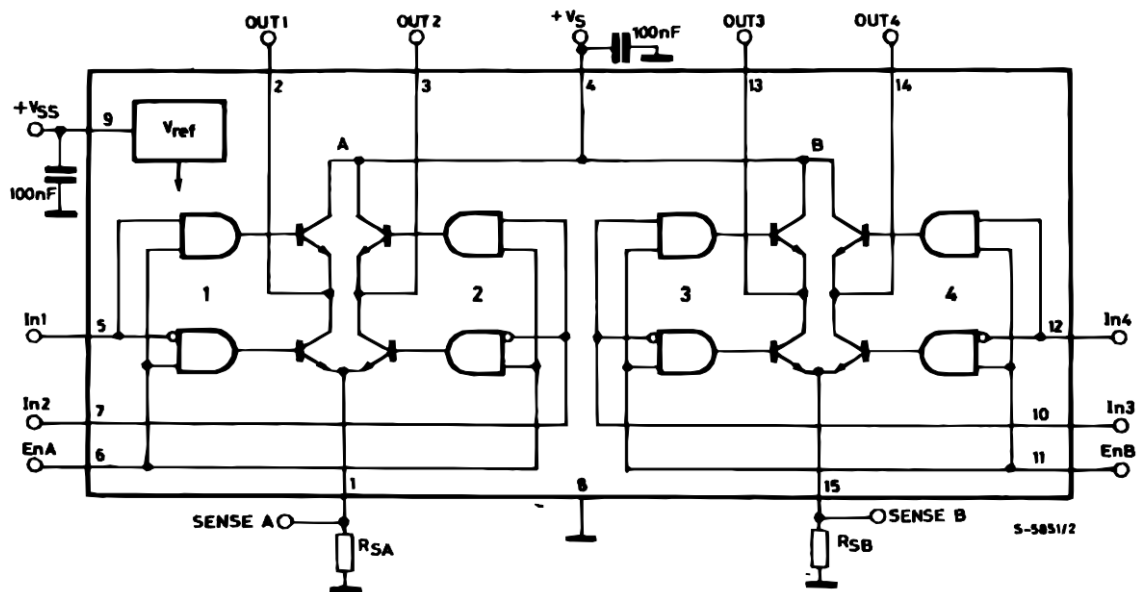
22. ábra: L297-es vázlatrajza [17]

3.2 Kéthidas motorvezérlő IC: L298

Az L297-esnek szüksége van egy meghajtó modulra ahhoz, hogy a léptetőmotort működtetni tudja. Gyakran használják erre a célra az L298-ast, mely teljesen kompatibilis az L297-essel, az áramkorlátozás és a tekercsek vezérlése is egyszerűen megoldható e két eszközzel.

Felépítését tekintve belül két H-híd helyezkedik el, külön földcsatlakozással. Ezáltal lehetőség van minden tekercshez egyedi ellenállást használni, mellyel egyedileg lehet az átfolyó áramot meghatározni és szabályozni. Folyamatosan maximum 2A áram folyhat át az L297-be, ezenkívül akár még 50V tápfeszültséget is lehetővé tesz az L298-as. A két H-híd mellett helyet kapott egy logikai egység, mely megakadályozza, hogy a H-híd egy oldalán mind a két tranzisztor vezessen, így elkerülve a rövidzárat. Rendelkezik még két engedélyező bemenettel, melyek külön egy-egy H-hídat teljesen kikapcsolhatnak [18].

Ha induktív terhelést használunk, akkor fontos a védődiódák alkalmazása.

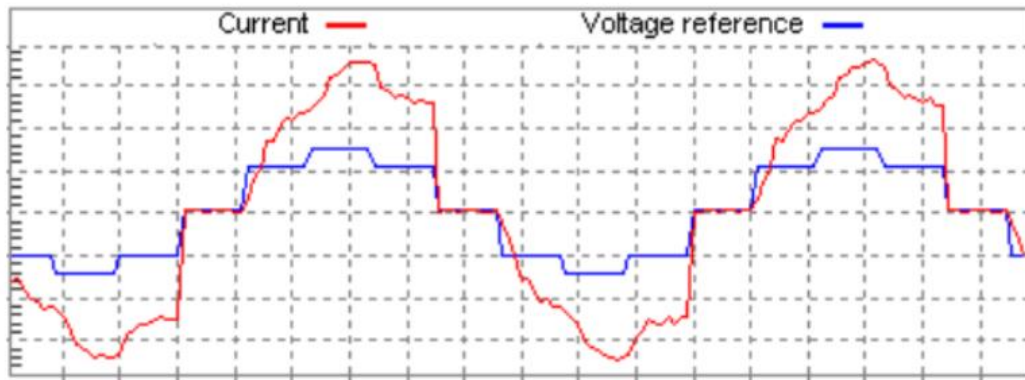


23. ábra: L298-as vázlatrajza [18]

3.3 Vezérlés (nyílt hurkú)

Léptetőmotorok vezérlése megvalósítható nyílt hurkú körben, ahol a kimenetről nincs visszacsatolás a bemenet felé. Az áram vezérlésének egy módja az impulzusszélességmoduláció (PWM) segítségével történő feszültségszintek beállítása. A PWM által létrejövő feszültség Ohm-törvényének értelmében lehetővé teszi a kívánt áram folyását. Ez a feszültség egy négyszögjel, állandó frekvenciával és változó szélességű impulzusokkal.

Mivel a visszacsatolás hiányzik a vezérlőkörből, nem lehet pontosan beállítani a megfelelő áramerősséget. Egy publikációban megjelenő eredmények, [19] alapján az mondható el, hogy néha jelentős eltéréseket is tapasztalhatunk a referenciaértéktől (24. ábra).

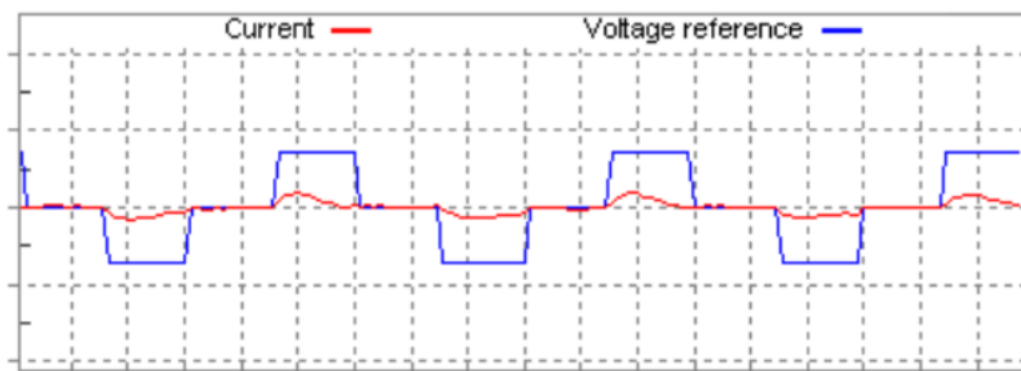


24. ábra: Nyílt hurkú feszültségvezérlés, féllépéses üzemmódban [19]

A fenti problémák nagyobb fordulatszámnál is fennállnak, mivel a lépések közti rövid idő alatt az áram értéke nem tud nagymértékben változni. Minél nagyobb a motor fordulatszáma, annál kisebb az áram amplitúdója (25. ábra).

Ez valamelyest orvosolható, ha a feszültséget növeljük. Ugyanakkor ezt csak egy bizonyos szintig tehetjük meg, nem a végtelenségig. Az eltérések pontatlanságokat okoznak a lépésszögben. Mikrolépéses üzemmódban az is elképzelhető, hogy lépésvesztés fog történni, vagyis teljes egészében kimarad egy lépés.

Ennek következtében a nyílt hurkú feszültségvezérlés sok helyen nem fogadható el, mivel a léptetőmotorokat nagyon pontosan szükséges pozícionálnunk, amihez nagyon precíz szabályozás szükséges. Például egy visszacsatolás, ami ennek a vezérlési kömek a hátrányait kiküszöböli.

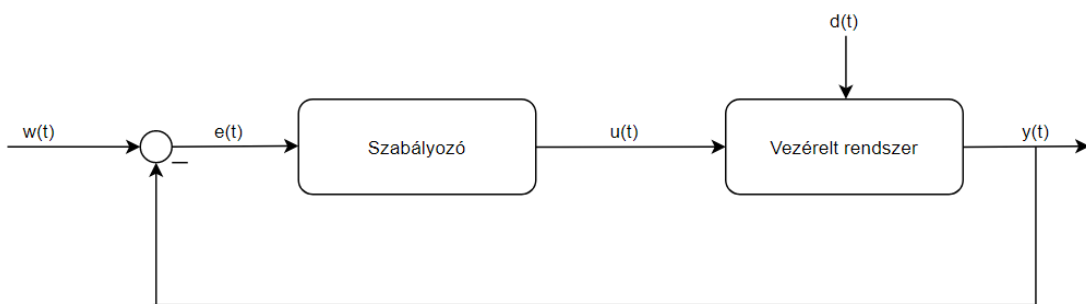


25. ábra: 120 RPM melletti feszültségvezérlés [19]

3.4 Zárt hurkú szabályozás

A zárt hurkú szabályozási kör abban különbözik az előzőekben bemutatott nyílt szabályozási körtől, hogy a kimenetről egy visszacsatolás érkezik a bemenetre. A szabályozó folyamatosan összehasonlítja a visszacsatolt $y(t)$ jel, és a bemenő jelünk $w(t)$ értékeit. A két jel különbségéből képzett $e(t)$ hibajelből meghatározható az új $u(t)$ beavatkozó jel.

A szabályozási kör feladata, hogy az alapjel értékét minél pontosabban és gyorsabban kövesse a szabályozott mennyiség. Vázlata az alábbi ábrán látható:



26. ábra: Szabályozási kör vázlata

Ahol:

- $w(t)$ az alapjel
- $y(t)$ szabályozott jellemző, melyet szenzorok segítségével mérnek, és csatolnak vissza a bemenetre
- $e(t)$ a hibajel
- $u(t)$ a beavatkozó jel
- $d(t)$ a vezérelt rendszerre ható zavarjel(ek).

Nézzük meg, hogy a fenti jelek, illetve a szabályozási kör elemei mik lennének, ha léptetőmotorról van szó.

A léptetőmotor tekercsáramának szabályozását szeretnénk megvalósítani, melyhez maga a tekercs megfeleltethető a vezérelt rendszernek. A zavarjelek, például az ellenállások hőmérsékleteinek változása lehetnek. A tekercs árama a kimenő jel, melyet folyamatosan mérünk és összehasonlítunk a névleges értékkel.

A következő fejezetben egy ilyen zárt hurkú szabályozó implementálását hajtom végre a fent leírtak szerint.

4 Implementálás

Az előző fejezetben bemutatott L297 és L298 IC-párosból csak az L298-at fogom a továbbiakban használni egy mikrokontrollerrel együtt.

Ebben a fejezetben a választott mikrokontroller alaptulajdonságait és funkcióit ismertetem. Mindenféleképpen szükség lesz impulzusszélesség moduláció és analóg-digitális átalakító implementálására is.

Az Atmel 8-bites mikrokontrollerei közül az Atmega8-ra esett a választásom, egyszerűsége és olcsó ára miatt. Továbbá hatalmas segítség az is, hogy egy nagyon egzakt gyári adatlap [20] érhető el a termékről. A fejlesztői környezet ingyenesen hozzáférhető, a kontroller C programozási nyelvben programozható fel. Integrálva van egy hardveres PWM, és AD átalakító is. 1Kbyte-os statikus RAM-mal rendelkezik, melynek elegendőnek kell lennie egy léptetőmotor meghajtásához [20].

4.1 PWM

Az impulzusszélesség-modulációval változó szélességű impulzusokból állandó frekvenciájú négyszögjelet kapunk. Az Atmega8 mikrokontrollernek három PWM kimenete van, ezek közül kettőt fogok használni.

A PWM jelet a TIMER1 modullal állítjuk elő, mely 8, vagy 16 bites módban is működhet. A továbbiakban kizárólag a 8 bites időzítőt fogom használni. PWM jel úgy generálódik, hogy folyamatos számolás jelenik meg a TCNT1 (*Timer/Counter*) regiszterben a maximális értékig (TOP=255), majd amint ezt eléri, kezdi előlről (BOTTOM=0) a számolást. A TCNT1 értékeit folyamatosan összehasonlítja az OCR1x (*Output Compare Register*) értékével. Ha ez a két szám megegyezik, akkor a komparátor kimenetén lévő logikai 1 megjelenik a Waveform Generator bemenetén, és törli az OCR1x (*Output Compare*) értékét majd csak a BOTTOM érték elérésekor kapcsolja vissza (nem-invertáló mód). Ezt nevezik Fast-PWM-nek, mivel itt a számlálót nem csökkentik, ezáltal kétszer akkora frekvencia érhető el.

Az Atmega8 a léptetőmotor meghajtásához 16 MHz frekvenciát biztosít. A maximális PWM frekvenciát 8 bites számlálóregiszter esetén a következő képlettel határozhatjuk meg:

$$f_{\text{PWM_max}} = \frac{16\text{MHz}}{2^8} = \frac{16\text{MHz}}{256} = 62500\text{Hz}, \quad (17)$$

Ahhoz, hogy a Timer1 modullal PWM jelet generálhassunk, szükség van bizonyos bitek beállításra a megfelelő regiszterekben. A TCCR1A (*Timer/Counter Control Register A*) WGM10 bitjét, illetve a TCCR1B (*Timer/Counter Control Register B*) WGM12 bitjét kell 1-be állítani a 8-bites Fast-PWM eléréséhez. A COM1A1 és COM1B1 biteket egy regiszteren belül kell beállítani a nem-invertáló mód aktiválásához. Az előosztót a TCCR1B regiszter CS11 bitjének 1-re állításával implementáltam. Megszakításhoz a TIMSK (*Timer/Counter Interrupt Mask Register*) TOIE1 bitjét kell beállítani. Ez a megszakítás a TCNT1 regiszter túlsordulásával hívódik meg.

```
TCCR1A = (1<<WGM10) | (1<<COM1A1) | (1<<COM1B1);
TCCR1B = (1<<WGM12) | (1<<CS11); //Előosztó: 8

DDRB |= (1<<PB1) | (1<<PB2);
TIMSK = (1<<TOIE1);
```

4.2 AD átalakító

Egy analóg-digitális átalakító az analóg jeleket digitális értékekké konvertálja át. Jelen esetben nincs szükségünk különálló AD átalakítóra, mivel az ATmega mikrokontrollerben integrálva van egy.

Az átalakító a földpotenciál és a referenciafeszültség között működik. A referenciafeszültség legyen 5V, hogy a referenciafeszültség megegyezzen a mikrokontroller tápfeszültségével.

Az integrált AD átalakítónak a legfontosabb tulajdonsága a felbontás és a mintavételi frekvencia („Sample per Second”, mértékegysége: SPS). Maximum 10 bites felbontás érhető el vele, ami azt jelenti, hogy a bemenő analóg jelet $2^{10} = 1024$ egyedi értékkel tudja felbontani. Ez 5V referenciafeszültség mellett 0,004883V felbontásnak felel meg. Az SPS érték az adatlap szerint 15 kSPS.

A 10 bites felbontás miatt az eredmény két regiszterbe kerül, ADCH és ADCL, ezeket össze kell vonni egy darab 16 bites regiszterbe (ADCW).

Ugyan úgy, ahogyan a PWM-nél, az AD átalakító megvalósításánál is bizonyos biteket be kell állítani, hogy a számunkra megfelelően működjenek. Elsőnek a referenciafeszültséget a tápfeszültségre állítottam az ADMUX (*ADC Multiplexer Selection Register*) REFS0 bitjével. Ebben a regiszterben választható ki a multiplexer bemenete is. Az átalakítót az ADCSRA (*ADC Control and Status Register A*) regiszterben az ADEN bittel lehet aktiválni. Az órajel előosztóját 16-os osztásra kell állítani, ezt az ADPS2 bit segítségével lehet megtenni. Az első átalakítás 25 ciklus hosszú, ez rögtön az AD átalakító konfigurálása után elkezdődik. Ekkor ugyanabban a regiszterben automatikusan az ADSC (*ADC Start Conversion*) bitet 1-be állítja a konverter, és ez addig így is marad, amíg az átalakítás zajlik. Ahhoz, hogy ezt deaktiváljuk szükséges definiálni egy ciklust, mely arra vár, hogy az átalakítás befejeződjön.

```
ADMUX = (1<<REFS0) | (1<<MUX0);
ADCSRA = (1<<ADPS2) | (1<<ADEN);
ADCSRA |= (1<<ADSC);
while (ADCSRA & (1<<ADSC)) {}
```

4.3 Szabályozó

A szabályozónk a 26. ábra szerint egy különbségképzésből eredő jelet kap a bemenetére és egy módosított jelet ad ki. A módosításhoz lényeges, hogy minél pontosabb egyenletet határozzunk meg a szabályozó számára [22]. Először a motor adattábláján szereplő paramétereket szükséges ismernünk, utána a processzor elvégzi velük a megfelelő műveleteket.

A lebegőpontos számokkal való matematikai műveletek nagyon lassúak, ezért fixpontos számokat használtam. Ez úgy érhető el, hogy az állandókat megszorozzuk kettő valamilyen hatványával. Az így kapott eredménnyel számol tovább a program. A kettővel való szorzás és osztás bitenkénti eltolást, shiftelést jelent, melyet a processzor rövidebb idő alatt le tud kezelni. A 16-tal való szorzás 4 bit eltolásnak felel meg, ez az állandók pontosságára szolgál. A 255-ös szorzót az indokolja, hogy a PI szabályozó 0 és 1 közötti számot ad a kimenetére, viszont a PWM-nek 0 és 255 közötti számra lenne szüksége. Az

utolsó, 5-ös szorzó az AD átalakító mért értékének Amperra konvertálása miatt kapott helyet az egyenletben.

```
#define KPI 12/((M_ind/M_ell)*tapfesz)
#define PI1 (int32_t)(KPI*(M_ind/M_ell + t_mintavetel/2)*16)*255*5
#define PI2 (int32_t)(KPI*(M_ind/M_ell - t_mintavetel/2)*16)*255*5
```

4.4 Lépésmódok

A mikrokontrollert úgy kell felprogramozni, hogy a 2.5 fejezetben leírt három darab meghajtási üzemmód kezelésére képes legyen. Ebben a fejezetben ezeknek a módoknak a megvalósítását írom le. A lépésmódok között az alábbi paraméter beállításával választhatunk:

```
#define lepesmod 1
// 1 :mikrolépés
// 2 :fél lépés
// 3 :teljes lépés
```

4.4.1 Mikrolépés

Először létrehoztam egy tömböt, melyet 16 diszkrét szinuszértékkel töltöttem fel. Az értékeket ábrázolva egy fél szinuszgörbét ábrázolnának. Majd mindegyik értéket megszorozva a megfelelő maximális áramértékkel és 1024/5-tel olyan értékeket kapunk, melyet az AD átalakító is tárol. Az 512 hozzáadása azért kell, mert 0A-nél 2,5V feszültséget mérünk az AD átalakítón.

```
float szinuszertek[16] = {0, 0.19, 0.37, 0.56, 0.73, 0.83, 0.92,
0.98, 1, 0.98, 0.92, 0.83, 0.73, 0.56, 0.37, 0.19};
//diszkrét szinuszértékek

for (int i=0; i<16; i++)
    MotorErtek[i]=(int16_t)(szinuszertek[i]*maxaram*1024/5)+512
    ;
```

Mikrolépéses módban az A és B nevű tekercsek szinusz értékei 90°-kal el vannak tolvva egymástól.

```

ISR(INT0_vect) { //megszakítás
  #if (lepesmod == 1)
    if (PIND & (1<<PD3)) //D3=1, jobbra forog
      lepes = lepes+1; //lépésindex növelés
    else
      lepes = lepes-1; //lépésindex csökkentés
  #endif

  aramA = MotorErtek[lepes & 15]; //A tekercsáram (sin)
  aramB = MotorErtek[(lepes + 8) & 15]; //B tekercsáram (cos)
}

```

A polaritásról információkat 0x10 maszkolásával kaphatunk. Amint az eredmény nem egyenlő nullával, a polaritás megfordul. Az L298-asban található két híd a PB4, PB5, PD6, PD7 pinekhez kapcsolódik, melyeket kimenetként definiáltam. Ezen hidak tranzisztorainak kapcsolásával szabályozható a tekercsben folyó áram.

```

//kimenet L298-nak
DDRB |= (1<<PB4) | (1<<PB5); //tekercs A
DDRD |= (1<<PD6) | (1<<PD7);

//tekercspolaritás
if ((lepes & 0x10))
  PORTB = (PORTB | (1<<PB4)) & ~(1<<PB5);
else
  PORTB = (PORTB | (1<<PB5)) & ~(1<<PB4);

if (((lepes + 8) & 0x10))
  PORTD = (PORTD | (1<<PD7)) & ~(1<<PD6);
else
  PORTD = (PORTD | (1<<PD6)) & ~(1<<PD7);

```

4.4.2 Teljes és féllépéses mód

A maximális nyomaték eléréséhez nem használható az a szinuszérték, mint ami a mikrolépéses üzemben. Mindkét tekercset a maximális áramerősséggel kell szabályozni. Ezek megváltoztatását szemlélteti az alábbi programkód.

A megszakítást is ezek alapján át kell alakítani. A teljes lépéses módnál minden nyolcadik értéket kell felhasználni. Féllépéses esetben 4-gyel nő, vagy csökken a lépésindex.

```

//Mikrolépés
#if (lepesmod == 1)
    float szinuszertek[16] = { 0, 0.19, 0.37, 0.56, 0.73, 0.83,
    0.92, 0.98, 1, 0.98, 0.92, 0.83, 0.73,
    0.56, 0.37, 0.19 }; //diszkrét szinuszértékek

    for (int i = 0; i < 16; i++)
        MotorErtek[i]=(int16_t)(szinuszertek[i]*maxaram*1024/
        5) + 512;

//Féllépés
#elif (lepesmod == 2)
    MotorErtek[4] = MotorErtek[8] = MotorErtek[12] = aramertek;

//Teljes lépésmód
#elif (lepesmod == 3)
    MotorErtek[0] = MotorErtek[4] = MotorErtek[8] =
    MotorErtek[12] = aramertek;

#endif

aramA = MotorErtek[lepes & 15];
aramB = MotorErtek[(lepes + 8) & 15];

#elif (lepesmod == 2) //Féllépés
    if(PIND & (1<<PD3))
        lepes = lepes+4;
    else
        lepes = lepes-4;

#elif (lepesmod == 3) //teljes lépés
    if(PIND & (1<<PD3)) //jobbra forog
        lepes = lepes+8;
    else
        lepes = lepes-8;

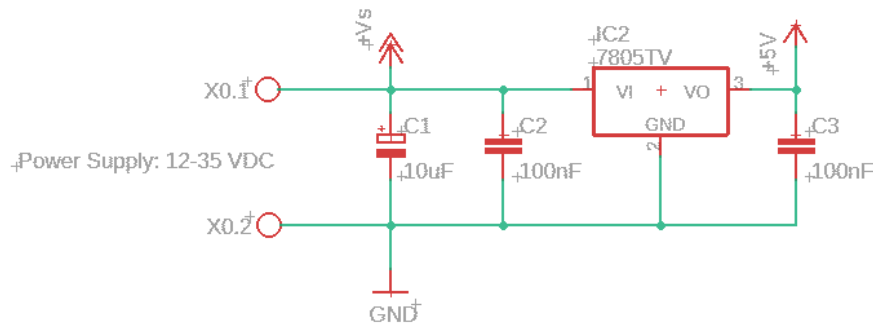
#endif

```

5 A hardver felépítése

5.1 Tápegység

A tápegység feladata, hogy az IC-eket és a motort is feszültséggel és árammal lássa el. A kapcsolásban szereplő áramkörök üzemeltetéséhez stabil 5V-ra van szükség, ugyanakkor egy léptetőmotor ennél magasabb feszültséget igényel. A különböző teljesítmények végett egy 7805-ös, 5V-os feszültségstabilizátor IC-t használtam a hozzá tartozó szűrő elemekkel együtt. Az általam megvalósított kapcsolás a 27. ábra látható.



27. ábra: A tápegység kapcsolási rajza

Az X0.1 és X0.2 sorkapocsra akkora feszültséget kell adni, mely a motort is biztosan meghajtja a kívánt fordulatszámmal. Ez a feszültség maximum 35V lehet, melyet a 7805-ös IC hosszútávon meghibásodás nélkül elbír. Az IC szekunder oldalán, funkciójából adódóan 5V jelenik meg.

A primer oldalon a 10µF-os polarizált kondenzátor kisimítja a nagy feszültségtüskéket, majd ezt egy második, egy 100nF-os kondenzátor újból megszűri. A szekunder oldalra a kisebb tüskék megjelenése miatt egy darab 100nF-os kondenzátort raktam.

A tápfeszültség csatlakoztatásakor figyelni kell a megfelelő polaritásra, ellenkező esetben az IC tönkre mehet.

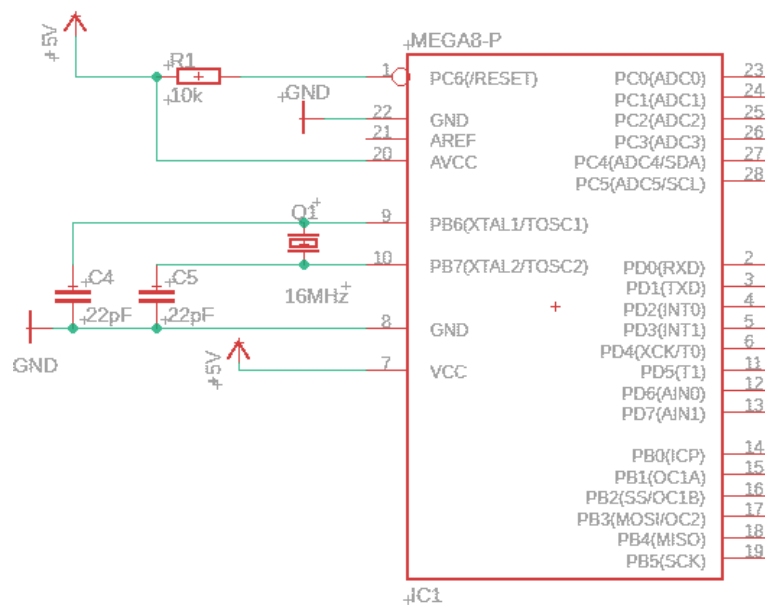
5.2 ATmega8 mikrokontroller

Az ATMEGA8L-8PU gyári jelöléssel ellátott mikrokontrollert használtam a vezérlés megvalósításához. A DIP tokozást választottam, mivel a lábtávolságok elég nagyok, így könnyebb forrasztani.



28. ábra: DIP tokozású ATmega8 mikrokontroller

Kapcsolási rajza a 29. ábra figyelhető meg a kiegészítőelemekkel együtt. A 16MHz-es órajel frekvencia stabilan tartásához egy kvarc kristályt használtam, mivel a mikrokontroller belső oszcillátora túl lassú. Az AVCC-re csatlakoztatott 5V az A/D átalakító üzemeléséhez szükséges feszültség. A PC6(/RESET) láb egy ellenálláson keresztül rá van kötve az 5V-ra azért, hogy a továbbiakban a kapcsolás bővíthető legyen egy nyomógombbal, aminek a másik vége GND-re van kötve.

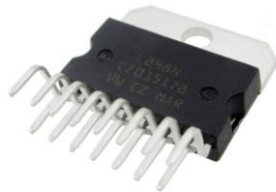


29. ábra: A mikrokontroller lábkiosztása

5.3 L298

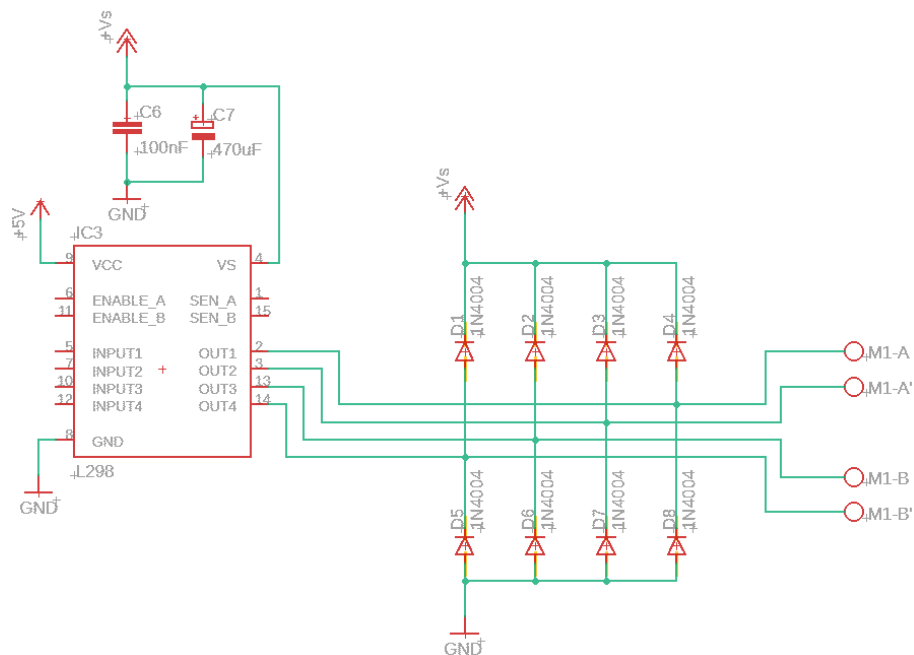
Magát az IC-t és tulajdonságait már bővebben kifejtettem a 3.2 fejezetben. Ebben a fejezetben a felhasznált konkrét darabról és bekötését ismertetem.

Az L298-as IC-ből a furatszerelt, álló kialakításút választottam a könnyű forraszthatóság miatt.



30. ábra: Furatszerelhető, álló L298-as

A 23. ábra megfigyelhető, hogy a kapcsolás nem tartalmaz védődiódákat a tranzisztorokkal ellenpárhuzamosan kötve. A védelem kialakítására nyolc darab diódát használtam, mivel a nyolc darab tranzisztor mindegyikéhez szükség van egyre. Az 1N4004 típusú diódára esett a választásom. Ez a típus maximum 1A áramot tud vezetni és viszonylag gyorsak is. A motorok és a védődiódák bekötésének módja a 31. ábra látható.



31. ábra: Az L298 és a védődiódák

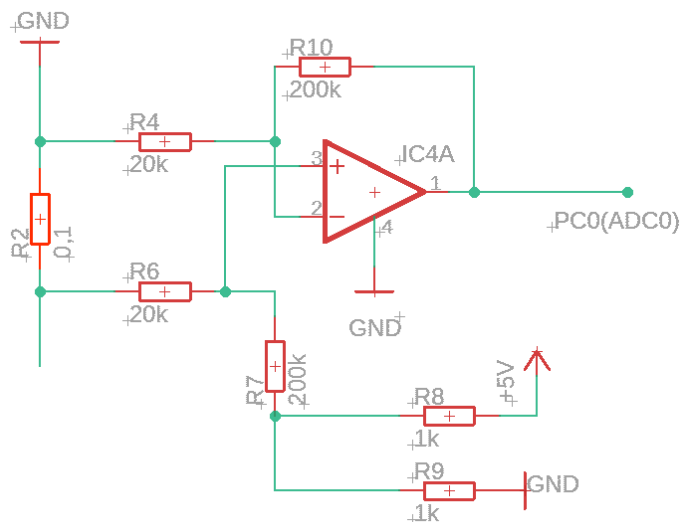
5.4 Árammérés és műveleti erősítő

Ahhoz, hogy az áramot mérni tudjuk a legegyszerűbb és legköltséghatékonyabb módszer egy mérőellenállás beiktatása [21]. Vegyünk alapul egy H-hidat (10. ábra), ahol a föld és a kapcsolás közé sorosan kössünk be egy mérőellenállást. Ezáltal a tranzisztorok bekapcsolásakor az áram átfolyik a mérőellenálláson. Az áram értéke ugyanannyi lesz az összes alkatrészen, Ohm-törvénye szerint azonban a feszültség csökken a mérőellenálláson, és ez a csökkenés arányos az áramerősséggel. Mindezt az teszi lehetővé, hogy söntellenállásnak alacsony ohmos ellenállást használnak. A kis ellenállásválasztásnak a teljesítményvesztés minimalizálása is a célja. A $P=R \cdot I^2$ képlet alapján egyértelműen látszódik, hogy minél kisebb az ellenállás, annál kisebb a hődisszipáció. Jellemzően $0,1\Omega$ értékű sönt ellenállás használatos mérőellenállás céljára [22].

5.4.1 Műveleti erősítő

A $0,1\Omega$ értékű ellenállás hátránya, hogy a mérésünk pontatlan lehet. $2,5A$ maximális áram esetén $0,25V$ feszültség mérhető az AD átalakító bemenetén. Előfordulhat olyan eset is, hogy a maximálisnál kevesebb áram folyik, így a söntnél evvel arányosan csökken a feszültség, akár olyan alacsony szintre is, hogy már nem tudjuk mérni. Ennek a helyzetnek az elkerülésére egy erősítő áramkört valósítottam meg, mely a mérendő feszültségünket tízszeresére erősíti.

A sönt (R_2) ellenállás alacsony értékét az erősítés kompenzálja. A méréshez a söntellenállást GND-re kötjük, másik végéhez csatlakoztatjuk a tekercsek védődiódáinak földjét. A két tekercs miatt két erősítő áramkörre van szükségünk.



32. ábra: Differenciálerősítő offset feszültséggel

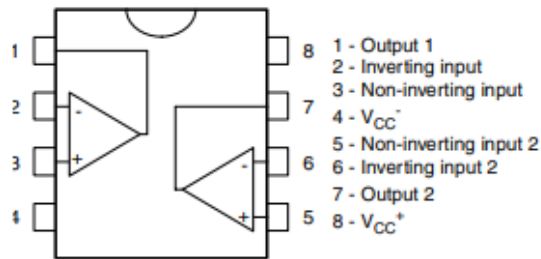
Az erősítés kiszámításának módja:

$$\begin{aligned}
 A &= \frac{R_6 + R_7}{R_6} \cdot \frac{R_{10}}{R_4 + R_7} \\
 &= \frac{R_6 + R_7}{R_6} \cdot \frac{R_7}{R_6 + R_7} \quad , \\
 &= \frac{R_7}{R_6} = \frac{200\text{k}\Omega}{20\text{k}\Omega} = 10
 \end{aligned}
 \tag{18}$$

Mivel negatív áramok is előfordulhatnak, ez offset feszültség megjelenését eredményezi. Ha a mérőellenálláson nem folyik áram, akkor a feszültségkülönbség a két bemenő jel között 0V, viszont az erősítő kimenetén 2,5V jelenik meg. Ezért egy feszültségosztót szükséges használni, ami megfelel az AD átalakító 5V-os referencifeszültségét.

$$U_2 = U_{\text{ref}} \cdot \frac{R_9}{R_9 + R_8} = 5\text{V} \cdot \frac{1\text{k}\Omega}{1\text{k}\Omega + 1\text{k}\Omega} = 2,5\text{V} \quad ,
 \tag{19}$$

Az általam megvalósított kapcsolásban az STMicroelectronics vállalat által gyártott LM358-as műveleti erősítőt használtam. Az eszköz DIP tokozású, furatszerelhető. Láb kiosztása a 33. ábra látható. Két csatornás, ezáltal a két tekercs csatlakoztatása nem okoz problémát.

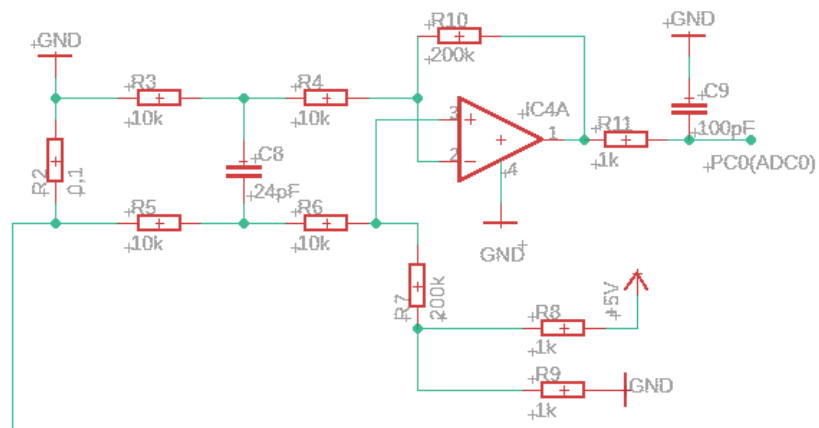


33. ábra: LM358 lábkiosztása [23]

5.4.2 Aluláteresztő szűrő

Az erősítés miatt az erősítőbe mind a bemenő, mind a kimenő jel zajos lesz, mely a mérés pontatlanságát okozhatja. Ezért [22] szerinti aluláteresztő szűrőket raktam a kapcsolásba az erősítő be- és kimenetére. Az aluláteresztő szűrő a kisfrekvenciájú jeleket változatlanul átengedi, a nagyobb frekvenciájú jeleket pedig csillapítja. A legegyszerűbb ilyen szűrő egy ellenállásból és kondenzátorból áll. Ezek értékeit az előbb említett [22] cikk alapján kerültek meghatározásra. Ismeretükben a vágási frekvencia az alábbi képlettel számítható ki:

$$f_0 = \frac{1}{2\pi RC}, \quad (20)$$



34. ábra: Erősítő aluláteresztő szűrőkkel a bemeneten és kimeneten

A 34. ábráról leolvashatóak az értékek, melyekre a vágási frekvenciák meghatározásához szükségünk van.

Az erősítő előtti szűrő vágási frekvenciája:

$$f_0 = \frac{1}{2\pi RC} = \frac{1}{2\pi \cdot 10k\Omega \cdot 24pF} \cong 663482Hz, \quad (21)$$

Ennek a szűrőnek az a feladata, hogy csökkentse a rövid feszültségcsúcsokat, mielőtt még tízszeresére erősítenék azt. A műveleti erősítő a valóságban zajt generál a kimeneten, ezt a második aluláteresztő szűrő elnyomja. R11 és C9 értékek behelyettesítésével ez a határérték is számolható:

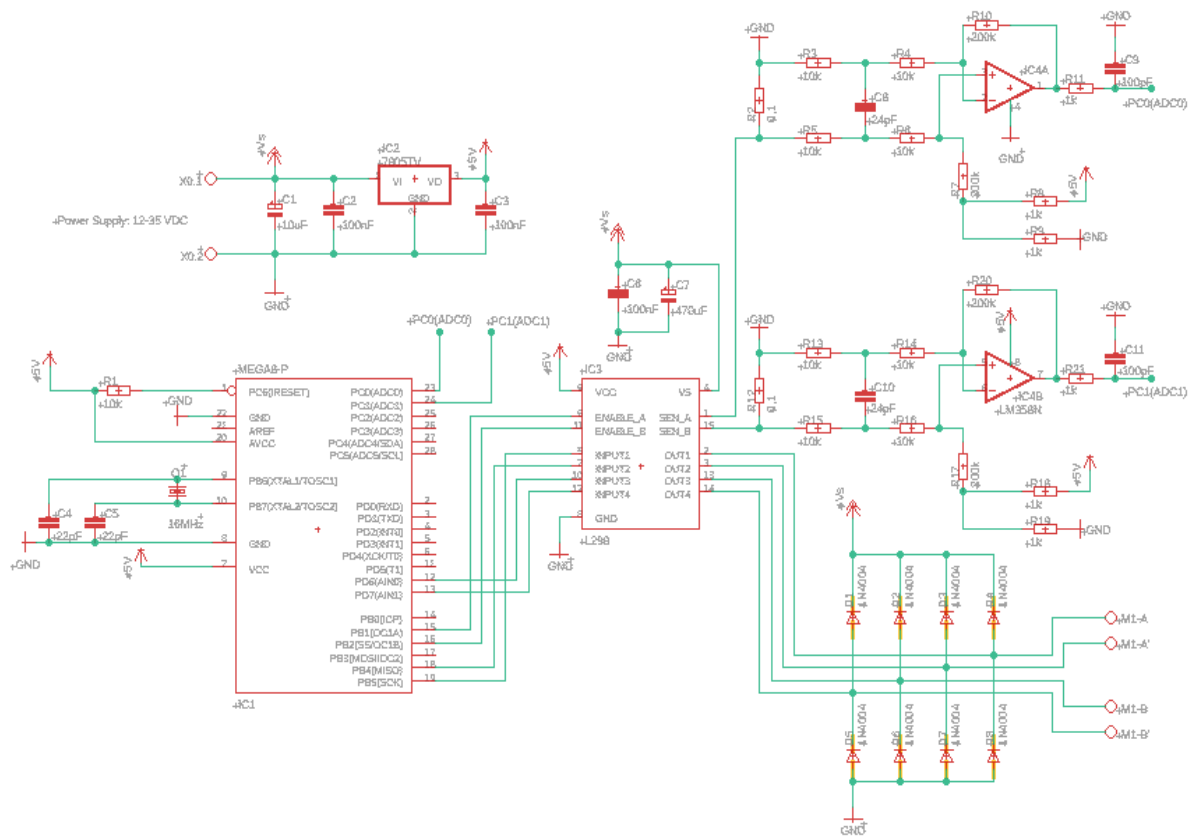
Az erősítő után:

$$f_0 = \frac{1}{2\pi RC} = \frac{1}{2\pi \cdot 1k\Omega \cdot 100pF} \cong 1592357Hz, \quad (22)$$

Ezek hatására a kimeneten egy csillapodott jel fog megjelenni.

6 A megvalósított vezérlő

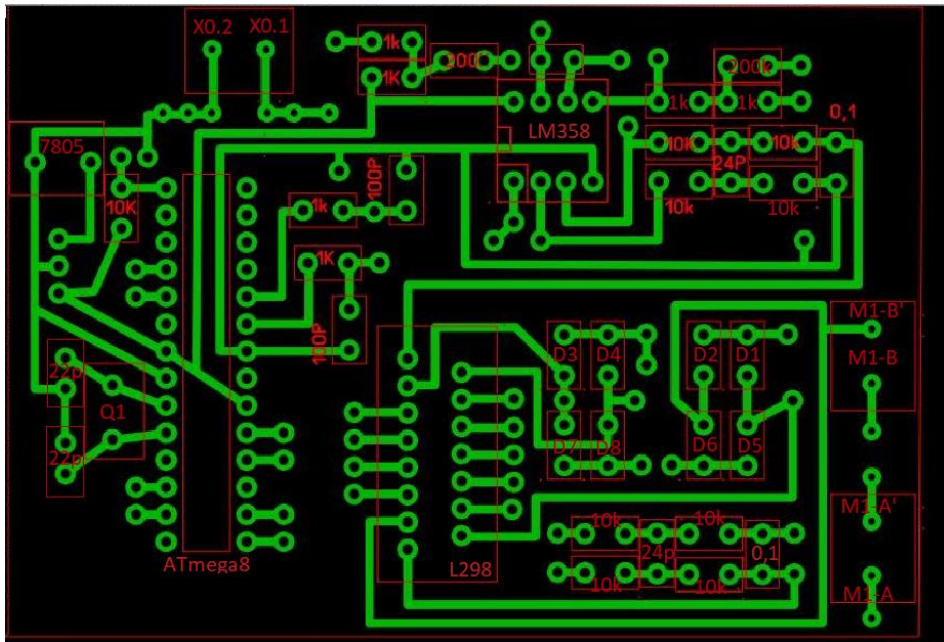
A teljes vezérlő kapcsolási rajza:



35. ábra: Motorvezérlő kapcsolási rajza

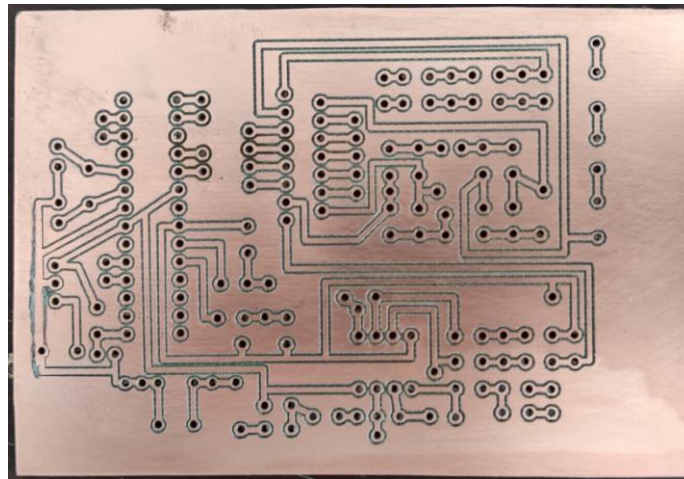
A kapcsolási rajzot az Eagle szoftver 9.6.2 verziójával készítettem.

A nyomtatott áramkör beültetési rajzát a Sprint-Layout 6.0 programmal terveztem meg. Azért esett erre a programra a választásom, mert CNC géppel marattam és furattam ki a nyákot, azonban a CNC-nek mindenképpen *.grbl kiterjesztésű fájlra volt szüksége a működéshez. Tapasztalataim alapján tudtam, hogy ebből a programból tudok ilyen típusú fájlt exportálni.



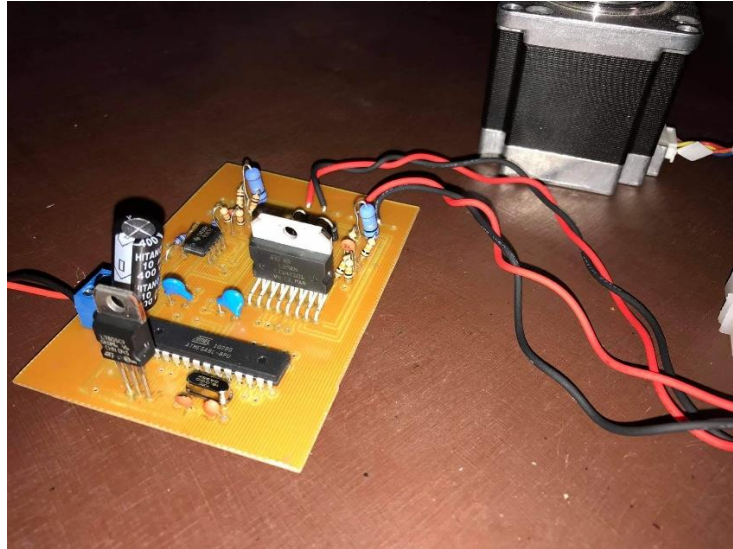
36. ábra: NYÁK-terv

A CNC gép mindenféle szükséges beállítást követően körülbelül 20 perc alatt végzett a feladattal.



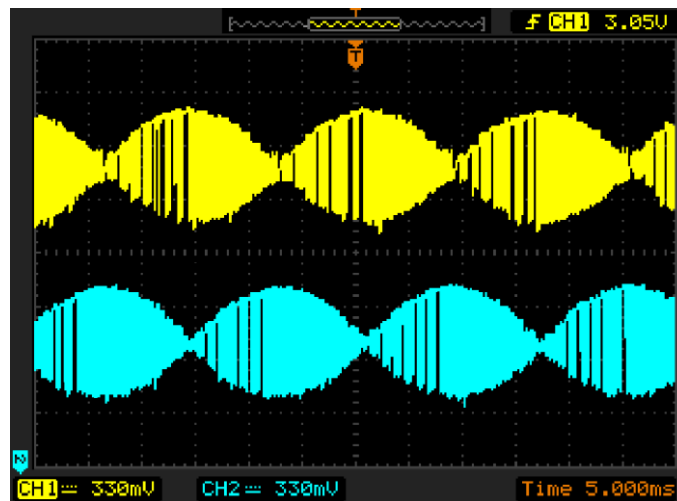
37. ábra: A kimarás és kifűrés utáni NYÁK-lap háttoldala

Az alkatrészek megfelelő helyre történő beforrasztása után csatlakoztattam a léptetőmotort a NYÁK-hoz (38. ábra). Az AD átalakító bemenetén az árammal arányos feszültséget Hantek 6074BE, 4 csatornás, 70 MHz-es, PC-kompatibilis oszcilloszkóppal rögzítettem (39. ábra).



38. ábra: A kész vezérlő léptetőmotorhoz csatlakoztatva

Az ábrából látható, hogy a két jel fázisa egymáshoz képest 90° -kal el van tolódva. Az eredmény megfelelt a várakozásaimnak, pontosan az a jelenség figyelhető meg, mint a mikrolépésnél már elemzett 14. ábrán.



39. ábra: Mikrolépés esetén mindkét tekercs árama (1kHz)

7 Összefoglalás

A diplomamunkám megírása során megismerhettem a léptetőmotorok tulajdonságait, típusait, melyeket részletesen be is mutattam. A dolgozat középpontjában a motorok vezérlésének fejlesztése és maga a vezérlőnek a megépítése állt.

A vezérlő megépítésénél két lehetőség jutott eszembe: az egyik, hogy egy L297-es és L298-as IC pár felhasználásával valósítom meg a hajtást, a másik, hogy egy mikrokontroller segítségével. Az utóbbi mellett döntöttem, mivel hasonló projektet még nem csináltam, így erre nagy kihívásként tudtam tekinteni. A mikrokontroller felprogramozásánál a nagyon komplex, és mindenre kitérő adatlapból szereztem az információkat.

A megvalósított vezérlővel az alapvető üzemmódok közül választhatunk, fél- és egészlépéses mód, illetve lehetőség van mikrolépéses módban is hajtani a léptetőmotort. A mikrolépés miatt szükség volt egy PI vezérlőt szoftveresen implementálni az áram szabályozhatósága végett.

A felhasznált mikrokontrollernek van néhány hátránya, például a lassú AD átalakító. Bár a jelenlegi felhasználás mellett azonban a pontos áramszabályozást nagyon jól el lehet vele érni. A másik hátránya a kibocsátott zaj. Az alkalmazott frekvencia végett sípoló hang hallható.

Összességében elmondható, hogy a diplomamunkám elkészítése közben rengeteg gyakorlati tapasztalatot szereztem a vezérlő tervezése és megépítése, és a megszerzett ismereteket az ipari környezetben elhelyezkedve hasznosítani tudom.

Függelék

A teljes programkód:

```
//main.c

#define F_CPU 16000000UL //16MHz

#define maxaram 2.38
#define M_ind 0.0028 //induktivitás
#define M_ell 0.75 //ellenállás
#define tapfesz 30 //tápfeszültség

#define lepesmod 1
// 1 :mikrolépés
// 2 :fél lépés
// 3 :teljes lépés
#define aramkorlatozas 1 //áramkorlátozás
#define idokorlatozas 7813 //1 = 128us
#define minPWM 4 //kb. 2 us
#define t_mintavetel 0.000128 //mintavételi idő, us

#define KPI 12/((M_ind/M_ell)*tapfesz)
#define PI1 (int32_t)(KPI*(M_ind/M_ell + t_mintavetel/2)*16)*255*5
#define PI2 (int32_t)(KPI*(M_ind/M_ell - t_mintavetel/2)*16)*255*5

#include <stdio.h>
#include <stdlib.h>
#include <avr/io.h>
#include <avr/interrupt.h>

int16_t PISum1 = 0;
int16_t PISum2 = 0;

int16_t elozoHiba1 = 0;
int16_t elozoHiba2 = 0;

int16_t PIout1 = 0;
int16_t PIout2 = 0;

int16_t PISum1; //Szabályozó 1: különbség
int16_t PISum2; //Szabályozó 2: különbség

int16_t elozoHiba1; //előző eltérés, 1.szab.
int16_t elozoHiba2; //előző eltérés, 2.szab.

int16_t PIout1; //PI1 kimenet
int16_t PIout2; //PI2 kimenet

int16_t aramA = 0;
int16_t aramB = 0;

uint8_t lepes = 0; //lépésindex
int16_t MotorErtek[16];

uint16_t szamlalo = 0;
```

```

int main() {

//PWM

TCCR1A = (1<<WGM10) | (1<<COM1A1) | (1<<COM1B1); //törli OC1A+OC1B,
//fast-PWM, 8-bit
TCCR1B = (1<<WGM12) | (1<<CS11); //Előosztó: 8

DDRB |= (1<<PB1) | (1<<PB2); //OC1A és OC1B - két kimenet PWM-nek
TIMSK = (1<<TOIE1); //Timer/Counter1, Overflow interrupt engedélyezés

//AD-átalakító

ADMUX = (1<<REFS0) | (1<<MUX0); //Referenciafeszültség & ADC1
ADCSRA = (1<<ADPS2) | (1<<ADEN); //előosztó: 16 & ADC aktiválás
ADCSRA |= (1<<ADSC); //átalakítás start
while (ADCSRA & (1<<ADSC)) {} //várakozás, amíg in process van

//Irány
DDRD &= ~(1<<PD3); //D3 bemenet

//kimenet L298-nak
DDRB |= (1<<PB4) | (1<<PB5); //tekercs A
DDRD |= (1<<PD6) | (1<<PD7); //tekercs B

//az áram kiszámítása
int16_t aramertek = (maxaram*1024/5) + 512; //Max. áram, 512-1024

//PI

int16_t PI_szab_1(uint16_t aramADC, int16_t aramRef) {

    int16_t mostaniHiba= aramRef - aramADC; //szabályozási eltérés
//számítás
    int32_t temp_PI = (PI1*(int32_t)mostaniHiba -
PI2*(int32_t)elozoHibal)>>14; //ideiglenes változó

    PISum1 = PISum1 + (int16_t)temp_PISum - ((PISum1-PIout1)>>4);
//különbség
    elozoHibal = mostaniHiba; //előző mérés mentés

//PWM limitálás
if (PISum1 > 255)
    PIout1=255;
else if (PISum1 < minPWM)
    PIout1=minPWM;
else
    PIout1=PISum1;

return PIout1;
}

```



```

int16_t PI_szab_2(uint16_t aramADC, int16_t aramRef){

    int16_t mostaniHiba = aramRef - aramADC;
    int32_t temp_PISum = (PI1*(int32_t)mostaniHiba -
    PI2*(int32_t)elozoHiba2)>>14;

    PISum2 = PISum2 + (int16_t)temp_PISum - ((PISum2-PIout2)>>4);
    elozoHiba2 = mostaniHiba

    if (PISum2 > 255)
        PIout2 = 255;
    else if (PISum2 < minPWM)
        PIout2 = minPWM;
    else
        PIout2 = PISum2;

    return PIout2;
}

//Mikrolépés
#if (lepesmod == 1)
    float szinuszertek[16] = { 0, 0.19, 0.37, 0.56, 0.73, 0.83, 0.92,
    0.98, 1, 0.98, 0.92, 0.83, 0.73,
    0.56, 0.37, 0.19 }; //diszkrét szinusztértékek

    for (int i = 0; i < 16; i++)
        MotorErtek[i] = (int16_t) (szinuszertek[i]*maxaram*1024/5)
        + 512;

//Féllépés
#elif (lepesmod == 2)
    MotorErtek[4] = MotorErtek[8] = MotorErtek[12] = aramertek;

//Teljes lépésmód
#elif (lepesmod == 3)
    MotorErtek[0] = MotorErtek[4] = MotorErtek[8] = MotorErtek[12] =
    aramertek;

#endif

aramA = MotorErtek[lepes & 15];
aramB = MotorErtek[(lepes + 8) & 15];

sei();
//globális megszakítások aktiválása
while (1){}

}

```

```

ISR(INT0_vect){ //megszakítás
#if (lepesmod == 1)
    if (PIND & (1<<PD3)) //D3=1, jobbra forog
        lepes = lepes+1; //lépésindex növelés
    else
        lepes = lepes-1; //lépésindex csökkentés
#elif (lepesmod == 3) //teljes lépés
    if (PIND & (1<<PD3)) //jobbra forog
        lepes = lepes+8;
    else
        lepes = lepes-8;

#elif (lepesmod == 2) //Féllépés
    if (PIND & (1<<PD3))
        lepes = lepes+4;
    else
        lepes = lepes-4;
#endif

aramA = MotorErtek[lepes & 15]; //A tekercsáram (sin)
aramB = MotorErtek[(lepes + 8) & 15]; //B tekercsáram (cos)

//tekerbspolaritás
if ((lepes & 0x10))
    PORTB = (PORTB | (1<<PB4)) & ~(1<<PB5);
else
    PORTB = (PORTB | (1<<PB5)) & ~(1<<PB4);
if (((lepes + 8) & 0x10))
    PORTD = (PORTD | (1<<PD7)) & ~(1<<PD6);
else
    PORTD = (PORTD | (1<<PD6)) & ~(1<<PD7);
szamlalo = 1; //számláló beállítása
}

ISR(TIMER1_OVF_vect){ //PWM Interrupt
ADCSRA |= (1<<ADSC); //ADC mérés tekercs1
while (ADCSRA & (1<<ADSC)){} //ADC-re várakozás ~13us
uint16_t aram1 = ADCW; //mért érték mentése

ADMUX &= ~(1<<MUX0); //multiplexer 1 bemenet
ADCSRA |= (1<<ADSC); //ADC mérés tekercs2
while (ADCSRA & (1<<ADSC)){}
uint16_t aram2 = ADCW;

ADMUX |= (1<<MUX0); //multiplexer 2 bemenet
OCR1A = PI_szab_1(aram1, aramA); //PI szabályozó OCR1A-ba
OCR1B = PI_szab_2(aram2, aramB); //PI szabályozó OCR1A-ba
#if aramkorlatozas == 1
    if (szamlalo != 0){
        szamlalo++;
        if (szamlalo > idokorlatozas){
            szamlalo = 0;
            aramA = ((aramA-512)>>1) + 512; //tekercs1 50%
            aramB = ((aramB-512)>>1) + 512; //tekercs2 50%
        }
    }
}

#endif
}

```

Irodalomjegyzék, források

- [1] Zribi M., Sira-Ramirez H., & Ngai A. “Static and dynamic sliding mode control schemes for a permanent magnet stepper motor.” *International Journal of Control*, Nov. 2010.
- [2] Sailaja S., Munta Padmavathi, “Micro – CNC Machines for Displacements along Z-Axis and on XY-Plane.” *International Journal of Innovative Research in Computer and Communication Engineering*, pp 840-841, vol. 4, Jan. 2016.
- [3] Mohamed Y. Tarnini “Fast and Cheap Stepper Motor Drive” 4th international conference on renewable energy research and application, IEEE, Italy, 2015.
- [4] <https://pt.slideshare.net/PreethiGamage/stepper-motors/4>, Online: 2021.02.24.
- [5] https://www.researchgate.net/figure/Stator-and-rotor-design-of-stepper-motors-a-Two-phase-Permanent-magnet-PM-stepper_fig1_334479131, Online: 2021.02.24.
- [6] László Számel, Tibor Vajsz, “The Special Characteristics of Stepping Motor Drives and a New Type of Classification” *Acta Polytechnica Hungarica*, pp. 86-92, Vol. 13, No. 7, 2016.
- [7] László Számel: “A léptetőmotoros hajtások különleges tulajdonságai”, ENELKO, Alba Iulia, Romania, 2012, pp. 150-154.
- [8] <https://www.digikey.at/en/articles/techzone/2011/oct/psoc-3-stepper-motor-control-with-precision-microstepping>, Online: 2021.03.05.
- [9] <https://www.orientalmotor.com/stepper-motors/technology/stepper-motor-overview.html>, Online: 2021.03.10.
- [10] Ahmad M. Harb, Ashraf A. Zaher, “Nonlinear control of permanent magnet stepper motors”, *Communications in Nonlinear Science and Numerical Simulation*, Vol. 9, pp. 443-458, Aug. 2004.
- [11] Yedamale P., Chattopadhyay S., “Stepper Motor Microstepping with PIC18C452”, 2002 Microchip Technology Inc.
- [12] <https://www.orientalmotor.com/stepper-motors/technology/hybrid-stepper-motors-v-hybrid-control.html>, Online: 2020.03.22.
- [13] <https://vshamu.files.wordpress.com/2011/03/unipolar.gif>, Online: 2020.05.10.
- [14] <http://fabacademy.org/archives/2014/students/begle.moritz/week13.html>, Online: 2021.03.22.
- [15] <https://www.rs-online.com/designspark/stepper-motors-and-drives-what-is-full-step-half-step-and-microstepping>
- [16] <http://homepage.cs.uiowa.edu/~jones/step/an907a.pdf>, Online:2021.04.15

- [17] <https://www.st.com/resource/en/datasheet/cd00000063.pdf>, Online:2021.04.15
- [18] <https://www.st.com/resource/en/datasheet/l298.pdf>, Online:2021.04.20
- [19] Sorin Manea: Stepper Motor Control with dsPIC® DSCs.
https://www.microchip.com/stellent/groups/techpub_sg/documents/appnotes/en546027.pdf, Online:2021.04.21
- [20] ATMEL: Atmega8. http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2486-8-bit-AVR-microcontroller-ATmega8_L_datasheet.pdf, Online:2021.04.10
- [21] Microchip: Motor Control Sensor Feedback Circuits.
<http://ww1.microchip.com/downloads/en/AppNotes/00894a.pdf>,
Online:2021.04.12
- [22] Microchip: AN1307 Tuning Procedure ReadMe.
<http://ww1.microchip.com/downloads/en/AppNotes/AN1307%20Tuning%20Procedure%20ReadMe.pdf>, 2021.04.12
- [23] LM358 műveleti erősítő adatlap:
https://www.hestore.hu/prod_getfile.php?id=8608, 2021.04.15
- [24] https://www.mikrocontroller.net/articles/AVR-GCC-Tutorial/Analoge_Ein-und_Ausgabe, 2021.04.12