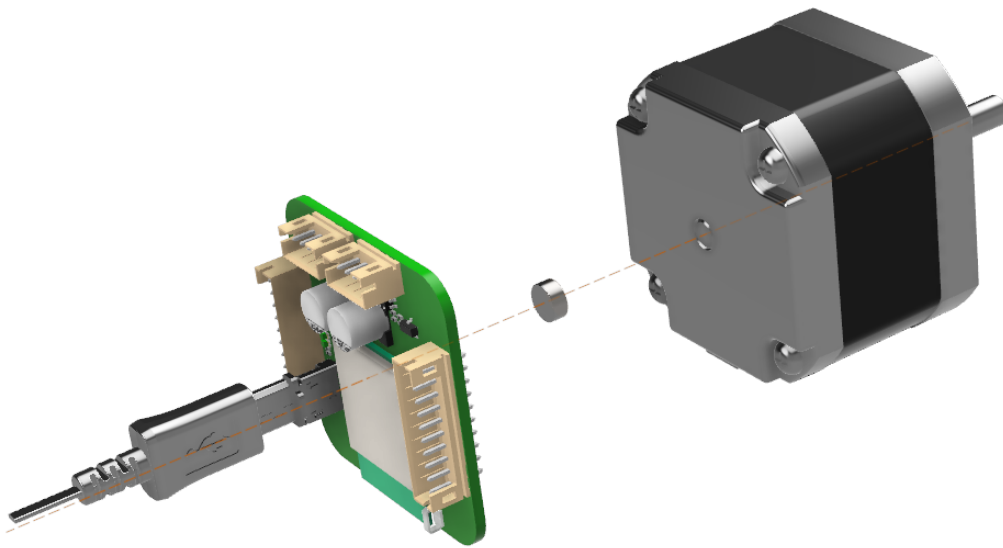




CHALMERS



Strategier för kontroll av stegmotorer

Examensarbete i Elektroteknik, vid Institutionen för Elektroteknik

JACOB ANDERSSON
LUDVIG JOHANSSON

EXAMENSARBETE

Strategier för kontroll av stegmotorer

JACOB ANDERSSON
LUDVIG JOHANSSON

Institutionen för Elektroteknik
CHALMERS TEKNISKA HÖGSKOLA
Göteborg, Sverige 2018

Strategier för kontroll av stegmotorer
JACOB ANDERSSON
LUDVIG JOHANSSON

© JACOB ANDERSSON, LUDVIG JOHANSSON, 2018

Examensarbete 2018:01
ISSN 1652-8557

Institutionen för Elektroteknik
Chalmers tekniska högskola
SE-412 96 Göteborg
Sverige
Telefon: +46 (0)31-772 1000

Omslag:
3D-vy av stegmotor, magnet och drivkrets

Chalmers Reproservice
Göteborg, Sverige 2018

Strategier för kontroll av stegmotorer

JACOB ANDERSSON

LUDVIG JOHANSSON

Institutionen för Elektroteknik , Chalmers tekniska högskola

Examensarbete

SAMMANFATTNING

Stegmotorer är mycket använda inom ett flertal områden tack vare deras förmåga att kunna positionera sig med hög precision. Men på grund av dess konstruktion så krävs det en form av enhet som ser till att motorn exekverar de steg som den är avsedd att ta. Företaget Pluspole har för avsikt att ta fram en egen stegmotordrivare, då liknande produkter på marknaden inte uppfyller deras krav, där hög tillförlitlighet och användarvänlighet är målet. Arbetet har grundat sig i att utveckla nödvändiga funktioner och utvärdera dess funktionalitet och tillförlitlighet. Arbetet har även varit givande i grundläggande förståelse för hur stegmotorer fungerar, behovet av en h-brygga för att kunna driva de olika faserna i motorn, användandet av en vinkelgivare för att kunna läsa av motoraxelns position samt användandet av en mikrokontroller och programmering av denna. Arbetet har dessutom givit kunskap och förståelse om hur Open-loop samt Closed-loop drift fungerar och hur dessa kan användas. Det slutgiltiga delen av projektet innehåller en implementerad Open-loop lösning där hastighet, position samt acceleration är möjligt att ställas in och där motorn utförde samtliga funktioner utan att förlora steg. En primitiv lösning på ett Closed-loop system togs även fram där det var möjligt att ange position, dock räckte inte tiden till för att utveckla en PID-regulator.

Nyckelord: Stegmotor, Pluspole, drivare, stegmotordrivare, h-brygga, vinkelgivare, mikrokontroller

ABSTRACT

Stepping motors are well used in several areas thanks to its ability to position with high accuracy. But because of its construction, it is necessary to use some kind of unit that makes sure that the motor execute the steps that it is assigned to do. The company Pluspole has the ambition to produce their own stepper motor driver, this because the existing products on the market do not meet the requirements, where high reliability and usability is the goal. The project has been based around the development of necessary functions and to evaluate its functionality and reliability. The project has also been rewarding in basic understanding how stepping motors work, the need of a full bridge to be able to drive the different phases of the motor, the use of an angle sensor for measuring the motor shafts angle and the use of a micro-controller and the programming of it. The project has also given deeper knowledge and understanding of Open- and Closed-loop systems and how they can be used. The final part of the project contained an implementation of an Open-loop solution where speed, position and acceleration is selectable, the motor should also be able to do these tasks without losing any steps. A primitive solution for a Closed-loop system was created where the user could choose a position. However the time was not enough for implementing a PID-regulator.

FÖRORD

Denna rapport är en del av det examensarbete som högskoleingenjörer inom elektroteknik vid Chalmers tekniska högskola, institutionen för Elektroteknik, skall genomföra under sista halvåret av årskurs tre. Arbetet är det avslutande momentet och motsvarar 15 av de 180 högskolepoäng för en högskoleingenjör. Projektet har utförts i lokalerna hos Pluspole i Mölndal där också större delen av arbetet har gjorts.

Vi vill tacka medarbetarna på Pluspole men framförallt vill vi rikta ett stort tack till våra handledare:

Mikael Bengtsson, Pluspole AB

Göran Hult, Institutionen för Elektroteknik på Chalmers tekniska högskola

Ludvig Johansson
Jacob Andersson
Göteborg, 2018

TERMINOLOGI

UART - Universal Asynchronous Receiver-Transmitter

GMR - Giant MagnetoResistance

EMK - Elektro-Motorisk Kraft

MCU - Micro Controller Unit

DAC - Digital to Analog Converter

ADC - Analog to Digital Converter

DSP - Digital Signal Processor

CPU - Central Processor Unit

SSC - Synchronous Serial Communication

SPI - Serial Peripheral Interface

LUT - Lookup table

FFT - Fast Fourier Transform

ISR - Interrupt Service Routine

FOC - Field Oriented Control

I/O - Input/Output

IoT - Internet of things

VR - Variable Reluctance

B - Magnetisk flödestäthet

T - Moment

INNEHÅLL

Sammanfattning	i
Abstract	ii
Förord	iii
Terminologi	v
Innehåll	vii
1 Introduktion	1
1.1 Bakgrund	1
1.2 Syfte	1
1.3 Avgränsningar	1
1.4 Precisering av mål	1
2 Teknisk bakgrund	2
2.1 Stegmotor	2
2.1.1 Funktionsprincip	2
2.1.2 Mikrostege, Mekanisk varv och Elektriskt varv	4
2.1.3 Vridmomentsbegrepp	6
2.2 Mikrokontroller	8
2.3 H-brygga	8
2.3.1 Funktionsprincip	8
2.3.2 H-brygga som integrerad krets	9
2.4 Vinkelgivare	10
2.4.1 Kompensering för mätfel	12
2.5 Open-loop	12
2.6 Closed-loop	13
2.6.1 Växlingsvinkel	13
3 Genomförande	15
3.1 Strömstyrning för mikrostege	15
3.2 Olika vinkelgivare	16
3.3 Acceleration och Varierad Steglängd	17
3.3.1 Accelerationsalgoritm	17
3.3.2 Varierad excitation	21
3.4 Closed-Loop experiment	22
4 Resultat	23
4.1 Sampling av Vinkelgivardata	23
4.2 Mätningar av acceleration	25
4.3 Closed-Loop	26
5 Förslag på vidareutveckling	28
6 Appendix	29
Referenser	31

1 Introduktion

I denna del kommer projektets bakgrund, syfte, begränsningar samt mål att beskrivas gällande denna rapport.

1.1 Bakgrund

En stegmotor är en mångsidig och stabil motor med möjlighet att implementeras i flera olika applikationer, med den stora fördelen att den går att styra med väldigt hög precision. Stegmotorer implementeras idag i allt från 3D-skrivare till positioneringssystem för industrimaskiner. Problemet med en stegmotor är dock att det krävs en styrkrets, också känt som en drivare, för att kontrollera motorn.

Företaget Pluspole, som utvecklar diverse prototyper, har tidigare nyttjat en tredjeparts drivare i sina projekt vilken de dock ej varit helt nöjda med. Detta har föranlett att företaget väljer att utveckla en egen styrkrets som skall kunna uppfylla deras behov på ett mer tillfredsställande sätt och vara enkel att implementera.

Företaget vill att styrkretsen enbart skall drivas med en matningsspänning och styras med olika gränssnitt så som UART, Wi-Fi samt Bluetooth. Det skall även finnas olika styrsätt, så kallade modes, som motorn skall kunna styras med. Dessa styrsätt innefattar att kunna positionera sig med hög noggrannhet, kunna rotera med bestämd hastighet och att kunna reglera vridmomentet.

Till förfogande finns en redan färdig hårdvara som är utvecklad av företaget. Företaget bidrog även till en arbetsplats, utrustning och material vilket ställdes till förfogande i deras lokaler i Mölndal.

1.2 Syfte

Projektets syfte är att utveckla mjukvaran till en styrkrets för att kunna manövrera en stegmotor. Styrkretsen skall efter avslutat projekt kunna manövrera motorn i olika modes så som Open-loop motorstyrning där motorn skall kunna skifta mellan hastighet och position. Det förväntas också att motorn ej skall missa steg. Det skall även implementeras en Closed-loop motorstyrning där en återkopplad reglerloop skall kunna styra position, vridmoment eller hastighet med hög precision.

1.3 Avgränsningar

Huvuduppgiften med motordrivaren är att den skall kunna styras trådlöst, dock kommer detta inte vara något som tas upp i denna rapport.

Syftet med denna styrkrets är att den skall kunna fungera med ett flertal olika hybrid- stegmotorer, dock kommer detta projekt enbart att använda en specifik motor från ACT MOTOR (part nr 17HS542). Motorn tillhör en typ som går under namnet NEMA17 vilket är ett samlingsnamn för motorer med samma dimensioner. Projektet kommer trots detta att undersöka vilka parametrar som är väsentliga för att kunna skifta till en annan typ av hybrid- stegmotor.

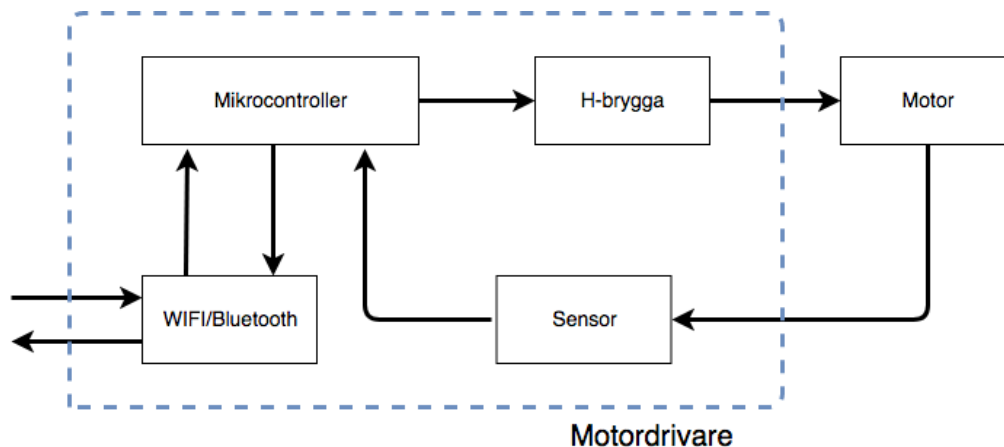
1.4 Precisering av mål

Styrkretsen ska vid projektets slut kunna:

- Kunna köras utan återkoppling där position, hastighet och acceleration kan bestämmas.
- Köras utan att missa steg vid en nominell belastning.
- Kunna köras med återkoppling där den reglerar med hjälp av en PID-regulator motorns hastighet, vridmoment och position.

2 Teknisk bakgrund

I figur 1 visas ett blockschema för en motordrivare som är avsedd att styra en bipolär hybrid- stegmotor. Denna motordrivare består av ett flertal olika komponenter som i figuren illustreras av de block som är placerade inuti den blå streckade linjen. Genom en Wifi och bluetooth modul kan kommunikation föras mellan en användare och systemet. Mikrokontrollern utför beräkningar och skickar signaler till H-bryggan som styr ström till motorns olika faser. Sensorn kan läsa av motoraxelns position och sedan vidarebefordra detta till mikrokontrollern för behandling.



Figur 1: Blockschema som illustrerar ett drivsystem för en stegmotor.

2.1 Stegmotor

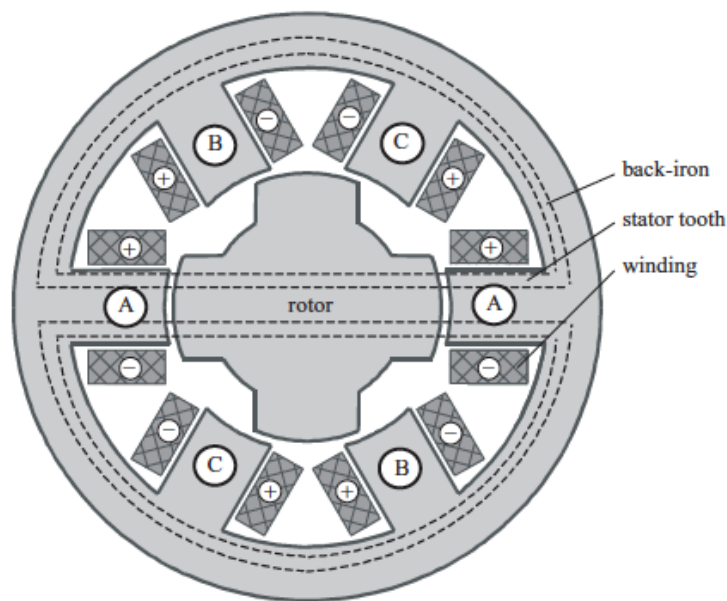
En stegmotor är en typ av motor som lämpar sig för användning då position och precision är av hög prioritet. Det finns ett flertal olika modeller av stegmotorer som variable reluctance (VR) motor, permanentmagnetsmotor och hybridmotor. Det som skiljer dessa åt är bland annat hur motorn är uppbyggd samt dess funktionsprincip. Gemensamt för de olika typerna är att de drivs framåt genom en exciteringssekvens vilket får dem att ta diskreta steg. Stegmotorer kan hålla sin last stilla och producerar då en s.k. "holding torque". Denna är proportionell mot strömmen. Sambanden mellan ström och vridmoment är dock inte lika enkelt medan motorn roterar. Detta kapitel behandlar i huvudsak funktionsprincipen hos en hybridmotor.

2.1.1 Funktionsprincip

Liksom de flesta motorer består en stegmotor av en stator och en rotor. Statorn består av lindningar runt utstick i järn som kallas tänder, även rotorn har tänder. Tänderna i de olika delarna kommer att ge upphov till de karaktäristiska steg som namnger motorn.

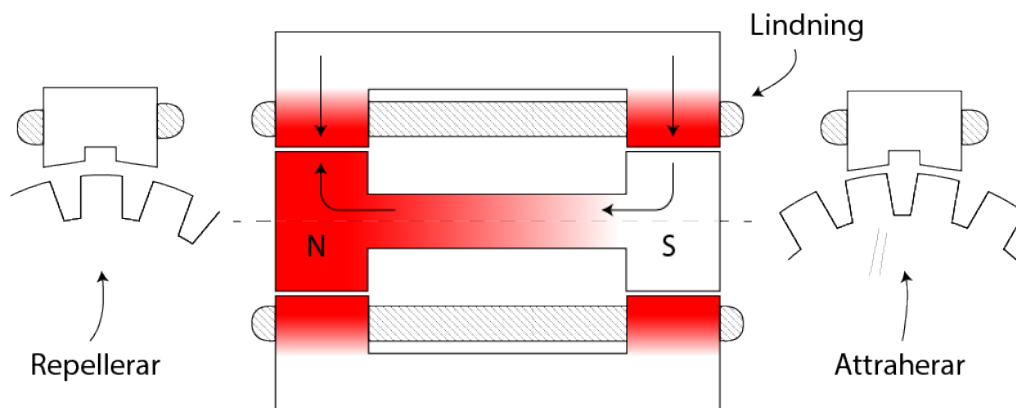
I en VR motor är rotorn gjord i järn. Den har minst tre faser och när en fas strömsätts ger detta upphov till magnetiskt flöde. Samma fas har lindningar mitt emot varandra likt figur 2. Dessa kommer att skapa ett fält som går igenom båda lindningarna. Det luftgap som finns mellan rotorn och statortänderna ger upphov till magnetisk reluktans. Eftersom även rotorn har tänder kommer den att naturligt söka den position där den magnetiska reluktansen är lägst och därmed fältet blir starkast. Denna position blir således där rotor- och statortänderna möter varandra. Det är genom att sedan excitera faserna i en speciell ordning som motorn roteras. Notera att ordningen A-B-C i figuren ger upphov till en rotation av rotorn motsols. En fördel med denna typ är att spänningsmatningen är unipolär samt att rotorn både är billig att tillverka och kan göras lättare än de andra typerna. Figuren ger även en förenklad bild. En vanlig typ som återfinns i praktiken är en motor med 3 faser och 15° steglängd. Detta ger 12 statortänder och 8 rotortänder[1].

Hybridmotorns funktionsprincip är mycket lik VR motorn. Enkelt sett går det att säga att den utnyttjar reluktans i kombination med en permanentmagnet. Rotorn består av en permanentmagnet där polerna ligger i axialled. Vid varje ände av rotorn finns en uppsättning tänder, vilka är fler än stators tänder. Samma lindning



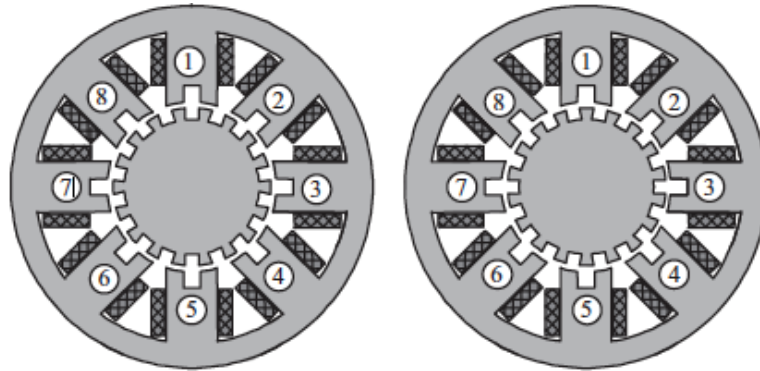
Figur 2: Genomsnitt av en "variable reluctance motor". Återskapad med tillstånd från [1]

ligger runt statorn i båda ändarna, vilket gör att när den magnetiseras kommer den att repellera rotorn i ena änden och attrahera i andra änden, se figur 3. På den attraherande sidan kommer därför tändarna att rada upp sig och på den repellerande sidan radar gapen upp sig. Detta leder till att man kan få ett högt vridmoment på en liten volym. Nackdelen är dock att permanentmagneten kommer att vilja hålla kvar motorn i vissa positioner även när den är strömlös. I jämförelse med en VR motor kan den inte köras lika fort men har högre precision ty de mindre stegen. Den kan inte heller accelereras lika fort eftersom permanentmagneten leder till en tyngre rotor. [1]



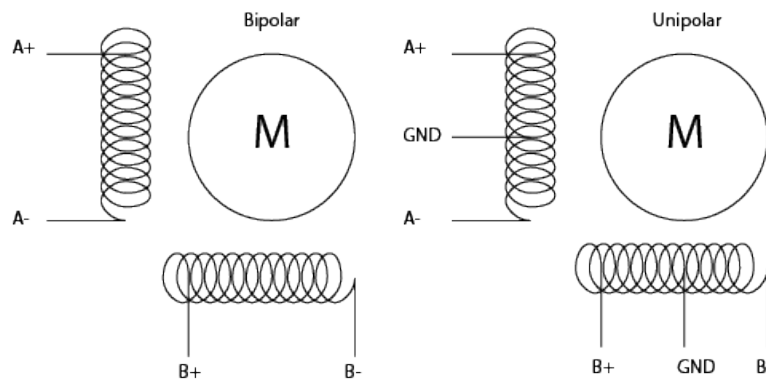
Figur 3: Funktionsprincip för en hybridmotor

Ett annat attribut är exciteringen. En hybridmotor har endast två faser. Eftersom rotorn har fler tänder än statorn går det att bestämma rotationsriktningen genom ordningen som faserna spänningssätts. En vanlig konfiguration är den som syns i figur 4. Antag att de udda lindningarna 1,3,5 och 7 är kopplade till fas A. Då är varannan lindning virad i motsatt håll. Då kan t.ex. en positiv ström på fas A attrahera vid lindning 1 och 5 och samtidigt repellera vid lindning 3 och 7 i ena änden av motorn samtidigt som att den gör det motsatta vid andra änden av motorn. Studera nu lindning 2 och 6 till vänster. Dessa ligger nästan i linje med rotorn. I nästa sekvens strömsätts fas B som är kopplad till de jämna lindningarna på ett sådan sätt att 2 och 6 attraheras och en bestämd rotationsriktning åstadkoms. I praktiken strömsätts båda lindningarna samtidigt i paren A+B+, A+B-, A-B-, A-B+ osv där +/- refererar till polariteten. Denna excitering brukar refereras till som helsteg eller "full stepping".



Figur 4: Genomsnitt av hybridmotor i vardera ände. Återskapad med tillstånd från [1]

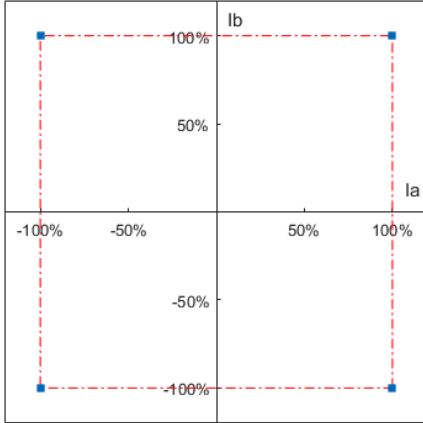
Detta leder till att faserna i en hybridmotor behöver kunna strömsättas i båda riktningarna vilket delar in typen i två underkategorier. Bipolär som känns igen på 4 ledningar samt unipolär som har 6 ledningar där varje linding är dubbel men lindad i motsatta riktningar. De båda typerna illustreras i figur 5. En 6-trådig motor går då att driva unipolärt men kostnaden blir att tråden antingen behöver göras smalare eller lindningsvarven färre vilket leder till att mindre effekt kan matas till motorn. För att kunna driva en bipolär motor krävs istället en H-brygga för varje fas vilket gör att konstruktionen kompliceras men effekten ökas.



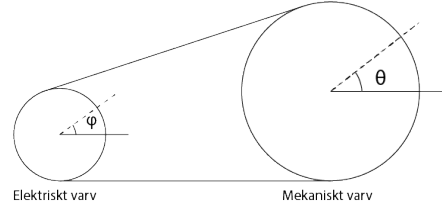
Figur 5: En hybridmotor kan ha 4 eller 6 ledningar

2.1.2 Mikrostepp, Mekanisk varv och Elektriskt varv

I del 2.1.1 refereras det till att en vanlig excitationsschema är helsteg. Till en stegmotor brukar det därför tillhandahållas information om hur många steg per varv den kan ta. Detta bestäms av dess geometri och för en hybridmotor är 200 steg per varv en vanlig siffra. Det innebär även att varje steg är 1.8° på motorns axel. Man kan också införa ett koordinatsystem som visar en "magnetisk" vinkel i motorn. Den refererar då till vilken riktning och magnitud B-fältet kommer att ha i motorn. Det kommer att styras av samt vara proportionellt mot strömmen i fas A (I_a) i ena axeln och strömmen i fas B (I_b) i andra axeln. Eftersom det faktiska B-fältet varierar med bland annat geometrin samt mättnad av kärnan upprättas istället namnet elektriskt varv som refererar till resultanten I_{tot} . Excitationsschemat hade dessutom fyra steg innan det repeterade, vilket gör att vi kan visa dessa fyra steg på det elektriska varvet enligt figur 6.



Figur 6: Helsteg i ett elektriskt varv.



Figur 7: Mellan det elektriska varvet och det mekaniska finns en tänkt utväxling

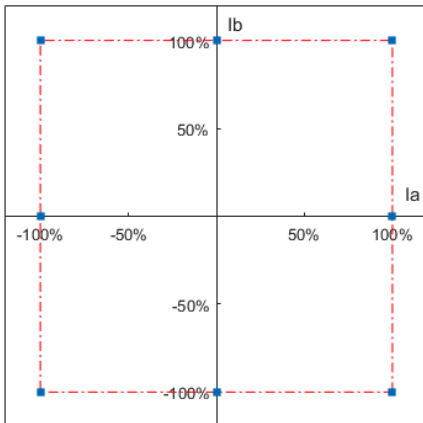
Därför införs här det elektriska varvet med vinkeln φ och det mekaniska varvet med vinkel θ . Då det finns 4 steg på ett elektriskt varv och 200 på mekaniskt varv finns det en utväxling mellan de två varven i ett förhållande som är 1:50. Antalet tänder betecknas p och i detta fall är $p = 50$. Denna utväxling kan illustreras enligt figur 7. Vidare kommer det finnas en skillnad mot den uppmätta vinkeln Θ och den faktiska vinkeln θ på axeln. Skillnaden mellan dessa kallas därför δ och leder till ekvation 1. δ är särskilt viktig för kalibrering då en vinkelgivare på motoraxeln kommer att ge θ . När den verkliga mekaniska vinkeln θ är känd kan φ bestämmas genom att skala upp θ för att genom en modulo operation bestämma φ , se ekvation 2.

$$\theta = \Theta + \delta \quad (1)$$

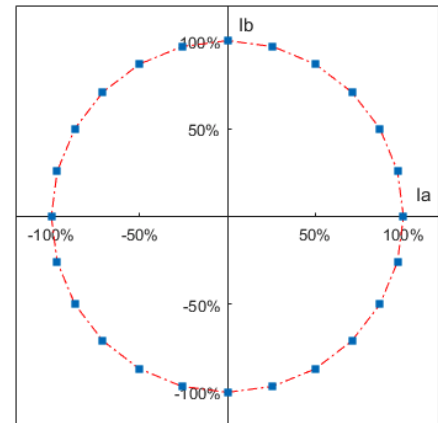
$$\theta \pmod{\frac{2\pi}{p}} = \varphi \cdot p \quad (2)$$

En stegmotor måste inte alltid ta helsteg. Om den tar ett steg som är mindre än det kallas det mikrostep. En specialvariant av mikrostep är halvsteg där man låter en fas vara strömlös i mellansteget. Faserna sätts då som A+B+, A+, A+B-, B- osv. Det ger excitation enligt figur 8. Den yttre streckade linjen är proportionell mot "holding torque". Nackdelen med denna typ av excitation är "holding torque" minskar i mellanstegen. Fördelen är dock enkelheten. Vid varje steg matas 100% av ström till fasen vilken ger en enklare kontroll. Det leder dock till en ojämn gång av motorn.

För att reducera denna ojämn gång samt kunna ta mindre steg är det nödvändigt att kunna reglera strömmens storlek. Detta möjliggör att kunna ta mikrostep enligt figur 9 och stegen kan teoretiskt bli oändligt små. Avkallet blir dock att det maximala vridmomentet minskar då strömmen I_{tot} begränsas av cirkelns radie. Hur denna strömkontroll sker beskrivs i avsnitt 2.3



Figur 8: Halvsteg är ett specialfall av mikrostep.



Figur 9: Om strömmens storlek kan regleras öppnar möjligheten för fina mikrostep och jämn gång

2.1.3 Vridmomentsbegrepp

Stegmotorns speciella egenskaper gör att vridmomentet hos motorn behöver delas upp i olika begrepp. I denna del kommer dessa att förklaras för att ge förståelse för de grundläggande begreppen.

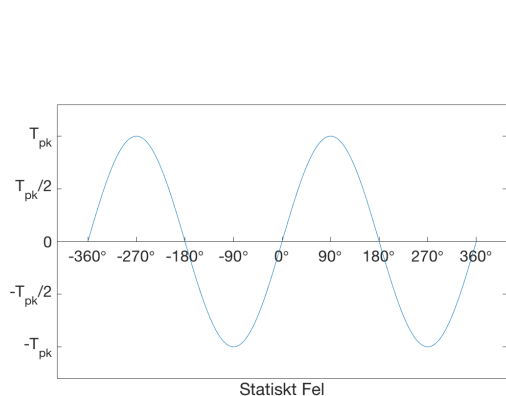
Statiskt vridmoment

När stegmotorn är i vila vid ett jämviktsläge producerar den inget vridmoment. För att exemplifiera, en stegmotor opereras i helsteg och har stannat vid steg n . Om rotorn också har den position som steg n motsvarar är motorn i jämviktsläge och producerar inget vridmoment. Pondera nu att en last finns på rotoraxeln som försöker dra motorn ur jämviktsläget. Då kommer ett vridmoment att uppstå som försöker dra rotorn tillbaka mot jämviktsläget. Detta vridmoment kallas statiskt vridmoment och kan beskrivas som en funktion av det statiska felet, alltså hur långt ifrån jämviktsläget rotorn befinner sig. Om det statiska vridmomenten övervinns kommer rotoraxeln att gå till nästa jämviktsposition. Detta ger de väldigt karaktäristiska klicken som känns om en stegmotor vrids ur position vilket också illustreras i figur 10. Eftersom negativt vridmoment försöker vrida motorn motsols och vice versa kommer alla $\varphi + n \cdot 360^\circ$ positioner att vara ett jämviktsläge. Intressant att notera är även att vridmomentet är som störst vid 90° avvikelse vilket kommer att användas senare för att driva motorn med återkoppling.

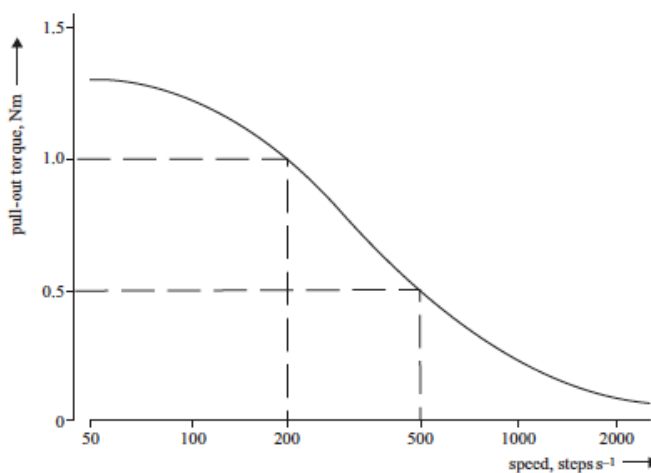
För en hybridmotor är det statiska vridmomentet linjärt proportionellt mot fasströmmen förutsatt att den magnetiska mättingen av kärnan förbises. Det innebär att det maximala vridmomentet T_{pk} i figur 10 är proportionellt mot I_{tot} . T_{pk} kan även benämnas som "holding torque" eftersom det är det största vridmoment motorn kan belastas stillastående utan att den byter jämviktsläge.

Pull-Out Torque

När en stegmotor är i rörelse kommer vridmomentet den producerar att vara en funktion av både det statiska felet och hastigheten. Om motorn körs utan återkoppling är det oönskat att den ska falla ur synkroniseringen mellan den mekaniska vinkeln och den elektriska. Därför blir istället det maximala vridmomentet som det går att belasta med utan att tappa steg intressant. Detta vridmoment brukar kallas "pull-out torque" och en typisk karaktäristik för detta vridmoment illustreras i figur 11. För att förklara denna karaktäristik delas problemet upp i två sektioner. Dels i låg hastighet där hastigheten är mindre än 100 steg/s samt hög hastighet.

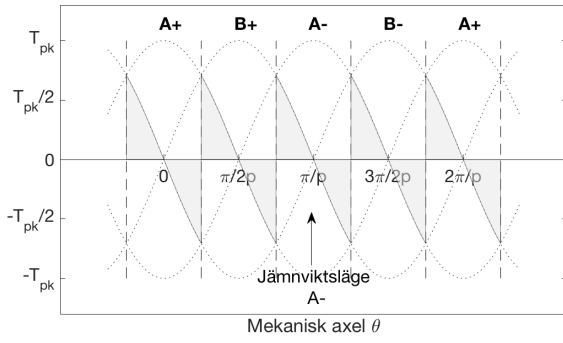


Figur 10: Det statiska felet i den elektriska vinkeln φ ger upphov till ett vridmoment.

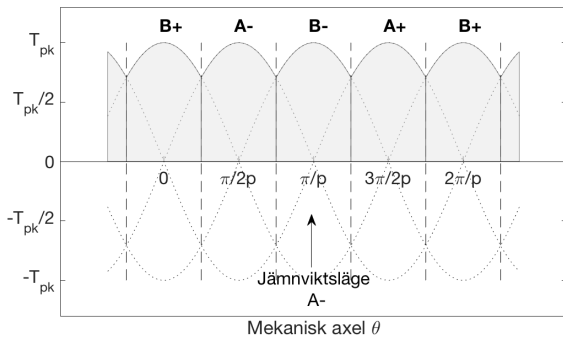


Figur 11: En typisk "pull-out torque" karaktäristik för en stegmotor. Återskapad med tillstånd från [1].

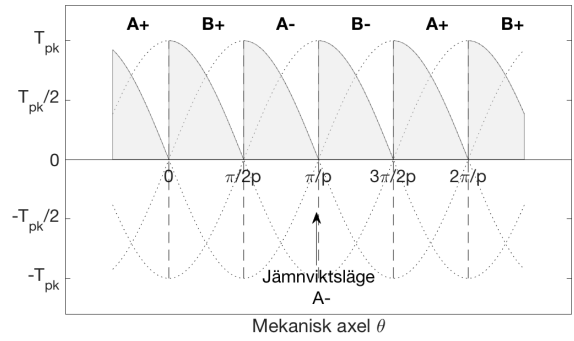
Vid låg hastighet finns ett starkt samband till statiskt vridmoment. Antag en hybridmotor som stegas i helsteg. Det innebär 4 steg per elektriskt varv. Det antas även att motorn rör sig med konstant hastighet och att lasten T_L är lika stor som motorns vridmoment T_M . Om $T_L = 0$ producerar motorn inget vridmoment över tid. I figur 12 illustreras detta. Faserna är exciterade en efter en, här benämnda A+, A-, B+, B- osv. Det



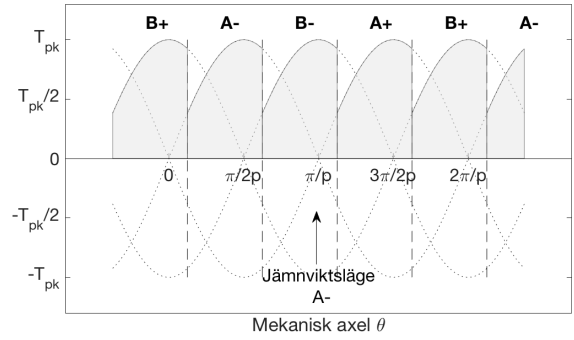
Figur 12: Roterande motor utan last.



Figur 14: När $T_L = T_{PO}$ produceras maxiamlt vridmoment.



Figur 13: En last T_L får motorn att producera mer positivt vridmoment p.g.a. det statiska felet.



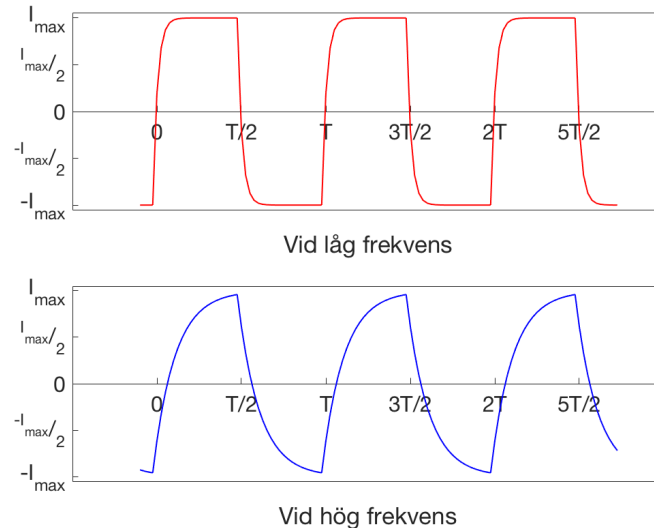
Figur 15: När $T_L > T_{PO}$ minskar motorns vridmoment vilket leder till överstegring.

går dock här att se att $T_M \neq 0$ utan varierar med axelns position eftersom det vid varje excitation uppstår ett statiskt fel som i sin tur ger upphov till ett statiskt vridmoment. Dock kommer genomsnittet att vara $T_M = 0$ vilket förmedlas genom att areorna, vilka motsvarar energin, över excitationsperioden är lika stora. Om T_L ökar kommer, under förutsättning konstant hastighet, en ny balans att infinna sig där $T_L = T_M$. I figur 13 illustreras detta lastfall och eftersom den skuggade arean nu är större är också T_M större. När lasten T_L ökar till läget i figur 14 når motorn det maximala vridmomentet den kan producera vid drift. Om lasten sedan ökar ännu mer kommer arean istället att minska (figur 15). Alltså blir $T_M < T_L$ och därmed saktar lasten ned och faller ur synkronisation med motorns excitering vilket leder till överstegring. Det maximala vridmoment som produceras kan i detta läge beräknas som medelvärde då exiteringen sker under perioden $-3\pi/4p < \theta < -\pi/4p$ och därmed kan "pull-out torque" T_{PO} beräknas vid låga hastigheter enligt ekvation 3. Observera att detta samband gäller vid helstegning. Om ett annat excitationsschema används blir intervallet mindre.

$$T_{PO} = T_{pk} \int_{-3\pi/4p}^{\pi/4p} \sin \theta d\theta \quad (3)$$

Vid högre hastigheter blir effekten av två andra fenomen mer påtagliga. Dels har motorns faser induktans L och resistans R vilket leder till att fasens excitering har en viss stigtid. Detta illustreras i figur 16. Denna stigtid leder i sin tur till två saker. Det första är att medelströmmen som går igenom faser blir mindre. Tidigare har det nämnts att T_{pk} är proportionell mot strömmen, med en högre stegfrekvens minskar även strömmen. Dessutom förskjuts strömmen och ligger kvar efter den önskade perioden och därför ger en negativ påverkan för vridmomentet. Det andra fenomenet är back-EMK som uppstår i motorn, alltså den generator effekt som motorn också har. Även denna kommer att motverka spänningen som läggs på spolarna och därmed ytterligare sänka strömmen. Dessa två effekter, där den ena är tidsberoende och den andra hastighetsberoende, leder till att T_{PO} minskar med ökad hastighet. Dessa två effekter kommer ej att matematiskt härledas i denna rapport. Däremot

förklarar det varför "pull-out torque" minskar vid högre hastigheter. Några korta kommentarer är att en stegmotors EMK kan mätas upp genom att driva motorn med en annan motor. Detta kan ge beräkningsgrunder för konstruktören. L/R-verkan går också att motverka genom att excitera fasen något tidigare vid högre hastigheter. Mer om detta i del 2.6.1.



Figur 16: Induktans och resistans gör att exciteringen av faser får en viss stigtid

2.2 Mikrokontroller

En mikrokontroller är en krets med inbyggd processor, arbetsminne, programminne och kan även innehålla övriga funktioner såsom klockgenerator och diverse I/O element. Mikrokontrollern är anpassad för att styra andra komponenter och det finns idag ett brett utbud av olika mikrokontroller med varierande funktionalitet och prestanda. De finns inuti allt från mobiltelefoner och elektriska tandborstar till bilar och motorer.

I detta projektet används Espressifs mikrokontroller ESP32 som är en mycket mångsidig microcontroller. Den består bland annat av en Xtensa dubbelkärnig 32-bitars mikroprocessor, 2.4 GHz 802.11 Wi-Fi och bluetooth chip, två stycken timer grupper med två stycken 64-bitars timers och en watchdog. Den innehåller även två stycken 8-bitars DAC kanaler som kan omvandla två stycken digitala signaler till två analoga spänningssignaler. ESP32:an är en lågenergi enhet som lämpar sig väl till trådlösa uppgifter och IoT. Det är mikrokontrollern som kommer utföra beräkningar.

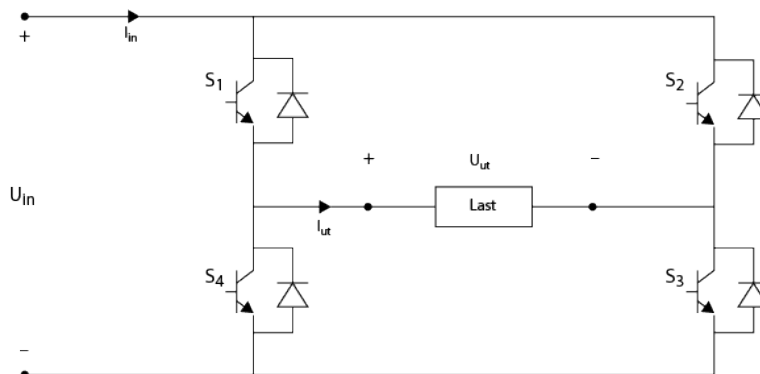
2.3 H-brygga

I tidigare kapitel har vi förklarat behovet av bipolar matning. Med bipolar matning menas att polariteten på drivarens utgångar kan vändas. Strömkällan kommer att ha en DC spänning vilken har fast polaritet, alltså behövs en metod för att kunna byta polaritet. En vanligt förekommande lösning på detta problem är att använda en H- brygga även kallad "full bridge". I denna del kommer funktionsprincipen för denna att förklaras samt hur den specifikt används i detta projekt.

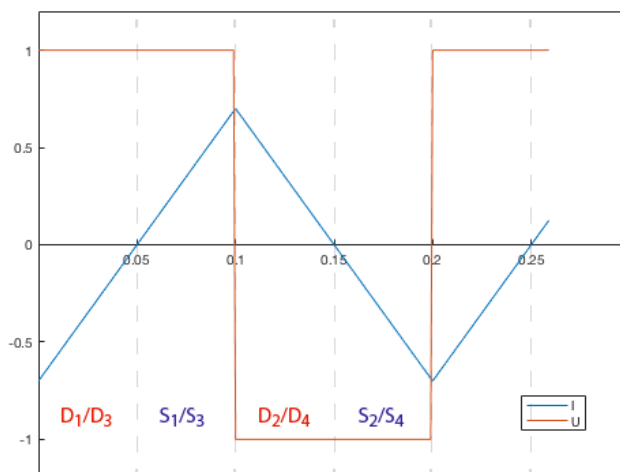
2.3.1 Funktionsprincip

För att kunna leverera ström och spänning i två riktningar är en vanlig lösning den som illustreras i figur 17. De fyra "brytarna", här i form av transistorer och dioder i backriktningen, är arrangerade så att polariteten kan väljas vid lasten. Genom att låta S_1 och S_3 vara slutna blir $U_{ut} = U_{in}$ men om istället S_2 och S_4 är slutna blir istället $U_{ut} = -U_{in}$. Viktigt är dock att hela benet inte får vara slutet samtidigt. Om t.ex. S_1 och S_4 är slutna i samma tidpunkt kortsletter de strömkällan. Diodernas funktion är att ta hand om strömmen som går i

backriktningen. H-bryggor är väldigt vanliga vid olika typer av motordrift. Elmotorer är väldigt induktiva', därför kan lasten i figuren antas vara rent induktiv, varför ström och spänning kommer att se ut som i figur 18.



Figur 17: En H brygga möjliggör bipolär matning



Figur 18: Vid en induktiv last är dess komponenter aktiva beroende på strömmen och spänningens riktning.

2.3.2 H-brygga som integrerad krets

I detta projekt är det inte praktiskt försvarbart att själv bygga upp bryggan. Istället används en integrerad krets med två stycken bryggor. Förutom att det för konstruktören är enklare att lägga till en IC-krets på kortet har kretsen även inbyggda skydd och andra stödfunktioner. I detta projekt används Allegro A4954. Den är konstruerad för att användas till DC-motorer och klarar en matning på 2-40 V in och levererar ut ± 2 A, ± 40 V. Styrningen sker genom två logiska ingångar för varje utgång. Varje utgång kan sättas i följande lägen 0 V, V_{in} , $-V_{in}$ eller Z (hög impedans). Kretsen som innehåller två bryggor gör att det räcker med 4 logiska signaler för att styra de båda faserna A och B till motorn. Då kretsen har inbyggd logik går det ej att kortsluta ett ben i bryggan (som nämndes i föregående del). Den har även skydd mot kortslutning både på utgång och ingång.

En användbar funktion med kretsen är att den erbjuder strömbegränsning genom ytterligare en analog ingång "Vref" för vardera brygga. Konstruktören kan själv välja att applicera ett referensmotstånd som sitter som strömschunt i vardera bryggas utgång. Denna schunt kommer att ge en analog spänning som är proportionell mot strömmen och som kan jämföras med "Vref". Därför kan bryggan avgöra om "är"- strömmen är större eller mindre än "bör"- strömmen som sätts med "Vref". Om "är"- strömmen är större än "bör"- strömmen kommer bryggan att begränsa strömmen ut genom att pulsbredsmodulera utgången. Denna funktion har utnyttjats i

projektet för att skapa en kreativ lösning som mikrostege motorn. Se mer om detta i del 3.1.

2.4 Vinkelgivare

För att kunna avläsa en axels position både under rotation men även efter det att den har stannat behövs en vinkelgivare. Det finns ett flertal olika tekniker för att göra detta möjligt, bland annat kan givaren antingen utnyttja hall-effekten eller Giant MagnetoResistance (GMR). Det finns även andra tekniker men de kommer ej att tas upp i denna rapport.

Då industriella platser ofta innefattar bullriga och smutsiga miljöer samt höga vibrationer och temperaturer har funktionen med vinkelavläsning varit problematisk. Dock finns det som nämnts ovan ett flertal tekniker på marknaden som löser detta på ett angenämt sätt, oftast optiska. Dessa är dock känsliga för den industriella miljön.

Funktionsprincipen för en Hall-effektsensor är följande. Ett magnetiskt fält appliceras på ett ledande material, så som en kopparplatta. När en ström flyter igenom plattan kommer detta att bidra till en kraft, känd som Lorentz kraften. Denna verkar på laddningarna som rör sig i kopparplattan. Laddningarna kommer då att böjas av från dess ursprungliga bana och detta kommer att bidra till en spänning som blir möjlig att mäta, vilket illustreras i figur 19. Denna effekt upptäcktes av Edwin Hall 1879 och fick därefter sitt namn Hall-effekten [5].

Den spänning som uppstår mellan långsidorna ändrar av exempelvis en kopparplatta kan beskrivas med

$$V_H = \int_{S_1}^{S_2} E_H ds$$

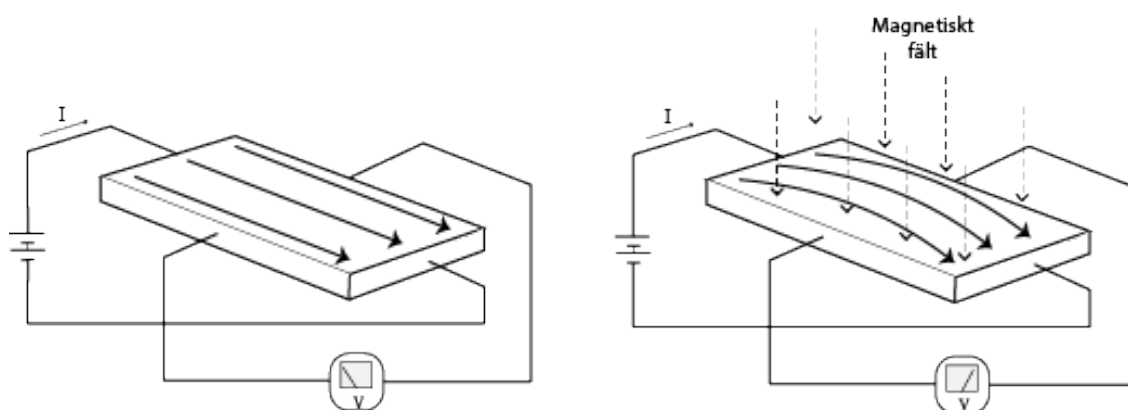
där V_H är Hall spänningen. S_1 och S_2 avser avståndet mellan ändarna och E_H är Hall elektriskt fält och fås genom

$$\mathbf{E}_H = -\mu_{H_n} [\mathbf{E}_e \times \mathbf{B}]$$

Här är $\mathbf{E}_e$ det elektriska fält som sträcker sig längs med det ledande materialet, $\mathbf{B}$ är den applicerade magnetiska flödestätheten och μ_{H_n} är Hall rörelsen för elektroner vilket ges av

$$\mu_{H_n} = r_H \cdot \mu_n$$

där r_H är halls spridningsfaktor och μ_n är rörligheten för elektroner.



Figur 19: Exempel på hur hall effekten genererar en spänning då ett externt magnetiskt fält appliceras över ett ledande material.

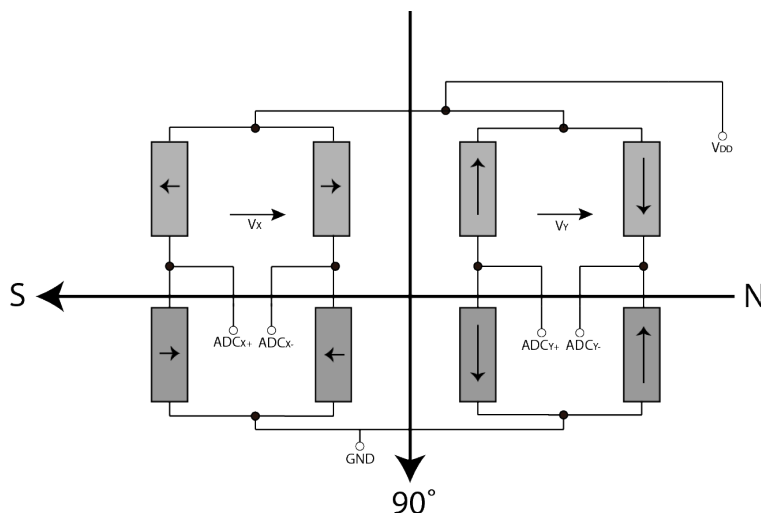
GMR bygger på den upptäckt för vilket forskarna Albert Fert och Peter Grünberg tilldelades Nobelpriset i fysik 2007. Upptäckten visade att när ferromagnetiskt material som separerats med icke-ferromagnetiskt material förs in i ett externt applicerat magnetiskt fält kommer resistansen att förändras [6]. Genom att använda denna teknik tillsammans med vad som kallas en Wheatstones brygga, se figur 20, kan en spänningsskillnad

uppmätas. För att kunna mäta fältet i 360 grader krävs två stycken Wheatstonebryggor med magnetiserade resistiva element vars GMR verkar i olika riktningar, detta för att kunna mäta magnetfältets X och Y komponenter.

Den resistans som uppstår med avseende på magnetfältets vinkel kan beskrivas genom följande formel

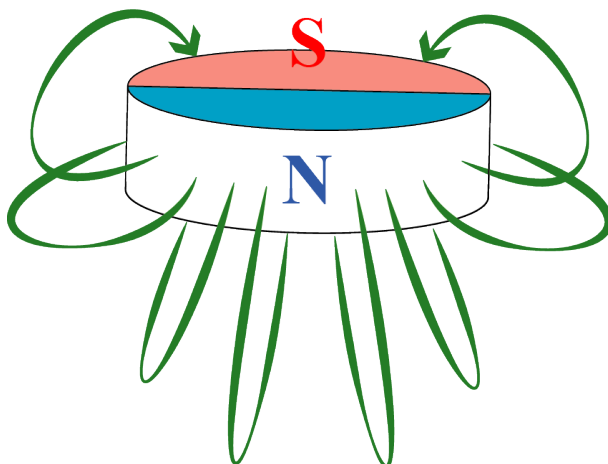
$$R(\theta) = R_{par} + (\Delta R/2)[1 - \cos(\theta)]$$

Där ΔR är den maximala ändringen av resistansen då vinkeln θ mellan det ferromagnetiska och det icke-magnetiska lagret går från parallell till vinkelrät. R_{par} är den resistans när lagren är parallella med det externa fältet [3].



Figur 20: Exempel på hur GMR med Wheatstones brygga ser ut. Då det externa fältet har samma riktigt som det magnetiska materialet kommer resistansen istället vara maximal.

Som nämnts ovan bygger de båda mätsätten på att ett externt magnetiskt fält tillförs. Detta görs genom att en permanentmagnet fästs på motorns axel (i detta fall en skivmagnet). Det magnetiska fältet som existerar runt en permanent skivmagnet illustreras i figur 21. Magneten kan alltså ses som en dipol där $B \sim 1/d^3$ [3]. När den permanenta skivmagneten roterar kommer vinkeln gentemot de resistiva komponenterna i sensorn att ändras. När magnetfältet varierar kommer det att bidra till spänningsskillnader. Genom att analysera dessa går det att avgöra axelns vinkel vid rotation samt stillastående.

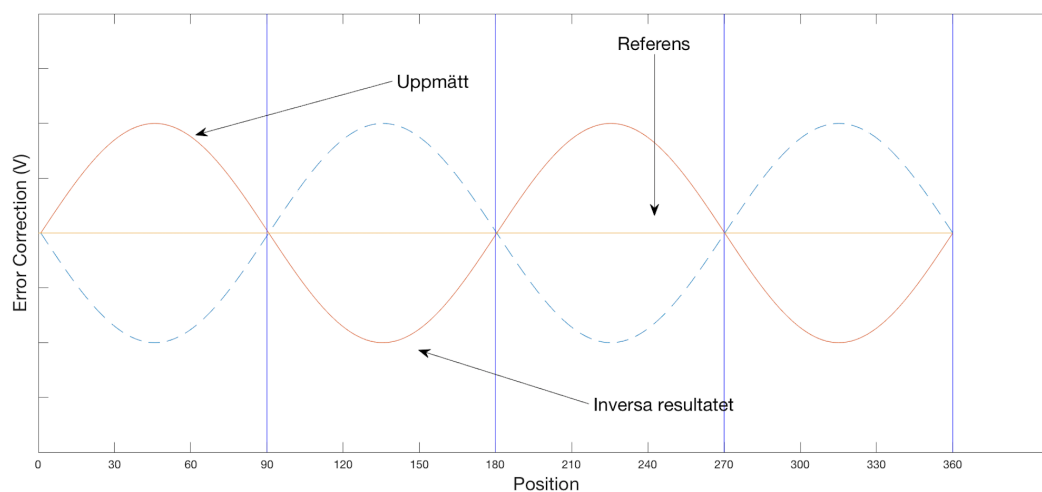


Figur 21: Illustration över hur fältlinjer breder ut sig för en skivmagnet.

2.4.1 Kompensering för mätfel

För sensorer där permanentmagneter används för att göra mätningar kan problem uppstå då dessa inte är helt idealiska. Som nämnts tidigare varierar en liten magnets flödestäthet med $B \sim 1/d^3$ vilket kan bidra till signifikanta mätfel då avståndet blir stort. Om enbart en riktning mäts kommer signalen att representeras av en sinuskurva. Den kan då ses som linjär i intervallet mellan $\pm 30^\circ$. Om istället två riktningar mäts vinkelräta mot varandra kommer signalen att representeras av både en sinuskurva och en cosinuskurva. Detta bidrar då till att vinkeln kan bli uppmätt i hela 360° intervallet.

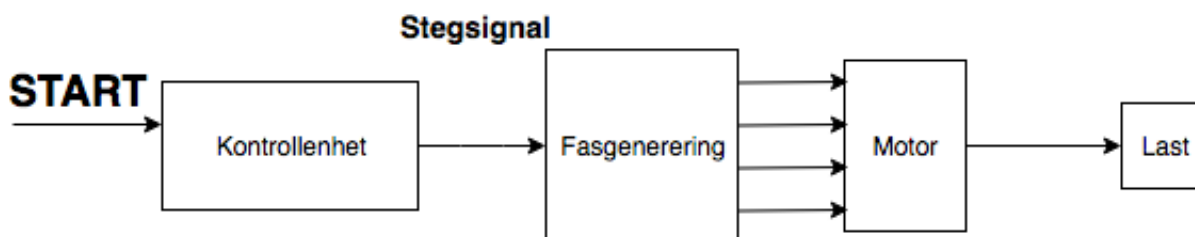
Ett annat sätt att kompensera för fel är att linjärisera signalen med harmonisk linjärisering. Harmonisk linjärisering använder sig utav Fourierserier för att avgöra den periodiska felkomponenten. Det som sker är att fourierkomponenter tas fram för att efterlikna en periodisk signal genom att summera ideala periodiska signaler. Genom att jämföra den uppmätta signalen med en referens går det att se hur felet varierar över varvet. Felet i sin tur kommer innehålla fourierkomponenter, alltså den del som är periodisk med varvet. Detta illustreras i figur 22



Figur 22: Illustration av harmonisk linjärisering.

2.5 Open-loop

Ett av de enklaste sätten att köra en stegmotor är genom konstant steghastighet samtidigt som man driver den enligt open-loop principen. I figur 23 visas ett blockschema över hur ett sådant system kan se ut. Genom att excitera faserna vid olika tidpunkter tar motorn flera steg. Beroende på ordningen som de exciteras kan motorn antingen rotera medurs eller moturs. Genereringen av dessa signalerna sker oftast med hjälp av någon logisk enhet, till exempel en mikrocontroller som är beskriven i sektion 2.2 [4].



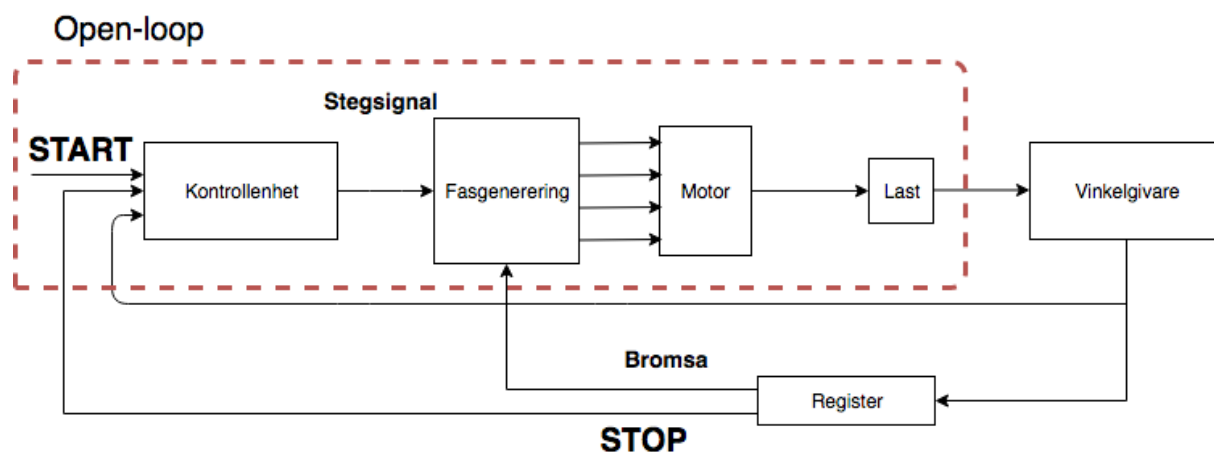
Figur 23: Ett blockdiagram över hur ett open-loop system kan se ut.

Ett system som drivs enligt open-loop principen har ingen återkoppling för position på den last som förflyttas, detta innebär att det inte går att avgöra om lasten verkligen har kommit till rätt position eller inte. Det sätter stora krav på att motorn exciterar stegen korrekt enligt de signaler som den får från mikrokontrollern. Det finns

också vissa restriktioner för hur motorn skall styras. Om det finns en hög tröghet i systemet kan motorn inte gå från stillastående till högsta hastigheten direkt, det måste därför finnas en rutin som accelererar motorn till den avsedda hastigheten och likaså retarderar den[1]. Denna rutin är något som kommer att beskrivas senare i rapporten.

2.6 Closed-loop

Ett system som fungerar enligt closed-loop principen bygger på att det på något sätt finns en återkoppling i systemet som gör att den faktiska positionen på motorns axel kan jämföras med ett "bör"- värde. Fördelen med ett closed-loop system jämfört med ett system som drivs enligt open-loop principen är att motorn inte tappar synkronisation och att lastens parametrar hela tiden observeras. I open-loop stegas motorn men det tas ingen hänsyn till om dessa steg faktiskt tagits eller inte.



Figur 24: Uppbyggnad av ett generellt closed-loop system. Området inom den streckade röda linjen symboliserar ett open-loop system.

I figur 24 ses ett schema över hur ett closed-loop system är uppbyggt. Ny position önskas, vilket kommer generera en signal till kontrollenheten. Kontrollenheten kommer börja accelerera lasten samtidigt används avläsningar av vinkelgivaren för att säkerställa att en rörelse har skett. När sedan lasten närmar sig sitt mål kommer kontrollenheten börja bromsa lasten för att sedan stanna vid destinationen. Skulle en störning ske under rörelsen kommer systemet att kompensera vilket ett open-loop system ej skulle göra. Beroende på lastens storlek kommer det ta längre tid att komma fram till det första steg, detta medför att tiden för instruktionen att ta nästa steg kommer att automatiskt justeras för att kompensera för den långsammare accelerationen. Motorn kommer tillslut att komma upp i en maxhastighet, som är beroende på vridmomentet och lasten, tills den kommer fram till den avsedda positionen. När lasten närmar sig sin avsedda destination kommer registret att skicka en bromsa stop signal till kontrollenheten för att stanna exekveringen. Om man antar de svåraste förhållandena gällande lasten för att exekvera Bromsa signalen går det att få en mjuk inbromsning då lasten närmar sig den slutliga destinationen [1].

2.6.1 Växlingsvinkel

I ett closed-loop system är det vinkelgivaren som mäter när ett nytt steg skall tas, när detta sker går att optimera för att maximera "pull-out torque". När hastigheten är liten är det relativt lätt att maximera vridmomentet genom att stegpulserna kommer i det läge då "pull-out torque" kan maximeras (se del 2.1.3). När hastigheten ökar fungerar inte detta sättet lika bra eftersom det ej tar hänsyn till förvrängingen av vågformen som uppkommer på grund av motorlindningarnas tidskonstant (L/R verkan) samt back-EMK. Detta innebär att strömmen fördröjs. För att komma över detta problem krävs det att faserna exciteras före den tänkta rotorpositionen, detta för att de ska hinna etablera en fasström. I ekvation 4 ges spänningarna i faserna. Observera att ω_φ är vinkelhastigheten på motorns elektriska fält.

$$v_a = V_a \cos(\omega_\varphi t) \quad (4)$$

och tidsvariationen för rotorns position ges av ekvation 5 enligt

$$p\theta = \omega_{\varphi}t - \zeta \quad (5)$$

där ω är vinkelfrekvensen av fundamentalkomponenten, p är antalet tänder på rotorn och ζ är impedansvinkeln. Med ekvation 6 går det att beskriva den maximala "pull-out torquen" vid en given hastighet. Här hör lastvinkeln ihop med den genomsnittliga lindningsinduktansen L i motorn samt den totala lindningsresistansen R .

$$\zeta = \tan^{-1}\omega_{\varphi}L/R \quad (6)$$

Som kan ses i ekvationen kommer impedansvinkeln att vara lika med noll vid låga hastigheter, där ω närmar sig noll. Det går då att hitta den korrekta rotorpositionen för excitering från den uppmätta vinkeln även vid högre hastigheter då $\zeta \neq 0$. Vid högre hastigheter kommer impedansvinkeln att öka och tillslut nå sitt maximum vid 90° . Om pull-out torque behövs över hela hastighetsintervallet måste exciteringsvinkeln kompenseras enligt ekvation 4 och 5. Växlingsvinkeln definieras som skillnaden mellan exciteringsvinkeln och den optimala vinkeln för exikvering och beskrivs i ekvation 7 [1].

$$\psi = \tan^{-1}\frac{\omega_{\varphi}L}{R} \quad (7)$$

3 Genomförande

I följande del beskrivs arbetes gång samt de lösningar som har tagits fram.

3.1 Strömstyrning för mikrostepping.

I tidigare kapitlet ”Teknisk Bakgrund” har det förklarats att det finns behov för att både styra utgångarna bipolärt men även storleken på ström för att kunna mikrosteppa.

En metod som användes tidigt i projektet var att styra utspänningens storlek från bryggan genom att pulsbredsmodulera signalen till de logiska ingångarna till IC- kretsen (A4954). Genom att begränsa spänningen ut blir även strömmen begränsad. Det ger dock vissa hinder. Bl.a. kommer induktansen i kretsen samt back-EMK från motorn att påverka strömmens storlek. Det går dock att anta att strömmen ut är proportionell mot spänningen ut, särskilt vid låga motorvarvtal, varför denna metod testades. Dock visade det sig att denna metod hamnade i konflikt med den inbyggda strömbegränsande funktionen hos IC- kretsen.

Problemet var följande. Utspänningens storlek från bryggan styrs genom att de logiska ingångarna pulsbredsmoduleras. Detta gör att utgången till kretsen kommer att skifta mellan 0 V och V_{in} med en hög frekvens. Plötsligt blir strömmen ut så pass stor att IC-kretsen vill gå in och strömbegränsa. Detta leder i sin tur till att IC- kretsen går in och pulsbredsmodulerar utgången i en annan frekvens än den som de logiska ingångarna moduleras i. Resultatet blir oförutsägbart och visade sig genom att motorn rörde sig på ett okontrollerat sätt. Enkelt uttryckt så blev en pulsbredsmodulerad signal pulsbredsmodulerad.

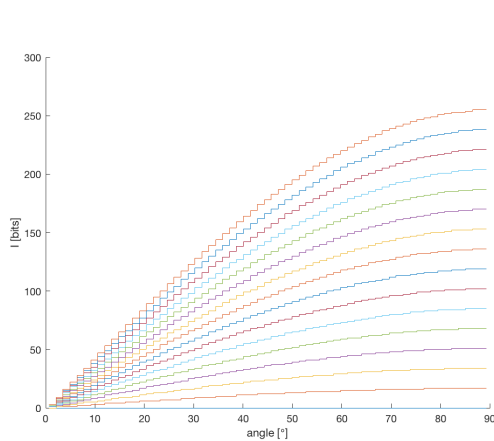
Det ovanstående problemet ledde till att nya lösningar fick sökas. En idé som dök upp var om det gick att utnyttja IC- kretsens inbyggda pulsbredsmodulering. Kretsen har två analoga ingångar, en för varje brygga, som sätter vid vilken nivå som bryggan ska strömbegränsa. MCU:n har två stycken 8 bitars DAC vilka kunde användas för att styra två analoga utgångar från MCU:n. Det innebar att dessa utgångar kunde sättas mellan 0- 3,3 V i 256 olika nivåer. För att 3.3V ska motsvara rätt storlek på ström så måste shuntmotståndet (som nämns i del 2.3) väljas. Från databladet återfinns följande formel;

$$I_{\text{TripMAX}} = \frac{V_{\text{REF}}}{A_V \cdot R_S}$$

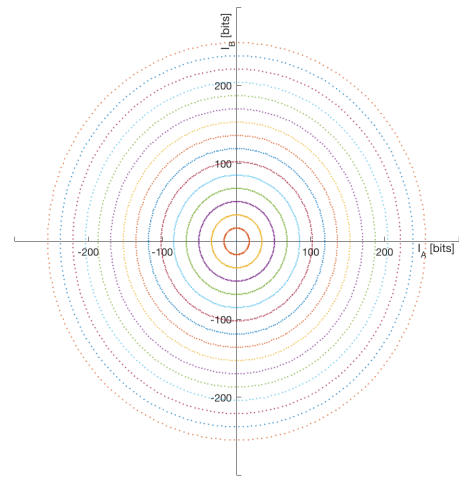
där I_{TripMAX} är max ström ut vid spänningen V_{REF} och R_S är shuntmotståndet samt A_V en förstärkningsfaktor. Genom att lösa ut R_S ur formeln samt sätta I_{TripMAX} till motorns ”maximum rated current” samt V_{REF} till 3.3V går det att välja ett motstånd som tillåter att strömmen sätts till ett värde inom det operationella området. Detta genom att styra ett värde till DAC. Observera att IC-kretsen innehåller två individuella h-bryggor vilket innebär att det är möjligt att styra fas A och B individuellt.

Eftersom det nu fanns möjlighet att styra strömmen för varje fas individuellt vill man kunna strömsätta motorn i en bestämd vinkel och magnitud, som beskrivet i del 2.1.2. Det kan komma att hända att denna vinkel och storlek behöver uppdateras relativt ofta och därför är det önskvärt att undvika beräkningar alltför ofta. Att rotera motorn ”steglöst”, vilket skulle kunna tänkas vara idealfallet, skulle innebära att fasernas storlek styrs av sinusfunktioner. Dessa operationer är kostsamma operationer för en mikrokontroller. Detta gäller särskilt för Xtensa processorn som används i projektet då den saknar DSP. DSP är en periferienhet som annars skulle kunna generera sinussignaler utan att belasta CPU:n. För att undvika dessa beräkningar gjordes istället en LUT, Look Up Table, som kompileras ned tillsammans med källkoden. Det möjliggör för en snabbare algoritm och för att ytterligare optimera den sparades minne genom att endast beskriva 0 – 90°. När denna första del kvantiserades, alltså ritades upp med diskreta steg kan tabellen visualiseras som i figur 25. De olika linjerna visar 16 olika strömnivåer från 0-15 där 15 motsvarar en ström som skall ge 100 % av ”rated current” som är bestämd med hjälp av R_S .

Nästa insikt blev att varje möjligt värde som kunde sättas på h-bryggans logiska ingångar motsvarade en kvadrant. Alltså kan algoritmen utnyttja samma metod om och om igen genom att bara känna till vilken kvadrant den ska vara i och relativ vinkel. Detta ledde till en snabb algoritm för att utnyttja LUT:en för att sätta utström. När algoritmen körs ger den de möjliga lägen enligt figur 26. Eftersom systemet är digitalt är det ofrånkomligt att det blir diskreta punkter. I figuren så består ett varv av 256 punkter. Vinkeländringen mellan

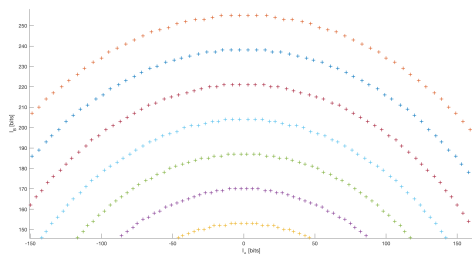


Figur 25: En sinus kvantiserad i första kvadranten med 8 bitar.

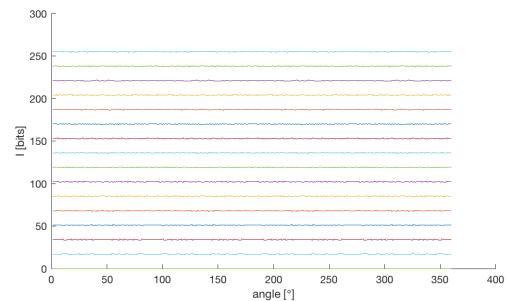


Figur 26: Varje punkt motsvarar en möjlig vektor.

två punkter motsvarar då 0.028° på den mekaniska axeln vilket innebär att den minsta möjliga stegändringen får anses vara tillräckligt liten. En förstoring av ytterkanten av cirklarna finns i figur 27. Genom att ta absolutbeloppet av varje möjlig vektor är det också möjligt att se om strömmens storlek genom de två faserna är relativt jämn över varvet och man kan se i figur 28 att man får ett litet kvantiseringsfel. Det är dock aldrig större än 2 bitar vilket skulle motsvara 15,6 mA om I_{TripMAX} är satt till 2 A eller ett fel på 0,78%.



Figur 27: En förstord bild av de möjliga vektorerna för att visa kvantiseringen.



Figur 28: När man tar absolutbeloppet av varje möjlig vektor kan man se att resultanten varierar något längs varvet.

Genom denna lösning styrs motorns elektriska vinkel φ på ett effektivt sätt som inte tar för mycket resurser från MCU:n. LUT:en har en storlek på ca 5 kB och har genererats genom ett script i Matlab, vilket lätt skulle kunna ändras i efterhand ifall t.ex. fler strömnivåer skulle önskas. Funktionen som används i mjukvaran för att hämta ett värde från LUT och sätta korrekt värde på bryggorna finns med i Appendix som listing 1

3.2 Olika vinkelgivare

För att kunna bygga ett system med closed-loop principen, som nämnts i sektion 2.6, krävs något sätt att kunna avläsa den faktiska vinkeln på motorns axel. Detta för att kunna jämföra detta med vinkelns börvärde. För detta ändamål krävs en vinkelgivare, även kallad encoder.

Från start var det ämnat att använda en vinkelgivare från Allegro av modellen A1335 vilket bygger på tekniken med Hall-effekt. A1335 är en 360 graders kontaktlös vinkelgivare. Den har ett 12-bitars register

med en upplösning på cirka 0.1 grader samt ett uppdateringsintervall på 32 μ s. Vinkelgivaren skall ha en matningsspänning på typiska 5V och minimum 4.5V med matningsström på 15 mA.

Då magnetiska fält inte är helt linjära, bland annat på grund av att vinkelgivaren inte är placerad helt i linje med magneten, krävs linjärisering. A1335 har två funktioner för linjärisering, harmonisk linjärisering och styckvis linjärisering. För en vinkelgivare som skall läsa av vinkeln för 360 grader är harmonisk linjärisering den funktion som fungerar bäst och den som valts att användas. Denna metod beskrivs i del 2.4.1.

Dock byttes den vinkelgivare som kom att användas i den slutgiltiga versionen, istället användes Infineons TLE5012B. Denna vinkelgivare använder sig av GMR för avläsning av det externa magnetiska fältet. Sensorn är kalibrerad från start och dessa värden är lagrade i lasersäkringar. Vid start laddas dess värden i minnen som då också gör det möjligt att modifiera dem i efterhand. För att öka precisionen ytterligare från vinkelgivaren finns inprogrammerade algoritmer i enheten för autokalibrering under drift. Kommunikationen med givaren sker via dubbelriktad SSC som är SPI kompatibel. Vinkelgivaren har en upplösning på 0.01 grader via ett 15-bitars register. Sensorn kan arbeta mellan temperaturer på -40°C till 150°C vilket gör den lämplig för processer som kan utsättas för höga temperaturer, som exempelvis vid drift av en motor. Den går även att driva med en matningsspänning på minimalt 3V och typiskt 5V med en typisk matningsström på cirka 14 mA. Mer än att bara kunna läsa av vinkelvärdet så kan även givaren registrera vinkelhastighet samt vinkelsträcka, detta är väldigt gynnsamt när man vill verifiera om den hastighet som är vald faktiskt stämmer och även registrera antal varv axeln har roterat. Vinkelsträcka är det totala antalet grader som motorn har roterat.

Huvudsakliga anledningen till bytet av vinkelgivare var dess egenskap att fungera med en matningsspänning på cirka 3 V. Som nämnts tidigare behövde A1335 en matning på 5V, men för att minska antalet spänningsnivåer på kortet valdes istället TLE5012B. Tanken framöver är även att kretsen skall kunna drivas via batteri som kopplas in via expansionsport, det är då fördelaktigt med komponenter som har lägre energiförbrukning.

3.3 Acceleration och Varierad Steglängd

Som nämnts tidigare i kapitel 2.1 finns önskan om att kunna göra mikrosteg. En av anledningen till varför man vill kunna mikrostega är framförallt för att kunna få en mer precis positionering, det är även fördelaktigt att använda olika steglängd då man vill få en mjuk acceleration och retardation av motorn.

I många tillämpningar fungerar en stegmotor utmärkt även utan återkoppling. Det är rentav motortypens stora styrka, att den rör sig bestämda avstånd samt att det går att anta att motorns position är den önskade. I den mest rudimentala tillämpningen låter man motorns hastighet bestämmas av den frekvens med vilken steg exekveras. En stor nackdel med denna typ av operation är att motorn antas kunna accelerera från vila till vald hastighet under tiden av vad ett steg tar. Denna metod fungerar bra när hastigheten är låg. Önskas en högre hastighet kommer motorns "pull-out torque" att överstigas, vilket innebär att steg hoppas över [1]. För att undvika detta behöver motorn accelereras långsamt nog att lasten aldrig blir större än "pull-out torque". Det kommer samtidigt att leda till att en högre maxhastighet kan uppnås.

3.3.1 Accelerationsalgoritm

I detta projekt har en algoritm som är starkt inspirerad av [2] skrivits. Grunden för algoritmen ligger i att steg med en bestämd steglängd ska utföras (t.ex. helsteg). Algoritmen drivs framåt av en räknare som gör ett interrupt när den kommer till ett särskilt värde. För att fullt förstå algoritmen måste vissa matematiska grunder klargöras.

Målet med algoritmen är att bestämma vid vilket heltal c som räknaren skall generera interrupt vid. Räknaren stegar upp enligt en frekvens f som går att konfigurera. Tiden mellan två händelser är δt enligt ekvation 8. Observera att c är den enda friheten då f hålls konstant genom operationen.

$$\delta t = \frac{c}{f} \quad (8)$$

Ett stegs storlek i radianer kallas α . Hastigheten ω_θ mäts i rad/s . Detta enligt ekvation 9.

$$\omega_\theta = \frac{\alpha f}{c} \quad (9)$$

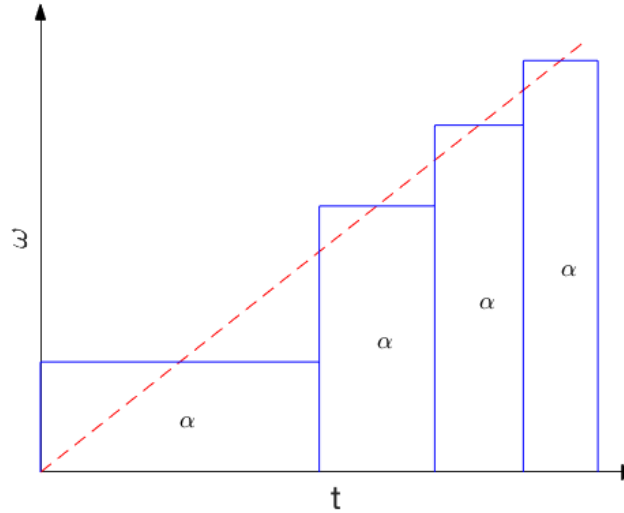
Ett exempel på detta är $f = 1$ MHz, $c = 2500$, $\alpha = \pi/100$ rad. Detta ger hastigheten

$$\omega_\theta = \frac{\frac{\pi}{100} \text{ rad} \cdot 1 \cdot 10^6 \text{ Hz}}{2500} = 12,56 \text{ rad/s} = 120 \text{ rpm}$$

När hastighet är definierad behöver även accelerationen ω'_θ definieras. Räknarbeloppet c_n motsvarar en hastighet enligt ekvation 9 eftersom f och α är konstanter. Därför kan accelerationen definieras som en funktion av c_n och c_{n+1} enligt ekvation 10. Alltså hastigheten i steg n och steg $n + 1$.

$$\omega'_\theta = \frac{2\alpha f^2(c_n - c_{n+1})}{c_n c_{n+1}(c_n + c_{n+1})} \quad (10)$$

Observera att ekvation 10 antar mittpunkten mellan två stegintervall enligt figur 29. Arean α motsvarar ett steg och är konstant så länge steglängden är densamma.



Figur 29: Arean motsvarar ett steg. Ekvation 10 antar mittpunkten mellan två intervall. I figuren är den streckade linjen ω'_θ

Axelns vinkel har i tidigare kapitel definierats som den mekaniska vinkeln θ . Vi använder θ för att skapa ett samband mellan tiden t och steget n enligt ekvation 11. Den sista identiteten gäller eftersom vinkeln måste vara lika stor som det antal steg som har tagits multiplicerat med stegstorleken α . Detta ger vidare ett samband för tiden t_n för det n:te steget enligt ekvation 12.

$$\theta(t) = \int_0^t \omega_\theta(\tau) d\tau = \frac{\omega'_\theta t^2}{2} = n\alpha \quad (11)$$

$$t_n = \sqrt{\frac{2n\alpha}{\omega'_\theta}} \quad (12)$$

I ekvation 12 beskrivs sambandet mellan stegnumret n och accelerationen ω'_θ i en bestämd tidpunkt t_n . Därför kan man beskriva räknarens värde som skillnaden mellan två tidpunkter enligt ekvation 13.

$$c_n = f(t_{n+1} - t_n) \quad (13)$$

Det första steget, $n = 0$, blir dock ett specialfall, som kallas för c_0 . Genom att kombinera ekvation 12 och 13 där $n = 0$ återfås sambandet i ekvation 14. Observera att c_0 innehåller information om frekvens, stegstorlek och acceleration. Detta gör att c_0 kan användas för att uttrycka c_n enligt 15.

$$c_0 = f \sqrt{\frac{2\alpha}{\omega'}} \quad (14)$$

$$c_n = c_0(\sqrt{n+1} - \sqrt{n}) \quad (15)$$

För att summera hur långt resonemanget har kommit finns nu ett uttryck som beskriver sambandet mellan steg n och heltalet c_n . Detta för att räknaren skall göra ett interrupt vid rätt tidpunkt för att en konstant acceleration skall uppnås. Eftersom c_0 bestämmer hur stor accelerationen ω'_θ kommer att bli skiftas intresset till förhållandet mellan två steg, alltså $\frac{c_n}{c_{n-1}}$. Detta samband kan skrivas som ekvation 16 genom att sätta in ekvation 15.

$$\frac{c_n}{c_{n-1}} = \frac{c_0(\sqrt{n+1} - \sqrt{n})}{c_0(\sqrt{n} - \sqrt{n-1})} = \frac{\sqrt{1 + \frac{1}{n}} - 1}{1 - \sqrt{1 - \frac{1}{n}}} \quad (16)$$

Roten ur operationen är dock en ”dyr” operation för en mikrokontroller. Taylor utveckling av $\sqrt{1 \pm \frac{1}{n}}$ ger följande

$$\sqrt{1 \pm \frac{1}{n}} = 1 \pm \frac{1}{2n} - \frac{1}{8n^2} + O\left(\frac{1}{n^3}\right)$$

Genom andra gradens approximation från Taylor-utvecklingen går ekvation 16 att skriva som ekvation 17.

$$\frac{c_n}{c_{n-1}} = \frac{4n-1}{4n+1} \quad (17)$$

Denna ekvation går sen att skriva om till

$$c_n = c_{n-1} - \frac{2 \cdot c_{n-1}}{4n+1}$$

som sedan bryts ut från det fysiska steget i från numer n på en ramp. Detta ger den mer generella ekvationen 18.

$$c_i = c_{i-1} - \frac{2 \cdot c_{i-1}}{4n_i+1} \quad (18)$$

Ekvationen beskriver då ett samband där ” n bestämmer accelerationen och inkrementeringen med i för konstant acceleration” [2]. Att starta från stop ger $n_i = i$ där $i = 1, 2, 3, 4 \dots$. Detta samband används även för att kunna retardera. m definieras som det totala antalet steg som ska genomföras, alltså skall motorn stanna på det m :te steget. Följdaktligen ger negativa n i ekvation 18 retardera (ty c_i växer när bråket blir negativt och därmed tiden mellan två steg längre). Genom att föra in $n_i = i - m$ återfås uttrycket för deceleration i ekvation 20. Acceleration med start från stillastående uttrycks som ekvation 19

$$c_i = c_{i-1} - \frac{2 \cdot c_{i-1}}{4i+1} \quad (19)$$

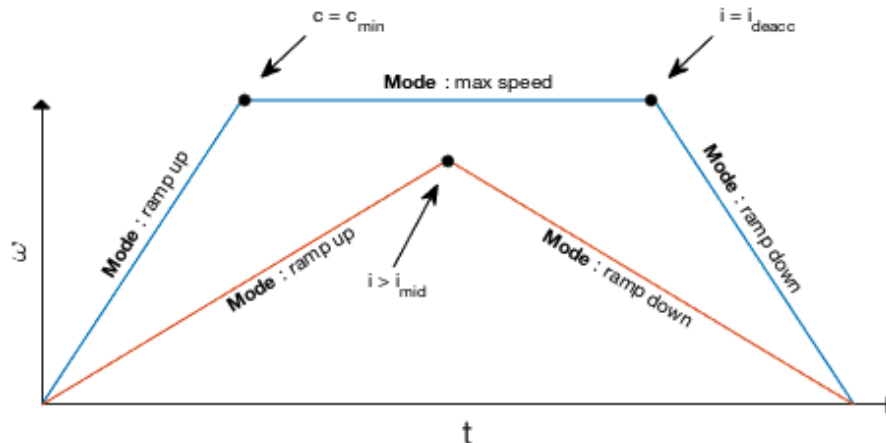
$$c_i = c_{i-1} - \frac{2 \cdot c_{i-1}}{4(i-m)+1}, \quad i < m \quad (20)$$

Genom detta resonemang finns nu en lösning för acceleration från stop och att decelera till stop. Vidare definieras en maxhastighet som det minsta värdet räknaren får anta, nämligen c_{min} . För att skriva en algoritm måste denna ta hänsyn till dessa gränser. Den behöver också veta antalet steg m och vart mitten i_{mid} på rampen är. Med psuedokod exemplifieras algoritmen nedan. ISR (Interrupt Service Routine) är den funktion som körs iterativt. I varje iteration kontrollerar algoritmen vilket ”mode” den borde vara i. Detta ”mode” bestämmer även vilken ekvation som skall användas. Längden mellan interrupten sätts med värdet på c . Figur 30 illustrerar rampens funktion.

```

1 func init
2     i = 0
3     m = target_pos - current_pos;
4     i_mid = m/2;
5     c_0 = f * sqrt( 2 * a / w' );
6     mode = ramp_up;
7     c = c_0;
8     setCounterISR(c);
9 end init
10
11 func ISR
12     select mode
13         ramp_up:
14             if (c < c_min)
15                 mode = max_speed;
16                 i_deacc = m-i;
17             end
18             if (i > i_mid)
19                 mode = ramp_down;
20             end
21             c = c - (2*c)/(4i+1);
22             break;
23
24         ramp_down:
25             if ( i == m)
26                 mode = stop
27             end
28             c = c - (2*c)/(4(i-m)+1);
29             break;
30
31         max_speed:
32             if (i == i_deacc)
33                 mode = ramp_down;
34             end
35             c = c;
36             break;
37
38         stop:
39             stopISR();
40             return;
41     end select
42
43     takeAstep();
44     setCounterISR(c);
45
46 end ISR

```



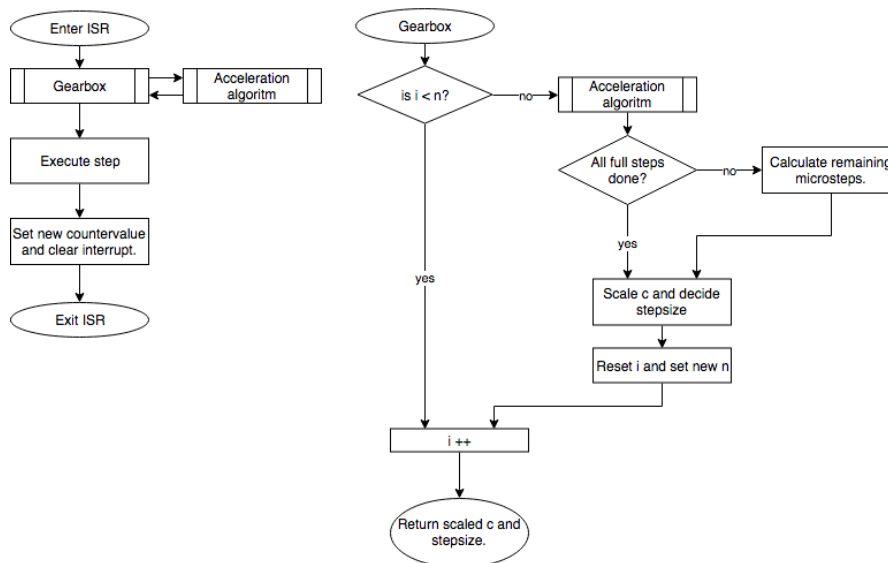
Figur 30: Algoritmen skapar en ramp när motorn rör sig mellan två punkter. Figuren illustrerar två olika scenarion, ett där max hastighet uppnås och en med långsammare acceleration och accelerationen avbryts för att stanna vid slutet.

3.3.2 Varierad excitation

Som det nämndes i inledningen till föregående del är algoritmen anpassad för att ta steg med en bestämd längd. Att konstant ta helsteg ger en ryckig gång, särskilt vid låga hastigheter, samt sänker precisionen. Att ta mikrosteg (ner till $1/64$ dels steg) höjer upplösningen, ger en jämnare gång men försämrar prestandan vid högre hastigheter. Detta för att den absoluta topphastigheten kommer att begränsas av hur snabbt mikrokontrollern kan exekvera.

Detta motiverar till att vilja kunna variera steglängden. Initialt gjordes därför ett försök att skriva om algoritmen för att passa varierade steglängder. Det visade sig dock komplicerat och svårt, istället föddes idén till att göra det modulärt. Grundidén är att algoritmen räknar ut rampen för hur stegen ska exekveras vid helsteg. Som en modul mellan algoritmen och funktionen som faktiskt stegar fram motorn läggs ytterliggare en modul till som bestämmer stegstorlek utefter den aktuella hastigheten, eller snarare storleken på c eftersom den är proportionell mot hastigheten. Modulen verkar som en slags växellåda och tar mindre steg när hastigheten är låg och större steg när hastigheten är hög. Funktionen döptes således till "gearbox" och har dessutom till uppgift att exekvera de sista mikrostegen om förflyttningen har högre upplösningen än kompletta helsteg.

Figur 31 visar schematiskt hur funktionen är uppbyggd, och hur gearbox faktiskt fungerar. Hela programmet drivs framåt av en ISR, vilket innebär att ett avbrott sker på mikrokontrollern. Avbrottet är i detta fall triggat från en timer. I korta drag förklaras förloppet enligt följande. Ett avbrott har skett (Enter ISR). Iterationsräknaren i jämförs med n som är antal mikrosteg som skall exekveras innan accelerationsalgoritmen körs igen. Accelerationsalgoritmen i sin tur kommer att returnera ett värde c för timern som styr ISR-anropen. Det första Gearbox gör efter accelerationsalgoritmen är att testa om motorn är framme vid sitt mål. Om detta är fallet kommer antalet mikrosteg som är kvarvarande att räknas ut. I normalfallet är inte alla helsteg exekverade och då kommer returvärdet c att avgöra hur stora steg som skall tas (kom ihåg att $c \sim 1/\omega_\theta$ dvs. hastigheten) samt hur många mikrosteg det motsvarar. n blir därför antalet steg och i nollställs. Till exempel bestäms att $1/4$ steg är en lämplig steglängd för den aktuella hastigheten. Då måste också 4 stycken $1/4$ steg utföras för att motsvara ett helsteg, vilket innebär att n sätts till 4.



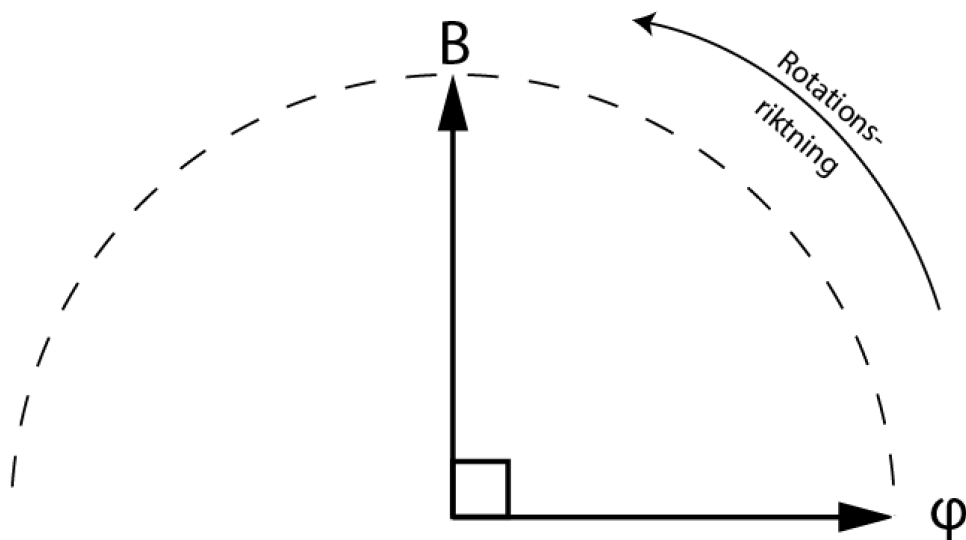
Figur 31: Schematisk bild som förklarar gearboxfunktionen och hur ISR rutinen är uppbyggd.

3.4 Closed-Loop experiment

En väldigt enkel closed-loop har utvecklats. Storheten som regleras är strömstyrkan och eftersom strömmen är proportionell mot vridmomentet i det ideala fallet, se del 2.1.3, regleras vridmomentet genom denna metod.

För att kunna sätta strömstyrkan används strömstyrningen som beskrivs i början av denna del. Genom denna metod går det att sätta strömstyrkan samt riktningen i det elektriska varvet φ . Vidare har det beskrivits i del 2.1.3 om vridmoment att det högsta vridmomentet uppstår när det statiska felet är 90° på det elektriska varvet.

Genom att läsa av det mekaniska varvet Θ går det att räkna om det till elektriskt varv φ enligt del 2.1.2. Denna beräkning kräver att man känner till fasskillnaden mellan Θ (avläst vinkel) och θ (faktisk vinkel). Fasskillnaden har här beräknats för just den motor som användes i experimentet genom avläsningar på kända punkter i det elektriska varvet, i detta fall motorns helsteg. När denna skillnad var känd kunde mikrokontrollen i realtid läsa av Θ och beräkna φ . När φ var känd kunde denna data användas för att alltid se till att det statiska felet hölls till 90° , vilket illustreras i figur 32 och gör att motorn ”jagar” det magnetiska fältet. Eftersom storleken på det magnetiska fältet kan styras genom strömstyrkan fås ett sätt att reglera motorns vridmoment. Detta gjordes genom att köra en ISR med känd frekvens där motorns elektriska fält kontinuerligt uppdaterades med hjälp av den senaste avläsningen som skedde i samma rutin. Eftersom att fältets riktning alltid uppdateras utefter avläsningarna kan motorn inte falla ur synkronisation. Denna metod visade sig fungera och resultaten är presenterade i del 4.3.



Figur 32: Det magnetiska fältet läggs före motorns position på det elektriska varvet.

4 Resultat

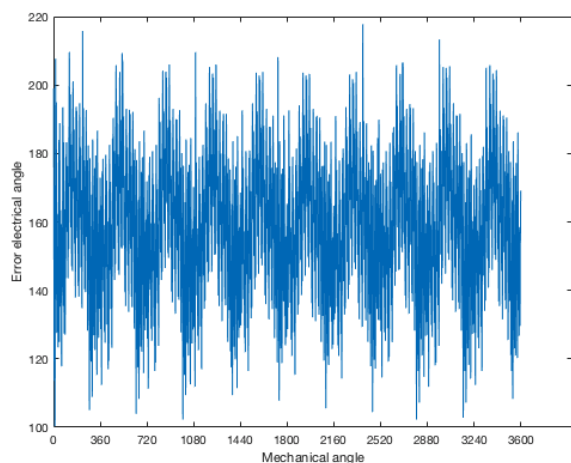
Nedan kommer de resultat som uppnåtts under projektets gång presenteras. Det beskrivs även hur de uppfyller projektets mål.

4.1 Sampling av Vinkelgivardata

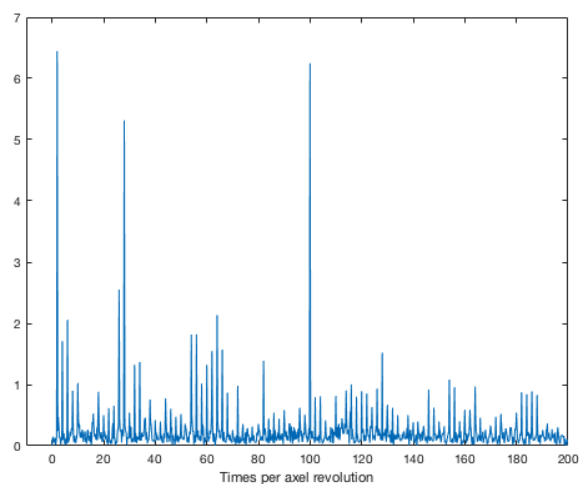
För att kunna utvärdera vinkelgivarens funktion har den samplats under olika förhållanden. Nedan presenteras i figurer mätningar där dels steglängden hos stegmotorn har varierats samt tiden mellan samplingarna, det vill säga hastigheten. För att förtydliga förfarandet så har rutinen sett ut som följer:

- Först tar motorn ett steg
- Väntar en bestämd väntetid
- Hämtar värdet från encodern

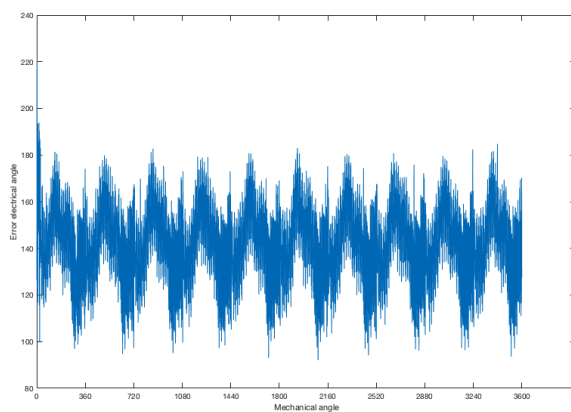
Hastigheten bestäms indirekt genom steglängd och väntetid. Under alla försök nedan är väntetiden 1 ms och motorn körs i 10 mekaniska varv. Figurerna till vänster visar vinkelfelet på det elektriska varvet jämfört med det uppmätta, här antas stegmotorn ta exakta steg och används därför som referens. Figurerna på höger sida visar en FFT utav samma mätningar, x axeln har normaliserats där frekvensen är per mekaniskt varv istället för per sekund. I figur 33 till 38 presenteras de olika mätningarna. Syftet är att sedan kunna använda denna data för att kalibrera vinkelgivaren i förhållande till motorn.



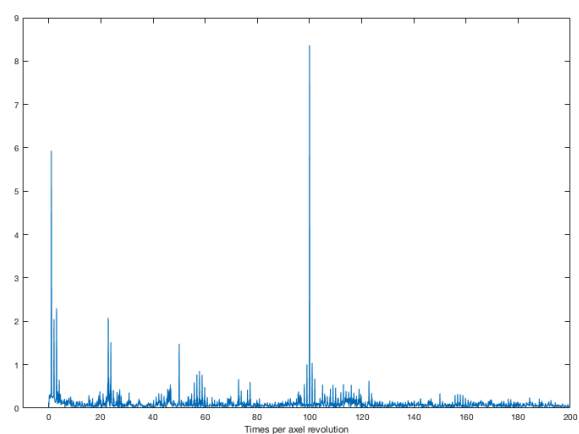
Figur 33: Steglängd: helsteg, väntetid: 1ms, längd: 10 varv.



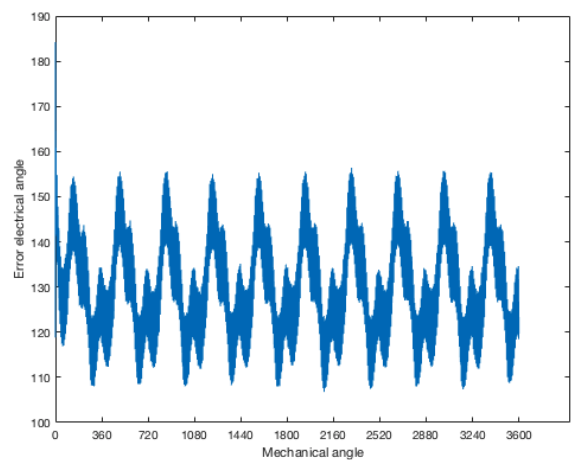
Figur 34: FFT av figur 33.



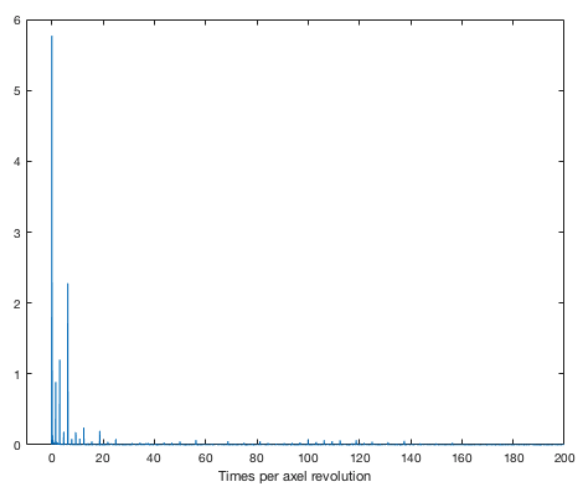
Figur 35: Steglängd: halvsteg, väntetid: 1ms, längd: 10 varv.



Figur 36: FFT av figur 35.



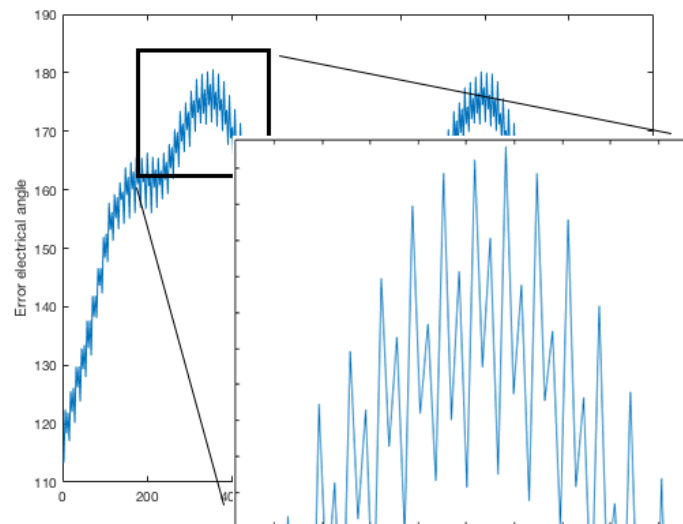
Figur 37: Steglängd: mikrostep, väntetid: 1 ms, längd: 10 varv



Figur 38: FFT av figur 37

Oavsett hastighet eller steglängd återkommer vissa komponenter, det går både att se i rumsdomän och i frekvensdomän. Man kan dela in dessa komponenter i två skilda delar, dels de som verkar återkomma i varje steg, alltså de som är periodiska med det elektriska varvet och dels de som verkar återkomma i varje varv, alltså de som är periodiska i det mekaniska varvet. Detta tydliggörs i figur 39. En intressant observation är även att bruset från vinkelgivaren verkar vara beroende av motorhastighet snarare än samplingshastighet, vilket man kan se om man jämför de olika samplingarna.

Detta resultat uppfyller inga av projektets mål i sig men är ett viktigt delresultat för att kunna skapa återkoppling i systemet.



Figur 39: Vissa utav frekvenskomponenterna verkar vara periodiska med det elektriska varvet.

4.2 Mätningar av acceleration

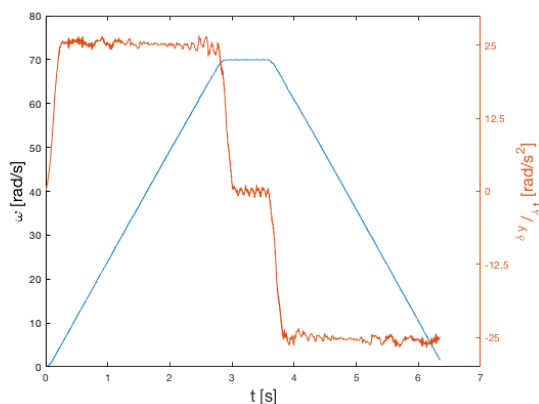
En del av projektet som fullföljdes var att skapa ett sätt att ta sig mellan två positioner utan att tappa steg. Genom en accelerationsalgorithm, som också finns beskriven i kapitel 3.3.1, kan motorn köras med en förutbestämd acceleration, hastighet samt sträcka. För att testa att funktionen fungerar som förväntat har undersökningar gjorts på två sätt. Det första sättet har varit genom att använda vinkelgivaren och läsa av absolutvinkeln i en given frekvens (som var 333 Hz). På detta sätt har datan, med hjälp av MatLab, kunnat tolkas för att framställa graferna nedan till vänster. Det bör även nämnas att signalen har blivit filtrerad genom att medelvärdesbilda de 50 senaste datapunkterna.

Det andra sättet har varit att skriva ut den data som algoritmen faktiskt producerar, alltså de värden den sätter på timer och "växel". Datan kommer således från varje iteration av algoritmen. Denna data bör tolkas som det sätt mikrokontrollern räknar ut hur den ska köra. Även denna data har tolkats genom MatLab för att skapa graferna nedan till höger där beräknad hastighet och "växel" visas. Dessa två grafer kan då jämföras som ett bör värde och ett är värde.

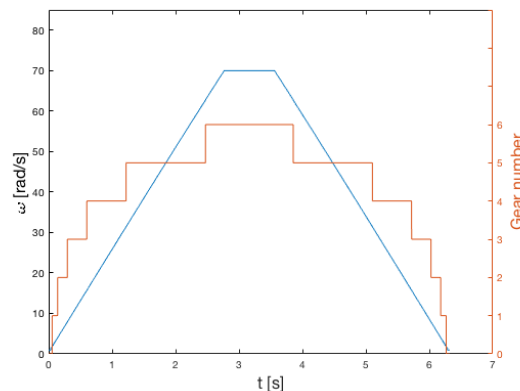
För att bevisa funktionen har två testfall skapats. I båda fallen så är maxhastigheten inställd till 70 rad/s samt accelerationen 25 rad/s². I det första fallet demonstreras hur algoritmen beter sig när den når maxhastighet. Dessutom används alla 6 steglängder ("växlar"). Detta visas i figurerna 40 och 41. Som det framgår i figuren stämmer både den verkliga och den beräknade grafen överens med varandra och dessutom är värdena de förväntade enligt ovan.

I det andra testfallet demonstreras istället funktionen när algoritmen aldrig når topphastigheten. Istället avbryts accelerationen och motorn börjar retardera för att stanna på det förväntade målet. Detta visas i figurerna 42 och 43.

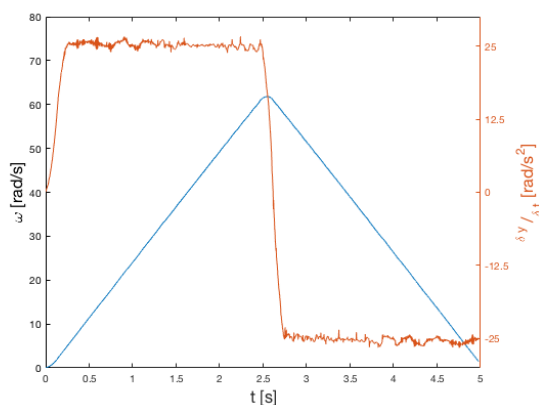
Inom ramen för de uppsatta målen om att kunna gå mellan två olika bestämda positioner samt göra detta utan att tappa synkronisation anses resultatet vara lyckat.



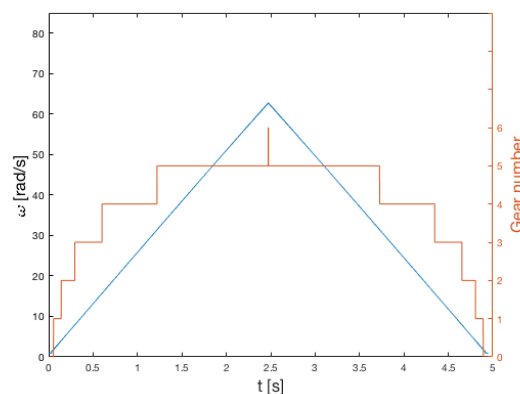
Figur 40: Sampling av vinkelgivare i testfall 1.



Figur 41: Data från algoritm i testfall 1.



Figur 42: Sampling av vinkelgivardata i testfall 2.



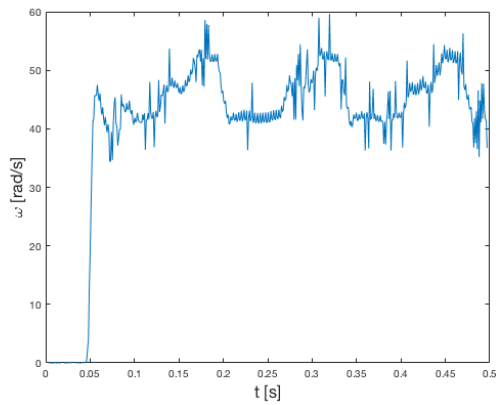
Figur 43: Data från algoritm i testfall 2.

4.3 Closed-Loop

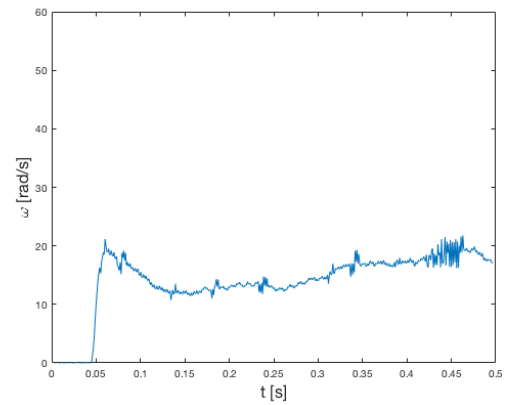
Ett av målen med projektet var att kunna köra motorn i closed-loop. På grund av tidsbrist så fullföljdes inte alla delar av detta moment. Det utvecklades dock en primitiv återkopplande funktion som är beskriven i kapitel 3.4. Tanken med denna funktion är att den reglerar vridmomentet. Nedan presenteras vissa mätningar där stegsvar utförs med och utan last. Datan har filtrerats genom att medelvärdesbilda över fem datapunkter för att ta bort en del av bruset. Detta visualiseras i figur 44 och 45, samplingshastigheten har i båda fallen varit 1 kHz. Lasten består av en kortsluten DC motor men det exakta vridmomentet är dessvärre ej känt varför storheter ej kan presenteras. Dock kan man se att med samma kontrollsignal stabiliserar sig motorns hastighet vid olika lägen. Detta är något som tyder på att kontrollsignalen faktiskt påverkar vridmomentet. Det finns dock inga beräkningar som understödjer detta och det är endast en teori.

En annan notering om motorns drift i Closed-Loop är dels att driften är ojämn och upplevs som ”oscillerande” över varvet. Detta tros bero på att vinkelavläsaren ger ett fel som varierar över varvet (se tidigare avsnitt i denna del). Eftersom vridmomentet som utvecklas i motorn är beroende av det statiska felet mellan motorns faktiska position och den elektriska vinkeln är den även känslig för när mätningen ej stämmer. I t.ex. figur

37 varierar den avlästa elektriska vinkeln $\pm 25^\circ$. Detta innebär även att vridmomentet varierar ungefär med $T = T_{pk} \cdot \sin(\pi/2 \pm 25^\circ)$ vilket innebär $\pm 10\%$ i praktiken. Oscilleringarna går även att se framåt slutet av stegsvarstestet i figur 44. En annan notis är även att maxhastigheten tycks vara kring 45 rad/s. Återkopplingen kompenserar inte för den L/R-verkan som beskrivs i både del 2.6.1 samt 2.1.3. Dessutom blir Back-EMK betydande vid dessa varvtal och en kombination av dessa reducerar vridmomentet.



Figur 44: Sampling av stegsvar utan last.



Figur 45: Sampling av stegsvar med last.

5 Förslag på vidareutveckling

Under projektets gång har många idéer uppstått om hur det skulle kunna föras vidare. Syftet med denna del är att belysa för företaget hur det skulle kunna gå vidare med utvecklingen av sin produkt. Därför belyses i punktform olika delar som borde få mer uppmärksamhet.

- **Kalibreringsrutin.** Genom resultaten i del 4.1 borde en kalibreringsrutin utvecklas. Dels är det önskvärt att hitta δ . I princip är δ medelfelet i samplingen som presenteras i samma del. δ kommer att vara olika för alla motorer som drivaren monteras på och kommer att bero på hur magneten är monterad. Likaså behövs en linjärisering ske enligt de metoder som beskrivs i den tekniska bakgrunden. Kalibreringen bör dessutom sparas i ett icke-volatilt minne hos mikrokontrollen. Detta för att slutanvändaren endast skall behöva kalibrera en gång och inte vid varje uppstart. Vinkelavläsaren har dessutom inbyggda kalibreringsfunktioner. Dessa har använts i detta arbete men kan komma i konflikt med en framtida kalibreringsrutin.
- **PID regulator.** En naturlig del av det fortsatta arbetet blir att försöka uppfylla de mål som detta projekt inte nådde upp till. Några observationer som har gjorts är bruset som uppstår i vinkelavläsaren. Därför förutspås att ett bra digitalt filter behöver konstrueras för att kunna bygga upp en bra regulator. Då naturen av systemet är ganska enkel när väl en stabil återkoppling finns rekommenderas en PID regulator. Regulatorn bör vara uppbyggd i olika delar. En design idé som har funnits är regulatorer som reglerar regulatorer, det vill säga kaskadreglering. För att förtydliga, en nedre regulator styr vridmoment. Börvärdet för vridmomentet kommer från en hastighetsregulator. Hastighetsregulatorn i sin tur får sitt börvärde från en positionsregulator. Då möjliggörs det att slutanvändaren kan gå mellan de olika "modes" som nämns i målen av denna rapport.
- **Hastighetsreglering i open-loop.** Ett delmål som förbisågs i projektets slut var att skapa hastighetsreglering i open-loop. Självklart skulle motorn lätt kunna köras i bestämda hastigheter men svårigheten ligger i att utveckla ett skydd för att överstegra motorn och därmed tappa synkronisation. Lösningen spås vara att återanvända accelerationsalgoritmen men att bygga om dess "state-machine" på ett sådant sätt att den går att använda utan att känna till dess målposition.
- **Hårdvara.** I nuläget så finns strömshuntar på varje utgång från H-bryggorna vilka återkopplas tillbaka till IC-kretsen själv. I det closed-loop experiment som gjordes togs inspiration från kontrollmetoden FOC (Field Oriented Control). En viktig byggsten som saknas för denna metod är dock att känna till strömmen i realtid. Därför föreslås att återkoppla strömschuntarna direkt till mikrokontrollern. På mikrokontrollern finns två stycken ADC men en av dessa är idag kopplad för att mäta matningsspänningen till kretsen. Förslaget är att man istället använder båda ADC för att övervaka fasströmmarna. Förmodligen behövs även operationsförstärkare för att möjliggöra detta.

6 Appendix

Listing 1: Ett utdrag från källkoden. Funktionen `currentStep()` hämtar värden från en LUT och uppdaterar inställningen på H-bryggorna.

```
1  /*
2  * Sets the electrical angle.
3  * Note: 360 degree divided into 0-255!
4  */
5  void currentStep(unsigned short angle, unsigned char power){
6      unsigned char quadrant = angle/N;
7      unsigned char dacA, dacB = 0;
8
9      switch (quadrant){
10         /*
11         * First quadrant
12         */
13         case 0 :          dacA = currentLUT[power][M-(angle%N)]; //cosine
14                          dacB = currentLUT[power][angle%N]; //sine
15
16                          dac_output_voltage(DAC_A, dacA);
17                          dac_output_voltage(DAC_B, dacB);
18
19                          //Coil A positive
20                          gpio_set_level(motor_pin_1, HIGH);
21                          gpio_set_level(motor_pin_2, LOW);
22                          //Coil B positive
23                          gpio_set_level(motor_pin_3, HIGH);
24                          gpio_set_level(motor_pin_4, LOW);
25
26                          break;
27
28         /*
29         * Second quadrant
30         */
31         case 1 :          dacA = currentLUT[power][angle%N]; //cosine
32                          dacB = currentLUT[power][M-(angle%N)]; //
33                          sine
34
35                          dac_output_voltage(DAC_A, dacA);
36                          dac_output_voltage(DAC_B, dacB);
37
38                          //Coil A negative
39                          gpio_set_level(motor_pin_1, LOW);
40                          gpio_set_level(motor_pin_2, HIGH);
41                          //Coil B positive
42                          gpio_set_level(motor_pin_3, HIGH);
43                          gpio_set_level(motor_pin_4, LOW);
44
45                          break;
46
47         /*
48         * Third quadrant
49         */
50         case 2 :          dacA = currentLUT[power][M-(angle%N)]; //cosine
51                          dacB = currentLUT[power][angle%N]; //sine
52
53                          dac_output_voltage(DAC_A, dacA);
54                          dac_output_voltage(DAC_B, dacB);
55
56                          //Coil A negative
57                          gpio_set_level(motor_pin_1, LOW);
58                          gpio_set_level(motor_pin_2, HIGH);
59                          //Coil B negative
60                          gpio_set_level(motor_pin_3, LOW);
61                          gpio_set_level(motor_pin_4, HIGH);
62
63                          break;
64
65         /*
```

```

65         * Fourth quadrant
66         */
67     case 3 :          dacA = currentLUT[power][angle%N]; //cosine
68                     dacB = currentLUT[power][M-(angle%N)]; //
                        sine
69
70
71                     dac_output_voltage(DAC_A, dacA);
72                     dac_output_voltage(DAC_B, dacB);
73
74                     //Coil A positive
75                     gpio_set_level(motor_pin_1, HIGH);
76                     gpio_set_level(motor_pin_2, LOW);
77                     //Coil B negative
78                     gpio_set_level(motor_pin_3, LOW);
79                     gpio_set_level(motor_pin_4, HIGH);
80
81                     break;
82     }
83 }

```

Referenser

- [1] P. P. Acarnley och Institution of Electrical Engineers. *Stepping motors : a guide to theory and practice*. Institution of Electrical Engineers, 2002, s. 170. ISBN: 9780852960295. URL: <http://library.books24x7.com.proxy.lib.chalmers.se/toc.aspx?site=Y7V97%7B%5C%7Dbookid=7145>.
- [2] D. Austin. Generate stepper-motor speed profiles in real time. *EET India* January (2005), 1–5. URL: <http://www.eetindia.co.in/articleLogin.do?artId=8800503136%7B%5C%7DfromWhere=/ART%7B%5C%7D8800503136%7B%5C%7D1800000%7B%5C%7DDTA%7B%5C%7Da4834a4e.HTM%7B%5C%7DcatId=1800000%7B%5C%7DnewsType=TA%7B%5C%7DpageNo=null%7B%5C%7Dencode=a4834a4e>.
- [3] P. R. (ed). *Magnetic Sensors and Magnetometers*. 2002. DOI: 10.1088/0957-0233/13/4/707. URL: <http://stacks.iop.org/0957-0233/13/i=4/a=707?key=crossref.f10181e26a4f8b380a7c230bd0bc>
- [4] T. Kenjo och A. Sugawara. Stepping Motors and Their Microprocessor Controls, Second Edition. *European Journal of Engineering Education* **20.3** (1995), 386. ISSN: 14695898. DOI: 10.1080/03043799508928291.
- [5] E. Ramsden. *Hall-Effect Sensors*. 2006. ISBN: 9780750679343. DOI: 10.1016/B978-0-7506-7934-3.X5000-5.
- [6] C. Reig, S. Cardoso de Freitas och S. Chandra Mukhopadhyay. *Giant Magnetoresistance (GMR) Sensors. From Basis to State-of-the-Art Applications*. 2013, s. x, 300. ISBN: 978-3-642-37171-4. DOI: 10.1007/978-3-642-37172-1. URL: <http://dx.doi.org/10.1007/978-3-642-37172-1>.