

Realisering og utprøving av hastighetsmåler for tredemølle

**Sven Verner Niklas
Bengtsson**

Master i teknisk kybernetikk og robotikk
Innlevert: juni 2014
Hovedveileder: Øyvind Stavadahl, ITK

Norges teknisk-naturvitenskapelige universitet
Institutt for teknisk kybernetikk

Abstract

The goal of this thesis is to build the velocity measurement device thought up, and theoretically developed, in the project work, last semester [4]. The objective of the velocity measurement device is to simplify monitoring and logging of treadmill velocities during scientific tests. It is a three part assignment, firstly identifying parts and components and mathematically acquiring their values. Secondly the mechanical parts needed to be designed, developed and assembled together with the electronics. Finally the system needs to be tested and validated for use with the institute for movement science's equipment.

The choices made, were made based on the specifications, availability and mathematical equations. The rotary encoder and the microcontroller that was chosen required that two voltage dividers and a differential amplifier was made, and that the voltage supply provided a negative voltage. After design, mathematically defining component values and testing, the system had to be assembled. With a mount for the rotary encoder and a container for the electronics. This was done in collaboration with the institutes mechanical workshop.

The result is shown on the front page and in the results-section (3.1) (see figure 3.1). The rotary encoder is attached to the treadmill by strong magnets, and is pressed to the moving surface with a spring. The container has a BNC-contact to interface with the Institute of movement science's equipment. The microcontroller is also available for resetting and other adjustments.

The full scale tests were mainly successfull, with some small hitches. There was a spacing issue with the mounting device, as it should have been higher. The values read were a bit off, and the higher the speed, the higher the error. This was probably due to a programming issue, where a comma was used instead of a decimal point.

Sammendrag

Denne oppgaven har som mål å sette til liv hastighetsmåleren som ble tenkt opp og teoretisk utarbeidet i prosjektoppgaven [4]. Denne skal gjøre det enklere å overvåke og logge hastigheten til en tredemølle ved laboratorieforsøk på Institutt for bevegelsesvitenskap. Oppgaven er tredelt, med identifikasjon av deler og komponenter først, inkludert matematiske beregninger av komponentverdier. Deretter skal det kjøpes inn, utvikles og settes sammen fysiske komponenter, programmering, testing og lodding av kretskort skjer også her. Til sist skal systemet testes og valideres for bruk mot Institutt for bevegelsesvitenskap sine instrumenter.

Valgene som ble gjort, ble tatt på bakgrunn av kravspesifikasjon, tilgjengelighet og matematiske beregninger. Valg av rotasjonsomformer og mikrokontroller førte til krav om spenningsdeler og en differensialforsterker og en spenningsforsyning med negativ spenning. Etter design, matematisk beregning av komponentverdier og testing, måtte systemet settes sammen med et oppheng for rotasjonsomformeren og en beholder for elektronikken. Dette ble gjort i samarbeid med instituttets mekaniske verksted.

Resultatet er vist på forsiden og i resultatseksjonen (3.1) (se figur 3.1). Rotasjonsomformeren festes med sterke magneter til tredemøllen og blir presset mot det bevegelige underlaget av en fjær. Beholderen har en kontakt tilgjengelig for å koble til Institutt for bevegelsesvitenskap sitt utstyr. Mikrokontrolleren er også tilgjengelig for nullstilling og endringer.

Fullskalatesten gikk hovedsaklig veldig bra, men det var noen små feil. Det oppsto problemer med festemekanismen da den kom borti båndet, dette burde ha vært løst ved å ha festet høyere. Det var også noen feilaktige verdier lest av, noe som mest trolig skyldes en programmeringsfeil, der et komma ble benyttet i stedet for et punktum.

Forord

Denne oppgaven har vært lærerik for meg. Det å gjøre ting i praksis og å få det til er heller uvanlig. I begynnelsen var det mye frem og tilbake, uten noe særlige fremskritt og spesielt diskusjonen om hvordan å få til forsterkingen, og dermed hvilken spenningsforsyning som måtte kjøpes, tok uproporsjonalt lang tid. Etter at alt var gjort, så var det lodding av kretskort som tok mye lenger tid enn jeg hadde sett for meg. Dette ga nærmest ingen tid til fullskalatesting. Om ikke annet, så har dette gitt meg uvurderlig erfaring i hvorfor man skal jobbe effektivt i starten av et prosjekt, selv om det på papiret ser ut som om man har god tid. Selve byggingen og utviklingen har ellers gått relativt knirkefritt.

Takk

Jeg vil rette et stort takk til Øyvind Stavdahl. Som veilederen min har han vært en uvurderlig støttespiller og en til å stole på da jeg selv var usikker. Jeg vil også takke for alle de mekaniske konstruksjonene, som jeg hverken har kunnskapen eller verktøyene til å få til selv. Resultatet han fikk til utifra mine evneveike tegninger var langt bortenfor det jeg kunne håpe på. Omega Verksted, Torbjørn Alvestrand og Trond Suleng må takkes for deres utlån av ekspertise og utstyr når jeg trengte det. Jeg vil til sist takke Suzanne N. Larsen for hjelp med språket og Britt O. Knutsen for å gi meg et spark der bak når jeg trengte det.

Innhold

1	Innledning	1
1.1	Innledning	2
1.1.1	Innledning	2
1.1.2	Oppgavetekst	3
1.2	Bakgrunn	4
1.2.1	Forprosjekt	4
1.2.2	Kravspesifikasjon	5
2	Bakgrunnsteori, valg av deler og utførelse	6
2.1	Teori	7
2.1.1	Kretskort/Elektronikk	7
2.1.2	Programmering	13
2.2	Valg av maskinvare	15
2.2.1	Rotasjonsomformer	15
2.2.2	Mikrokontroller	17
2.2.3	Op-Amp	18
2.2.4	Strømforsyning	19
2.3	Utførelse	20
2.3.1	Formler	20
2.3.2	Identifikasjon, design og ferdigstilling av krets- kort	21
2.3.3	Design og konstruksjon av mekaniske deler	29
2.3.4	Programvare og kode	34
3	Resultat, diskusjon og videre arbeid	38

3.1	Resultater	39
3.1.1	Hastighetsmåleren, arkitektur og brukergrensesnitt	39
3.2	Diskusjon	44
3.3	Fremtidig arbeid	46
3.3.1	Mangler i henhold til kravspesifikasjon	46
3.3.2	Mangler eller forbedringspotensiale	47
3.3.3	Egne tilleggsidéer	49
A	Forkortelser	53
B	Prosjektoppgave	54
C	Bruksanvisning og guide til demontering/montering	55
D	Komponentliste	56
E	Verdier fra fullskalatesten over to minutter	57
F	Arduino kildekodefil	58

Figurer

1.1	Legomodell av endelig løsning	4
2.1	Kretsskjema	7
2.2	Signalskjema	8
2.3	Bipolar transistor	9
2.4	Åpen kollektor	10
2.5	Spenningsdeler	10
2.6	Differensialforsterker	11
2.7	Digital til Analog-omformer [17]	12
2.8	Diagram over kvadraturpulser	13
2.9	Puls-Bredde-Modulasjon	14
2.10	Omron E6C2-CWZ5B	15
2.11	Arduino DUE	17
2.12	Traco Power spenningsforsyning	19
2.13	Den ferdige spenningsdelingen	22
2.14	Den ferdige differensialforsterkeren	25
2.16	“Breadboard”	26
2.15	Det ferdige kretskortet	27
2.17	Koblinger på Arduino DUE-mikrokontrolleren	28
2.18	Ikke-inverterende forsterker	29
2.19	Oppheng	31
2.20	Beholderen	33
3.1	Den endelige måleenheten	39
3.2	Kretskort med forklaringer	40
3.3	Resultatene fra fullskalatesten over 2 minutter	42

3.4	Utsnitt av resultatene ved 0V	43
3.5	Feilene som oppsto ved fullskalatesting	48

List of Equations

2.1	Spenningsdeling	11
2.2	Differensialforsterker	11
2.3	Digital til Analog-omformer	12
2.4	Puls-Bredde-Modulasjon	14
2.5	Minimal hastighet	16
2.6	Maksimal hastighet	16
2.7	Hastighet	21
2.8	Spenning	21
2.9	Spenningsdeling, med beregnede verdier	22
2.10	Spenningsdeling, utgangsverdi	22
2.11	Spenningsnivå	23
2.12	Ikke-inverterende forsterker og spenningsdeling	24
2.13	Konstante til utregning av differensialforsterker	25
2.14	Differensialforsterker, første del	25
2.15	Differensialforsterker, andre del	26
2.16	Differensialforsterker, bevis	26
3.1	Hastighet per volt	42
3.2	Hastighet gitt av spenningen	42

Kapittel 1

Innledning

1.1 Innledning

1.1.1 Innledning

Konstruksjon av instrumentet krever først og fremst valg av deler. Spesielt viktig er rotasjonsomformer, strømforsyning, mikrokontroller og operasjonsforsterker (Op-Amp). Etter at hovedbestanddelene er valgt, må systemet identifiseres. Det må bestemmes hvilke kretser som trengs, hvilke koblinger og hvilken funksjonalitet må være med for å tilfredsstille kravspesifikasjonen (se seksjon 1.2.2). Disse to stegene er avhengige av hverandre. For eksempel er det viktig ved valg av strømforsyning at det er gjort nøye rede for hvilke spenningsnivåer Op-Amp'en trenger. Siden det må leveres $+5 - -5V$ (se seksjon 1.2.2), må også strømforsyningen levere negativ spenning. Tett etterfulgt identifiseringen av systemet kommer bestemmelse av komponenter, med verdier basert på matematiske utregninger. Mikrokontrolleren må også programmeres, slik at hastigheten beregnes og sendes ut korrekt. Når systemet er fungerende og alle komponenter valgt, skal også en fysisk konstruksjon designes. Den krever en feste-anordning for å feste sensoren til tredemøllene og en boks eller et hus for komponentene. Denne konstruksjonen blir gjort i samarbeid med instituttets mekaniske verksted. Når valg av komponenter, lodding av kretskort, programmering av mikrokontroller og konstruksjonen er ferdig, må systemet testes. Systemet skal fungere opp mot Institutt for bevegelsesvitenskap sin "Kistler's 64 ch DAQ system (model 5695A)" [13].

1.1.2 Oppgavetekst

Institutt for bevegelsesvitenskap, NTNU, bruker i sin forskning bl.a. tredemøller der forsøkspersonen løper på et bevegelig underlag med variabel hastighet og stigning. På grunn av den mekaniske konstruksjonen kan løpebanen gli i forhold til drivverket, og den reelle hastigheten til løpebanen er derfor ikke nøyaktig kjent. I et tidligere prosjekt har en etablert en funksjonsspesifikasjon og identifisert egnet måleteknikk. Du skal nå realisere et system basert på disse funnene og dokumentere systemets egenskaper.

1. Foreta, med utgangspunkt i resultater fra det tidligere prosjektet, et konkret og begrunnet valg av instrumentering og annen maskinvare for realisering av systemet. Gi en komplett beskrivelse av systemets maskin- og eventuelt programvare der arkitektur og alle relevante grensesnitt er dokumentert.
2. Beskriv systemets mekaniske deler, og få delene produsert og montert av eller i samarbeid med instituttets mekaniske verksted.
3. Implementer systemets elektroniske maskinvare og dets programvare. Kommersielt tilgjengelige komponenter anskaffes i samråd med veiledere.
4. Test systemet og diskuter resultatene i lys av foreliggende funksjonsmessige og tekniske spesifikasjoner.

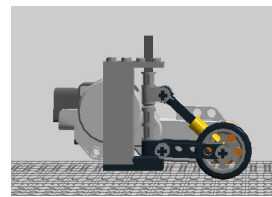
1.2 Bakgrunn

1.2.1 Forprosjekt

Bakgrunnen for denne oppgaven var å utvikle et mer egnet system for hastighetsmålinger av båndet til tredemøller. Mer spesifikt tredemøller brukt av Institutt for bevegelsesvitenskap. Tidligere har et system med fototachometer, og reflektorer teipet fast til båndet, blitt benyttet. Den store ulempen med dette er at reflektorene lett har blitt revet av midt under forsøket, og at forsøket dermed må gjentas. [4]

Det ble gjennomført et forprosjekt der ulike målemetoder ble veiet opp mot hverandre. De ble delt opp i to kategorier, kontaktløse og kontaktkrevende målemetoder. De reelle kandidatene var radar- og laserdoppler, rotasjonsomformer og bildebehandlingsverktøy som ser på et objekt som har blitt “dratt ut” eller “smurt utover” i fartsretningen, på bildet. Ved hjelp av kjennskap til blenderåpning, lukketid og objektets egentlige form, kan man utvikle formler som regner ut hastigheten til objektet. [4]

Forprosjektet ble avsluttet med et begrunnet valg av endelig løsning og en skissering av oppsettet. Valget ble tatt med hensyn til pris, gjennomførbarhet og hvor tilgjengelig teknologien var. Radardoppler ble tidlig luket ut da kravene til oppløsning og hastighet ble endret for å omfatte flere ulike tredemøller, og de produktene som var tilgjengelige ikke klarte å oppfylle disse kravene. Laserdoppler ble luket ut på grunn av prisen, som var ti til tyve ganger så høy som for de andre kandidatene. Bildebehandlingsmetoden som ble vurdert, var fortsatt altfor eksperimentell til å være et reelt alternativ. Nærmere undersøkelser viste at det ikke fantes noen fullt fungerende instrumenter som benyttet denne målemetoden, og ei heller en god beskrivelse av hvordan å gjenskape resultatene fra artiklene



Figur 1.1: Legomodell av endelig løsning

som fantes på temaet. Den gjenstående kandidaten, og den som ble valgt med best resultat i alle kriterier, var rotasjonsomformereren. [4]

1.2.2 Kravspesifikasjon

Produktet skal:

1. Hente inn hastigheten til løpebåndet på en tredemølle
 - Hastighet mellom 0 m/s – 12 m/s
 - Nøyaktighet ned til 0.1 m/s
2. Ha festeanordning på tredemølle
3. Ha et databehandlingsverktøy som kan håndtere dataen og sende den ut i sanntid
4. Sende ut et -5 – 5V analogt signal, også i sanntid
5. Utgang med tilkobling for BNC-kontakt for tilkobling til Institutt for bevegelsesvitenskap sin teststasjon.

Produktet bør:

1. Ha et aksesspunkt på databehandlingsverktøyet for å hente ut rådata til
 - Beregning av distanse
 - Logging
2. Knapp for nullstilling (set signal output to "Zero"), dersom dette ikke gjøres automatisk

Kapittel 2

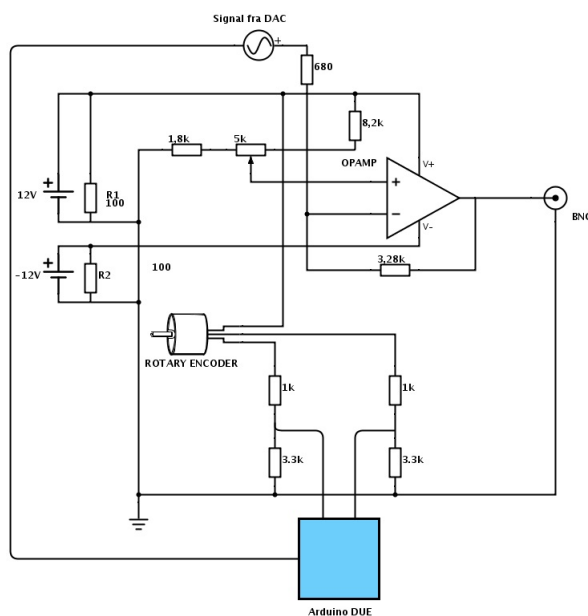
Bakgrunnsteori, valg av deler og utførelse

2.1 Teori

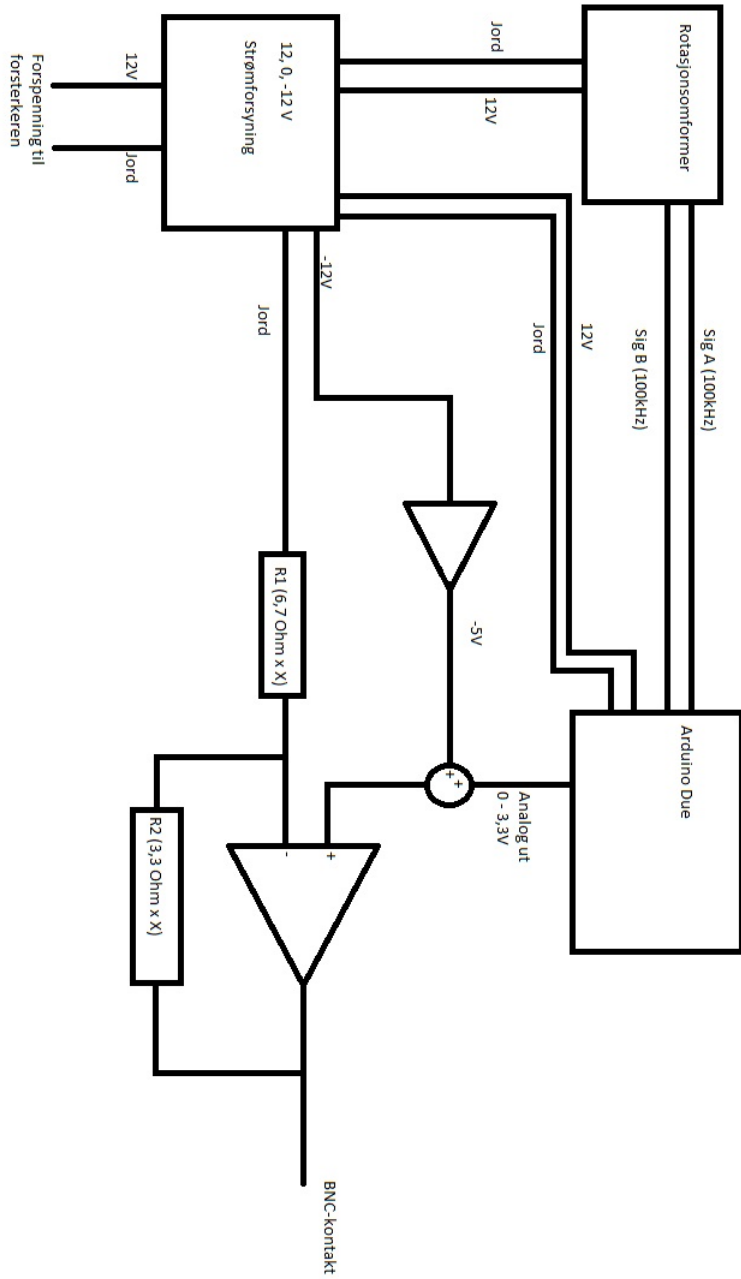
I prosessen med å lage et målingsverktøy for å gjennomføre hastighetsmålinger på tredemøller ble det identifisert fire hoveddeler. Disse var kretskort/elektronikk, strømforsyning, mikrokontroller og rotasjonsomformer og dens oppheng. For å fullføre dette prosjektet var det nødvendig med en digital til analog omformer, enten eksternt på kretskortet eller, som i dette tilfellet, integrert i mikrokontrolleren.

2.1.1 Kretskort/Elektronikk

Elektronikken som kreves for å få et fungerende system, som svarer til kravspesifikasjonen, er en spenningsdeler for inngangssignalet fra rotasjonsomformerens, en forsterkerkrets (differensialforsterker) og termineringsmotstandere med 100Ω ekvivalentmotstand.



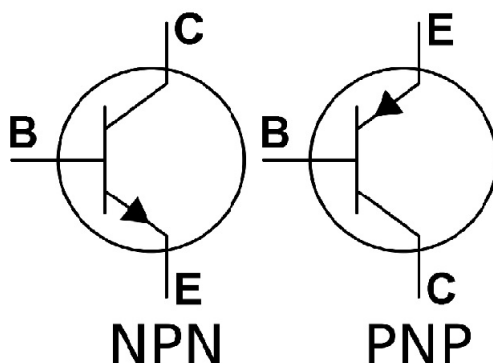
Figur 2.1: Kretsskjema



Figur 2.2: Signalskjema

Bipolare transistorer

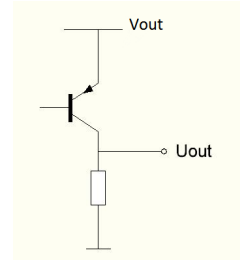
En bipolar transistor er bygget opp av to dioder, eller to “PN”-overganger, som da gir tre terminaler kalt kollektor, emitter og base. Basen er enten “N”- eller “P”-siden fra begge diodene, og de kalles da for “PNP” eller “NPN”, der den midterste bokstaven representerer basen. “PN”-overganger er en silisiumkrystall der ene siden er positivt ladd og den andre negativt ladd, men som en helhet er den nøytral. De to “PN”-overgangene kalles for Base-Emitter-dioden og Kollektor-Base-dioden (forkortet “BE” og “KB”). Spenningen over “BE”-dioden styrer oppførselen til transistoren. Når det blir satt spenning over “BE”-dioden, blir “PN”-overgangens balanse forstyrret og en strøm vil trekke gjennom den (retningen til strømmen kommer an på om det er “PNP” eller “NPN”). Dette vil forplante seg og åpne “KB”-dioden, slik at den sender ut en strøm (fra basen til kollektoren) i en “PNP”, eller trekker strøm (fra kollektoren til basen) i en “NPN”. [18]



Figur 2.3: Bipolar transistor

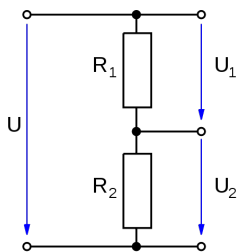
“Åpen kollektor”-utgang (PNP)

En “åpen kollektor”-utgang er vanlig å finne blant komponenter laget for innebygde kretser. Dette er en krets der utgangen er koblet til kollektoren i en bipolar transistor, mens utgangssignalet ligger over “BE”-dioden. Noe som fører til at det digitale signalet som man skal ha tak i, virker som en bryter for kollektorutgangen. Dersom transistoren er av typen “NPN”, kobler man en motstand parallelt til en spenningskilde. Når det ikke er spenning over “BE”-dioden, vil det ikke være en sluttet krets gjennom transistoren, og linjen “trekkes” høy. Dersom det er spenning over “BE”-dioden, sluttes kretsen gjennom transistoren ned til jord og linjen vil være lav. Motstanden kalles en “Pull up”. Dersom transistoren er av typen PNP, er fremgangsmåten den samme med unntaket at man kobler motstanden til jord, og hver gang det settes en spenning over “BE”-dioden går linjen høy. Med “PNP”-transistor kalles motstanden for en “Pull down”. [10]



Figur 2.4: Åpen kollektor

Spenningsdeler



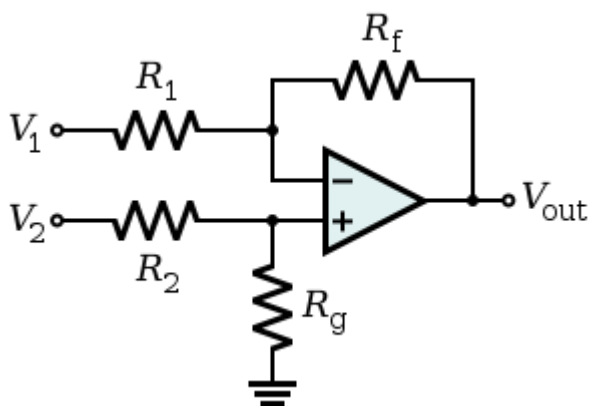
Figur 2.5: Spenningsdeler

Rotasjonsomformerer som benyttes har en utgang av typen “PNP open collector”. Spenningen den tilfører er 12V og siden mikrokontrollerens digitalpinner tåler maksimum 3.3V, må spenningen senkes. Dette gjøres ved hjelp av en spenningsdeler som består av to motstander i serie. Motstandene er koblet til to ulike spenningsnivåer og man henter ut spenningen over motstand R_2 . [18]

$$U_2 = U \cdot R_2 / (R_1 + R_2) \quad (2.1)$$

Differensialforsterker

Differensialforsterkeren består av en operasjonsforsterker og fire motstander. Forsterkeren virker på forskjellen mellom inngangene kontra å forsterke et av inngangssignalene. [18]



Figur 2.6: Differensialforsterker

$$V_{out} = V_2 \cdot \left(\frac{(R_f + R_1)R_g}{(R_g + R_2)R_1} \right) - V_1 \left(\frac{R_f}{R_1} \right) \quad (2.2)$$

Termineringsmotstand

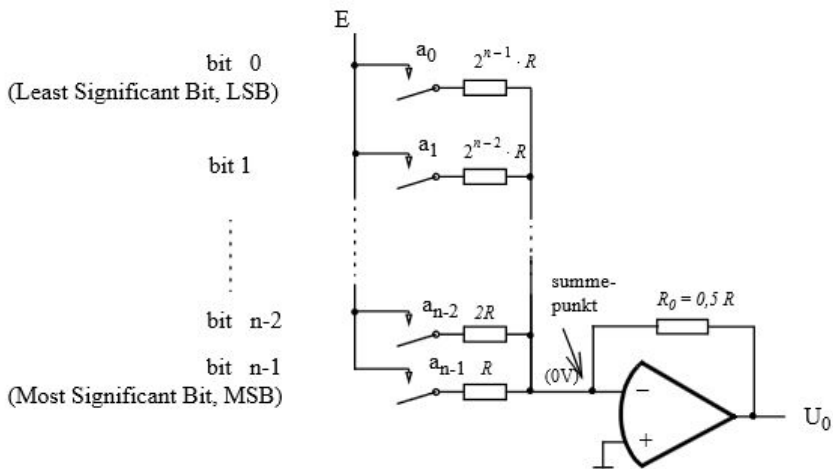
Er en motstand koblet over utgangen mellom signal og jord, parallelt til resten av kretsen. I dette tilfellet mellom 12V og jord, og mellom -12V og jord. Dette minker reflektert støy i kablene og holder her utgangsspenningen jevn.

Digital til Analog-omformer (DAC)

En Digital til Analog-omformer benytter seg av en digital inngangsverdi og en referansespenning til å skape et analogt signal. Den enkleste modellen er at hvert bit styrer en vektet motstand koblet mot referansespenningen, slik at dersom alle har verdien 1 sendes referansespenningen ut og dersom alle har verdien 0 sendes ingenting ut. Mest signifikante bit vil da ha en mye mindre motstand enn minst signifikante bit, siden den skal representere en større spenning. Ligningen blir da:

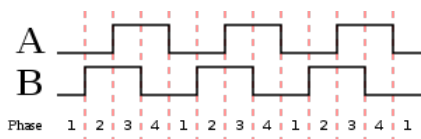
$$A = K \cdot E \cdot X \quad (2.3)$$

der E er referansespenningen, A er utgangen, K er en proporsjonal konstant og X er den digitale inngangsspenningen [17] [18].



Figur 2.7: Digital til Analog-omformer [17]

Kvadraturpuls



Figur 2.8: Diagram over kvadraturpulser

før signal B, roterer hjulet mot klokken. Dersom signal A ligger etter signal B, det vil si går fra høy til lav **etter** signal B, roterer hjulet med klokken. Enklere forklart, dersom signal B er høyt når signal A blir høyt roterer den med klokken og dersom signal B er lavt når signal A blir høyt roterer den mot klokken [14].

Kvadraturpuls er en type utgang vanlig å benytte seg av i rotasjonsomformere. Det baserer seg på to firkantpulser som er 90° forskjøvet i forhold til hverandre. Dette sikrer at man vet hvilken retning hjulet roterer. Dersom signal A ligger foran signal B, det vil si går fra høy til lav

2.1.2 Programmering

All programmering er gjort på en Arduino DUE, og følgelig med Arduinos eget språk. Det er blitt brukt en avbruddsrutine på A-signalet fra rotasjonsomformeren og en etterfølgende sjekk på B-signalet for å avgjøre fartsretning. Det er også laget en metode for å regne ut hastighet ved hjelp av antall pulser, og en metode for å regne ut spenning gitt av hastighet. Når alle beregninger er gjort skal det analoge signalet sendes ut, enten via Puls-Bredde-Modulasjon eller en DAC.

Avbruddsrutine

En avbruddsrutine, eller et avbrudd (Eng. “Interrupt”), er når en handling stopper den normale flyten i et program for å håndtere en hendelse. Hendelsen som fører til et avbrudd er ofte tidskritisk for funksjonaliteten til programmet eller for flyten til brukeren, som for eksempel bruk av mus eller tastatur på en PC [19]. Ved et avbrudd blir programflyten stoppet. Den tilstanden, eller plasseringen,

programmet er i blir lagret på stakken og avbruddsrutinen blir gjennomført. Etter dette blir tilstanden hentet fra stakken og programmet gjenopptas fra der avbruddet skjedde [5] [6]

Puls-Bredde-Modulasjon (PWM)

En digital utgang gir ut en firkantpuls med spenningsnivå typisk mellom 3 - 5V. Dersom man påvirker hvor lenge en utgang er høy i hver periode og deretter glatter utgangssignalet, vil man få et analogt utgangssignal. Dersom spenningen har 0V som minimumsnivå, vil utgangsspenningen være lik antall prosent av perioden signalet er høyt, og dette multiplisert med spenningen (se ligning 2.4). Det fremgår av figur 2.9 at amplituden til sinusbølgen er høyest der firkantpulsene er på sitt breieste [12]. Variabelen D tilsvarer antall prosent av perioden der signalet er høyt, og det er antatt at nullnivået til utgangsspenningen faktisk er null.



Figur 2.9: Puls-Bredde-Modulasjon

$$V_{out} = D \cdot V_{max} + (1-D) \cdot V_{min} \Rightarrow V_{out} = D \cdot V_{max} + (1-D) \cdot 0 \Rightarrow V_{out} = D \cdot V_{max} \quad (2.4)$$

2.2 Valg av maskinvare

Valg av deler er en essensiell del av oppgaven. Delene må være gode nok til å svare på funksjonsspesifikasjonen (se seksjon 1.2.2), rimelige nok til å kunne forsvares (se seksjon 1.2.1) og om ikke tilgjengelig i instituttets inventar, i hvertfall tilgjengelig hos en hovedleverandør. Det har også blitt tatt hensyn til leveringstid og lagerstatus, slik at ikke unødvendig lang tid har gått med til å vente på deler.

“Elfa distrelec” er en av nordens største katalogdistributører av elektronikk, de er også en av hovedleverandørene til instituttet og et firma forfatteren har god personlig erfaring med. De ble dermed benyttet som førstevalg dersom deler skulle bestilles.

2.2.1 Rotasjonsomformer



Figur 2.10: Omron E6C2-CWZ5B

Ved valg av rotasjonsomformer var det viktigste at oppløsningen var høy nok, men at det også var teoretisk mulig å nå maksimal hastigheten, uten å benytte seg av et så stort hjul at oppløsningen ble lidende. Minimumshastighet beregnes ut ifra laveste oppløsning (10 pulser/rotasjon (PPR)), og minste hjuldiameter (55mm, basert på diameteren til sensoren) [15]. Dette blir ca. 0.017m/s dersom man regner med minimum ett sekund mellom hver puls (se

ligning 2.5), noe som er godt under kravet om 0.1m/s (se seksjon 1.2.2). For å regne ut maksimal hastighet må man ha responsen og maksimum rotasjoner per minutt (RPM). Den laveste responsen til denne modellen er på 50kHz, men man må i tillegg ta hensyn til RPM. Med maksimalt 6000RPM begrenser dette mer enn responsen [15]. Med en maksimal hastighet på 17m/s (se ligning 2.6) er dette over minimumskravet i kravspesifikasjonen (se seksjon 1.2.2).

16KAPITTEL 2. BAKGRUNNSTEORI, VALG AV DELER OG UTFØRELSE

I det endelige valget ble lagerstatus tellende og modellen Omron *E6C2 – CWZ5B* med 360PPR ble valgt, selv om dette er unødvendig høy oppløsning.

Verdier benyttet for å regne ut “worst-case” til *OmronE6C2*:

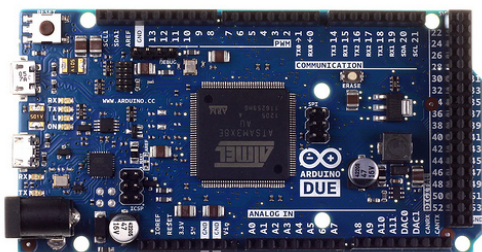
- Minste antall pulser per rotasjon = 10 PPR
- $D = 55\text{mm}$
- 6000RPM = 100 rotasjoner per sekund (R/s)
- Minste hastighet = 10 sekund per rotasjon(s/R)

$$\text{min.hastighet} = \frac{\text{Omkrets}}{s} = \frac{2\pi r}{10s/R} = \frac{(0.055 * \pi)m/R}{10s/R} = 0.017m/s \quad (2.5)$$

$$\text{maks.hastighet} = 100R/s * (0.055 * \pi)R/m = 17m/s \quad (2.6)$$

2.2.2 Mikrokontroller

Valg av mikrokontroller var litt enklere. Det finnes så mange mikrokontrollere på markedet, prisene så lave og ytelsen så høy, at de aller fleste tilfredsstiller kravene. Kravene den måtte oppfylle var først å fremst å kunne utføre koden til avbruddsrutinen på minimum $36ms$ ($360PPR * 100R/s$). Ifølge koden (se seksjon 2.3.4) trengte prosessoren å klare det å sette en variabel, sjekke en digital inngang, og å gå igjennom to “if”-setninger. Den skulle også kunne sette den analoge utgangsverdien en gang hvert sekund. Dette er en operasjon som krever fire “if”-setninger, at syv variabler settes, to multiplikasjoner og én addisjon. I tillegg må den også sette den analoge utgangen. En av de beste kandidatene var Arduino DUE, da denne hadde alt av inn- og utganger som trengtes, inkludert en fullverdig DAC, noe som ikke er særlig vanlig. En fullverdig DAC gjør at man slipper å benytte seg av PWM, og dermed ei heller må glatte utgangssignalet, slik at man sparer seg for elektroniske kretser. For å kunne ta det endelige valget måtte kravene sjekkes opp mot mikrokontrollerens hastighet.



Figur 2.11: Arduino DUE

ad” ca. $3,7\mu s$ å gjennomføre og metoden “analogWrite” ca. $7.8\mu s$ [22]. Disse kildene var selvfølgelig ikke hundre prosent troverdige, men de ga et godt nok grunnlag til å danne seg et bilde av hvilken ytelse man kunne forvente. Avbruddsrutinen tar med disse verdiene ca. $3,7\mu s + if$ og resten ca. $9,76\mu s + if$. Dessverre var det ikke å oppdri-

Kilder på internett viste til at det å sette en variabel kun tar $62,5ns$ å gjennomføre[21], multiplikasjon og addisjon tar like lang tid å gjennomføre, ca. $0.38\mu s$ per operasjon [9]. Det som tar lengst tid er å sette eller lese I/O-pinner, det tar metoden “digitalRe-

ve noen gode eksempler på hvor lang tid en “if”-setning tar, men det skal være godt under tiden til I/O-operasjoner. Dersom verdiene hadde vært i grenseland av hva den må klare hadde det vært grunn nok til å droppe den, men man kan se at den endte godt under det som krevdes. Når “Elfa distrelec” også hadde denne på lager til særdeles lav pris ble valget enkelt.

2.2.3 Op-Amp

Valg av Op-Amp hadde kun noen få kriterier. De mest åpenbare kriteriene var å takle forspenning på $-12 - +12V$ og å ha evnen til å forsterke negativt, altså sende ut negativ spenning. Ellers var også generelle kriterier gjeldende, som høy inngangsimpedans, høy stigehastighet (eng. “slew-rate”) og lav inngangsstrøm. Båndbredden er også tellende, men når utgangsspenningen er likestrøm og kun endres en gang i sekundet, er den ikke like prekær som de andre kravene. Standard “Common mode rejection ratio” (CMRR) på rundt $90dB$ ble ansett som nok for dette prosjektet.

Instituttet hadde mange ulike operasjonsforsterkere i sitt inventar, og det virket derfor unødvendig å bestille fra eksterne leverandører. Søk i datablader til de forskjellige kandidatene, og tilgjengelighet, gjorde at valget falt på en “intersil CA3140”. Den har en stigehastighet på $9V/\mu s$ [11], noe som tilsier at den kun vil slite dersom tredemøllen går fra maksimal hastighet fremover til maksimal hastighet bakover, som tilsvarer $10V$ spenningsforskjell (se seksjon 1.2.2). Den har meget høy inngangsmotstand og lav inngangsstrøm ($1.5T\Omega$ og $10pA$) [11]. Den takler maksimalt $-18 - +18V$ som forsyningsspenning og takler utgangsspenninger mellom $-14,4V$ og $+13V$ [11]. Da den i tillegg hadde en båndbredde på $4,5MHz$ og “CMRR” på $90dB$ var denne Op-Ampen som skapt for oppgaven.

2.2.4 Strømforsyning



Figur 2.12: Traco Power spenningsforsyning

Valg av strømforsyning baserte seg hovedsaklig på valgene av rotasjonsomformer, Op-Amp og mikrokontroller. Det faktum at alt skulle bygges inn i en beholder gjorde at en strømforsyning med åpne kontakter kunne velges, dette halverte prisen. Siden rotasjonsomformeren og mikrokontrolleren kun har et felles spenningsnivå, som var på 12V forsyningsspenning [16] [1], var dette et naturlig utgangspunkt.

Måleenheten skal levere ut + – 5V og Op-Ampen trenger derfor mer forsyning enn dette, noe 12V innfrir, men den trenger også en negativ spenningsforsyning for å levere negativ spenning. For å kunne drive alle komponentene må den også levere nok effekt. Arduino trekker maksimalt 800mA [1] mens rotasjonsomformeren 35mA [16]. Tar man med de andre komponentene i beregningen ble det vurdert at forsyningen burde kunne levere i området 1.5A. Søk etter produkter som oppfylte disse kravene på “Elfa distrelec” sine sider, og hensyn til lagerstatus, etterlot kun én mulig kandidat. Valget av strømforsyning falt derfor på “Traco Power TML 40212C” [20].

2.3 Utførelse

Gjennomføring av dette prosjektet har basert seg primært på identifikasjon og valg av deler og komponenter. Etter at valgene var gjort, måtte alle deler av systemet beregnes slik at mindre, men like viktige, passive komponenter kunne plukkes ut fra skolens lager. Det å regne på verdiene til motstandene brukt i spenningsdelerne og differensialforsterkeren var essensielt. Før et endelig kretskort kunne produseres måtte løsningene testes på et “breadboard”.

Etter at hovedkomponenter var valgt og kjøpt inn, parallelt med identifikasjon og konstruksjon av kretskort, skulle også opphenget konstrueres. I samarbeid med instituttets mekaniske verksted ble en idé for festemekanisme og en måte å presse ned rotasjonsomformerens hjul mot underlaget utviklet og maskinert. Det ble også gjort i stand en lukket boks med alle komponenter innvendig og godt sikrede hull til de eksterne koblingene og ledningene.

Når deler, konstruksjon og kretskort var på plass og fungerende, skulle mikrokontrolleren programmeres. I de innledende fasene ble også forarbeidet til programmeringen gjort. Dette ved at all funksjonalitet ble listet opp og det ble da klart hvilke deler som kunne, burde og måtte gjøres av mikrokontrolleren. De primære arbeidsoppgavene utført av mikrokontrolleren var å matematisk kalkulere hastighet og spenning, ut ifra rotasjonsomformerens kvadratursignal.

2.3.1 Formler

I koden benyttet det to matematiske formler kontinuerlig. Det er en for å regne ut hastigheten ut ifra antall pulser, og en for å regne ut spenningen ut ifra hastigheten.

Hastighet

Formelen for å regne ut hastigheten baserer seg på omkretsen på hjulet, dividert på antall pulser på en omdreining. Dette tallet gir antallet meter hjulet har rullet per puls.

$$\begin{aligned}
 \text{Omkrets} &= 2 \cdot \pi R = \pi \cdot \text{Diameter} = 0,1885m \\
 \text{avstand per puls} &= \frac{0,1885}{360} = 0,0005236 \text{Hastighet} = (\text{Pulses}/10000) \cdot 5,236
 \end{aligned}
 \tag{2.7}$$

Spenning

Formelen for å beregne spenningen beregner først ut tallet for hvert spenningsnivå, gitt av tallet for maksimal spenning dividert på antallet spenningsnivåer (her 24, ett nivå per m/s). Formelen tar deretter inn hastigheten og multipliserer med tallet for spenningsnivået.

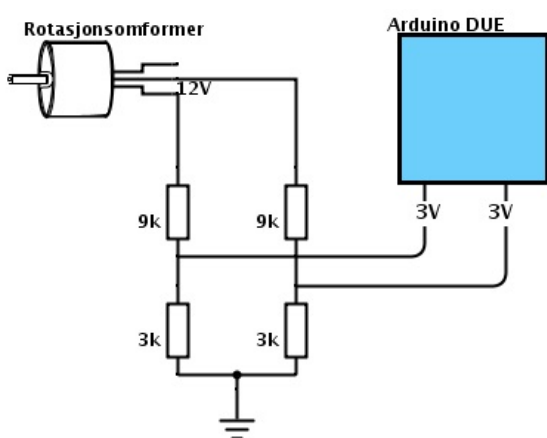
$$\begin{aligned}
 \text{spenningsniva} &= \frac{4096}{24} = 170,666666667 \\
 \text{Spenning} &= \text{Hastighet} \cdot 170,666666667
 \end{aligned}
 \tag{2.8}$$

2.3.2 Identifikasjon, design og ferdigstilling av krets-kort

Temaene tatt opp under temaet Valg av maskinvare (se seksjon 2.2) har også stor innflytelse på valgene tatt her. Valget av rotasjonsomformerer krevde at det ble satt opp to spenningsdelere, en for hvert utgangssignal. Dette på grunn av at rotasjonsomformerer sender ut 12V mens mikrokontrolleren, som kun trenger 3V, ikke tåler mer enn 3,3V [1]. Utgangsspenningen fra DAC'en på mikrokontrolleren ga, i likhet med inngangsspenningene, maksimum 0 – 3,3V. Dette gjorde at det krevdes en differensialforsterker for å oppnå en utgangsspenning på –5 – +5V.

Spenningsdeler

Spenningsdelerens oppgave er å ta inngangssignalet fra rotasjonsomformerens, spre det over to motstander i serie og plukke ut spenningen over den siste motstanden. Her må motstandene beregnes matematisk ved hjelp av reglene for spenningsdeling slik at utgangsspenningen blir liggende mellom $3V$ og $3,3V$. Utgangsspenningen er gitt av ligning (se ligning 2.1).



$$U_{ut} = U_{inn} \cdot \frac{R_2}{(R_1 + R_2)}$$

$$3,3V = 12V \cdot \frac{R_2}{(R_1 + R_2)}$$

$$\frac{3,3V}{12V} = \frac{R_2}{(R_1 + R_2)}$$

Velger R_2 til $3,3k\Omega$

$$\Rightarrow \frac{3,3}{12} = \frac{3,3k\Omega}{(R_1 + 3,3)k\Omega}$$

$$\Rightarrow R_1 k\Omega = 12 - 3,3$$

$$R_1 = 8,7k\Omega$$

Figur 2.13: Den ferdige spenningsdelingen (2.9)

Med hensyn til at $3,3V$ er det absolutte maksimum av hva mikrokontrolleren tåler, og siden også $3V$ blir detektert som digital høy på inngangen, ble R_2 valgt til litt lavere enn utregnet og R_1 litt høyere.

$$U_{ut} = 12V \cdot \frac{3k\Omega}{(9k\Omega + 3k\Omega)} = 3V \quad (2.10)$$

Differensialforsterker

Hvordan å løse problematikken med å forsterke et signal fra $0 - 3,3V$ til $-5 - +5V$ var ikke så ukomplisert som en kunne ha forventet seg. Det var ikke å finne en løsning på problemet som var vanskelig, men det å anpasse løsningen til kretskortet og de andre funksjonalitetene. Det ble derfor tidlig valgt en løsning med Op-Amp på grunn av mulighetene og brukervennligheten til en så enkel og tilgjengelig komponent.

Den første idéen som ble utviklet var å forsterke lineært opp til $0 - 10V$ for så å trekke ned spenningen med en spenningsdeler koblet til negativ spenning, kontra kobling til jord. Det ble benyttet et potensiometer, for å trimme utgangen, slik at man forsikret seg om at spenningen var $0V$ når tredemøllen står i ro. Litteratursøk i bøker og på internett, og fysisk testing (se seksjon 2.3.2), ledet til å avkrefte dette som en gjennomførbar metode. Det som gjorde denne løsningen vanskelig var at inngangsspenningen varierte, slik at strømmen varierte. Man ser på ligningene at dette ikke gir det ønskede utnivået på $-5 - +5V$ (se ligning 2.12).

Spenningsdeling med negativ spenning krever at man differensierer mellom spenning over motstand og utgangsspenning. Dersom en vil ha $5V$ som utgangsspenning, må man begynne med grunnspenningen og deretter adderer med spenningen over utgangsmotstanden (se ligning 2.11).

$$U_{ut} = U_{-inn} + U_{R_2}$$

For eksempel

$$U_{ut=} = -12V + 17V = 5V \tag{2.11}$$

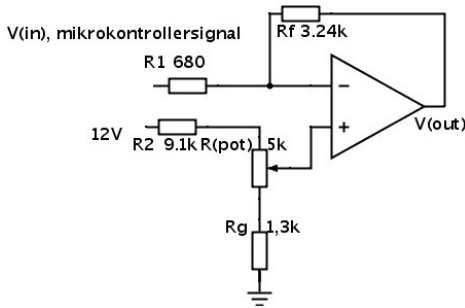
$$\begin{aligned}
U_{R_1} &= U_{ut} + 12V = U_{inn} \cdot \frac{R_2}{(R_1 + R_2)} \\
17V &= (10V - (-12V)) \cdot \frac{R_2}{(R_1 + R_2)} \\
\frac{17V}{22V} &= \frac{R_2}{(R_1 + R_2)} \\
\text{Velger } R_2 &\text{ til } 17k\Omega \\
\Rightarrow R_1 k\Omega &= 22 - 17 \\
R_1 &= 5k\Omega
\end{aligned} \tag{2.12}$$

Som leder til denne ligningen ved 0V inn

$$\begin{aligned}
U_{R_1} &= 12 \cdot \frac{17}{(5 + 17)} \\
U_{R_1} &= \frac{17 \cdot 12}{(5 + 17)} \\
U_{R_1} &= 9,3V \\
\Rightarrow U_{ut} &= -12V + 9,3V \\
U_{ut} &= -2,7V \neq -5V
\end{aligned}$$

Videre ble det spekulert i om spenningsdeleren kunne benyttes til å gi ut midtpunktet mellom minimal og maksimal utgangsspenning. Denne spenningen kunne så benyttes som nullpunktsreferanse til Op-Ampen, men idéen ble avkreftet etter å ha lest databladet. Dette på grunn av at "offset null"-inngangen ikke kan benyttes for å gi Op-Ampen en ny nullpunktsreferanse [11].

Den neste idéen var å benytte den inverterende inngangen til forsterkning, og den ikke-inverterende inngangen som referanse. Dette virket teoretisk som en god løsning, men resultatene fra testen tydet på at denne metoden ikke virket som først antatt. Et besøk innom wikipediaartikkelen som omhandlet bruksområder til OpAmper, etablerte at en differensialforsterker måtte anvendes for å få ønsket effekt(http://en.wikipedia.org/wiki/Operational_amplifier_application



Figur 2.14: Den ferdige differensialforsterkeren

$$U_{ut} = U_2 \cdot \left(\frac{(R_f + R_1)R_g}{(R_g + R_2)R_1} \right) - U_1 \left(\frac{R_f}{R_1} \right)$$

de kjente verdiene er

$$U_{ut} = -5 \text{ når } U_1 = 2,75$$

$$U_{ut} = 5 \text{ når } U_1 = 0,65$$

$$U_2 = 12$$

(2.13)

For å bestemme verdiene på de fire motstandene ble den effektive, men lite vitenskapelige, metoden “prøv og feil” benyttet. Siden den inverterende utgangen benyttes for forsterkingen, må nødvendigvis høyere spenning ut fra mikrokontrolleren gi lavere spenning totalt ut. Det var også lett å konkludere med at R_2 måtte være større enn R_g for å motvirke effekten av at $U_2 > U_1$. Det var klart at motstandene burde ha ligget i $k\Omega$ -regionen, for å holde stømmen på et forsvarlig nivå. Ut ifra ligningene var det tydelig at det kunne være en god idé å velge R_1 lav. R_1 og R_f ble først valgt ut ifra tilgjengelige restekomponenter. Disse komponentene ble valgt til $R_1 = 680\Omega$ og $R_f = 3,24k\Omega$. Dermed har man kun to ukjente igjen.

$$U_{ut} = U_2 \cdot \left(\frac{(R_f + R_1)R_g}{(R_g + R_2)R_1} \right) - U_1 \left(\frac{R_f}{R_1} \right)$$

$$5 = 12 \cdot \left(\frac{(R_f + R_1)R_g}{(R_g + R_2)R_1} \right) - 0,65 \left(\frac{R_f}{R_1} \right) \quad (2.14)$$

velger verdien til $R_g = 1,8k\Omega$

(Verdien er basert på forskjellen i størrelse mellom teller og nevner, og på tilgjengelige komponenter)

Dette ledet til denne utregningen av R_2 :

$$R_2 = \frac{((R_f + R_1) \cdot R_g U_2) - (U_{ut} R_g R_1 + U_1 R_f R_g)}{U_{ut} R_1 + U_1 R_f}$$

$$R_2 = \frac{((3,24 + 0,68) \cdot 1,8 \cdot 12) - (5 \cdot 1,8 \cdot 0,68 + 0,65 \cdot 3,24 \cdot 1,8)}{5 \cdot 0,68 + 0,65 \cdot 3,24}$$

$$\Rightarrow R_2 = 13,58 \tag{2.15}$$

Tester for det andre ekstremalpunktet til inngangsspenningen:

$$R_2 = \frac{((R_f + R_1) \cdot R_g U_2) - (U_{ut} R_g R_1 + U_1 R_f R_g)}{U_{ut} R_1 + U_1 R_f}$$

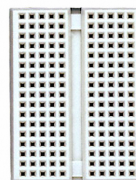
$$R_2 = \frac{((3,24 + 0,68) \cdot 1,8 \cdot 12) - (5 \cdot 1,8 \cdot 0,68 + 2,75 \cdot 3,24 \cdot 1,8)}{5 \cdot 0,68 + 2,75 \cdot 3,24}$$

$$\Rightarrow R_2 = 13,57 \tag{2.16}$$

Det ble en liten avrundingsfeil her, men den ble løst av potensiometeret. Det kan også noteres at verdiene for maksimal og minimal inngangsspenning, som ble brukt i ligningene, ikke er $0V$ og $3,3V$ som nevnt tidligere i oppgaven og i databladet [1]. Dette fordi det ikke var den faktiske utgangsspenningen (se seksjon 2.3.2).

Testing

Før alle komponenter ble loddet fast på et “veroboard” for å lage et kretskort, var det viktig å teste at de oppførte seg som forventet. Alle passive komponenter ble målt for å kartlegge avvik fra leverandørens oppgitte verdi. Komponentene ble satt sammen til sine kretser



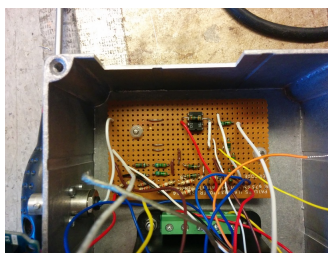
Figur 2.16: “Breadboard”

på et “breadboard” (se figur 2.16). Hver krets ble først testet separat og deretter ble hele systemet satt sammen og testet, dette for å underlette feilsøking. Alt ble til slutt testet opp mot rotasjonsomformerens og mikrokontrolleren.

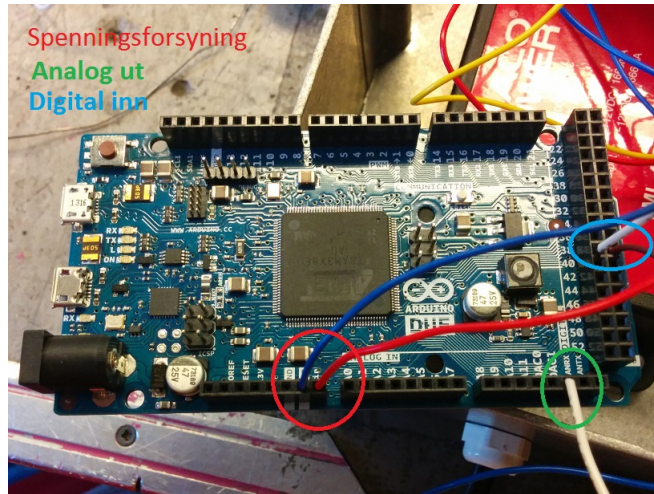
Rotasjonsomformerens testing inkluderte også testing av mikrokontrolleren. For å kunne teste funksjonaliteten til rotasjonsomformerens, måtte det etableres om mikrokontrolleren klarte å lese pulsene. Fordelen med produkter fra Arduino er at de er “open source” og har et stort nettverk, det vil si at det som oftest er noen som allerede har gjort det man prøver å utrette og lagt ut en forklaring på nett. I tilfellet hvordan å ta inn signal fra en rotasjonsomformer var det ikke bare én løsning, men mange. Etter sammenligninger og grundige undersøkelser ble det brukt en guide utviklet av Kevin Darrah [7]. Spesielt nyttig var kodeeksempelet, og denne oppgaven benyttet seg fritt av biter fra den koden [8].

Mikrokontrolleren ble koblet til PC med micro-USB, og rotasjonsomformerens ble koblet til en 12V spenningskilde. Det ble valgt to digitale inngangsporter på mikrokontrolleren, og koden ble endret for å tilpasses de inngangene som ble benyttet (se seksjon 2.3.4). Koblingene inn til mikrokontrolleren kan observeres i figur 2.17.

Rotasjonsomformerens brukt i dette prosjektet gir ut et såkalt “PNP åpen kollektor”-signal, dette vil si at det trenger en kobling



Figur 2.15: Det ferdige kretskortet



Figur 2.17: Koblinger på Arduino DUE-mikrokontrolleren

med en motstand til jord parallelt med koblingen inn (se seksjon 2.1.1). De digitale inngangene til Arduino har muligheten til å fungere som “pull up”, men ikke “pull down”. Arduino anbefaler da å benytte en $10k\Omega$ motstand koblet til jord [2].

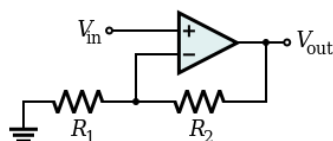
De initiale testene ga ingen resultater, men måling med multimeter viste at det kom et utgangssignal fra rotasjonsomformeren. Målingene viste at signalet fra rotasjonsomformeren lå på 12V, noe som kunne ødelegge mikrokontrolleren, og dette introduserte kravet om spenningsdelere.

Etter installasjon og testing av spenningsdelere, fungerte det utmerket. Signalet fra rotasjonsomformeren ble registrert og behandlet som forventet.

Spenningsdelerne ble konstruert og verdier på motstandene valgt som beskrevet i seksjon 2.3.2. Testing av spenningsdelerne alene ga ønsket resultat og testen ble dermed utvidet med rotasjonsomformeren og mikrokontrolleren. Også denne testen ga positive resultater og alt tydet på at rotasjonsomformeren, spenningsdeleren og koden for

å behandle signalet, fungerte som ønsket.

Differensialforsterkeren ble først satt opp etter at det ble testet en ikke-inverterende forsterker som ble koblet til en spenningsdeler mellom utgangen og $-12V$. Dette ga ikke de ønskede resultatene og ved å gå tilbake til de matematiske formlene ble det åpenbart hvorfor (se ligning 2.12). Det ble da klart at det måtte benyttes en differensialforsterker (se seksjon 2.3.2).



Figur 2.18: Ikke-inverterende forsterker

Differensialforsterkeren ble koblet opp som vist i figur 2.14, og til den inverterende inngangen ble utgangen fra mikrokontrollerens DAC koblet til. Mikrokontrolleren var kodet til å sende ut et signal som sakte gikk opp fra null til maksimum for så å starte på nytt igjen, som et testsignal. Kun et lite problem ble møtt på i denne testen, og det var grunnet løse kontakter i koblingen til potensiometeret. Disse kontaktpunktene ble derfor loddet fast og testen ble gjennomført en gang til. Dette ordnet problemene og resultatet ble som ønsket. Det kan være verdt å notere seg at for å stille inn potensiometeret riktig må det måles med et multimeter eller voltmeter.

Etter at testoppsettet ga ønskede resultater, ble alle deler satt sammen. Rotasjonsomformerene, spenningsdelerne, mikrokontrolleren og differensialforsterkeren ble koblet sammen, og alt ble testet som én enhet. Dette ga ønskede resultater allerede fra første test, prosessen med å få laget et kretskort ble dermed satt igang.

2.3.3 Design og konstruksjon av mekaniske deler

De mekaniske delene ble konstruert av instituttets mekaniske verksted, etter design utviklet i samarbeid med veileder og personell derfra. Det ble laget to forskjellige mekaniske konstruksjoner, et oppheng

for rotasjonsomformer og en beholder for de elektroniske komponentene (strømforsyning, kretskort og mikrokontroller).

Oppheng for rotasjonsomformer

Kravene som ble satt til opphenget var at:

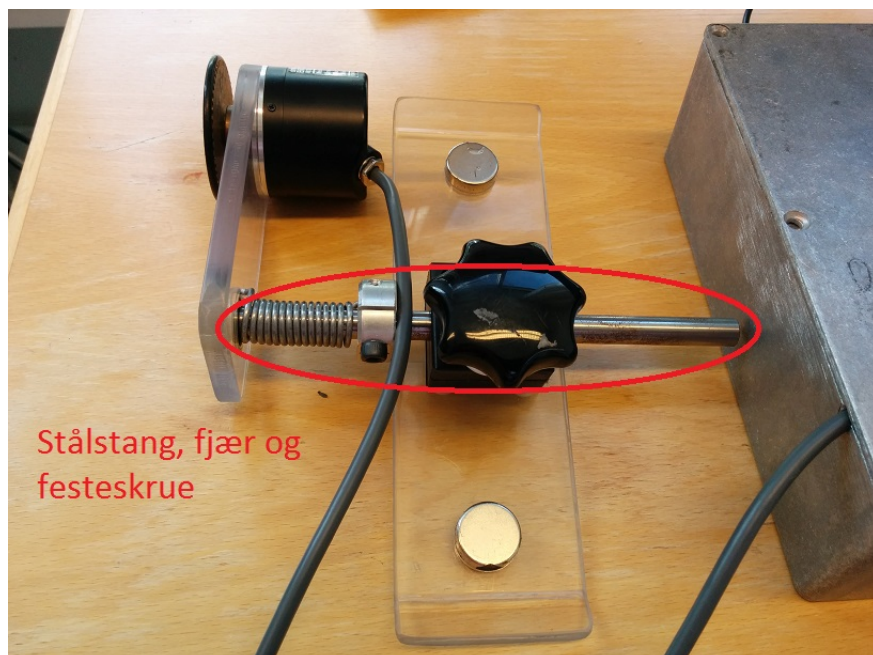
- det skulle ha en enkel festemekanisme som kunne benyttes på flere tredemøller uten nevneverdig arbeid
- den måtte presses ned mot det bevegelige underlaget, slik at hjulet til enhver tid rullet og ikke sklei
- det måtte ta minst mulig plass, slik at det ikke var i veien for forsøksperson
- lengden mellom rotasjonsomformer og festet burde kunne endres

Punkt én var viktig for at måleren skulle kunne bli tatt vekk mellom forsøk og for å kunne benyttes på ulike tredemøller, og det var derfor viktig at det ikke ble en fast installasjon. Det andre punktet var avgjørende for å få korrekte målinger. Punkt tre var der for å påse at opphenget var praktisk mulig å bruke til forsøk. Det siste punktet gjør det mulig å benytte måleutstyret på forskjellige tredemøller som har ulik avstand mellom rullebånd og festeplate.

Resultatet som oppfylte kravene, ble en konstruksjon laget av hardplast. Opphenget besto av tre deler, et feste med to sterke magneter, et oppheng for rotasjonsomformer og en stålstang for å feste dem til hverandre (se figur 2.19). Stålstangen kunne også variere avstanden og den fungerte også som feste for den ene enden av fjæren. Skruen som festet stålstangen ble benyttet dersom avstanden skulle reguleres og dersom fjæren måtte strammes (se figur 2.19).



(a) Oppheng med forklaringer
nr.1

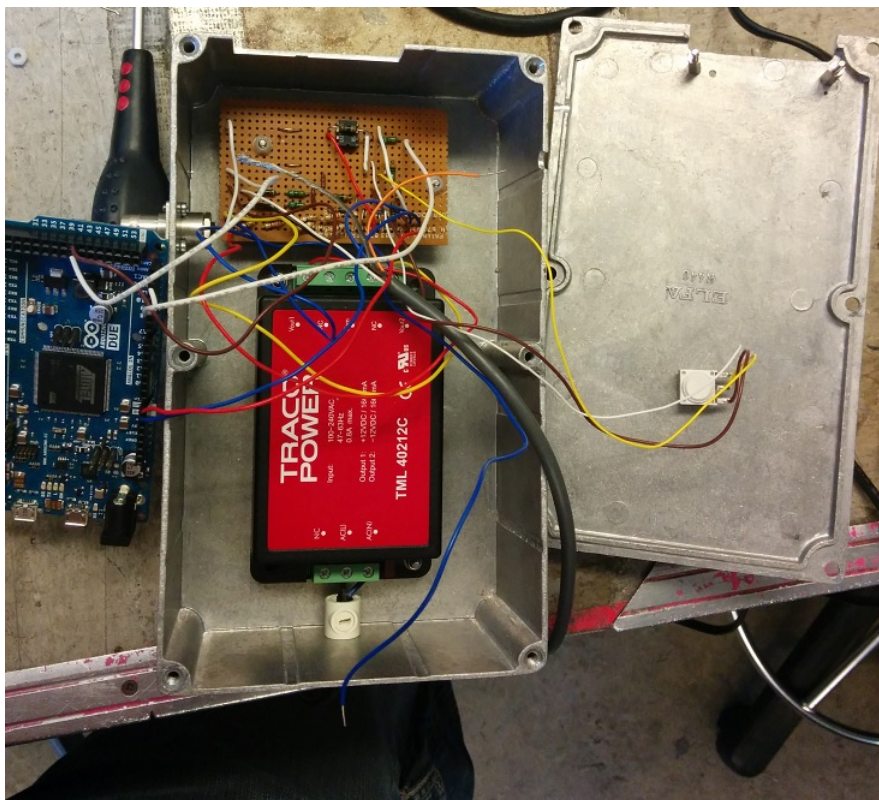


(b) Oppheng med forklaringer
nr.2

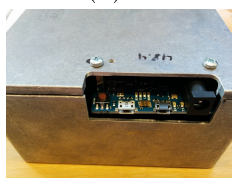
Figur 2.19: Oppheng

Beholder for elektronikk

I tillegg til opphenget ble det også konstruert en beholder for all elektronikken. Denne skal beskytte brukere fra de åpne koblingene i spenningsforsyningen og samtidig beskytte kretskortet og koblingene fra ytre påvirkninger. Dette var viktig da mange av koblingene sitter utsatt til på kretskortet og mikrokontrolleren. Spenningsforsyningen får også nettspenning inn i to åpne kontakter, dette hadde vært livsfarlig uten tildekking. Løsningen ble en stålboks der kretskort og spenningsforsyningen ble festet i bunnen. Mikrokontrolleren ble festet i lokket og det ble laget en åpning, slik at den var mer tilgjengelig. Åpningen ble laget for å lettere komme til nullstillingsknappen som benyttes hver gang man starter et forsøk, og for å kunne programmere om dersom det trengs. Åpningen gjør det også mulig å koble den til en PC, slik at man kan overvåke hastigheten direkte fra mikrokontrolleren. I tillegg er det to kabler ut, en for strøm og en til rotasjonsomformerens. På ene siden er det festet en BNC-kontakt, for tilkobling av utstyr fra Institutt for bevegelsesvitenskap [13]. Det er også et potensiometer festet i lokket, slik at man kan trimme utgangen (se figur 2.20).



(a) Innsiden av beholderen, mikrokontrolleren ikke festet



(b) Mikrokontroller



(c) BNC-kontakt



(d) Potensiometer

Figur 2.20: Beholderen

2.3.4 Programvare og kode

Programmeringen har blitt gjort i Arduino sitt eget programmeringsspråk som er basert på språket “Wiring” (<http://wiring.org.co/>) [3]. En av de store fordelene med å benytte Arduino er at det er “open source”. Dette fører flere gode ting med seg. For det første er det fritt frem å bruke og å installere slik som man vil. For det andre vil det finnes mange guider og biblioteker med funksjoner og kode laget av tredjeparter. I praksis vil dette si at man sjeldent trenger å programmere så mye, men heller lete etter en ferdig kode som utfører det man trenger å gjøre og deretter tilpasse den.

I denne oppgaven ble koden til Kevin Darrah benyttet [8]. Etter noen initiale tester ble koden tilpasset til å kun ha en avbruddsrutine, i motsetning de fire originale. Dette ble gjort da testene viste at mikrokontrolleren ikke klarte å holde følge med koden dersom hastigheten overskred $2m/s$.

Forklaring av koden

Koden begynner med å sette opp de globale variablene som programmet trenger. Dette består av syv variable eller konstante verdier. Variabelen “Pulses” som teller opp antall pulser. Konstanten “Interval” som er en referanse og variabelen “previousMillis” som lagrer antallet millisekunder siden programstart, benyttes for å regne ut når det har gått et sekund. Variablene “A_SIG” og “B_SIG” som holder styr på verdiene fra rotasjonsomformerene. Sist variablene “speed” og “voltage” som holder styr på farten målt av rotasjonsomformerene og spenningen som skal sendes ut.

Setup setter opp alle initiale verdier og moduser. De digitale “pinene” 40 og 41 blir satt opp som innganger, pin 40 får i tillegg en avbruddsrutine (kalt “A_RISE”) koblet til seg på stigende flanke. Metoden “analogWriteResolution” setter oppløsningen på DAC’en, her settes den til 12 bit. Det blir også satt opp mulighet for seriell kommunikasjon med en “baud rate” (bit per sekund) på 115200.

Main Loop kjører når programmet kjører, denne løkken blir kjørt om og om igjen. I dette programmet sjekker den kontinuerlig om det har gått et sekund. Dette gjøres ved å sette variabelen “currentMillis” lik verdien hentet av metoden “Millis”, som er antall millisekunder siden programstart. Det gjøres en sammenligning av “currentMillis” med “previousMillis”, dersom det har gått et sekund kjøres innholdet i “if-setningen”.

Farten blir kalkulert ut ifra “pulses”-variabelen og spenningen som skal sendes ut blir kalkulert ut ifra “speed”-variabelen. Til slutt blir spenningen satt med “analogWrite”, “pulses” blir nullstilt og “previousMillis” blir satt lik “currentMillis”.

“**A_RISE**” er avbruddsrutinen, den venter på en høy flanke inn på pinne 40. Når det blir oppdaget en stigende flanke avbrytes programflyten og “A_RISE”-koden blir gjennomført (for detaljer, se seksjon 2.1.2). Inngangen på pinne 41 blir sjekket og lagret i variabelen “B_SIG”. Dersom “B_SIG” er lav beveger seg hjulet fremover, og vice versa dersom “B_SIG” er høy (se seksjon 2.1.1).

Speed er en metode som tar inn antall pulser per sekund, ganger opp tallet med antall meter mellom hver spalteåpning ($\frac{\text{omkrets}}{360}$) og returnerer resultatet som hastighet i meter per sekund (se seksjon 2.3.1).

Voltage er en metode som tar inn farten og regner ut spenningen som skal settes på den analoge utgangen. DAC’en har en oppløsning på 12 bit, det vil si 4096. Siden hastigheten, og dermed også hastighetsvariabelen, kan være negativ, mens spenningsnivået ikke kan være negativt, må variabelen endres. Man adderer med 12, som er den maksimale negative (og positive) verdien (se seksjon 1.2.2) slik at hastigheten varierer mellom $0 - 24m/s$ i stedet for $-12 - +12m/s$. Dette forsikrer at dersom båndet står i ro blir det sendt 2048 til DAC’en, som blir forsterket til $0V$. Det er også satt opp noen “if-

setninger” for å sikre at verdier utenfor grenseområdet ikke ødelegger systemet 2.3.1.

Kode Den fullstendige koden:

```

int pulses , A_SIG=0, B_SIG=1;
unsigned long previousMillis=0;
int interval=1000;
double speed=0;
double voltage=0;

void setup() {
  //setter opp avbrudd og input paa
  //portene fra Rotasjonsomformereren
  pinMode(40, INPUT);
  pinMode(41, INPUT);
  attachInterrupt(40, A_RISE, RISING);
  Serial.begin(115200);

  analogWriteResolution(12);
}

void loop() {
  //sett spenning og regn ut farten hvert sekund
  //sjekker om det har gaatt et sekund,
  //dersom ja, regn ut fart og nullstill pulsvariabel
  unsigned long currentMillis = millis();
  if ((unsigned long)(currentMillis - previousMillis) >= interval) {
    speed=SPEED(pulses);
    voltage = Voltage(speed);
    analogWrite(DAC1, voltage );
    pulses=0;
    previousMillis=currentMillis;
  }
}

```

```
//avbruddsrutinen , ved avbrudd sjekkes B-signalet
//for aa avgjøre fartsretning og pulsvariabelen oeker/minker
void A_RISE(){
  A_SIG=1;
  B_SIG=digitalRead(41);
  if(B_SIG==0)
    pulses++;//moving forward
  if(B_SIG==1)
    pulses--;//moving reverse
}

//Metode som tar inn antall pulser og returnerer fart
double SPEED(double Pulses){
  double V=0;
  V=(Pulses/10000)*5.236;
  return V;
}

//Metode som tar inn farten og
//regner ut hvilken spenning som skal settes ut
double Voltage(double speed){
  double S=0;
  double V=0;
  S = speed + 12;
  V = S * 170,6666666667;

  if(S == 12)
    V = 2048;
  if(S <= 0)
    V = 0;
  if(S >= 24)
    V = 4096;

  return V;}

```

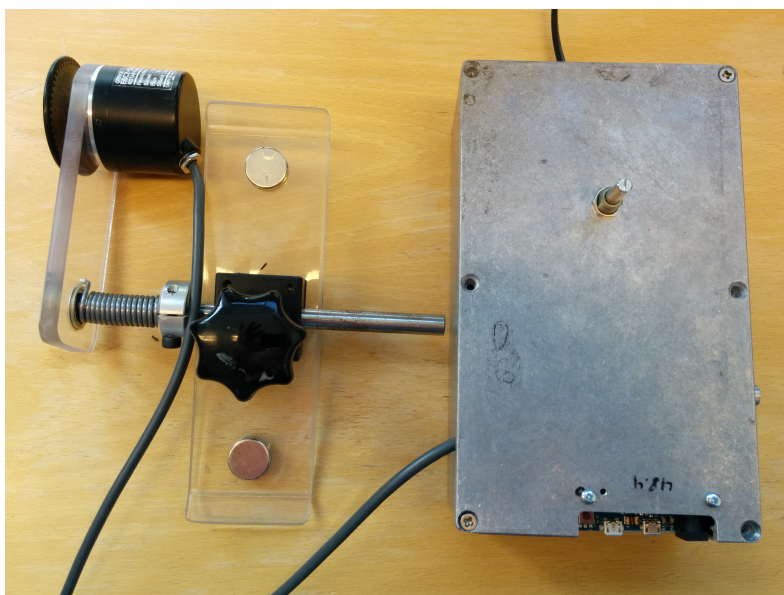
Kapittel 3

Resultat, diskusjon og videre arbeid

3.1 Resultater

3.1.1 Hastighetsmåleren, arkitektur og brukergrensesnitt

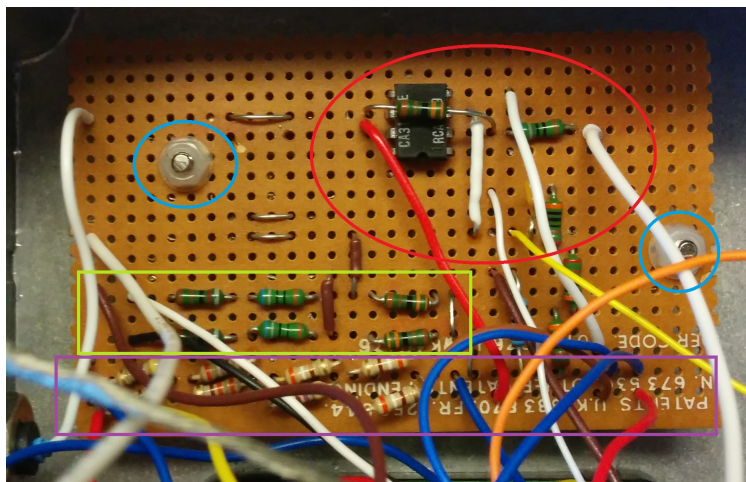
Det endelige resultatet ble, som tidligere forklart, en hastighetsmåler bestående av en rotasjonsomformer, festet på en fjærbelastet arm og som videre var festet til en stang med variabel avstand til et magnetisk feste. En metalbeholder som inneholder all elektronikk, med kabler til rotasjonsomformeren, strøm og uttak Institutt for bevegelsesvitenskaps utstyr. Metalbeholderen inneholder spenningsforsyningen, kretskortet (med to spenningsdelere og en differensialforsterker) og mikrokontrolleren. Mikrokontrolleren tar inn ett spenningsdelt digitalt kvadratursignal fra rotasjonsomformeren og sender ut en analog spenning. Det analoge signalet blir forsterket på kretskortet og sendt ut på en BNC-kontakt.



Figur 3.1: Den endelige måleenheten

Kretskort

Kretskortet ble laget på et “veroboard”, der alle hull er koblet vertikalt sammen. Spenningsforsyningen ble plassert på tre linjer, med 12V, 0V og $-12V$ på linjene. Som termineringsmotstand ble det plassert fire motstander i parallell, med samlet verdi på 100Ω , over hver av utgangene. Antallet var for å spre strømmen over flere komponenter for å slippe risiko for å ødelegge dem. Ved siden av ble de to spenningsdelerne koblet opp, med inngang fra rotasjonsomformerens og utgang til mikrokontrolleren. Til sist ble Op-Amp og differensialforsterkeren satt opp, med inngang fra mikrokontrolleren og utgang til BNC-kontakten.



Differensialforsterker med Op-Amp

Spenningsforsyning med termineringsmotstand (4 i parallell for å minke strømmen gjennom hver enkelt)

Spenningsdelere

Feste- og avstandsskruer

Figur 3.2: Kretskort med forklaringer

Der alt følger kretsskjemaet vist i figur 2.1 på side 7.

Beholder for elektronikk og brukergrensesnitt

Beholderen ble laget for å beskytte brukere fra de åpne koblingene direkte til nettspenning og for å beskytte kretskortet, mikrokontrolleren, spenningsforsyningen og mer spesifikt, kablingen mellom dem. Den virker også som brukergrensesnitt med potensiometer og BNC-kontakt festet til utsiden og et hull for å underlette tilgangen til mikrokontrolleren. På mikrokontrolleren er nullstillingsknappen, mikro-USB-kontakter og strømpluggen tilgjengelig. Det er også kabel til strøm og kabel til rotasjonsomformerer ut fra beholderen (se figur 2.20 på side 32).

Rotasjonsomformer og festemekanisme

Det ble laget et oppheng for rotasjonsomformerer. Målet til denne var å feste rotasjonsomformerer til tredemøllen, stille inn avstanden til rullebåndet og å presse rotasjonsomformerer og hjulet ned mot båndet. Brukeren må feste magnetene på dekslet ved siden av båndet på tredemøllen. Å stramme fjæren og endre avstanden styres av den svarte skruen. For å stramme fjæren må man vri armen med klokken, forutsatt at måleren festes på høyre side av tredemøllen. Avstanden reguleres også med den svarte skruen og stålarmen. Alle deler er vist i figur 2.19 på side 30.

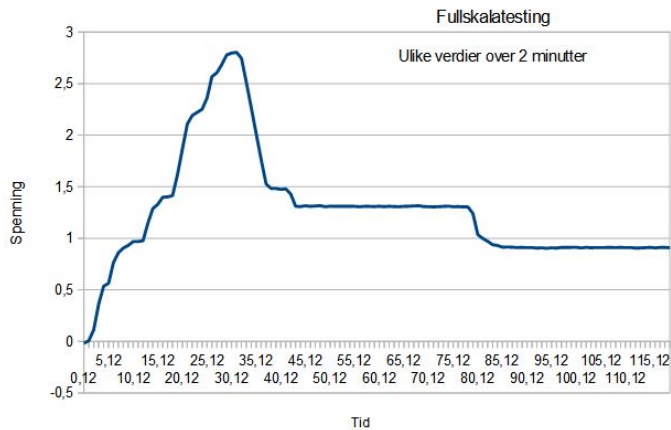
Fullskalatesting

Ved fullskalatesting ble det oppdaget en del små feil og mangler, nevnt i seksjon 3.3.2. Det ble utarbeidet en ikke-permanent løsning, der opphenget ble stablet på to treplanker og en person måtte stå på disse for å stabilisere dem under forsøk. Dette var naturlig nok ikke holdbart, men det tillot testing. Testen startet med å måle nullnivået, for å se om det var noen statiske avvik. Det ble oppdaget noen små variasjoner på inngangen, disse var på omtrentlig $0,01V$ "peak-to-peak" (se figur 3.4), noe som tilsvarer $0,024m/s$ (se likning 3.1). Dette er ikke i nærheten av kravspesifikasjonens oppløsning på $0,1m/s$ (se

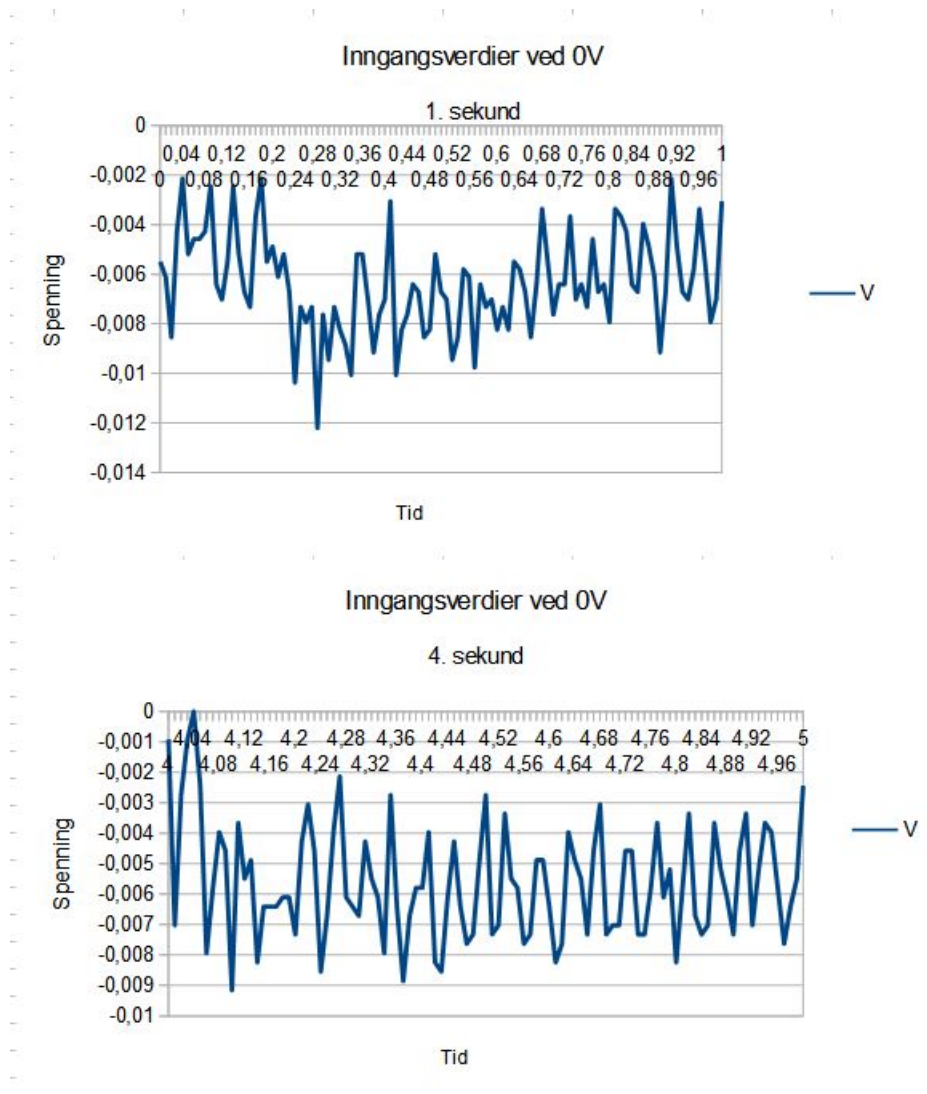
seksjon 1.2.2).

$$\begin{aligned}
 12m/s &= 5V & 2,8V \cdot 2,4 \frac{m/s}{V} &= 6,72m/s \neq 5,85m/s \\
 1V &= 2,4m/s & 1,3V \cdot 2,4 \frac{m/s}{V} &= 3,12m/s \neq 3m/s \\
 \frac{1V}{100} &= \frac{2,4}{100} & 0,9V \cdot 2,4 \frac{m/s}{V} &= 2,16m/s \neq 2m/s \\
 \Rightarrow 0,01V &= 0,024m/s & & \\
 & (3.1) & & (3.2)
 \end{aligned}$$

Når man tar og tester over to minutter, med forskjellige hastigheter, slik at man må zoome ut for å se hele bildet, ser det mye bedre ut. Nivåene som ble testet var stigning opp til maksimal fart ($5,85m/s$). Deretter ble hastigheten senket ned til ca. $3m/s$ en stund, før hastigheten ble senket enda en gang til ca. $2m/s$. Dersom vi ser på verdiene i grafen, og i vedlegg E, ser vi at de ligger rundt $2,8V$, $1,3V$ og $0,9V$. Hastighetene er ikke helt riktige (se likning 3.2), dette kan skyldes målefeil i tredemøllen, det at likningen var feil i programmet (se seksjon 3.3.2) og delvis at avlest hastighet ikke er nøyaktig. Det siste kan skyldes menneskelig svikt, statisk avvik og støy.



Figur 3.3: Resultatene fra fullskalatesten over 2 minutter



Figur 3.4: Utsnitt av resultatene ved 0V

3.2 Diskusjon

I en oppgave av dette slaget, vil det alltid være en stor porsjon “etterpåklokskap” til stede ved innlevering. Det er mye som kunne og burde vært gjort annerledes i denne oppgaven.

Rotasjonsomformerer benyttet i denne oppgaven hadde utganger av typen “PNP” der man får et høyt signal fra den integrerte kretsen, og denne må trekkes lav når det ikke kommer signal. Dette førte, i denne oppgaven, til kravet om spenningsdelere (se seksjon 2.3.2). Det er vanlig at integrerte kretser har enten “PNP”- eller “NPN”-utganger, og her hadde det vært bedre å benytte seg av “NPN”. Med “NPN”-utgang trekker man linjen høy med en motstand koblet til en høyere spenning, som kalles en “pull up”. Dette kan gjøres internt på mikrokontrolleren [2] og hadde da ikke krevd noen ekstra kretser. Den eneste konsekvensen hadde vært at man måtte ha koblet avbruddsrutinen mot synkende flanke, for å ikke telle de høye pulsene som kommer mellom signalets lave pulser.

K.I.S.S. eller “Keep it simple, stupid” er en læresetning som de aller fleste ingeniørstudenter glemmer så fort de skal lage noe. Det er viktig å ikke prøve å gjøre mer enn man må, og å unngå å gjøre det på en vanskeligere måte enn det som er absolutt nødvendig. Valg av deler går veldig mye raskere og lettere, dersom man vet hva komponentene må utrette og hva de formelle kravene er. Dersom man har mange idéer man vil utrette, blir det mange flere avveininger som må gjøres og man kan fort bli sittende handlingslammet.

Programmeringen som førte til at antallet avbruddsrutiner gikk fra fire til én, som nevnt i seksjon 2.3.4, var mest trolig unødvendig. Etter å ha lest igjennom den brukte koden [8], er det trolig at det er “serial.println”, som kjøres i hvert avbrudd, som tar mest tid og som fører til at den ikke klarer å henge med. Selv om $(360 * 4) * 100 =$

144000 er mange pulser, hadde det kunnet holde med en halvering av antall pulser.

Tidsplanen er der for å hjelpe, og det er derfor alltid en god idé å følge tidsplanen. Det kan hende det virker som god tid i begynnelsen, men det kommer alltid uforutsette utsettelse mot slutten. I dette prosjektet førte manglende tidshold til at fullskalatesting ble særdeles forsinket og dermed fikk man nesten ikke testet måleutstyret før innlevering.

3.3 Fremtidig arbeid

“A designer knows he has achieved perfection not when there is nothing left to add, but when there is nothing left to take away”

— Antoine de Saint-Exupery

Det er alltid vanskelig å si at man er ferdig med noe og det finnes alltid noe mer man har lyst til å tilføye. Kanskje spesielt som ingeniør, da det alltid er en teknisk finesse eller enda en ekstra funksjon man vil legge til. Det er i de tilfeller Antoines sitat er godt å huske. Når det er sagt, så finnes det fortsatt mye som kan forbedres ved dette prosjektet.

3.3.1 Mangler i henhold til kravspesifikasjon

Ut ifra kravspesifikasjonen (se seksjon 1.2.2) kan vi se at det er noen “bør”-punkter som ikke blir oppfylt. Det ferdige produktet har hittil intet aksesspunkt for logging av distanse. Det er ikke manko på aksesspunkt siden begge mikro-USB-portene er tilgjengelige, men de benyttes ikke til noe for øyeblikket. Det enkleste ville vært å benytte seg av seriell kommunikasjon over USB, og deretter ha et kjørende program på mottakersiden som lagret informasjonen i en fil. Informasjonen kan være:

- hastigheten, slik at mottakerprogrammet regner ut og lagrer distansen
- distansen, regnet ut på sendersiden og så sendt ut på BNC-kontakten;
 - som endelig distanse, når programmet avsluttes
 - periodisk, som distansen hittil, for å forsikre seg om at man har et resultat

Det viktigste hensynet å ta når man velger hvordan å sette opp dette, er tidsbruk av mikrokontrolleren. Det å sende ut hastigheten fører til færrest kalkulasjoner på mikrokontrolleren, men å sende informasjon ut er en krevende operasjon som vil ta opp mye av tiden til prosessoren. Det å kun sende ut den endelige distansen er lite krevende for mikroprosessoren, men det kan fort ende i at distansen ikke blir sendt, dersom ikke nøyaktig riktig avslutningsprosedyre blir fulgt. En mellomting vil være å sende ut distansen med periodiske mellomrom, noe som sikrer at det finnes en distanse, men uten at det må skje så ofte at det går ut over de andre delene av programmet. Dette kan, uheldigvis, føre til store avvik mellom virkelig distanse og oppgitt distanse, dersom det har gått lang tid mellom siste mottatte distanse og avsluttning av forsøk.

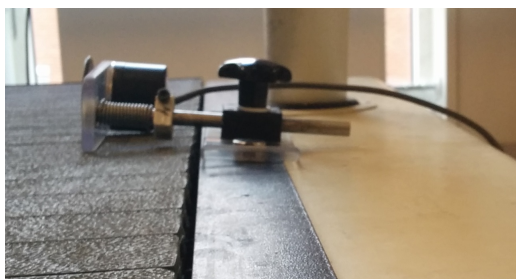
3.3.2 Mangler eller forbedringspotensiale

Ved fullskalatesting ble det oppdaget noen forbedringspotensialer ved konstruksjon, programmering og kretsdesignet. Spesielt var det tydelig når testoppsettet besto av to planker under opphenget, som måtte ha en person oppå seg for å ha tilstrekkelig stabilitet.

Mangler ved konstruksjon

Rotasjonsomformeren burde ha hatt et større hjul. Det hjulet som ble festet var kun akkurat større enn omkretsen til rotasjonsomformeren, og dette førte til at oppheng og rotasjonsomformeren lett kom borti båndet og risikerte skader på utstyr og ødelagte forsøk (se figur 3.5).

Opphenget til rotasjonsomformeren burde vært hevet høyere over festeplaten til tredemøllen. Dette ville også sikret minst mulig risiko for uønsket kontakt mellom konstruksjon og det bevegelige båndet (se figur 3.5).



Figur 3.5: Feilene som oppsto ved fullskala testing

Ventilasjon inn i stålbeholderen burde vært laget, for å forhindre overoppheting av komponenter og spenningsforsyningen.

Mangler ved elektronikk

Potensiometeret burde ha vært et som stilles inn kun én gang, siden det ikke er noen verdier som vandrer. Ved å benytte et potensiometer med skrue på utsiden av beholderen, har man lagt til en ekstra feilkilde.

Jording av stålkabinettet burde vært gjort. En jordet kabel ble valgt vekk da spenningsforsyningen ikke skulle ha dette inn, men det burde vært tatt hensyn til sikkerheten til brukeren ved å jorde stålkabinettet.

Mangler ved programmering

Spenningsnivået regnes ut ved formelen gitt av likning 2.8, men i koden er det benyttet komma i tallet 170,6666667 mens språket krever punktum. Dette vil gi en liten feil i utgangsverdi, som vil øke i takt med hastigheten.

3.3.3 Egne tilleggsidéer

Enda en fordel med Arduino er at det finnes utallige tilleggsmoduler man kan benytte seg av. Dette gjør at det er lett å utvide mikrokontrolleren med flere funksjoner, uten å måtte gjøre alt for mye arbeid.

Display: Et display som til enhver tid viser hastighet og/eller utgangsspenning hadde underlettet testingen. Det hadde gjort det enklere å se om man må trimme utgangen med potensiometeret.

Trådløs forbindelse: Dersom man legger til en WiFi- eller bluetooth-modul, kan man slippe å koble seg opp med kabel mot testutstyret. Dette fremmer mobiliteten til testpersonell, slik at de kan være stasjonert lenger unna uten å måtte tenke på faren for kabler som detter ut.

Grafisk brukergrensesnitt: Et grafisk brukergrensesnitt, på PC eller som applikasjon til mobil og nettbrett, hadde gjort det lettere å foreta tester med utstyret. Med trådløs kommunikasjon ville det også ha vært enkelt å starte testen ved å kunne nullstille, lese av og logge testverdier kontinuerlig. Trådløs kommunikasjon til en mobil enhet, med en applikasjon, vil gjøre dette enda enklere og i tillegg øke mobiliteten til testpersonellet.

Bibliografi

- [1] Arduino. *Arduino Due*. <http://arduino.cc/en/Main/ArduinoBoardDue>.
- [2] Arduino.cc. Arduino digital pins. *http* :
[//arduino.cc/en/Tutorial/DigitalPins](http://arduino.cc/en/Tutorial/DigitalPins), 2014.
- [3] Arduino.cc. Arduino introduction. *http* :
[//arduino.cc/en/Guide/Introduction](http://arduino.cc/en/Guide/Introduction), 2014.
- [4] Niklas Bengtsson. Prosjektoppgave: Hastighetsmåling av trede-
mølle.
- [5] Rolv Bræk and Øystein Haugen. *Engineering real-time systems:
an object-oriented methodology*. Prentice Hall, 1993.
- [6] Alan burns and Andrew J. Wellings. *Real-Time Systems and
Programming Languages: Ada, Real-Time Java and C/Real-
Time POSIX*. Addison-Wesley, 2009.
- [7] Kevin Darrah. Rotary encoder tutorial. *http* :
[//www.kevindarrah.com/?page_id=1348](http://www.kevindarrah.com/?page_id=1348), 2012.
- [8] Kevin Darrah. Rotary encoder tutorial. *http* :
[//www.kevindarrah.com/download/arduino_code/Encoder.pde](http://www.kevindarrah.com/download/arduino_code/Encoder.pde),
2012.
- [9] EngBlaze. Faster code fridays: Understand division and speed
of operations. [http://www.engblaze.com/faster-code-fridays-
understand-division-and-speed-of-operations/](http://www.engblaze.com/faster-code-fridays-understand-division-and-speed-of-operations/), 2012.

- [10] Paul Horowitz and Winfield Hill. *The Art of Electronics*. The press syndicate of the university of Cambridge, 1989.
- [11] Intersil. *intersil CA3140*. [http : //www.intersil.com/content/dam/Intersil/documents/ca31/ca3140a.pdf](http://www.intersil.com/content/dam/Intersil/documents/ca31/ca3140a.pdf).
- [12] Y. H. Liu J. Arrillaga and N.R. Watson. *Flexible Power Transmission: The HVDC Options*. JohnWileyandSons, Ltd, 2007.
- [13] Kistler. *64ch DAQ System with BioWare® 4.1, type 5695A*.
- [14] Gregory K. McMillan and Douglas Considine. *Process/Industrial Instruments and Controls Handbook*. McGraw Hill Professional, 1999.
- [15] Omron. *Omron E6C2*. [http : //docs - europe.electrocomponents.com/webdocs/1027/0900766b81027e9e.pdf](http://docs.europe.electrocomponents.com/webdocs/1027/0900766b81027e9e.pdf).
- [16] Omron. *Omron E6C2-CWZ5B*. [https : //www1.elfa.se/data1/wwwroot/assets/datasheets/E6C2 - C_eng_tds.pdf](https://www1.elfa.se/data1/wwwroot/assets/datasheets/E6C2 - C_eng_tds.pdf).
- [17] Odd Pettersen. Digital - analog og analog - digital omsetting. Kompendium i Elektriske kretser.
- [18] Adel S. Sedra and Kenneth C. Smith. *Microelectronic Circuits*. Oxford UNiversity Press, Inc., fifth edition, 2004.
- [19] William Stallings. *Operating Systems: Internals and Design Principles*. Prentice Hall International, 2001.
- [20] Traco Power. *TracoPowe TML 40212C*. [https : //www1.elfa.se/data1/wwwroot/assets/datasheets/tml20 - 40_eng_tds.pdf](https://www1.elfa.se/data1/wwwroot/assets/datasheets/tml20 - 40_eng_tds.pdf).
- [21] westfw. Re: How fast does arduino process each line of code. [http : //forum.arduino.cc/index.php/topic,6632.0.html](http://forum.arduino.cc/index.php/topic,6632.0.html), 2011.

- [22] Jean-Claude Wippler. Pin i/o performance. *http* :
//jeelabs.org/2010/01/6/pin - io - performance/, 2010.

Tillegg A

Forkortelser

DAC Digital til Analog-omformer

Op-Amp Operasjonsforsterker

PWM Puls-Bredde-Modulasjon

BNC Bayonet Neill-Concelman

BE Base-Emitter

KB Kollektor-Base

PPR Pulser per rotasjon

RPM Rotasjoner per minutt

s/R Sekunder per rotasjon

R/s Rotasjon per sekund

I/O Input/Output

CMRR Common mode rejection ratio

K.I.S.S Keep it simple, stupid

Tillegg B

Prosjektoppgave

Tillegg C

Bruksanvisning og guide til demontering/montering

Tillegg D

Komponentliste

Tillegg E

Verdier fra fullskalatesten over
to minutter

Tillegg F

Arduino kildekodefil