



Ingeniørvitenskap og teknologi

## **Solvinkel optimaliserer**

Solcelleanlegg optimalisering

Bjarne Nilsen

Masteroppgave i Electrical Engineering, ELE-3900, mai 2021





# Forord

Temaet for masteroppgaven er optimaliseringsalgoritmen til solcelleanlegg med dobbelaksestyring. En oppgave som handler om algoritme, programmering, montering og testing. Jeg valgte denne oppgaven på grunn av min tidligere erfaring med programmering og fordi den virket interessant. Det har vært spennende å forske på feilsøking og hvordan man kan løse diverse problemer.

Masterstudiet ble litt annerledes enn det jeg hadde sett for meg, i og med at verden ble rammet av koronapandemien, studiestedet stengte ned og mye av undervisningen har foregått digitalt på hjemmekontor. Samvær og samarbeid med medstudenter har også foregått digitalt. Det har krevd selvdisciplin og det har vært lærerikt.

Jeg vil først og fremst takke min veileder, Trond Østrem, førsteamanuensis ved UiT Narvik – Institutt for elektroteknologi, for mye god hjelp. Han har vært veldig tilgjengelig og stilt velvillig opp med veiledningsmøter - også på overtid.

Jeg vil takke lærerne som gjorde skolegangen min mulig. Takk til Ketil Hansen, overingeniør ved UiT Narvik fordi han stilte opp og loddet kontakter for meg, da jeg ikke hadde tilgang på loddelabben.

Jeg vil takke leksekompien min, Kristian Stausland for godt samarbeid gjennom mastergraden.

Til slutt vil jeg takke mine foreldre for god hjelp med å lese korrektur på oppgaven.

Narvik, 15. mai 2021

Bjarne Nilsen

## Sammendrag

Snø og andre ting kan reflektere lys på solcellepanelene slik at de produserer mer energi. Refleksjonene kan gjøre slik at den optimale vinkelen til panelene ikke er rett mot solen. Denne rapporten går fort igjennom noen forslag til optimaliseringsalgoritmer for vinklene til solcellepanelene. Videre går den dypere inn på en algoritme som heter «test og sjekk».

Kort fortalt går denne algoritmen ut på å endre vinklene til panelene, for så å sjekke om den avgitte effekten økte eller minket. Hvis effekten økte, skal den igjen øke vinkelen i samme retning. Hvis effekten minket skal den stille panelene tilbake for så å teste en vinkel i en annen retning. Når den har testet alle retningene uten at den har klart å øke produksjonen betyr det at den har nådd et maksimumspunkt. Algoritmen skal vente en stund når den finner et maksimumspunkt.

Rapporten skal handle litt om hvordan diverse utstyr og hvordan det skal implementeres for at systemet skal fungere.

Deretter går den inn på en PLS-kode som skal styre en solcellerigg med dobbelaksstyring, der test-og-sjekk algoritmen nevnt over skal benyttes.

Utstyret ble montert og testet på benk. Til slutt går den igjennom diverse testresultater og foreslår videre arbeid.

## Nomenklatur

$LSTM$	lokal-standard-tid-meridian
$\Delta T_{UTC}$	Tidssone
$EoT$	«Equation of time» en tidskorreksjonsfaktor
$TC$	En tidskorreksjonsfaktor
$LST$	Lokalsoltid – tidssone med oppløsning på minutter
$LT$	Lokaltid
$HRA$	Timevinkel – hvor mange grade solen har beveget seg siden 00:00 (lokalsoltid)
$\delta$	Deklinasjonsvinkelen
$\alpha$	Elevasjonsvinkelen til solen
$d$	Dagsnummeret - dager siden 1.jan
<i>Asimut</i>	Asimutalvinkel til solen
$S_{vind}$	Sensitiviteten til vindmåleren
$U_{vind-maks}$	Høyeste mulige spenning ut av vindmåleren
$V_{vind-maks}$	Høyeste målbare vindhastigheten til vindmåleren
$S_{pyranometer}$	Sensitiviteten til pyranometret
$U_{pyranometer-maks}$	Høyeste mulige spenning ut av pyranometret
$E_{pyranometer-maks}$	Høyeste målbare innstrålingen til pyranometret
$R_{opto}$	Strømbegrensningsmotstand til optokobler
$U_{signal-opto}$	Signalspenningen til optokobler
$U_{diode-opto}$	Diodespenning til optokobler
$I_{diode}$	Diodestrømmen til optokobler
$U_{kilde-opto}$	Spenningen til transistorsiden til en optokobler
$I_{transistor}$	Transistorstrømmen til en optokobler
$R_{tran-opto}$	Strømbegrensningsmotstand til en optokobler

$G$	Forsterkning
$R_{f-opamp}$	Tilbakekoblingsmotstand til operasjonsforsterker
$R_{1-opamp}$	Jordingsmotstand til den inverterende inngangen til operasjonsforsterker
$\vartheta$	Girutvekslingen til motorene
$\omega$	Vinkelhastighet til riggen
$X$	Utlest verdi
$\theta_{ny}$	Ny posisjon
$\theta_{gammel}$	Gammel posisjon
$\Delta t$	Delta tid, tidsforskjellen mellom hver gang PLS-en integrere, gitt i sekunder.
$\omega_{gammel}$	Gammel vinkelhastighet til riggen
$\omega_{ny}$	Ny vinkelhastighet til riggen
$E$	Innstrålt effekt
$U_{maks-A/D}$	Den høyeste målbare verdien til 0-10V A/D omformerer
$E_{maks}$	Den høyeste målbare verdien pyranometret
$U_{pyranometer-maks}$	Den høyeste mulige spenningen ut av pyranometret
$X_{maks}$	Den høyest mulige utleste verdien til A/D omformerne
$V_{maks}$	Den høyeste målbare verdien til vindmåleren
$V_{grense}$	En gitt grenseverdi for når riggen skal gå i <b>vind</b> -modus
$X_{vindgrense}$	Den utleste verdien for den gitte vindgrensen
$V$	Vindhastighet

# Innholdsfortegnelse

Sammendrag.....	3
Nomenklatur.....	4
1.1 Ulike strategier for optimalisering.....	1
Første algoritme – test og sjekk.....	1
Andre algoritme – Sirkler med økende radius .....	2
Tredje algoritme – Kontinuerlig spiral.....	3
Fjerde algoritmen – Sikksakk.....	3
1.2 Solens posisjon [1].....	4
2 Om riggen.....	6
3 Komponentene .....	7
3.1 Motorene.....	7
3.2 PLS-modulene .....	7
Frekvensteller Wago 750-404/000-003.....	7
0-10V DC analog-til-digitalomformer Wago 750-459 .....	8
4-20mA analog-til-digitalomformer Wago 750-455.....	8
24V DC digitalutgang Wago 750-504 .....	8
Real Time Clock (RTC) Wago 750-640 .....	8
Strømforsyninger.....	8
3.3 Sensorene.....	9
0-10V analog vindsensor.....	9
4 Praksis .....	11
4.1 Oppkoblinger .....	11
Kobling for drivning av motorene.....	11
Kobling for hallsensor og tilhørende krets .....	12
Kobling for pyranometret og tilhørende krets.....	13
Kobling for vindsensor med tilhørende krets .....	14
Kobling av induksjonssensorene og tilhørende krets.....	14
4.2 Implementeringer.....	15
Implementering av RTC-modulen .....	15
Implementering av posisjonssystem.....	15
Implementering av drivning av motorene .....	16
Implementering av pyranometret .....	16
Implementering av vindsensoren .....	17
Implementering av induksjonssensorene .....	17
5 Programmet .....	18

5.1	Koden.....	18
5.2	Hovedmodiene.....	20
	Følg-Solen-modusen .....	21
	Kjør-algoritmen-modusen .....	22
6	Testresultat .....	25
6.1	Test 1 .....	25
6.2	Test 2 .....	28
6.3	Test 3 .....	30
6.4	Test 4 .....	31
6.5	Test 5 .....	31
7	Videre arbeid .....	32
	Referanseliste .....	33
	Vedlegg .....	34
	Koden .....	34
	Datablad til motoren.....	51
	Datablad til pyranometret.....	54
	Datablad til induksjonssensorene .....	54
	Datablad til Vindmåleren Product number 4.3303.22.661.....	54
	Datablad til optokoblerne .....	54
	Datablad til operasjonsforsterkerne.....	54
	Oppgavetekst.....	54

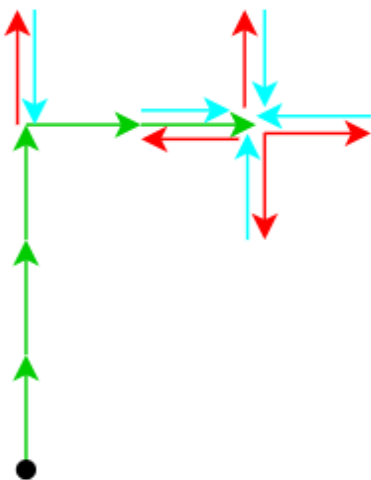


## 1.1 Ulike strategier for optimalisering

Siden alle algoritmene bruker en test-og-sjekkmetoden med hensyn til produksjon er de utsatt for ytre forstyrrelser som endrer produksjonen, for eksempel skyer, fugler og andre ting som kan skygge for panelene. Hvis riggen er for rask, kan det interferere med bestpunktstyringsalgoritmen til dc-dc omformerer til solcellene. Dette kan bli hindret ved å holde panelene stille og måle effekten over en periode. Deretter velge den største målte effekten som den gjeldende måleverdien, dette vil filtrere ut støy. Alle algoritmene starter med panelene vinkelrett mot solen. Referer til kapittel solens posisjon [1] **Feil! Ugyldig selvreferanse for bokmerke.** for utregning av solposisjon. Forfatteren foreslår de følgende optimaliseringsmetodene.

### Første algoritme – test og sjekk

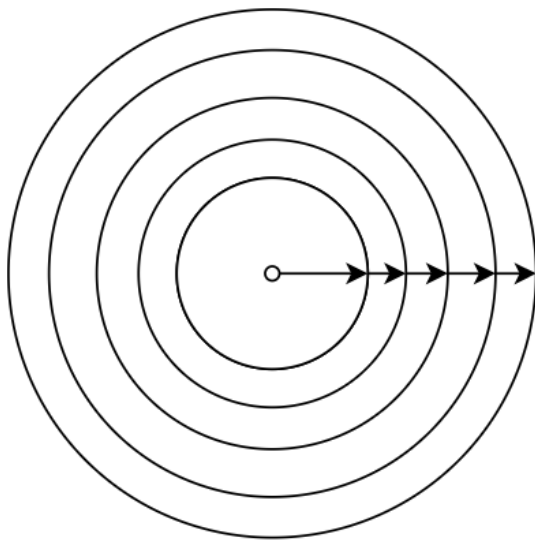
Figur 1 illustrerer test og sjekk algoritmen. Kjør panelene i et stykke i en retning og sjekk om produksjonen øker. Hvis produksjonen øker kan den kjøre lengere i den samme retningen, hvis ikke kan den gå tilbake å prøve en annen retning. Når anlegget har testet i fire retninger og ingen av dem øker produksjonen har anlegget nådd et maksimumspunkt. Denne algoritmen kan kanskje nå et lokalt maksimum og i verstefall miste synkronisering med solen. Derfor kan det være lurt å sjekke avstanden mellom algoritmereferansen med utregnet solreferanse og sette in et maksimum tillatt avstand. Hvis avstanden overstiger en gitt verdi skal solreferansen overta, og det hadde vært fint med en feilmelding slik at man får vite om dette er et reelt problem. Dette er den enkleste algoritmen og den kan kjøres kontinuerlig.



Figur 1 test og sjekk

## Andre algoritme – Sirkler med økende radius

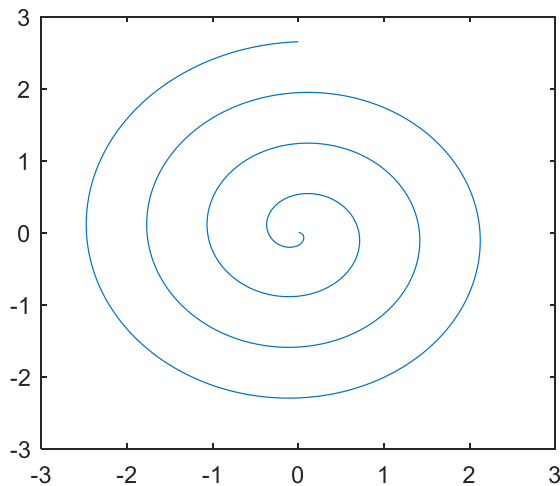
Figur 2 illustrerer sirkler med økende radiusalgoritmen. Denne algoritmen kjører i sirkler rundt den kalkulerte sol-referansen med større og større radius, og loggfører effekten med de korresponderende vinklene. Deretter tilter den panelene til de vinklene med den høyeste effekten. Fordelen med denne algoritmen er at den vil ikke nå et lokalt maksimum eller miste synkronisering med solen. Ulempen er at det vil ta lang tid å skanne så mange vinkler og at motorene må slås mye av og på for at vinkelkoordinatene skal danne en sirkel. Det kan føre til at gjennomsnittshastigheten til panelene blir lav. Dette kunne vært løst med motorer med lavere girutveksling og varierende hastighet. Denne algoritme vil ha mulighet for å finne ut om lokalt maksimum er et reelt problem.



*Figur 2 sirkler med økende radius*

### Tredje algoritme – Kontinuerlig spiral

Figur 3 illustrerer kontinuerlig spiralalgoritmen. Starter med den kalkulererte sol-referansen og legger på et tillegg som er lik en vektor med gradvis økende vinkel og økende lengde. Dette vil resultere i et lignede system som i den andre algoritmen, men med et kontinuerlig mønster. Dermed vil den ha noen av de samme styrkene og svakhetene. Denne algoritmen hadde vært mer relevant med motorer med varierende hastighet og lavere girutveksling.



Figur 3 kontinuerlig spiral

### Fjerde algoritmen – Sikksakk

Figur 4 illustrerer sikksakkalgoritmen. I likhet med den andre og tredje algoritmen skanner den mange vinkler og velger den som yter høyest effekt. Den kan gå for eksempel fra øst til vest og deretter tilte litt ned og så gå fra vest til øst og så videre. Dette lager rette linjer som vil gjøre slik at motorene ikke trenger å bli slått av og på mange ganger for å lage en sirkel. En ulempe er at det vil ta lang tid hvis riggen er treg.



Figur 4 sikksakk

## 1.2 Solens posisjon [1]

Solens posisjon regnes ut med å bruke formler fra [1], med bredde og høydegrad kombinert med klokke og dato som input. LSTM er lokal-standard-tid-meridian som er en referanselinje for tidssonen som går fra nord til Sørpolen hvor solen passerer over klokken 12:00, og den regnes ut etter formel (1), hvor  $\Delta T_{UTC}$  er tidssonen til området. For Norge vil  $\Delta T_{UTC}$  være +1 for vintertid og +2 for sommertid.

$$LSTM = 15^\circ * \Delta T_{UTC} \quad (1)$$

$EoT$  er en korreksjonsfaktor som regnes ut etter formel (2). Denne faktoren korrigerer for eksentrisiteten til jordens bane, og den aksiale hellingen til jorden.

$$EoT = 9.87 \sin(2B) - 7.53 \cos(B) - 1.5 \sin(B) \quad (2)$$

Hvor  $B$  regnes ut etter formel (3), og  $d$  er dagsnummeret, som er antall dager fra og med 1. januar i innværende år til den bestemte dagen.

$$B = \frac{360}{365}(d - 81) \quad (3)$$

$TC$  er en tidskorreksjonsfaktor som korrigerer for avvik mellom breddegraden og lokal-standard-tid-meridian, den inkluderer og  $EoT$ . Siden jorden roterer  $1^\circ$  hvert fjerde minutt er det en faktor på fire, og  $TC$  regnes ut i formel (4).

$$TC = 4(\text{Lengdegrad} - LSTM) + EoT \quad (4)$$

Lokal soltid er som en minitidssone med oppløsning på minutter, og er definert slik at solen står nøyaktig i sør observert fra den nordlige halvkulen, og nøyaktig i nord observert fra den sørlige halvkulen når klokken er 12:00. Lokal soltid eller LST vil avvike fra klokken med minutter med mindre man befinner seg på LSTM. LST er gitt ved (5), der LT er lokaltid oppgitt timer.

$$LST = LT + \frac{TC}{60} \quad (5)$$

Timevinkel eller  $HRA$  konverterer  $LST$  til hvor mange grader solen har beveget seg over himmelen. Timevinkelen er per definisjon  $0^\circ$  klokken 12:00. Solen beveger seg  $15^\circ$  hver time, dette gir formel (6).

$$HRA = 15^\circ(LST - 12) \quad (6)$$

Jorden har en helling på  $23.45^\circ$  sammenlignet med planet til banen rundt solen. Dette gir jorden en helling sammenlignet med solen som varierer gjennom året, kjent som deklinasjon. Formel (7) regner ut deklinasjonen, hvor  $d$  er dagsnummeret og  $\delta$  er deklinasjonsvinkelen.

$$\delta = 23.45 \sin \left[ \frac{360}{365} (d - 81) \right] \quad (7)$$

Elevasjonsvinkel til solen regnes ut ved formel (8), hvor  $\alpha$  er elevasjonsvinkelen,  $\delta$  er deklinasjonsvinkelen,  $\varphi$  er høydegraden til området og  $HRA$  er timevinkelen.

$$\alpha = \sin^{-1}[\sin\delta \sin\varphi + \cos\delta\cos\varphi \cos(HRA)] \quad (8)$$

Asimutalvinkelen er en vinkel i himmelretningsplanet med nullpunkt mot nord og positiv retning mot øst. Hvis  $LST$  er større enn 12 er asimutalvinkelen gitt ved (9), hvis  $LST$  er mindre enn 12 er den gitt ved (10), hvor  $\delta$  er deklinasjonsvinkelen,  $\varphi$  er høydegraden til området,  $HRA$  er timevinkelen og  $\alpha$  er elevasjonsvinkelen.

$$Asimut = 360 - \cos^{-1} \left[ \frac{\sin\delta\cos\varphi - \cos\delta\sin\varphi\cos(HRA)}{\cos\alpha} \right] \quad (9)$$

$$Asimut = \cos^{-1} \left[ \frac{\sin\delta\cos\varphi - \cos\delta\sin\varphi\cos(HRA)}{\cos\alpha} \right] \quad (10)$$

## 2 Om riggen

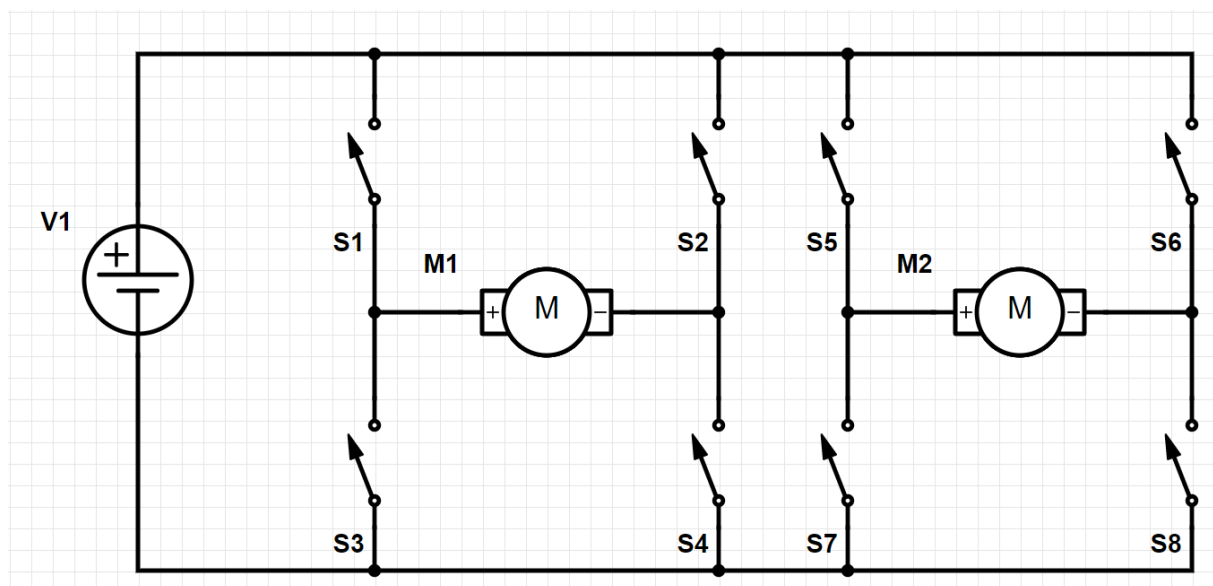
Riggen består av en mast med 2-aksemotor med ormegir på toppen. Ormegir vil låse orienteringen til panelene med mindre motorene drives. Det er festeramme for solcellepanel. Hvis motoren kjøres for langt i elevasjonsretningen vil rammen kjøres i masten og riggen vil bli skadet. Hvis motorene kjøres for langt i asimutalretningen vil det bli strekk i kablene slik at noe vil bli skadet.

Det skal monteres fire induksjonssensorer som skal oppdage når braketten er i ferd med å treffe masten, og når den har rotert til en grenseverdi, da skal riggen stoppe. To av sensorene skal benyttes til å nullstille posisjon i programmet. Det er tenkt at det skal være en fysisk sikring som for eksempel en mekanisme som kutter kablene til motoren hvis riggen kjøres for langt i en retning, i tilfelle induksjonssensorene svikter.

Motorene har to hallsensorer som vil sende informasjon til PLS-en i form av et pulstog som vil bli regnet om til posisjon, slik at PLS-en til enhver tid vet hvordan riggen er orientert.

Riggen skal utstyres med en vindmåler og når vindstyrken overstiger en gitt verdi skal panelene orienteres horisontalt slik at det blir mindre vindlast.

Anlegget har to 24 volts DC motorer som skal kobles med en dobbel H-bru, i henhold til Figur 5, slik at man kan endre polariteten og dermed retningen motoren roterer. Figuren viser at hvis S1 og S4 er lukket vil motor-1 rotere i positiv retning, og hvis S2 og S3 er lukket vil motor-1 rotere i negativ retning. Det er viktig at bare ett sett med reléer per motor er lukket på en gang, ellers vil det oppstå en kortslutning. Motorenes dreieretning må være i samsvar med programmet til PLS-en.



Figur 5 H-bru prinsippskjema

## 3 Komponentene

### 3.1 Motorene

Motorene er 24V DC, utstyrt med ormegir med en utveksling på 17228:1. Ormegirene låser motorakslingen når den ikke drives av motoren. Motorene kan kjøres i begge retningen ved å endre polariteten. Det er to halleffektsensorer på hver motor som lager et pulstog hver, med to pulser per omdreining. Det er viktig at hallsensorene ikke kobles feil da de vil bli ødelagt hvis de får for høy spenning eller feil polaritet. Motorene har et nominelt turtall på 2200rpm, og trekker opptil 7A under last.

### 3.2 PLS-modulene

#### Frekvensteller Wago 750-404/000-003

Pulstogene vil bli telt av to frekvenstellere-moduler, som heter Wago 750-404-000-003. Den teller frekvensen ved måletiden til periodene til et 24V signal, som blir sendt inn i en klokkeinngang. Der -3V til 5V telles som et lavt signal og 15V til 30V telles som et høyt signal. Modulene er i stand til å telle frekvenser mellom 0,1Hz og 100kHz. De har fire modi som man kan velge ved og stille kontrollbitene «RANGE SEL1» og «RANGE SEL2». Tabell 3-1 viser hvordan kontrollbitene påvirker karakteristikken til modulene.

Tabell 3-1 Innstilling av kontrollbitene til frekvenstillerne

RANGE SEL1	RANGE SEL2	Måleperiode	Måleintervall	Oppgitt verdi
0	0	1 periode	0,1 til 8 kHz	1/1000Hz
0	1	4 periode	0,25 til 32 kHz	1/100Hz
1	0	16 periode	1 til 100 kHz	1/10Hz
1	1	16 periode	1 til 100 kHz	Hz

Modulen har en såkalt «watch dog timer», det er en stoppeklokke som sier hvor lenge den skal vente på den neste økende flanke til at den oppgir at frekvensen er 0, der tiden oppgis i millisekunder. Altså «watch dog timeren» brukes til å sette en minimumsgrense for den innleste frekvensen. Modulen har to digitalutganger på 24V og 0,5A, som er kortslutningssikret. Disse utgangene kan brukes til å forsyne klokkesignalet med 24V spenning, og drive en 24V inngang som heter «Gate». Siden modulen måler spenningen mellom inngangene og det interne jordpotensialet. Hvis en 24V spenning blir oppdaget på «Gate» vil modulen slutte å telle frekvensen.

## **0-10V DC analog-til-digitalomformer Wago 750-459**

Modulene har fire inngangskanaler. Alle kanalene deler samme jordpunkt, som er isolert fra jordpunktet til PLS-en internt. De har et målingsintervall på 0-10V, en inngangsresistans som er over 100k $\Omega$ , og de tåler ikke mer enn  $\pm 40V$ . Spenningen blir lest inn i datatypen WORD som er en fortegnsløs 16 bit datatype. De tre første bitene er reserverte statusbit, og den siste biten er ikke i benyttet. Det vil si at utgangsverdien har en numerisk oppløsning på 8 og vil gå fra 0 til 32760. Verdien 3 er reservert for spenning under 0V, og verdien 32763 er reservert for spenning over 10V. Dette gir en måleoppløsning på 12 bit til inngangsspenningen. Modulene har en toleranse på  $\pm 0,1\%$  av fullskalaverdi ved 25°C. Hver kanal har en korresponderende varsellampe som lyser hvis målespenningen er utenfor målingsintervallet.

Det benyttes fire moduler av denne typen.

## **4-20mA analog-til-digitalomformer Wago 750-455**

Disse modulene har fire inngangskanaler. Alle kanalene deler samme jordpotensial, som er isolert fra resten av PLS-en internt. De har et målingsintervall på 4-20mA, en inngangsresistans som er under 100 $\Omega$  ved 20mA, og de tåler ikke mer enn 32V. Strømmen blir lest inn i datatypen WORD. De tre første bitene er reserverte statusbit, og den siste biten er ikke i benyttet. Det vil si at den innleste verdien har en numerisk oppløsning på 8 og går fra 0 til 32760. Verdien 3 er reservert for strøm under 4mA, og verdien 32763 er reservert for strøm over 20mA. Dette gir en måleoppløsning på 12bit til inngangsstrømmen. Modulene har en toleranse på  $\pm 0,1\%$  av fullskalaverdi ved 25°C. Hver kanal har en korresponderende varsellampe som lyser hvis målestrømmen er utenfor målingsintervallet.

Det benyttes to moduler av denne typen.

## **24V DC digitalutgang Wago 750-504**

Modulen har fire 24V DC digitalutganger som har en maks strøm på 0,5A og er korslutningssikret, men den har ikke overbelastningsvern. Induktive laster bør styres med friløpsdioder, slik at de mulige overspenningene ikke vil skade modulen. Den har ingen jordterminal. Digitalutgangene brukes til å drive reléene som skal styre motorene til riggen. Hver kanal har et korresponderende lys på modulen, som lyser når utgangen er høy.

## **Real Time Clock (RTC) Wago 750-640**

Denne modulen har mulighet for å koble til en antenne, da har den mulighet til å ta inn sanntidsklokke og dato via et satellittsignal. Antenne kan forsynes med 24V direkte fra modulen. Modulen blir brukt med et bibliotek som har en standardfunksjon slik at man enkelt får lest ut tidspunkt og dato. Denne informasjonen blir brukt til å estimere solens posisjon. Det er et statuslys på modulen som indikerer om den har antennesignal, feil med antennesignalet, har signal og venter på rett telegram, og har signal uten noen feil.

## **Strømforsyninger**

PLS-en har en 24V DC strømforsyning som forsyner PLS-en og alle modulene, enten via den interne busen eller via shunter. Motoren og vindmåleren forsynes av en annen 24V DC forsyning. Begge strømforsyningene har intern kortslutningssikring.



## 3.3 Sensorene

### Induksjonssensorer

Disse sensorene trenger en to-leder-kabel som forsyner en spenning på 12-24V DC, strømsignalet går gjennom den samme kabelen som forsyner sensoren. De generer en målestrøm som er i området 4-22mA. Når det ikke er et ferromagnetisk objekt foran sensorene vil målestrømmen være 22mA, og strømmen går ned mot 4mA når det er ferromagnetisk objekt foran, hvor mye strømmen går ned vil være avhengig av ferromagnetismen og avstanden til objektet.

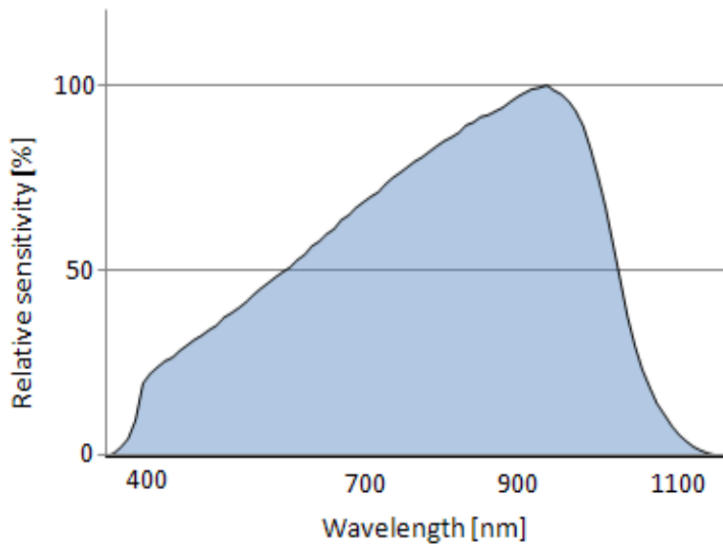
### 0-10V analog vindsensor

Den har et måleområde fra 0,3 til 60m/s, med en toleranse på  $\pm 2,5\%$  av den målte verdien. Det er et 20W varmeelement som skal hindre at den fryser fast, varmeelementet trengs ikke å brukes. Den har en fempolet plugg som kan kobles på to forskjellige måter. Hvordan pluggen blir koblet vil avgjøre om varmeelementet brukes eller ikke. En av polene skal ikke kobles til noe, to av de skal forsyne måleren med 24V DC, og to av de skal sende et 0-10V signal. Signalkabelen bør skjermes. Spenningen til signalet er lineært med vinden, og sensitiviteten er 0,167Vs/m, gitt ved (11).

$$S_{vind} = \frac{U_{vind-maks}}{V_{vind-maks}} \quad (11)$$

## Pyranometer

Responskurven til pyranometret med tanke på bølgelengden til lyset er gitt ved Figur 6, der bølgelengden er på x-aksen og relativsensitivitet er på y-aksen. Den måler direkte og indirekte lys, har samme responstid og temperaturrespons som en PV-modul. Videre har den et synsfelt på 180° og en skjermet signalkabel.



Figur 6 Pyranometrets responskurve med tanke på bølgelengden til lyset [2]

Utgangsspenning til pyranometret er i området 0-100mV, som er lineært med målingsintervallet på 0-2000W/m<sup>2</sup>. Det gir en sensitivitet på 0,05mV\*m<sup>2</sup>/w i henhold til (12).

$$S_{pyranometer} = \frac{U_{pyranometer-maks}}{E_{pyranometer-maks}} \quad (12)$$

## 4 Praksis

Solcelle riggen ble ikke montert og drift satt, men PLS-en, induksjonssensorene, vindmåleren, pyranometret, H-bruen til motorene, motorene og hallsensorene ble montert og testet på benk. Strøm og spenningsmålingene til solcelleanlegget ble ikke montert eller testet.

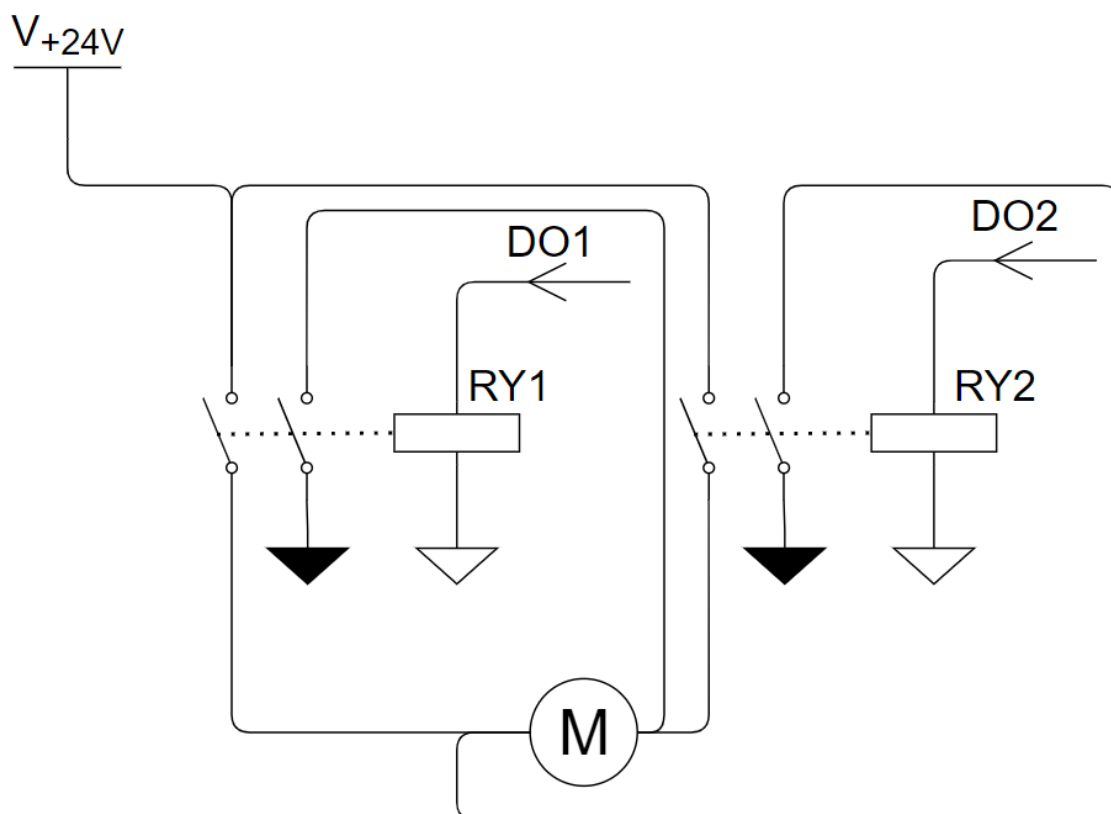
### 4.1 Oppkoblinger

Dette kapitlet skal gå igjennom hvordan utstyret ble koblet. Noen av PLS-modulene trenger strøm via en ekstern 24V tilkobling. Det er slik fordi, hvis man har mange moduler så skal de ikke overbelaste spenningskilden eller den interne bussen.

#### Kobling for drivning av motorene

Figur 7 viser hvordan selve H-bruen ble koblet. Reléene får spenning fra PLS-en og motoren får spenning fra en annen spenningskilde. Reléene RY1 og RY2 utgjør H-bruen for den ene motoren, H-bruen for den andre motoren ville vært koblet i parallell og vært identisk med den første H-bruen. Reléene og PLS-en har samme jording, men motorene har egen jording.

Det kan hende det bør være friløpsdioder på reléene slik at PLS-en ikke blir skadet over tid.



Figur 7 Koblingsskjema for driving av motorene

## Kobling for hallsensor og tilhørende krets

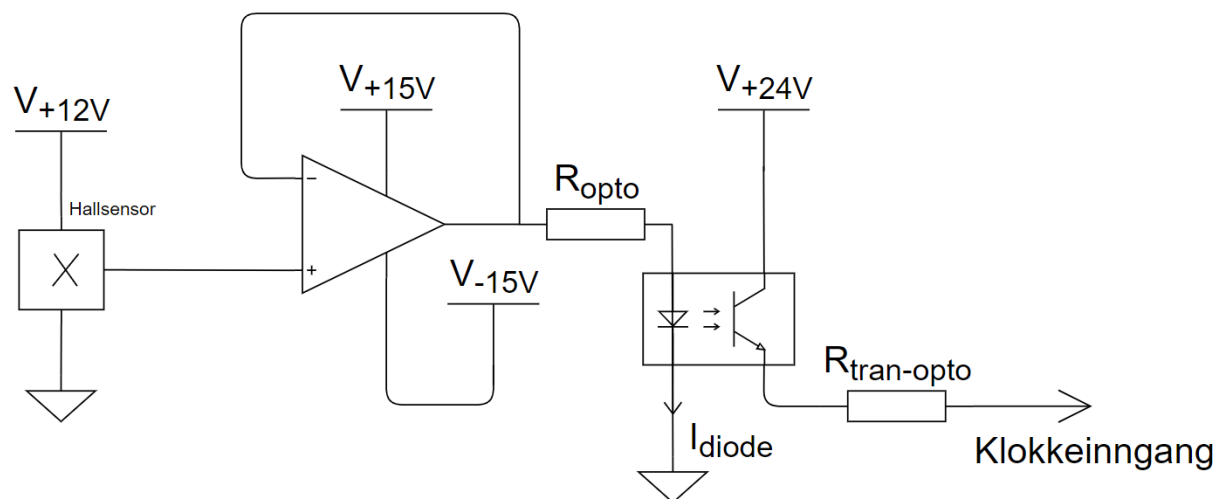
Figur 8 viser koblingsskjema for hallsensor og tilhørende krets. Hallsensoren er ikke i stand til å levere mer enn 1mA, som ikke er nok til å drive en optokobler. Derfor bufres signalet med operasjonsforsterker med «unity gain». En symmetrisk ±15V spenningskilde ble laget ved å seriekoble to 15V DC kilder. Hallsensoren ble forsynet med en 12V DC kilde. Signalet til hallsensoren vil være 12V når den er høy og 0V når den er lav. Diodestrømmen til optokobleren må være mellom 16-100mA, og den ble valgt til 20mA. En strømbegrensningsmotstand for dioden til optokobleren på 540Ω ble valgt, gitt ved (13). Der  $U_{signal-opto}$  er signal spenningen,  $U_{diode-opto}$  er diodespenningen og  $I_{diode}$  er diodestrømmen.

$$R_{opto} = \frac{U_{signal-opto} - U_{diode-opto}}{I_{diode}} \quad (13)$$

Transistorstrømmen kan være opp til 100mA derfor ble en transistorstrøm på 20mA valgt. En strømbegrensningsmotstand for transistoren til optokobleren på 1200Ω ble valgt for sikkerhetsskyld, gitt ved (14). Der  $U_{kilde-opto}$  er spenningen til kilden for transistorsiden og  $I_{transistor}$  er den ønskete transistorstrømmen. Det kan hende at motstanden til frekvenstilleren er mer enn 1200Ω, da er det ikke nødvendig med en strømbegrensningsmotstand til transistorsiden.

$$R_{tran-opto} = \frac{U_{kilde-opto}}{I_{transistor}} \quad (14)$$

PLS-en gir 24V fra en av digitalutgangene som går igjennom transistorene og strømbegrensningsmotstandene og deretter inn på klokkekontaktene. Frekvenstilleren teller de økende flankene til et 15-30V pulstog mellom klokkekontakten og det interne jordpunktet.



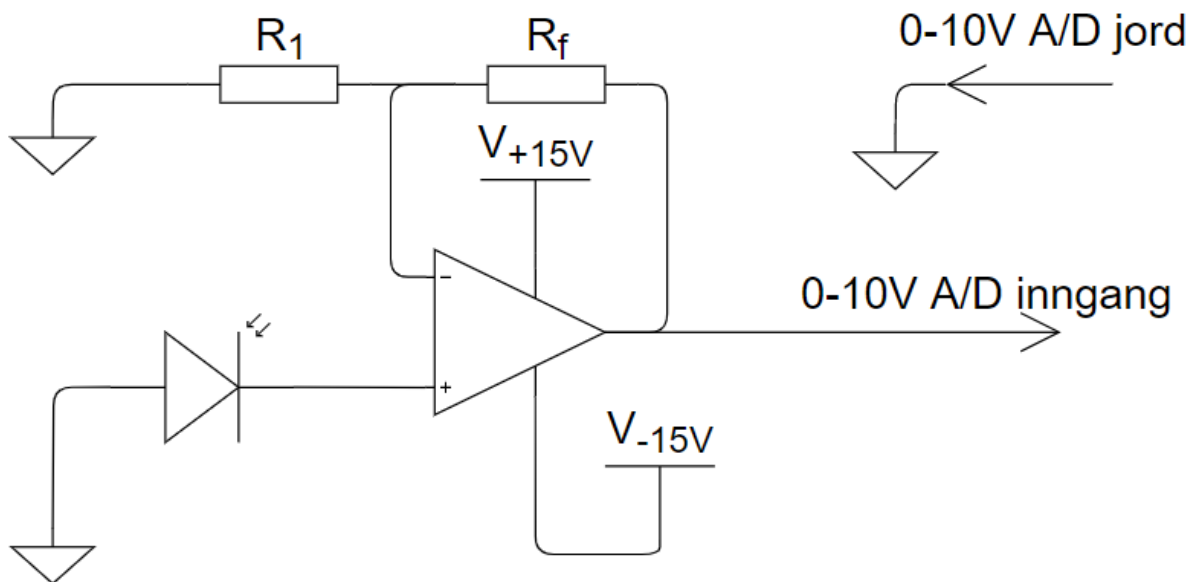
Figur 8 Koblingsskjema for hallsensor og tilhørende krets

## Kobling for pyranometret og tilhørende krets

Figur 9 viser koblingsskjema for pyranometret med tilhørende krets. Siden utgangspenningen til pyranometret er i området 0-100mV og analog-til-digitalmodulen har et måleintervall på 0-10V DC må spenningssignalet forsterkes. Dette gjøres med en ikke-inverterende operasjonsforsterker. En forsterkning på 100 ganger vil gjøre slik at det høyeste utslaget fra pyranometret vil gi en spenning på 10V inn på modulen. På grunn av at det var et begrenset utvalg av motstander, ble forsterkningen 101 ganger gitt ved (15).

$$G = 1 + \frac{R_f}{R_1} \quad (15)$$

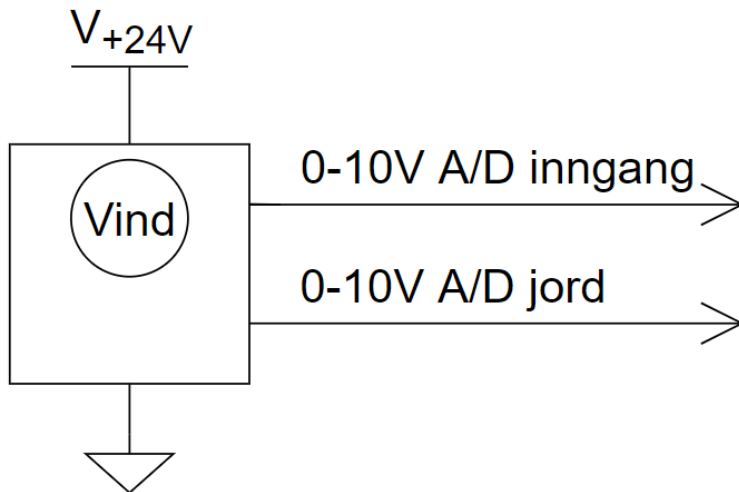
Denne forsterkeren har samme spenningskilde som operasjonsforsterkerne til hallsensorene. Utgangen til pyranometerforsterkeren kobles på inngangene til en av de fire 0-10V analog-til-digital-omformerne, og jordpunktet kobles til en av de fire jordpunkts-kontaktene til modulen.



Figur 9 Koblingsskjema for pyranometret og tilhørende krets

## Kobling for vindsensor med tilhørende krets

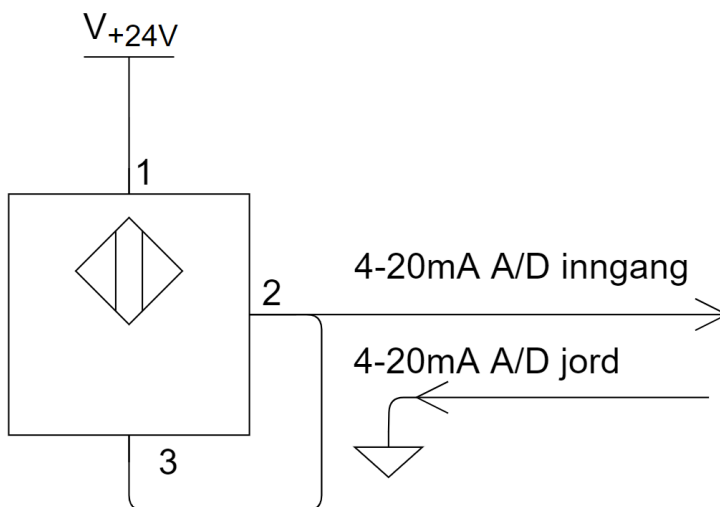
Vindsensorene trenger 24V DC og 20W for oppvarming, det er mulig å koble den slik at den ikke bruker varmeelementet, da trenger den bare 24V. Strømforsyningen til motorene brukes, slik at strømforsyningen til PLS-en ikke risikerer å bli overbelastet, siden strømforsyningen til PLS-en er bare på 60W. Figur 10 viser koblingsskjema for vindsensoren. Signalkabelen bør skjermes, men det ble brukt to par tvinnede en-leder-kabler for benktesten.



Figur 10 Koblingsskjema for vindsensor og tilhørende krets

## Kobling av induksjonssensorene og tilhørende krets

Figur 11 viser koblingsskjema for induksjonssensorene. Sensoren får 24V fra PLS-en, da må en av jordterminalene til modulen kobles til jordingen til PLS-en. Inni induksjonssensorene er det tre nummererte skruer, der to av de skal kobles sammen, og de vil være utgangen til sensoren. Utgangen til sensorene skal kobles på inngangene til 4-20mA analog-til-digital-omformereren.



Figur 11 Koblingsskjema for induksjonssensorene med tilhørende krets

## 4.2 Implementeringer

### Implementering av RTC-modulen

RTC-modulen trenger ingen koblinger for å bli brukt, men den må ha en antenne tilkoblet for at den skal kunne ta inn signaler fra satellittene til å stille klokken. En antenne ble ikke tilkoblet, dermed ble klokken til PLS-en feil. Det er mulighet for å stille klokken til PLS-en manuelt i koden. En feilkode i PLS-en indikerte at den ikke hadde antennesignal.

### Implementering av posisjonssystem

Motoren har mulighet til å lage to pulstog som har to pulser per omdreining. Dette gir mulighet til å vite posisjonen til motorakslingen med en oppløsning på  $180^\circ$ . Orienteringen til riggen vil da ha en maksimumsoppløsning på  $0,01^\circ$ , gitt ved (16), der  $\vartheta$  er utvekslingen til motorene. Tregghetsmomentet til motoren vil gjøre slik at det ikke vil være mulig å kjøre riggen med en oppløsning på  $0,01^\circ$ , med en H-bru og uten et avansert kontrollsystem, da riggen vil ha marginalstabilitet.

$$\frac{180^\circ}{\vartheta} \quad (16)$$

Referansevinkelene og posisjonsvinklene konverteres fra typen LREAL til INT, altså hele tall. Dette gjør at riggen vil operere med en oppløsning på  $1^\circ$ , selv om riggen har en maksimumsoppløsning på  $0,01^\circ$ . Dette vil gi en toleranse på  $\pm 0,5^\circ$ , som vil oppføre seg som en hysteresis effekt, og føre til at motorene ikke trenger å starte og stoppe så ofte.

Hallsensorene til motorene genererer to pulser per omdreining, og motorenes nominale turtall 2200 omdreininger i minuttet. Derfor stilles begge kontrollbitene til «0» slik at modulen teller frekvensen på den mest nøyaktige måten. Da vil modulene oppdatere de utleste verdien hver puls, og oppgir de med en oppløsning på 0,001Hz, og de vil ha et måleintervall på 0,1Hz til 8kHz. Dette gir et forhold mellom den utleste verdien og vinkelhastighet til riggen i henhold til (17). Der  $X$  er den utleste verdien, faktoren på 360 er for å gjøre det om til grader, det må deles på to fordi det er to pulser per omdreining, det må deles på 1000 fordi den oppgis i 0,001Hz og den må deles på gir forholdet som er  $\vartheta$ .

$$\omega = \frac{X*360}{2*1000*\vartheta} \quad (17)$$

Posisjonene til riggen regnes ut ved å integrere vinkelhastigheten diskret med respekt til tiden, i henhold til (18). Der  $\theta_{ny}$  er ny posisjon,  $\theta_{gammel}$  er gammel posisjon,  $\omega$  er vinkelhastigheten til riggen og  $\Delta t$  er tidsintervallet som vinkelhastigheten skal integreres over.

$$\theta_{ny} = \theta_{gammel} + \omega * \Delta t \quad (18)$$

Vinkelhastigheten har positivt fortegn hvis motoren går framover og negativt fortegn hvis den går bakover. Det er viktig at integratoren fortsetter å integrere etter at motoren ikke har spenning, fordi tregheten til motorene vil drive motoren et lite stykke lengere. Ved å bruke tidsstempel kan PLS-en regne ut hvor mye  $\Delta t$  skal være for hver syklus til programmet. Modulen oppdaterer den utleste verdien hver gang den oppdager en økende flanke, ellers venter den i 10 sekunder eller til at «watch dog timer»-en har utløpt til at den oppdaterer verdien og sier at den er 0Hz. «Watch dog timer»-en stilles til 200ms slik at integratoren ikke fortsetter å integrer lenge etter at motorene har stoppet.

For å minimere avvike mellom utregnet posisjon og faktisk posisjon implementeres trapesmetoden. Da blir utregnet posisjon gitt ved (19), der  $\omega_{ny}$  er den nye vinkelhastighet og  $\omega_{gammel}$  er den gammel vinkelhastighet.

$$\theta_{ny} = \theta_{gammel} + (\omega_{gammel} + \frac{\omega_{ny} - \omega_{gammel}}{2}) * \Delta t \quad (19)$$

## Implementering av driving av motorene

Når en posisjonsvinkel til riggen er mindre enn referansevinkelen som riggen skal følge, kjøres den tilsvarende motoren framover, når den er større kjøres den i revers, og når de er lik kjøres ikke motoren. Det er viktig at samme motor ikke kjøres i begge retningene samtidig, da det vil resultere i en kortslutning.

## Implementering av pyranometret

Pyranometret kan brukes til å loggføre lysstyrke. Forholdet mellom den utleste verdien og innstrålt effekt er gitt ved (20), der  $U_{maks-A/D}$  er den høyeste måleverdien til 0-10V-modulen, faktoren på  $E_{maks}$  er den største måleverdien til pyranometret,  $G$  er forsterkningen til signalet til pyranometret, og faktoren på  $X_{maks}$  er den største mulige innleste verdien.

$$E = \frac{X * U_{maks-A/D} * E_{maks}}{G * U_{pyranometer-maks} * X_{maks}} \quad (20)$$



## Implementering av vindsensoren

Når vindstyrken overstiger en gitt grense, skal riggen gå i vindmodus hvor den blir orientert horisontalt. Den utleste verdien til grensen er gitt ved (21), der  $V_{maks}$  er den høyeste måleverdien til vindmåleren,  $V_{grense}$  er vindhastighetens-grensen og  $X_{maks}$  er den største mulige utleste verdien.

$$X_{vindgrense} = V_{grense} * \frac{X_{maks}}{V_{maks}} \quad (21)$$

Hver gang den målte verdien overstiger  $X_{vindgrense}$  skal riggen gå i vindmodus og en stoppeklokke med av-forsinkelse på 15 minutter skal aktiveres. Dette gjør at riggen vil være i vindmodusen helt til det har gått 15 minutter hvor vinden ikke har oversteget grenseverdien. Etter at stoppeklokken har blitt lav skal riggen fortsette som vanlig, altså hovedmodusen **kjør-solen** skal starte.

Forholdet mellom utlest verdi og vindhastigheten er gitt ved (22).

$$V = \frac{X * V_{maks}}{X_{maks}} \quad (22)$$

## Implementering av induksjonssensorene

Det defineres tre trinn på 18,4mA, 8,8mA og mindre enn 4mA, det vil tilsvare en avstand mellom sensoren og et ferromagnetisk objekt, på henholdsvis ca. 24mm, ca. 10mm, eller et ledningsbrudd. Hvis noen av sensorene har et utslag som underskrider 8,8mA betyr det at noe har gått galt og riggen er i ferd med å bli skadet ved å kjøre motorene for langt. Hvis målestrømmen til en sensor er mindre enn 4mA betyr det at noe er galt, antakelig ledningsbrudd. Derfor skal riggen fryses hvis en av sensorene har en målestrøm som er under 8,8mA eller under 4mA. Dette skal hindre at riggen blir skadet, eller at en fysisk sikring skal trå inn.

Siden induksjonssensorene har en målestrøm mer enn 20mA når de ikke har noe utslag, vil varsellampene til A/D-modulen lyse selv om alt fungerer som det skal. Det vil da ikke være mulig å se på varsellampene om det er ledningsbrudd. Det er mulig å programmere noen ander lamper til å lyse hvis en av de innleste verdiene for induksjonssensorene er mindre enn 4mA. PLS-en har noen innebygde lamper som kan brukes til indikatorer, ellers kan en lampe drives via en av digitalutgangene som er ledig.

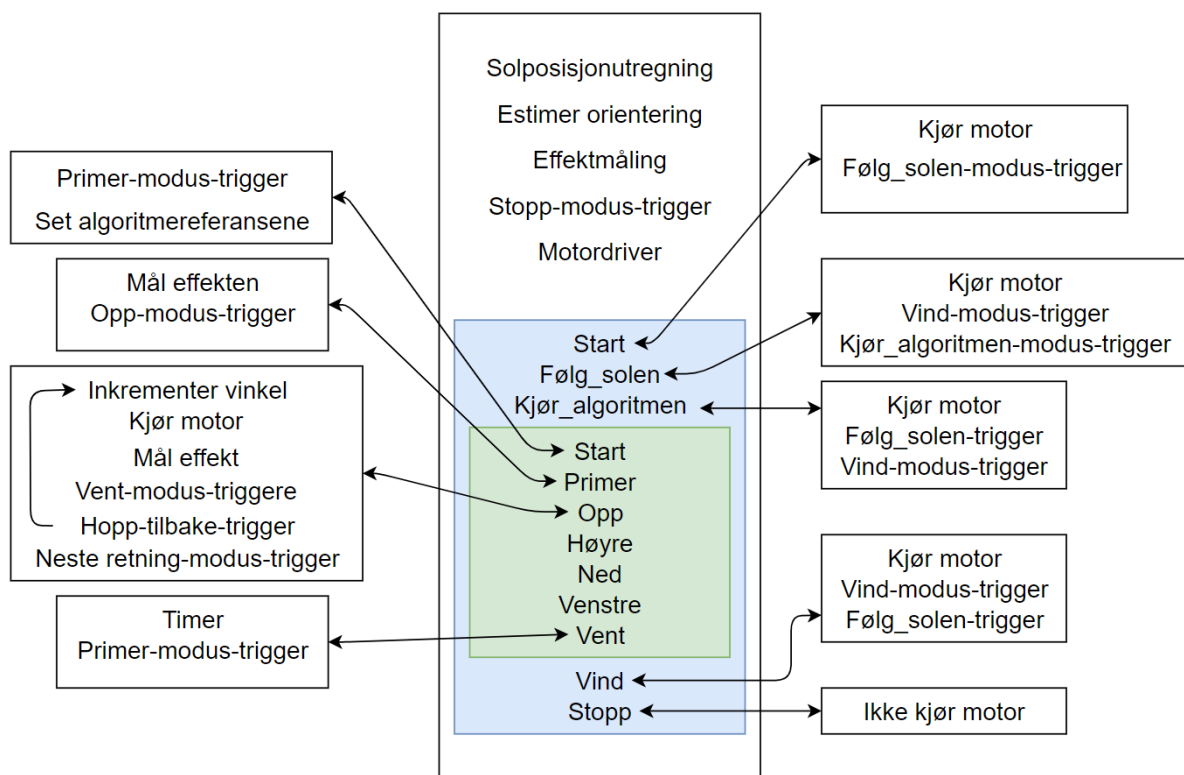
En sensor på elevasjonsaksen og en sensor på asimutalaksen skal benyttes til å nullstille posisjonen til programmet. En sensor skal plasseres slik at den gir et 18,4mA utslag når riggen er orientert i 15° på asimutalaksen, denne sensoren blir kalt induksjonssensor-1. En annen sensor skal plasseres slik at den har samme utslag når riggen er orientert i 0° på elevasjonsaksen, denne sensoren blir kalt induksjonssensor-2. Det er viktig at de monteres slik at de ikke kan bli utløst av noen andre deler mens riggen er i drift. Hvis det skjer, vil den enten nullstille posisjonssystemet til en av aksene eller stoppe riggen. Hvis posisjonen til en av aksene blir nullstilt når riggen ikke er i nullposisjon for den aksene vil det bli et stort avvik mellom posisjonssystemet og posisjonene til riggen.

## 5 Programmet

### 5.1 Koden

Hele koden ble skrevet i e!cockpit med strukturert tekst, selv om noen av delene kunne vært i stige eller blokk diagram.

PLS-en luper gjennom koden hele tiden. Figur 12 viser overordnet struktur over koden. Den grønne firkanten illustrer case-setningen til **kjør\_algoritmen**, der de forskjellige modiene er listet inni. De hvite firkantene som er tilkoblet de forskjellige modiene med piler inneholder alle komponentene til de modiene. Alle modiene har minst en oppgave de skal utføre og en eller flere trigger som skal sette systemet i en annen modus. Den blå firkanten illustrere case-setningen til hovedmodiene, der de forskjellige modiene er listet inne. Den store hvite firkanten i midten er hele programmet, der alle komponentene i hoveddelen listetes opp inni. Hver gang programmet luper blir komponentene i hoveddelen utført. Dette gjør at de har en global virkning sett i fra case-setningene.



Figur 12 overordnet struktur over koden

Solposisjonen må være tilgjengelig når riggen er i **følg\_solen**, **kjør\_algoritme** og **vind-**modiene. Derfor skal solposisjon utregningene være i hoveddelen av koden. Utregningene er i henhold til kapittel solens posisjon [1].

Posisjonsestimatorene må være utfor case-setningene slik at den hele tiden estimerer posisjonen.

Effekt måleren er utfor case-setningene slik at all under-modiene kan bruke den samme koden til å måle og filtrere effekten. Den filtrer ut små forstyrrelser ved å måle effekten i fem sekunder og velge den største effekten. Effekt måleren får beskjed om å måle effekten av algoritmen, den sier ifra når den er ferdig med å måle effekten. Måleren må få beskjed om å bli nullstilt av algoritmen når algoritmen er ferdig med måleverdien.

**Stopp**-modus-triggeren skal utløses hvis en av induksjonssensorene har en målestrøm som er mindre enn 8,8mA. Dette er den eneste triggeren som er i den utfor case-setningene, fordi alle modiene skal ha en vei til **stopp**-modusen. Det er viktig at ingen andre triggerer er utfor case-setningene fordi de vil kunne ta riggen ut av **stopp**-modusen.

Motordriveren består bare av boolsk logikk, derfor kan denne delen enkelt være i stige eller blokk diagram. Motordriverens oppgave er å gjøre slik at motorene kan drives med bare fire boolske variabler, så skal motordriveren ta seg av resten av logikken. Det vil innebære å starte og stoppe motorene, og stille integreringsretningen eller telleretningen hvis en opp/ned teller blir benyttet.

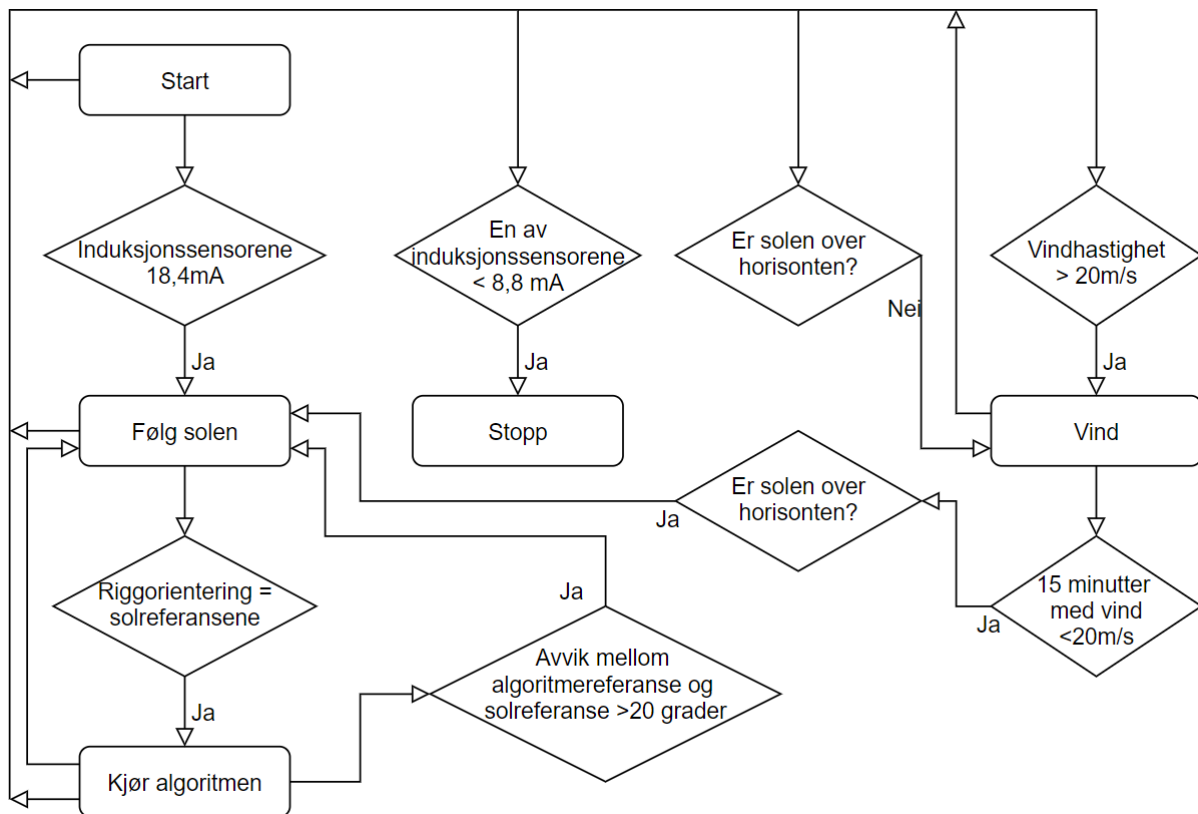
## 5.2 Hovedmodiene

Riggen har de fem hovedmodiene **start**, **følg\_solen**, **kjør\_algoritmen**, **vind** og **stopp**. Figur 13 viser et flyttdiagram for hovedmodiene. I **start**-modusen kjører begge motorene i revers. Når induksjonssensor-1 har et utslag på 18,4mA skal asimutalmotoren stoppe og den estimerte posisjonen for asimutalaksen skal stilles til 15°. Tilsvarende, når induksjonssensor-2 har et utslag på 18,4mA skal motoren på elevasjonsmotoren stoppe. Når begge sensorene har et utslag på 18,4mA skal riggen gå i **følg\_solen**-modusen.

**følg-solen**-modusen kjører riggen til solreferansene, deretter vil riggen gå i **kjør\_algoritmen**-modusen.

Riggen vil forbli i **kjør\_algoritmen**-modusen til at vinden overstiger en gitt verdi, en av induksjonssensorene har en målestrøm som er mindre enn 8,8mA, en tilnærmet avstand mellom solreferansene og algoritmereferansens overstiger en gitt verdi, eller at solen går under horisonten. Når solen passerer den øver asimutal grensen vil solreferansen stilles til 15°, dette vil gjøre at den tilnærmede avstanden mellom solreferansens og algoritmereferansens overstiger den gitte verdien. Dette gjør at riggen vil kjøre til 15° i asimutalaksen på natten. Riggen vil ikke veksle mellom **følg\_solen** og **kjør\_algoritmen** fordi algoritmen bør være i stand til å følge solen med en viss margin.

Hvis vindhastigheten overstiger en gitt verdi, vil riggen gå i **vind**-modusen, så lenge den ikke allerede er i **stopp**-modusen. Riggen vil forbli i **vind**-modusen helt til en av-forsinket stoppeklokke på 15 minutter har utløpt og at solen er over horisonten. Deretter vil den gå i **følg\_solen**-modusen. Hver gang vindhastigheten overstiger den gitte verdien vil stoppeklokken bli tilbakestilt.



Figur 13 Flytdiagram for hovedmodiene

## Følg-Solen-modusen

RTC modulen oppgir hvor mange dager har passert siden 1.1.1970 med en oppløsning på en dag oppgitt i sekunder. En funksjon i PLS-en regner det om til hvor mange dager har passert i innværende år, heretter kalt dagsnummeret. Klokkeslettet blir oppgitt av RTC-modulen og regnes om til timer, slik at 11:30 vil bli om regnet til 11,5.

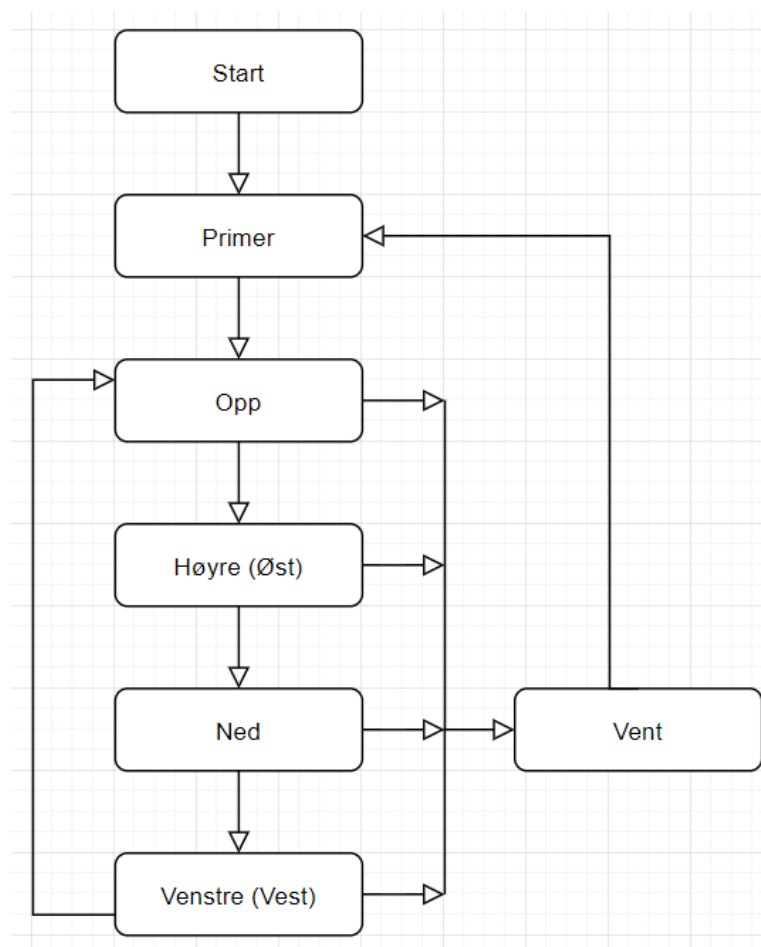
Dagsnummeret, klokkeslettet, tidssone, lengde og breddegrad blir brukt til å regne ut solposisjonen oppgitt i en asimutalvinkel og en elevasjonsvinkel. Asimutalvinkelen er definert med  $0^\circ$  mot den geografiske Nordpolen og positiv retning mot øst. Elevasjonsvinkelen er definert med  $0^\circ$  mot horisonten og positiv retning opp. Refererer til kapittel solens posisjon [1] for hvordan solvinkelen blir kalkulert.

Det er to solreferansevinkler, en for asimutal og en for elevasjonsvinkel. Panelene skal ikke peke mot solen når den er under horisonten, for å ikke skade seg eller utløse noen sikkerhetstiltak. Derfor overskrives solreferansevinkelen for elevasjonsaksen til  $0^\circ$  når den kalkulerede elevasjonsvinkelen til solen er under  $0^\circ$ , ellers vil solreferansevinkelen for elevasjonsaksen være lik den kalkulerede elevasjonsvinkelen til solen. Det skal være et mekanisk sikkerhetssystem som skal hindre riggen fra å skade seg selv ved å rotere for langt. Denne bør helst ikke utløses. Derfor skal riggens asimutalvinkel ikke overstige  $345^\circ$ . Hvis den kalkulerede asimutalvinkelen til solen overstiger  $345^\circ$  vil solreferansevinkelen til asimutalaksen overskrives til  $15^\circ$ , ellers vil solreferansevinkelen for asimutalaksen være lik den kalkulerede asimutalvinkelen til solen.

## Kjør-algoritmen-modusen

Algoritmen har de syv modiene; **start**, **primer**, **opp**, **høyre**, **ned**, **venstre** og **vent**. **Start**-modusen setter algoritmereferansene lik solreferansens, så går den i **primer**-modusen som bare måler effekten før den går i **opp**-modusen. **Opp**-modusen vil øke elevasjonsvinkelen til riggen med en grad om gangen til effekten slutter å øke, deretter vil den stille riggen en grad tilbake og gå i **høyre**-modusen. **Høyre**, **ned** og **venstre**-modiene fungerer på tilsvarende måte. Modiene vil sirkulere som følger; **opp**, **høyre**, **ned**, **venstre** og tilbake til **opp**-modusen, og så videre.

Når riggen er orientert i et maksimumspunkt kan den ikke øke produksjonene med å endre vinklene til panelene. Dermed, hvis riggen går igjennom fire av modiene som tilsvarer en retning på rad, uten å være i stand til å øke produksjonen, skal riggen gå i **vente**-modusen. **Vente**-modusens formål er å holde riggen i maksimumspunktet lengst mulig. Når riggen søker etter en bedre vinkel vil den midlertidig minke produksjonen når riggen tester en dårligere vinkel. Dette er unødvendig hvis den allerede er i et maksimumspunkt. Etter at en stoppeklokke på 10 minutter har utløpt skal den gå til **primer**-modusen. Tidsintervallet på 10 minutter er åpent for endring. Figur 14 viser hvordan modiene kan endre seg som forklart ovenfor, men figuren inneholder ikke betingelsene for de endringene.



Figur 14 Flyttdiagram for hvordan modiene til algoritmen kan endre seg.

Som et eksempel viser Figur 15 flytskjema for algoritmefasen opp som forklart under, hvor «Alg\_alp» er algoritmereferansen for elevasjonsaksen, «++» betyr inkrementer med én, «--» betyr inkrementer med minus én.

Algoritmefasene som tilsvarer en retning altså, **opp**, **høyre**, **ned** og **venstre** har underfasene **øk\_vinkelen**, **kjør\_motorene**, **mål\_effekten** og **evaluer**. Fasen **øk\_vinkelen** vil inkrementere algoritmereferanse-vinkelen i den retningen som algoritmen skal sjekke med en grad, så lenge at det ikke vil gjøre at algoritmereferansens oversiger grenseverdiene.

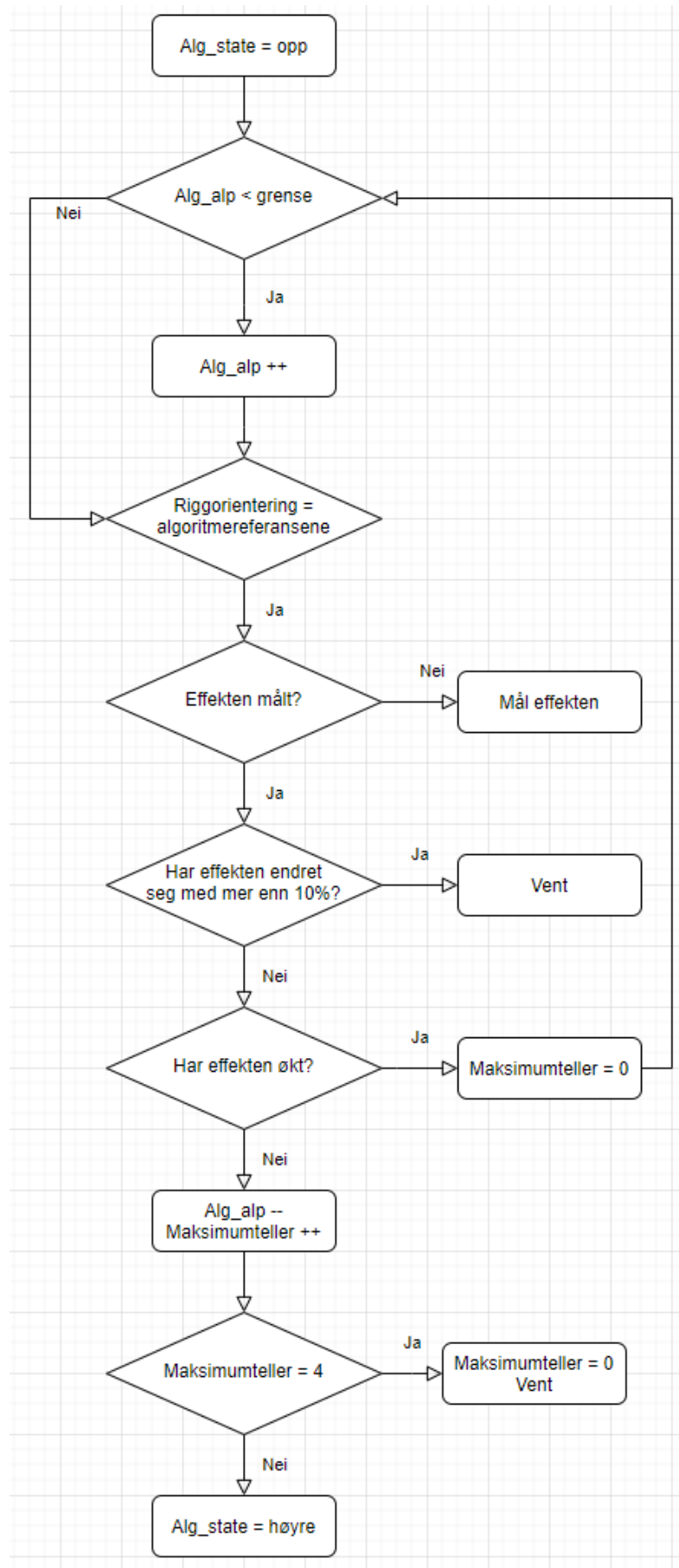
**kjør\_motorene**-fasen vil vente til riggen har orientert seg til algoritmereferansene, så vil den gå til **mål\_effekten**-fasen.

**Mål\_effekten**-fasen vil fortelle en annen del av programmet til å måle effekten, og vente på resultatet. Deretter vil den gå til en **evaluer**-fasen.

**Evaluer**-fasen vil sammenligne den nylig målte effekten med den forrige målte effekten. Hvis effekten økte eller minket med mer enn 10% kan det komme av en eller annen forstyrrelse, derfor skal den gå i en ventemodus. Faktoren på 10% er åpen for endring, og om den burde algoritmemodusen til **primer**, i stedet for å gå i **vente**-modus.

Hvis effekten har økt med mindre enn 10% skal en maksimumteller nullstilles, verdien for den forrige effekten bli erstattet med verdien for den nye effekten, og den går tilbake til fasen **øk\_vinkelen**.

Hvis effekten ikke økte skal den inkrementere vinkelen tilbake, og en maksimumteller skal inkrementeres opp med én. Hvis maksimumtelleren er fire, betyr det at algoritmen har kjørt gjennom alle fasene som tilsvarer en retning uten å være i stand til å øke produksjonen. Det vil si at riggen har nådd en maksimumteller. Derfor skal den gå i en ventemodus hvis maksimumtelleren blir fire. Hvis maksimumteller ikke er fire skal den fortsette til neste algoritmefase som tilsvarer en retning.



Figur 15 Flytskjema for algoritmemodus opp.



## 6 Testresultat

Lokalsoltid til solutregningene ble manuelt overskrevet til en gitt verdi, for at solreferansevinklene skulle være nærmere startpunktene for å gjøre testene raskere. Ventemodusen var redusert til 10 sekunder. Siden solcelleanlegget ikke ble montert og strøm og spenningsmålingene ikke ble målt, ble en justerbar DC kilde brukt til å emulere effekten fra solcelleanlegg, ved å koble kilden til en 0-10V-analog-til-digitalingang.

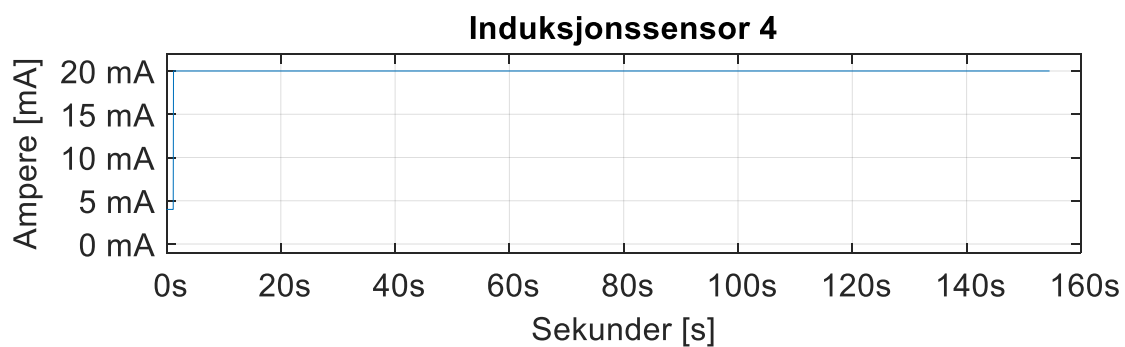
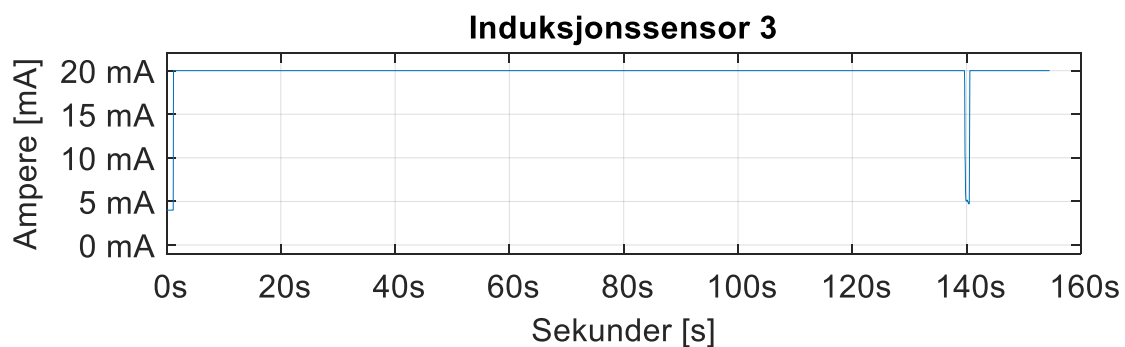
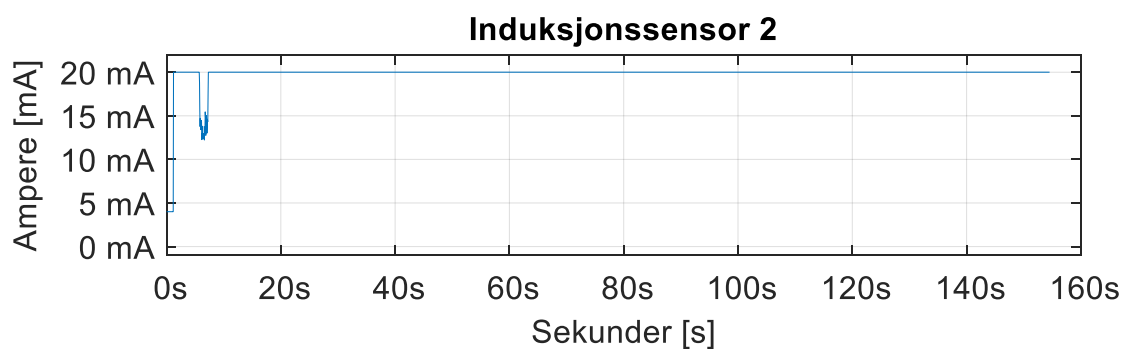
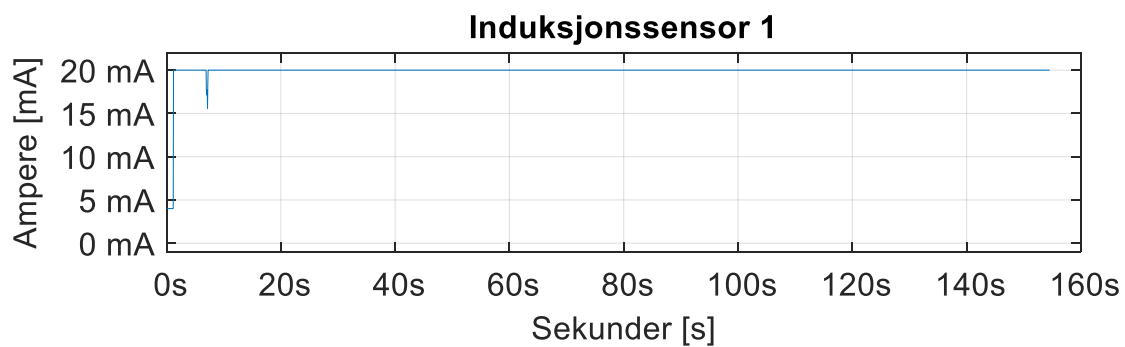
### 6.1 Test 1

Figur 16 viser målestrømmen til induksjonssensorene i løpet av test 1. Der kan man se at sensor-1 og sensor-2 hadde en målestrøm som var mindre enn 18,8mA og mer enn 8,8mA på samme tid i begynnelsen av testen. Sensor-3 hadde en målestrøm som var mindre enn 8,8mA på slutten av testen.

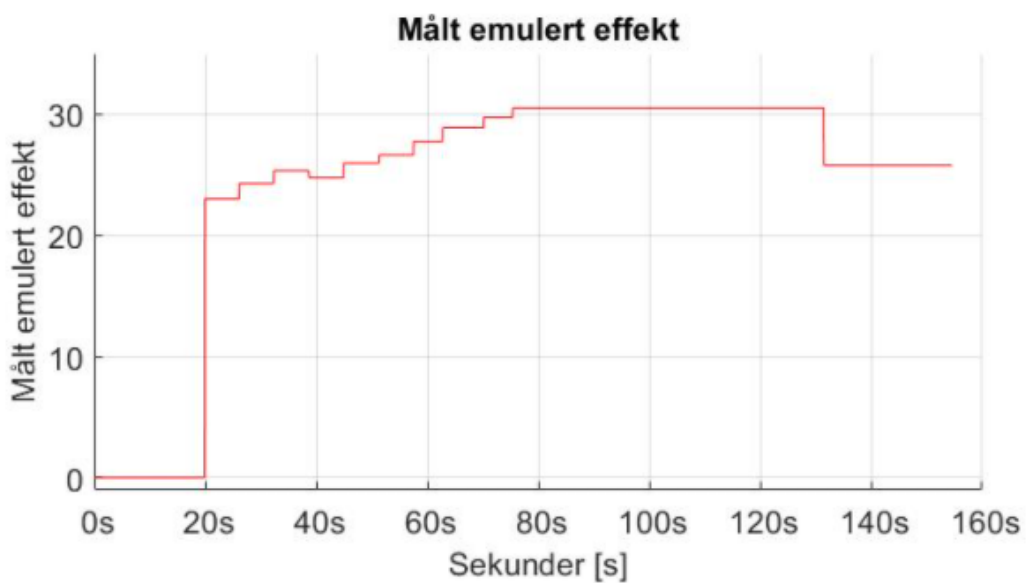
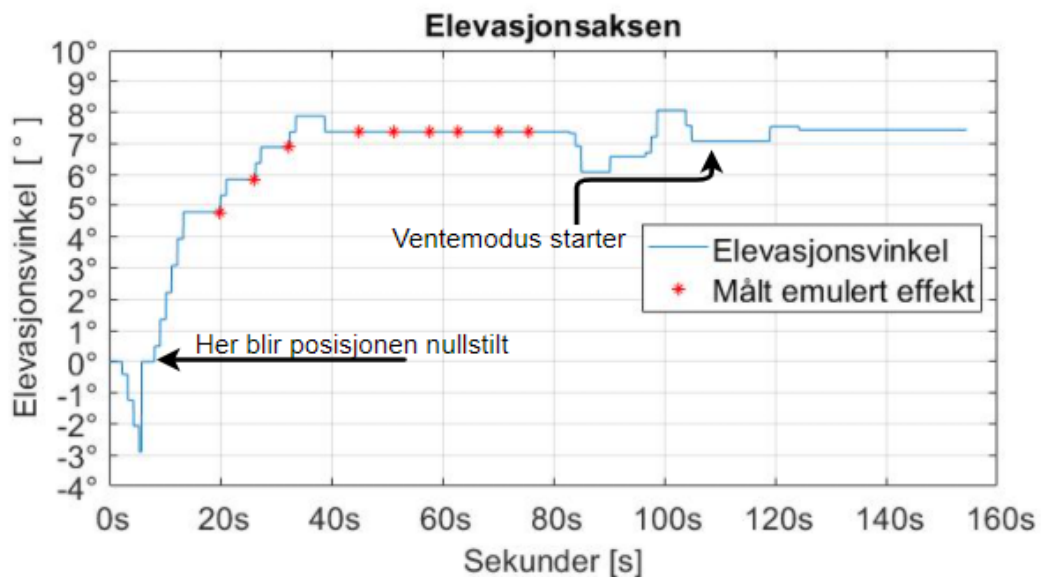
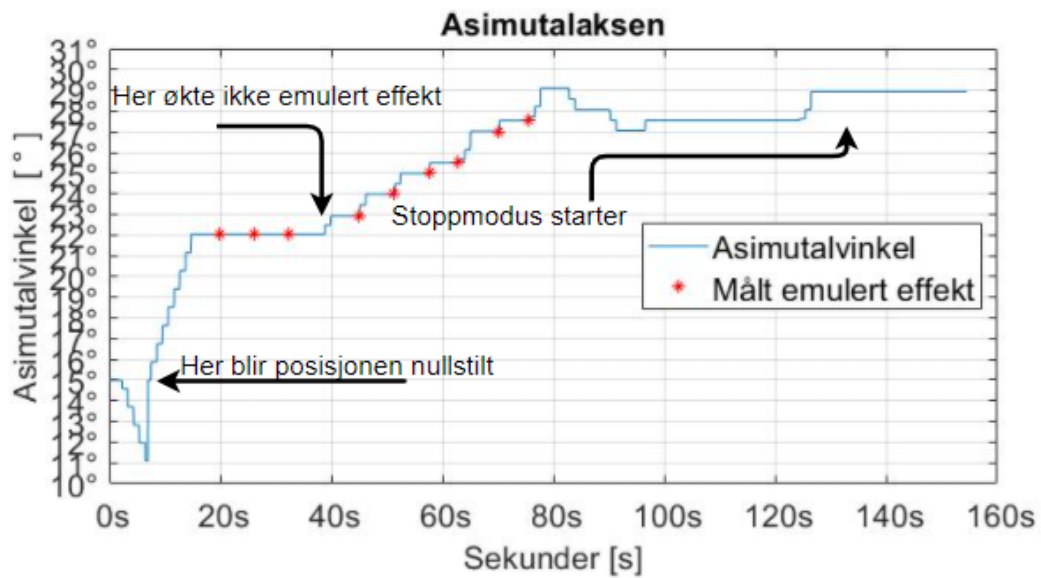
Figur 17 viser den estimerte posisjonen til riggen under test 1 og den målte emulerte effekten. Der ser man at begge motorene begynte å gå i revers. De induksjonssensor-1 og induksjonssensor-2 hadde utslagene tidlig i testen ble posisjonssystemet nullstilt, og motorene begynner å gå fremover. Litt før 20 sekunder i testen hadde riggen nådd solreferansene og riggen gikk i **kjør\_algoritmen**-modusen.

De røde stjernene markerer når den målte emulerte effekten økte, effekten ble testet ca. hvert 5. sekund. Det vil si at når det er en rød stjerne skal algoritmen sjekke litt lengere i den retningen som ble testet.

Figur 17 viser at den emulerte effekten økte tre ganger på rad når algoritmen testet positiv elevasjonsvinkel, for perioden 20-40 sekunder. Deretter økte effekten ikke, og algoritmen testet så positiv asimutalvinkel. Den emulerte effekten økte seks ganger på rad når den testet positiv asimutalvinkel. Så sluttet effekten å øke, og algoritmen testet i alle retningene til den hadde testet fire retninger uten at effekten hadde økt, da gikk den i ventemodus. Ventemodusen startet litt etter 100 sekunder og var ferdig litt før 120 sekunder. Etter 130 sekunder stoppet begge motorene. Figur 16 viser at induksjonssensor-3 hadde et utslag som satte riggen i hovedmodusen **stopp**.

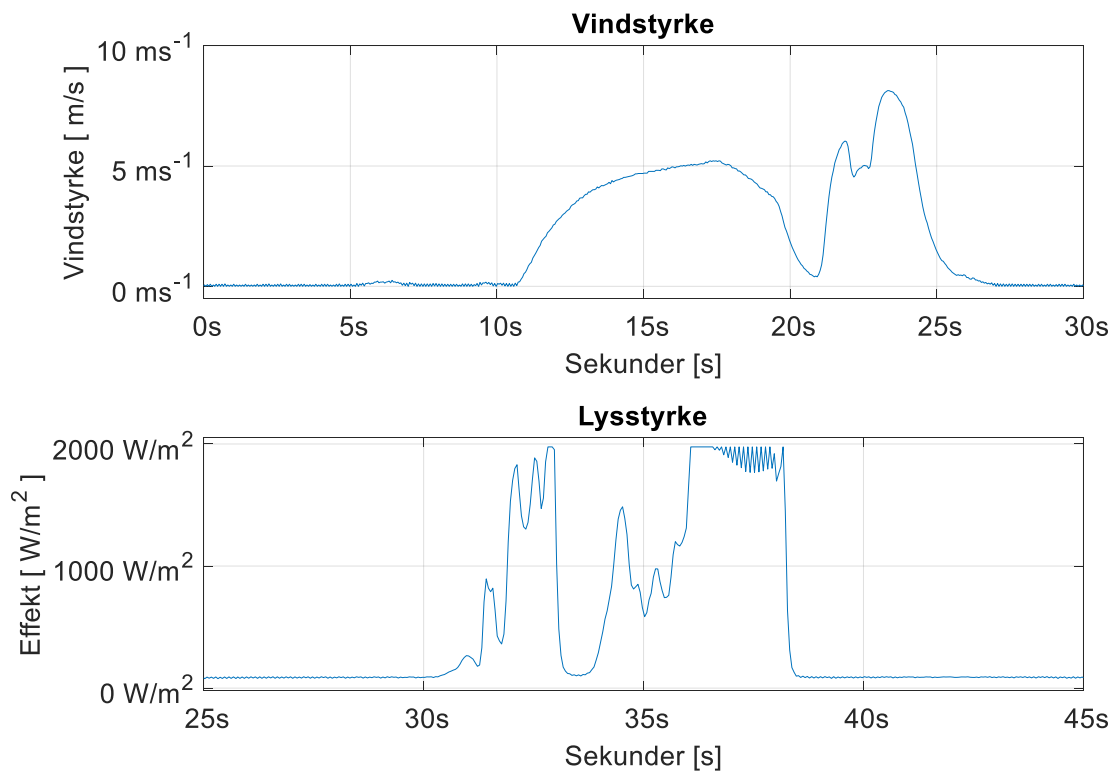


Figur 16 Induksjonssensor målestrøm test 1

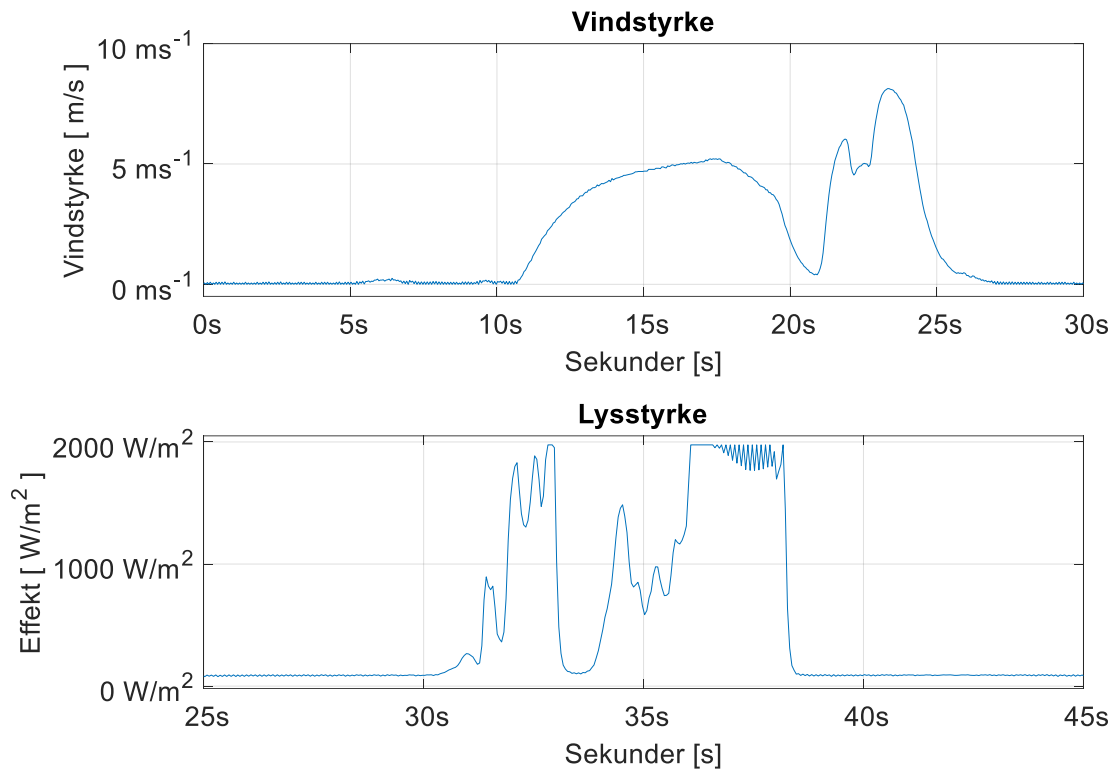


Figur 17 Posisjon til riggen test 1

## 6.2 Test 2



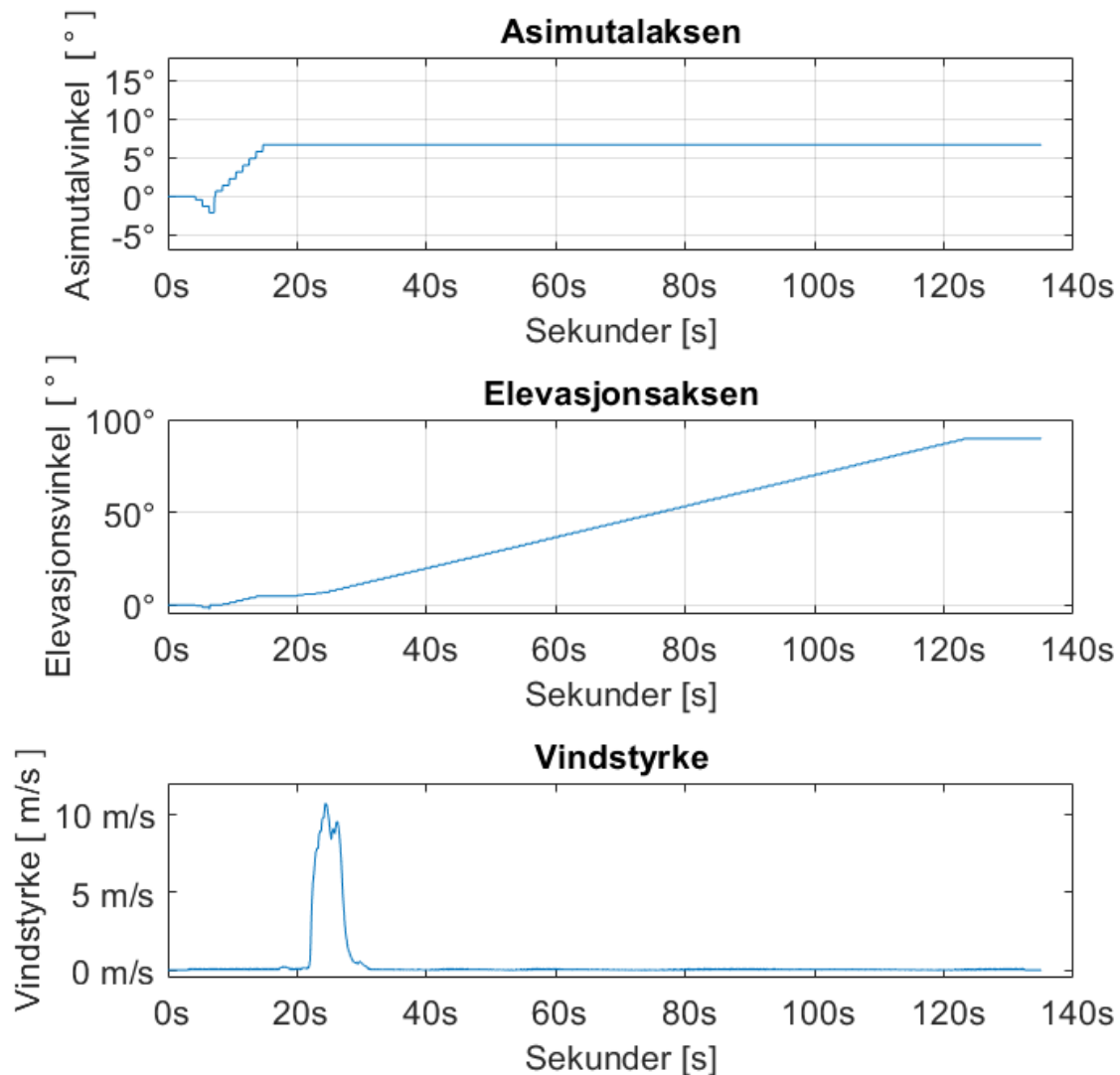
Figur 18 viser lysstyrke og vindstyrke. Pyranometret ble eksitert med en lommelykt som ble lyst direkte på sensoren på kort avstand. Vindmåleren ble eksitert først med en hårføner, så med en hand. Når hårføneren ble bruke ble det en jevnere og lavere topp, enn når en hand ble brukt.



Figur 18 Lys og vindstyrke test 2

### 6.3 Test 3

Test 3 skulle vise at vindmodusen fungerer. Vindgrensen ble stilt fra 20m/s ned til 7,3m/s for denne testen. Figur 19 viser posisjonene til riggen under test 2. Der kan man igjen se at motorene startet med å gå i revers til posisjonen ble nullstilt, så begynte de å gå framover til de nådde solreferansene. 20 sekunder i testen var algoritmen i ferd med å teste positiv elevasjonsvinkel, men vindsensoren utløste vindmodusen og elevasjonsmotoren kjørt til 90° og stoppet der. Asimutalmotoren ble heller ikke kjørt etter at vindmodusen var utløst. Figur 19 viser den estimerte posisjonen til riggen og den målte vindstyrken.



Figur 19 Posisjon til riggen og vindstyrken test 3

## **6.4 Test 4**

Riggen ble programmert til å kjøre elevasjonsvinkelen  $1080^\circ$  kontinuerlig, for å sjekke at integratoren til posisjonssystemet fungerte. Den roterende delen på motoren ble merket, slik at det var lett å se når motoren hadde tatt et omløp. Denne metoden målte et avvik som er mindre enn 0,1%.

## **6.5 Test 5**

Denne testen skulle teste avviket til posisjonssystemet under intermitterende drift. Siden endringer i hastigheten vil føre til unøyaktigheter til integratoren. Riggen ble programmert til å kjøre elevasjonsmotoren et sekund på og et sekund av, i  $360^\circ$ . Den roterende delen på motoren var fremdeles merket. Denne metoden målte et avvik på 1,5-2%. Dette kommer av at PLS-en ikke klarer å integrere med stor nok oppløsning.

## 7 Videre arbeid

RTC-modulen trenger en antenne for å lese sanntidsklokke fra satellittene.

Det anbefales at Wago 750-404/000-003 frekvenstillerne byttes med Wago 750-404/000-000 opp/ned tellere. Da vil det ikke bli noe avvik mellom posisjonssystemet og orienteringen til riggen. Det er viktig at posisjonssystemet fortsatt er aktivt etter at motoren ikke blir drevet fra PLS-en, fordi treghetsmomentet til motoren vil drive motoren et lite stykke lengere.

Signalkabelen til vindsensoren bør skjermes når anlegget blir montert.

Strøm- og spenningsmålingene til solcelleanlegget må benyttes. Det er viktig at det ikke blir «overflow» når man kalkulerer effekten.

test-og-sjekk algoritmen kan testes ved at den inkrementere med for eksempel  $5^\circ$  eller  $10^\circ$  istedenfor  $1^\circ$ . Når den har nådd et maksimum kan algoritmen starte på nytt der den inkrementere med  $1^\circ$ , og når den har nådd et maksimum igjen kan den gå i en ventemodus.

Det er mulighet for å loggføre verdier med PLS-en. Da kan man loggføre data og sammenligne resultatene til forskjellige algoritmer, som nevnt i avsnittet over, eller ingen algoritme. Det kan være interessant å vite strøm, spenning, og avgitteffekt til solcelleanlegget, da kan man evaluere om algoritmen har noen vinning. Videre kan det være interessant å vite vindstyrke, lysstyrken og hvor stor forskjell det er mellom algoritmereferansene og utregnet solposisjon. Loggen kan bli skrevet til hver gang testeffekten blir oppdater, da vil den kun loggføre når algoritmen kjøres, da vil den ikke loggføre noe mens motoren kjører og den vil kun logge den nøyaktige testeffekt.

Tiden til ventemodusen kan kanskje optimaliseres.

LEM-elementer skal benyttes for å skalere strøm til spenning og skalere spenning til spenning som skal kunne kobles direkte på 0-10V DC analog-til-digitalmodulene.



## Referanseliste

[1] The Sun's Position. Hentet 24.02.2021. [Internett]. Tilgjengelig:  
<https://www.pveducation.org/pvcdrom/properties-of-sunlight/the-suns-position>

[2] ML-01 Manual. Hentet 14.05.2021. [Internett] Tilgjengelig:  
<https://eko-eu.com/products/solar-energy/si-pyranometers/ml-01-si-pyranometer>

# Vedlegg

## Koden

```
1 PROGRAM PLC_PRG
2 VAR
3     State      : states      := start;
4     Alg_state  : Alg_states   := Start_alg;
5     Alg_sub_states : Alg_sub_states := a ;
6     RTC        : FbRtcBaseFunctions() ;
7     Wind_timer : TOF() ;
8     Alg_timer  : TON() ;
9     Power_meas : TON() ;
10
11
12     DvDiag : dword;
13     dttt : DT;
14     datee : DATE;
15     timeee : TOD ;
16     Time_stamp_int, prev_time_stamp_int :TOD;
17     prev_timel, prev_time2 : TOD;
18     prev_HZ1, current_HZ1 : LREAL;
19     prev_HZ2, current_HZ2 : LREAL;
20     Delta_Motor1_pos, Delta_Motor2_pos : LREAL;
21
22     MIN_per_rev : REAL;
23
24     M1F, M1R, M2F, M2R      : BOOL := FALSE;
25     C1F, C2F                : BOOL := FALSE;
26     Induction_sensord_limit : REAL := 9832; // = 8.8mA
27     Induction_sensord_initial : REAL := 29488 ; // = 18.4mA
28     Induction_sensord_error : REAL := 3; // > 4mA
29     Wind_speed_limit        : WORD := 4000; // endret for vindtest;
30     Wind_pos_alp            : INT := 90;
31     Motor1_pos, Motor2_pos, sol_reff1, sol_reff2 : LREAL;
32     LSTM, EoT, BBB, TC, LST, HRA, Delta, Alphaa, Azimuthh, LT1 : LREAL;
33     pi : LREAL := 3.14159265359;
34     Lengdegrad : LREAL := 17.42;
35     Breddegrad, Phi : LREAL := 68.43;
36     day_temp : LWORD ;
37     day, Alpha, Azimuth : INT ;
38     Azimuth_limit_upper : INT := 345;
39     Azimuth_limit_lower : INT := 15 ;
40     Alpha_limit_upper : INT := 90 ;
41     Alpha_limit_lower : INT := 0 ;
42     Avstand_limit : INT := 20 ;
43     Alg_azi, Alg_alp : INT ;
44     Alg_azi_temp, Alg_alp_temp : INT ;
45     d_temp : LWORD ;
46     Alg_jump_back : BOOL := FALSE;
47     Alg_jump_back_false_count : INT := 0 ;
48     avstand : REAL ;
49     Power_measured, Power_measure : BOOL := FALSE;
50     power, pre_power : UDINT ;
51     Power_previous, Power_test : UDINT ;
52     Voltage, Amps : UINT ;
53     prev_tod : DWORD ;
54     reset_power : BOOL ;
55     Current_DT :DT ;
56     Pyranometer :WORD ;
57     cur : WORD;
58     Truee : BOOL := TRUE;
59 END_VAR
```

Figur 20 innføring av variabler og funksjonsblokker.

```

28 // Solkalkulasjoner
29 day := FudaysSinceJan01 (RTC.dLocalDate); // Her hentes dagsnummeret
30 LTI := TOD_TO_REAL(RTC.todLocalTime)/(1000*60*60); // Her hentes klokken og gjør den om til "timeklokke"
31 LSTM := 15*2;
32 BBB := 360*(day-81)/365;
33 Eot := 9.87*sin_1(2*BBB*pi/180)-7.53*cos_1(BBB*pi/180)-1.5*sin_1(BBB*pi/180);
34 TC := 4*(Lengdegrad - LSTM) + Eot;
35 LST := LTI + TC/60;
36 HRA := 15*(LST-12);
37 Delta := 23.45*sin_1(360*(day-81)*pi/(365*180));
38 Alpha := arcsin( sin_1(Delta*pi/180) * sin_1(Phi*pi/180) + cos_1(Delta*pi/180) * cos_1(HRA*pi/180) ) * (180/pi);
39 Azimuth := arccos(((sin_1(Delta*pi/180)*cos_1(Phi*pi/180)) - (cos_1(Delta*pi/180)*sin_1(Phi*pi/180)*cos_1(HRA*pi/180)))/cos_1(Alpha*pi/180))*(180/pi);
40 IF LST >= 12 THEN
41 Azimuth := 360 - Azimuth; // Hvis lokal soltid er større enn 12 er azimuthvinkelen 360- uttrykket over.
42 // Ellers er det likt uttrykket over.
43 END_IF
44 IF LREAL_TO_INT(Alpha) < Alpha_limit_lower THEN // Her overskrives referanseverdiene hvis de er utfor de tillatte verdiene.
45 Alpha := Alpha_limit_lower; // For at panelene skal ikke peke mot solen når den er under horisonten.
46 ELSE
47 Alpha := LREAL_TO_INT(Alpha);
48 END_IF
49 IF LREAL_TO_INT(Azimuth) > Azimuth_limit_upper THEN // Dette vil gjøre sånn at azimuthvinkelen til riggeren går til 15 når solen passerer 345 grader.
50 Azimuth := Azimuth_limit_lower; // Ved å trigge maksimum tillat avvik mellom run_alg og run_sol)
51 ELSE
52 Azimuth := LREAL_TO_INT(Azimuth);
53 END_IF
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Figur 21 kodekomponenten for solposisjon-utregningene

## Estimering av orienteringen

Hvis det hadde blitt benyttet en opp/ned teller istedenfor en frekvensteller hadde koden vært som vist på Figur 22.

```
102 // Slik hadde det vært hvis en 750-404/000-000 opp-ned tellermodulen hadde vært benyttet.
103 // Resten av logikken hadde vært i driveren til motorene.
104 IF IoConfig_Globals_Mapping.Induc_sensor1 <= Induction_sensor_initial THEN
105     TELLERMODUL1 := Azimuth_limit_lower*(73*236*2)/360;
106 END_IF
107 // Når induksjonssensor1 har et utslag på 18,4mA stilles tellermodulen til 15*(73*236*2)/360 asimutalaksen.
108 IF IoConfig_Globals_Mapping.Induc_sensor2 <= Induction_sensor_initial THEN
109     TELLERMODUL2 := Alpha_limit_lower*(73*236*2)/360;
110 END_IF
111
112 // Når induksjonssensor2 har et utslag på 18,4mA stilles posisjonsystemet til 0 grade for elevasjonsaksen.
113 Motor1_pos := DWORD_TO_INT(IoConfig_Globals_Mapping.Current_count1*360/(73*236*2));
114 Motor2_pos := DWORD_TO_INT(IoConfig_Globals_Mapping.Current_count2*360/(73*236*2));
115
```

*Figur 22 kodekomponent for estimering av orienteringen – opp/ned teller alternativ.*

På grunn av at en frekvensteller måtte anvendes ble koden slik som vist på Figur 23.

```

56 //Posisjonsovervåking
57 IF IoConfig_Globals.Mapping.Induc_sensor1 <= Induction_sensord_initial THEN
58   Motor1_pos := Azimuth_limit_lower; // Når induksjonsensoren har et utslag på 18,4mA stilles posisjonsystemet til 15 grade for asimuthalaksen.
59 END_IF
60 IF IoConfig_Globals.Mapping.Induc_sensor2 <= Induction_sensord_initial THEN
61   Motor2_pos := Alpha_limit_lower; // Når induksjonsensoren har et utslag på 18,4mA stilles posisjonsystemet til 0 grade for asimuthalaksen.
62 END_IF
63 Time_stamp_int := RTC.todLocalTime; // Her hentes tidsstempel til integratoren
64 current_HZ1 := TO_LREAL(IoConfig_Globals.Mapping.Counter_HZ1) * Fullmedifference(Time_stamp_int,prev_time_stamp_int) * 360 / (73*236*1000*2);
65 current_HZ2 := TO_LREAL(IoConfig_Globals.Mapping.Counter_HZ2) * Fullmedifference(Time_stamp_int,prev_time_stamp_int) * 360 / (73*236*1000*2);
66 prev_time_stamp_int := Time_stamp_int; // Her lagres tidsstempel for neste syklus
67 IF IoConfig_Globals.Mapping.Counter_HZ1 = 4294967295 THEN // Linje 58 og 59; Herter utlest frekvens og regner det til grader/sekund for riggen
68   current_HZ1 := 0; // og integrere det over tidsforskjellen mellom tidsstemplene.
69 END_IF // girforholdet er 73*236, utlest frekvens er oppgitt i 0.001Hz, hallensoren
70 IF IoConfig_Globals.Mapping.Counter_HZ2 = 4294967295 THEN // gir 2 pulser per omdreining. Det må ganges med 360 for å gjør det om til grader.
71   current_HZ2 := 0; // Utlest frekvens oppgis som 4294967295 hvis den er Hz0.
72 END_IF
73
74
75
76
77 IF C1F = TRUE THEN // C1F bestemmer hvilken vei integratoren skal oppbevare.
78   Delta_Motor1_pos := prev_HZ1 + (current_HZ1 - prev_HZ1)/2; // implementering av trapsemetoden
79   Motor1_pos := Motor1_pos + Delta_Motor1_pos; // Her blir posisjonen til riggen sumert med
80   ELSEIF C1F = FALSE THEN // endringen til posisjonen til riggen.
81     Delta_Motor1_pos := prev_HZ1 + (current_HZ1 - prev_HZ1)/2; // her integrere den i negativ retning hvis C1F er false.
82     Motor1_pos := Motor1_pos - Delta_Motor1_pos;
83   END_IF
84   prev_HZ1 := current_HZ1; // Her lagres den gamle integreringsverdien til trapsemetoden for neste syklus.
85   // Den må stå på utsiden slik at den blir null når maskinen står stille.
86
87   IF C2F = TRUE THEN // Tilsvarende som for motor 1
88     Delta_Motor2_pos := prev_HZ2 + (current_HZ2 - prev_HZ2)/2;
89     Motor2_pos := Motor2_pos + Delta_Motor2_pos;
90   ELSEIF C2F = FALSE THEN
91     Delta_Motor2_pos := prev_HZ2 + (current_HZ2 - prev_HZ2)/2;
92     Motor2_pos := Motor2_pos - Delta_Motor2_pos;
93   END_IF
94   prev_HZ2 := current_HZ2;

```

Figur 23 kodekomponent for estimering av orientering.

```

104 //Effektmåling      105: Når strøm OG spenning blir målt
105 // power := TO_UDINT(IoConfig_Globals.Mapping.V_current_sensor) * TO_UDINT(IoConfig_Globals.Mapping.Volt_sensor);
106 power := TO_UDINT(IoConfig_Globals.Mapping.V_current_sensor); // Hvis effekten regnes ut etter 105 må de gjøres om til
107 // UDINT eller tilsvarende for å unngå OVERFLOW.
108 // 106: Bruker spenningen til A/D omformer til å emulere effekten.
109 IF reset_power = TRUE THEN // Denne delen brukes til å resette effektmåleren.
110   Power_measure := FALSE;
111   Power_measured := FALSE;
112   pre_power := 0;
113   Power_meas(IN := FALSE);
114   reset_power := FALSE;
115 END_IF
116
117 IF Power_measure = TRUE THEN
118   Power_meas(IN := TRUE, PT := TO_TIME(5*1000)); // Her får effekt måleren beskjed om å starte å måle effekten.
119   // Timer for hvor lenge effekten skal bli målt.
120   IF power > pre_power AND Power_meas.Q = FALSE THEN
121     pre_power := power; // Her velges den størst målte effekten.
122   ELSIF Power_meas.Q = TRUE THEN
123     Power_measured := TRUE; // Når timeren har utløp sier den ifra at effekten er målt
124     power_test := pre_power; // og den valgte verdien sendes ut.
125 END_IF
126 END_IF

```

## Effektmåling

Figur 24 viser hvordan kodekomponenten til effektmåleren ble kodet. Effektmåleren måler effekten når den får beskjed om det, og den sender resultatet til optimaliseringsalgoritmen slik at den kan gjøre valgene. Den må bli resettes etter at optimaliseringsalgoritmen er ferdig med måleresultatet.

Figur 24 kodekomponent for effektmåling.

## Stopp-modus-trigger

Figur 25 viser hvordan stopp-modus-triggeren ble kodet.

```
150 //Sikkerhetsnett
151 IF IoConfig_Globals_Mapping.Induc_sensor1 < Induction_sensord_limit OR
152   IoConfig_Globals_Mapping.Induc_sensor2 < Induction_sensord_limit OR
153   IoConfig_Globals_Mapping.Induc_sensor3 < Induction_sensord_limit OR
154   IoConfig_Globals_Mapping.Induc_sensor4 < Induction_sensord_limit THEN
155   state := stop;
156 // Hvis noen av sensorene har et utslag som er lavere enn 8,8mA skal riggen gå i stopp-modusen
157 ELSIF IoConfig_Globals_Mapping.Induc_sensor1 = Induction_sensord_error OR
158   IoConfig_Globals_Mapping.Induc_sensor2 = Induction_sensord_error OR
159   IoConfig_Globals_Mapping.Induc_sensor3 = Induction_sensord_error OR
160   IoConfig_Globals_Mapping.Induc_sensor4 = Induction_sensord_error THEN
161   state := stop;
162 END_IF
163 // Hvis noen av sensorene har et utslag som er lavere enn 4mA skal riggen gå i stopp-modusen
164 // linjene 144, 145,146, 147 og 148 er strengt tatt overflødig, men det kan legges
165 // til en varselslamp som sier at det er mindre en 4mA her.
166
```

Figur 25 kodekomponent for **stopp-modus-trigger**.

## Motordriver

Figur 26 viser hvordan motordrivere ble kodet. Integratoren skal ikke endre stopp eller retningen når riggen skal stopp, fordi den må fortsette å integrer på grunne av treghets momentet. Det samme gjelder hvis en opp/ned teller benyttes.

```

155 //Motordriv
156 // Hvis motor1 skal kjøres framover skal integratoren stilles til positiv retning.
157 // Integrator retningen skal bli hengende for at den må fortsette å integrere
158 // når motoren ikke får spenning lengere, på grunn av treghetsmomentet.
159 // Hvis en opp/ned teller brukes vil C1F bli erstatet med en digitalutgang som brukes
160 // til å drive en ingang på modulen som bestemmer telleretningen.
161 IF M1F = FALSE THEN
162     IoConfig_Globals_Mapping.Motor1Forward      := FALSE;
163     ELSIF M1F = TRUE THEN
164         IoConfig_Globals_Mapping.Motor1Forward  := TRUE;
165         C1F                                     := TRUE;
166     END_IF
167
168 IF M1R = FALSE THEN
169     IoConfig_Globals_Mapping.Motor1Revers       := FALSE;
170     ELSIF M1R = TRUE THEN
171         IoConfig_Globals_Mapping.Motor1Revers   := TRUE;
172         C1F                                     := FALSE;
173     END_IF
174
175 IF M2F = FALSE THEN
176     IoConfig_Globals_Mapping.Motor2Forward      := FALSE;
177     ELSIF M2F = TRUE THEN
178         IoConfig_Globals_Mapping.Motor2Forward  := TRUE;
179         C2F                                     := TRUE;
180     END_IF
181
182 IF M2R = FALSE THEN
183     IoConfig_Globals_Mapping.Motor2Revers       := FALSE;
184     ELSIF M2R = TRUE THEN
185         IoConfig_Globals_Mapping.Motor2Revers   := TRUE;
186         C2F                                     := FALSE;
187     END_IF
188

```

*Figur 26 kodekomponenten for motordriveren.*

Hovedmodiene

Figur 27 viser kodekomponenten til hovedmodusen **start**.



## Hovedmodusen **Start**

```
205 | //Steitan
206 | CASE state OF
207 |     start:      // Her kjøres motor1 i revers til induksjonsensor1 har utslag på 8,8mA
208 | IF IoConfig_Globals_Mapping.Induc_sensor1 <= Induction_sensord_initial THEN
209 |     M1R := FALSE;
210 |     ELSIF IoConfig_Globals_Mapping.Induc_sensor1 > Induction_sensord_initial THEN
211 |     M1R := TRUE;
212 | END_IF
213 |           // Her kjøres motor2 i revers til induksjonsensor2 har utslag på 8,8mA
214 | IF IoConfig_Globals_Mapping.Induc_sensor2 <= Induction_sensord_initial THEN
215 |     M2R := FALSE;
216 |     ELSIF IoConfig_Globals_Mapping.Induc_sensor2 > Induction_sensord_initial THEN
217 |     M2R := TRUE;
218 | END_IF
219 |           // Når begge sensorene har utslag på 8,8mA går den til "Følg_solen"
220 | IF IoConfig_Globals_Mapping.Induc_sensor1 <= Induction_sensord_initial AND
221 |     IoConfig_Globals_Mapping.Induc_sensor2 <= Induction_sensord_initial THEN
222 |     state := run_sol;
223 | END_IF
```

*Figur 27 kodekomponenten for hovedmodusen **start***

## Hovedmodusen **Følg\_solen**

Figur 28 viser kodekomponenten for kjøring av motorene til hovedmodusen **følg\_solen**. Figur 29 viser kodekomponenten til **kjør\_algoritmen**-modus-triggeren. Figur 30 viser kodekomponenten til **vind**-modus-triggeren.

```

213     run_sol: // Motor1 azimuth, motor 2 tilt.
214     // Riggins estimerte orientering er omgjort til INT,
215     // dette gir en hysteresis virkning.
216     // Rigger kjøres til solreferansens.
217 IF TO_INT(Motor1_pos) > Azimuth THEN
218     M1F := FALSE;
219     M1R := TRUE;
220 ELSIF TO_INT(Motor1_pos) < Azimuth THEN
221     M1R := FALSE;
222     M1F := TRUE;
223 ELSIF TO_INT(Motor1_pos) = Azimuth THEN
224     M1R := FALSE;
225     M1F := FALSE;
226 END_IF;
227
228 IF TO_INT(Motor2_pos) > Alpha THEN
229     M2F := FALSE;
230     M2R := TRUE;
231 ELSIF TO_INT(Motor2_pos) < Alpha THEN
232     M2R := FALSE;
233     M2F := TRUE;
234 ELSIF TO_INT(Motor2_pos) = Alpha THEN
235     M2R := FALSE;
236     M2F := FALSE;
237 END_IF;
238

```

Figur 28 kodekomponenten for kjøring av motorene til hovedmodusen **følg\_solen**.

```

252     // Kjør_algoritmen-trigger
253     // Når motoren har nådd solreferansens stilles algoritmerefransene
254     // til solreferansene og algoritmen starter.
255 IF TO_INT(Motor1_pos) = Azimuth AND TO_INT(Motor2_pos) = Alpha THEN
256     Alg_azi := LREAL_TO_INT(Motor1_pos);
257     Alg_alp := LREAL_TO_INT(Motor2_pos);
258     Alg_state := Start_alg;
259     Alg_sub_states := a;
260     State := run_alg;
261 END_IF

```

Figur 29 kodekomponenten til **kjør\_algoritmen**-modus-triggeren for hovedmodusen **følg\_solen**.

```

250     // Vind-trigger
251     // Hver gang vinden oversiger en grenseverdi skal en timer starte/restarte.
252     // Når timeren har utløpt har det gått 15 minutter der vinden har vært
253     // mindre enn grensen.
254 IF IoConfig_Globals_Mapping.Wind_sensor > Wind_speed_limit THEN
255     Wind_timer      ( IN := TRUE , PT := TO_TIME(15*60*1000) );
256     state           := wind;
257     ELSIF Alphaa < 0 THEN// Hvis solen er under horisonten skal den gå i vind-modus
258     state           := wind;
259     ELSE Wind_timer ( IN := FALSE );
260 END_IF
261

```

Figur 30 Kodekomponenten til **vind**-modus-triggeren for hovedmodusen **følg\_solen**.

Hovedmodusen vind

Figur 31 viser hvordan kodekomponenten for kjøring av motorene for hovedmodusen **vind** ble kodet. Figur 32 viser **vind**-modus-triggeren til hovedmodusen **vind**. For at vindmodus-stoppeklokken skal resettes hver gang vinden overstiger grensen, må **vind**-modus-triggeren være i **vind**-modusen. Hvis solen går under horisonten, skal riggen gå i **vind**-modusen. Figur 33 viser **følg\_solen**-modus-triggeren for hovedmodusen **vind**.

```

499     wind:
500     M1R := FALSE; //Azimutalvinkelen trenger ikke endre seg når den går i vindmodus.
501     M1F := FALSE;
502     // Her kjøres riggens elevsjonsakse til vindposisjonene, som er 90grader.
503 IF TO_INT(Motor2_pos) > Wind_pos_alp THEN
504     M2F := FALSE;
505     M2R := TRUE;
506 ELSIF TO_INT(Motor2_pos) < Wind_pos_alp THEN
507     M2R := FALSE;
508     M2F := TRUE;
509 ELSIF TO_INT(Motor2_pos) = Wind_pos_alp THEN
510     M2R := FALSE;
511     M2F := FALSE;
512 END_IF;
513

```

Figur 31 kodekomponenten for kjøring av motorene til hovedmodusen **vind**.

```

516     // Vind-trigger
517     // Hver gang vinden oversiger en grenseverdi skal en timer starte/restarte.
518     // Når timeren har utløpt har det gått 15 minutter der vinden har vært
519     // mindre enn grensen.
520 IF IoConfig_Globals_Mapping.Wind_sensor > Wind_speed_limit THEN
521     Wind_timer      ( IN := TRUE , PT := TO_TIME(15*60*1000) );
522     state           := wind;
523     ELSIF Alphaa < 0 THEN// Hvis solen er under horisonten skal den gå i vind-modus.
524     state           := wind;
525     ELSE Wind_timer ( IN := FALSE );
526 END_IF
527

```

Figur 32 Kodekomponenten til **vind**-modus-triggeren for hovedmodusen **vind**.

```

542 // Trigger for å forlate vind-modusen.
543 // Hvis det er mer enn 15 minutter siden vinden oversteg vindgrensen og solen er over horisonten,
544 IF Wind_timer.Q = FALSE AND Alphaa > 0 THEN // skal hovedmodusen stilles til følg_solen-modusen.
545     state := run_sol; // Den kan eventuelt stilles til start-modusen for å nullstille
546 END_IF // eventuelle avvik mellom estimert orientering og faktisk orientering.
547

```

Figur 33 kodekomponenten til **følg\_solen**-modus-triggeren for hovedmodusen **vind**.

### Hovedmodusen **stopp**

Figur 34 viser koden for hovedmodusen **stopp**.

```

550     stop: (*Nødstop - Stopper alle motorene, ingen vei ut*)
551     M1F := FALSE;
552     M1R := FALSE;
553     M2F := FALSE;
554     M2R := FALSE;
555 END_CASE

```

Figur 34 kodekomponenten til hovedmodusen **stopp**.

### Hovedmodusen **Kjør\_algoritmen**

Figur 35 viser **vind**-modus-triggeren til hovedmodusen **kjør\_algoritmen**. Figur 36 viser kodekomponenten for kjøring av motorene til hovedmodusen **kjør\_algoritmen**. Figur 37 viser **følg\_solen**-modus-triggeren til hovedmodusen **kjør\_algoritmen**. Når avstanden mellom algoritmereferansene og solreferansene overstiger en gitt verdi, skal hovedmodusen **følg\_solen** aktiveres.

```

262     run_alg:
263     // Vind-trigger
264     // Hver gang vinden oversiger en grenseverdi skal en timer starte/restarte.
265     // Når timeren har utløpt har det gått 15 minutter der vinden har vært
266     // mindre enn grensen.
267 IF IoConfig_Globals_Mapping.Wind_sensor > Wind_speed_limit THEN
268     Wind_timer      ( IN := TRUE , PT := TO_TIME(15*60*1000) );
269     state           := wind;
270 ELSIF Alphaa < 0 THEN // Hvis solen er under horisonten skal den gå i vind-modus
271     state           := wind;
272 ELSE Wind_timer ( IN := FALSE );
273 END_IF
---
```

Figur 35 kodekomponenten til **vind**-modus-triggeren til hovedmodusen **kjør\_algoritmen**.

```

276      // Her kjøres riggen til algoritme referanden.
277      // Den estimerte posisjonen til riggen konvertres
278      // til INT, for å oppnå hysterese effekt
279  IF TO_INT(Motor1_pos) > Alg_azi THEN
280      M1F := FALSE;
281      M1R := TRUE;
282  ELSIF TO_INT(Motor1_pos) < Alg_azi THEN
283      M1R := FALSE;
284      M1F := TRUE;
285  ELSIF TO_INT(Motor1_pos) = Alg_azi THEN
286      M1R := FALSE;
287      M1F := FALSE;
288  END_IF;
289
290  IF TO_INT(Motor2_pos) > Alg_alp THEN
291      M2F := FALSE;
292      M2R := TRUE;
293  ELSIF TO_INT(Motor2_pos) < Alg_alp THEN
294      M2R := FALSE;
295      M2F := TRUE;
296  ELSIF TO_INT(Motor2_pos) = Alg_alp THEN
297      M2R := FALSE;
298      M2F := FALSE;
299  END_IF;
300

```

Figur 36 kodekomponenten til kjøring av motorene til hovedmodusen **kjør\_algoritmen**.

```

301      // Her regnes et overslag av avstanden mellom solreferansene og algoritmereferansene
302      // Hvis avstanden er større enn en gitt grense skal den gå i følg_solen-modusen
303  avstand := sqrt_r(sqrt_r(TO_REAL(Azimuth) - TO_REAL(Alg_azi)) +
304      sqrt_r(TO_REAL(Alpha) - TO_REAL(Alg_alp)));
305  IF avstand > Avstand_limit THEN
306      State := run_sol;
307  END_IF

```

Figur 37 kodekomponenten for **følg\_solen**-modus-triggeren til hovedmodusen **kjør\_algoritmen**.

## Algoritmehodene

Figur 38 viser koden for algoritmehodene **start**. Figur 39 viser triggeren for algoritmehodene **primer**. Figur 40 viser koden for algoritmehodene **vent**.

```
311 // Når den kan kjøre igjennom alle statene som tilsvarer en retning
312 // uten å hoppe tilbake har den nådd et maksimumspunkt.
313 // Når den har nådd et maksimumspunkt skal den vente i 10 minutter
314 CASE Alg_state OF
315     Start_alg:           // Dette er en resett, for å være sikker på at
316     Alg_alp := Alpha;    // algoritmen starter som den skal.
317     Alg_azi := Azimuth;
318     Alg_state := primer;
319
```

Figur 38 kodekomponenten for algoritmehodene **start**.

```
320 primer: // Her måles effekten når rigen er orientert som den skal, deretter tester den opp-retningen.
321 IF TO_INT(Motor1_pos) = Alg_azi AND TO_INT(Motor2_pos) = Alg_alp AND Power_measured = FALSE THEN
322     Power_measure := TRUE;
323 ELSIF Power_measured = TRUE THEN
324     Power_previous := power_test;
325     reset_power := TRUE;
326     Alg_state := up;
327 END_IF
328
```

Figur 39 kodekomponenten for algoritmehodene **primer**.

```
485 wait: // Vent-modus, her starter en timer på 10 minutter
486 Alg_timer(IN := TRUE , PT := TIME#10M);
487 IF Alg_timer.Q = TRUE THEN // Når timeren har utløpt starter
488     Alg_timer(IN := FALSE); // optimaliserings algoritmen på nytt der den
489     Alg_sub_states := a; // sluttet forrige gang, (til primer).
490     Alg_state := primer;
491 END_IF
```

Figur 40 kodekomponenten for algoritmehodene **vent**.

Figur 41 viser koden for alle undermodene til algoritmehodene **opp**. Koden er i henhold til flytskjema oppgitt i Figur 15.

```

330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366

up:
CASE Alg_sub_states OF
  a: // Her inkrementerte vinkelen som skal testes.
    IF Alg_alp < Alpha_limit_upper THEN
      Alg_alp := Alg_alp + 1;
    END_IF
  Alg_sub_states := b;

  b: // Her måles effekten når riggen er orientert som den skal.
    IF TO_INT(Motor1_pos) = Alg_azi AND TO_INT(Motor2_pos) = Alg_alp AND Power_measured = FALSE THEN
      Power_measure := TRUE;
    ELSEIF Power_measured = TRUE THEN
      Alg_sub_states := c;
    END_IF

  c:
    IF TO_REAL(Power_test) < TO_REAL(Power_previous) * 0.9 OR TO_REAL(Power_test) > TO_REAL(Power_previous) * 1.1 THEN
      Alg_state := wait; // Hvis effekten endret seg med mer enn 10% skal den gå i vente-modus og vente 10 minutter.

    ELSEIF Power_previous < Power_test THEN // Ellers, hvis effekten økte skal den hoppe tilbake til "a" for å øke vinkelen
      Power_previous := Power_test; // igjen, for å teste om effekt vil øke mer.
      Alg_jump_back_false_count := 0; // Her nullstilles en teller som viser hvor mange ganger algoritmen ikke har
      Alg_sub_states := a; // vært i stand til å øke effekten.
    ELSE
      Alg_alp := Alg_alp - 1; // Hvis effekten ikke økte skal riggen stilles en grad tilbake,
      Alg_jump_back_false_count := Alg_jump_back_false_count + 1; // slik at riggen er orientert tilbake der effekten var høyere.
      Alg_sub_states := a; // 356: inkrementerer telleren som viser hvor mange ganger algoritmen ikke har vært i stand til å øke effekten.
      Alg_state := right; // Her stilles den til å sjekke neste retning.
      IF Alg_jump_back_false_count >= 4 THEN // Når denne telleren har nådd 4 betyr det at den har nådd et maksimumspunkt.
        Alg_jump_back_false_count:= 0; // Her nullstilles telleren.
        Alg_state := wait; // Her går den i vente modus, fordi den har nådd et maksimumspunkt.
      END_IF
    END_IF
  END_IF
  reset_power := TRUE; // Effektmåleren må få beskje om å resette seg.
END_CASE

```

Figur 41 kodekomponenten for under-modiene til algoritmmodusen **opp**.

```

381      right: // @st
382      CASE Alg_sub_states OF
383      a:
384          IF Alg_azi < Azimuth_limit_upper THEN
385              Alg_azi := Alg_azi + 1;
386          END_IF
387          Alg_sub_states := b;
388
389      b:
390          IF TO_INT(Motor1_pos) = Alg_azi AND TO_INT(Motor2_pos) = Alg_alp AND Power_measured = FALSE THEN
391              Power_measure := TRUE;
392          ELSEIF Power_measured = TRUE THEN
393              Alg_sub_states := c;
394          END_IF
395
396      c:
397          IF TO_REAL(Power_test) < TO_REAL(Power_previous) * 0.9 OR TO_REAL(Power_test) > TO_REAL(Power_previous) * 1.1 THEN
398              Alg_state := wait;
399
400          ELSEIF Power_previous < Power_test THEN
401              Power_previous := Power_test;
402              Alg_jump_back_false_count := 0;
403              Alg_sub_states := a;
404
405          ELSE
406              Alg_azi := Alg_azi - 1;
407              Alg_jump_back_false_count := Alg_jump_back_false_count + 1;
408              Alg_sub_states := a;
409              Alg_state := down;
410              IF Alg_jump_back_false_count >= 4 THEN
411                  Alg_jump_back_false_count := 0;
412                  Alg_state := wait;
413              END_IF
414          END_IF
415          reset_power := TRUE;
416      END_CASE
417

```

Figur 42 kodekomponenten for under-modiene til algoritmemodusen **høyre**.



```

420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456

down:
CASE Alg_sub_states OF
a:
  IF Alg_alp > Alpha_limit_lower THEN
    Alg_alp := Alg_alp -1;
  END_IF
  Alg_sub_states := b;
b:
  IF TO_INT(Motor1_pos) = Alg_az1 AND TO_INT(Motor2_pos) = Alg_alp AND Power_measured = FALSE THEN
    Power_measure := TRUE;
  ELSEIF Power_measured = TRUE THEN
    Alg_sub_states := c;
  END_IF
c:
  IF TO_REAL(Power_test) < TO_REAL(Power_previous) * 0.9 OR TO_REAL(Power_test) > TO_REAL(Power_previous) * 1.1 THEN
    Alg_state := wait;
  ELSEIF Power_previous < Power_test THEN
    Power_previous := Power_test;
    Alg_jump_back_false_count := 0;
    Alg_sub_states := a;
  ELSE
    Alg_alp := Alg_alp +1;
    Alg_jump_back_false_count := Alg_jump_back_false_count +1;
    Alg_sub_states := a;
    Alg_state := left;
    IF Alg_jump_back_false_count >= 4 THEN
      Alg_jump_back_false_count:= 0;
      Alg_state := wait;
    END_IF
  END_IF
END_IF
reset_power := TRUE;
END_CASE

```

Figur 43 kodekomponenten for under-modiene til algoritmemodusen **ned**.

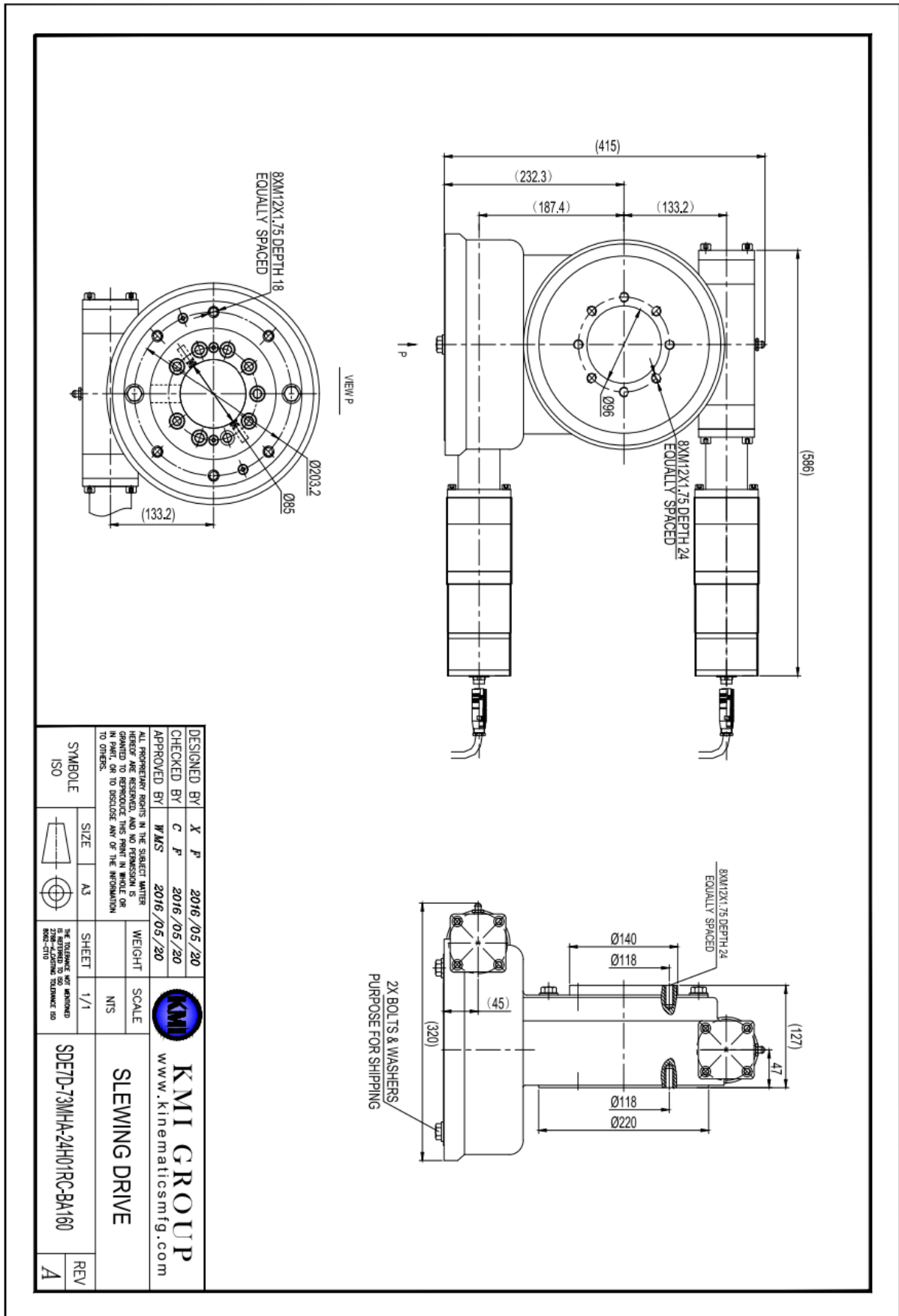
```

458 left: // vest
459 CASE Alg_sub_states OF
460 a:
461 IF Alg_azi > Azimuth_limit_lower THEN
462 Alg_azi := Alg_azi - 1;
463 END_IF
464 Alg_sub_states := b;
465
466 b:
467 IF TO_INT(Motor1_pos) = Alg_azi AND TO_INT(Motor2_pos) = Alg_alp AND Power_measured = FALSE THEN
468 Power_measure := TRUE;
469 ELSEIF Power_measured = TRUE THEN
470 Alg_sub_states := c;
471 END_IF
472
473 c:
474 IF TO_REAL(Power_test) < TO_REAL(Power_previous) * 0.9 OR TO_REAL(Power_test) > TO_REAL(Power_previous) * 1.1 THEN
475 Alg_state := wait;
476
477 ELSEIF Power_previous < Power_test THEN
478 Power_previous := Power_test;
479 Alg_jump_back_false_count := 0;
480 Alg_sub_states := a;
481
482 ELSE
483 Alg_azi := Alg_azi + 1;
484 Alg_jump_back_false_count := Alg_jump_back_false_count + 1;
485 Alg_sub_states := a;
486 Alg_state := up;
487 IF Alg_jump_back_false_count >= 4 THEN
488 Alg_jump_back_false_count := 0;
489 Alg_state := wait;
490 END_IF
491 END_IF
492 reset_power := TRUE;
493 END_CASE
494

```

Figur 44 kodekomponenten for under-modiene til algoritmemodusen **venstre**.

# Datablad til motoren

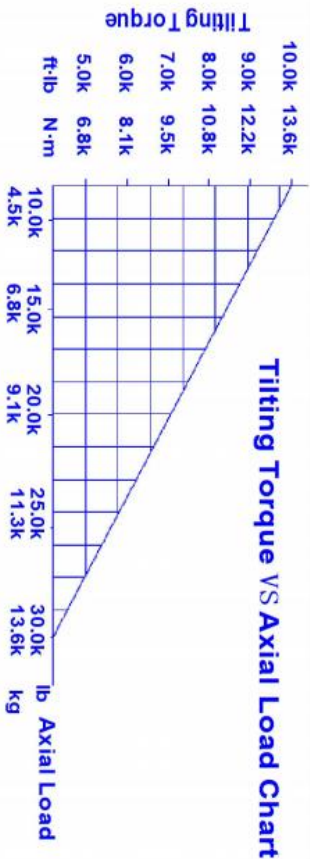


## SLEWING DRIVES PERFORMANCE DATA

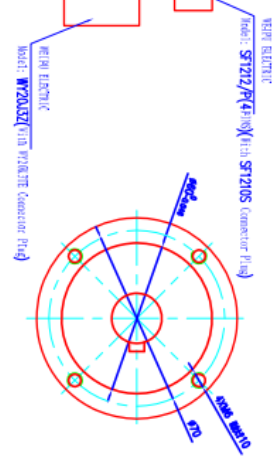
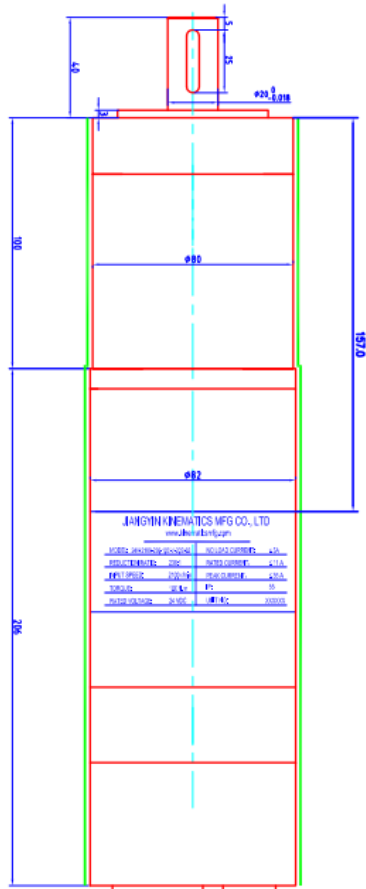
Model Code : SDE7D-73MHA-24H01RC-BA160

		Gearmotor	
Slewing Drive Ratio	73:1	Type Code	24H-1560-223-60-V-20S-70
Rotating Output Rated Speed	0.1 rpm	Rated Voltage	24 VDC
Efficiency	30%	Output Speed	7 rpm
International Protection (IP)	55	Rated Current	5.0 A
Slewing Drive Temperature	- 20°C to + 80°C	Output Power	45.0 W
Rated Output Torque	1314N·m	Motor Rated Speed	1560 rpm
Max. Output Torque	2628N·m	Output Torque	60 N·m
Torsion Stiffness	762Nm/mRad	Stall Torque	120 N·m
Bending Stiffness	1083Nm/mRad		

		HA Type	
Slewing Drive Loading Data			
Normal Output Torque	2,010	N·m	1,483 ft·lb
Max. Output Torque (3 Sec.)	4,020	N·m	2,965 ft·lb
Backwards Holding Torque	10,338	N·m	7,625 ft·lb
Tilting Torque	13,556	N·m	9,999 ft·lb
Static Radial Rating	53.28	kN	11,977 lb
Static Axial Rating	133.20	kN	29,943 lb
Dynamic Radial Rating	27.91	kN	6,274 lb
Dynamic Axial Rating	31.90	kN	7,171 lb



DESIGNED BY	<i>X F</i>	2016/05/20	 <b>KMI GROUP</b> www.kinematicsmfg.com
CHECKED BY	<i>C P</i>	2016/05/20	
APPROVED BY	<i>WMS</i>	2016/05/20	
ALL RESPECTIVE SIGNS IN THE SUBJECT MATTER HEREBY ARE RESERVED, AND NO PERMISSION IS GRANTED TO REPRODUCE THIS PRINT IN WHOLE OR IN PART, OR TO DISCLOSE ANY OF THE INFORMATION TO OTHERS.			
SYMBOL	SIZE	SHEET	<b>SLEWING DRIVE</b> SDE7D-73MHA-24H01RC-BA160
ISO	A4	1/1	
			REV <b>A</b>



- WY20J3Z Electric Connection:**
- Pin 1: Motor, 0V (MOTOR INPUT), (Black thick wire inside)
  - Pin 2: Motor, +24V (MOTOR INPUT), (Black thick wire inside)
  - Pin 3: Motor, GND (MOTOR INPUT), (Black thick wire inside)
  - SF12/P4 Pins Electric Connection:
  - Pin 1: Hall, 0V (HALL INPUT), (Black thin wire inside)
  - Pin 2: Hall, +12V(+24) (HALL INPUT), (Red thin wire inside)
  - Pin 3: Hall, Signal A (HALL OUTPUT square wave signal)(Green thin wire inside)
  - Pin 4: Hall, Signal B (HALL OUTPUT square wave signal)(Yellow thin wire inside)

**ATTENTION**  
Do not exchange the connection for Hall 0V and Hall 12V!  
If you do so, Hall will be destroyed!

Motor Parameters	
Rated voltage U <sub>n</sub> (VDC)	24
Rated speed V <sub>n</sub> (rpm)	2250±8%
Rated current I <sub>n</sub> (A)	<5
Current I <sub>h</sub> (A)	<1
Speed V <sub>h</sub> (rpm)	2100±8%
Efficiency η (%)	>70
Planetary Reducer Parameters	
Ratio	236: 1
Stage	3
Rated output torque T <sub>2</sub> (Nm)	100
Inherent output torque T <sub>1</sub> (Nm)	150
Maximum output torque T <sub>3</sub> (Nm)	200
Backlash (arcmin)	16
Efficiency η (%)	>70
Hall Parameters	
No. of Hall	2
Hall model	A3282
Hall signal	Open Collector
Other Parameters	
Average lifespan (h)	2000h
Noise (dB)	55
Environmental temperature T <sub>a</sub> (°C)	-40 - 80
IP	50
MAGNETIC PULSE GENERATOR DATA	
Excitation	4.5VDC
Current	100mA
Output	5VDC
Resolution	10°
Resolution rate	10°/10ms
Speed range	100-2000rpm
Operating temperature	-40~+85
Storage temperature	-40~+125
Shock resistance	1000g
Vibration resistance	10g
Weight	100g

DESIGNED BY **WJS** 2012.10.27

CHECKED BY **F J** 2012.10.27

APPROVED BY **JIM** 2012.10.27

ALL PROPRIETARY RIGHTS IN THE SUBJECT MATTER HEREOF ARE RESERVED, AND NO PERMISSION IS GRANTED TO REPRODUCE THIS PRINT IN WHOLE OR IN PART, OR TO DISCLOSE ANY OF THE INFORMATION TO OTHERS.

SIZE **A3** SHEET

WEIGHT SCALE **1:1.5**

THE TOLERANCE NOT MENTIONED IS REFERRED TO ISO 2768-M

**KMI GROUP**  
WWW.KINEMATICS.MFG.COM

**Brushed DC Motor with Hall**

**24H-2100-236-100-V-20S-70**

REV **D**

**Datablad til pyranometret**

**Datablad til induksjonssensorene**

**Datablad til Vindmåleren Product number 4.3303.22.661**

**Datablad til optokoblerne**

**Datablad til operasjonsforsterkerne**

**Oppgavetekst**





# SMALL SENSOR ML-01



## Small sensor, great performance

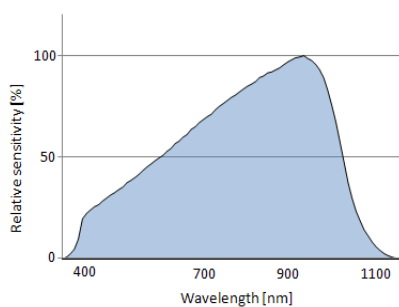
The ML-01 is an industrial grade solar sensor made for PV performance ratio measurements. Due to the compact dimensions of the sensor it can be easily integrated into any application.



The Si-photodiode sensor ML-01 is the link between reference cells and broadband thermopile pyranometers. Compared to the reference cells it has a proper cosine response and it is more compact. Moreover, it also benefits from the same characteristics as a PV module (response time, spectral and temperature response).

Due to its cone shaped diffuser, the ML-01 has a proper response to the incoming radiation at low solar elevation angles. This shape also minimizes soiling effects which could alter the quality of the measurement.

ML-01 is calibrated according to the international calibration method applied to PV reference cells (25°C / AM1.5G / 1000W/m<sup>2</sup>, AAA Solar Simulator IEC 60904-3 Spectral distribution).



Spectral characteristics

### Features

- Same spectral response as a PV module
- Low temperature dependency
- Fast response photodiode detector
- Small and lightweight
- Optimized directional response function
- Mounting plate with spirit level

### Specifications

	ML-01
Spectral response	400 - 1100nm (Mono-Crystalline)
Measuring range	0~2000 W/m <sup>2</sup>
Output (0~100mV Range)	0~100mV
Response time	< 1ms
Operating temperature	-30~+70°C
Temperature dependency (-10~+40°C)	< 3%
Directional response (0~80°)	< ±5%
Field of View	180°
Non-stability	< 2%/year
Output cable	5m for standard version (10m, 30m, or 50m also available)





## XS9C4A2A2P20

inductive sensor XS9 40x40x117 - PBT - Sn25 mm - 12..24VDC - terminals



[Download your XS9C4A2A2P20 datasheet](#)



[Characteristics](#) | [Documents & Downloads](#)

### Main

Hide

<b>Range of product</b>	OsiSense XS
<b>Series name</b>	General purpose
<b>Sensor type</b>	Inductive proximity sensor
<b>Device application</b>	-
<b>Sensor name</b>	XS9
<b>Sensor design</b>	Form 40 x 40 x 117
<b>Size</b>	117 mm
<b>Body type</b>	Fixed
<b>Detector flush mounting acceptance</b>	Non flush mountable
<b>Material</b>	Plastic
<b>Enclosure material</b>	PBT
<b>Type of output signal</b>	Analogue
<b>Wiring technique</b>	2-wire
<b>[Sn] nominal sensing distance</b>	25 mm
<b>Output circuit type</b>	DC
<b>Analogue output range</b>	4...20 mA
<b>Electrical connection</b>	Screw-clamp terminals, clamping capacity: 4 x 1.5 mm <sup>2</sup>
<b>[Us] rated supply voltage</b>	12...24 V DC with reverse polarity protection
<b>IP degree of protection</b>	IP65 conforming to IEC 60529 IP67 conforming to IEC 60529 IP69K conforming to DIN 40050

### Complementary

Hide

<b>Detection face</b>	5 positions turret head
<b>Front material</b>	PBT
<b>Operating zone</b>	2...27 mm
<b>Differential travel</b>	3...15% of Sr
<b>Repeat accuracy</b>	<= 3% of Sr
<b>Linearity error</b>	<= 3% of Sr
<b>Cable entry number</b>	1 tapped entry for M20 x 1.5 cable gland
<b>Status LED</b>	1 LED yellow for output state
<b>Supply voltage limits</b>	12...36 V DC
<b>Current consumption</b>	0...4 mA at no-load
<b>Delay first up</b>	<= 7.5 ms
<b>Delay response</b>	<= 6 ms
<b>Delay recovery</b>	<= 6 ms
<b>Marking</b>	CE
<b>Height</b>	40 mm

<b>Length</b>	40 mm
<b>Width</b>	117 mm
<b>Product weight</b>	0.244 kg


## Environment

 Hide

<b>Product certifications</b>	CCC CSA UL
<b>Ambient air temperature for operation</b>	-25...70 °C
<b>Ambient air temperature for storage</b>	-40...85 °C
<b>Vibration resistance</b>	25 gn amplitude = +/- 2 mm (f = 10...55 Hz) conforming to IEC 60068-2-6
<b>Shock resistance</b>	50 gn for 11 ms conforming to EN 60068-2-27

## Offer Sustainability

 Hide

<b>Sustainable offer status</b>	Green Premium product
<b>RoHS (date code: YYWW)</b>	Compliant - since 1213 - Schneider Electric declaration of conformity
<b>REACH</b>	Reference not containing SVHC above the threshold
<b>Product environmental profile</b>	Available  <a href="#">download Product environmental</a>
<b>Product end of life instructions</b>	Need no specific recycling operations

## Contractual warranty

 Hide

<b>Period</b>	18 months
---------------	-----------

## WIND

### Wind Transmitter Classic

**Part number: 4.3303.22.xxx**

The wind transmitter is equipped with a contact-free opto-electronic scanner, which causes an extremely low starting speed. At the output the measuring value is available as digital signal.

The heating is electronically controlled. A plug-connection is situated in the shaft of the instrument. The instrument is mounted preferably onto a mast or traverse. All essential parts are made of anodised aluminium.



### Specification

**Part number: 4.3303.22.xxx**

Wind speed	
Measuring range	0.3 ... 50 m/s
Accuracy	< 2 % or ±0,3 m/s of meas. value
Delay distance	< 5 m
Data output digital	
Frequency	1042 Hz at 50 m/s
Operating voltage	
Electronic	3.3 ... 47 V DC
Heating	24 V AC/DC, 20 W
General	
Ambient temp.	-35 ... +80 °C
Electr. connection	see models
Mounting	onto mast tube Ø 1.5 ``
Protection	IP 55
Survival speed	60 m/s
Weight	1 kg
Fixing boring	Ø 50 x 50 mm

### Versions

As per 4.3303.22.xxx, but:

**Product number 4.3303.22.000**

Data output digital	
Frequency	0 ... 1042 Hz
Live zero support	yes

**General**

Electr. connection	5 pol. Stecker
--------------------	----------------

**Product number 4.3303.22.007**
**Data output digital**

Frequency	0 ... 1042 Hz
Live zero support	no

**General**

Electr. connection	7 pol. Stecker
--------------------	----------------

**Product number 4.3303.22.008**
**Wind speed**

Measuring range	0.5 ... 75 m/s
Accuracy	±0.5 m/s or ±2 % of meas. value

**Data output digital**

Frequency	0 ... 754 Hz
Live zero support	ja

**Operating voltage**

Electronic	3.3 ... 47 V DC
Heating	24 V AC/DC, 20 W

**General**

Electr. connection	5 pol plug connection
--------------------	-----------------------

**Product number 4.3303.22.018**
**Wind speed**

Measuring range	0.5 ... 75 m/s
Accuracy	±0.5 m/s or ±2 % of meas. value

**Data output digital**

Frequency	0 ... 754 Hz
Live zero support	nein

**Operating voltage**

Electronic	3.3 ... 47 V DC
Heating	24 V AC/DC, 20 W

**General**

Electr. connection	5 pol plug connection
--------------------	-----------------------

**Product number 4.3303.22.040**
**Wind speed**

Measuring range	0.3 ... 50 m/s
Accuracy	±0.4 m/s or ±2.5 % of meas value

**Data output analog**

Wind speed	0 ... 20 mA
------------	-------------

**Operating voltage**

Electronic	15 ... 24 V DC
Heating	24 V AC/DC, 20 W

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.041**
**Wind speed**

Measuring range	0.3 ... 50 m/s
Accuracy	±0.4 m/s or ±2.5 % of meas value

**Data output analog**

Wind speed	4 ... 20 mA
------------	-------------

**Operating voltage**

Electronic	15 ... 24 V DC
Heating	24 V AC/DC, 20 W

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.060**
**Wind speed**

Measuring range	0.3 ... 50 m/s
Accuracy	±0.4 m/s or ±2.5 % of meas value

**Data output analog**

Wind speed	0 ... 1 V
------------	-----------

**Operating voltage**

Electronic	15 ... 24 V DC
Heating	24 V AC/DC, 20 W

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.061**
**Wind speed**

Measuring range	0.3 ... 50 m/s
Accuracy	±0.4 m/s or ±2.5 % of meas value

**Data output analog**

Wind speed	0 ... 10 V
------------	------------

**Operating voltage**

Electronic	15 ... 24 V DC
Heating	24 V AC/DC, 20 W

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.073**
**Wind speed**

Measuring range	0.3 ... 50 m/s
Accuracy	±0.4 m/s or ±2.5 % of meas value

**Data output analog**

Wind speed	0 V ... 5 V
------------	-------------

**Operating voltage**

Electronic	15 ... 24 V DC
------------	----------------

Heating	24 V AC/DC, 20 W
---------	------------------

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.640**

**Wind speed**

Measuring range	0.3 ... 60 m/s
-----------------	----------------

Accuracy	±0.4 m/s or ±2.5 % of meas value
----------	----------------------------------

**Data output analog**

Wind speed	0 ... 20 mA
------------	-------------

**Operating voltage**

Electronic	15 ... 24 V DC
------------	----------------

Heating	24 V AC/DC, 20 W
---------	------------------

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.641**

**Wind speed**

Measuring range	0.3 ... 60 m/s
-----------------	----------------

Accuracy	±0.4 m/s or ±2.5 % of meas value
----------	----------------------------------

**Data output analog**

Wind speed	4 ... 20 mA
------------	-------------

**Operating voltage**

Electronic	15 ... 24 V DC
------------	----------------

Heating	24 V AC/DC, 20 W
---------	------------------

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.660**

**Wind speed**

Measuring range	0.3 ... 60 m/s
-----------------	----------------

Accuracy	±0.4 m/s or ±2.5 % of meas value
----------	----------------------------------

**Data output analog**

Wind speed	0 ... 1 V
------------	-----------

**Operating voltage**

Electronic	15 ... 24 V DC
------------	----------------

Heating	24 V AC/DC, 20 W
---------	------------------

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.661**

**Wind speed**

Measuring range	0.3 ... 60 m/s
-----------------	----------------

Accuracy	±0.4 m/s or ±2.5 % of meas value
----------	----------------------------------

**Data output analog**

Wind speed	0 ... 10 V
------------	------------

**Operating voltage**

Electronic	15 ... 24 V DC
Heating	24 V AC/DC, 20 W

**General**

Electr. connection	5 pol. plug connection
--------------------	------------------------

**Product number 4.3303.22.673**

**Wind speed**

Measuring range	0.3 ... 60 m/s
Accuracy	±0.4 m/s or ±2.5 % of meas value

**Data output analog**

Wind speed	0 ... 5 V
------------	-----------

**Operating voltage**

Electronic	15 ... 24 V DC
Heating	24 V AC/DC, 20 W

**General**

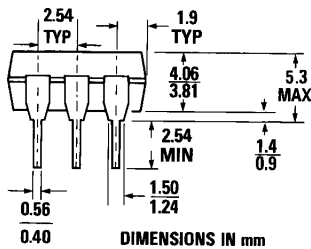
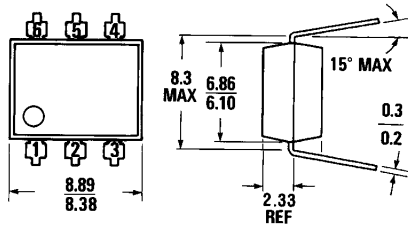
Electr. connection	5 pol. plug connection
--------------------	------------------------

## Accessories

No accessories are available for this product.

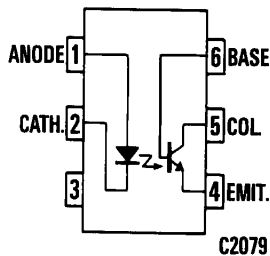


**PACKAGE DIMENSIONS**



DIMENSIONS IN mm  
PACKAGE CODE K

ST1603A



C2079

**DESCRIPTION**

The TIL111 is a phototransistor-type optically coupled isolator. An infrared emitting diode manufactured from specially grown gallium arsenide is selectively coupled with an NPN silicon phototransistor. The device is supplied in a standard plastic six-pin dual-in-line package.

**FEATURES**

- Underwriters Laboratory (UL) recognized File #E90700

**APPLICATIONS**

- Power supply regulators
- Digital logic inputs
- Microprocessor inputs
- Appliance sensor systems
- Industrial controls

**ABSOLUTE MAXIMUM RATINGS** ( $T_A=25^\circ\text{C}$  Unless Otherwise Specified)

**TOTAL PACKAGE**

Storage temperature .....	-55°C to 150°C
Operating temperature .....	-55°C to 100°C
Lead temperature	
(soldering, 10 sec) .....	260°C
Total package power dissipation at 25°C	
(LED plus detector) .....	260 mW
Derate linearly from 25°C .....	3.3 mW/°C

**INPUT DIODE**

Forward DC current .....	100 mA
Reverse voltage .....	3 V
Peak forward current	
(1 $\mu\text{s}$ pulse, 300 pps) .....	3.0 A
Power dissipation 25°C ambient .....	150 mW
Derate linearly from 25°C .....	2 mW/°C

**OUTPUT TRANSISTOR**

Power dissipation at 25°C .....	150 mW
Derate linearly from 25°C .....	2 mW/°C
$V_{CE0}$ .....	30 V
$V_{CBO}$ .....	70 V
$V_{ECO}$ .....	7 V
Collector current (continuous) .....	100 mA



**ELECTRICAL CHARACTERISTICS** (At 25°C Free-Air Temperature)

**INDIVIDUAL COMPONENT CHARACTERISTICS**

PARAMETER	SYMBOL	TIL111			UNIT	TEST CONDITIONS
		MIN.	TYP.	MAX.		
<b>INPUT DIODE</b> Input diode static reverse current	$I_R$			10	$\mu A$	$V_R = 3 V$
Input diode static forward voltage	$V_F$		1.2	1.4	V	$I_F = 16 mA$
<b>OUTPUT TRANSISTOR</b> Collector-base breakdown voltage	$V_{(BR)CBO}$	70			V	$I_C = 10 \mu A, I_E = 0, I_F = 0$
Collector-emitter breakdown voltage	$V_{(BR)CEO}$	30			V	$I_C = 1 mA, I_B = 0, I_F = 0$
Emitter-base breakdown voltage	$V_{(BR)EBO}$	7			V	$I_E = 10 \mu A, I_C = 0, I_F = 0$
Transistor static forward current transfer ratio	$h_{FE}$	100	300			$V_{CE} = 5 V, I_C = 10 mA, I_F = 0$

**TRANSFER CHARACTERISTICS**

PARAMETER	SYMBOL	TIL111			UNIT	TEST CONDITIONS	
		MIN.	TYP.	MAX.			
On-state collector current	Phototransistor operation	$I_{C(on)}$	2	7	$mA$	$V_{CE} = 0.4 V, I_F = 16 mA, I_B = 0$	
	Photodiode operation	$I_{C(on)}$	7	20	$\mu A$	$V_{CB} = 0.4 V, I_C = 16 mA, I_E = 0$	
Off-state collector current	Phototransistor operation	$I_{C(off)}$		1	50	$nA$	$V_{CE} = 10 V, I_C = 0, I_B = 0$
	Photodiode operation	$I_{C(off)}$		0.1	20		$V_{CB} = 10 V, I_C = 0, I_E = 0$
Collector-emitter saturation voltage	$V_{CE(sat)}$		0.25	0.4	V	$I_C = 2 mA, I_F = 16 mA, I_B = 0$	

**SWITCHING CHARACTERISTICS** (At 25°C Free-Air Temperature)

PARAMETER	SYMBOL	TIL111			UNIT	TEST CONDITIONS	
		MIN.	TYP.	MAX.			
Rise time	Phototransistor operation	$t_r$		5	10	$\mu s$	$V_{CC} = 10 V, I_{C(on)} = 2 mA, R_L = 100 \Omega$
Fall time							
Rise time	Photodiode operation	$t_r$		1		$\mu s$	$V_{CC} = 10 V, I_{C(on)} = 20 \mu A, R_L = 1 k\Omega$
Fall time							

**ISOLATION CHARACTERISTICS**

PARAMETER	SYMBOL	TIL111			UNIT	TEST CONDITIONS
		MIN.	TYP.	MAX.		
Input-to-output internal resistance	$r_{io}$	$10^{11}$			$\Omega$	$V_{ISO} = \pm 1.5 kV$
Input-to-output capacitance	$C_{io}$		1	1.3	pF	$V_{in-out} = 0, f = 1 MHz, \text{ See Note 6}$
Isolation voltage	$V_{iso}$		7500		VAC-PEAK	$I_{i.o} \leq 1 \mu A, 1 \text{ minute}$
			5300		VAC-RMS	$I_{i.o} \leq 1 \mu A, 1 \text{ minute}$

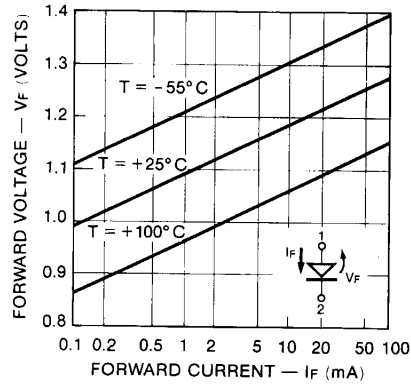


Fig. 1. Forward Voltage vs. Current

C1686

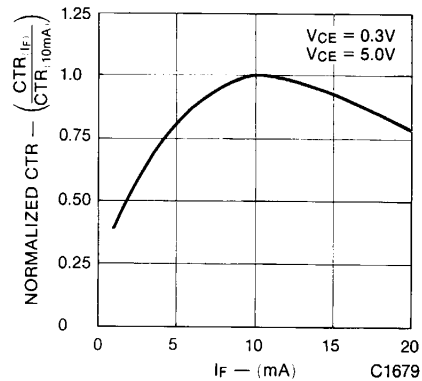


Fig. 2. Normalized CTR vs. Forward Current

C1679

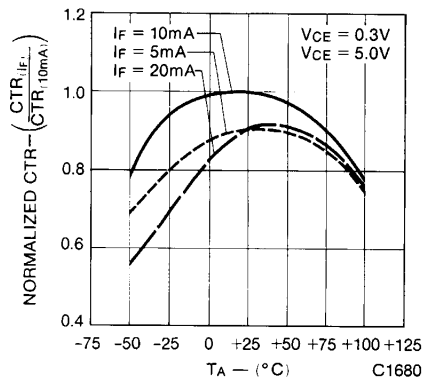


Fig. 3. Normalized CTR vs. Temperature

C1680

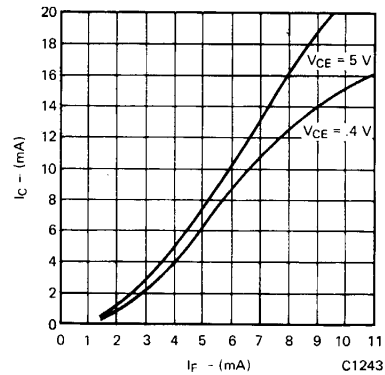


Fig. 4. Collector Current vs. Forward Current

C1243

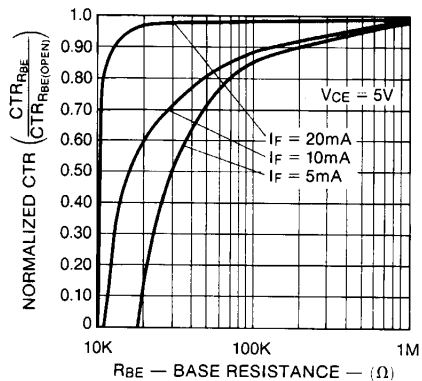


Fig. 5. CTR vs. RBE (Unsaturated)

C1681

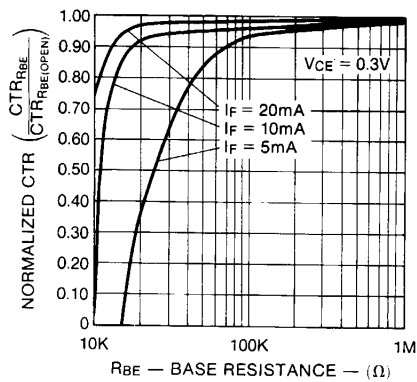


Fig. 6. CTR vs. RBE (Saturated)

C1682

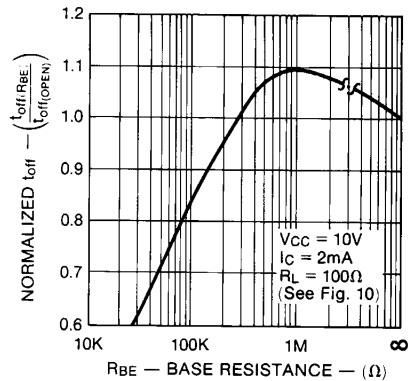


Fig. 7. Normalized  $T_{OFF}$  vs.  $R_{BE}$

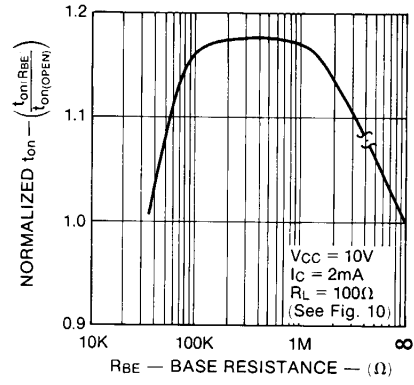


Fig. 8. Normalized  $T_{ON}$  vs.  $R_{BE}$

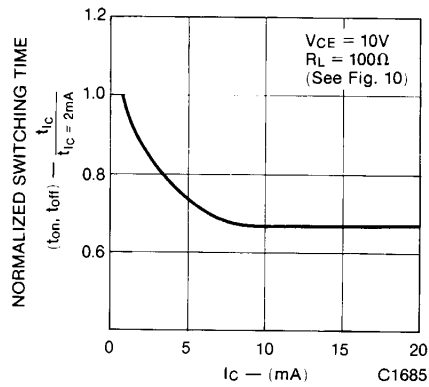


Fig. 9. Switching Time vs.  $I_C$

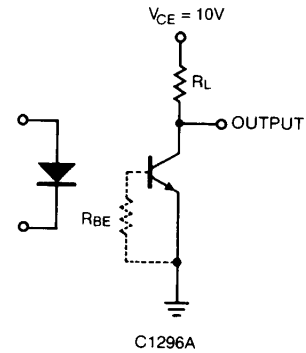


Fig. 10. Switching Time Test Circuit

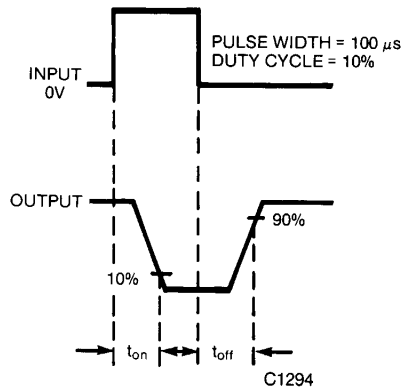


Fig. 11. Switching Time Waveforms



## PHOTOTRANSISTOR OPTOISOLATOR

---

### **DISCLAIMER**

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

### **LIFE SUPPORT POLICY**

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF FAIRCHILD SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury of the user.
2. A critical component in any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

This datasheet has been download from:

[www.datasheetcatalog.com](http://www.datasheetcatalog.com)

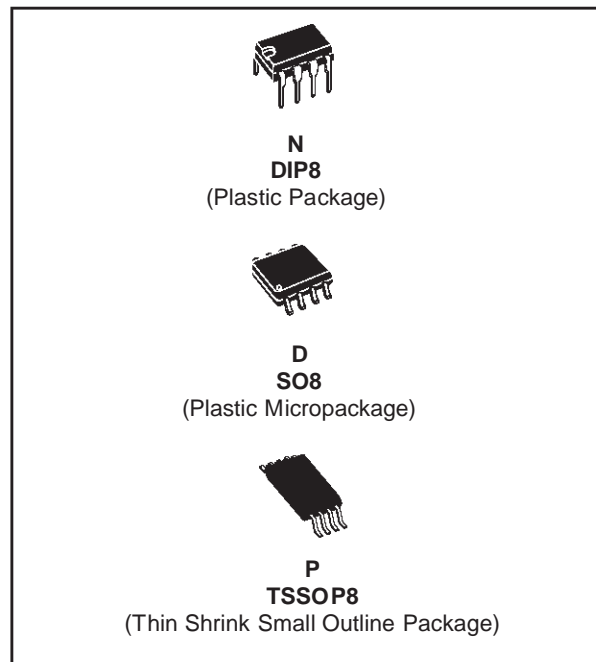
Datasheets for electronics components.



# TL082 TL082A - TL082B

## GENERAL PURPOSE J-FET DUAL OPERATIONAL AMPLIFIERS

- WIDE COMMON-MODE (UP TO  $V_{CC}^+$ ) AND DIFFERENTIAL VOLTAGE RANGE
- LOW INPUT BIAS AND OFFSET CURRENT
- OUTPUT SHORT-CIRCUIT PROTECTION
- HIGH INPUT IMPEDANCE J-FET INPUT STAGE
- INTERNAL FREQUENCY COMPENSATION
- LATCH UP FREE OPERATION
- HIGH SLEW RATE : 16V/ $\mu$ s (typ)

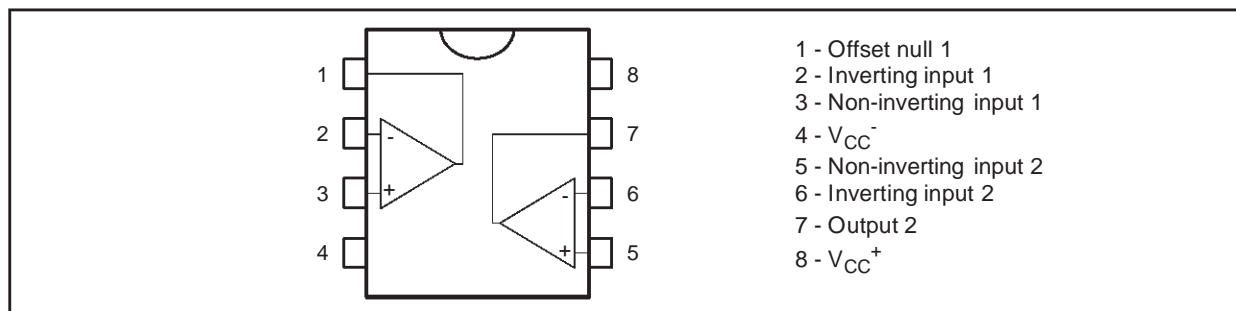


### DESCRIPTION

The TL082, TL082A and TL082B are high speed J-FET input dual operational amplifiers incorporating well matched, high voltage J-FET and bipolar transistors in a monolithic integrated circuit.

The devices feature high slew rates, low input bias and offset current, and low offset voltage temperature coefficient.

### PIN CONNECTIONS (top view)



### ORDER CODE

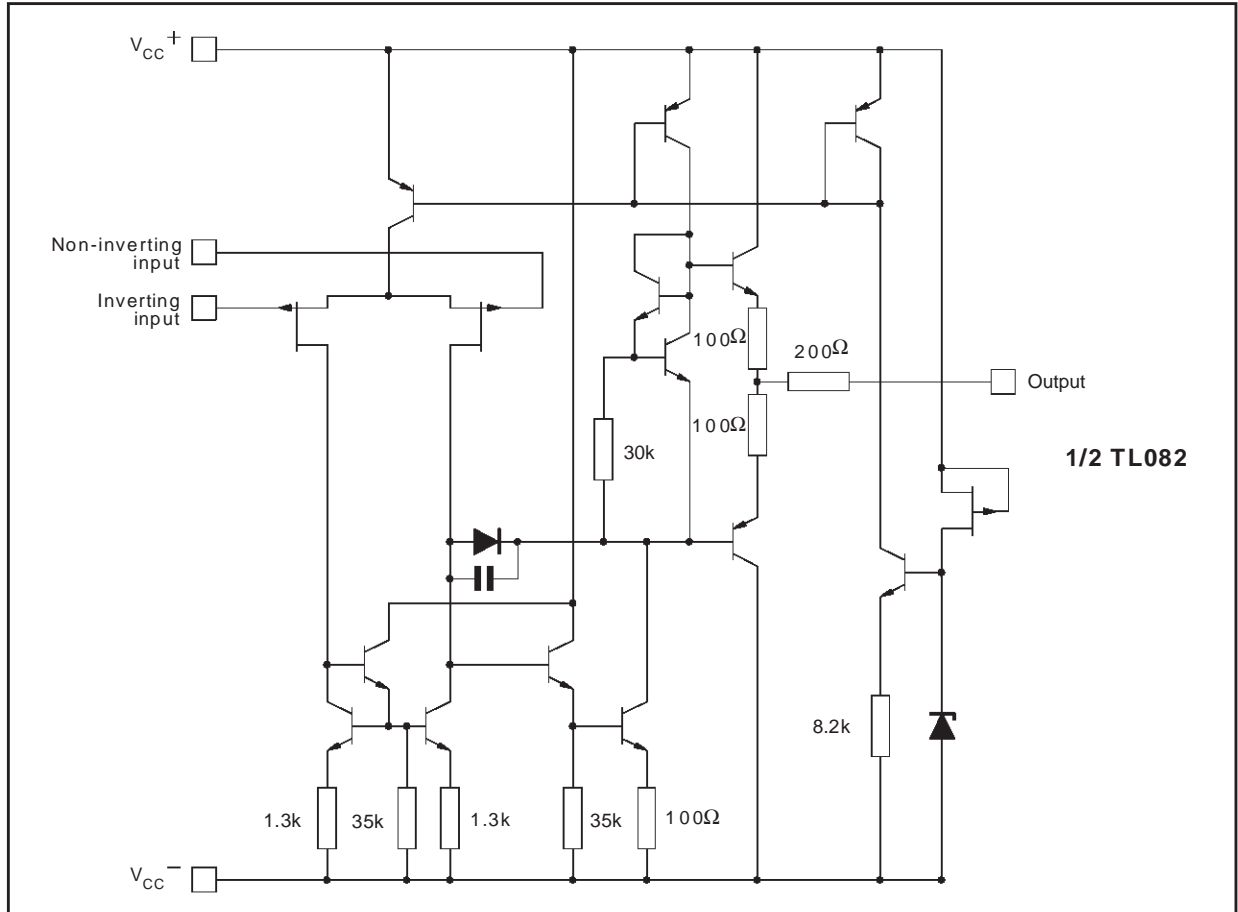
Part Number	Temperature Range	Package		
		N	D	P
TL082M/AM/BM	-55°C, +125°C	•	•	•
TL082I/AI/BI	-40°C, +105°C	•	•	•
TL082C/AC/BC	0°C, +70°C	•	•	•

**Example** : TL082CD, TL082IN

**N** = Dual in Line Package (DIP)  
**D** = Small Outline Package (SO) - also available in Tape & Reel (DT)  
**P** = Thin Shrink Small Outline Package (TSSOP) - only available in Tape & Reel (PT)

# TL082 - TL082A - TL082B

## SCHEMATIC DIAGRAM



## ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	TL082M, AM, BM	TL082I, AI, BI	TL082C, AC, BC	Unit
$V_{CC}$	Supply voltage - note 1)	±18			V
$V_i$	Input Voltage - note 2)	±15			V
$V_{id}$	Differential Input Voltage - note 3)	±30			V
$P_{tot}$	Power Dissipation	680			mW
	Output Short-circuit Duration - note 4)	Infinite			
$T_{oper}$	Operating Free-air Temperature Range	-55 to +125	-40 to +105	0 to +70	°C
$T_{stg}$	Storage Temperature Range	-65 to +150			°C

- All voltage values, except differential voltage, are with respect to the zero reference level (ground) of the supply voltages where the zero reference level is the midpoint between  $V_{CC+}$  and  $V_{CC-}$ .
- The magnitude of the input voltage must never exceed the magnitude of the supply voltage or 15 volts, whichever is less.
- Differential voltages are the non-inverting input terminal with respect to the inverting input terminal.
- The output may be shorted to ground or to either supply. Temperature and/or supply voltages must be limited to ensure that the dissipation rating is not exceeded.

**ELECTRICAL CHARACTERISTICS**

$V_{CC} = \pm 15V$ ,  $T_{amb} = +25^{\circ}C$  (unless otherwise specified)

Symbol	Parameter	TL082I,M,AC,AI,AM, BC,BI,BM			TL082C			Unit
		Min.	Typ.	Max.	Min.	Typ.	Max.	
$V_{io}$	Input Offset Voltage ( $R_S = 50\Omega$ ) $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$ TL082 TL082A TL082B TL082 TL082A TL082B		3 3 1	10 6 3 13 7 5		3	10 13	mV
$DV_{io}$	Input Offset Voltage Drift		10			10		$\mu V/^{\circ}C$
$I_{io}$	Input Offset Current - note 1) $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$		5	100 4		5	100 10	pA nA
$I_{ib}$	Input Bias Current -note 1 $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$		20	200 20		20	400 20	pA nA
$A_{vd}$	Large Signal Voltage Gain ( $R_L = 2k\Omega$ , $V_o = \pm 10V$ ) $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$	50 25	200		25 15	200		V/mV
SVR	Supply Voltage Rejection Ratio ( $R_S = 50\Omega$ ) $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$	80 80	86		70 70	86		dB
$I_{CC}$	Supply Current, no load $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$		1.4	2.5 2.5		1.4	2.5 2.5	mA
$V_{icm}$	Input Common Mode Voltage Range	$\pm 11$	+15 -12		$\pm 11$	+15 -12		V
CMR	Common Mode Rejection Ratio ( $R_S = 50\Omega$ ) $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$	80 80	86		70 70	86		dB
$I_{os}$	Output Short-circuit Current $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$	10 10	40	60 60	10 10	40	60 60	mA
$\pm V_{opp}$	Output Voltage Swing $T_{amb} = +25^{\circ}C$ $T_{min} \leq T_{amb} \leq T_{max}$ RL = 2k $\Omega$ RL = 10k $\Omega$ RL = 2k $\Omega$ RL = 10k $\Omega$	10 12 10 12	12 13.5		10 12 10 12	12 13.5		V
SR	Slew Rate ( $T_{amb} = +25^{\circ}C$ ) $V_{in} = 10V$ , $R_L = 2k\Omega$ , $C_L = 100pF$ , unity gain	8	16		8	16		V/ $\mu s$
$t_r$	Rise Time ( $T_{amb} = +25^{\circ}C$ ) $V_{in} = 20mV$ , $R_L = 2k\Omega$ , $C_L = 100pF$ , unity gain		0.1			0.1		$\mu s$
$K_{ov}$	Overshoot ( $T_{amb} = +25^{\circ}C$ ) $V_{in} = 20mV$ , $R_L = 2k\Omega$ , $C_L = 100pF$ , unity gain		10			10		%
GBP	Gain Bandwidth Product ( $T_{amb} = +25^{\circ}C$ ) $V_{in} = 10mV$ , $R_L = 2k\Omega$ , $C_L = 100pF$ , $f = 100kHz$	2.5	4		2.5	4		MHz
$R_i$	Input Resistance		$10^{12}$			$10^{12}$		$\Omega$

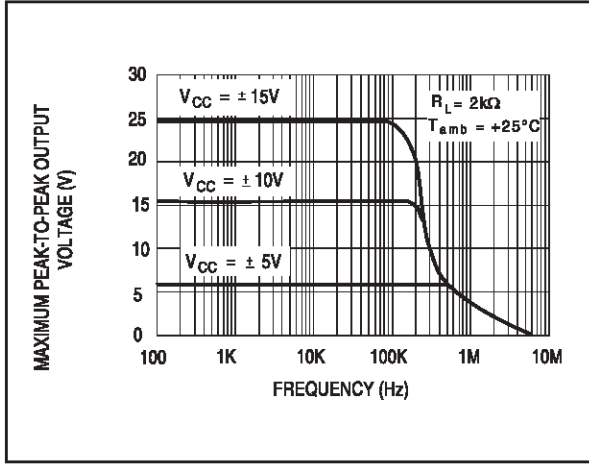


## TL082 - TL082A - TL082B

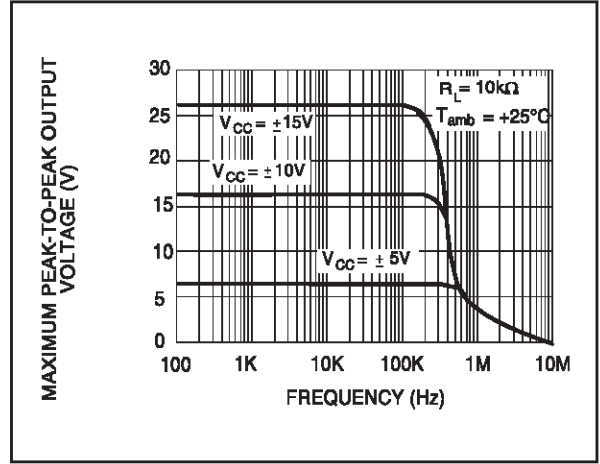
Symbol	Parameter	TL082I,M,AC,AI,AM, BC,BI,BM			TL082C			Unit
		Min.	Typ.	Max.	Min.	Typ.	Max.	
THD	Total Harmonic Distortion ( $T_{amb} = +25^{\circ}\text{C}$ ), $f = 1\text{kHz}$ , $R_L = 2\text{k}\Omega$ , $C_L = 100\text{pF}$ , $A_V = 20\text{dB}$ , $V_o = 2V_{pp}$		0.01			0.01		%
$e_n$	Equivalent Input Noise Voltage $R_S = 100\Omega$ , $f = 1\text{KHz}$		15			15		$\frac{\text{nV}}{\sqrt{\text{Hz}}}$
$\phi_m$	Phase Margin		45			45		degrees
$V_{o1}/V_{o2}$	Channel Separation $A_V = 100$		120			120		dB

- The input bias currents are junction leakage currents which approximately double for every  $10^{\circ}\text{C}$  increase in the junction temperature.

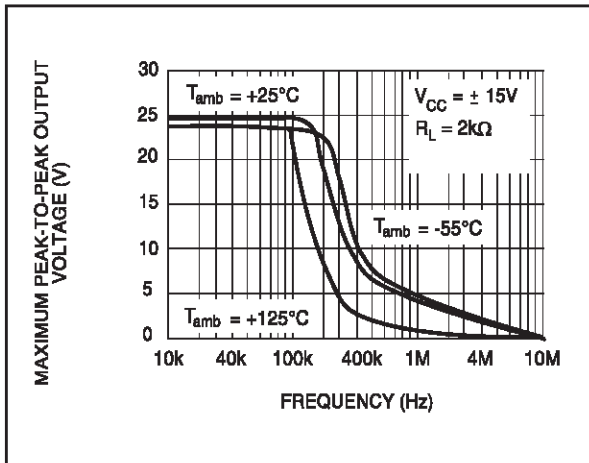
**MAXIMUM PEAK-TO-PEAK OUTPUT VOLTAGE versus FREQUENCY**



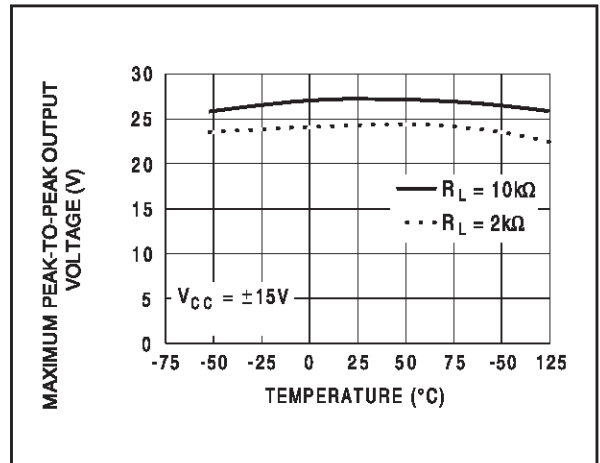
**MAXIMUM PEAK-TO-PEAK OUTPUT VOLTAGE versus FREQUENCY**



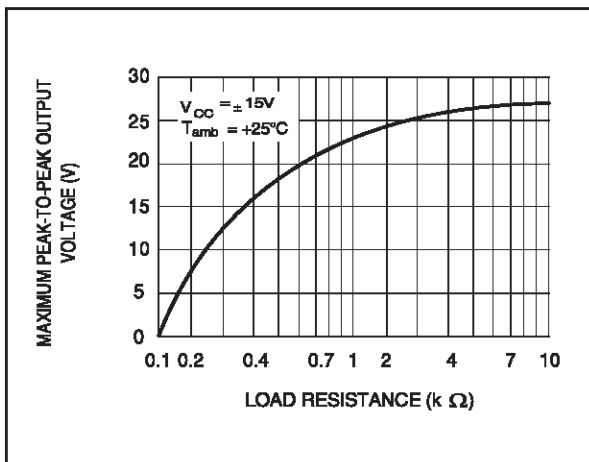
**MAXIMUM PEAK-TO-PEAK OUTPUT VOLTAGE versus FREQUENCY**



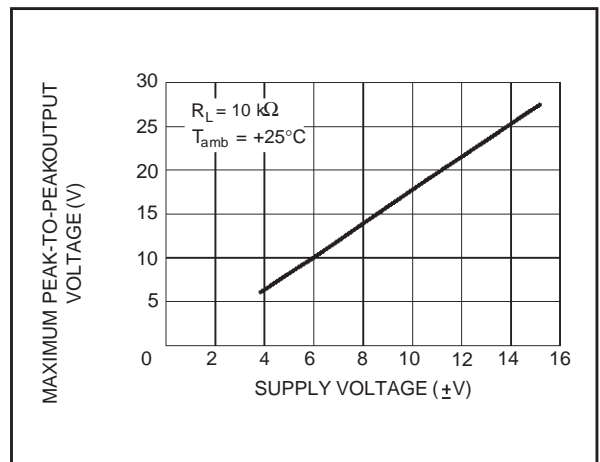
**MAXIMUM PEAK-TO-PEAK OUTPUT VOLTAGE versus FREE AIR TEMP.**



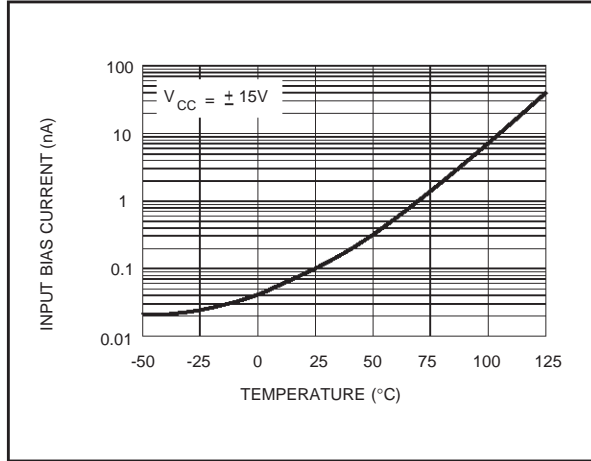
**MAXIMUM PEAK-TO-PEAK OUTPUT VOLTAGE versus LOAD RESISTANCE**



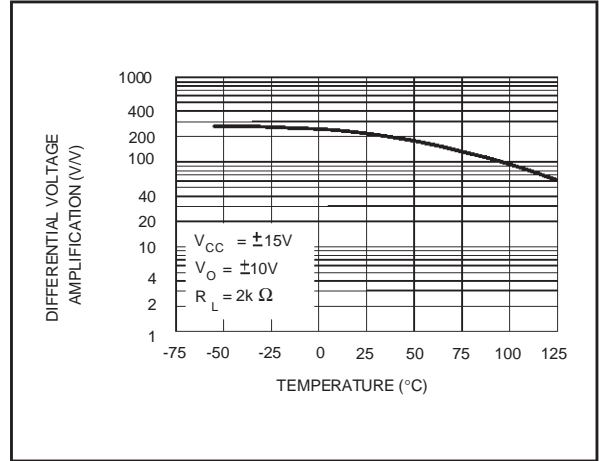
**MAXIMUM PEAK-TO-PEAK OUTPUT VOLTAGE versus SUPPLY VOLTAGE**



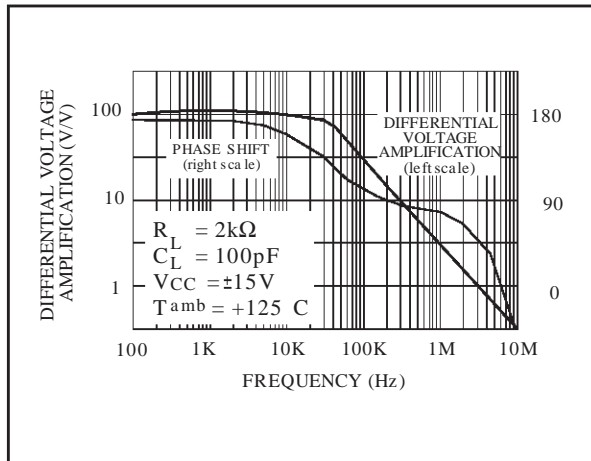
**INPUT BIAS CURRENT versus FREE AIR TEMPERATURE**



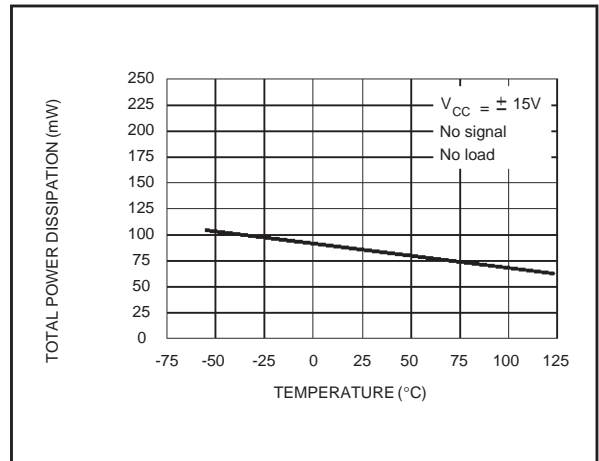
**LARGE SIGNAL DIFFERENTIAL VOLTAGE AMPLIFICATION versus FREE AIR TEMP.**



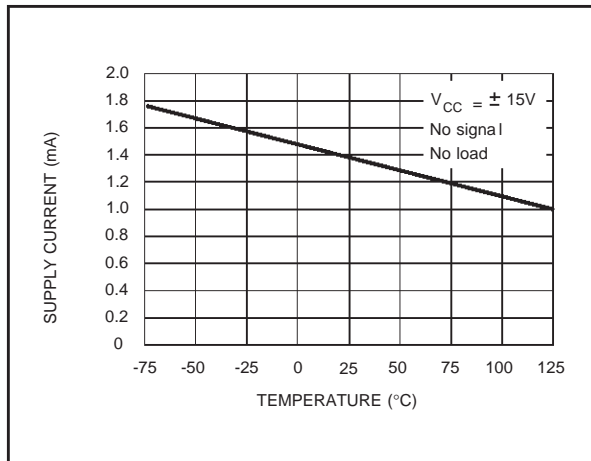
**LARGE SIGNAL DIFFERENTIAL VOLTAGE AMPLIFICATION AND PHASE SHIFT versus FREQUENCY**



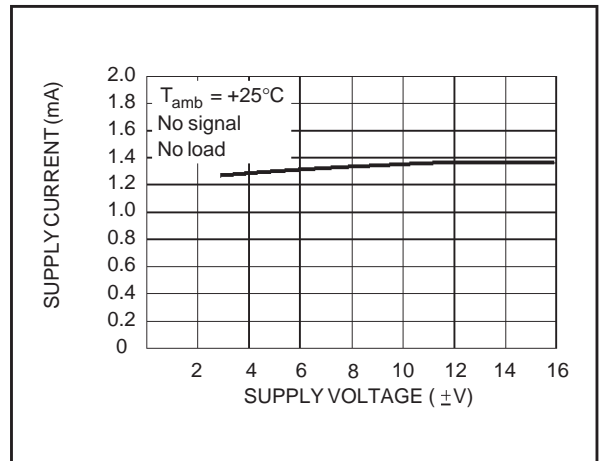
**TOTAL POWER DISSIPATION versus FREE AIR TEMPERATURE**



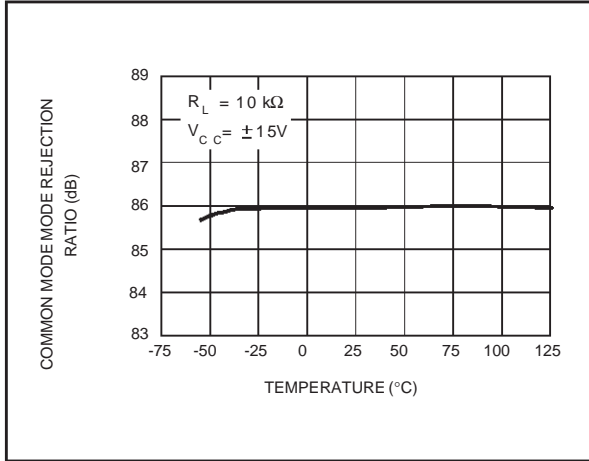
**SUPPLY CURRENT PER AMPLIFIER versus FREE AIR TEMPERATURE**



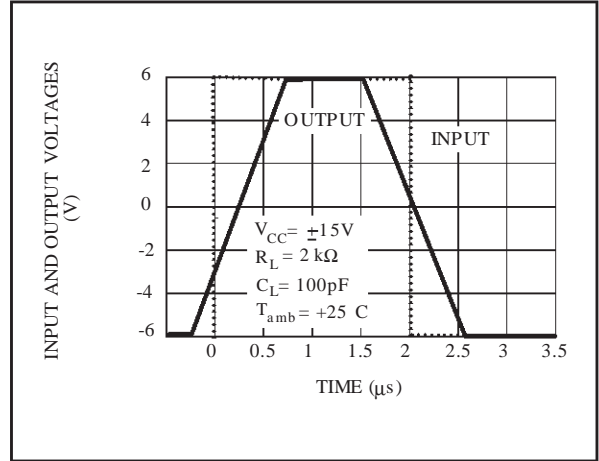
**SUPPLY CURRENT PER AMPLIFIER versus SUPPLY VOLTAGE**



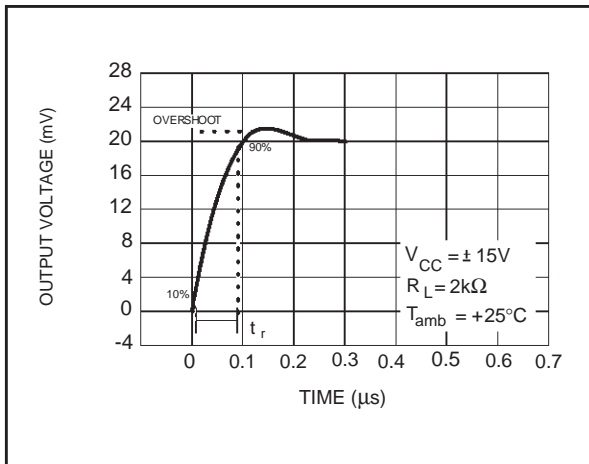
**COMMON MODE REJECTION RATIO versus FREE AIR TEMPERATURE**



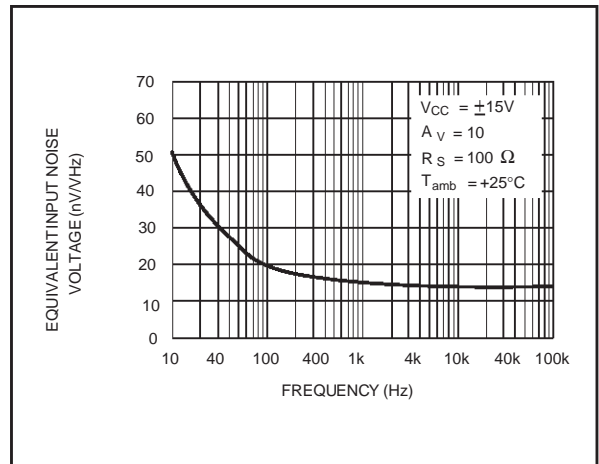
**VOLTAGE FOLLOWER LARGE SIGNAL PULSE RESPONSE**



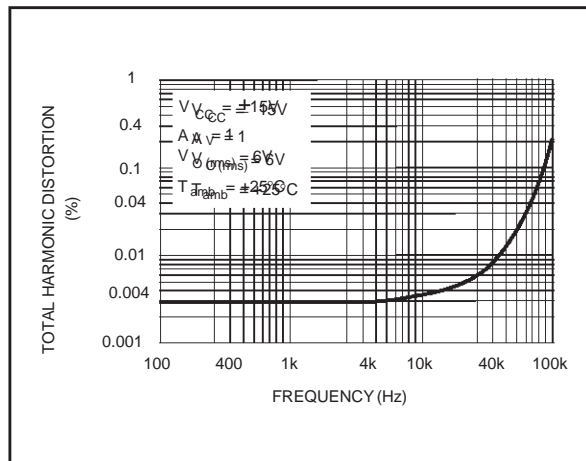
**OUTPUT VOLTAGE versus ELAPSED TIME**



**EQUIVALENT INPUT NOISE VOLTAGE versus FREQUENCY**



**TOTAL HARMONIC DISTORTION versus FREQUENCY**



PARAMETER MEASUREMENT INFORMATION

Figure 1 : Voltage Follower

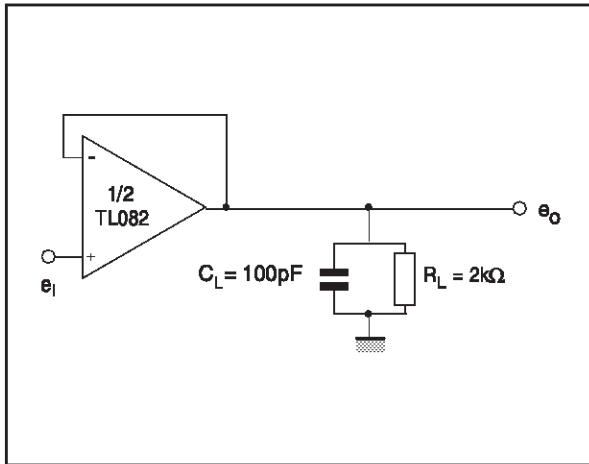
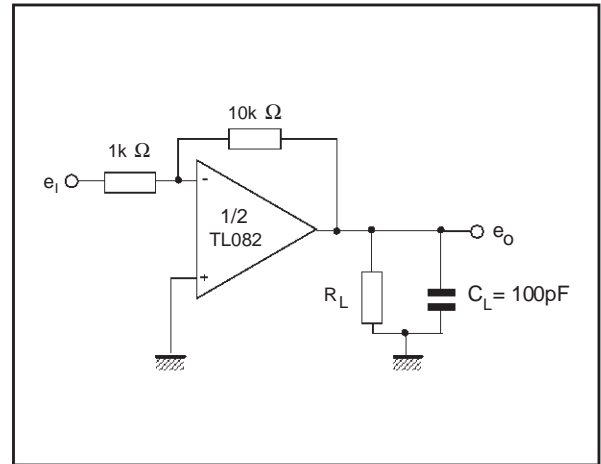
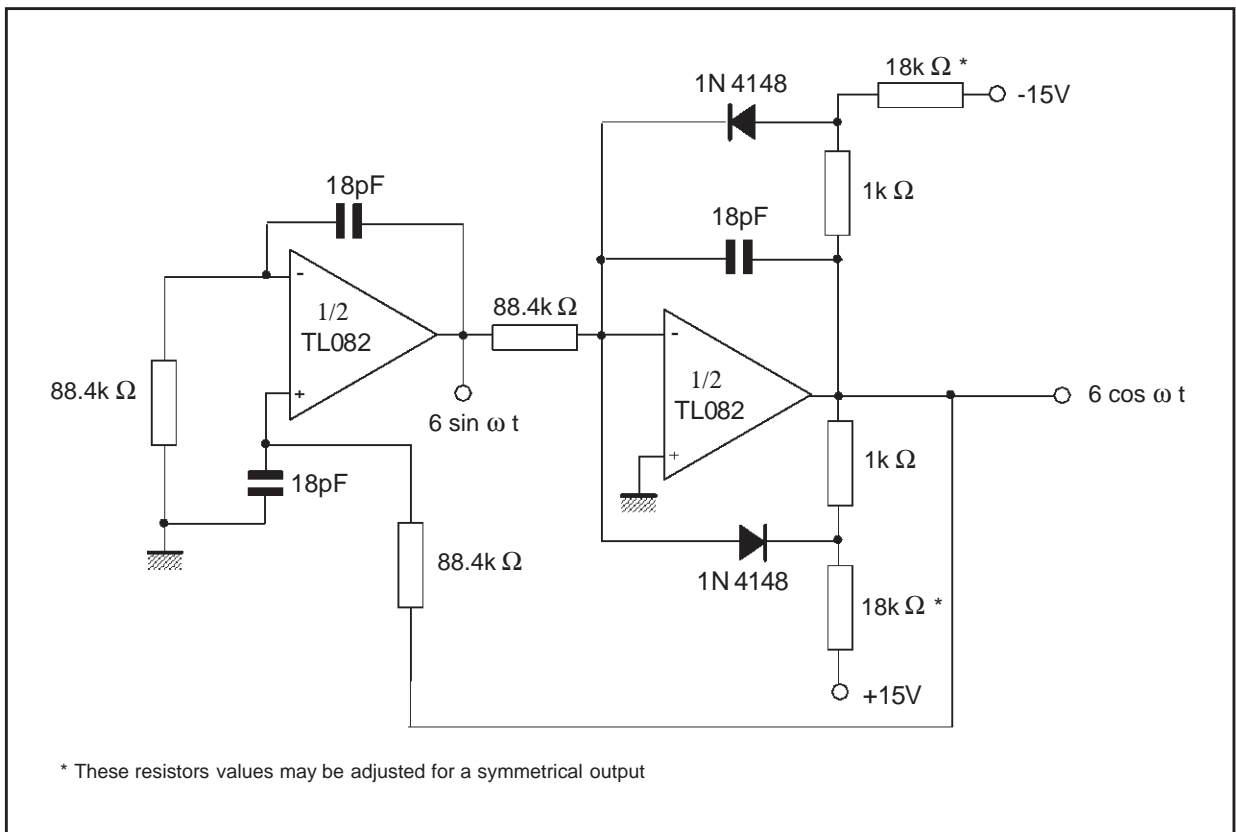


Figure 2 : Gain-of-10 Inverting Amplifier

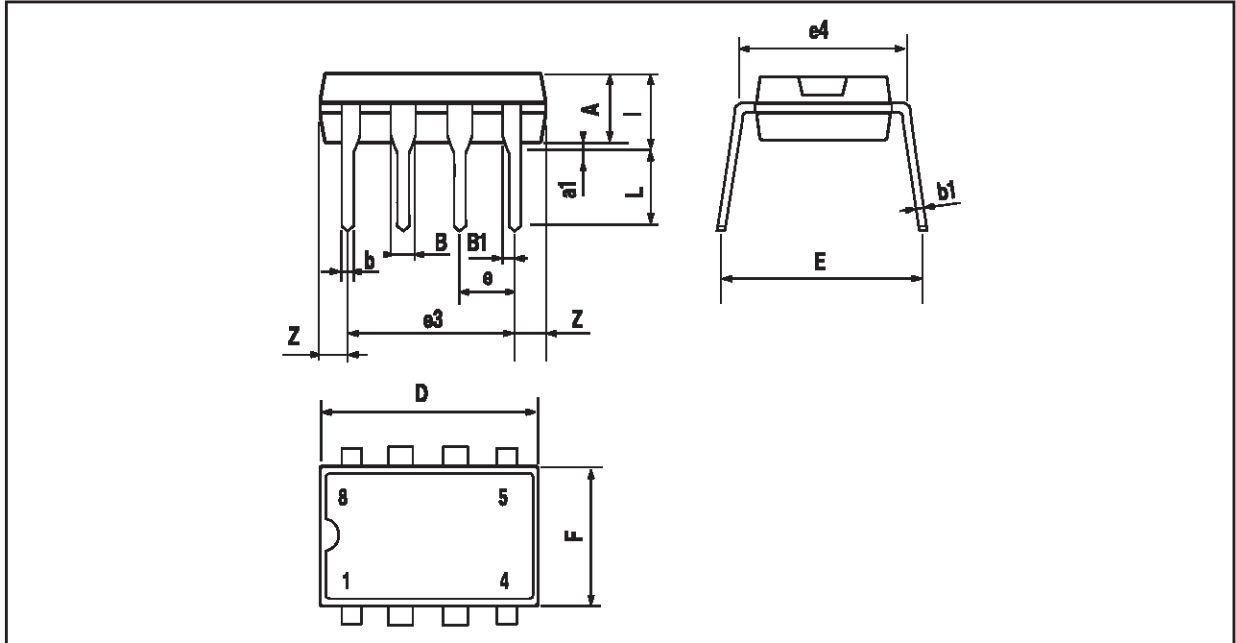


TYPICAL APPLICATIONS

100KHz QUADRUPLE OSCILLATOR



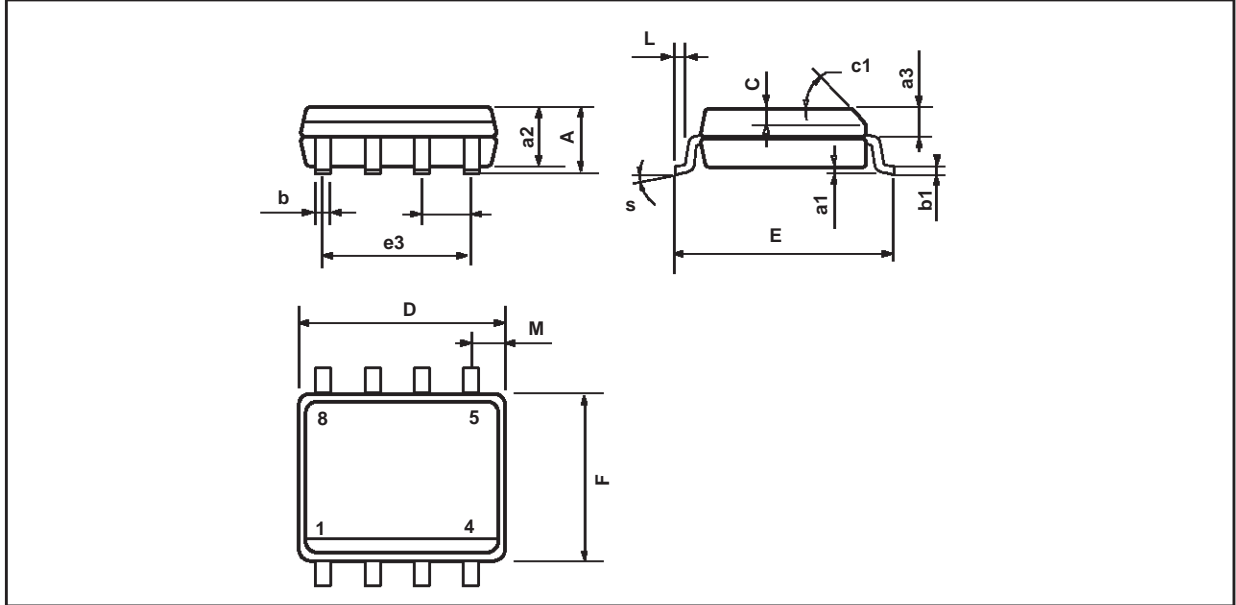
**PACKAGE MECHANICAL DATA**  
8 PINS - PLASTIC DIP



Dim.	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A		3.32			0.131	
a1	0.51			0.020		
B	1.15		1.65	0.045		0.065
b	0.356		0.55	0.014		0.022
b1	0.204		0.304	0.008		0.012
D			10.92			0.430
E	7.95		9.75	0.313		0.384
e		2.54			0.100	
e3		7.62			0.300	
e4		7.62			0.300	
F			6.6			0.260
i			5.08			0.200
L	3.18		3.81	0.125		0.150
Z			1.52			0.060

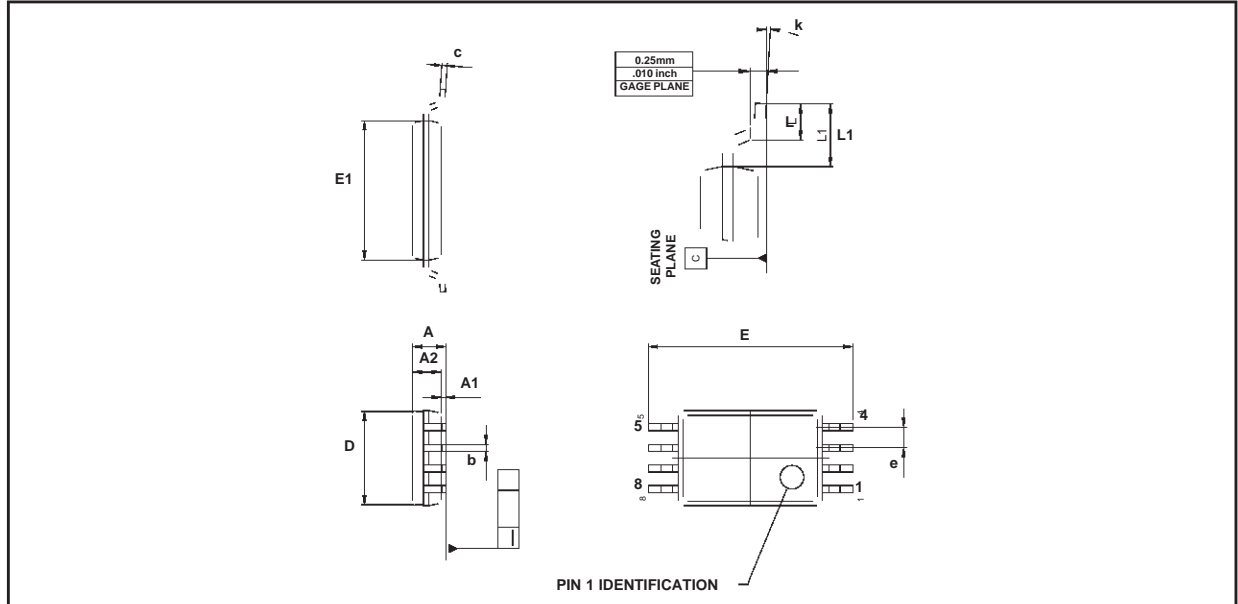
**TL082 - TL082A - TL082B**

**PACKAGE MECHANICAL DATA**  
8 PINS - PLASTIC MICROPACKAGE (SO)



Dim.	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A			1.75			0.069
a1	0.1		0.25	0.004		0.010
a2			1.65			0.065
a3	0.65		0.85	0.026		0.033
b	0.35		0.48	0.014		0.019
b1	0.19		0.25	0.007		0.010
C	0.25		0.5	0.010		0.020
c1	45° (typ.)					
D	4.8		5.0	0.189		0.197
E	5.8		6.2	0.228		0.244
e		1.27			0.050	
e3		3.81			0.150	
F	3.8		4.0	0.150		0.157
L	0.4		1.27	0.016		0.050
M			0.6			0.024
S	8° (max.)					

**PACKAGE MECHANICAL DATA**  
**8 PINS - THIN SHRINK SMALL OUTLINE PACKAGE**



Dim.	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A			1.20			0.05
A1	0.05		0.15	0.01		0.006
A2	0.80	1.00	1.05	0.031	0.039	0.041
b	0.19		0.30	0.007		0.15
c	0.09		0.20	0.003		0.012
D	2.90	3.00	3.10	0.114	0.118	0.122
E		6.40			0.252	
E1	4.30	4.40	4.50	0.169	0.173	0.177
e		0.65			0.025	
k	0°		8°	0°		8°
l	0.50	0.60	0.75	0.09	0.0236	0.030
L	0.45	0.600	0.75	0.018	0.024	0.030
L1		1.000			0.039	

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

© The ST logo is a registered trademark of STMicroelectronics

© 2001 STMicroelectronics - Printed in Italy - All Rights Reserved  
 STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - Canada - China - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia  
 Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States

© <http://www.st.com>





## Diploma thesis for Master degree in Technology

Spring 2021

for

**stud. tech. Bjarne Nilsen**

candidate no. 169868

Electrical Engineering at UiT campus Narvik

Solar angle optimizer

### Background



Fig. 1 Solar panel rig with two-axis orientation

The PV system on Fig. 1 is to be controlled by a PLC. The rig has two motor drives, one for the azimuth angle and one for the tilt angle. In order to obtain maximum power output, reference angles are inputted on the PLC. These reference angles should keep the panels perpendicular to the sun rays at any time.

The PLC is equipped with a time module which outputs UTM time data. An existing algorithm calculates the corresponding solar trajectory which provides the reference angles for the rig.

The output power will be measured, and through perturb-and-observe algorithms the system is seeking for the optimal angle, which may be affected by snow reflections etc.

The PLC program should include these algorithms, in addition to control of the motor drives, and the rig should also be equipped with position sensors, wind speed measurement and solar irradiation measurement.

The entire system should be installed and put into operation.

**General requirements:**

Results from the preliminary work is included in the master report introduction and other chapters in a natural way. Reports of work progress and discrepancy could be included as appendices to the main report.

The paper is edited as a research report with abstract, conclusion, literature list, references, table of contents etc. Statements should be based on proofs, references or logical reasoning.

In writing the report, the candidate should emphasize making the text easy to follow, well written and well documented.

Materials developed as part of the diploma work, such as software/source code or physical equipment, should be considered a part of the answer. Documentation for proper use of this software/equipment, should also be enclosed to the report, if possible.

The problem description should be enclosed to the report.

If the work is performed in cooperation with an external actor, the candidate should follow the current guidelines given by this, as well as possible further instructions from the actual business management.

The candidate is not allowed to intervene in the information systems, production systems etc of the external actor. If such an intervention should be relevant during the work accomplishment, special permission should be granted from the management.

Expenses such as travel expenses, copying and phone/fax expenses should be carried by the candidate himself unless other agreements are made.

If the candidate during the diploma work encounters difficulties which could not be foreseen during defining the work, and which possibly may lead to changes or deletion of certain problem issues, this should immediately be discussed with UiT through the supervisor.

The work should lead to a final report and forms the basis of evaluation and grading. The report with enclosed materials should be submitted electronically in Wiseflow. The submission page will appear as a "flow": <https://europe.wiseflow.net/>.

Standard title page should be used, available on UiT home page. The title page template, in addition to some useful links are to be found here:

<https://en.uit.no/student/oppgaveskriving>

Additionally it is referred to the diploma thesis subject Canvas page.

The report with enclosed materials should be submitted no later than the given submission date.

Release date: 11.01.2021

Submission date: 15.05.2021

Supervisor at UiT: Trond Østrem  
Phone: 76966250/99317710  
E-mail: [trond.ostrem@uit.no](mailto:trond.ostrem@uit.no)

Contact information: Candidate name: Bjarne Nilsen

E-mail: [bni037@post.uit.no](mailto:bni037@post.uit.no)