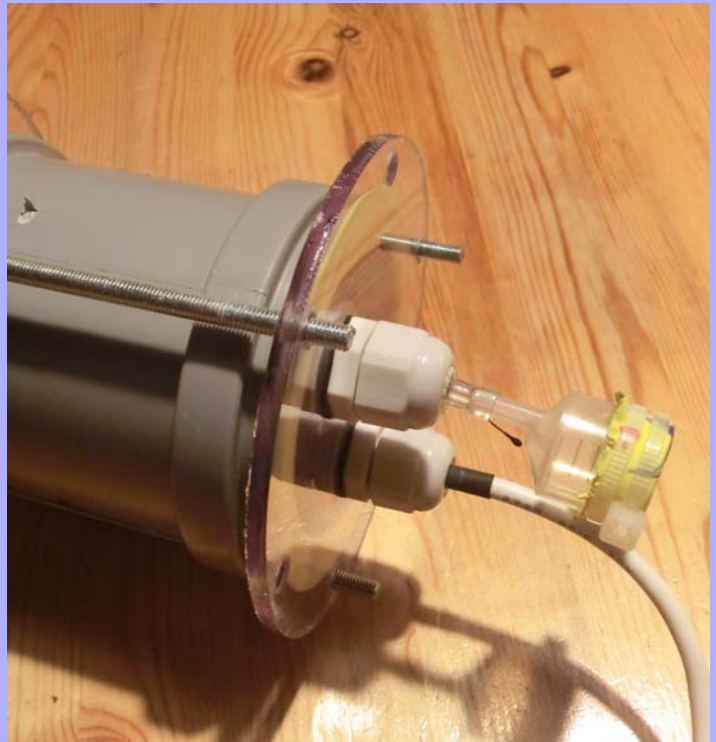
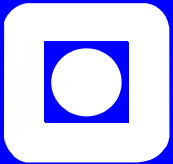


*Nils Kr. Rossing*

# ROV med trykk- og temperaturmåling



NTNU



Trondheim

Program for  
lærerutdanning

Skolelaboratoriet  
for matematikk, naturfag  
og teknologi

November 2017

Denne siden er blank

# ROV med trykk- og temperaturmåling

Nils Kr. Rossing

## **ROV - med trykk- og temperaturmåling**

Trondheim 2017

Layout og redigering: Nils Kr. Rossing, Skolelaboratoriet ved NTNU

Tekst og bilder: Nils Kr. Rossing, Skolelaboratoriet ved NTNU

Faglige spørsmål rettes til:

**Skolelaboratoriet for matematikk, naturfag og teknologi**

v/Nils Kr. Rossing, 73 55 11 91

[nils.rossing@ntnu.no](mailto:nils.rossing@ntnu.no)

Skolelaboratoriet ved NTNU

Realfagbygget,

Høgskoleringen 5,

7491 Trondheim

Telefon: 73 55 11 43

Telefaks: 73 55 11 40

<http://www.ntnu.no/skolelab/>

Rev 3.1 – 20.11.17



## Forord

Det undervisningsopplegget som er beskrevet i dette heftet er blitt til i et samarbeid mellom ToF koordinator Hilde Ervik (Skolelaboratoriet) og Håvard Holm ved Marintek i forbindelse med utviklingen av en enkel ROV for bruk i undervisningen av faget Teknologi og Forskningslære (ToF) i videregående skole.

Temaet “enkel ROV” ble først introdusert høsten 2015 som del av en av modulene til serien av ToF videreutdanningskurs ved NTNU. Høsten ble kurset holdt for andre gang og inkluderer en målestasjon for måling av trykk og temperatur. Høsten 2017 ble det så utviklet en prototyp av en komplett ROV som inkluderer måling av trykk og temperatur. Fortsatt preges dette undervisningsopplegget av at prototypingen ennå ikke er avsluttet. Også kurset høsten 2017 preges derfor av at utstyret er i en utforskningsfase. For et fag som ToF trenger ikke et slikt utgangspunkt være så dumt, siden også faget i seg selv skal ha et preg av utforskning. Vi ønsker derfor å inkludere deltagerne i prosessen fram mot et ferdig produkt.

Etter kurset 19. – 20. nov. 2016 ble det lagt ut et nytt shield-kort (versjon 2.0). Følgende endringer ble gjort:

- Kondensatoren  $C_1$  og  $C_2$  ble lagt til for å fjerne ev. støy inn på trykkmåleren fra den digitale delen av kretskortet. Dessuten ble ledningsføringen lagt noe om av samme grunn.
- Det ble også lagt ut et jordplan omkring trykkmåleren for å stabilisere jordpotensialet nær denne viktig komponenten. Årsaken var fortsatt å redusere støy inn på denne komponenten.
- Plasseringen av “ON” ved strømbryteren er rettet opp.

Til kurset i november 2017 ble det laget nok et nytt shield-kort (versjon 3.0) hvor det ble lagt inn stiftlister for å lett å kunne koble til tre servoer for motorer og Joy sticks for styring av motorene via kabel. Det ble også gitt mulighet for å koble opp separat strømforsyning med avkobling til servoene.

I tillegg er det bygget et nytt skrog for ROV'en og en terminal for styring med Joy sticks for uttesting under kurset.

Skolelaboratoriet ved NTNU

November 2017

Nils Kr. Rossing



6 ROV – med trykk- og temperaturmåling



## Innhold

<b>1</b>	<b>Innledning .....</b>	<b>9</b>
1.1	Gangen i konstruksjonsprosessen .....	9
1.2	Bakgrunn .....	10
1.3	Systemoversikt måleenhet .....	12
<b>2</b>	<b>Montering av enhet for datainnsamling og målejigg for kalibrering ....</b>	<b>14</b>
2.1	Byggebeskrivelse for datainnsamlingsenhet (shield-kort) .....	14
2.1.1	Montering av shield-kortet .....	15
2.1.2	Montering av NTC motstand (temperatursensor) .....	21
2.1.3	Montering av batteri .....	22
2.1.4	Montering av trykksensor – utvendig del .....	23
2.2	Byggebeskrivelse for målejigg versjon 2.0 .....	25
<b>3</b>	<b>Brukerveiledning .....</b>	<b>32</b>
3.1	Installasjon av programvaren (IDE) .....	32
3.1.1	Arduino programeditor .....	32
3.1.2	Programstruktur .....	35
3.2	Bruk av enhet for datainnsamling .....	35
<b>4</b>	<b>Litt sensorteori .....</b>	<b>38</b>
4.1	Kretsskjema for datainnsamlingsenheten (shield-kortet) .....	38
4.2	Temperatursensor (NTC og PTC -motstander) .....	38
4.3	Trykksensor (SPD030G) .....	43
4.4	Kalibrering og omregning .....	46
4.4.1	Prosedyre for kalibrering av temperatursensoren .....	46
4.4.2	Prosedyre for kalibrering av trykksensoren .....	48
<b>5</b>	<b>Bygging av skroget .....</b>	<b>51</b>
5.1	Komponentoversikt skrog med elektronikk .....	51
5.2	Byggebeskrivelse av skroget .....	52
5.3	Montering av rørmuffer for våt-testing .....	66
5.4	Byggebeskrivelse styringskonsoll .....	71
5.4.1	Systemoversikt med sensorer og motorkontroll .....	71
5.4.2	Brukerveiledning .....	71
5.4.3	Oppbygging av styringskonsoll .....	72
5.5	Uttesting av styrefunksjon .....	73
<b>6</b>	<b>Utfordringer ved uttesting .....</b>	<b>76</b>



---

6.1	Rapport fra arbeidet (5 - 8 sider, 15%) .....	76
6.2	Didaktisk refleksjonsdokument (2 - 3 sider, 15%) .....	77
<b>7</b>	<b>Referanser .....</b>	<b>78</b>
	<b>Vedlegg A Programvare .....</b>	<b>79</b>
A.1	Program for uttesting av sensorene .....	79
A.2	Program for uttesting av styringskonsollet og motorene .....	82
	<b>Vedlegg B 3D-printede deler .....</b>	<b>84</b>
B.1	Muffe mellom motor og aksling .....	84
B.2	Endestykke for propellaksling .....	84
B.3	Muffe mellom servomotor og gjengestang for tiltlodd .....	85
B.4	Eksentrisitet for tiltlodd .....	85
B.5	Brakett for kontrollerkort .....	86
B.6	Foring for å holde kretskort for motorstyring .....	87
	<b>Vedlegg C Kretskort .....</b>	<b>89</b>
	<b>Vedlegg D .....</b>	<b>91</b>
D.1	Design av endeplatene til målejigg .....	91
D.2	Design av ROV skrog .....	93
	<b>Vedlegg E Komponent- og utstyrlister .....</b>	<b>98</b>
E.1	Utstyrliste til datainnsamlingsenhet .....	98
E.2	Utstyrliste for målejigg: .....	98
E.3	Utstyrliste til ROV skrog .....	99





# 1 Innledning

## 1.1 Gangen i konstruksjonsprosessen

Forslag til rekkefølge i framstillingen og kalibrering av måleutrustning:

Dag 1

1. Systemoversikt – presentasjon (avsnitt 1.3 på side 12)
2. Bygging av shield-kort – verksted (avsnitt 2.1.1 på side 15)
3. Montering av batteri – verksted (avsnitt 2.1.3 på side 22)
4. Kort innføring i sensorteori – presentasjon (kapittel 4, side 38)
5. Bygging av temperatursensor – verksted (avsnitt 2.1.2 på side 21)
6. Bygging av ekstern trykksensor – verksted (avsnitt 2.1.4 på side 23)
7. Innlasting av kalibreringsprogram – verksted (side 36)
8. Testing av temperatur- og trykksensor – verksted (side 36)
9. Kalibrering av temperatursensor – verksted (avsnitt 4.4.1 på side 46)
10. Testing av skriving til SD-kort – verksted (avsnitt 3.2 på side 35)
11. Montering av testjigg – verksted (avsnitt 2.2 på side 25)
12. Lekkasjetest med kabel og ekstern trykksensor – verksted (side 31)
13. Kalibrering av dybdemåler – verksted (avsnitt 4.4.2, side 48)

Etter bygging av sensorkortet deles gruppen i to. Den ene gruppen arbeider videre med en ROV

Etter at sensorene er testet og kalibrert, demonteres testjiggen og Arduinokort med shield-kort og sensorer tas ut. De skal nå bygges inn i ROV'en. På grunn av begrenset tid, har vi valgt ikke å inkludere sensorene når vi skal forsøke å kjøre den endelige ROV'en i vann. Årsaken er at det lett blir for mange faktorer å ta hensyn til samtidig. En slik integrering er det naturlig å gjøre med elevene i klasserommet.

14. Oppbygging av skroget til ROV uten rørmuffer (avsnitt 5.2 på side 52)
15. Testing og styring av motorer (avsnitt 5.5, side 73)

### Dag 2 - Det er imidlertid noe uklart hvor langt vi kommer dag 1

16. Montering av rørmuffer for våt-testing (avsnitt 5.3 på side 66)
17. Lekkasjetesting (ikke beskrevet i heftet)
18. Flytetesting og avbalansering i vannet (ikke beskrevet i heftet)
19. Testing av tilt-mekanisme for å endre dybde (ikke beskrevet i heftet)
20. Testing av motorer og framdriftssystem i vann (ikke beskrevet i heftet)



## 1.2 Bakgrunn

### Minimumversjon med syltetøyglass

Ett utgangspunkt er at det skal utvikles en farkost som kan bevege seg og gjøre enkle målinger under vann, utstyret skal være billig og lett tilgjengelig for de skolene som ønsker å bruke dette opplegget i egen undervisning. For det enkleste alternativet ble det derfor anvendt et syltetøyglass med påmonterte motorer. Motorene var rimelige demonterte servomotorer som ble montert på utsiden av syltetøyglasset, åpent i vannet, mens styreelektronikken var plassert inne i glasset. Dette kan gå bra en stund i rent ferskvann, men børstene i motorene vil etter hvert korrodere og gi problemer. På sikt er det derfor aktuelt å finne mer permanente løsninger hvor motorene beskyttes bedre. Bildene under er hentet fra kurset holdt høsten 2015.

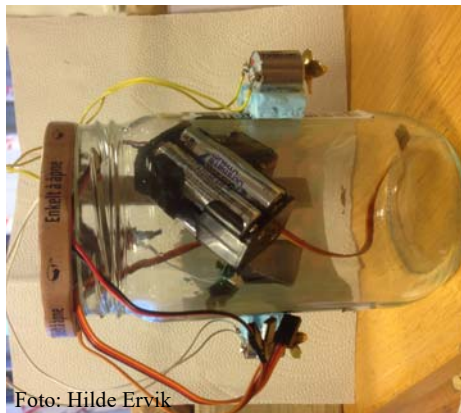


Foto: Hilde Ervik



Foto: Hilde Ervik

For å justere helningsvinkelen på farkosten ble det benyttet en servo hvor selve drivarmen var montert til en brakett montert til lokket. Siden servoer har stor utveksling kan de yte en betydelig kraft. I denne sammenhengen nok til å vippe på et messinglodd slik at tyngdepunktet i farkosten forskyver seg slik at den tiltes i den ene eller andre retningen. Dette er vist på bildet til høyre.

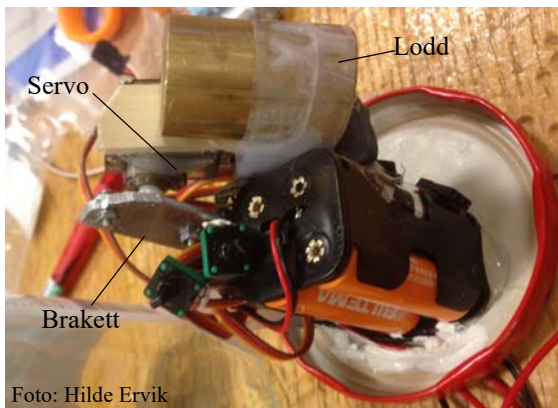


Foto: Hilde Ervik

### En mer avansert farkost

For kurset høsten 2017 ble utviklet ulike ROV farkoster fra de helt enkle til mer avanserte konstruksjoner. Heftet beskriver en farkost som er blitt utviklet med følgende spesifikasjoner:

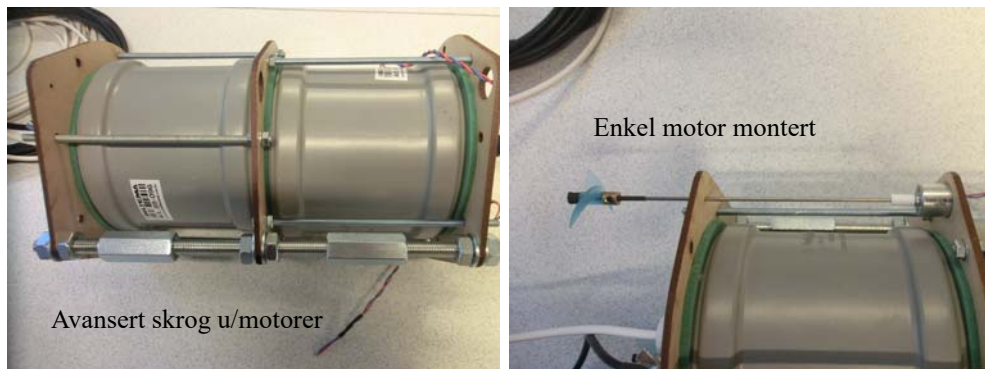
- Relativt rimelig med lett tilgjengelige deler
- Skal kunne lese av måledata mens ROV'en er operativ under vann
- Skal kunne styres ved hjelp av kabel som på sikt gir større muligheter for å hente opp større datamengder

10 ROV – med trykk- og temperaturmåling



- Helningsvinkelen på farkosten skal kunne styres for å kunne gå til overflaten eller ned mot bunnen
- Den skal ha innebygget batteri slik at den kan operere uten USB-kabel.

Bildet under viser en mer avansert ROV, dog fortsatt meget enkel, laget av to rørmuffer.



### Innholdet i kurset

Kurset beskriver mekanisk oppbygging av ROV'en og virkemåten til en trykk- og en temperaturmåler. Måleutstyret er bygget opp omkring en Arduino UNO med mulighet for lagring av data på SD-kort. Det samme kortet styrer også motorene.

Bakgrunnen for at vi har valgt denne løsningen er sammensatt:

- Arduino-teknologien begynner å bli ganske utbredt på skolene og er ikke særlig kostbar
- Synergier med beslektede prosjekter (f.eks. CanSat) som benytter samme plattform
- Løsningen blir svært fleksibel og kan bygges ut med ulike sensorer
- Store mengder data kan lagres på SD-kort
- Dimensjonene og strømtrekk for måledelen kan gjøres små om det er ønskelig
- Arduino'er av denne typen kan også styre servoer og mindre små servomotorer direkte uten ekstra drivere.
- Bruk av rørmuffer og gjengestenger gir en billig og fleksibel løsning
- Store muligheter til differensiering, fra at man kan bruke ferdig ROV'skrog, programvare og elektronikk, til at elevene selv bygger skroget, elektronikken og lager selv programmet til Arduino-kortet
- Styring med kabel gir mulighet til å overføre store mengder realtime data fra ROVen.

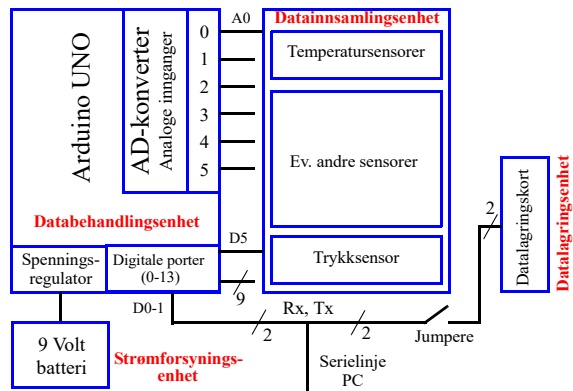
Heftet går grundig gjennom oppbygging, testing og kalibrering av en trykk- og temperaturmåler basert på en Arduino UNO. Videre beskrives i detalj en ROV bygget opp av to rørmuffer og en indre og en ytre struktur som holder skroget samlet.



### 1.3 Systemoversikt måleenhet

Måleenheten består hovedsakelig av fire deler:

1. Databehandlingsenhet  
(plattform - Arduino UNO)
2. Datainnsamlingsenhet  
(nyttelast med til hørende sensorer)
3. Datalagringseenhet  
(datalogger for lagring av data)
4. Strømforskyningseenhet  
(batteri med spenningsregulator)

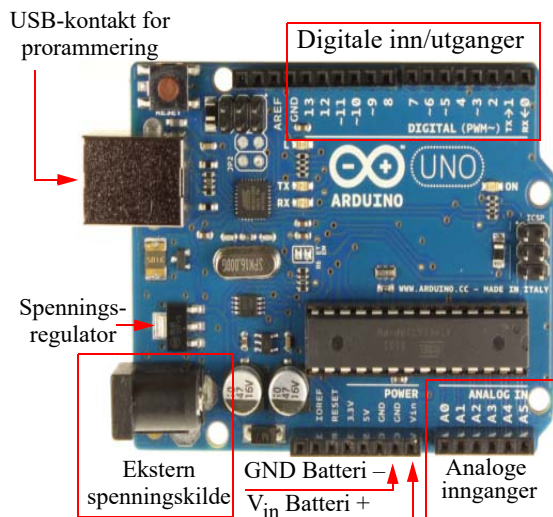


Blokkdiagrammet til høyre viser hvordan de ulike enhetene er forbundet med hverandre.

Databehandlingsenheten består av en tradisjonell Arduino UNO, som er ett av de mest populære i mikrokontrollerkortene i Arduino-familien, og dermed leveres til en overkommelig pris (www.kultogbillig.no pris kr. 129,- inkl. MVA + frakt).

Kortet er bygget opp omkring Atmel prosessoren ATmega328P med en klokkefrekvens på 16 MHz og et flash lager på 32 kbyte, SRAM 2 kbyte og EEPROM 1 kbyte. Kortet har dessuten:

- **Digitale I/O-porter**  
Kortet har 14 digitale inn/utporter (I/O-porter) som alle kan programmeres til enten å være inn- eller utganger. Seks av disse (3, 5, 6, 9, 10 og 11) kan *pulsbreddemoduleres* (pwm), disse er merket med ~ på kortet. Maksimal strøm for hver av I/O portene er 40 mA.
- **Analoge innganger**  
Kortet har 6 analoge innganger som kan registrere analoge spenninger fra 0 – 5 V, med en nøyaktighet (oppløsning) på ca. 5 mV.
- **USB-kontakt** for direkte tilkobling av PC og programmering av kortet. Når USB-kabelen er tilkoblet tilfører den kortet spenning slik at det er unødvendig med batteri. Dersom denne belastes med mer enn 500 mA vil strømforsyningen bli brutt inntil strømtrekket reduseres under denne grensen.





---

- **Strømtilførsel**

Tilkoblingspunkter for batterieliminatør, anbefalt spenning 7 – 12 V (grenseverdier 6 – 20 V). Batteriet kan enten tilkobles eliminatorpluggen (2.1 mm og + senter) eller via  $V_{in}$  (+) og GND (-). 5V utgangen leverer spenning til f.eks. datainnsamlingskortet<sup>1</sup>. Enkelte komponenter trenger lavere spenning som ev. kan leveres fra 3.3 V utgangen.

- **Reset**

Kortet inneholder en RESET-knapp som starter programmet på nytt

Kantkontaktene er montert slik at tilleggskort kan monteres rett ned på Arduino-kortet, også kalt “skjold-kort” eller “shield-kort”.

---

1. I dette tilfellet kobles batteriledningen til shield-kortet som er forbundet til  $V_{in}$  via en glidebryter og stiftkontaktene.



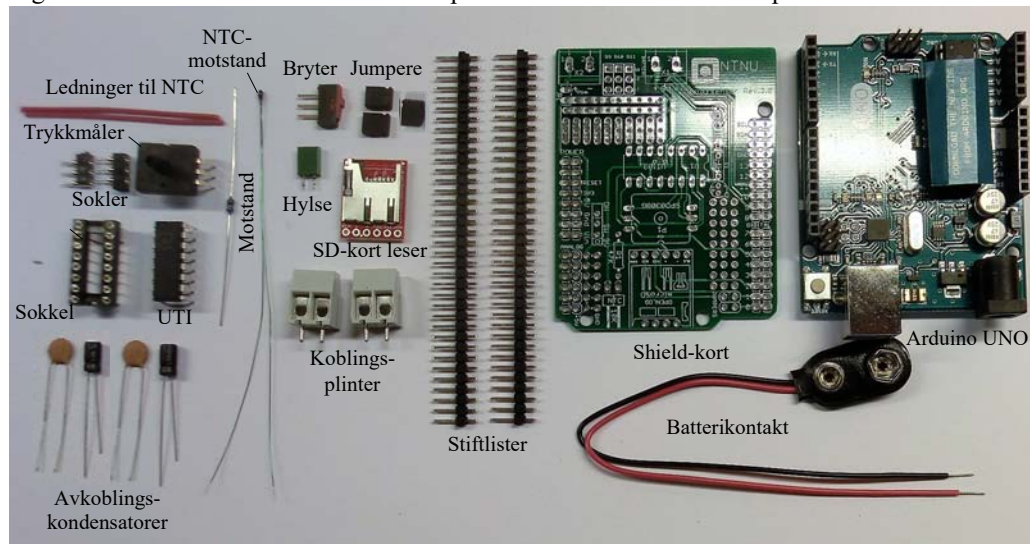
## 2 Montering av enhet for datainnsamling og målejigg for kalibrering

Dette kapittelet beskriver i detalj hvordan enheten for datainnsamling, målejigg og trykksensor kan bygges opp.

### 2.1 Byggebeskrivelse for datainnsamlingsenhet (shield-kort)

Avsnittet gir en kort beskrivelse av monteringen av ROV shield-kortet.

Figuren under viser en oversikt over komponentene som skal monteres på shield-kortet.



#### Komponentliste - shield-kort versjon 3.0 (se også vedlegg D)

- 1 stk Arduino UNO
- 1 stk. shield-kort versjon 3.0
- 1 stk. trykksensor SPD030G
- 1 stk. universal transduser interface UTI Smartec
- 1 stk SD-kortleser, OpenLog Sparkfun
- 1 stk. sokkel 16 DIL
- 1 stk. sokkel 6 DIL (2 x 3)
- 2 stk stiftlist, 36 pin
- 1 stk. hylselist - 2 polt
- 1 stk. glidebryter
- 1 stk motstander 51 kOhm
- 1 stk NTC-motstand – 50 kOhm
- 1 stk batterikontakt 9 V
- 2 stk koblingspilint, 2 polt
- 2 stk kondensatorer 100 nF
- 2 stk kondensatorer 10  $\mu$ F
- 3 stk jumpere
- Diverse ledning 0,6 mm

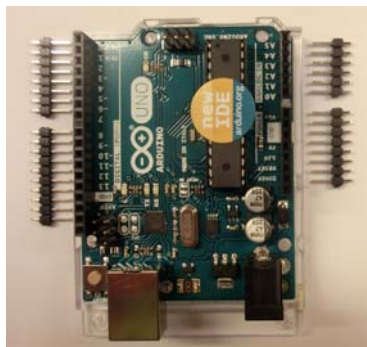
14 ROV – med trykk- og temperaturmåling



## 2.1.1 Montering av shield-kortet

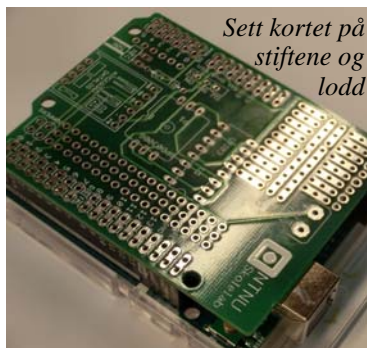
### 1. Montering av stiftlister

*Stiftlistene forbinder shield-kortet med Arduino-kortet. Kutt opp stiftlisten i riktige lengder (10 pinner, 2 x 8 pinner og 1 x 6 pinner) og plasser dem i hylsekontaktene på Arduino-kortet med den lange enden ned i hylselista.*

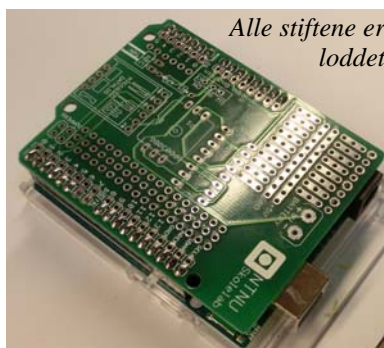


### 2. Plasser shield-kortet<sup>2</sup>

*Plasser ROV shield-kortet på stiftene som stikker opp, slik alle stikker så vidt opp over kanten av shield-kortet. Lodd fast alle pinnene i hver rad på oversiden (komponentsiden) av shield-kortet.*



Sett kortet på stiftene og lodd



Alle stiftene er loddet

---

2. Noen av bildene viser shield-kort versjon 2.0 og noen versjon 3.0. Det kan derfor være noe forskjeller på kortene på bildene.



3. Lag en stiftlist med to stifter og klipp av stiftene på siden der de er lengst. Disse skal være en støtte for SD-kortleseren og skal monteres helt forrest der SD-kortholderen skal stå, som vist på figuren under.

Lag en stiftlist med to stifter.  
Klipp av stiftene på den lange siden



Monter dem på kanten av kortet som vist

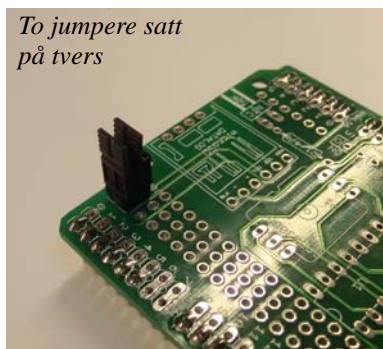


4. Lag to stiftlister med to stifter hver. Sett to på jumpere som vist på figuren til høyre (**MERK:** Jumperne skal normalt plasseres den andre veien) for å holde stiftlistene i riktig posisjon og lodd deretter fast stiftlistene på undersiden (**MERK:** Jumperne kan ha noe forskjellig utforming).

Lag to stiftlister a to stifter hver.



To jumpere satt på tvers



5. Lag en stiftlist med seks stifter. Stikk de lange endene gjennom hullene på SD-kortholderen, slik at de lange endene stikker opp over oversiden av kortet. Pass på at stiftene står vinkelrett på kortet og lodd på loddessiden. Om det er vanskelig å få pinnene til å stå rett, så lodd en pinne først for så å justere denne når du ser hvordan det blir.

Lag en stiftlist med seks stifter



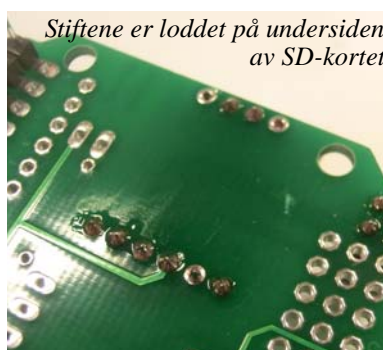
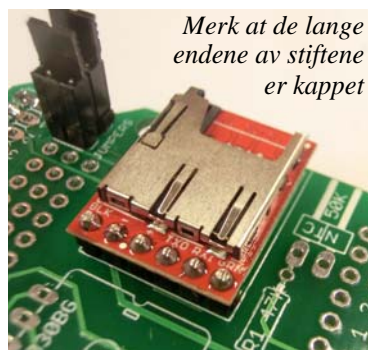
Monter stiftene langs kanten av kortet som vist



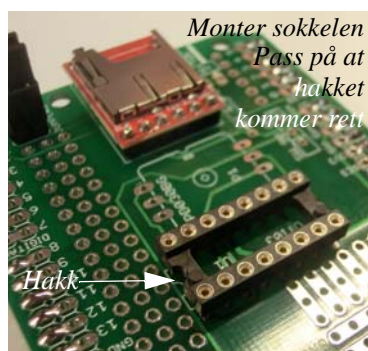




6. De korte endene av de seks stiftene loddes til kortets underside (loddesside). De lange stiftene på toppen kuttes med en sideavbiter



7. Monter en 16-pins sokkel som vist til venstre på figuren under. Sørg for at **hakket** kommer på rett side. Lodd på undersiden.
8. Monter glidebryteren som vist til høyre på figuren under. Hensikten med bryteren er å kunne slå av og på strømmen fra batteriet. Det spiller ingen rolle hvilken vei den monteres.

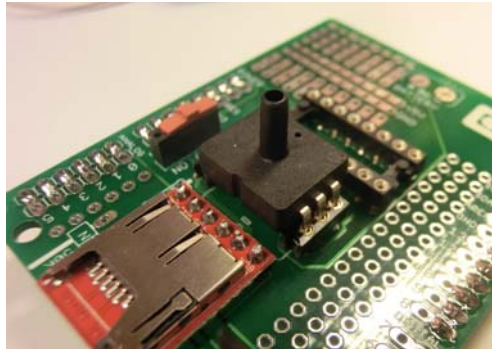


9. Sett de to tre-pinns sokkene på beina til trykksensoren som vist til høyre på figuren under. Pass på å rette opp beina på sensoren slik at de er parallelle, som vist til høyre.





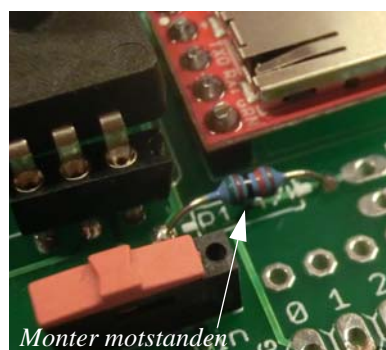
10. Plasser trykksensoren med hylsene i hullene på kortet som vist i figuren under, og lodd.



11. Sett IC-kretsen (UTI) i sokkelen. Før IC-en plassers, kan det være lurt å gjøre beina parallelle. Dette kan gjøres ved å legge kretsen på siden og presse den forsiktig ned mot bordet med tommelen som vist til venstre i figuren under.

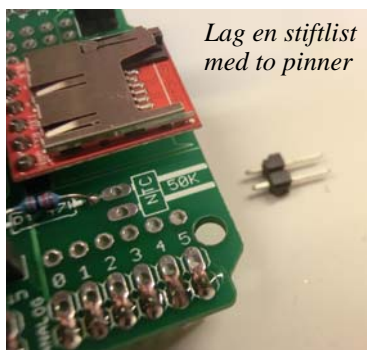


12. Monter motstanden som vist til høyre på figuren under. Bøy beina slik at avstanden mellom beina passer til hullene på kortet. Vi har valgt å bruke en motstand på 51 kOhm i stedet for 10 kOhms som er vist på bildet.

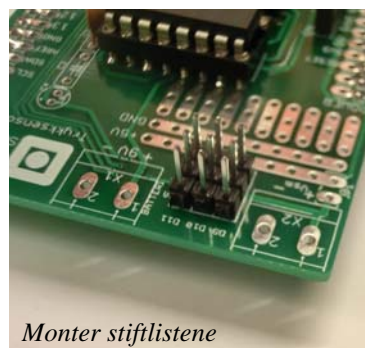




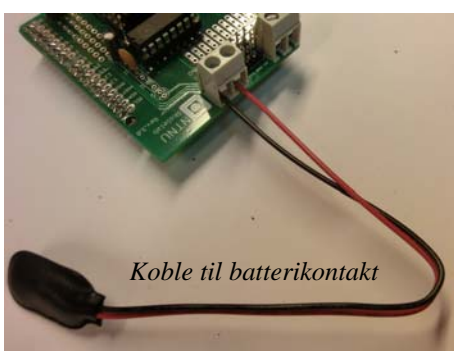
13. Lag en stiftlist med to pinner. Den skal plasseres som vist til høyre på figuren og være tilkoblingen for NTC-motstanden (temperatursensoren).



14. Klipp opp tre stiftlister a tre stifter og monter dem side om side som vist på bildet til høyre under.

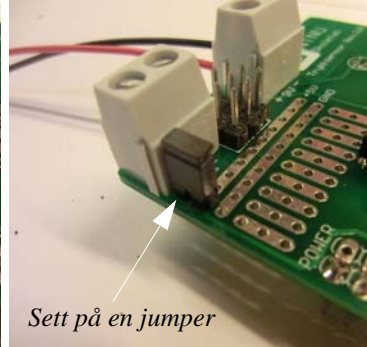
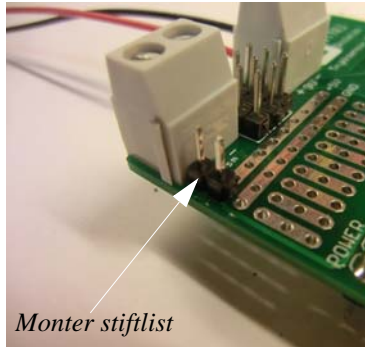


15. Monter koblingsplintene som vist på bildet under til venstre og koble til batterikontakten. De to koblingsplintene gjør det mulig å koble separat spenningskilde til servomotorene. Denne kobles da til koblingsplinten til høyre. Påse at rød ledning kobles til pluss og sort til minus som vist på bildet under.

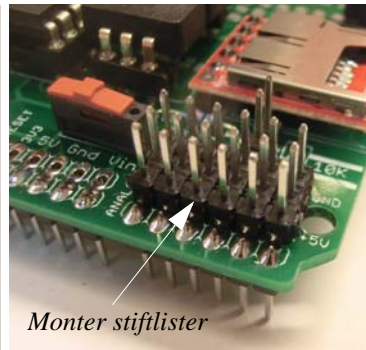




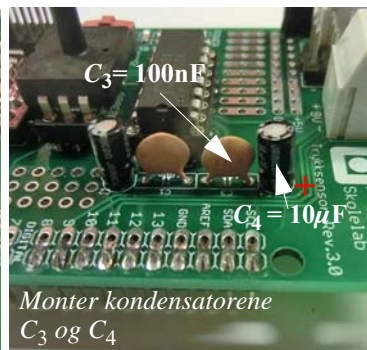
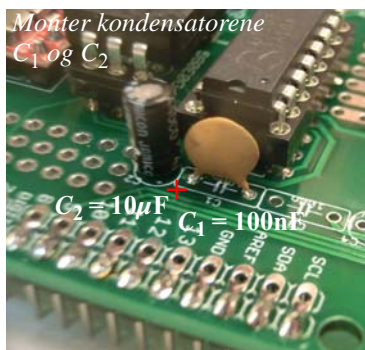
16. Klipp til en stiftlist på to kontakter og monter den bak koblingsplinten som vist til venstre på bildet under. Dersom man ønsker å bruke samme spenningskilde til både sensor kortet og til servoene setter man på en jumper som vist på bildet til høyre.



17. Monter stiftlister for tilkobling av sensorer og Joy sticks. Klipp til tre stiftlister hver med 6 stifter og monter dem i hjørnet av kortet nær SD-kortleseren som vist på bildet under til høyre.



18. Monter avkoblingskondensatorene  $C_1 = 100\text{nF}$  og  $C_2 = 10\mu\text{F}$ . Hensikten er å fjerne støy fra kortet. Pass på polariteten på  $C_2$ . Det lange beinet på kondensatoren skal plasseres nær + (se bildet under til venstre).

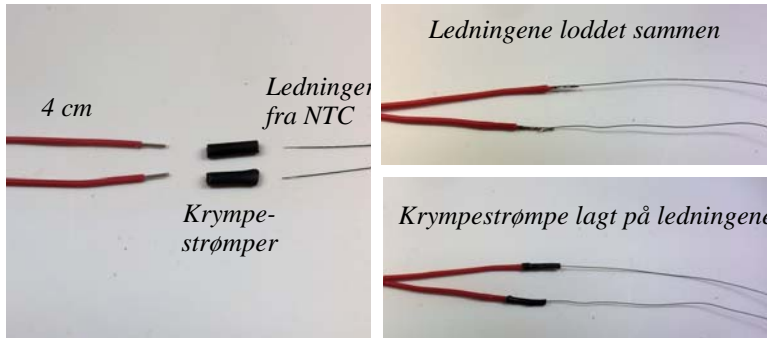


Monter avkoblingskondensatorene  $C_3 = 100\text{nF}$  og  $C_4 = 10\mu\text{F}$ . Pass på polariteten på  $C_4$ .



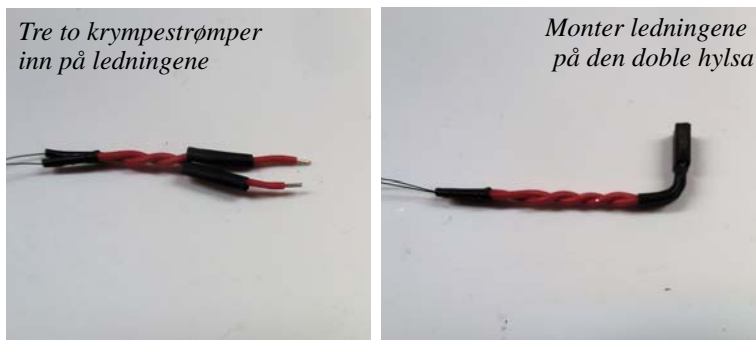
## 2.1.2 Montering av NTC motstand (temperatursensor)

19. Avisoler endene til to 4 cm lange ledninger som vist til venstre på figuren under. Lag to krympestrømper med lengde ca. 10 mm og diameter 1,5 mm. Lodd ledningene til endene til NTC-motstanden. Tre krympestrømpene på ledningen og plasser den over skjøten og varm opp.

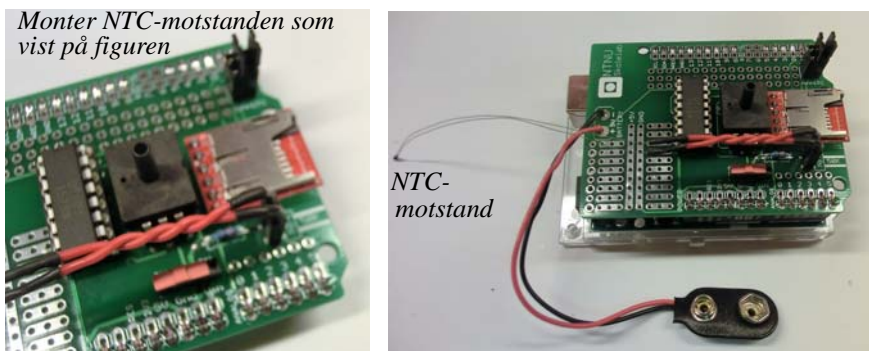


Det kan være tilstrekkelig å varme opp med loddekolben.

20. Tvinns sammen ledningene og la det være litt plass på endene slik at en kan tre to nye krympestrømper inn på ledningen. Pass på at krympestrømpene ikke kommer for nær loddepunktene, i så fall vil de krympe i gal posisjon. Lodd endene til en dobbel hylse og tre krympestrømpene ned over loddene som vist til høyre på figuren.



21. Plasser NTC-motstanden på stiflista som vist på figuren under.

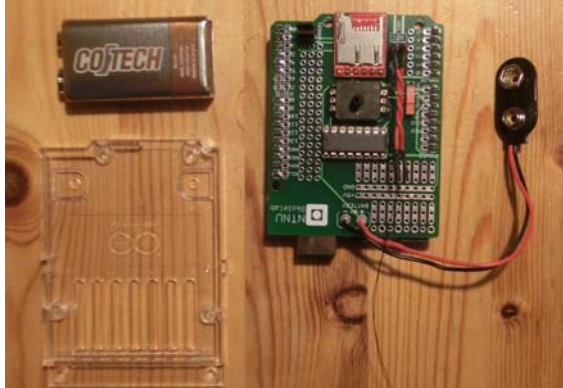




22. Shield-kortet er ferdig og klart til å plasseres på “ryggen” til Arduino-kortet.

### 2.1.3 Montering av batteri

23. Arduino-kortet leveres med en plastholder som egner seg godt som feste for batteriet (MERK: Bildet viser det gamle shield-kortet, versjon 1.0).

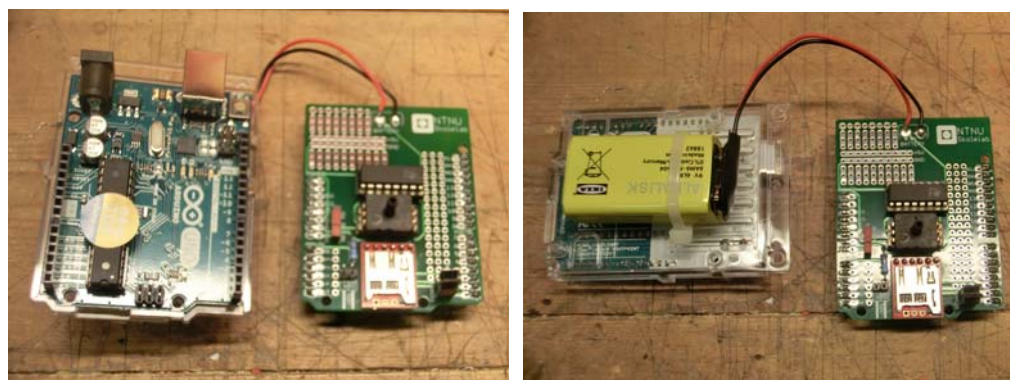


Batteriet festes med en strips til slissene i plastholderen. Pass på at stripsens stramme-anordning plasseres på siden av batteriet (se figuren under til høyre), da den ellers kan komme i konflikt med veggene i muffa når den skal monteres.

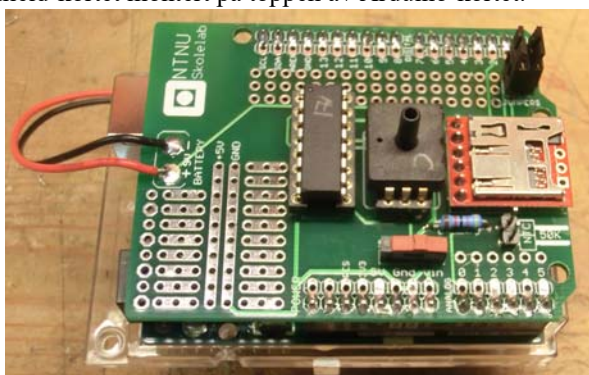


## Det ferdige shield-kortet

Bildene under viser det ferdige shield-kortet og Arduino-kretsen. Batteriet er montert på baksiden av Arduino-kortet som leveres med en plasholder.



Bildet under viser shield-kortet montert på toppen av Arduino-kortet.



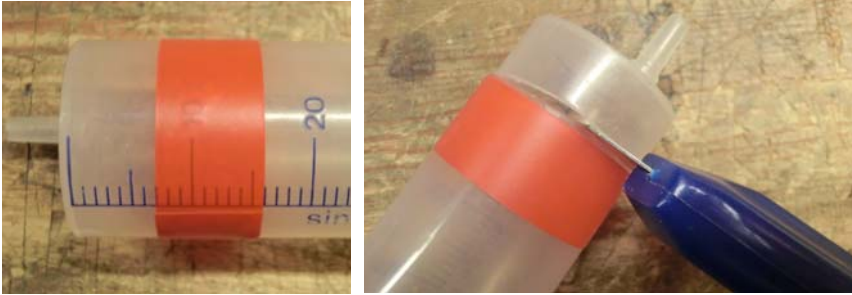
### 2.1.4 Montering av trykksensor – utvendig del

Her skal vi se hvordan vi kan lage trykksensorens utvendige del. Den består av framdelen av en sprøyte (50 ml), en plastslange (Ø5 mm), en ballong og to plaststrips (15 cm).

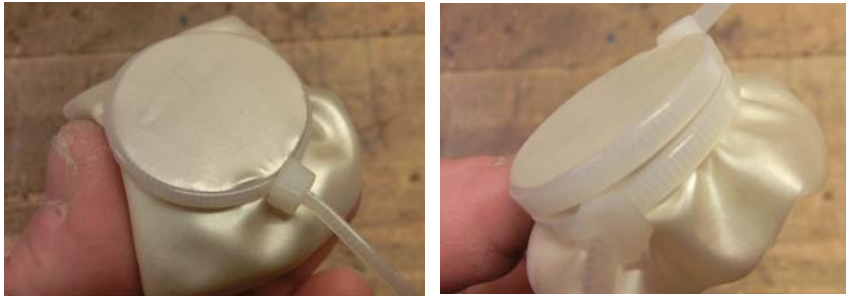
Sprøyta kappes ca. 1,5 – 2 cm fra stedet hvor spissen begynner. Pass på å bruke en sprøyte som har en spiss som er liten nok til å kunne presses inn i slangen. Puss eller bruk varmepistol på den sirkulære kutt-kanten slik at skarpe kanter etter beskjæringen fjernes. Ved å bruke en så stor sprøyte som mulig, blir diameteren på ballongflata så stor som mulig. Det er rimelig å anta at følsomheten øker med økende areal av ballongflata. Samtidig bør kammeret gjøres så lite som mulig slik at volumet av luftrommet over sensoren blir så lite som mulig, hvilket også synes å påvirke følsomheten. Dette kan være to motstridende krav, slik at det må gjøres et kompromiss.



Det er viktig at kuttet i sprøyta er rett og rent, dvs. uten for skarpe kanter. Ved å legge en tape rundt sprøyta kan en sikre seg at kuttet blir rett. Dessuten kan en gjøre snittet mindre skarpt ved å varme opp med varmpistol, eller bruke en fin fil



Klipp sideflatene ut av en ballong og legg en av sidene stramt over den overskårne enden av sprøyta og fest den med en strips rundt sylindren. Bruk to strips og pass på at festeordningen til stripsene legges på hver sin side av sylindren. Dette er viktig for å unngå lekkasje av vann. Stripsene strammes godt.



Koble til slangen.







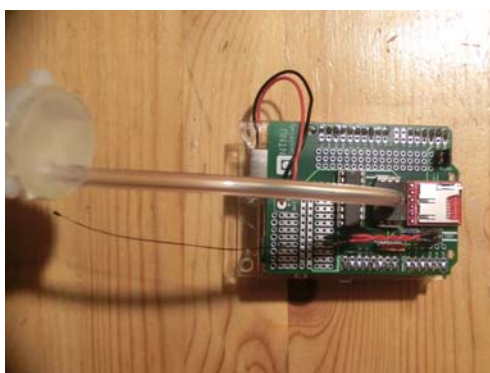
Sjekk at ballongkammeret er tett ved å blåse i slangen mens kammeret er senket ned i vann. Boller vil stige opp fra kammeret dersom det er utett.



Senkes under vann,  
bruk ev. en lengre slange ...



Når kammeret er tett, klippes overflødige ender av stripsen bort og ballongkammeret kan kobles til sensoren på kortet.



## 2.2 Byggebeskrivelse for målejigg versjon 2.0

Denne målejiggen tillater bruk av en USB-kabel som føres gjennom veggen i kammeret, i tillegg til ballongsensoren for trykksensoren og NTC-motstanden som måler temperaturen.



1. Figuren under viser hva som trengs for å lage en målejigg av denne typen. En liten skiftnøkkel og en rørpumpetang kan også være greit å ha (komponentliste se vedlegg E.2, side 98).



### Komponentliste - målekammer

- 3 stk. gjengstang 3 stk. Ø 5 mm a 130 mm
- 3 stk. vingemuttere, M5
- 6. stk. muttere, M5
- 9 stk skiver, M5
- 2 stk. kabelforskruinger, 5 – 8 mm, M16
- 1 stk. avløpsrør skjøt (muffe) Ø 110 mm
- 1 stk sprøyte 50 ml for tilkobling av slange
- 1 stk USB-kabel 2.0, A – B, 1,8 m eller lengre
- 1 ark med mosegummi 1 mm
- 1 stk plastslange 135 mm, Ø 3 innvendig
- 2 stk runde plater, skåret i 3 – 4 mm akryl
- 1 stk ballong

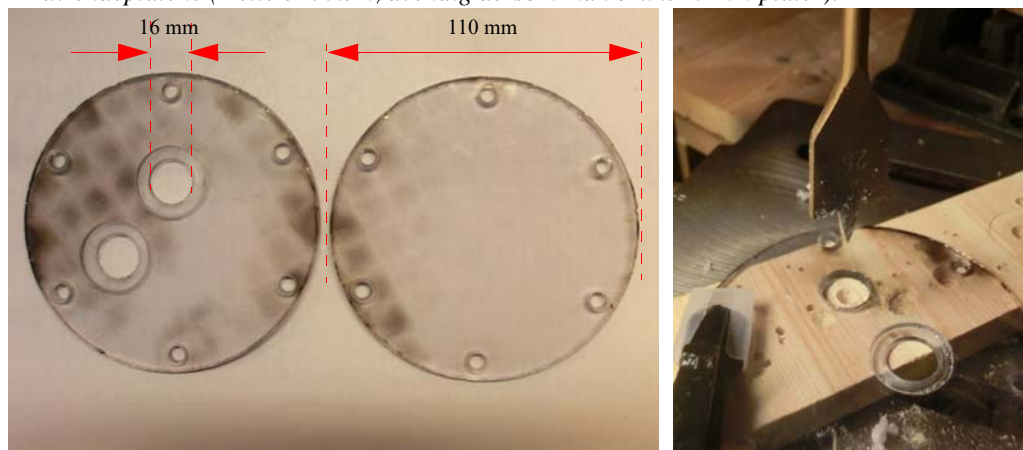
2. Endeplatene kan skrives ut med laserkutter i 4 mm plastglass kjøpt hos Biltema (3 mm akryl er imidlertid bedre), eller skjæres ut med kontursag og bormaskin. Platene rengjøres og grader fjernes. Siden kabelforskruingene<sup>3</sup> ikke har dype nok gjenger slik at mutteren får tak, så brukes et flattbor på 20 mm (se bildet under til høyre) til å redusere tykkelsen på plata omkring

---

3. Kabelforskruingene er den minste typen solgt av Biltema 5 – 8 mm, M16 (Biltema nr.: 350101).



*hullet. Tykkelsen reduseres til det halve, fra 4 til 2 mm. Husk å foreta fortynningen fra innsiden av endeplatene (Dette er ikke nødvendig dersom man bruker 3 mm plater).*



3. Lag to pakninger ved å klippe ut to ringer med bredde på ca. 10 – 15 mm av mosegummi. Diameteren må være tilpasset rørmuffa. I stedet for passer kan man presse rørmuffa ned mot mosegummien slik at det settes et merke. Dette kan brukes når man klipper.



4. Skru en mutter og en skive inn på gjengestengene (130 mm x Ø5 mm) som vist til venstre på figuren under. Skru gjengestangen fast til ett av kanthullene på endeplata (uten hull for gjennomføring av kabel) med en skive og en utvendig mutter som vist til høyre på figuren under.

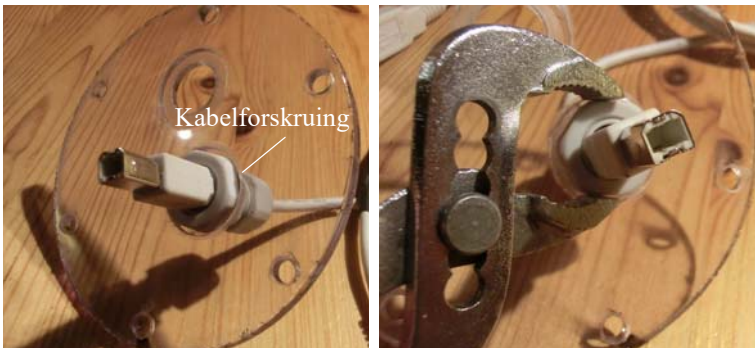




5. Til venstre på figuren under ser vi de tre gjengestengene montert i tre av hullene. Deretter legges ringen av mosegummi mellom de tre gjengestengene..



6. Monter kabelforskruingen (vanntett gjennomføring) i hullet nærmest sentrum og tre mutteren over USB-kontakten og skru fast i gjengene. Det er viktig at forsenkningen rundt hullet er på samme siden som USB-kontakten, dvs. på innsiden av kammeret, og at pakningen blir liggende på utsiden der den møter den glatte overflaten til endeplata. Skru mutteren godt til med en vannpumpetang eller skiftnøkkel på innsiden av plata. Det er spesielt viktig at denne er vanntett.

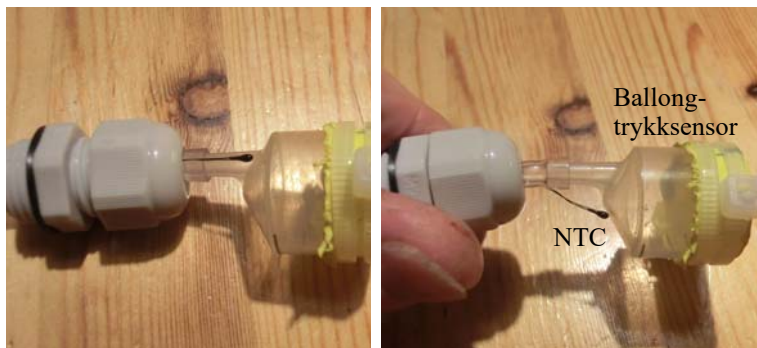


7. Sørg for at strammemutteren på utsiden av kammeret også er meget godt skrudd til, siden denne skal sørge for at vann ikke trenger inn langs kabelen.

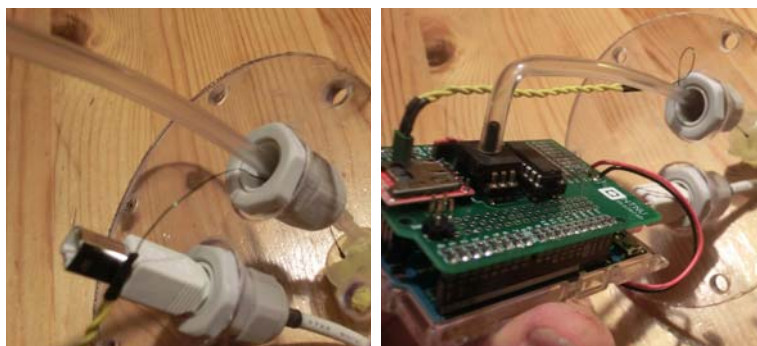




8. Før slangen med ballongsensoren og ledningen med NTC-motstanden gjennom den andre kabelforskruingen som vist til venstre på figuren under.



9. Monter kabelforskruingen med ballongsensor og NTC-sensor i det siste hullet i pleksiplate som vist til venstre på figuren under. Koble til Arduino-kortet til USB-kontakten og koble NTC-motstanden til stiftklemmene på shield-kortet. Slangen kobles til røret som stikker ut av trykksensoren.



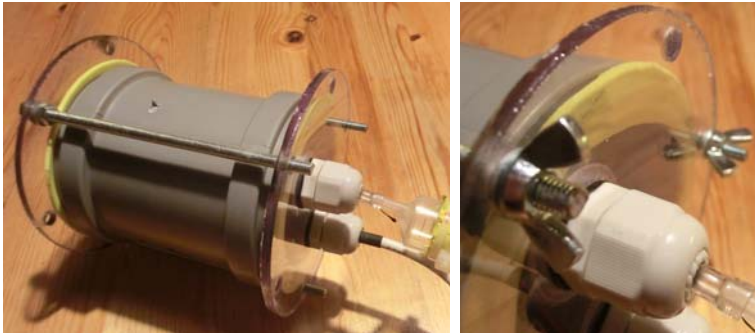
Før man monterer Arduino-kortet i kammeret, bør man vurdere å kjøre en lekkasjetest uten elektronikk i kammeret. En unnlater da å sette inn kortet, men monterer kammeret som beskrevet.

10. Tre rørmuffa ned over kortet. Pass på at batteriet kommer klar av den innvendige kanten.

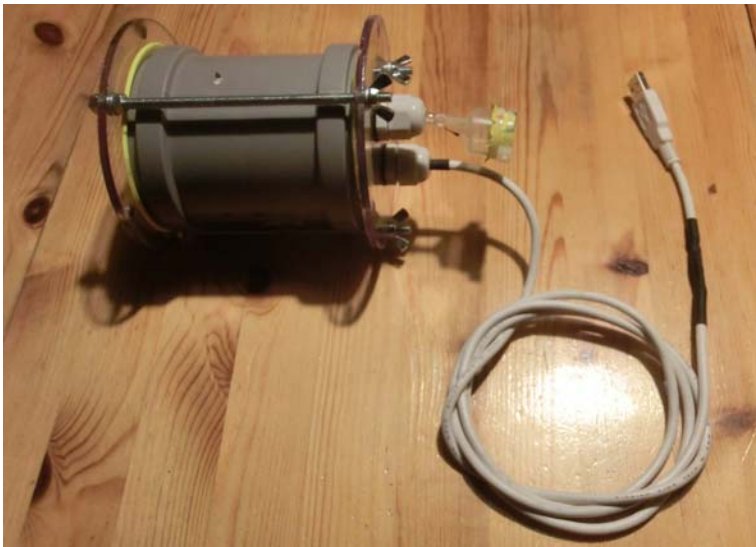




11. Tre så endeplata med gjengestangen over muffa og ned i tre av hullene i plastplata i den andre enden av muffa. Pass på at mosegummipakningene ligger godt mellom muffa og endeplatene slik at en unngår lekkasje. Legg på en skive på hvert gjengestang og skru de tre vingemutrene godt til.



12. Dermed er testjiggen ferdig og klar for testing i vann.





## Testing for lekkasje

For å teste om målejiggen er vanntett, tømmer den for elektronikk slik at den kun inneholder USB-kabel og slange for trykksensor påmontert membran for trykkregistrering. Boksen skrues godt sammen og holdes under vann en tid for å sjekke om den er tett. Bildene under viser målejiggen nedsenket i vann.



Et elektriskerrør fastspennet med en tvinge er brukt for å holde målejiggen under vann. Vi ser balongmembranen på toppen av målejiggen.



## 3 Brukerveiledning

I dette korte kapittelet skal vi se litt på hvordan vi bruker Arduino-kortet med datainnsamlingsenhet. La oss først installere Arduino programvaren.

### 3.1 Installasjon av programvaren (IDE)

For å kunne programmere Arduinokretsen er det nødvendig å installere en programpakke:

- Arduino programeditor (IDE)

#### 3.1.1 Arduino programeditor

##### Litt historikk

Arduino er et mikrokontrollerkonsept utviklet i den vesle italienske byen *Ivrea* på begynnelsen av dette årtusen. Hensikten var å lage et kontrollerkort som skulle gjøre det enklere og billigere for studenter å lære seg å programmere og bruke mikrokontrollere. Det første Arduino-produktet ble utviklet av grunnleggerne *Massimo Banzì* og *David Cuartielles*. De oppkalte prosjektet etter *Arduin of Ivrea* som var den vesle byens historiske heltefigur. Navnet betyr *sterk venn* og burde passe godt for et kraftig kontrollerkort. Det som startet som et lokalt prosjekt for studenter i 2005, hadde i 2010 spredd seg til hele verden. I april 2013 hadde de solgt over 700 000 eksemplarer, i tillegg til at de regnet med et lignende antall kloner. Den tilhørende programvaren ble utviklet av studenten *Hernando Barragán* ved det lokale universitetet i Ivrea, som en “open source” kode, derfor er kloner velkomne.

Kortet var bygget opp omkring *AVR mikrokontrollere* fra Atmel (hovedsakelig ATmega8, ATmega168, ATmega328, ATmega1280, og ATmega2560). Dette er en serie kontrollere som anvender RISC-arkitektur, en svært enkel, men meget effektiv arkitektur. Det er moro å vite at den første kontrolleren i denne serien ble utviklet av studentene *Alf-Egil Bogen* og *Vegard Wollan* ved NTH på begynnelsen av 1990-tallet. Etter endt studium tok de med seg konseptet inn i firmaet Nordic VLSI (nå NORDIC Semiconductor), hvor det ble videreutviklet. I 1995 gikk de ut av Nordic VLSI og ble snart kjøpt opp av Atmel og driver i dag firmaet Atmel Norge. De sier selv at AVR ikke har noen spesiell betydning, men det er allment akseptert at det opprinnelig sto for Alf (Egil Bogen) and Vegard (Wollan) 's Risc processor. Pr. 2016 er begge gründerne ute av Atmel Norge og har startet nye virksomheter.

##### Nedlasting av programvare

Arduino-editoren og kompilatoren, kort sagt det som trengs hentes fra:

<http://arduino.cc/hu/Main/Software>

Aktuell versjon av Arduino er Arduino 1.6.12. Filen som har navnet *arduino-1.6.12-windows* er pakket som en zip-fil. En tilsvarende fil er tilgjengelig for Mac

**ARDUINO 1.6.12**  
The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.  
This software can be used with any Arduino board. Refer to the Getting Started page for installation instructions.


Windows Installer  
Windows ZIP file for non-admin install  
Windows app (Microsoft Store)  
Get it  
Mac OS X 10.7 Lion or newer  
Linux 32 bits  
Linux 64 bits  
Linux ARM (experimental)  
Release Notes  
Source Code  
Checksums (sha256)





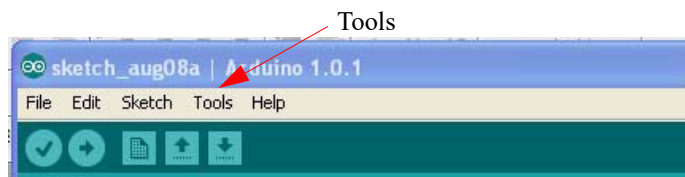
fra samme nettsted. En kan også laste ned en installasjonsfil, f.eks. Windows installer.

### Installasjon av programvaren:

1. Klikk på fila **arduino-1.6.12.zip**
2. For å pakke ut fila trenger du programmet WinRAR som kan hentes fra: <http://www.rar-lab.com/download.htm> eller Winzip som kan hentes fra: [www.winzip.com/downwz.htm](http://www.winzip.com/downwz.htm)
3. Velg *Extract to* fra menylinjen øverst og velg f.eks. C:/Programfiler og trykk OK. Programfilene legges da i en egen underkatalog (arduino-1.6.12) i katalogen *Programfiler*.
4. Programmet startes ved å klikke på programikonet:  .


5. Koble til USB-kabelen til ønsket port.

6. Klikk på *Tool* på menylinjen og velg *Board*. Her velges hvilken variant i Arduino familien du skal jobbe med. I dette tilfellet velg: *Arduino/Genuino UNO*..









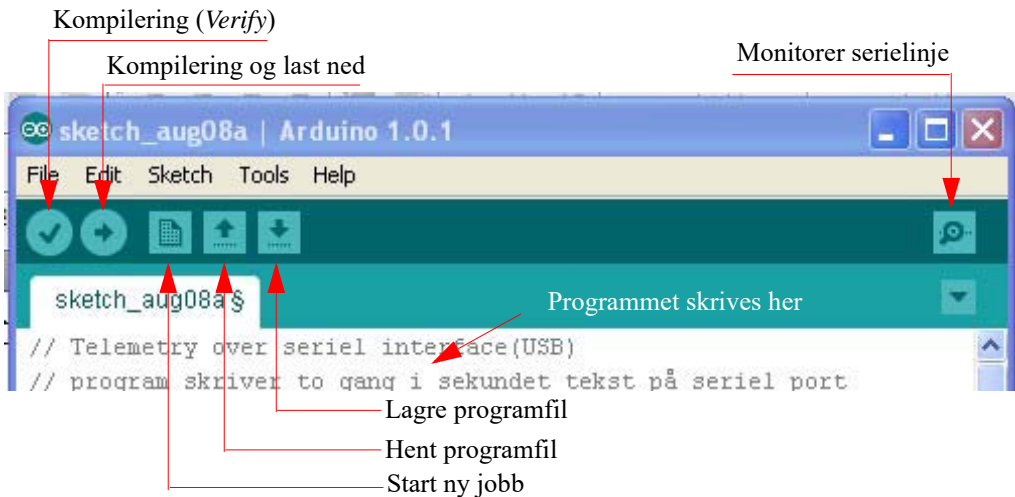
7. Klikk på *Tool* på menylinjen og velg *Serial Port*. Sjekk at riktig port (Com?) er valgt. Den kommer ofte opp med beskjed om at den har funnet kortet og har gitt den et Com nr.

Programmet er klart til bruk og du kan skrive inn programlinjene. Når programmet er ferdig skrevet, skal det *kompileres*, dvs. overføres til en binærkode som mikrokontrolleren forstår. Dersom programmet inneholder ulovlige kommandoer eller skrivefeil, så vil kompilatoren varsle om det og vise på hvilken linje feilen er avslørt. Det er ikke nødvendigvis alltid der feilen er gjort.

Dernest skal programmet lastes ned til kontrollenhetens minne (Arduino-kortet). Dette gjøres ved å trykke på knappen .

### Kort oversikt over Arduino-editoren

-  Verifiser at koden er riktig
-  Kompiler og last ned programmet til kontrollenheten
-  Hent nytt "arbeidsark"
-  Hent en eksisterende programfil
-  Lagre programfil
-  Monitorer data sendt tilbake på serielinjen



### Manglende kontakt med kortet

Det hender at en ikke oppnår umiddelbar kontakt med Arduino-kortet når en forsøke å laste ned et program. Feilmeldingen: **avrdude: stk500\_getsync(): not in sync: resp=0x00** i meldingsvinduet betyr at det ikke oppnås kontakt med kortet. Dette kan skyldes flere ting:

- Kablen er ikke tilkoblet eller ødelagt
- Feil port er valgt av programeditoren  
Endres ved å velge: *Tool* og *Serial Port* fra menylinjen i editoren
- Feil type kontroller-kort valgt  
Endres ved å velge: *Tool* og *Board* fra menylinjen for så å velge rett kort, i vårt tilfelle *Arduino UNO*.
- Rx/Tx linjene har en tilleggsfunksjon som ikke er frakoblet under programmeringen. F.eks. ved at datainnsamlingsenheten ikke er frakoblet, eller at strappene på kortet ikke er fjernet under programmeringen.
- Manglende driver

Ved installasjon av drivere gjøres følgende (Windows 7 - tilsvarende kan gjøres for Windows 8 og 10):

- Åpne kontrollpanelet
  - Velg: *System og sikkerhet*
  - Velg: *System*
  - Velg: *Enhetsbehandling* (venstre meny)
  - Velg: *Andre enheter* eller *Porter* og finn den porten som Arduino er tilkoblet

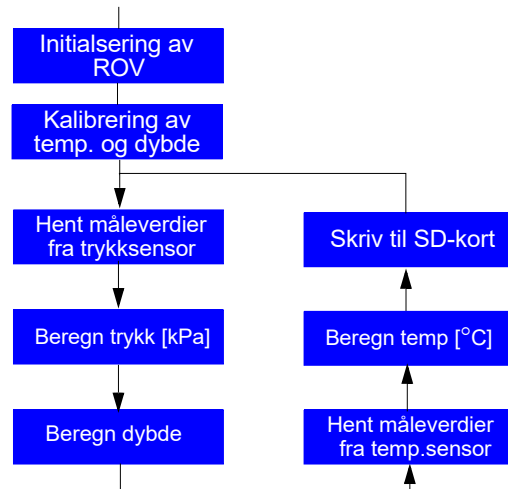


- Høyreklikk på porten og velg *Oppdater driverprogramvare* fra menyen
- Velg: *Søk på datamaskin etter driverprogramvare*
- Bla gjennom og finn katalogen *Arduino 1.6.12/Drivere* - pek på den og start installasjon

Om du er heldig installeres driveren og du får kontakt med Arduino-kortet.

### 3.1.2 Programstruktur

Databehandlingsenheten (Arduino UNO) styrer hele prosessen i CanSat'en. Den henter inn data fra datainnsamlingsenheten (*shield* kortet), omregner fra tallverdi til trykk, temperatur og dybde, tilrettelegger og sørger for å sende data til datalagringseenheten (Openlog). Dette gjøres i en gjentakende sløyfe som vist på figuren til høyre. En slik oversikt over programmet kalles et *flytdiagram*.



### 3.2 Bruk av enhet for datainnsamling

#### Koble til batteriet

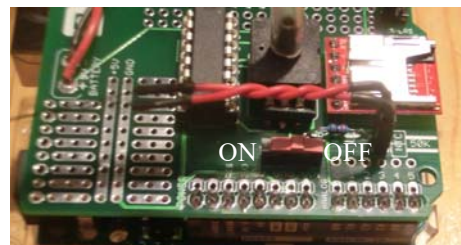
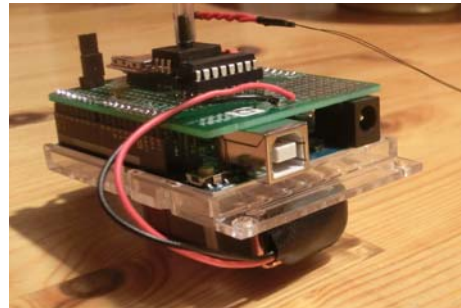
Batteriet kobles til ved å presse batteriklemmene på batteriet. Selv om klemmen bare kan sette på riktig vei, så kan kortet få gal spenning dersom klemmen holdes inntil batteripolene feil vei. Kortene får spenning fra PC-en når USB-kabelen er tilkoblet. Batteriet kobles automatisk ut når USB-kabelen plugges i. Når batteriet er tilkoblet vil dette automatisk ta over idet kabelen kobles fra.

#### Av/På bryteren

ON ved glidebryteren på kortet viser når spenningen fra batteriet er tilkoblet kortet.

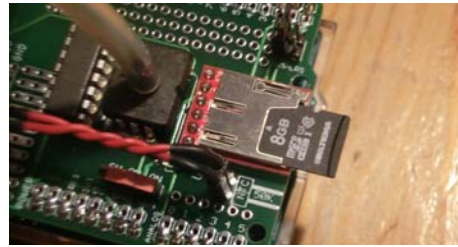
#### Sett i mikro SD-kortet

SD-kortet kommer i to deler. Selve mikro SD-kortet og en adapter som gjøre det mulig å lese data via kortleseren på PC-en.





Mikro SD-kortet trykkes inn i kortleseren og låser seg i posisjonen ved å avgi et knepp. Kortet frigjøres ved å trykke det inn. Kortet på bildet til høyre er på vei inn i SD-kort leseren. Når det er på plass er det helt inne i kortleseren.

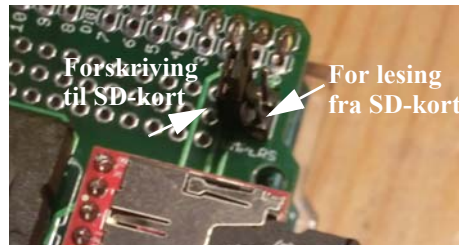


## Jumpere


Ved siden av SD-kort leseren står to svarte jumpere. Disse åpner for skriving til og lesing fra SD-kortet.

Siden vi kun skal *skrive* til kortet, så er det bare nødvendig å sette inn jumperen lengst fra kanten som vist på bildet.



**NB! Når kortet skal programmeres er det viktig at vi tar ut jumperen**



## Legg inn programmet i Arduino-kortet

1. Start Arduino-programmet ved å klikke på programikonet:  .
2. Hent opp filen *ROV\_test* fra minnepinnen ved å velge file/open fra menylinja



3. Koble USB-kabelen mellom Arduino-kortet og en av USB-portene på PC'en
4. Velg riktig kort ved å velge Tool/Kort/ fra menylinjen og velg Arduino/Genuino UNO
5. Velg riktig port ved å velge Tool/Port velg porten med høyest nummer eller som indikerer at porten er tilkoblet Arduino-kortet
6. Kompiler og last programmet fra PC'en over til Arduino-kortet ved å trykke 
7. Velg å se på Monitoren , dvs. hva som sendes tilbake fra programmet. Monitoren vil gi en liste over målinger: Dybde og Temperatur
8. Hold fingeren rundt NTC-motstanden (temperatursensoren) og se at temperaturen endrer seg.

Du skal nå ha opprettet kontakt mellom PC'en og Arduino-kortet.

## Monitorering og datalagring

Det er lagt inn to blokker med utskrifter:

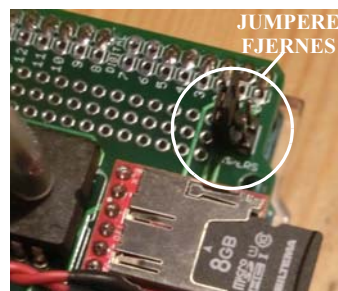
36 ROV – med trykk- og temperaturmåling



## 1. Utskrift for test og kalibrering

Denne gir mye informasjon om både analoge verdier og kalibrerte verdier av trykk/dybde og temperatur. Denne er ment for bruk under uttesting og kalibrering:

```
Serial.print("Nr.: ");  
Serial.print(i);  
Serial.print(", Analogt trykk: ");  
Serial.print(U_D5, 2);  
Serial.print(", Beregnet dybde: ");  
Serial.print(D_beregn, 2);  
Serial.println(" m");  
Serial.print("Digital temp.: ");  
Serial.print(U_A0);  
Serial.print(", Beregnet temp.: ");  
Serial.print(t_Beregn);  
Serial.println(" C");
```



Under uttesting er det greit at *jumperne er ute* slik at det er greit å programmere Arduino-kortet fortløpende etter behov. Dvs. at data ikke skrives til SD-kortet, men bare ut på monitoren.

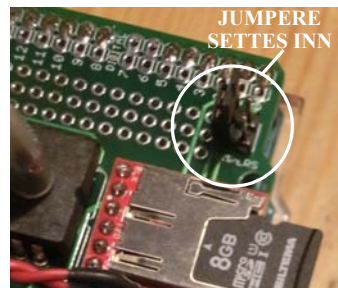
## 2. Utskrift for datainnsamling

Denne gir kun målnummer, dybde [m] og temperatur [°C]:

```
Serial.print(i);  
Serial.print(", ");  
Serial.print(D_beregn, 2);  
Serial.print(", ");  
Serial.print(t_Beregn, 2);  
Serial.println(","); */
```

Verdiene skiller med komma som gjør det lett å laste verdiene inn i regneark

De to variantene velges inn ved å bruke kommentarsyntaksen.



Under datainnsamling må *jumperne være på plass* slik at data kan skrives til SD-kortet.



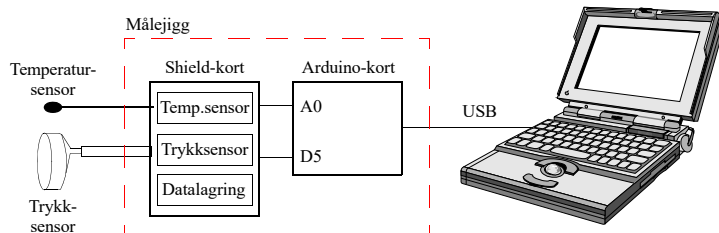
## 4 Litt sensorteori

I dette kapitlet vil vi omtale de to sensorene som benyttes i dette prosjektet: Temperatur og trykksensoren.

La oss først se på koblingskjemaet for måleenheten (shield-kortet).

### 4.1 Kretsskjema for datainnsamlingsenheten (shield-kortet)

Figuren under viser et blokkdiagram over datainnsamlingsenheten koblet til Arduino-kortet og en bærbar laptop. I denne delen vil vi ikke omtale den delen av kontrollerkortet som omhandler styring av motorene.



La oss først se litt nærmere på de to sensorene, temperatur og trykk.

### 4.2 Temperatursensor (NTC og PTC -motstander)

Metaller vil normalt ha økende resistans med økende temperatur. I et halvledermateriale vil flere ladningsbærere løftes opp i ledningsbåndet slik at ledningsevnen går opp, dvs. at resistansen blir mindre.

De fleste motstandsmaterialer endrer resistans som funksjon av temperaturen. Som regel er dette uønsket, men i noen spesielle tilfeller ønsker man nettopp en slik variasjon og utformer komponenten og materialet deretter. Slike motstander brukes også i forbindelse med måling eller deteksjon av temperaturendringer, eller til å motvirke uønsket temperaturdrift i elektronisk utstyr.

- NTC - *Negative Temperatur Coefficient*, dvs. at resistansen avtar med økende temperatur.
- PTC - *Positive Temperatur Coefficient*, dvs. at resistansen øker med økende temperatur.

#### NTC-motstanden

NTC-motstander er laget av et materiale hvis resistivitet varierer sterkt med temperaturen. Som navnet sier (*Negative Temperature Coefficient* - NTC) så avtar resistansen med økende temperatur.

NTC-motstander er derfor vanligvis bygget opp som en polykrytalinsk halvleder som kan bestå av en blanding av krom, mangan, jern, kobolt og nikkell, som sintres<sup>4</sup> sammen med et plastisk bindemiddel.

---

4. Sintring betyr at metallpulver knyttes sammen ved hjelp av oppvarming, men uten å smelte.



En forenklet sammenheng mellom resistansen ( $R$ ) og temperaturen ( $T$ ) kan uttrykkes som:

$$R = Ae^{B/T} \quad (4.1)$$

hvor  $A$  og  $B$  er “konstanter” bestemt av materialet og temperaturen. Konstantene kan betraktes som *tilnærmet konstante* innen begrensede temperaturområder.

I datablader for NTC-motstander oppgis gjerne resistansen ( $R_r$ ) for en referansetemperatur ( $T_r$ ). I et temperaturområde rundt denne referansetemperaturen antas  $B$ -verdien å være tilnærmet konstant ( $B_{0/100}$  –  $B$ -verdien er tilnærmet konstant innen området  $0^\circ\text{C}$  til  $100^\circ\text{C}$ ).

Vi kan da sette opp følgende:

$$R = Ae^{\frac{B_{0/100}}{T}} \quad (4.2)$$

$$R_r = Ae^{\frac{B_{0/100}}{T_r}} \quad (4.3)$$

Ved å eliminere  $A$  fra disse uttrykkene, kommer vi fram til følgende sammenheng, løst med hensyn til resistansen  $R$ :

$$R = R_r \cdot e^{\left(\frac{B_{0/100}}{T} - \frac{B_{0/100}}{T_r}\right)} \quad (4.4)$$

Dette uttrykket går under betegnelsen *Beta-formelen*.

Når vi skal beregne verdien for en NTC-motstand ved en gitt temperatur, slår vi opp  $B$ -verdien,  $R_r$  og  $T_r$  i databladet, sørger for at de aktuelle temperaturene ligger innenfor området til  $B$ -verdien, og beregner  $R$  ved å sette inn ønsket temperatur  $T$ . Temperaturen angis i grader Kelvin.



## B57550G1103F005 (B57550G)<sup>5</sup>

Fra databladet<sup>6</sup> for B57550G finner vi følgende:  $R_{25}$  er referansemotstand ( $R_T$ ) ved 25 °C ( $T_T = 298$  K):

$R_{25}$ Ω	No. of R/T characteristic	$B_{25/85}$ K	$B_{0/100}$ K	$B_{25/100}$ K	Ordering code
2 k	8401	3420	3390 ±1%	3436	B57550G0202+000
2 k	8401	3420	3390 ±1%	3436	B57550G0202+002
5 k	8402	3480	3450 ±1%	3497	B57550G0502+000
5 k	8402	3480	3450 ±1%	3497	B57550G0502+002
10 k	7003	3612	3586	3625 ±1%	B57550G1103+005
10 k	7003	3612	3586	3625 ±1%	B57550G1103+007
10 k	8307	3478	3450	3492 ±1%	B57550G1103+000
10 k	8307	3478	3450	3492 ±1%	B57550G1103+002
20 k	8415	3992	3970 ±1%	4006	B57550G0203+000
20 k	8415	3992	3970 ±1%	4006	B57550G0203+002
30 k	7002	3973	3944	3988 ±1%	B57550G1303+005
30 k	7002	3973	3944	3988 ±1%	B57550G1303+007
50 k	8403	3992	3970 ±1%	4006	B57550G0503+000
50 k	8403	3992	3970 ±1%	4006	B57550G0503+002
100 k	8304	4072	4036	4092 ±1%	B57550G1104+000
100 k	8304	4072	4036	4092 ±1%	B57550G1104+002

*Datablad for NTC-motstand B57550G, 2–100 kΩ*

Med disse dataene kan vi skrive:

$$R = 10\text{k} \cdot e^{\left(\frac{3586}{T} - \frac{3586}{298}\right)} \quad (4.5)$$

hvor  $B_{0/100} = 3586$  (B57550G1103F005 – 10 kΩ) og referansetemperaturen  $T_T = 298$  K.

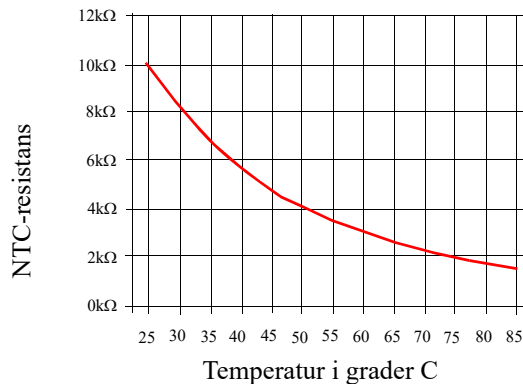
5. Vi har tidligere benyttet RH16 50kOhm. Vi har imidlertid at denne er gått ut av sortementet til ELFA. Vi har derfor valgt alternativet B57550G1103F005 – 10 kOhm som alternativ

6. Databladet er hentet fra: [https://www.elfa.se/Web/Downloads/\\_t/ds/B57550G-B57550G1\\_eng\\_tds.pdf?mime=application%2Fpdf](https://www.elfa.se/Web/Downloads/_t/ds/B57550G-B57550G1_eng_tds.pdf?mime=application%2Fpdf)





Dersom vi beregner verdier for R i temperaturområdet 25°–85°C, får vi følgende graf:



NTC resistans som funksjon av temperaturen B57550G1103F005 10 kΩ

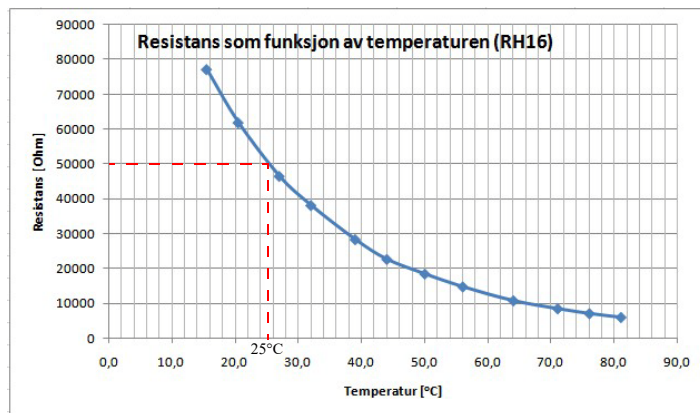
En annen viktig parameter for NTC-motstander, er hvor raskt resistansen endrer seg med temperaturen. Denne parameteren betegnes *NTC-motstandens tidskonstant* ( $\tau$ ), og angir den tiden det tar for resistansen og endre seg til 63,2% av den nye resistansen etter at temperaturen har endret seg 1 K (Kelvin) over omgivelsestemperaturen. En antar at temperaturendringen ikke er forårsaket av indre oppvarming på grunn av elektrisk strøm som flyter gjennom motstanden.

I vårt eksempel er  $\tau = 7$  sek i luft. (se datablad). Legg merke til at tidskonstanten er angitt for kontakt med luft. Ved kontakt med vann, vil denne reduseres betydelig, så den burde derfor ikke skape problemer.

### Registrering av kurveform for RH16<sup>7</sup>

Vi har gjort målinger på RH16 – 50 kΩ og fått følgende kurve.

Målingen ble gjort med et Ohm-meter. NTC-motstanden ble senket ned i et glass med ca. 2 dl springvann oppvarmet med en vannkoker før det fikk lov til å kjøle seg i romtemperatur. Vi ser at ved 25°C er verdien omtrent 50 kΩ som spesifisert i databladet.



Resistans som funksjon av temperatur for RH16 – 50 kΩ

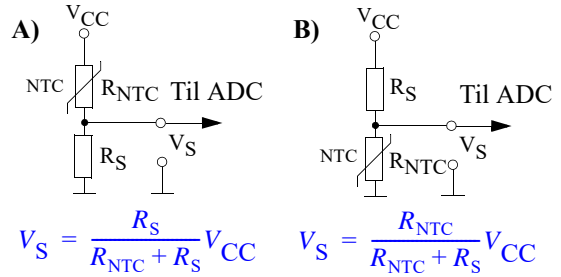
7. Det er foreløpig ikke foretatt målinger av B57550G1103F005, da vi ennå ikke har fått bestilt den.



En datalogger vil også kunne benyttes.

### Oppkobling mot ADC

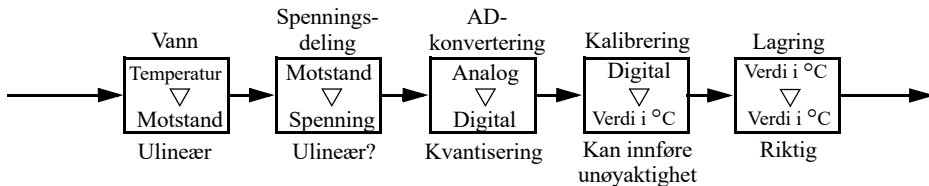
Siden grensnittet til kontrolleren krever en spenning, kobles NTC-motstanden i serie med en motstand som vist i figuren under. Velg verdien på seriemotstanden lik den nominelle verdien på NTC-motstanden ( $R_{25}$ ). Spenningsnivået  $V_S$  beregnes fra formlene som antydnet på figuren under. Legg merke til at oppkoblingen på tegning A gir økende spenning  $V_S$  med økende temperatur, mens oppkoblingen i tegning B gir fallende spenning med økende temperatur.



På bakgrunn av ligningene foran kan en utvikle et uttrykk for temperaturen som funksjon av spenningen som kan legges inn i prosessoren.

### Måleusikkerhet for temperatursensoren

NTC-motstanden er en relativt kjapp sensor hovedsakelig pga av sine små dimensjoner. Det skal lite energiflyt til for å varme opp eller avkjøle materialet. Siden NTC-motstandens i seg selv er sterkt ulinear så vil en linearisering uansett være en tilpasning som kun gjelder for en relativt snevert temperaturområde. Figuren under har vi satt opp en kjede som beskriver de ulike overgangene fra temperaturen i vannet til lagret verdi på SD-kort ev. overføring av verdien på kabelen.

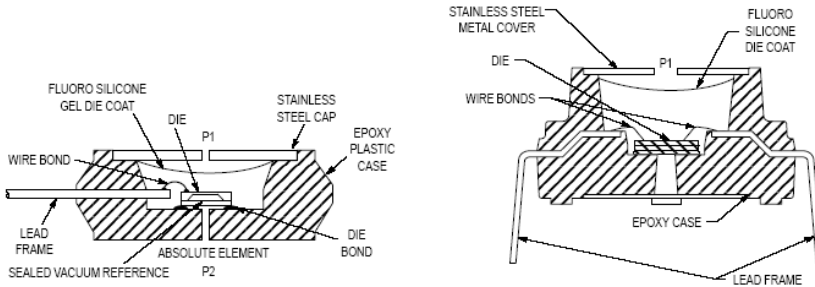


Uten at vi har gjennomført nøyaktige undersøkelser så er det rimelig å anta at de største feilene innføres ved å anta at overgangen fra motstand til spenning i spenningsdeleren er lineær. Feil vil også kunne innføres under kalibreringen, hvor vi er avhengig av presisjonen til det kalibrerte termometeret og hvor nøyaktig vi klarer å lese av verdien til NTC-motstanden. Dessuten vil AD-konverteren innføre kvantiseringsstøy siden den analoge spenningen fra spenningsdeleren må tilordnes diskrete verdier.



### 4.3 Trykksensor (SPD030G)

Smartec trykksensor er basert på piezo-resistive elementer som trykkfølere. Dvs. at lufttrykket presser på framsiden av en piezo-resistiv skive. Baksiden av skiven har et veldefinert trykk (f.eks. vakuum) slik at sensoren skal kunne måle absolutt lufttrykk. Figuren under viser et eksempel på hvordan en slik sensor kan utformes. Vi legger merke til under selve skiva (DIE) så finnes et forseglet vakuumkammer gjør at denne sensoren kan måle absolute lufttrykkverdier.



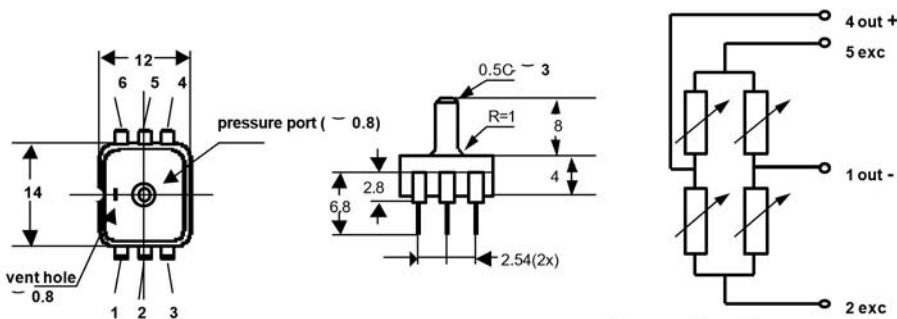
*Utforming av trykksensor med piezo-elektriske elementer.*

Hos sensoren SPD030G, som vi bruker, arbeider den piezo-resistive skiven mot atmosfærisk trykk eller trykket som til en hver tid omgir sensoren. Vår sensor vil dermed måle den relative verdien mellom lufttrykket inne i kammeret (ROV) og lufttrykket som til en hver tid tilføres framsiden av sensoren gjennom slangen, som igjen er en funksjon av vanntrykket og dermed dybden.



Grunnen til at vi valgte SPD030G er at denne har et måleområde fra 0 - 2 atmosfærers trykk, hvilket betyr at vi kan måle ned til 10 m dybde, hvilket er tilstrekkelig i de fleste tilfeller.

Selve sensorelementet er utformet slik at de piezo-resistive elementene i skiva kan inngå i en målebros slik at endring i resistivitet kan omgjøres til en spenning. Denne kan så forsterkes til ønsket nivå.



Til høyre på figuren over ser vi målebros som består av fire motstander. En eller flere av disse motstandene er den piezo-resistive plata i sensoren. Når trykket endrer seg vil også plata bøyes opp eller ned avhengig av om lufttrykket på oversida av plata er mindre eller større enn lufttrykke



i omgivelsene. Vi setter en spenning på 5 V mellom pinne 5 (exc) og 2 (exc). Et økende lufttrykk vil øve trykk på den piezo-resistive plata som dermed endrer resistansen. Denne endringen vil forskyve likevekten i målebroa og vi måler en endring i spenning mellom pinne 4 (out +) og 2 (exc).

Denne sensoren er tilpasset en krets (UTI) som gjør små endringer i spenning om til et tog av pulser som i mange tilfeller vil være mer robust for å overføre informasjonen på en kabel med to ledninger.

### Universal Transducer Interface (UTI)

Små endringer i spenning er ganske utsatt for støy dersom de skal overføres langs en kabel, derfor er det utviklet en krets, en såkalt Universal Transducer Interface (UTI), som omformer de små spenningsvariasjonene til varierende avstand mellom pulser. Vi har valgt en slik her, ikke først og fremst for å unngå støy, men mest for at vi da kan benytte en utprøvd metode. Dessuten finnes det ferdig programvare for å koble den opp mot Arduino.

Kretsen kan brukes på mange forskjellige måter. Vi har valgt å koble den opp for å måle spenningen mellom pinne 2 og 4 i målebroa omtalt foran.

Når vi kobler den opp slik, betyr det at vi kan ta ut +/- 200 mV over broa og gjøre om dette til et tog av pulser hvor avstanden mellom pulsene ( $T_x$ ) er et mål for spenningen over broa (mellom pinne 1 og 4). Figuren til høyre viser et eksempel på et slikt tog av pulser:

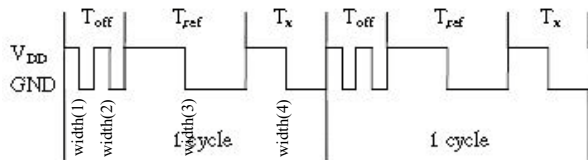


Figure 2. The output signal of the UTI for a 3-phase mode

Som vi ser inneholder pulstøget mer informasjon enn  $T_x$  som er et uttrykk for den aktuelle spenningen over målebroa som vi først og fremst er interessert i.  $T_{off}$  er et uttrykk for offsetspenningen og  $T_{ref}$  et uttrykk for referansespenningen. Offset er en uønsket målefeil som f.eks. kan skyldes at varierende temperatur endrer målingen på en uønsket måte.  $T_{off}$  gjør at vi til en viss grad kan korrigere for slike feil. Mens  $T_{ref}$  er et mål for batterispenningen som ligger over målebroa dvs. 5 V, den samme spenningen som i formelen under oppgis som  $V_{ref}$ .

Programmet i Arduinoen må da måle lengden av disse tre periodene i mikrosekunder og på bakgrunn av det beregne spenningen. Dette kan som kan uttrykkes slik:

$$V_x = \frac{T_x - T_{off}}{T_{ref} - T_{off}} \cdot V_{ref} \quad (4.6)$$

Her er  $V_{ref} = 5$  V, og  $T_x$  og  $T_{off}$  må vi måle.

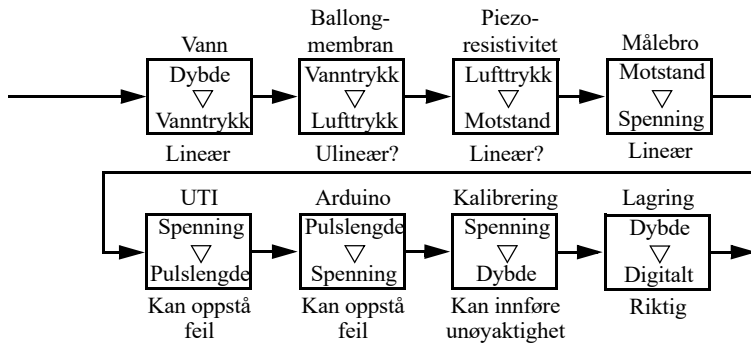
Denne formelen er lagt inn i programvaren, se avsnitt A.1 på side 79.

### Måleusikkerhet for trykksensoren

Etter at Arduino-kortet har hentet inn data fra sensorene og regnet ut  $V_x$ , så må denne spenningen regnes om til dybde. Det vi kan anta, men det er slett ikke sikkert, er at det innenfor det måleområdet vi skal bruke denne sensoren (0 – 10 meter), så er det en lineær sammenheng mellom



spenningen ut av målebroa og trykket. Dessuten antar vi en lineær sammenheng mellom trykk og dybde. Imidlertid er det ganske usikkert hvor lineært ballongmembranen oppfører seg. Her ligger det an til en liten forskningsoppgave. Figuren under gir en oversikt over målekjeden for trykk og mulige ulinearitet:



I denne kjeden er det mange ting som kan gå galt. På figuren har vi forsøkt å antyde hvor vi må være oppmerksomme. Det vi har oppdaget så langt er følgende:

- **Observasjon:**

Vi ser at spenningen på utgangen varierer tilfeldig rundt en middelvei. Foreløpig er det uklart hvor stor denne usikkerheten er i forhold til det utslaget reel endring i dybden utgjør.

**Årsak:**

Årsaken er usikker, men en foreløpig hypotese er at det er elektrisk støy som slår inn på målebroa og dermed medfører en endring av måleverdien.

**Tiltak:**

Støy kan muligens dempes med å avkoble referansespenningen ( $V_{ref}$ ) til målebroa med kondensatorer. Dvs. koble en eller to kondensatorer mellom spenningen på målebroa og jord (dette ble gjort i versjon 2.0 og 3.0 av kortet). Støy kan sannsynligvis også reduseres med å kjøre på ren batteridrift. En kan også gjøre flere målinger og beregne en middelvei. Siden måletakten er relativt lav, vil dette gjøre oppdateringen langsom. Kanskje 20 – 30 sekunder mellom hver måleverdi, hvilket høres lenge ut.

- **Observasjon:**

Måleverdier i spenningen på utgangen som gjøre store og urimelige hopp.

**Årsak:**

Årsaken er usikker, men feilen oppstår sannsynligvis i dekodningen av signalet fra UTI-kretsen, dvs. en unøyaktighet i den målingen programmet utfører.

**Tiltak:**

Gjøre programmet mer robust. En enkel løsning er å la programmet filtrere ut verdier som opplagt er feil: En kan ikke hoppe fra 1,5 meters dybde til 150 meter i løpet av 1 sek. Det ser dessuten ut til at denne feilen forsvinner nesten helt dersom man øker tiden mellom hver måling til 5 sekunder.



## 4.4 Kalibrering og omregning

Som det tydelig framkommer av avsnittet foran så er det mange kilder til feil. Siden målingene som oftest er relative så må vi foreta kalibreringer hvor vi knytter spenningsverdier til henholdsvis temperatur og trykk. Vi foretar ikke en kalibrering for hvert trinn, men tar hele kjeden under ett. La oss se på hver av de to sensorene:

### 4.4.1 Prosedyre for kalibrering av temperatursensoren

Til kalibrering av temperaturmålingen trenger vi følgende:

- Et kalibrert termometer
- Vann med forskjellige temperaturer
- Ett begerglass til å ha vannet i
- NTC-sensoren med tilhørende elektronikk

Slik går vi fram ved kalibreringen:

1. Fyll begerglasset med vann med en temperatur på rundt 10 °C (springvann)
2. Installer måleprogrammet i Arduino-kortet slik at programmet starter
3. Åpne monitorvinduet og finn spenningsverdiene fra NTC-motstanden: “Digital temp.”
4. Dypp NTC-motstanden i vannet med *lav* temperatur ( $t_L$ ) og noter en tilnærmet middelverdi for den digitale spenningsverdien for temperaturen: “Digital temp.” ( $U_L$ )
5. Les av termometeret som viser temperaturen på vannet, og noter verdien ( $t_L$ )
6. Fyll begerglasset med lunkent vann f.eks. 30 °C
7. Åpne monitorvinduet og studer spenningsverdiene fra NTC-motstanden: “Digital temp.”



```
COM7 (Arduino/Genuino Uno)
Digital spenningsverdi for temp. fra NTC

Nr.: 1, Analogt trykk: 2.83, Beregnet dybde: -0.10 m
Digital temp.: 470, Beregnet temp.: 20.39 C
Nr.: 2, Analogt trykk: 3.13, Beregnet dybde: -0.08 m
Digital temp.: 471, Beregnet temp.: 20.49 C
Nr.: 3, Analogt trykk: 3.70, Beregnet dybde: -0.03 m
Digital temp.: 471, Beregnet temp.: 20.49 C
Nr.: 4, Analogt trykk: 2.97, Beregnet dybde: -0.09 m
Digital temp.: 471, Beregnet temp.: 20.49 C
```



8. Dypp NTC-motstanden i vannet med høy temperatur ( $t_H$ ) og noter en tilnærmet middelverdi for den digitale spenningsverdien for temperaturen: "Digital temp." ( $U_H$ )
9. Les av temperaturen på vannet og noter verdien ( $t_L$ )

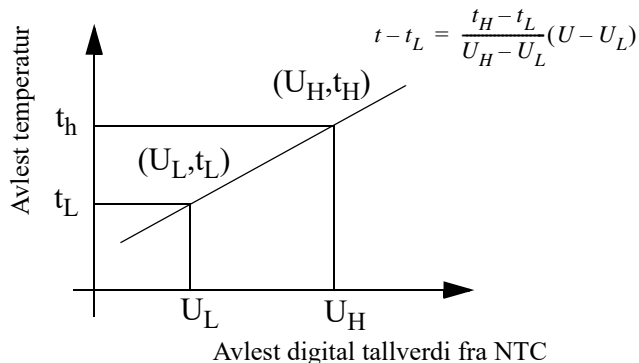
Vi har nå fire verdier, to temperaturer ( $t_L$  og  $t_H$ ) og to spenninger ( $U_L$  og  $U_H$ ), vi har dermed to punkter på en linje som beskriver sammenhengen mellom temperatur og spenning under forutsetning av at den er ganske lineær.



Måling	Avlest "Digital temp."	Avlest temp. termometer
Ved lav temperatur	$U_L =$	$t_L =$
Ved høy temperatur	$U_H =$	$t_H =$

### Omregning fra digital spenning til temperatur (RH16)

Figuren under illustrerer hvordan vi bruker topunksformelen i denne sammenhengen.



På dette grunnlaget kan vi sette opp følgende formel:

$$t - t_L = \frac{t_H - t_L}{U_H - U_L} (U - U_L) \quad (4.7)$$

Topunksformelen for lineære ligninger kan også skrives slik:

$$t = ((t_h - t_L) / (U_H - U_L)) * (U - U_L) + t_L \quad (4.8)$$

Hvis  $k = (t_h - t_L) / (U_H - U_L)$  (stigningskoeffisienten), kan vi skrive ligningen slik:



$$y = k(U - U_L) + t_L \quad (4.9)$$

**Denne formelen er realisert i Arduino-programmet.** For oss er det derfor bare å sette inn de målte verdiene for den digitale verdien for spenningen ved lav og høy temperatur og temperaturene i grader C som vist under.

I programmet har vi valgt følgende uttrykk for de fire verdiene (de verdiene som står er eksempler):

// Skriv inn de avleste og målte verdiene for spenning og temperatur her

```
int U_Lav_A0 = 382;      // UL (digital verdi for spenningen på A0, registrert lav temperatur)
int U_Hoy_A0 = 686;     // UH (digital verdi for spenningen på A0, registrert ved høy temperatur)
float t_Lav = 12;       // tL (lav kalibreringstemperatur i grader C)
float t_Hoy = 41;       // tH (høy kalibreringstemperatur i grader C)
```

// På bakgrunn av disse verdiene leser programmet av og beregner følgende verdier:

```
int U_A0 = 0;           // U (avlest digitalverdi for spenning på A0 ved aktuell posisjon)
float t_Bereg = 0;      // t (beregnet temperatur ved aktuell posisjon)
```

#### 4.4.2 Prosedyre for kalibrering av trykksensoren

Til kalibrering av trykksensoren trenger man følgende:

- En stor plastsylinder på over en meter og en diameter som gjør det mulig å senke målekammeret under vann
- Vann for å fylle plastsylinderen
- En stang eller et plastrør som kan brukes til å holde plastsylinderen under vann (ev. lodd)
- En tommestokk eller metermål
- Trykk-sensoren med tilhørende elektronikk

Slik går man fram ved kalibreringen:

1. Fyll plastsylinderen med vann, pass på at det ikke er mer vann enn at måleutrustningen kan dyppes helt under uten av det renner over
2. Koble måleutrustningen til PC-en og start monitoren i Arduino-programmet, se at det kommer måledata fra trykksensoren







3. Noter spenningsverdien fra trykksensoren (“Analogt trykk” ( $U_0$ )) mens den ennå er over vann og sett dybden til 0,00 meter ( $D_0$ )
4. Senk måleutrustningen ned til bunnen av plastsylindere og noter spenningsverdien fra trykksensoren (“Analogt trykk” ( $U_1$ )) og avstanden fra vannflaten og ned til sensoren (ballongmembranen) ( $D_1$ ).

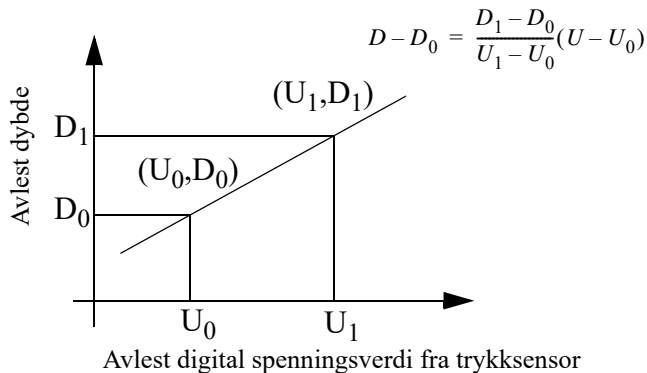
```
COM7 (Arduino/Genuino Uno)
Måleverdier fra trykksensoren
-----
Nr.: 1, Analogt trykk: 2.83, Beregnet dybde: -0.10 m
Digital temp.: 470, Beregnet temp.: 20.39 C
Nr.: 2, Analogt trykk: 3.13, Beregnet dybde: -0.08 m
Digital temp.: 471, Beregnet temp.: 20.49 C
Nr.: 3, Analogt trykk: 3.70, Beregnet dybde: -0.03 m
Digital temp.: 471, Beregnet temp.: 20.49 C
Nr.: 4, Analogt trykk: 2.97, Beregnet dybde: -0.09 m
Digital temp.: 471, Beregnet temp.: 20.49 C
```

Måling	Avlest “Analogt trykk”	Avlest dybde, metermål
Ved overflata (i luft)	$U_0 =$	$D_0 =$
Ved bunnen	$U_1 =$	$D_1 =$

Vi har nå fire verdier, to dybdemålinger ( $D_0$  og  $D_1$ ) og to spenninger ( $U_0$  og  $U_1$ ), vi har dermed to punkter på en linje som beskriver sammenhengen mellom dybde og spenning under forutsetning av at den er tilnærmet lineær.

### Omregning fra digital spenning til dybde (SPD030)

Figuren under illustrerer hvordan vi bruker topunktsformelen i denne sammenhengen.



På dette grunnlaget kan vi sette opp følgende formel:

$$D - D_0 = \frac{D_1 - D_0}{U_1 - U_0} (U - U_0) \quad (4.10)$$

Topunktsformelen for lineære ligninger kan også skrives slik:

$$D = ((D_1 - D_0) / (U_1 - U_0)) * (U - U_0) + D_0 \quad (4.11)$$

Hvis  $k = (D_1 - D_0) / (U_1 - U_0)$  (stigningskoeffisienten), kan vi skrive ligningen slik:



$$D = k (U - U_0) + D_L \quad (4.12)$$

**Denne formelen er realisert i Arduino-programmet.** For oss er det derfor bare å sette inn de målte verdiene for den digitale verdien for spenningen ved overflata ( $U_0$ ) og ved dybde 1 ( $U_1$ ), dybden ved overflata ( $D_0$ ) og dybden målt ved bunnen ( $D_1$ ) målt i meter.

I programmet har vi valgt følgende uttrykk for de fire verdiene:

// Skriv inn de avleste og målte verdiene for spenning og temperatur her

```
float U_0 = 4.00;           // U0 (verdi for spenningen ved overflata (“Analogt trykk”))
float U_1 = 12.00;         // U1 (verdi for spenningen ved bunnen (“Analogt trykk”))
float D_0 = 0.00;          // D0 (Kalibreringsdybde ved overflata, 0,00 meter)
float D_1 = 0.92;          // D1 (Kalibreringsdybde ved bunnen, ca. 0,92 meter)
```

// På bakgrunn av disse verdiene leser programmet av og beregner følgende verdier:

```
int U_D5 = 0;               // U (avlest digitalverdi for spenning ved aktuell posisjon)
float D_beregn = 0;         // D (beregnet dybde ved aktuell posisjon)
```

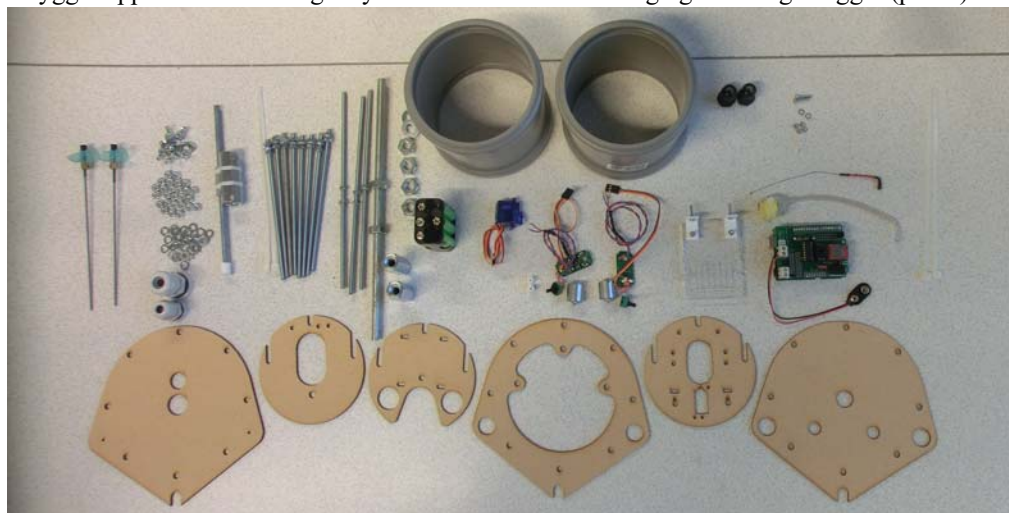


## 5 Bygging av skroget

I dette avsnittet skal vi gi et eksempel på hvordan en ROV kan bygges opp med billige materialer. Det må presiseres at dette ikke er en endelig versjon som er gjennomtestet, men utgangspunkt for videre arbeid mot en mer funksjonell utgave.

### 5.1 Komponentoversikt skrog med elektronikk

Bildet under viser de fleste av komponentene som skal til for å bygge opp et ROV-skrog. Skroget er bygget opp med en indre og en ytre struktur med innvendig og utvendige vegger (plater).





### Komponentliste:

- 2 stk. glidemuffe Ø 11 cm x 11 cm (Biltema)
- 1 stk. skjøtemuffe 6 mm
- 1 stk. skjøtemuffe 12 mm
- 2 stk skjøtemuffe 10 mm
- 8 stk. gjengestang 150 x 5 mm (Biltema)
- 3 stk. gjengestang 210 x 5 mm (Biltema)
- 1 stk. gjengestang 175 x 6 mm (Biltema)
- 1 stk. gjengestang 275 x 10 mm (Biltema)
- 40 stk. muttere M5
- 48 stk. stoppskiver M5
- 8 stk. muttere M10
- 8 stk. stoppskiver M10
- 8 stk. vingemuttere 5 mm
- 2 stk. 3 x 10 mm maskinskrue m/mutter M3
- 2 stk. 3,2 x 2 mm avstandstykke nylon (ELFA)
- 2 stk. PLA plastforing hjemmelaget
- 4 stk. 3 x 15 mm maskinskrue m/mutter M3
- 1 stk. kabelforskruing, 6-12 mm, 20 mm (Biltema)
- 1 stk. kabelforskruing, 4-8 mm M 16 mm (Biltema)
- 2 stk. kabelforskruing 1,5-4 mm, 12 mm (ELFA)
- 1 stk. batteripakke 6 x AA (ELFA)
- 2 stk. propeller (China)
- 2 stk. festeanordning propeller (China)
- 2 stk sveisetråd 2 mm
- 2 stk. PLA skjøtemuffe
- 2 stk. PLA endestopper for propell
- 1 stk. endeplate foran (akryl)
- 1 stk. midtplate (akryl)
- 1 stk. endeplate bak (akryl)
- 1 stk. indre plate foran (MDF 3 mm)
- 1 stk. indre plat midt (MDF 3 mm)
- 1 stk. indre plate bak (MDF 3 mm)
- 1 stk. shield-kort med trykk- og temp. sensor
- 1 stk. Arduino UNO (ELFA)
- 2 stk PLA-holder for kontroller kort
- 2 stk. rød ledning 30 cm
- 2 stk. sort ledning 30 cm
- 1 stk rød ledning en-tråd for feste
- 1 stk. krympestrømpe tynn
- 1 stk krympestrømpe tykk

## 5.2 Byggebeskrivelse av skroget

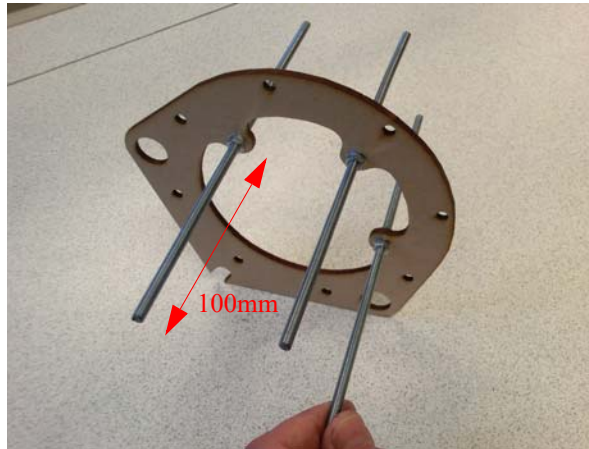
Vi skal nå vise et eksempel på en eksperimentoppstilling av en ROV bygget opp med to 110 mm rørmuffer fra Biltema. Det er stort sett brukt deler som skal være billige og lette å få tak i. Under veis vil vi påpeke kritiske punkter i designet slik det er skissert. Tekst skrevet i kursiv er byggeveiledning, mens normal skrift gir bakgrunnsinformasjon.

### 1. Gjengestenger for montering av indre seksjons vegger

Vi har valgt å bygge opp ROV med en indre fast struktur festet til den ytre seksjons-veggen i midten. Den indre strukturen består av tre 3,3 mm MDF-plater som henholdsvis holder elektronikk og servoen for vektforskyvningen (bakre), som holder batteriet (midtre) og som lagrer opp gjengestangen for vektforskyvningen (fremre). Disse tre platene kan forskyves fritt i forhold til hverandre.



Start med å monter de tre 21 cm lange gjengestengene i den ytre seksjons-veggen i midten (akryl) slik at de stikker like langt ut på begge sider. Merk at på bildene er akryl-platene erstattet med MDF-plater for lettere å sees.



## 2. Monter batteripakken

En 9V batteripakke er montert på den indre midtre seksjons-veggen. Dette gjør at vi i prinsippet kan droppe USB-kabelen opp til overflata dersom vi nøyer oss med at data kun lagres på SD-kortet. De indre monteringsplatene eller veggene er utstyrt med spalter i stedet for hull slik at de lett kan tas av og byttes individuelt. Et viktig moment er at batteripakken utgjør en betydelig vekt i den ene enden av ROV som bør kompenseres for, for å unngå skjevheter i vannet. *Batteripakken monteres med to lange plast-strips som vist på figuren under.*

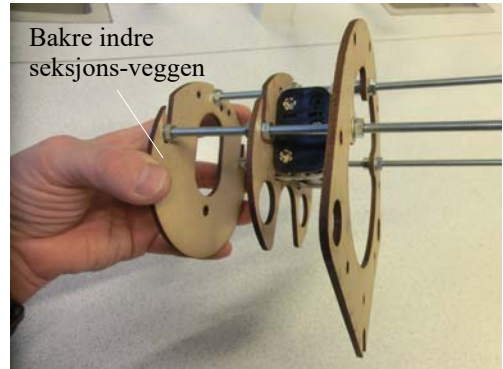
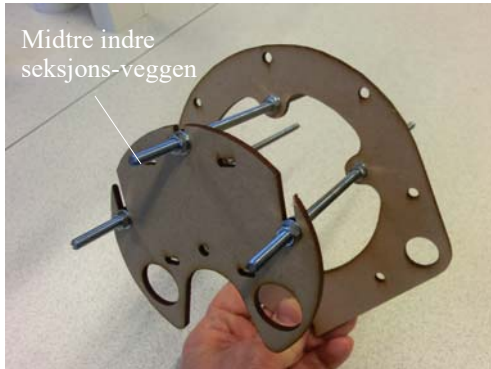


## 3. Montering av indre midtre seksjon

Midtre seksjon må i praksis monteres ganske langt bak for å gi plass til USB-kontakten som skal inn på kontrollert-kortet. Dette medfører som nevnt at vekten blir ganske skjevt fordelt. På bildet under er midtseksjonen plassert litt for langt foran og må trekkes tilbake når fremre



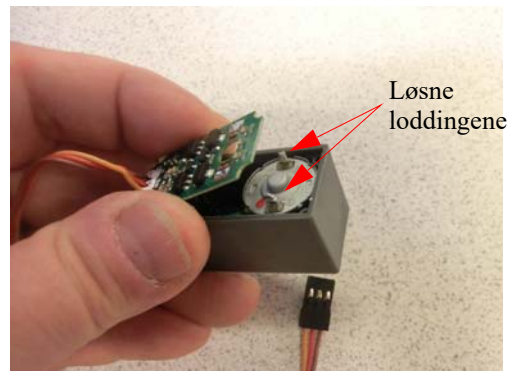
indre seksjons-vegg skal monteres. *Tre mutrene og skivene inn på gjengestengene slik at seksjonen kan festes i ønsket posisjon, ev. juster etter behov. Det legges en stoppskive på hver side av seksjons-veggen.*



#### 4. Klargjøring av motorer med forlengelsesledning

Motorene er servomotorer fra HobbyKing, HK15138. Disse er normalt 180° servoer som modifiseres til å bli 360°, dvs. servoer som roterer kontinuerlig som motorer.

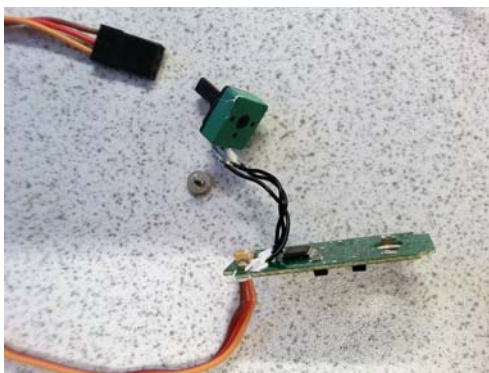
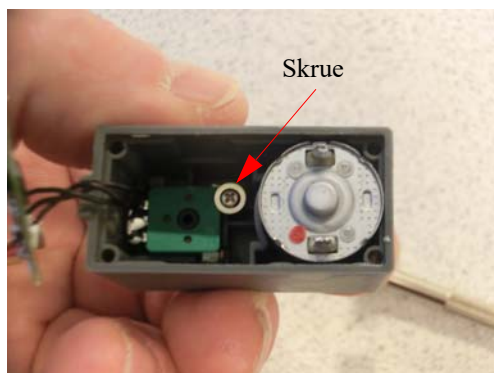
*Skru opp de fire skruene på baksiden og ta av dekslene. Ta av tannhulene. Løsne loddingene som holde motoren og løft av kretskortet.*



**OBS!** Merk hvilket tilkoblingspunkt på kortet som er koblet til tilkoblingen på motoren merket med en rød prikk. Det er viktig at disse forbindes på samme måte etter at forlengelsesledningen er tilkoblet.



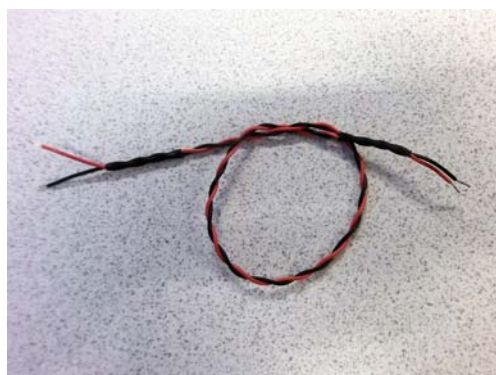
*Fjern skruen som holder potensiometeret og ta ut potensiometeret.*



*Dra ut motoren med en tang.*

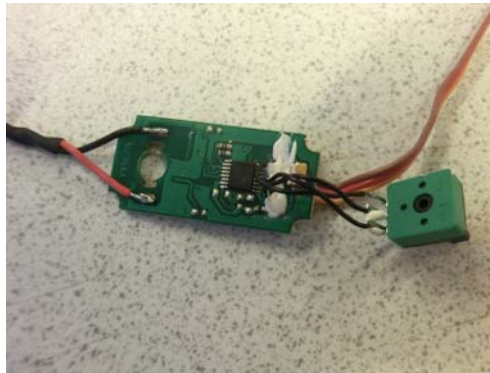


*Lag en tvinnet to-leder med en rød og en svart ledning. Bruk krympestrømpe i hver ende for å holde ledningene samlet.*





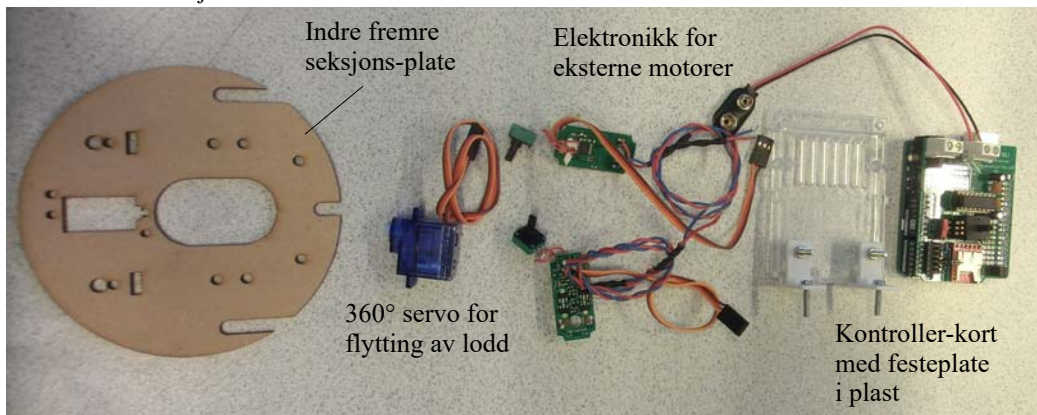
*Snu kortet og lodd ledningene på baksiden.*



*Når ledningene skal kobles til motoren skal den røde ledningen kobles til motortilkoblingen merket med den røde prikken.*

#### 5. Montering av bakre indre seksjons-veggen

Bildet over til høyre viser monteringen av bakre indre seksjon. Siden bildet ble tatt er hullet i veggen endret noe. *Bakre indre seksjon monteres med mutere og stoppskiver foran og bak som midtseksjonen.*



#### 6. Montering av elektronikk for styring av motorer

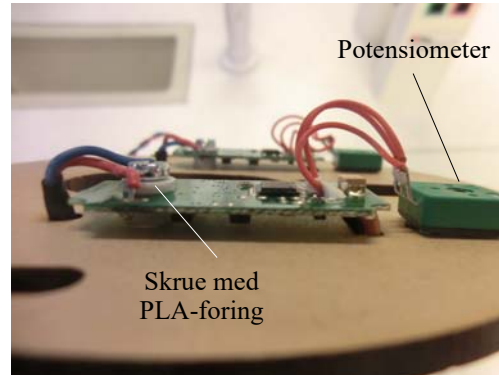
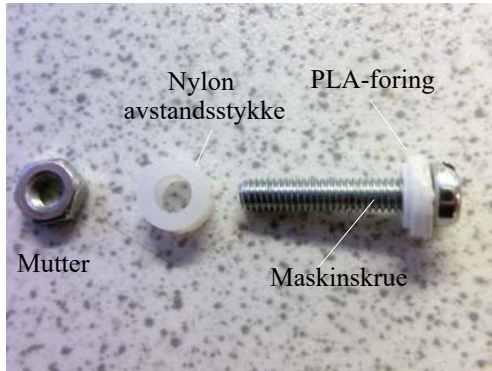
Styreelektronikken for de to utvendige motorene er tatt ut av to 180° servoer. Disse styres normalt av en puls med en gitt lengde. Lengden bestemmer vinkelutslaget. Et internt potensiometer sørger for en mekanisk/elektrisk tilbakekobling som posisjonerer motoren. Når en slik servo demonteres, forsvinner den mekaniske tilbakekoblingen og motoren vil rotere fritt med en hastighet bestemt av pulslengden. Potensiometeret vil bestemme pulslengden for null rotasjon. Det vanlige er at en pulslengde på 1500  $\mu\text{sek}$  gir stillstand. Økes pulslengden ut over dette går den framover, reduseres den under denne verdien vil den gå bakover.

*De to potensiometrene presses på plass i hullene og de to kretskortene skrues fast i den bakre*



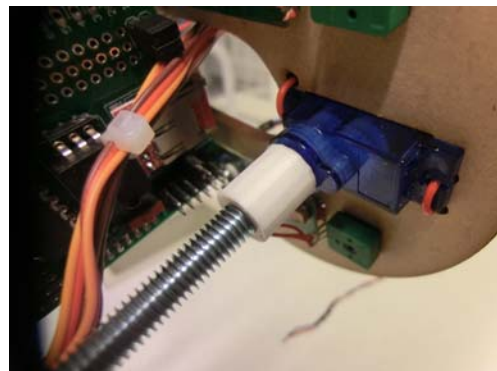
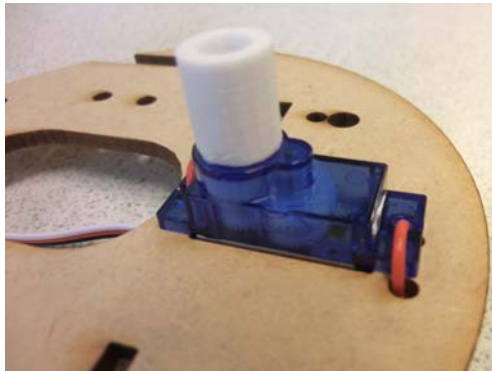


ytre veggen med en maskinskrue med mutter. Kretskortet løftes opp fra plate med et nylon avstandsstykke og en PLA-foring legges mellom kretskortet og skruehodet som vist på figuren under. Pass på at innsnevringen til foringen går ned i hullet på kretskortet. Ledningen som skal til kontrollkortet (m/3 polt hylsekontakt) føres ut gjennom hull i veggen. Likeledes føres ledningene til motorene gjennom et rundt hull i veggen.



#### 7. Montering av 360° servomotor for forskyvning av balanseledd

Servomotoren presses ned i plata slik at akslingen peker bakover. Servoen fastes ved to ledningsstumper som tres gjennom festehullene og gjennom plata og tvinnes på baksiden og strammes. Den hvite skjøtemuffa monteres på servomotorens aksling





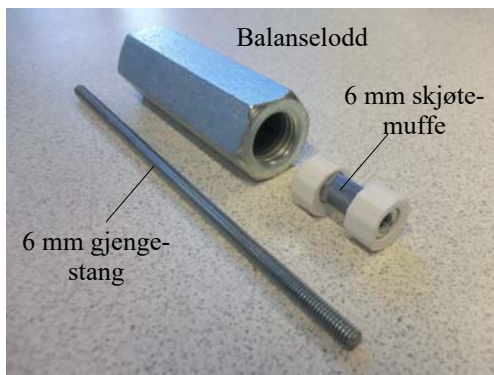
## 8. Montering av skrue og balansevekt

**OBS!** Dette er et springende punkt. For at ROV'en skal kunne bevege seg opp og ned i vannet må den tiltes. For denne modellen har vi valgt å forskyve en vekt langs en skrue (gjengestand). Det er imidlertid noe uklart hvor stor denne vekten bør være for å utgjøre en tilstrekkelig helningsvinkel.

Her har vi endt opp med et lodd på ca. 375 g som kan forskyves fram og tilbake mellom endene til ROV'en. For at loddet skal skru seg bortover skruen, må det festes eksentrisk om akslingen slik at tyngdekraften vil holde det i posisjon.



Vi har vurdert to løsninger. Den ene baserer seg på å montere to eksentriske muffe som skyves inn i balanseloddet som antydnet på bildet under. De to eksentriske muffene er skrevet ut på 3D-printer. Eksentrisiteten er litt liten slik at man kan risikere at loddet blir med rundt istedet for å skru seg langs gjengestangen



Den andre løsningen er å henge loddet under den 6 mm skjøtemuffa. Denne løsningen er mer eksentrisk slik at loddet lettere skrur seg langs gjengestangen. På den andre siden er det så pass lett at det neppe vil ha den ønskede virkningen. Vi velger derfor å gå videre med den først omtalte løsningen med et tungt balanseloddet.

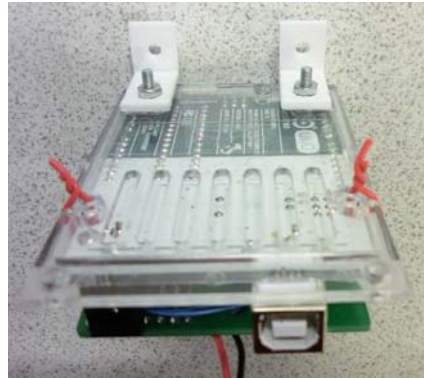
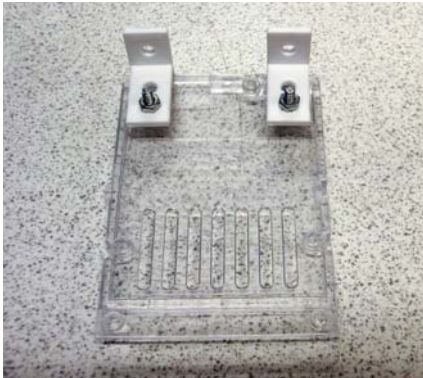




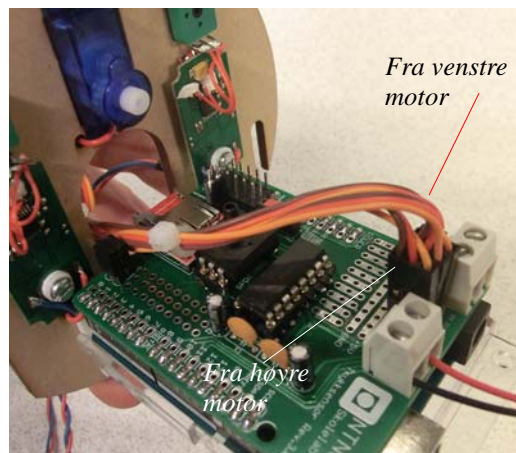
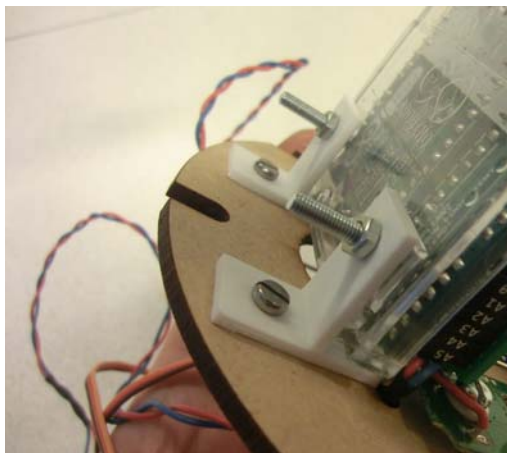
Monter PLA-eksentrisitetsmuffene på den 6 mm skjøtemuffa og press den inn i balanseloddet. Om det går tregt, fil av kantene på eksentrisitetsforingene. Pass på at eksentrisiteten er på samme siden av skjøtemuffa (6 mm). Skru gjengestaget inn i skjøtemuffa og fest den akslingen til muffa på servomotoren.

#### 9. Montering av kontrollerkort

Kontrollerkortet skal monteres øverst på fremre indre seksjons-vegg ved hjelp av to hvite braketter (vist til venstre på bildet under) og 3 mm skruer med mutter. Fest kortet til plastramma med ledninger (røde ledninger som vist på bildet under til høyre) slik at USB-kontakten peker bakover og SD-kortleseren peker forover.



Braketten skrues deretter fast i seksjons-veggen som vist på figuren under. Deretter føres ledningene fra servomotorene fram gjennom hullet i midten og trykkes ned i stiftkontakten mellom de to koblingsplintene på kontrollerkortet (tilkoblingspunktene for strømtilførsel)..

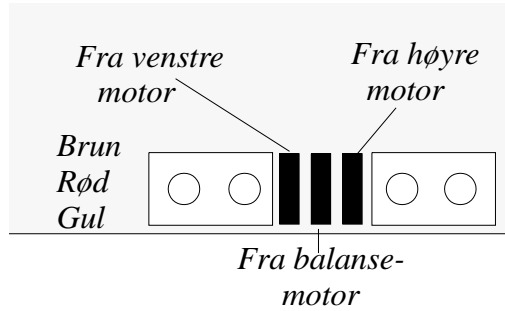




Ledningen fra venstre motor kobles til venstre stiftlist med den brune ledningen lengst foran

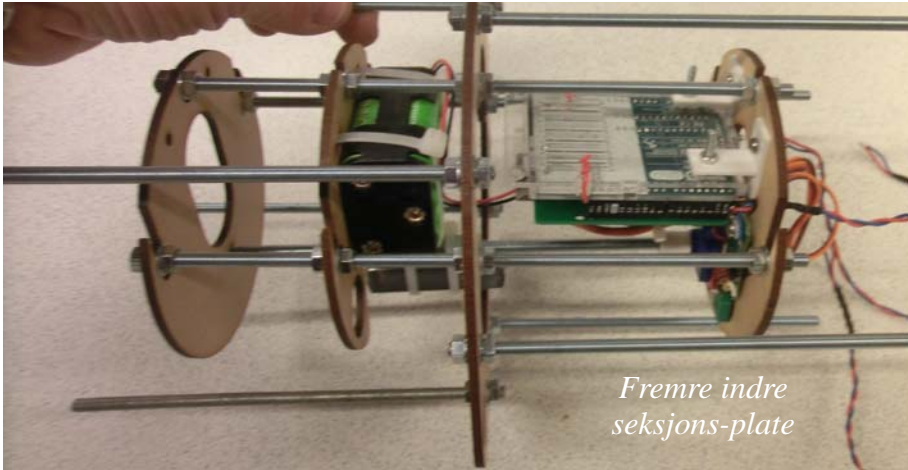
Ledningen fra høyre motor kobles til høyre stiftlist med den brune ledningen lengst foran

Ledningen fra balansemotoren kobles til stiftlista i midten med den brune ledningen lengst foran.



#### 10. Montering av fremre indre seksjons-vegg

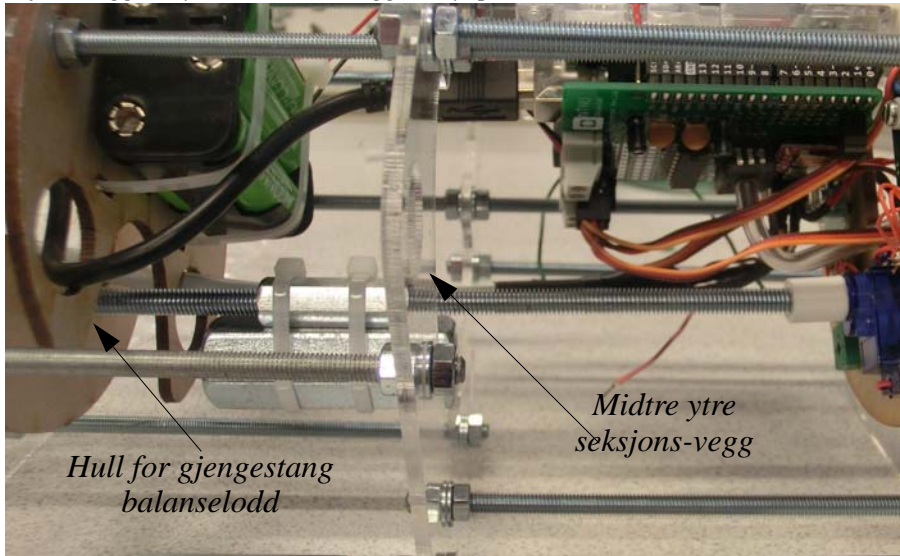
Så monteres fremre indre seksjons-vegg. Pass på at gjengestangen for balanseloddet kommer inn i hullet i bakre indre seksjons-vegg.



Legg merke til at ledningene til motorene kommer ut gjennom hull over kretskortene med styringselektronikken.



*Juster avstanden mellom fremre og bakre seksjons-vegg slik at gjengestangen til balanse-  
loddet ligger godt an i hullet i bakre indre seksjons-vegg. Legg merke til at midtre ytre  
seksjons-vegg er byttet ut med en vegg i akryl på bildet under.*

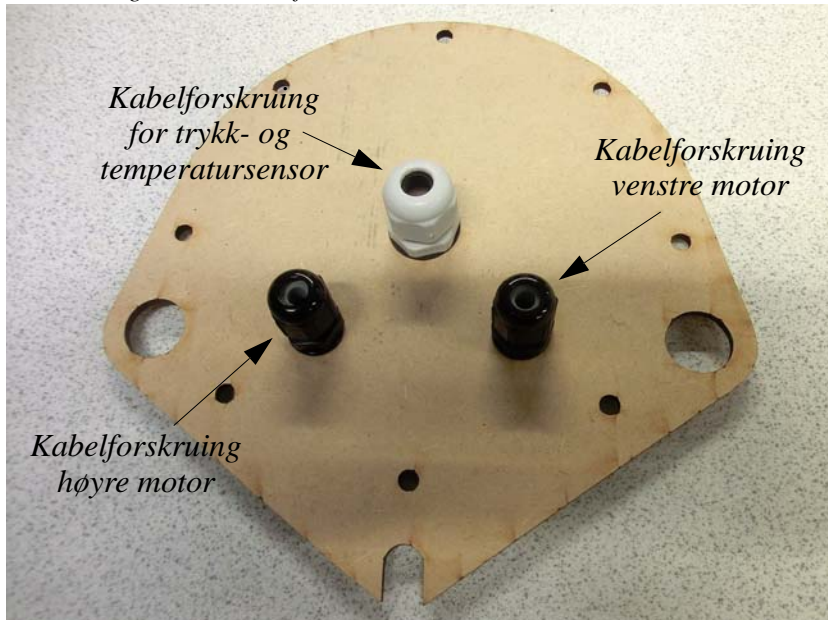




### 11. Montering av fremre ytre seksjons-vegg

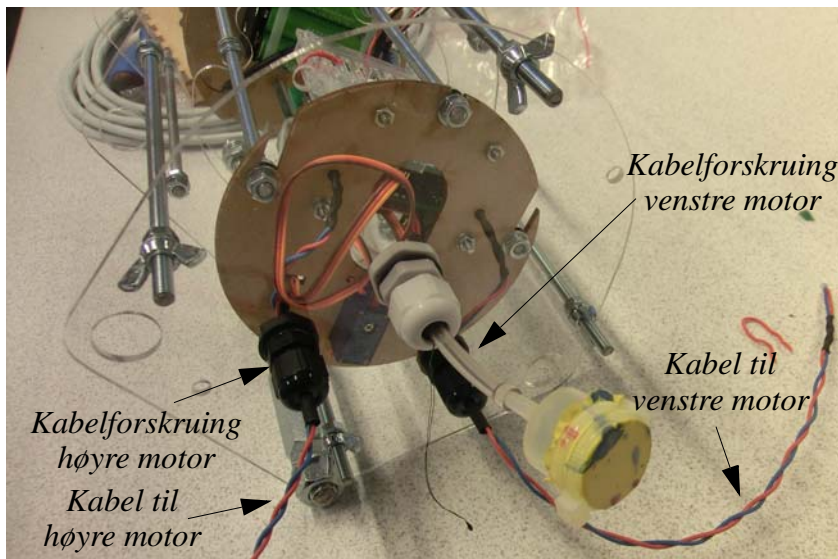
**OBS: Vi velger å vente med rørmuffene til etter testing av motorene.**

Monter to sorte (M12) og en hvit (M16) kabelforskruing. Pass på å monter o-ring på siden som vender mot vannet. Kabelforskruingene festes ved å skru til mutteren på innsiden. Mutteren strammes godt med en skiftnøkkel da den må være tett.





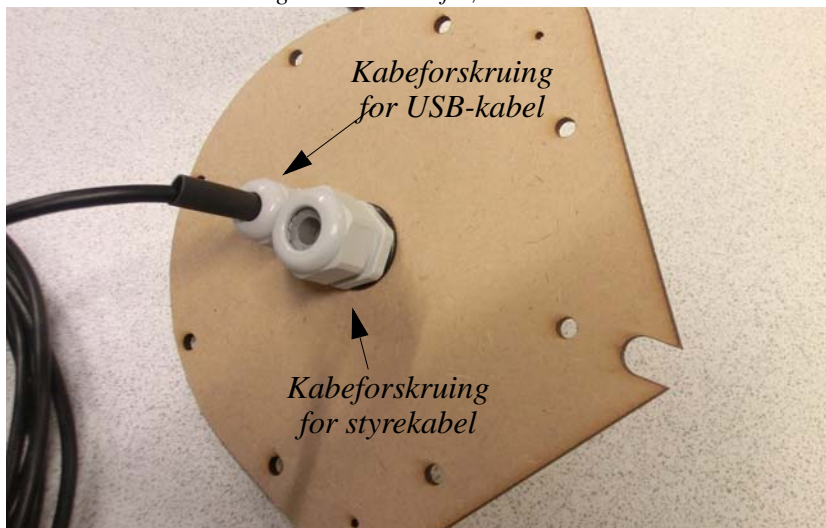
Bildet under viser hvordan det tar seg ut når fremre seksjons-vegg er laget av akryl, her er også trykk- og temperatursensor montert. Det er strengt tatt ikke nødvendig før motorene er testet.



Mutteren på framsiden strammes slik at gummipakningen slutter tett om kabelen. Vær oppmerksom på at disse lett kan slippe inn vann. For å unngå det kan man smøre litt fett rundt kabelen.

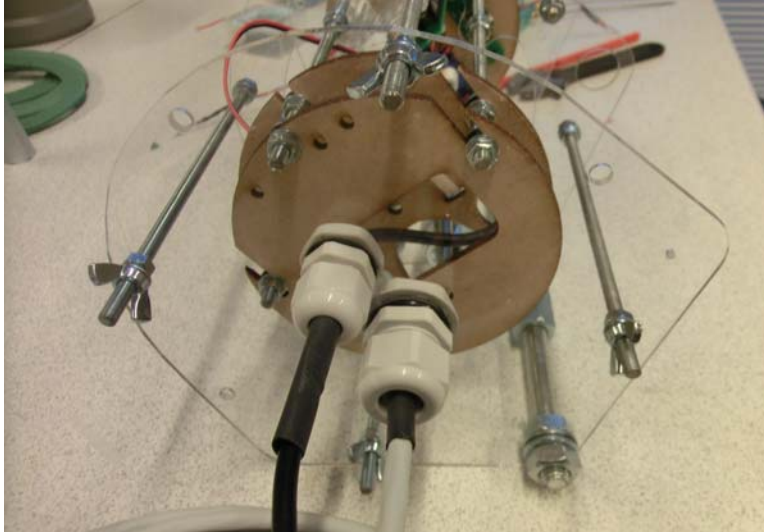
## 12. Montering av fremre ytre seksjons-veggen

Monter to hvit (M16 - for USB-kabel) og en hvit (M20 - for styrekabel) kabelforskruing. Pass på å monter o-ring på siden mot vannet. Kabelforskruingene festes ved å skru til mutteren på innsiden. Mutteren strammes godt med en skiftnøkkel da den må være tett.



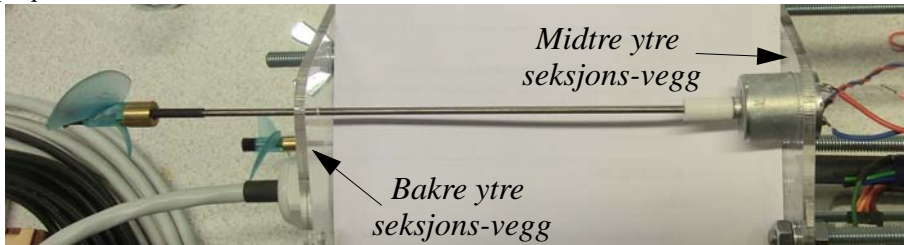


Bildet under viser bakre ytre seksjons-vegg i akryl hvor begge kablene er montert.



### 13. Montering av motorer

*Motorene presses inn i hullene på midtre eller fremre ytre seksjons-vegg. Det finnes også ekstra bakre ytre seksjons-vegger med hull for montering av motorene. Disse kan egne seg dersom man ønsker å forsøke å montere motorene helt bak med korte akslinger. Klipp til sveistråden slik at den blir passe lang og kan fungere som en aksling. Bruk motormuffa til å koble sammen akslingen og propellen. Hullet i bakre ytre seksjons-vegg er ment som støtte for akslingen. Ledningene til motorene loddes på, pass på at rød ledning kobles til loddeøret merket med en rød prikk. Propellen monteres med messingmuffene som følger med den fleksible akslingen. Et 3D-printet endestykke er laget for å plasseres ytterst på akslingen for at propellen ikke skal falle av.*

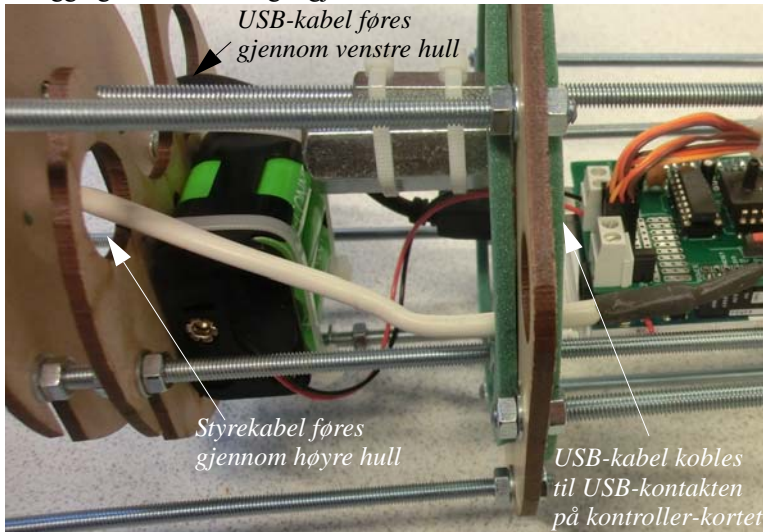




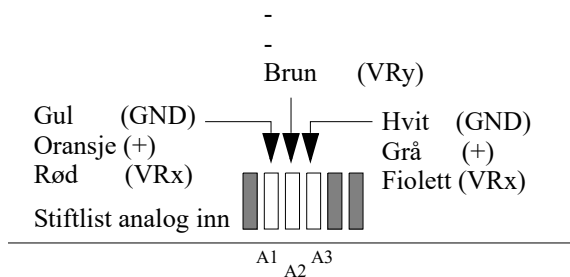


#### 14. Montering av USB- og styrekabel

Disse skal gå gjennom de to kabelforskruingene i bakre ytre seksjons-vegg som vist på bildet under. Videre går begge ledningene gjennom senterhullet i bakre indre seksjons-vegg. Styrekabelen og USB-kabelen skiller så lag. Styrekabelen går gjennom høyre hull i midtre indre seksjons-vegg og USB-kabelen går gjennom det venstre hullet.



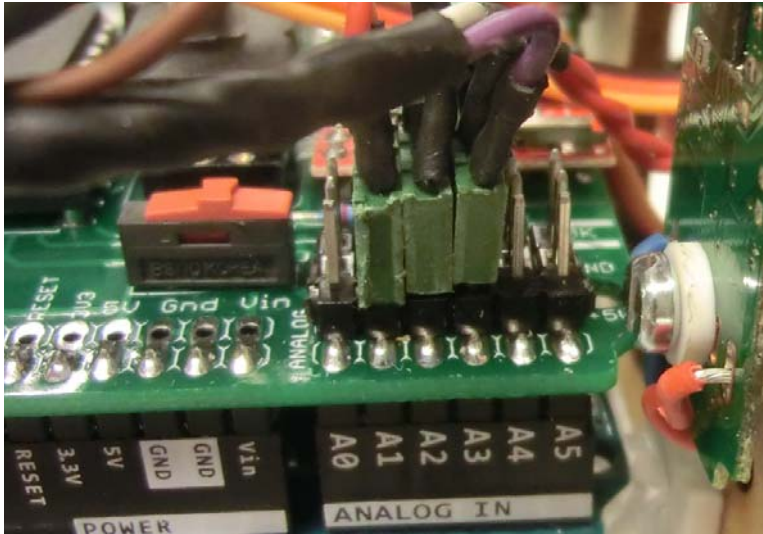
USB-kabelen kobles til USB-kontakten på kontrollert-kortet, mens styrekabelen kobles til stiftlistene til høyre på kontrollert-kortet, dvs de analoge inngangene, som vist på figuren til høyre. **OBS! Det viktig at de tre hylsekontaktene settes rett vei på stiftlista. Tar man feil her kan man risikere kortslutning av 5V..**



- Venstre Joy stick fremover/bakover kobles til analog inngang A1 (gul - oransje - rød)
- Venstre Joy stick sideveis kobles til analog inngang A2 (brun)
- Høyre Joy stick framover/bakover kobles til analog inngang A3 (hvit - grå - fiolett)



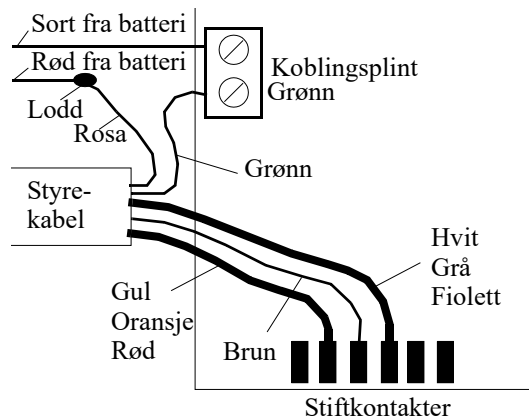
Bildet under viser hvordan det kan se ut når styrekabelen er koblet til de analoge inngangene. Legg også merke til den røde på/av bryteren til venstre for hylsekontaktene..



### 15. Oppkobling av På/Av bryteren på styre-konsollet

I utgangspunktet går det to ledninger fra batteripakken til koblingsplinten (den hvite kontakten) på kontrollerkortet. For å kunne slå av og på strømmen fra styrings-konsollet, må bryteren kobles inn i denne kretsen. Gjør følgende:

- Koble fra den røde ledningen fra batteripakken fra koblingsplinten (den hvite kontakten) på kontrollerkortet
- Lodd den røde ledningen fra batterikontakten til den rosa ledningen fra styrekabelen, isoler med krympestrømpe
- Koble den forlengede grønne ledningen til + inngangen til koblingsplinten



Nå er alt klart for å teste styrings-konsollet og motorene. Gå til avsnitt 5.5 på side 73 for testprosedyre.

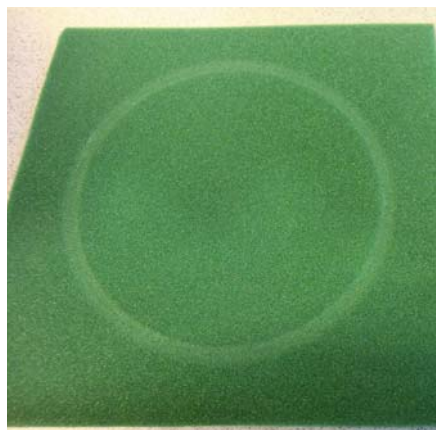
## 5.3 Montering av rørmuffer for våt-testing

### 16. Montering av bakre rørmuffe med pakninger

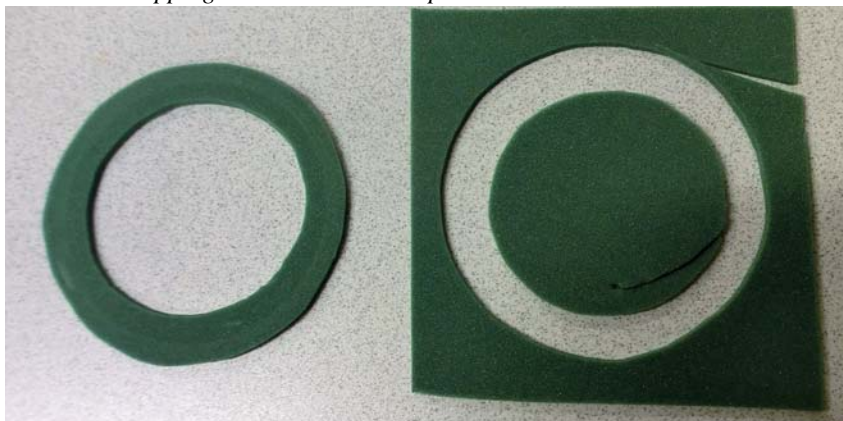
Vi har valgt å bruke mosegummi for å lage pakningene.



*Ta en rørmuffe og press den hardt ned mot mosegummien slik at det blir et tydelig merke. Klipp på utsiden og på innsiden slik at en får en ca. 1 cm bred ring av mosegummi, som vist på figuren under.*



*Resultatet etter klipping kan da bli som vist på bildet under.*





Lag fire slike ringer og plasser to av dem, en på hver side av midtre ytre seksjons-vegg som vist på bildet under.



Ta av fremre og bakre ytre seksjons-vegg og løsne ledningene. Tre så ledningene gjennom de ytterste pakningene og muffene og ut gjennom de respektive kabelforskruingene. Pass på at alle delene er med.





*Tre så muffen over de indre seksjons-veggene slik at muffa ligger an mot pakningen mot den ytre midt seksjons-veggen. Følg på med den bakre ytre seksjons-veggen.*



*Pass på at pakningen ligger riktig, sett på vingemutrene og skru til.  
Skru til tetningsmutterne på kabelforskruingene.*



#### **17. Montering av fremre rørmuffe med pakninger**

*Lag to pakninger som beskrevet under forrige punkt.*

*Dersom kontrollerkortet med sensorer er montert løsne den ytterste delen av trykksensoren fra plastslangen slik at denne kan slippe gjennom kabelforskruingen sammen med temperatursensoren.*



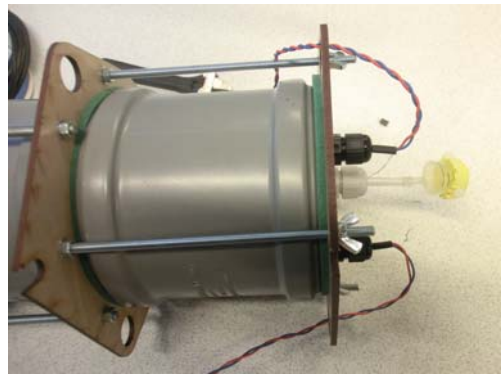
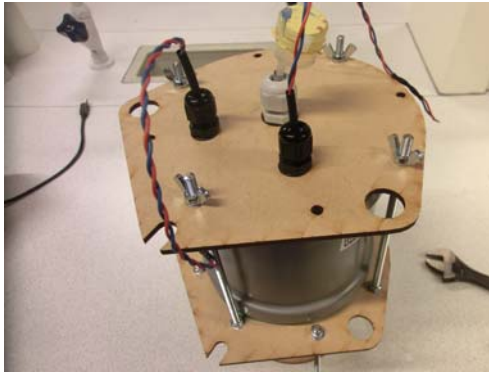
Legg pakningen mot midtre ytre seksjons-vegg på plass og tre rørmuffa inn over de indre seksjons-veggene. Legg på plass pakningen på toppen av rørmuffa.

Stikk ledningene til motorene gjennom de to sorte kabelforskruingene og før plastrøret til trykksensoren og temperatursensoren gjennom den hvite kabelforskruingen.

**OBS! Dersom man bare ønsker å teste motorene i vann, kan en ev. sette inn en blendingspinne i kabelforskruingen til trykksensoren inntil videre.**



Før den fremre ytre seksjons-veggen ned på gjengestengene og sett på vingemutrene. Skru til slik at det blir tett.



På bildet er de ytre seksjons-veggene av MDF, mens i den endelige versjonen benyttes akryl.

### 18. Montering av ballast

I underkant av ROVen monteres en ballast som består av en eller to skjøtemuffer (M10). Ev. kan tyngre muffe henges på som ballast under disse muffene. Ballastloddene kan skrues fra og tilbake for å justere tyngdepunktet. Stanga er lett å montere da den skyves inn i spaltene i de ytre seksjons-veggene.





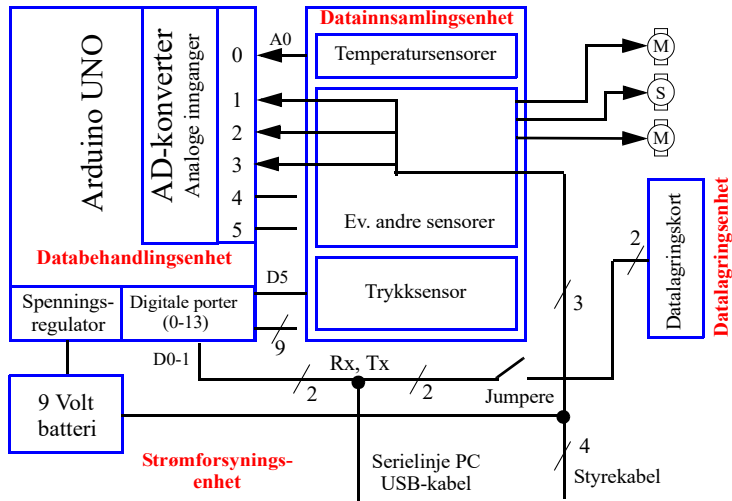
## 19. Monter motorene

Motorene monteres på nytt som omtalt i punkt 13, side 64. Ledningene må lodde på på nytt. Husk at rød ledning skal til motortilkobling med rød prikk.

## 5.4 Byggebeskrivelse styringskonsoll

### 5.4.1 Systemoversikt med sensorer og motorkontroll

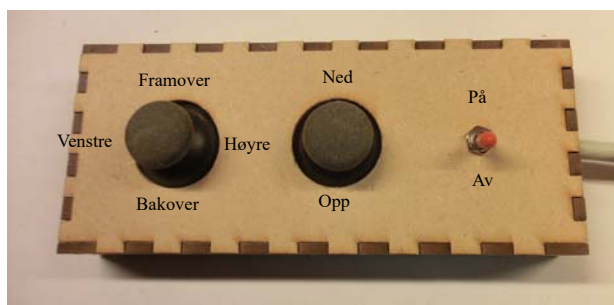
Vi har tidligere vist et blokkdiagram over elektronikken (avsnitt 1.3, side 12) Vi skal nå ganske kort inkludere styringen av servoen og motorene.



I denne figuren har vi inkludert de tre signalene fra Joy stickene og kabelen som kan slå av og på den interne strømforsyningen.

### 5.4.2 Brukerveiledning

Styrekonsollet består av to Joy sticks og en vippebryter:





- **Venstre Joy stick ...**  
styrer hastigheten til de to motorene.  
Venstre spak framover: Økende hastighet framover  
Venstre spak bakover: Økende hastighet bakover  
Venstre spak til venstre: Hastigheten til høyre motor økes, venstre motorhast. reduseres  
Venstre spak til høyre: Hastigheten til venstre motor økes, høyre motorhast. reduseres.
- **Høyre Joy stick ...**  
kjører tiltloddet framover og bakover  
Høyre spak framover: Loddet flyttes framover  
Høyre spak bakover: Loddet flyttes bakover
- **Bryter til høyre ...**  
slår av og på batteripakken i ROV'en  
Bryter fram, intern batteripakke slås på  
Bryter tilbake, intern batteripakke slås av

### 5.4.3 Oppbygging av styringskonsoll

Konsollet består av to billige Joy sticks, som også har en intern svitsj når Joy sticken trykkes ned (brukes ikke). I tillegg har konsollet en enkel av/på bryter for å slå av den interne batteripakken i ROV'en. En kabel på 7,5 meter med 12 enkeltledere, fører signalene mellom konsollet og ROV'en.

Følgende tabell gir en oversikt over oppkoblingen av Joy stickene på konsollet.

Venstre Joy stick Forover/bakover/til siden		Høyre Joy stick Opp (forover) - Ned (bakover)		Bryter	
Sort	SW 1	Grønn	SW 2	Lyserød	SW 3
Brun	VRy < - >	Blå	VRy < - >	Lysegrønn	SW 4
Rød	VRx	Fiolett	VRx		
Oransje	+ 5 V	Grå	+ 5 V		
Gul	GND	Hvit	GND		

I andre enden er følgende signaler koblet til tre 3-polte hylsekontakter som er tilkoblet tre av de analoge inngangene på Arduinoen.

Hylsekontakt A1 Venstre Joy stick		Hylsekontakt A2 Høyre Joy stick		Hylsekontakt A3	
Rød	VRx	Brun	VRy	Fiolett	VRx





Hylsekontakt A1 Venstre Joy stick		Hylsekontakt A2 Høyre Joy stick		Hylsekontakt A3	
Oransje	+ 5 V	-	-	Grå	+ 5 V
Gul	GND	-	-	Hvit	GND

Den midterste hylsekontakten har bare en tilkobling. Siden de andre to inneholder spenningene 5V og GND så er det tilstrekkelig.

## 5.5 Uttesting av styrefunksjon

Det kan være lurt å gjøre noen innledende tester før hele farkosten bygges sammen. Etter at styringskonsollet og USB-kablene er tilkoblet, kan man kjøre følgende program<sup>8</sup> for å teste at styrekonsollet og motorene fungerer:

// Styring av høyre og venstre servomotor for rettlinjet bevegelse.

```
#include <Servo.h>
```

```
Servo myleftservo; // Definer venstre servomotor
```

```
Servo myrightservo; // Definer høyre servomotor
```

```
Servo mytiltservo; // Definerer tiltservoen
```

```
int V_x1, V_y1, V_x2;
```

```
int korV_x1 = 508; // Korreksjonsverdi for Joy stick V_x1 i hvileposisjon legges inn her
```

```
int korV_y1 = 514; // Korreksjonsverdi for Joy stick V_y1 i hvileposisjon legges inn her
```

```
int korV_x2 = 500; // Korreksjonsverdi for Joy stick V_x2 i hvileposisjon legges inn her
```

```
void setup()
```

```
{
```

```
  pinMode(9, OUTPUT);
```

```
  pinMode(10, OUTPUT);
```

```
  pinMode(11, OUTPUT);
```

```
  myrightservo.attach(9); // attaches the right servo on pin 9 to the servo object
```

---

8. Programmet heter: ROV\_test (Ligger på minnepinne)



```
myleftservo.attach(11); // attaches the left servo on pin 11 to the servo object
mytiltservo.attach(10); // attaches the tilt servo on pin 10 to the servo object
Serial.begin(9600);
}

void loop()
{
  V_x1 = analogRead(1)-korV_x1;
  V_y1 = analogRead(2)-korV_y1;
  V_x2 = analogRead(3)-korV_x2;

  // Skriv Joy stick verdier til monitor
  Serial.print("V_x1: ");
  Serial.print(V_x1);
  Serial.print(", V_y1: ");
  Serial.print(V_y1);
  Serial.print(", V_x2: ");
  Serial.println(V_x2);

  delay(20);
  mytiltservo.writeMicroseconds(1500+V_x2); // Kjør tiltservo
  myleftservo.writeMicroseconds(1500+V_x1-V_y1); // Kjør venstre servo
  myrightservo.writeMicroseconds(1500+V_x1+V_y1); // Kjør høyre servo
}
```

### **Kalibrering av Joy stick**

Slik kalibreres motorer og Joy stickene:

1. La Joy stickene stå i hvileposisjon
2. Sett korreksjonen for nullstilling av Joy stickene i programmet til null (disse verdiene 508, 514, 500 settes til 0)
3. Kompiler og last opp programmet
4. Juster potensiometrene ved styreelektronikken slik at motorene står stille



5. Les av verdiene til Joy stickene i monitoren
6. Legg inn de avleste korreksjonsverdiene i programmet (under deklarasjonen av variable, der du i sted skrev 0)
7. Kompiler og last opp programmet igjen

Nå skal Joy stickene være kalibrert

8. Sjekk at motorene oppfører seg fornuftig når Joy stickene beveges.  
Legg ev. inn mulighet for å justere følsomheten til Joy stickene

Det må presiseres at dette er et enkelt testprogram ment for videreutvikling av brukerne. F.eks. bør man vurdere følsomheten av Joy stickene. Slik programmet nå er laget skal det svært lite til fra null respons til maks respons.

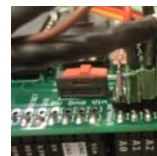
Det vil også være aktuelt å integrere styre og måleprogrammet hvilket kan by på noen utfordringer siden det pr. i dag ligger en ventekommando i måleprogrammet. En løsning som bør vurderes er at programmet har hovedfokus på styring for så å gå ut å utføre målinger i korte intervaller inni mellom styringen av motorene.

En kan også vurdere at måling kun gjøres når motorene ikke er aktive. En slik løsning vil redusere faren for målestøy fra motorene.

### **Uttesting av Av/På bryteren**

Slik testes om det innebygde batteriet og bryteren på styrekonsollet fungerer:

1. Koble til batteriet (husk å koble bryteren på konsollet inn i kretsen mellom batteriet og koblingsplinten, se punkt 15, side 66)
2. Slå på glidebryteren på kortet (rød og grå bryter inne på kortet, se bildet)
3. Last opp styreprogrammet og sjekk at motorene kan styres som forventet
4. Trekk ut USB-kabelen fra PC'en
5. Slå på strømmen med bryteren på styringskonsollet
6. Se om motorene kan styres med Joy stickene som forventet



Om dette fungerer burde alt være klart for uttesting vann.



## 6 utfordringer ved uttesting

Under arbeidet med målekortet og ROV-skroget bør man skrive en laboratoriejournal. Dette er særdeles viktig som underlag for problemløsning og videreutvikling av utstyret. Som tidligere nevnt er utstyret beskrevet i dette heftet kun en tidlig prototyp som i liten grad er uttestet<sup>9</sup>.

Følgende er to dokumenter som skal leveres inn etter at arbeidet er utført og som i stor grad cil bygge på laboratorierapport.

### 6.1 Rapport fra arbeidet (5 - 8 sider, 15%)

#### Måling av trykk og temperatur

Rapportering fra arbeidet på laboratoriet bør ha fokus på følgende:

1. Meget kort beskrivelse av kortets oppbygning og virkemåte, gjerne med bilder
2. utfordringer ved oppbygging av kortet vurderingen av byggeveiledningens egnethet
3. Erfaringer gjort under kalibrering og måling av trykk og temperatur.
4. En kortfattet konklusjon

#### Oppbygging av ROV-skroget

Rapportering fra arbeidet på laboratoriet bør ha fokus på følgende:

5. Kort beskrivelse av ROV'ens oppbygging og virkemåte, gjerne med bilder
6. utfordringer ved oppbyggingen av modellen
7. Erfaringer gjort under byggeprosessen
8. En kortfattet konklusjon

#### Uttesting

Beskriv hvilke utfordringer uttestingen av ROV bød på med hensyn til:

9. Evne til å holde vannet ute (lekkasje)
10. Evne og stabilitet i vannet
11. Motorenes evne til å endre farkostens bevegelse, sideveis, opp og ned
12. Tiltmekanismens evne til å endre farkostens stilling i vannet og til å kunne bevege seg opp og ned?

---

9. En laboratoriejournal er også helt avgjørende når man skal skrive rapport fra arbeidet og refleksjonsdokument som del av et studium eller videreutdanningskurs.



## 6.2 Didaktisk refleksjonsdokument (2 - 3 sider, 15%)

Refleksjonsdokumentet skal vurdere de didaktiske sidene ved å bruke en ROV med måleutrustning i ToF-faget.

### Måling av trykk og temperatur

1. Vurdering ved å la elevene bygge, feilfinne og anvende elektronikk av denne typen i ToF-faget
2. En kortfattet vurdering av hvilke deler av virkemåten som passer å ta med i undervisningen
3. Tanker om hvordan denne delen av prosjektet bør tilrettelegges for elevene.  
Hva bør være klart på forhånd, hvilke utfordringer bør elevene selv løse og hvordan bør prosjektet veiledes?

### Oppbygging av ROV-skroget

4. Vurdering av å la elevene bygge opp en slik farkost
5. Noen tanker om hvordan denne delen av prosjektet bør tilrettelegges for elevene.  
Hva bør være klart på forhånd, hvilke utfordringer bør elevene selv løse og hvordan bør prosjektet veiledes?
6. En kortfattet konklusjon av egnethet for undervisning i ToF-fagene  
Gi gjerne en oversikt over kompetansemål som kan oppfylles ved et slikt prosjekt

Det vil bli størst vekt på punkt 3 og 5



---

## 7 Referanser

- [1] Datablad UTI03 (SMARTEC)  
<https://store.comet.bg/download-file.php?id=14625>
- [2] Datablad SPD030G (SMARTEC)  
<https://store.comet.bg/download-file.php?id=2612>
- [3] Datablad NTC-motstand 10kOhm (B57550G1103F005, EPCOS)  
[https://www.elfa.se/Web/Downloads/\\_t/ds/B57550G-B57550G1\\_eng\\_tds.pdf?mime=application%2Fpdf](https://www.elfa.se/Web/Downloads/_t/ds/B57550G-B57550G1_eng_tds.pdf?mime=application%2Fpdf)



# Vedlegg A Programvare

## A.1 Program for uttesting av sensorene

```
#include <stdio.h>
#include "pins_arduino.h"

#define UTI_PIN 5

// -----Definisjon av variable temperaturmåling -----//
int U_Lav_A0 = 382; // U_L (digital verdi for spenningen på A0, registrert lav temperatur)
int U_Hoy_A0 = 686; // U_H (digital verdi for spenningen på A0, registrert ved høy temperatur)
float t_Lav = 12; // t_L (lav kalibreringstemperatur i grader C)
float t_Hoy = 41; // t_H (høy kalibreringstemperatur i grader C)

// På bakgrunn av disse verdiene leser programmet av og beregner følgende verdier:

int U_A0 = 0; // U (avlest digitalverdi for spenning på A0 ved aktuell posisjon)
float t_Beregn = 0; // t (beregnet temperatur ved aktuell posisjon)

// -----Definisjon av variable dybdemåling -----//

float U_0 = 4.00; // U_0 (digital verdi for spenningen ved overflata)
float U_1 = 12.00; // U_1 (digital verdi for spenningen ved bunnen)
float D_0 = 0.00; // D_0 (Kalibreringsdybde ved overflata, 0,00 meter)
float D_1 = 0.70; // D_1 (Kalibreringsdybde ved bunnen, ca. 0,7 meter)

// På bakgrunn av disse verdiene leser programmet av og beregner følgende verdier:

float U_D5 = 0; // U (avlest digitalverdi for spenning ved aktuell posisjon)
float D_beregn = 0; // D (beregnet dybde ved aktuell posisjon)

int i = 0;

void setup()
{
  Serial.begin(9600);
  pinMode(UTI_PIN, INPUT);
```



```
}

void loop()
{
  double Toff, Tref, Tx;
  i = i+1;

  //----- Leser pulsbreddene og beregner avlest spenning -----//

  if(ReadUTI(UTI_PIN,&Toff,&Tref,&Tx))
  {
    U_D5 = 500.0 * (float)(Tx - Toff) / (float)(Tref - Toff);

    D_beregn = ((D_1 - D_0)/(U_1 - U_0))*
              (U_D5 - U_0) + D_0;

    //-----Skriv ut for test og kalibrering -----//
    Serial.print("Nr.: ");
    Serial.print(i);
    Serial.print(", Analogt trykk: ");
    Serial.print(U_D5, 2);
    Serial.print(", Beregnet dybde: ");
    Serial.print(D_beregn, 2);
    Serial.println(" m");
    Serial.print("Digital temp.: ");
    Serial.print(U_A0);
    Serial.print(", Beregnet temp.: ");
    Serial.print(t_Beregn);
    Serial.println(" C");

    //----- Skriv ut for datainnsamling -----//
    /* Serial.print(i);
    Serial.print(", ");
    Serial.print(D_beregn, 2);
    Serial.print(", ");
    Serial.print(t_Beregn, 2);
    Serial.println(","); */
  }
}
```





```
else
{
  Serial.print("Not Found");
}

//----- Leser inn spenningen for temperaturen og beregner kalibrert temperatur -----//

U_A0 = analogRead(A0);
t_Beregn = ((t_Hoy - t_Lav)/(U_Hoy_A0 - U_Lav_A0))*
           (U_A0 - U_Lav_A0) + t_Lav;

delay(5000);
}

int ReadUTI(uint8_t pin, double * Toff, double * Tref, double * Tx)
{
  int state = HIGH;
  int i,startindex=-1;
  uint8_t bit = digitalPinToBitMask(pin);
  uint8_t port = digitalPinToPort(pin);
  uint8_t stateMask = (state ? bit : 0);
  unsigned long width[9] = {0,0,0,0,0,0,0,0};
  double noff=0, nref=0, nx=0;

  while ( (*portInputRegister(port) & bit) != stateMask);

  for(i=0;i<9;i++)
  {
    while ( (*portInputRegister(port) & bit) == stateMask)
      width[i]++;
    while ( (*portInputRegister(port) & bit) != stateMask)
      width[i]++;
  }

  for(i=0;i<9;i++)
  {
    if(i<6)
    {
```



```
if(width[i]<width[i+2]&&width[i]<width[i+3]&&width[i+1]<width[i+2]&&width[i+1]<width[i+3])
{
  startindex=i;
  i=9;
  //Serial.print("Startindex: ");
  //Serial.println(startindex);
}
}
}
if(startindex!=1) {
  *Toff=width[startindex]+width[startindex+1];
  *Tref=width[startindex+2];
  *Tx=width[startindex+3];

  return 1;
}
else
{
  return 0;
}
}
```

## A.2 Program for uttesting av styringskonsollet og motorene

// Styring av høyre og venstre servomotor for retlinjet bevegelse.

```
#include <Servo.h>
```

```
Servo myleftservo; // Definer venstre servomotor
```

```
Servo myrightservo; // Definer høyre servomotor
```

```
Servo mytiltservo; // Definerer tiltservoen
```

```
int V_x1, V_y1, V_x2;
```

```
int korV_x1 = 508; // Korreksjonsverdi for Joy stick V_x1 i hvileposisjon legges inn her
```

```
int korV_y1 = 514; // Korreksjonsverdi for Joy stick V_y1 i hvileposisjon legges inn her
```

```
int korV_x2 = 500; // Korreksjonsverdi for Joy stick V_x2 i hvileposisjon legges inn her
```

```
void setup()
```

```
{
```

```
  pinMode(9, OUTPUT);
```

```
  pinMode(10, OUTPUT);
```

82 ROV – med trykk- og temperaturmåling



---

```
pinMode(11, OUTPUT);
```

```
myrightservo.attach(9); // attaches the right servo on pin 9 to the servo object
myleftservo.attach(10); // attaches the left servo on pin 10 to the servo object
mytiltservo.attach(11); // attaches the tilt servo on pin 11 to the servo object
Serial.begin(9600);
}
```

```
void loop()
```

```
{
  V_x1 = analogRead(1)-korV_x1; // Korreksjonsverdi for Joy stick V_x1 i hvileposisjon
  V_y1 = analogRead(2)-korV_y1; // Korreksjonsverdi for Joy stick V_y1 i hvileposisjon
  V_x2 = analogRead(3)-korV_x2; // Korreksjonsverdi for Joy stick V_x2 i hvileposisjon

  // Skriv Joy stick verdier til monitor
  Serial.print("V_x1: ");
  Serial.print(V_x1);
  Serial.print(", V_y1: ");
  Serial.print(V_y1);
  Serial.print(", V_x2: ");
  Serial.println(V_x2);

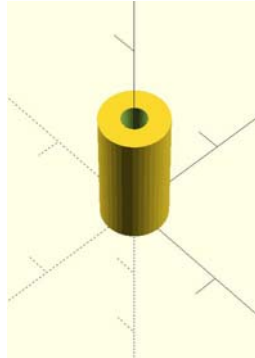
  delay(20);
  mytiltservo.writeMicroseconds(1500+V_x2); // Kjør tilt servo
  myleftservo.writeMicroseconds(1500+V_x1-V_y1); // Kjør venstre servo
  myrightservo.writeMicroseconds(1500+V_x1+V_y1); // Kjør høyre servo
}
```



## Vedlegg B 3D-printede deler

### B.1 Muffe mellom motor og aksling

Denne muffa danner overgangen mellom motorakslingen og propellakslinge.

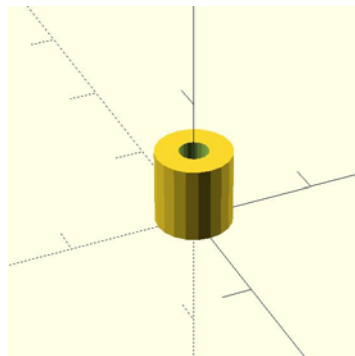


#### OpenSCAD-kode

```
difference()
{
cylinder($fn = 40, h = 12, r1 = 3, r2 = 3, center = false);
translate([0,0,-0.1]) cylinder($fn = 40, h = 4, r1 = 1.82, r2 = 1.82, center = false);
translate([0,0,2.9]) cylinder($fn = 40, h = 9.2, r1 = 1.125, r2 = 1.125, center = false);
}
```

### B.2 Endestykke for propellaksling

Endestykket sørger for at propelleing ikke faller av akslingen.





### OpenSCAD-kode

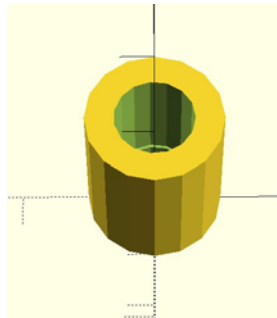
```
difference()
{

translate ([0, 0, 0]) cylinder($fn = 20, h = 6, r1 = 3, r2 = 3, center = false);
translate ([0, 0, 1]) cylinder($fn = 20, h = 6, r1 = 1.19, r2 = 1.19, center = false);

}
```

### B.3 Muffe mellom servomotor og gjengestang for tiltlodd

Denne muffa kobler sammen akslingen til servoen og gjengestanga som forskyver loddet.



### OpenSCAD-kode

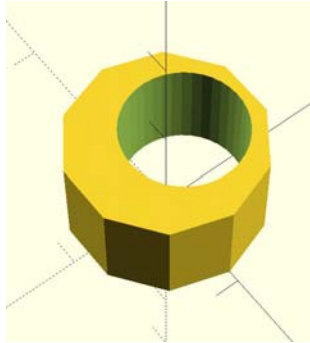
```
difference()
{

cylinder(h = 12, r1 = 5, r2 = 5, center = false);
translate([0,0,-0.1]) cylinder($fn = 15, h = 3.1, r1 = 2.6, r2 = 2.7, center = false);
translate([0,0,2.9]) cylinder($fn = 15, h = 9.2, r1 = 2.9, r2 = 2.9, center = false);

}
```

### B.4 Eksentrisitet for tiltlodd

Denne eksentrisiteten sørger for å holde tilloddet eksentrisk slik at kan skrus langs gjengestangen.

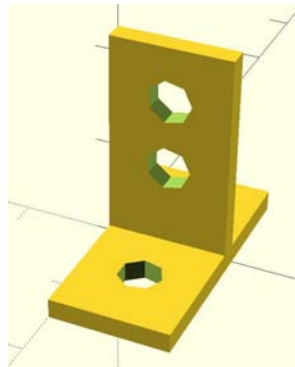


### OpenSCAD-kode

```
difference()
{
cylinder($fn = 10, h = 10, r1 = 8.9, r2 = 8.9, center = false);
translate([0,1.9,-0.1]) cylinder($fn = 40, h = 11, r1 = 5.6, r2 = 5.6, center = false);
}
```

### B.5 Brakett for kontrollkort

Braketten fester kontrollerkortet til fremre indre seksjons-veggen.



### OpenSCAD-kode

```
rotate ([90, 0, 0])
{
difference()
{
```



```
union()
{
cube(size = [12,22.5,2], center = true/false);
translate([0,0,-10]) cube(size = [12,2,25], center = true/false);

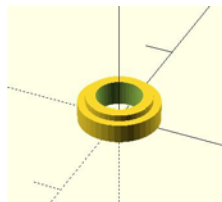
}

translate([6,17,-1 ]) cylinder(h = 4, r1 = 2.2, r2 = 2.2, center = true/false);
translate([6,10,-1 ]) cylinder(h = 4, r1 = 2.2, r2 = 2.2, center = true/false);

rotate ([90, 0, 0])
{
  translate([6,8,-3]) cylinder(h = 4, r1 = 2.2, r2 = 2.2, center = true/false);
}
}
}
```

## B.6 Foring for å holde kretskort for motorstyring

Foringen gjør at kortet sitter godt og at skruen ikke kortslutter noen elektriske baner.



### OpenSCAD-kode

```
difference()
{
union()
{
  cylinder($fn = 40, h = 2, r1 = 2.6, r2 = 2.6, center = false);
  cylinder($fn = 40, h = 1.5, r1 = 3.2, r2 = 3.2, center = false);
}
}
translate([0,0,-0.1]) cylinder($fn = 40, h = 4, r1 = 1.80, r2 = 1.80, center = false);
```



---

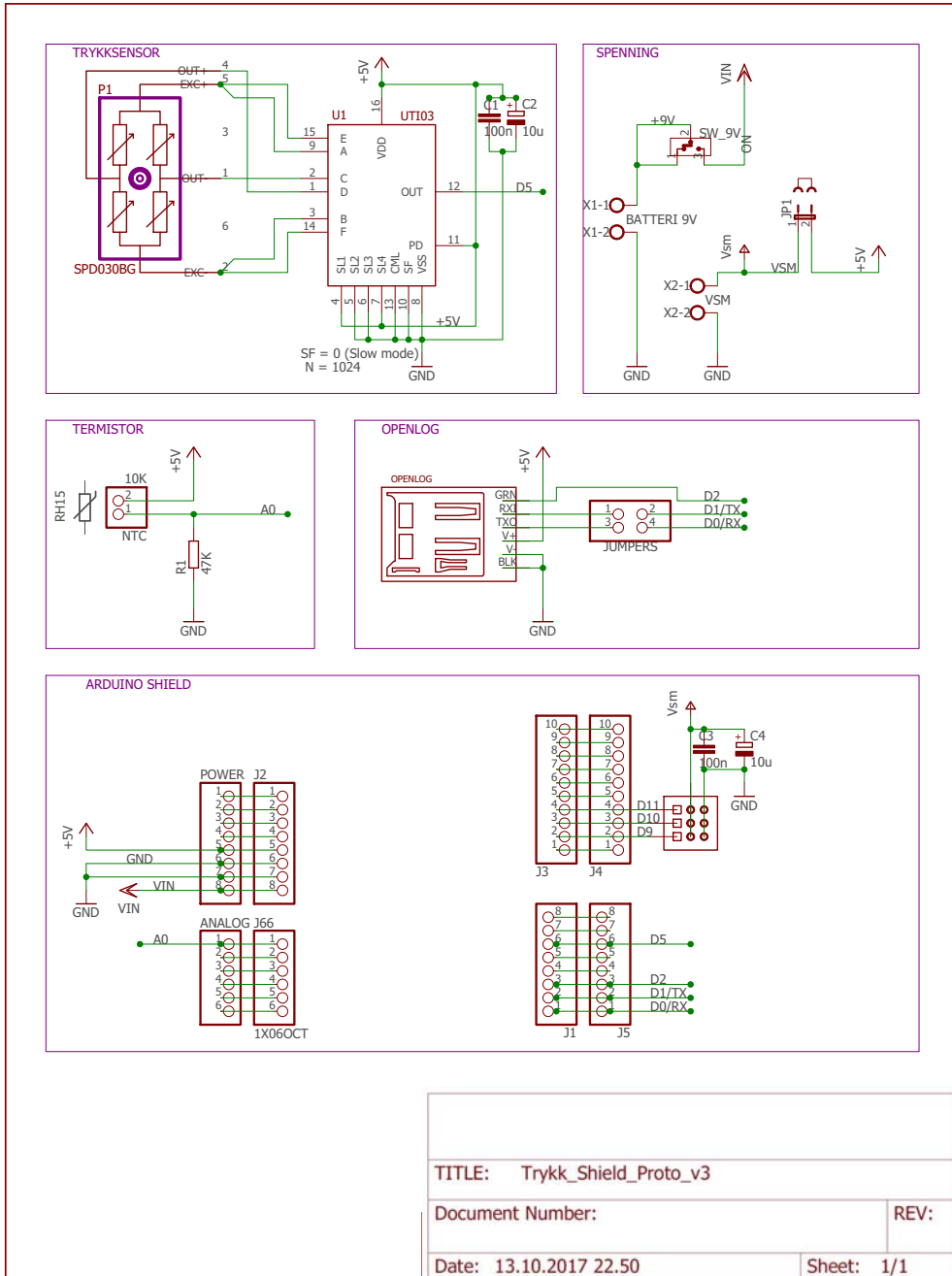
}





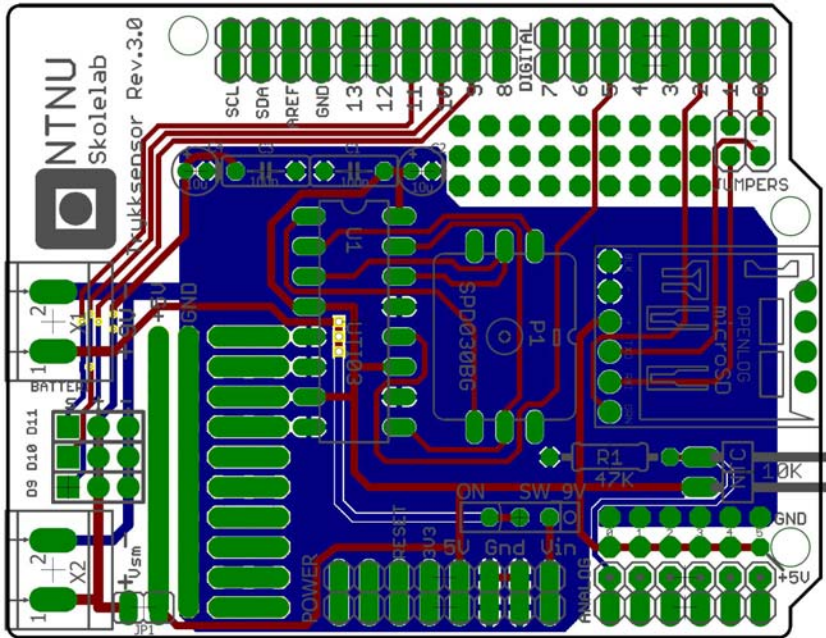
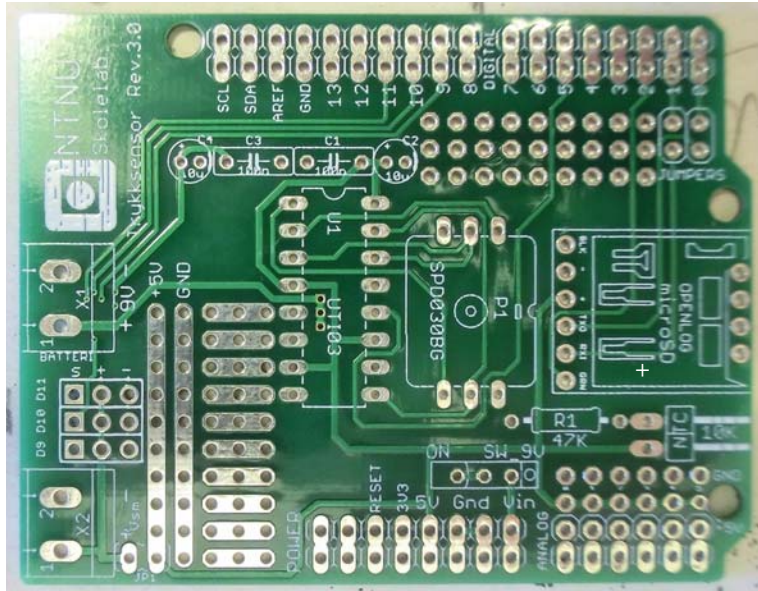
# Vedlegg C Kretskort

Kretsskjema Rev. 3.0. Layout er utført av Steinar Haugen





Layout Rev. 3.0



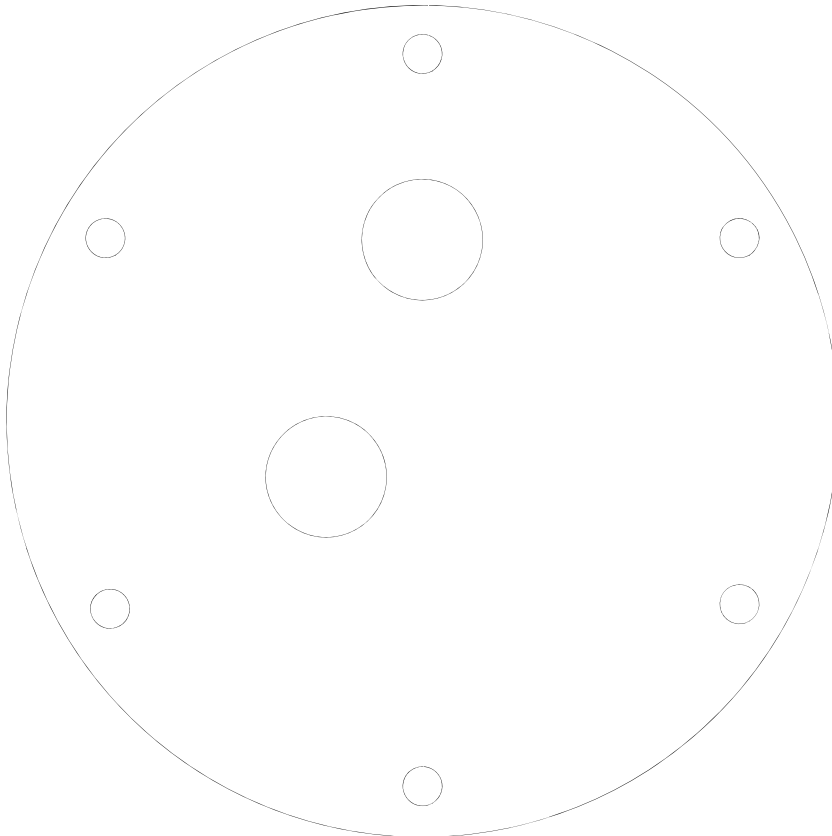
90 ROV – med trykk- og temperaturmåling



## Vedlegg D

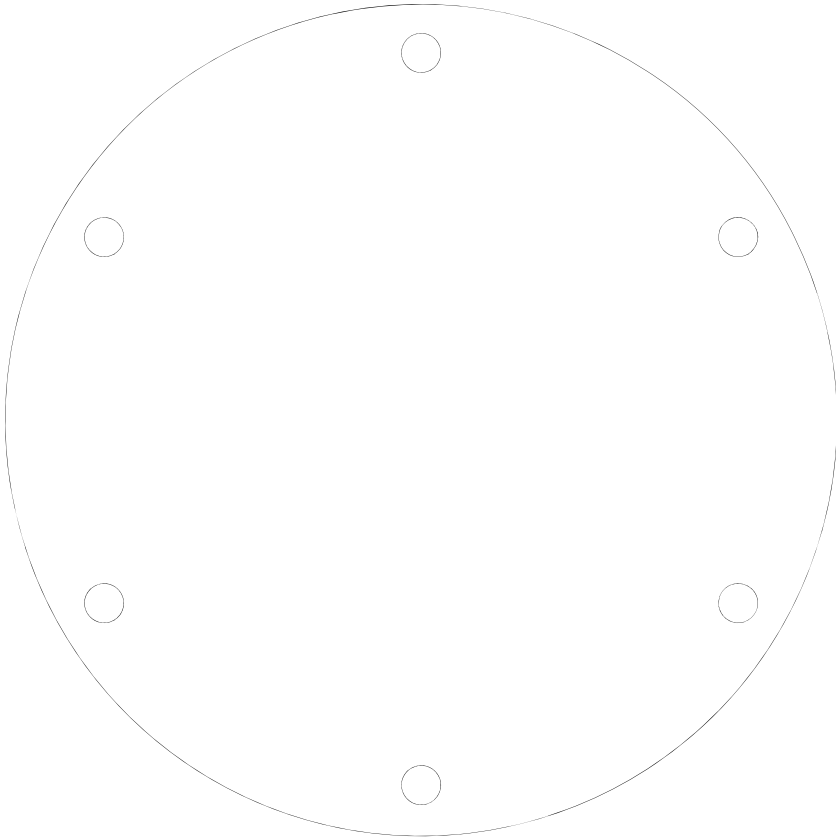
### D.1 Design av endeplatene til målejigg

Endeplatene er skåret i 4 mm pleksiglass (uten klor) kjøpt hos Biltema. Dette glasset er godkjent for bruk i laserkutter da det er klorfritt. Diameteren til platen er 110 mm og hullene langs kanten er Ø 5,5 mm og tilpasset Ø 5 mm gjengestål. Hullene til kabelforskruingen er Ø16 mm..



Siden lengden av den gjengete delen er for kort til å kunne stikke gjennom 4 mm plastglass, så må glasset gjøres tynnere med Ø20 mm flatbor. Alternativt har vi laget endeplater med tykkelse 2 mm.

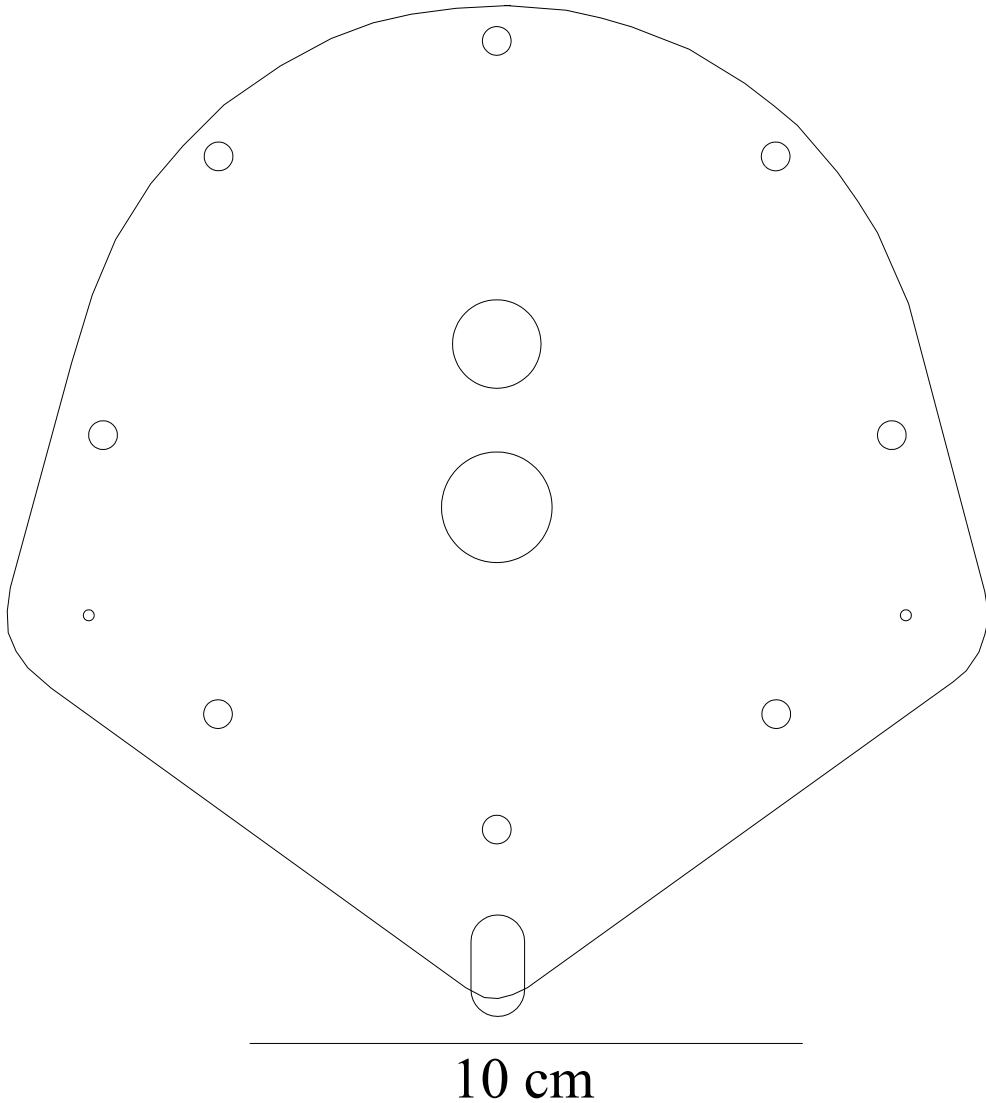


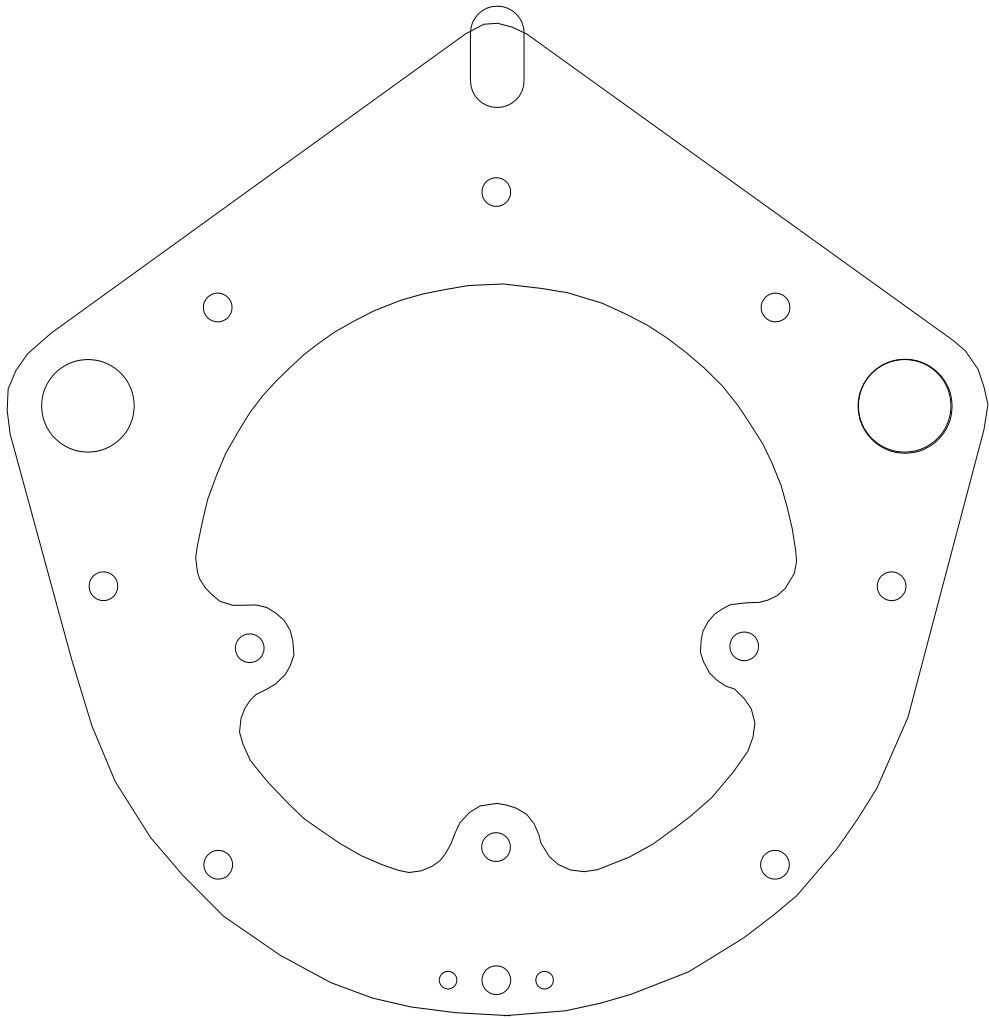




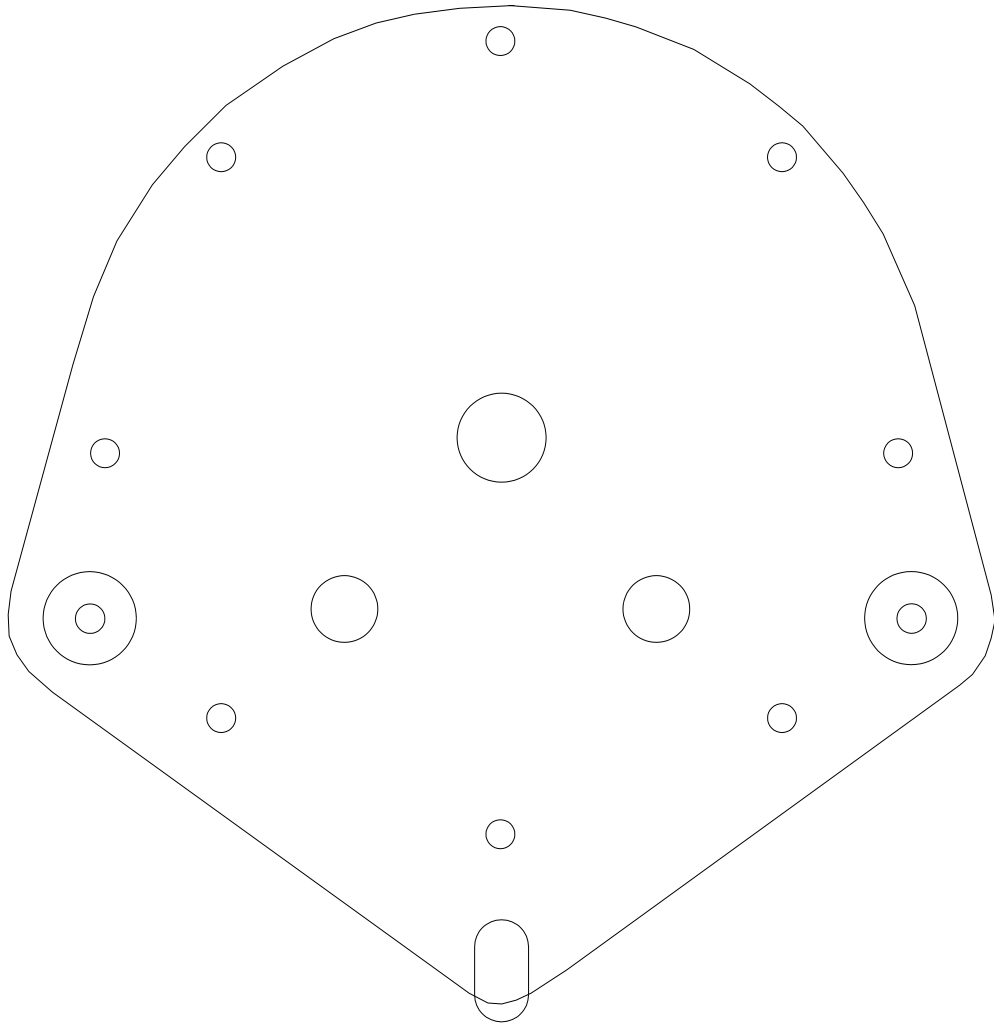
## D.2 Design av ROV skrog

Disse platene må skaleres slik at de får riktig størrelse. Ellers vil ikke dimensjonene passe til komponentene som brukes. En 10 cm strek nederst på alle tegningene kan være en god hjelp for skaleringen

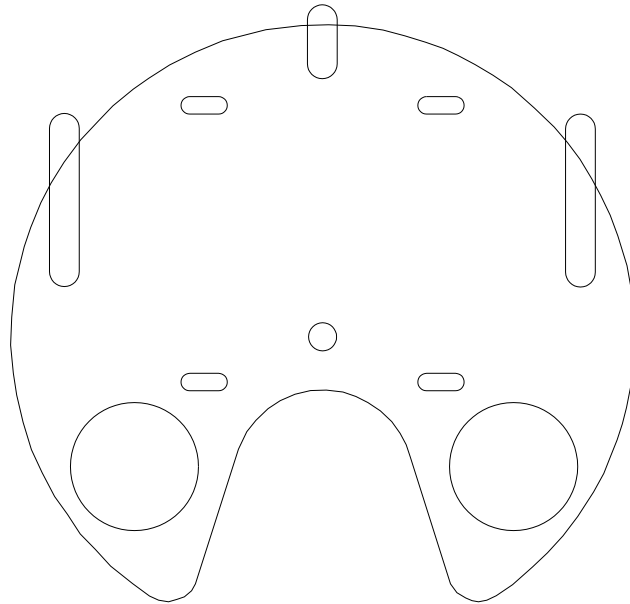
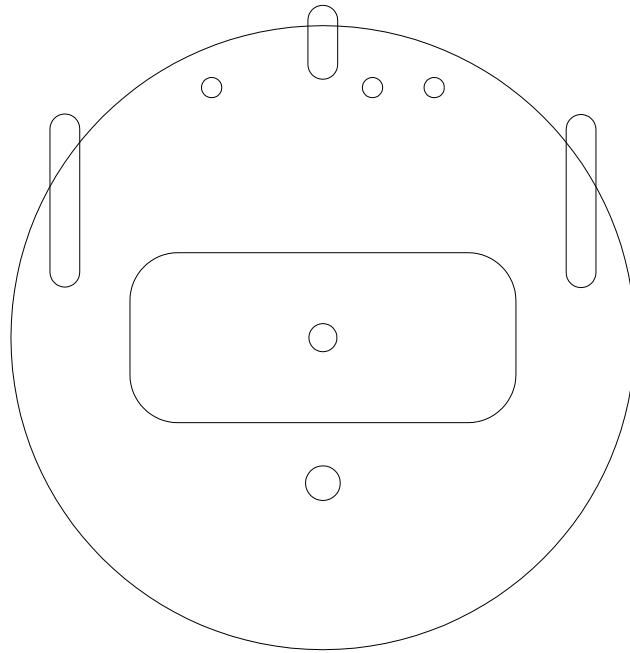




10 cm

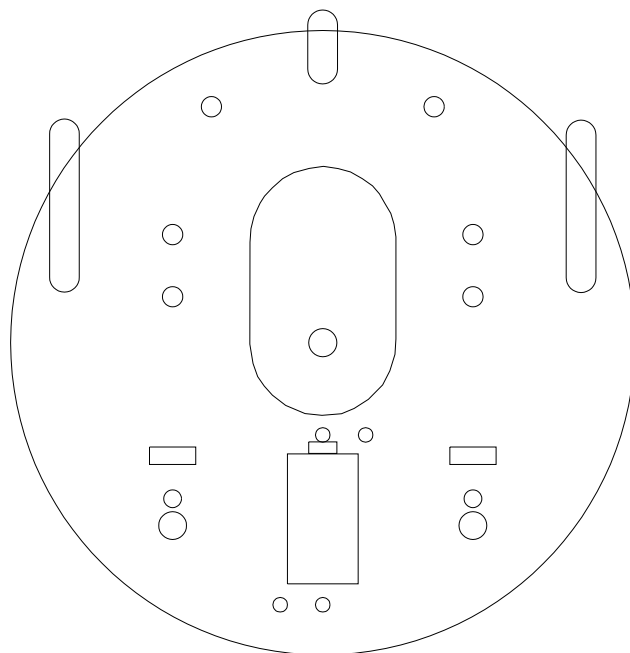


10 cm



10 cm





10 cm



## Vedlegg E Komponent- og utstyrslister

### E.1 Utstyrsliste til datainnsamlingsenhet

Det kan se ut til at RH16 50kOhm er tatt ut av sortementet til ELFA, i så fall kan B57550G1103F005 – 10kOhm brukes. Om denne brukes må motstanden byttes ut med en på 10 kOhm.

- 1 stk Arduino UNO ELFA nr. 110-38-919 (185,00 kr)
- 1 stk. shield-kort (Skolelaboratoriet ved NTNU) (15,00 kr)
- 1 stk. SD-kort leser OpenLog - Sparkfun USA (\$14,95)  
<https://www.sparkfun.com/products/13712>
- 1 stk. trykksensor SPD030G, ELFA nr. 173-97-677 (69,10 kr)
- 1 stk. universal transduser interface UTI Smartec ELFA nr. 173-08-836 (54,90 kr)
- 1 stk. sokkel 16 DIL ELFA nr. 148-14-801 (7,39 kr.)
- 1 stk. sokkel 6 DIL ELFA nr. 148-13-713 (12,50 kr.)
- 2 stk. stiftlist ELFA nr. 143-83-947 (11,20 kr.)
- 1 stk. sokkellist ELFA nr. 143-83-998
- 1 stk. bryter ELFA nr. 135-36-018 (11,90 kr.)
- 1 stk. NTC-motstand B57550G1103F005, EPCOS – 10 kOhm ELFA nr. 160-29-067
- 1 stk. motstand 10 kOhm ELFA nr. 160-55-270
- 2 stk. kondensator, 100 nF ( $C_1$ ,  $C_3$ ) ELFA nr. 165-65-659
- 2 stk. elektrolyttkondensator ( $C_2$ ),  $C_4$  ELFA nr. 167-00-249
- 1 stk. batterikontakt 9V ELFA nr. 142-00-150 (11,60 kr)
- 2 stk. koblingsplinter ELFA nr. ()
- 2 stk. jumpere ELFA nr.: 124-50-887 (1,13 kr.)
- Diverse ledning 0,6 mm

### E.2 Utstyrsliste for målejigg:

- 3 stk. gjengstang 3 stk. Ø 5 mm a 130 mm (Biltema nr.):
- 3 stk. vingemuttere, M5
- 6. stk. muttere, M5
- 9 stk skiver, M5
- 2 stk. kabelforskruinger, Biltema 4 – 8 mm, M16 (Biltema nr.: 350101)
- 1 stk. avløpsrør skjøt (glidemuffe) Ø 110 mm (Biltema nr.: 87268)



- 1 stk sprøyte 50 ml for tilkobling av slange (apotek?)
- 1 stk USB-kabel 2.0, A – B, 1,8 m eller lengre (Biltema nr.: 84881)
- 1 ark med mosegummi 1 mm, Panduro
- 1 stk plastslange 135 mm, Ø 3 innvendig (Clas Ohlson: 49-406-10)
- 2 stk runde plastglassplater, skåret av 4 mm (Biltema nr.: 26-208) se også vedlegg D, side 91
- 1 stk ballong (REMA 1000)

### **E.3 Utstysrliste til ROV skrog**

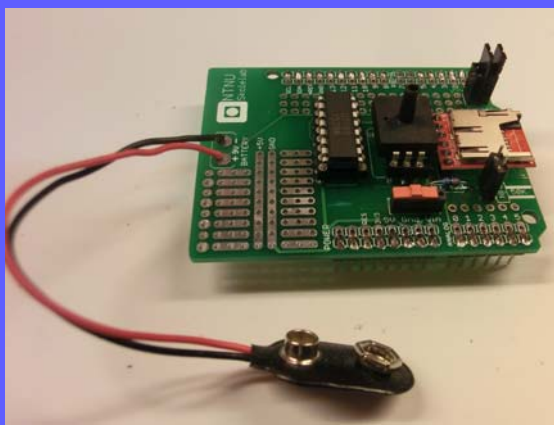
- 2 stk. glidemuffe Ø 11 cm x 11 cm (Biltema)
- 1 stk. skjøtemuffe 6 mm (Biltema)
- 1 stk. skjøtemuffe 12 mm (Biltema)
- 2 stk skjøtemuffe 10 mm (Biltema)
- 8 stk. gjengestang 150 x 5 mm (Biltema)
- 3 stk. gjengestang 210 x 5 mm (Biltema)
- 1 stk. gjengestang 175 x 6 mm (Biltema)
- 1 stk. gjengestang 275 x 10 mm (Biltema)
- 40 stk. muttere M5 (Biltema)
- 48 stk. stoppskiver M5 (Biltema)
- 8 stk. muttere M10 (Biltema)
- 8 stk. stoppskiver M10 (Biltema)
- 8 stk. vingemuttere 5 mm (Biltema/Clas Ohlson)
- 2 stk. 3 x 10 mm maskin-skrue m/mutter M3 (Biltema)
- 2 stk. 3,2 x 2 mm avstandstykket nylon (ELFA)
- 2 stk. PLA plastforing hjemmelaget
- 4 stk. 3 x 15 mm maskin-skrue m/mutter M3 (Clas Ohlson)
- 1 stk. kabelforskruing, 6-12 mm, 20 mm (Biltema)
- 1 stk. kabelforskruing, 4-8 mm M 16 mm (Biltema)
- 2 stk. kabelforskruing 1,5-4 mm, 12 mm (ELFA)
- 1 stk. batteripakke 6 x AA (ELFA)
- 2 stk. propeller (China)
- 2 stk. festeanordning propeller (China)
- 2 stk sveisetråd 2 mm (Vitensenteret i Trondheim)
- 2 stk. PLA skjøtemuffe, 3D-printet
- 2 stk. PLA endestopper for propell, 3D-printet
- 1 stk. fremre ytre seksjons-vegg (akryl), Laserkuttet ved Vitensenteret



- 
- 1 stk. midtre ytre seksjons-vegg (akryl) Laserkuttet ved Vitensenteret
  - 1 stk. bakre ytre seksjons-vegg (akryl) Laserkuttet ved Vitensenteret
  - 1 stk. fremre indre seksjons-vegg (MDF 3 mm) Laserkuttet ved Vitensenteret
  - 1 stk. midtre indre seksjons-vegg (MDF 3 mm) Laserkuttet ved Vitensenteret
  - 1 stk. bakre indre seksjons-vegg (MDF 3 mm) Laserkuttet ved Vitensenteret
  - 1 stk. Arduino UNO (ELFA)
  - 1 stk. shield-kort for Arduino UNO med trykk- og temp. sensor (Skolelaboratoriet)
  - 2 stk PLA-holder for kontroller kort, 3D-printet
  - 2 stk. rød ledning 30 cm (ELFA)
  - 2 stk. sort ledning 30 cm (ELFA)
  - 1 stk rød ledning en-tråd for feste (ELFA)
  - 1 stk. krympestrømpe tynn (ELFA)
  - 1 stk krympestrømpe tykk (ELFA)







Det undervisningsopplegget som er beskrevet i dette heftet er blitt til i et samarbeid mellom ToF koordinator Hilde Ervik (Skolelaboratoriet) og Håvard Holm ved Marintek i forbindelse med utviklingen av en enkel ROV for bruk i undervisningen av faget Teknologi og Forskningslære (ToF) i videregående skole.

Temaet “enkel ROV” ble først introdusert høsten 2015 som en del av en av modulene i serien av ToF-videreutdanningskurs ved NTNU. Høsten 2016 holdes kurset for andre gang og inkluderer en målestasjon for måling av trykk og temperatur. Høsten 2017 ble det også bygget et skrog for uttesting. Foreløpig preges dette undervisningsopplegget av at prototypingen ennå ikke er avsluttet, kurset blir derfor en del av utviklingen av en ferdig ROV og sensormodul. For et fag som ToF trenger ikke et slikt utgangspunkt være så dumt, siden også faget i seg selv skal ha et preg av utforskning. Vi ønsker derfor å inkludere lærerne i prosessen fram mot et ferdig produkt.

### ***Nils Kr. Rossing***

Dosent ved Skolelaboratoriet

E-post: [nils.rossing@ntnu.no](mailto:nils.rossing@ntnu.no)



Trondheim

## Program for lærerutdanning

**Skolelaboriet**  
for matematikk, naturfag  
og teknologi

Tlf. 73 55 11 43

Faks 73 55 11 40

<http://www.skolelab.ntnu.no>