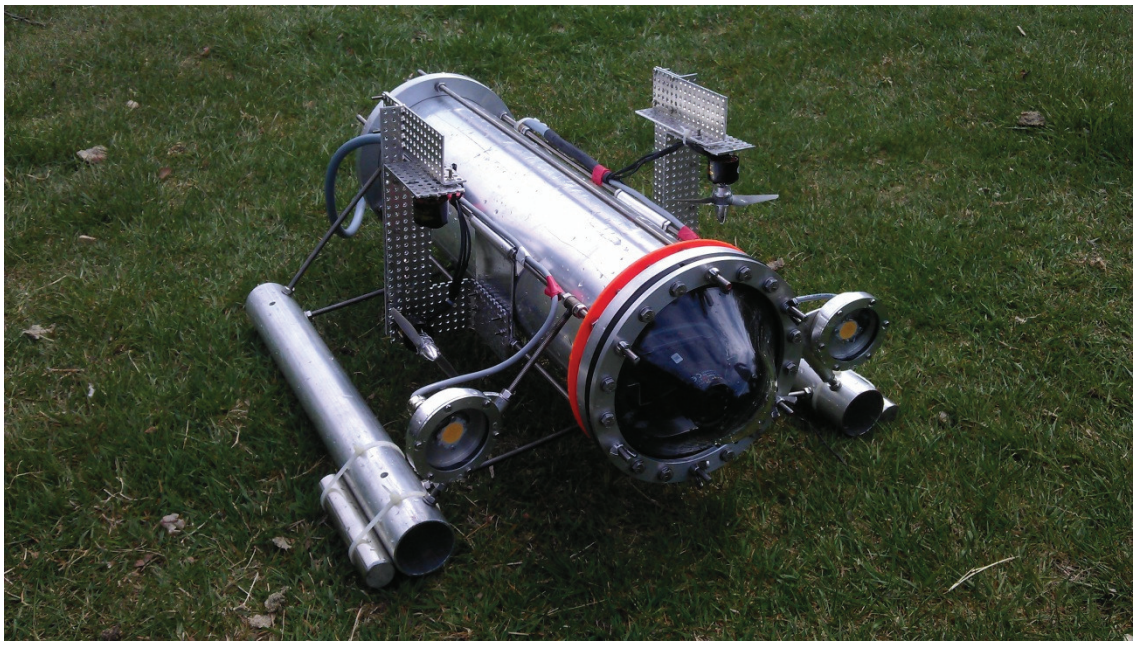
CHALMERS



ROV – Remotely Operated underwater Vehicle

Fjärrstyrd undervattensfarkost

Anders Markoff
Jonathan Flod
Oscar Wennerström
Pontus Carlsson
Viktor Bennersten

Institutionen för tillämpad mekanik
Avdelningen för fordonsteknik och autonoma system
CHALMERS TEKNISKA HÖGSKOLA
Göteborg, Sverige 2012

Kandidatarbete 2012:06

ROV – Remotely Operated underwater Vehicle

Fjärrstyrd undervattensfarkost

Institutionen för tillämpad mekanik
Avdelningen för fordonsteknik och autonoma system
CHALMERS TEKNISKA HÖGSKOLA
Göteborg, Sverige 2012

ROV – Remotely Operated underwater Vehicle
Fjärrstyrd undervattensfarkost

Kandidatarbete

© 2012
Anders Markoff
Jonathan Flod
Oscar Wennerström
Pontus Carlsson
Viktor Bennersten

Handledare: Göran Stigler

Examinator: Krister Wolff

Institutionen för tillämpad mekanik
Avdelningen för fordonsteknik och autonoma system
Chalmers Tekniska Högskola
SE-412 96 Göteborg
Sverige
Telefon: +46 (0)31-772 1000

Omslag: Foto av den konstruerade ROV:en.

Förord

Detta kandidatarbete har utförts under våren 2012 vid Chalmers Tekniska Högskola. Vi vill tacka vår handledare Göran Stigler som hela tiden fått oss att tänka kritiskt och vår examinator Krister Wolff. Vi vill även tacka Jan Bragee och Reine Nohlborg från prototyplabbet för all information och hjälp vi fått vid tillverkning av farkosten. Ett tack går också till AXIS Communications som lånade ut en kamera till projektet. Slutligen vill vi även tacka Jonas Jarvoll som genom ett tidigare projekt inspirerat oss i vår process.

Abstract

This thesis describes the development and construction of a ROV-system, which includes an underwater vehicle, a surface unit and a communication link that binds them together. The system was divided into sub problems which, as far as possible, will be resolved with existing technology. These solutions had to be assembled, waterproofed and reliability tested. Various solutions to the sub problems were examined, evaluated and ranked relatively to each other. The best solution for each sub problem was then chosen and comparisons between all solutions were made to make sure they were compatible with each other. Thus were the plans for the final prototype made.

Thereafter the project moved into a construction phase. The main hull was an aluminum tube with two end pieces. A translucent dome was placed in the front piece and cable glands were mounted in the rear piece. Between the front and the rear, six threaded rods with nuts were placed in order to tighten the structure and hold it together. Motors, lamp housings and skids were also mounted on the threaded rods with nuts. A very light-sensitive IP camera from Axis was placed in the dome. In addition, the vehicle light up its surrounding with two bright LED's which was mounted in self-made, waterproof lamp housings. The vehicle is controlled from the surface using an Xbox 360 controller, while the information and video from the vehicle is displayed on a computer.

The project was completed with very good results. A well functioning prototype was constructed which met all requirements from the specifications. The vehicle is well balanced and runs smoothly in the water, though it is sometimes difficult to navigate in waters with limited visibility.

Sammanfattning

Denna rapport beskriver framtagandet och konstruktionen av ett ROV-system, som innefattar en undervattensfarkost, en ytenhet och en kommunikationslänk som binder dem samman. Systemet delades in i delproblem som i så stor utsträckning som möjligt skulle lösas med färdiga produkter som endast behövde monteras, vattentätas samt driftsäkras. Olika lösningar på delproblemen undersöktes, utvärderades och viktades. De lösningar som viktats högst och fungerade med övriga lösningar inom systemet valdes till den slutgiltiga totallösningen.

Därefter övergick projektet i en byggfas där ett skrov i form av ett aluminiumrör med två ändstycken utgjorde huvudkroppen. I framstycket placerades en genomskinlig kupol, i bakstycket monterades kabelgenomföringar. Mellan dessa ändstycken placerades sex gängstänger som håller samman konstruktionen. På dessa gängstänger monterades även motorer, lamphus och medar. Innanför kupolen placerades en mycket ljuskänslig IP-kamera från Axis. Dessutom lyses dess omgivning upp av två starka LED:s som monterades i egentillverkade vattentäta lamphus. farkosten styrs från ytan med hjälp av en Xbox 360-kontroll samtidigt som information och video från densamma visas på en dator.

Projektet är avklarat med mycket gott resultat i form av en färdig prototyp där alla krav i kravspecifikationen uppfylldes. ROV:en är väl avvägd och smidig att köra i vatten men det kan vara svårt att navigera i sjöar eller hav där sikten är begränsad.

Innehållsförteckning

1 Inledning 1
1.1 Bakgrund 1
1.2 Syfte 2
1.3 Mål 2
1.4 Avgränsningar 2
2 Genomförande 3
2.1 Projektmetodik 3
2.2 Kravspecifikation 4
2.3 Uppgiftsidentifiering 6
3 Lösningar 11
3.1 Farkost 11
3.2 Kommunikationslänk 29
3.3 Ytenhet 31
4 Resultat 35
4.1 Tillverkning 35
4.2 Uppfyllande av kravspecifikation 45
5 Diskussion 47
5.1 Diskussion av resultat 47
5.2 Förbättringsförslag 49
6 Slutsats 50
7 Referenser 51
Figurförteckning 52
Appendix A - Tidsplanering 54
Appendix B – Foton från tillverkning 55
Appendix C - Energiförsörjning 66
Appendix D - Batterialternativ 67
Appendix E – Ritningar 68
Appendix F – Kopplingsschema 72
Appendix G – Arduinokod 73
Appendix H – Datorkod 79

1 Inledning

Kapitlet ger en bakgrund till inom vilka ramar projektet genomförs och vilka avgränsningar som finns.

1.1 Bakgrund

Att kunna utforska saker under vattenytan har länge varit intressant för människan men tyvärr är våra kroppar inte anpassade att vara under ytan en längre tid. Detta har dock visat sig ha en lösning på flera olika sätt. De två vanligaste sätten är att antingen ha med sig tryckluftstankar med komprimerad luft för att andas under ytan, eller att åka ner i vattnet inuti en farkost. Ett tredje sätt är att på något sätt föra ner en kamera under vattnet vilket ger människan möjlighet att betrakta undervattensmiljön utan att själv befinna sig där. Detta kan bland annat ske genom att använda en fjärrstyrd undervattensfarkost.

Fjärrstyrda undervattensrobotar utvecklades på 1950-talet och kallades från första början för "CURV" vilket stod för "Cable controlled Underwater Recovery Vehicle". Numer går de under namnet "Remotely Operated underwater Vehicles". De var i huvudsak finansierade av amerikanska marinen vilket gjorde att de främst användes av militären för att kontrollera skador på fartyg och hämta borttappade övningstorpeder. Ett bra exempel är år 1966 då ett B-52 flygplan kraschade i havet utanför Palomares vilket ledde till att en atombomb tappades bort. Den återfanns sedan med hjälp av en ROV(Wikipedia, 2012). Senare har ROV:ar använts inom många användningsområden och förenklat livet för människan. Ett av de absolut största områdena är olje- och gasindustrin till havs där djupen är för stora för att en människa skall kunna gå ner. Där utför ROV:ar allehanda arbetsuppgifter som svetsning, lägga pipelines, underhåll och dylikt.

1.2 Syfte

Projektets syfte är att utveckla en prototyp av en ROV tillsammans med dokumentation av utvecklingen i form av en teknisk rapport. Prototypen skall utvecklas genom att metodiskt sätta samman existerande komponenter till en fungerande enhet.

1.3 Mål

Projektets mål är att producera en fjärrmanövrerad och tät farkost som skall klara tio meters djup. Farkosten skall kunna framföras under vatten och skötas från ytan av en person som har tillgång till realtidsvideo, informationsamt kontroller för att styra den. Systemet skall vara möjligt att lyfta och ta med sig.

1.4 Avgränsningar

Projektet skall i slutändan leda till en färdig prototyp vilken består av många delsystem som var för sig skulle kunna göras komplexa. Inom tidsramen för projektet är det ej möjligt att utveckla varje del i prototypen från grunden. Istället ligger fokus på att i största möjliga utsträckning hitta existerande lösningar och integrera dessa till ett komplext men driftsäkert system.

För att göra farkosten robust men även för att hålla nere komplexiteten hos projektet kommer inga försök att skicka information trådlöst till och från farkosten att göras. För att prioritera funktionen hos farkosten kommer även utseendet vara av sekundär betydelse. Vad gäller framdriften har denna avgränsats till elmotorer, inga försök med någon annan lösning kommer utvärderas. Vertikal rörelse i vattnet kommer styras med motorer och alltså kommer inte användandet av ballasttankar utvärderas.

Även ekonomin är en begränsning. Budgeten för projektet är 10 000 kr för inköp av delar och material. Om en dyrare produkt skulle skänka stort mervärde till projektet kan detta diskuteras med institutionen, men dyra komponenter måste kunna motiveras.

Farkosten skall inte ha något externt verktyg som kan utföra fysiska uppgifter under ytan.

2 Genomförande

Projektet delas in i delproblem, för att lättare kunna överblicka vad som behövs göras. För att projektets genomförande skall fortlöpa så effektivt som möjligt kommer projektdeltagarna att fokusera på olika delproblem och arbeta parallellt med varandra. Större beslut fattas dock givetvis gemensamt. För att inte begå samma misstag, som andra ROV projekt stött på, skall liknande projekt studeras.

2.1 Projektmetodik

Systemet kan delas in i mindre deluppgifter som behöver lösas för att skapa en helhetslösning. Deluppgifterna tilldelas två till tre personer, för att ingen skall behöva jobba ensam och alla alltid skall ha någon att diskutera med. På varje deluppgift skall sedan minst två lösningar tas fram och jämföras. Lösningarna skall utvärderas mot kravspecifikationen och rangordnas efter vilken som löser uppgiften bäst. Hela lösningsgången skall dokumenteras i en loggbok och i en individuell tidsrapport. När samtliga deluppgifter är lösta kombineras möjliga lösningar ihop och rangordnas igen med hjälp av kravspecifikationen.

Uppgiften att lösa är främst av praktisk karaktär vilket innebär mindre utrymme för litteraturstudier. Till många av problemen finns en tidigare lösning som ej är applicerad under vatten vilket i stor utsträckning medför att praktiska prover och utvärdering av resultaten.

Nedan listas sätt att samla in information på.

- Boka möte med Jonas Jarvoll på Hobbytronik.se. Jonas Jarvoll har konstruerat en ROV, dokumenterat detta på www.hobbytronik.se och anses ha god kunskap angående tillverkningen och lösningarna.
- Läsa *The ROV manual* som kan lånas på Chalmers biblioteket. Dock endast de delar som är intressanta för oss.
- Websökningar.
- Kontakta expert inom området.

2.2 Kravspecifikation

Utifrån projektets uppgift och mål har en kravspecifikation sammanställts enligt Tabell 2.1 och Tabell 2.2. Där det står att kravet specificerades i uppgiften innebär det att detta krav ställdes redan i beskrivningen av projektuppgiften medan övriga krav tagits fram under projektplaneringen.

Tabell 2.1– Kravspecifikation, krav

Krav	Kriterier	Enhet	Antal	Förklaring
A	Minsta djup i saltvatten	m	10	Specificerades i uppgiften.
B	Visa realtidsvideo på ytan	St	1	Specificerades i uppgiften.
C	Maximal vikt	Kg	20	ROV:en skall kunna bäras utan alltför stora bekymmer.
D	Minsta aktionsradie	m	20	Ett krav är 10 m djup, för att kunna röra sig lite i sidled med anses det rimligt att dubbla denna siffra.
E	Lägsta operationstid vid maxfart framåt på 10m djup	min	10	10 minuter är en uppskattad tid vad det tar att dyka ner och titta på en sak och tillbaka till ytan.
F	Minsta maxfart under vatten	m/s	0,5	Ett antagande gjort utifrån att ROV:en skall kunna köras även i strömt vatten.
G	Farkosten skall vara tät	-	-	Elektroniken tål ej saltvatten.
H	Farkosten skall kunna manövreras av en person	-	-	Det skall inte behövas fler personer för att styra ROV:en.
I	Information från farkosten skall visas i styrenheten	-	-	Specificerades i uppgiften.
J	Dragsäker länk till farkosten	-	-	Ifall motorerna slutar fungera skall ROV:en kunna dras upp till ytan i kabeln.

Tabell 2.2 - Kravspecifikation, önskemål

Önskemål	Rang*	Kriterier	Enhet	Antal	Förklaring
A	5	Minsta dykdjup	m	20	Det är fördelaktigt att kunna dyka djupt, under 20 m finns dock inte lika mycket att se.
B	7	Maximal vikt	kg	10	Enkel att bära.
C	4	Fungera upprepade gånger i saltvatten	-	-	ROV:en önskas användas i saltvatten.
d	1	Hållbar konstruktion	-	-	En stabil och robust konstruktion är vital för att ha en fungerande farkost.
e	11	Minsta aktionsradie	m	100	För att operera på längre avstånd med mindre begränsningar men samtidigt inte få för mycket kabel eller problem med dataöverföring anses 100 m vara ett bra önskemål.
f	8	Minsta maxfart	m/s	1	0,5 m/s är en uppskattning av hur strömt det kan vara i en hamn och farkosten bör kunna köra uppströms
g	10	Tid att frilägga elektroniken på farkosten för en van användare	min	30	Elektroniken skall kunna friläggas och felsökas utan större svårigheter.
h	2	Farkosten bör ha belysning	Lumen	>400	För att kameran skall kunna se bra krävs belysning, dyklampor på 400 finns och fungerar väl. (Kinetics, 2012)
i	12	Farkosten bör vara genomskinlig			Dels för att lätt upptäcka läckage och för att kunna demonstrera innehållet.
j	3	Kostnad för farkosten bör ej överstiga	Kr	10000	Budget från institutionen, får endast överstigas med mycket goda argument.
k	6	Tillverkad av standardkomponenter	-	-	För att enkelt kunna bytas och vara lättförståliga.

*Rangordningen kommer från viktningsmatrisen i Tabell 2.3.

Tabell 2.3 - Viktning av önskemål. Kolumnen till vänster vägs mot de andra önskemålen och en etta betyder att önskemålet är viktigare, en nolla att det är mindre viktigt. Näst sista kolumnen är summan och sista är rangordningen.

	a	b	c	d	e	f	g	h	i	j	k	Σ	Rangordning
a - 20m djup		1	0	0	1	1	1	0	1	0	1	6	5
b - vikt 10 kg	0		0	0	1	1	1	0	1	0	0	4	7
c - saltvatten	1	1		0	1	1	1	0	1	0	1	7	4
d - hållbar	1	1	1		1	1	1	1	1	1	1	10	1
e - 100m aktionsradie	0	0	0	0		0	0	0	1	0	0	1	10
f - 1 m/s	0	0	0	0	1		1	0	1	0	0	3	8
g - öppna och stänga	0	0	0	0	1	0		0	1	0	0	2	9
h - belysning	1	1	1	0	1	1	1		1	1	1	9	2
i - genomskinlig	0	0	0	0	0	0	0	0		0	0	0	11
j - kostnad 10k	1	1	1	0	1	1	1	0	1		1	8	3
k - enkel/standardiserad	0	1	0	0	1	1	1	0	1	0		5	6

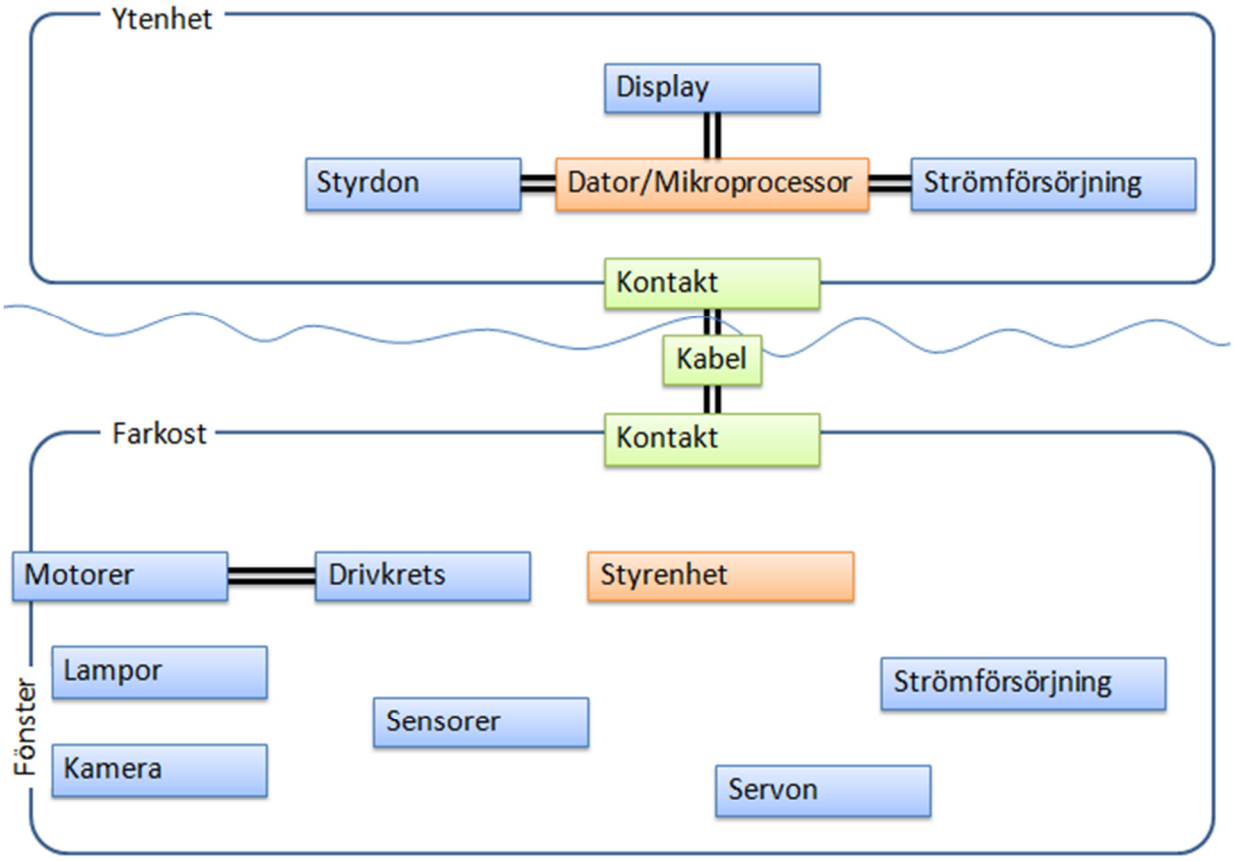
2.3 Uppgiftsidentifiering

För att göra uppgiften överskådlig och skapa en bild över vilka problem som måste lösas har den ursprungliga uppgiften utvecklats och delats upp i en mängd mindre delar. Efter att ha studerat flera liknande projekt har uppgiften delats in i tre fysiska delar; farkost, kommunikationslänk och ytenhet.

Ytenheten behöver ha ett användargränssnitt som visar video och information för användaren via en display samt ett styrdon som genererar insignaler till farkosten. Det finns inget krav på att ytenheten skall vara vattentät, men det vara av fördel om den tål vattenstänk då den förmodligen kommer befinna sig i närheten av vatten.

Länken skall binda samman farkost och ytenhet. Video, styrkommandon och eventuellt andra signaler skall överföras. Länken skall vara dragfast så att farkosten kan dras in med hjälp av länken och den skall ha trycksäkra kopplingar mot farkosten.

Alla delar som har identifierats har kategoriserats under de olika fysiska enheterna.



Figur 2.1 - Schematisk bild av systemet

2.3.1 Farkost

Delar av farkosten behöver kapslas in i ett eller flera höljen. Vad som då är viktigt är antal vattentäta enheter, avvägning, tätning, kabelgenomföringar, hållfasthet, kondensproblematik.

- Farkosten kan vara en stor enhet med alla komponenter inuti eller flera små enheter som alla är vattentäta.
- Farkosten behöver ligga stabilt i vattnet och bör därför konstrueras så dess tyngdpunkt ligger lågt.
- Tätningen är väldigt viktig ty det kommer med stor sannolikhet att finnas elektronik i farkosten. Utifrån att ha studerat liknande projekt framgår att den mest kritiska tätningen med största sannolikhet kommer vara kabelgenomföringarna.
- Farkosten bör vara hållfast då ett önskemål är att i framtiden klara större djup än tio meter. Robustheten hos farkosten är viktig då konstruktionen måste vara tålig mot yttre påverkan.
- Ett problem som uppstår vid temperaturskillnader i täta enheter under vatten är kondens. Denna farkost kommer antagligen inte vara förskonad och därför behövs något fuktabsorberande material som t.ex. kristallzeoliter, papper eller ris.

Farkosten behöver en energikälla för att driva de motorer som förflyttar farkosten framåt. De olika typerna av energikälla kan delas in i tre olika kategorier: Ytdrivna, farkostdrivna eller hybridsystem(Christ, 2007).

- Ytdrivna farkoster måste erhålla sin energi från ytan och blir därför alltid uppbundna med en koppling upp till ytan. Farkosten har i detta fall ingen energiackumulator.
- Farkostdrivna farkoster lagrar all energi i ett batteri, bränslecell eller liknande. Denna form kan således vara trådlös.
- Ett hybridsystem kan innehålla både en koppling till ytan där det tar emot energi såväl som en ackumulator ombord på farkosten.

För att ta sig fram måste farkosten ha motorer med tillhörande propellrar och motorstyrning.

- För att välja motor bör det beaktas att trögheten i vatten är mycket större än luft vilket ställer högre krav på motorns utvecklade moment.
- Till motorn behövs väl anpassade propellrar som kan flytta rätt mängd vatten utan att röra upp alltför mycket sediment om farkosten befinner sig på botten.
- För att kunna köra motorerna på olika varvtal, och således flytta farkosten långsammare eller snabbare i olika situationer, måste motorstyrningar kopplas till motorerna.

Projektuppgiften innefattar en möjlighet att se realtidsvideo från farkosten. För att tillgodose detta behöver farkosten en kamera, möjlighet till belysning och bildöverföring till ytan.

- Vid val av kamera bör det beaktas att det med stor sannolikhet kommer vara mörkt även med egen belysning, samt vilken typ av signal som genereras av kameran.
- Den egna belysningen bör vara kraftfull för att ge en så god sikt som möjligt. Detta är med stor sannolikhet en energikrävande del av farkosten.
- Videoöverföringen bygger till stor del på vilken typ av utsignal kameran genererar. Överföringen kan ske via ett flertal olika gränssnitt.

Farkosten, som kommer vara obemannad, behöver någon form av styrenhet som tolkar insignaler och genererar utsignaler för att styra farkosten. Enligt National Institute of Standards and Technology kan obemannade farkoster manövreras på olika sätt; fullt autonom, halvautonom, telemanövrerad och fjärrmanövrerad(Christ, 2007).

- Fullt autonoma farkoster utför uppdrag inom givna ramar utan mänskligt ingripande.
- Halvautonoma farkoster utför givna uppgifter med viss mänsklig integration.
- Telemanövrerade farkoster har en mänsklig operatör som via video och andra sensorer får beslutsunderlag när denne styr farkosten eller ger den nya uppdrag att utföra. Farkosten har begränsad initiativförmåga.
- Fjärrmanövrerade farkoster tar bara kommande av en mänsklig operatör och kan själv inte ta initiativ.

2.3.2 Kommunikationslänk

Länken mellan farkosten och ytenheten, även kallad navelsträngen, kan spela en mer eller mindre viktig roll för systemet i sin helhet. Aspekter att beakta är huruvida den skall leverera energi till farkosten, vilken och hur mycket information som skall skickas, styrsignaler, kopplingar och kabelhantering.

- Energiöverföring via kommunikationslänken kan potentiellt användas för att dels ladda ackumulatorer i farkosten såväl som enbart spänningssätta styrenheten. Att enbart spänningssätta styrenheten kan vara behjälpligt vid felsökning och göra systemet till viss del redundant.
- I enlighet med avgränsningarna kommer videoöverföring ske genom en trådbunden kommunikationslänk . Kabelvalet påverkas av kvaliteten på videon som skickas.
- Styrsignaler som skickas till styrenheten i ROV:en kräver ytterligare elektriska ledare mellan ytstationen och farkosten via kommunikationslänken.
- Vilken kabeltyp som används ger ett direkt utslag på vilken typ av koppling som behövs gentemot farkosten och ytenheten.
- Kabelns styvhet avgör hur lätthanterlig den är. En styvare kabel kan dessutom påverka ROV:ens framdrift.

2.3.3 Ytenhet

Ytenheten kan utformas på ett flertal olika sätt för att täcka de grundläggande krav som ställs på den i systemet. Ytenheten kommer att behöva ha ett styrdon, en display, strömförsörjning, styrenheten.

- Ytenheten behöver ett styrdon för att generera styrsignaler till styrenheten och i slutändan farkosten.
- Då farkosten skall generera realtidsvideo behövs en display för att visa videon. Displayen skulle även kunna visa relevant information från farkosten som temperatur, luftfuktighet och batteristatus.
- Ytenheten kommer behöva strömförsörjning, och då den i slutändan skall användas vid vattnet är det sannolikt att den behöver en ackumulator.
- Ytenheten behöver omvandla insignaler och skicka dem till styrenheten i farkosten. Detta görs lämpligen med någon form av styrenhet.

3 Lösningar

För att undvika onödiga misstag och oväntade problem inleds arbetet med att ta fram alla relevanta lösningar. Alla lösningar vägs ur ett flertal olika perspektiv och betygssätts därefter. De lösningar som är kompatibla och har högst betyg tillämpas.

3.1 Farkost

Farkosten är den del av systemet som kommer befinna sig under vattnet och som i förväg betraktas som den svåraste delen att realisera. Detta beror till stor del på att den måste vara vattentät. Farkosten delas in i farkostkropp, strömförsörjning, motorer, belysning, kamera och styrenhet.

För att göra uppgiften överskådlig och skapa en bild över vilka problem som måste lösas har den ursprungliga uppgiften utvecklats och delats upp i en mängd mindre problem. Efter att ha studerat flera liknande projekt har uppgiften delats in i tre fysiska delar; farkost, kommunikationslänk och ytenhet.

3.1.1 Farkostkropp

Designen av farkostkroppen är en mycket viktig del. Den skall bestå av ett skrov som kan hålla ROV:en neutral eller svagt positiv i vatten, inrymma plats för motorer och lampor. Var kameran, eventuellt batteri och övrig elektronik skall placeras måste utredas. Utöver det måste kabelgenomföringarnas placering bestämmas och göras täta. Trycket på tio meters djup uppgår till en bar, vilket alla täta delar måste tåla med minst en faktor två. För att kunna laga och korrigera utrustningen i ROV:en måste farkosten på ett smidigt sätt kunna öppnas.

Design av vattentätt skrov

Skrovets främsta uppgift är att hålla ROV:en flytande i vattnet, så nära ett neutralt läge som möjligt. Av säkerhetsskäl anses det vara bättre att ROV:en är lite lättare än vatten eftersom den då skulle flyta upp om något går fel. Skrovet skall också innehålla kamera och elektronik som skall hållas torra, den måste alltså vara vattentät men fortfarande gå att öppna om något slutar fungera och skulle behöva korrigeras. I fören skall materialet vara genomskinligt för kamerans skull.

Flera förslag till form av skrov har utvecklats och diskuterats. Den mest uppenbara formen är att ha en lång cylinder med platta ändar, eftersom andra projekt som studerats ofta använt sig av sådana. Förutom detta har en lådformad och sfärisk kropp diskuterats, men även antalet vattentäta kroppar. Huruvida ändarna skall vara platta eller konvexa måste också utredas.

Att ha en två- eller flerdelad kropp innebär att det blir enklare att placera en motor mellan kropparna för upp- eller nedkörning som sitter väl skyddad och strategiskt placerad. Den hamnar varken över ytan, eller för nära botten så att slam och bottensediment rörs upp. Nackdelarna överväger dock denna fördel då det blir fler delar att täta vilket ofta visat sig vara ett stort problem när det kommer till amatörbyggda ROV:ar. Fler delar gör konstruktionen dyrare och det krävs en mer komplex ytterkonstruktion för att hålla ihop delarna. Balanseringen blir svårare då exempelvis kameran sitter i en del och eventuell energikälla i den andra. För att slippa extra kabelgenomföringar kan ett vattentätt rör fästas som leder från ena kroppen till den andra, men detta skulle också bli en för avancerad lösning att konstruera.

En sfärisk kropp är svår att bygga och det skapar svårigheter att bygga ett ytterhölje att fästa lampor och motorer i. Då det är mycket lättare att göra kabelgenomföringarna på en plan yta förkastas detta förslag.

En lådformad kropp har fördelen att kabelgenomföringarna kan sitta var som helst på kroppen, det är lätt att strukturera placeringen av komponenter inuti och lätt att fästa saker både på insidan och på utsidan. Det skulle även gå att få i vilken storlek som helst eftersom det går att tillverka relativt enkelt av exempelvis plåt eller aluminium och är då inte låst till standardstorlekar. Nackdelen med en lådformad kropp är att trycket ger upphov till högre spänningar i materialet än för till exempel en cylinder.

En cylinderformad kropp kan alltså ha mindre godstjocklek och blir därmed lättare och billigare än ett rätblock. Det finns att köpa i olika material med stor variation av diametrar. Med minst en platt ände är placeringen av kabelgenomföringarna inget problem. Förslagsvis är det då i aktern som dessa placeras så att de inte stör kameran och framdriften

Att ha rundade ändar har fördelar ur trycksynpunkt och för en eventuell rörlig kamera. En plan skiva utsätts för mycket större spänningar och buktar mer, något som också kan leda till läckor i infästningen. Att dra kabelgenomföringarna genom en kupol är dock mer komplext än genom en plan skiva och det är mycket större risk för att det inte blir tätt. För en cylinderformad kropp sätts förslagsvis en kupol i fören och en plan skiva i aktern.

Tabell 3.1 - Jämförelse mellan olika former för vattentätt skrov

Egenskaper	Lådformad	Cylindrisk
Hållfasthet	2	4
Anslutningsmöjlighet	5	4
Summa	7	8

Utifrån diskussionen anses en cylinderformat skrov vara det bästa alternativet. Detta eftersom det står emot vattentrycket bättre utan att deformeras. Att använda en kupol i fören anses också fördelaktigt medan bakstycket bör vara slätt för kabelgenomföringarnas skull.

Om andra lösningar förhindrar användandet av en cylinderformat skrov kan en lådformad användas och skulle av någon anledning en kupol inte passa kan en slät skiva användas både i fören och i aktern.

Materialval till skrov

ROV:ens framända måste vara genomskinlig och tillverkas förslagsvis i plexiglas eller någon annan plast. Glas är inte att föredra eftersom det är svårt att jobba med. Materialval och godstjocklek är direkt beroende av varandra och detta är något som måste beaktas. De material som primärt diskuterats är plexiglas, polykarbonat, aluminium och stål. I fallet att en cylinderformad kropp och rundade ytor används kan materialet vara tunnare och vekare än vid användning av plana skivor. Ett önskemål är att skrovet skall vara genomskinligt för att innehållet skall kunna demonstreras och undersökas utan att ROV:en måste öppnas.

Om ett skrov av metall används måste korrosion tas i beaktande. Dels kommer metallerna erodera i saltvattnet och om de används tillsammans med ädlare metaller kan galvaniska strömmar uppstå. Ett sådant exempel är om aluminium används tillsammans med stål, dåskulle aluminiumet korrodera på grund av stålet och de galvaniska strömmarna. Detta kommer med största sannolikhet inte bli ett problem då ROV:en bara kommer att vara i vattnet korta stunder och detta är något som blir ett problem först efterlång tid. För säkerhets skull kan offeranoder användas för att skydda aluminiumet men detta kommer inte vara aktuellt förrän senare.

Fördelen med plexiglas och polykarbonat är att de är genomskinliga och relativt billiga. Aluminium är inte mycket tyngre än plasterna, men starkare och därmed fördelaktigt till exempelvis bakstycket där eventuella kabelgenomföringar skall gängas. Aluminium är dessutom korrosionståligt. Stål är starkt och relativt lätt att svetsa i, dock är det mycket tungt vilket är en stor nackdel för ROV:en som helst bör vara så lätt som möjligt. Materialens mekaniska egenskaper finns i Tabell 3.2. Kostnadsaspekten måste också tas hänsyn till.

Tabell 3.2 - Mekaniska egenskaper hos material

Material	Densitet	Sträckgräns	Utmattningsgräns	Källa
Plexiglas (PMMA)	1,18 g/cm3	48 - 76 MPa	11 - 12 MPa	(Matbase, PMMA, 2009)
Aluminium*	2,63 g/cm3	90 - 300 MPa	140 - 155 MPa	(Matbase, Aluminium 6060, 2009)
Stål*	8 g/cm3	580 - 760 MPa	240 MPa	(Matbase, Stainless steel 304, 2009)

*Tabellen utgår från de vanligaste materialtyperna, aluminium 6060 och rostfritt stål typ 304

Materialvalet är beroende av de andra lösningarna på farkosten och bör därför anpassas efter dessa. Den viktigaste egenskapen är dock robusthet och hållbarhet. Ett mycket viktigt värde för att jämföra material med avseende på detta är utmattningsgräns dividerat med densitet. Detta tal är nästan dubbelt så stor som för aluminium som för stål och nästan 60 gånger större för aluminium som för plexiglas. Plexiglas förvisso genomskinligt, vilket var ett önskemål, men eftersom detta är lågt prioriterat anses det vara sekundärt i jämförelse. Materialen rangordnas således enligt följande:

1. Aluminium
2. Stål

3. Plexiglas

Genomföringar och bakstycke

Det är mycket lättare att göra täta kabelgenomföringar på en slät yta än på en som buktar. Om kabelgenomföringarna har en liten diameter kan dock en buktad yta jämnas ut, antingen genom att slipa ner den eller genom att tillverka en upphöjning exempelvis att limma på en skiva vars ena sida är konvex och passar mot den buktande ytan.

För att kunna täta de delar som måste kunna öppnas används med fördel en o-ring. För att kunna applicera en sådan krävs ett fräst spår vilket måste göras på en plan yta. Alltså vore en flat akterspegel i tjockt gods där navelsträngen ansluts vara bästa plats att även samla genomföringarna till motorer och lampor.

Ett platt och tjockt bakstycke ger möjlighet att fräsa spår för bättre tätning mot skrovet och för kabelgenomförningar. Det ger även en platt yta att montera de komponenter inne i skrovet som behöver kylas, till exempel motorstyrkretsarna.

Yttre struktur

Utanpå skrovet skall lampor och motorer fästas. För att öka robustheten och minska risken för läckor bör inget skruvas direkt i skrovet. Att limma eller svetsa fast något minskar modulariserbarheten. Därav bör någon form av struktur sitta runt farkosten. Dess utformning påverkar hur lättlampor, motorer, motorskydd och eventuella fenor och plattor kan fästas och justeras. Strukturen agerar dessutom som en form av skyddsbur som skyddar skrovet mot stötar. Kablar och fästen till de yttre komponenterna bör dessutom kunna sitta snyggt och skyddat.

Strukturen bör inte klämma mot skrovet eftersom det ger upphov till spänningar i materialet som samverkar med vattentrycket. Den bör istället sitta löst runt kroppen och fästas i flänsarna fram och bak, alternativt sitta fast i en fläns och ligga an mot kroppen.

En sådan konstruktion bör vara så smidigt som möjligt för att inte hindra ROV:ens rörelse. Om den sitter fast i flänsarna bör därför de delar som går mellan framdelen och bakdelen sitta nära skrovet. Ett bra förslag är att sätta gängstänger från främre flänsen till den bakre. Dessa kan eventuellt samtidigt användas för att fästa bak- och framstycket. Om istället främre delen av strukturen ligger löst mot skrovet och bara fästs i bakstycket kan fasta stänger eller avlånga skivor användas för att hålla samman dessa.

För att kompensera för asymmetriska former som gör att ROV:en vrider sig vid framdrift eller för att förhindra för att ROV:en driver okontrollerat i sidled kan välriktade fenor byggas in i konstruktionen. Dessa är dock mycket svåra att justera rätt och kan medföra att ROV:en lättare dras med i strömmar och därför skall sådana inte användas.

Nederst bör någon form av ben eller medar placeras. Detta eftersom ROV:en bör kunna stå stabilt men framför allt eftersom den med stor sannolikhet ofta kommer stöta i botten. Genom

att studera liknande projekt och videor från andra ROV-körningar framgår att operatören sällan hinner upptäcka botten och sakta in innan det är för sent.

För att ROV:en skall kunna avvägas i vattnet måste det finnas utrymme för att balansera och lägga till vikter. Eftersom ROV:en är så gott som helt symmetrisk i sidled behöver den förmodligen inte balanseras för att förhindra att den kantrar. Däremot är den på grund av kablar, batterier och kamera inte avvägd i längsgående riktning. Det måste därmed finnas någon funktion i ramverket som gör att den går att balanseras i denna led, så att inte framänden tippar nedåt eller uppåt. Vikter för att göra ROV:en neutral i vatten bör av stabilitetsskäl placeras så lågt som möjligt. Om farkosten likväl blir ostabil eftersom lyftkraftcentrum och tyngdpunktscentrum hamnar för nära varandra i vertikalled måste detta justeras genom att antingen flytta ner vikterna ännu lägre eller lägga till en flytkropp, i form av ett rör, överst.

En kropp som fästs med gängstänger i flänsarna bak och fram anses vara den bästa lösningen. Om kroppen skulle bli för lång kan en extra ring på mitten behövas för att stabilisera. I dessa gängstänger kan sedan motorer, vikter och medar hängas på och fästas med muttrar. Eftersom dessa enbart sitter i gängstängerna kan de flyttas fram och tillbaka vilket gör det enkelt att balansera farkosten.

Placering av belysning

Belysningen placeras långt fram och helst i underkant, detta eftersom botten generellt är mer intressant att belysa samt eftersom det är viktigt att inte blända kameran. Minst två lampor ger ROV:en en symmetrisk ljusbild men givetvis kan flera lampor användas vilket gör att de kan placeras både högt och lågt. För kamerans skull bör de dessutom sitta brett isär för att ge ett så spritt ljus som möjligt, speciellt om kameran är rörlig då det är smidigare än att även lamporna rör sig.

Ytterligare en sak som måste tas hänsyn till är att lamporna inte får placeras så att de stör kameran genom reflektioner, detta är framför allt ett argument till att inte placera dem inuti skrovet. Ytterligare en anledning till att inte placera lamporna inuti skrovet är värmeutvecklingen. Sitter de ute i vattnet finns ingen risk för att de överhettas.

Det mest fördelaktiga anses utifrån detta att placera lamporna långt fram, lågt och brett. Detta gäller främst för två lampor men skulle fler användas kan även några placeras högt.

Lamporna skulle kunna placeras längst fram på vardera av de två medarna under skrovet. Detta hade gett en bra, låg och bred placering. Dock finns risken för att dessa går ner i bottenslammet ofta och risken finns då att lamporna blir igenslammade och därför förkastades denna idé.

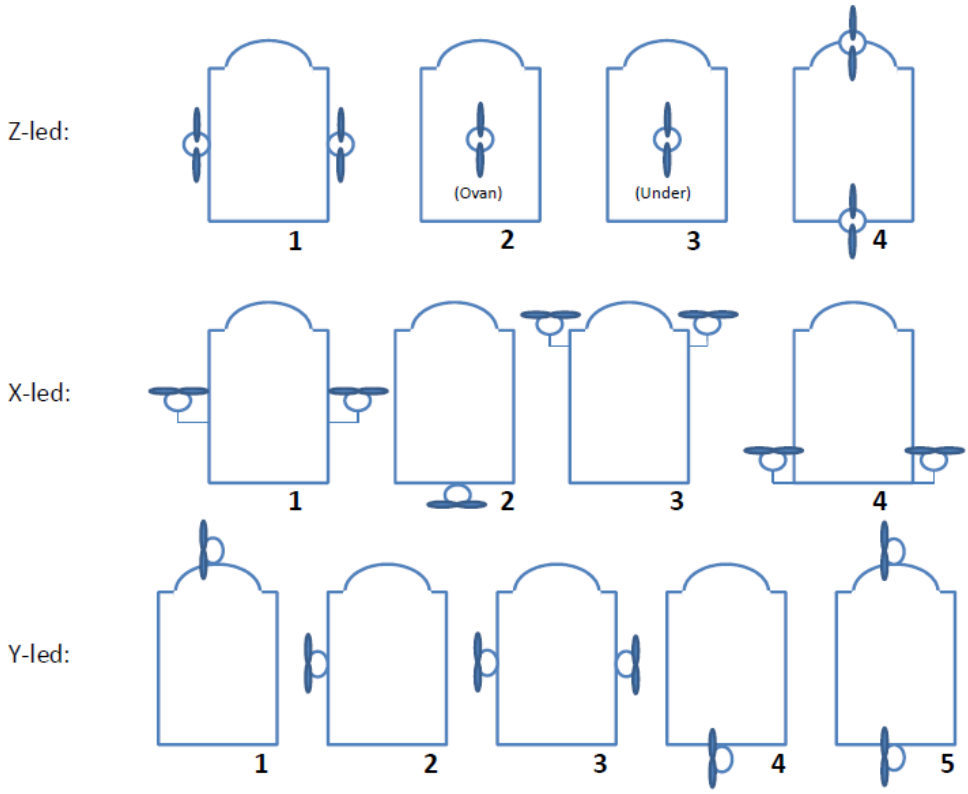
Förslag till motorplacering

ROV:ens skrov är centralt placerad och motorernas placering får anpassas utifrån denna. De djupreglerande propellrarna måste vara under vattnet när ROV:en är i ytläge, så att den kan dyka utan att någon behöver trycka ned den. De bör dock inte sitta så lågt att de rör upp för mycket sediment från botten. Det nedströmmande vattnet skulle kunna styras åt sidan av plattor men detta försämrar samtidigt framdriften.

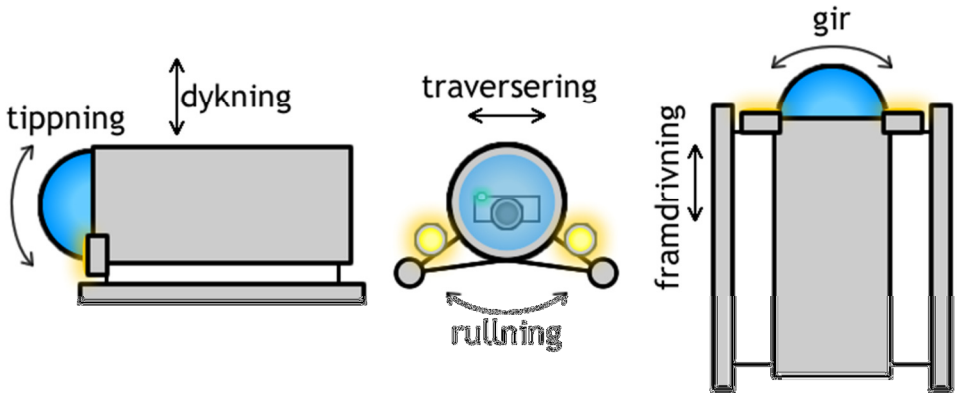
Propeller för förflyttning i sidled behövs i ett första stadie inte för en ROV som bara filmar, då den inte har så stort behov av att hålla sig på samma ställe som en ROV som skall reparera något.

Skrovet måste vara symmetriskt och propellrarna måste sitta i centrum av sin förflyttningsriktning för att inte få slagsida av vattenmotståndet när den rör sig. Då konstruktionen helst bör vara så enkel som möjligt kommer inga fenor eller roder användas som annars skulle kunna kompensera för detta. Navelsträngen kommer kunna innebära störningar i framdriften då den bromsar lite när ROV:en kör. Detta är framför allt ett problem då mycket kabel ligger i vattnet och måste dras med. På samma sätt kan strömmar och vågor lätt dra i kabeln vilket medför att ROV:en flyttas eller svängs ofrivilligt. För att kunna få farkosten att röra sig med liten påverkan av navelsträngen bör motorfästena gå att justera så att motorerna kan flyttas, både bakåt eller framåt och uppåt eller nedåt.

Förslag till motorplacering togs fram och sammanställdes, se Figur 3.1. Den konvexa änden på ROV:en i skissen är framänden och således är z-led upp/ned, x-led fram/bak och y-led är sidled. Även vinklade motorer eller rör genom farkostkroppen med motorer i diskuterades men då dessa lösningar ansågs för komplicerade att realisera förkastades de.



Figur 3.1 - Förslag till motorplacering



Figur 3.2 - Frihetsgrader för ROV

Lösningsförslagen för motorplacering har tagits fram och finns att se i Figur 3.1. De har sedan ställts mot varandra för att ta fram ett högsta prioriterat förslag i varje riktning. Genom olika viktade önskemål har förslagen jämförts i Tabell 3.3, Tabell 3.4 och Tabell 3.5. Där har de rankats hur väl de uppfyller varje önskemål på en skala från 1-5. ROV:ens frihetsgrader visas i Figur 3.2 och dessa ord kommer användas i texten och tabellerna. Eftersom det anses självklart att motorplaceringen i z-led och x-led skapar dykning respektive framdrivning, då detta är deras främsta syfte, har dessa inte tagits med som önskemål.

Det är mycket viktigt att ROV:en kan förflytta sig uppåt och nedåt i vattnet. I denna riktning har även motorerna störst risk för att röra upp bottensediment då de accelererar ROV:en uppåt nära botten. Ifall vattnet blir grumligt kan det ta flera minuter innan detta lägger sig igen, därför är detta av högsta vikt i Tabell 3.3. Näst viktigast anses det vara att den är energisnål och att inte propellrarna hamnar över vattenytan då ROV:en placeras i vattnet. Näst efter detta är det viktigt att motorerna i z-led har stor kraft och kan förflytta ROV:en snabbt. Det minst viktiga anses vara tippning, rullning och estetik. För en förklaring av tippning, rullning och andra frihetsgrader för en ROV, se Figur 3.2. Tippning och rullning anses inte vara så viktigt eftersom ROV:ens balansering kommer motverka dessa rörelser och estetiken är enligt våra önskemål och avgränsningar lågt prioriterat. En jämförelse mellan de olika alternativen finns i Tabell 3.3.

Tabell 3.3 - Motorplacering i z-led

Z-led	Alt. 1	Alt. 2	Alt. 3	Alt. 4	Viktning
Energisnål	3	4	4	3	0,7
Kraftfull	5	3	3	5	0,5
Placerad under ytan	4	1	5	4	0,7
Rör ej upp sediment	4	5	1	4	0,8
Tippning	1	1	1	5	0,3
Rullning	5	1	1	1	0,3
Estetik	5	3	2	2	0,3
Summa	13,9	10,5	9,8	13	

Det är även viktigt att ROV:en kan förflytta sig framåt, bakåt och svänga (gir). Högst prioriterat är att motorplaceringen medför att den kan svänga och att den är energisnål. Därefter anses det vara viktigt att kameran ej störs genom vattenströmmar eller roterande delar. Efter det anses det prioriteras att den är kraftfull. Sedan väljs rotation kring egen axel då detta underlättar för operatören vid körning i trånga utrymmen eller vid svängning med högre precision. Lägst prioriterat är estetiken. Jämförelsen mellan de olika alternativen finns i Tabell 3.4.

Tabell 3.4 - Motorplacering i x-led

X-led	Alt. 1	Alt. 2	Alt. 3	Alt. 4	Viktning
Energisnål	3	4	3	3	0,7
Kraftfull	5	3	5	5	0,5
Stör ej kameran	4	5	2	5	0,6
Gir	5	2	5	5	0,7
Gir kring egen axel	5	2	3	2	0,4
Estetik	5	4	4	5	0,3
Summa	14	10,7	11,7	13,4	

Att ROV:en kan förflytta sig i sidled är inte särskilt högt prioriterat, skulle dock den funktionen byggas in bör ändå de olika lösningsförslagen vara utvärderade. Sidledsförflyttning kan vara användbart då det ökar möjligheten att effektivt förflytta sig runt ett objekt och samtidigt observera objektet eftersom inget verktyg skall tillverkas till farkosten

så är denna förflyttning inte lika viktig. Dessutom så krävs extra komponenter vilket leder till ökad komplexitet vilket är en nackdel.

Eftersom motorerna i y-led kan användas enbart för gir-funktionen finns dock även traversering med som önskemål. Traversering är då givetvis högst viktat, eftersom den funktionen är viktigast med denna motorplacering.

Det näst viktigaste anses vara energisnålhet, följt av att kameran inte störs. Därefter kommer kraftfullheten, följt av gir. Precis som för motorplaceringen i x-led är gir bara viktigt då inte den funktionen kan lösas av motorerna i x-led. I sidled önskas även en god symmetri så inte ROV:en blir obalanserad. Detta går dock lätt att justera på den färdiga produkten och därför är det tillsammans med estetiken lägst prioriterat. Jämförelsen mellan de olika alternativen finns i Tabell 3.5.

Tabell 3.5 - Motorplacering i y-led

Y-led	Alt. 1	Alt. 2	Alt. 3	Alt. 4	Alt. 5	Viktning
Energisnål	5	4	3	5	3	0,7
Kraftfull	3	2	4	3	5	0,5
Stör ej kameran	1	3	3	5	1	0,6
Gir	4	1	1	4	5	0,4
Traversering	1	4	5	1	5	0,8
Symmetri	2	1	5	2	5	0,3
Estetik	1	1	5	2	4	0,3
Summa	8,9	9,8	13,3	11,6	13,9	

Slutsatsen från tabellerna är att en lösning med två propellrar, en framåtriktad och en uppåtriktad, på varje sida mitt på ROV:en är den bästa lösningen. Skulle traverserings funktionen önskas är det bästa alternativet en motor längst fram och en längst bak, tätt följt av ytterligare en motor på varje sida i mitten av ROV:en.

Två motorer för framdrivning och svängning i xy-planet valdes, placerade på varsin sida om skrovet placerade nära centrum av kroppen. Ren sidledsförflyttning kräver en komplexare lösning och är inte nödvändigt i en ROV avsedd för filmning. För stigning och dykning (z-led) en lösning med två motorer placerade högt på var sin sida valts. Propellrarna är då i vatten när farkosten är i ytläge och samtidigt placerade så långt från botten som möjligt, för att minska risken för att röra upp bottensediment.

Slutgiltigt lösningsförslag för farkostkropp

Utifrån alla dellösningarna har ett bästa alternativ tagits fram. Hur lösningen ser ut syns i Figur 3.3.

Skrovet är cylinderformad och denna cylinder är i aluminium. Framstycket är i plexiglas och denna sitter tillsammans med en aluminiumfläns fast i cylindern. Bakstycket är en tjock aluminiumskiva där den inre delen sticker in i den cylinderformade kroppen. Där inne sitter två O-ringar mot insidan av cylindern för att täta.

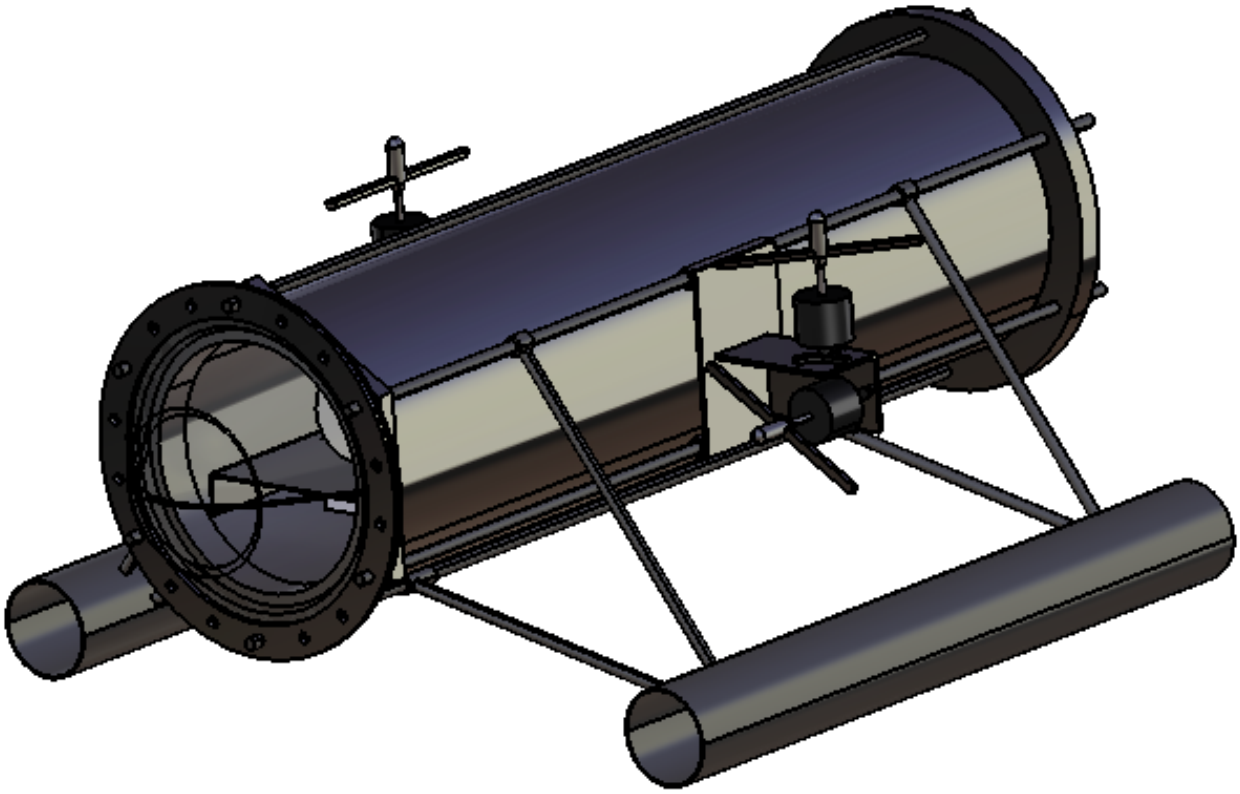
Mellan fram- och bakstycket går sex gängstänger på vilka motorer, vikter och medar är fästa. Motorerna sitter i vinkeljärn som i sin tur sitter på plattor vilka hänger i de två vinkeljärnen som går på sidan. Dessa hålls på plats med muttrar och motorskydden anpassas efter hur lösningen kommer bli i realiteten. Offeranoder av zink placeras för säkerhets skull ut på skrovet då gängstängerna kommer vara i stål och sitta i aluminiumflänsarna.

Medarna är tillverkade i aluminium och sitter fast i gängstängerna. Medarna kan skjutas framåt eller bakåt för att balansera ROV:en i längsgående riktning och fästs också med muttrar i den mittersta gängstången under skrovet. Lamphusen skall fästas i framstycket, alternativt i gängstängerna.

Kroppen har diametern 150 mm och är 500 mm lång. Om mer flytkraft skulle behövas kan dock kroppen förlängas eller i andra hand diametern ökas.

All elektronik placeras i en släde som ligger i skrovet. Denna sitter fast i bakstycket där kabelgenomföringarna går ut. Eftersom framstycket sitter fast i kroppen kan ROV:en demonteras utan att några kablar behöver lossas. Detta kan göras genom att skruva av muttrarna till gängstängerna fram för att sedan dra av framstycke och huvudkropp från gängstängerna vilket frilägger all elektronik medan motorer och stödrören underst fortfarande hänger på gängstängerna som sitter fast i bakstycket.

Skulle det bli svårt att komma åt elektroniken kan den översta gängstången dras ut med skrovet ifall dess mutter lossas i bakstycket istället för där fram.



Figur 3.3 - Slutgiltigt koncept för farkostkroppen

3.1.2 Strömförsörjning

Att mata farkosten med ström från ytan har undersökts som ett alternativ till att ha ett batteri ombord. En annan möjlighet är att kombinera strömmatning från ytan och batteri i farkosten. Beräkningar visar dock att spänningsfallet och därmed effektförlusten i kabeln kan bli betydande vid lägre spänningar. För att minska spänningsfallet kan en högre spänning användas, det minskar strömmen som flyter. Dock krävs det speciell utbildning att konstruera utrustning med höga spänningsnivåer.

I Tabell 3.6 uppskattas hur hög effekt de elektroniska komponenterna kommer förbruka.

Tabell 3.6 - Tidigt uppskattad effektförbrukning

Komponent	Kamera	Motorer	Motorstyrning	Styrenhet	Lampor	Övrigt
Effekt [W]	5	150	5	10	30	100

En snabb överslagsberäkning visar att ROV:en behöver kunna nyttja 300W i tio minuter vid spänningsnivån 12V. Detta ger en minsta batterikapacitet på 4,17Ah. Batterikapaciteten kan beräknas med följande formel:

$$K = \frac{P}{U} * \frac{t}{60} \qquad (1)$$

Överslagsberäknad maximal effekt, P = 300 [W]

Minsta specificerade körtid vid max effekt, t = 10 [min]

Spänningsbehov, U = 12 [V]

Minsta batteri kapacitet, K = (P/U) * (t/60) = (300/12) * (10/60) = 4,17 [Ah]

Extern energilänk

Energiförsörjning till farkosten via extern energikälla kräver relativt grova kablar. Med 100m kabel och den uppskattade effekten på 300W finns ett par alternativ för energiförsörjning till farkosten från ytenheten. Alternativen som jämförs är 12V med full effekt, 48V full effekt, 48V med låg effekt för att driva styrenhet och slutligen ingen energiöverföring. Då 12V med full effekt både ger en hög effektförlust och tung kabel är den lösning ej realiserbar. Övriga lösningar har bedömts och viktats i tabellen som finns i Appendix C - Energiförsörjning.

Ytterligare ett alternativ hade varit att skicka 230V till ROV:en. Med så hög spänning blir effektförlusterna i kabeln låga. Detta hade dock krävt tillgång till vägguttag, vilket kraftigt begränsar var den kan köras. Dessutom krävs speciell utbildning för att handhava så hög spänning. Av dessa anledningar har strömförsörjning med 230V uteslutits. Alternativet att ha flera seriekopplade12V-batterier vid ytan är heller inte aktuellt då detta skall kunna förflyttas enkelt av en person.

Hybridsystem

Den främsta tanken med att ha ett hybridsystem är att ROV:ens styrenhet kan hållas vid liv från ytan medan ett batteri förser de mer strömkrävande komponenterna så som motorer och belysning med ström. Detta ger möjlighet att kommunicera med ROV:en även utan batterispänning. Detta kan användas för att hitta fel, upptäcka läckage och dylikt. Om högre strömmar skickas från ytan kan även batteriet laddas under körning. Detta ökar körtiden och ger föraren möjligheten att låta ROV:en ligga still och ladda batterierna under vatten. Nackdelen med en hybridlösning är att den både får nackdelarna som batterilösning har och de som extern energilänk har.

Val av energiförsörjning

Först och främst jämfördes alternativet att strömförsörja ROV:en från ytan mot att ha en intern strömförsörjning, eller en kombination av dessa. Dessa förslag jämfördes sedan sinsemellan och detta finns presenterat i Appendix C - Energiförsörjning.

Det alternativ som uppfyller kraven på bästa sätt är alternativet med enbart batterier i farkosten och utan energiöverföring. Den främsta anledningen är komplexiteten, det är enklare att konstruera ett system med batterier i farkosten. Samtidigt blir det mindre energiförluster då avstånden inuti ROV:en är små. Till de stora nackdelarna hör att det tar relativt mycket plats i farkosten och laddning av batteriet blir svårare.

Batterival

Då det bästa alternativet ansågs vara att ha batterier i farkosten måste det bästa valet av batterier utvärderas. Information om olika alternativ togs fram och jämfördes sedan mot varandra, detta återfinns i Appendix D - Batterialternativ.

Det alternativ som på bästa sätt uppfyller kravspecifikationen är Li-Po. De största fördelarna med alternativet är kapaciteten och energitätheten. De stora nackdelarna är att de är svåra att ladda, måste hanteras varsamt och kan ta eld vid felaktig hantering. Om batterierna placerats vid ytan och energiöverföring använts hade vikten spelat mindre roll, vilket med stor sannolikhet hade föranlett en annan batterityp som kostar mindre.

3.1.3 Motorval

Framdriften delades upp i två delar. Först utvärderades om motorerna skall monteras i ett slutet hus avskilt från vattnet eller inte. Att ha motorerna i en tät kropp ger många fördelar då motorer och växellådor oftast ej är särskilt hållbara i vatten och i synnerhet då saltvatten. Nackdelarna är att lösningen blir dyrare och mer komplex än alternativet att ha motorer och eventuella växellådor liggande fritt i vattnet.

Det finns olika typer elmotorer som kan användas. En likströmsmotor med borstar har som fördel att den kan styras linjärt med hjälp av spänningen, nackdelen är att den måste vara avskild från vattnet då den annars kortsluts. Borstlösa motorer har fördelen att alla ledningar i motorn kan lackas så att de är åtskilda vilket betyder att de kan köras i vatten som de är. Nackdelar med borstlösa motorer är att hastigheten regleras av växelspänningens frekvens vilket betyder att elektronik som genererar växelspänning behövs. Sådan elektronik har också sina begränsningar och utan sensorer på motorerna så kommer dessa rotera på relativt hög hastighet.

Propellrarna för en ROV bör flytta stora mängder vatten långsamt eftersom turbulenta strömmar och hastiga vattenflöden lätt rör upp skit från botten. En sådan propeller kräver hög effekt per varv vilket inte passar optimalt för elmotorer som oftast snurrar mycket snabbare än vad det önskade propellervarvtalet är. Därför bör en växellåda användas mellan propeller och motor. Nackdelarna med växellåda är att de oftast ej är hållbara i saltvatten och att de är relativt dyra.

Tabell 3.7 - Utvärdering av motorhus eller outrunner

Egenskap	Slutet hus	Outrunner	Vikt
Pris	0	1	4
Enkel lösning	0	1	3
Möjlighet till växellåda	1	0	2
Motorvalfrihet	1	0	1
Summa	3	7	

Tabell 3.8 - Utvärdering av motorval

Egenskaper för motorer	Borstlös	Borstlös sensor	Borstmotor	Vikt
Verkningsgrad	4	4	4	1
Robusthet	5	4	3	4
Steglöshet	3	5	5	2
Komplexitet	3	2	5	3
Summa	39	36	41	

Den lösning som valdes var borstlösa motorer med relativt små propellrar direkt på motoraxeln. Dessa skall placeras på utsidan av farkosten utan någon avskiljning mot vattnet. Argumentet är att denna lösning är enkel, priseffektiv och att liknande projekt har studerats där just den typen av motorer använts och det har fungerat(Jarvoll, 2009).

3.1.4 Belysning

Det är mycket viktigt med stark belysning under vattnet eftersom operatören skall kunna se och styra farkosten. Om kamerans ljuskänslighet är låg krävs det en stark ljuskälla för att bilden skall bli bra.

Utvärdering av belysningsalternativ

För att ta fram det belysningsalternativ som passar farkosten bäst har ett antal intressanta alternativ bedömts och viktats.

Tabell 3.9 - Utvärdering belysning

Lösningsförslag	Fördelar	Nackdelar
LED, bygga själv	Anpassningsbart	Komplext
LED, köpa färdig	Enkelt	Mindre ljusstark, dyrare
HID, bygga själv	Finns vattentäta lampor	Utvecklar mer värme
Köpa färdig dyklampa	Befintliga och täta lösningar	Dyrt!

För att undersöka vilket alternativ som är lämpligast har de blivit rankade inom varje relevant område.

Tabell 3.10 - Rankning av belysningsalternativ med avseende på egenskaper

Egenskap	LED, bygga själv	LED, köpa färdig	HID, bygga själv	Köpa färdig dyklampa
Ljusstyrka	5	2	5	5
Spridning av ljus	5	4	5	4
Komplexitet	3	5	1	5
Jobb	2	4	1	5
Verkningsgrad	5	5	2	4
Pris	4	3	3	1
Summa	24	23	17	24

I egenskapstabellen så är det väldigt jämnt mellan de tre bästa alternativen. För att enklare kunna ta ett beslut har varje egenskap viktats enligt Tabell 3.11.

Tabell 3.11 - Viktning av belysningsegenskaper

Egenskap	LED, bygga själv	LED, köpa färdig	HID, bygga själv	Köpa färdig dyklampa	Viktning
Ljusstyrka	1,25	0,5	1,25	1,25	0,25
Spridning av ljus	0,4	0,32	0,4	0,32	0,08
Komplexitet	0,36	0,6	0,12	0,6	0,12
Jobb	0,4	0,8	0,2	1	0,2
Verkningsgrad	0,5	0,5	0,2	0,4	0,1
Pris	1	0,75	0,75	0,25	0,25
Summa	3,91	3,47	2,92	3,82	-

Efter viktningen visar tabellen att två alternativ utmärker sig, att bygga en egen LED-lampa eller att köpa en färdig dyklampa. Vid konsultation med en dykare framgick att en dyklampa med ljusstyrkan 400 lumen fungerar väl. Utifrån detta har beslutades att ROV:en borde ha lampor av den dubbla ljusstyrkan för att säkerställa sikten. Tidigt i projektet införskaffades två stycken LED om 800 lumen för tester. Testerna gav ett väldigt gott resultat.

För att kunna använda LED under vatten behöver de sitta i ett vattentätt lamphus. För att ljuset skall nå ut måste det finnas ett klart stycke plast eller glas i ena ändan av lamphuset. Plast väljs med fördel då det inte går sönder lika lätt som glas. Nackdelen med plast är dock att det inte tål värme lika bra som glas. Två sorters plast har undersökts, polykarbonat och plexiglas. Polykarbonat är bättre än plexiglas då det tål högre temperaturer.

Ett första lamphus byggdes för att testa dess funktion och utvärdera valet av konstruktion. Lamphuset tillverkades i aluminium och polykarbonat och tätningen mellan delarna gjordes med epoxylim. Lamphuset fylldes med vatten invändigt och inga läckor hittades. Slutsatsen som drogs från detta var att det går utmärkt att vattentäta med epoxylim.

Nästa kritiska del att undersöka var kylningen eftersom LED utvecklar en icke försumbar mängd värme. Testet genomfördes i två steg, ett i vatten och ett i luft. En LED monterades i det sedan tidigare tillverkade lamphuset och sänktes därefter ner i vatten och kopplades till 12V. Testet visade att LED:arna drar 1A vid 12V, det vill säga effekten är 12W. Det konstaterades också att värmeavledningsförmågan i husen var tillräckligt bra då det inte blev någon märkbar skillnad i temperatur. Testet fortsatte genom att testköra LED:arna i luft. Efter ca tio minuter vid maximal effekt konstaterades att värmeavledningsförmågan i husen inte är tillräckligt bra för att kunna köra dessa en längre period i luften. Det konstaterades att det krävs mycket kylning om LED skall användas i lamphusen.

Det gjordes även ett tredje test med ett nytt lamphus där glaset klämdes fast med en o-ring som tätning istället för att använda expoxylim. Detta hus blev också helt vattentätt, och är att föredra då det är lätt att öppna det huset.

Reglering av belysning

För att inte använda onödigt mycket energi till belysning kan lampornas uteffekt regleras. För att åstadkomma detta moduleras signalen från styrenheten till lamporna väldigt fort med en teknik som kallas för pulsbreddsmodulering. Styrenheten kan dessvärre inte leverera den effekt som krävs och får därför reglera en drivkrets som i sin tur driver lamporna. De LED:s som är tilltänkta drar upp till 20W vardera. Utifrån det valdes MOSFET:ar som klarar effekten.

3.1.5 Kamera

En av de viktigaste funktionerna på farkosten är kameran, då den ger den huvudsakliga feedbacken till operatören vid manövrering. För att göra ett objektivt kameraval måste många aspekter vägas mot varandra. En viktig faktor som ej går att bortse från är priset, en bra kamerabild vid dåliga ljusförhållanden kostar en ansenlig summa pengar.

Nästa egenskap av intresse var ljuskänsligheten hos en kamera. Samma webkamera placerades i en ryggsäck tillsammans med en ljuskälla. Av det testet konstaterades att en kamera med mycket stor ljuskänslighet kommer vara viktigt om kamera skall kunna registrera intressanta ting under vattnet.

I Tabell 3.12 poängsätts de olika kameralösningarnas egenskaper på en skala på 1-5. Därefter multipliceras poängen med egenskapens vikt och en total summa för varje kameralösning återfinns längst ner.

Tabell 3.12 - Utvärdering av kameraalternativ

Egenskaper	IP-Kamera	USB-kabel Webbkamera	+	Koaxialkabel Kamera	+	Vikt
Bildkvalité	5	4		2		0.3
Kostnad	3	3		2		0.5
Enkelhet	5	3		5		1
Ljuskänslighet	4	2		4		1
Storlek	2	5		5		0.2
Pan/tilt	5	2		2		0.3
Summa	13.9	9.3		12.2		

Efter att egenskaper vägts mot varandra visar det sig att en IP-kamera är bäst lämpad för projektet. En IP-kamera som väl uppfyller de efterfrågade egenskaperna samt dessutom är relativt prisvärd är VIVOTEK FD8133. Den levererar video med 1MP upplösning i 30 bilder per sekund, dessutom har den inbyggd pan och tilt.

Projektet tilldelades en mycket ljuskänslig IP-kamera av modell Q1602 av företaget Axis. I och med detta valdes att använda Axis kameran.

3.1.6 Styrenhet

I styrenheten måste signaler från ytenheten bearbetas och omvandlas till styrsignaler för ROV:ens ställdon. Önskvärt är också att skicka tillbaks bearbetad sensordata. För att utföra detta behövs en mikroprocessor som bearbetar data mellan in- och utgångar. Vid val av mikroprocessor finns många alternativ att välja på, dock undersöktes endast Atmels Mega serie och Arduino-systemet. Det är en plattform som innefattar en Atmel mikroprocessor samt all elektronik som behövs för att programmera och köra den. Två av Arduino modeller har undersökts, Arduino MEGA och Arduino UNO. En jämförelse av deras egenskaper finns i Tabell 3.13.

Tabell 3.13 - En jämförelse av egenskaperna mellan Arduino Mega och Uno

Egenskap	MEGA	UNO
In- & Utportar	54st	14st
PWM kanaler	14st	6st
Servo kanaler	48st	12st
Flashminne	256KB	32KB
Klockfrekevens	16MHz	16MHz
Kostnad (Farnell)	543 sek	262 sek

Båda modellerna har tillräckligt med in- och utgångar samt motorstyrmöjligheter för att kunna styra ROV:en. UNO modellen har precis så många portar som behövs och därmed finns ingen möjlighet till framtida påbyggnader. Därmed beslutades att MEGA-modellen lämpar sig bäst för att styra ROV:en.

3.2 Kommunikationslänk

Styrenheten i farkosten skall använda sig av en mikroprocessor och i ytenheten skall en logisk enhet av någon typ finnas. Dessa skall därefter kommunicera med varandra med hjälp av ett lämpligt gränssnitt. Vidare skulle farkosten kunna köra fast och då kan en dragsäkring vara praktisk för att dra upp den. Kommunikation till kameran sker via Ethernet eftersom en IP-kamera valdes.

3.2.1 Gränssnitt

Vid överföring av styrsignaler och information är det viktigt att ha ett datagränssnitt som är lämpligt för den specifika situationen. Flera olika gränssnitt studerats och sammanställts nedan.

UART ⇔ UART/USB

Mikroprocessor med UART ⇔ splitter/joiner ⇔ CAT-kabel ⇔ splitter/joiner ⇔ Logisk enhet

Denna lösning har testats med ca 40 meter kabel och verkar fungera bra. Testmiljön var dock relativt störningsfri. Det är ej säkert att lösningen fungerar då motorerna i ROV:en kommer generera mycket störningar. Eventuell felhantering måste kodas från grunden.

UART ⇔ RS485 ⇔ UART/USB

Mikroprocessor med UART ⇔ RS485 Driver ⇔ splitter/joiner ⇔ CAT-kabel ⇔ splitter/joiner ⇔ RS485 Driver ⇔ Logisk enhet

Denna lösning är i princip densamma som föregående. Skillnaden är att en RS485 drivkrets kopplats in innan signalen skickas i CAT-kabeln. Därmed omvandlas pulserna i signalen till en differens mellan två ledare. RS485 är utformat för att tåla störningar genom att bara mäta skillnaden mellan ledarna och är specificerat att kunna överföra data på sträckor upp till 1200m. Dock måste en eventuell felhantering fortfarande skrivas från grunden. (Bies, 1997)

Ethernet

En möjlig lösning är att låta mikroprocessorn kommunicera direkt med en PC, i ytenheten, via ethernet. En sådan lösning är dock svår att implementera på mikroprocessorns sida, eftersom ethernet kräver mycket överlagrad data på det som skickas. Dessutom behövs en router om en IP-kamera används.

USB Server

Om en styrenhet och en webkamera ansluts till en USB-server i farkosten kan denna överföra data till en logisk enhet på ytan. Det är dock osäkert huruvida styrenheten fungerar tillsammans med en USB server då det inte finns dokumenterade tester av detta. Det är troligt att en ny drivrutin skulle behöva skrivas och därmed blir lösningen mycket komplex. Hade det funnits färdiga drivrutiner skulle denna lösning vara mycket enkel att genomföra. Dock skulle valet av kamera vara låst till en webbkamera.

CAN-bus

CAN används i många industriella applikationer och bygger på ett bussystem. Det återfinns i stort sett i alla moderna fordon utom riktigt stora som till exempel oljetankers. Den stora fördelen med CAN är att det är ett färdigt interface som har en inbyggd felhantering. Nackdelen är att det är ett förhållandevis komplext system att sätta upp innan det fungerar.

Slutsats

Gränssnittet som skickar data via RS485-standarden valdes. Detta då det är utformat för att skicka data över långa sträckor i miljöer där det förekommer mycket störningar. Samtidigt är det ett relativt enkelt system att använda och implementera. Eventuellt kan en USB till RS485 adapter användas och mikroprocessorn i ytenheten elimineras.

3.2.2 Dragsäkring

Ytterligare länk mellan ytan och farkosten som dragsäkring är överflödig eftersom den skulle främst vara klumpig och bidra med extravikt. En alternativ lösning är att dragsäkra den kommunikationslänk som ändå måste gå mellan ytenheten och farkosten genom att använda en tillräckligt dragstark kabel.

3.2.3 Kabeltyp

Då projektet har en begränsad budget och då endast signaler behöver skickas via kommunikationslänken valdes att använda vanlig nätverkskabel.

3.3 Ytenhet

För att kunna styra farkosten behövs en ytenhet utrustad med styrdon, display och direkt kommunikation till farkosten. Hur denna är tänkt att konstrueras återfinns i efterföljande stycken.

3.3.1 Display

För att operatören överhuvudtaget skall ha en möjlighet att styra farkosten behöver denne se vad farkosten ser. Detta möjliggörs genom den tidigare omnämnda kameran som genererar bild till en display. Olika displayalternativ har jämförts nedan.

Bärbar dator

En bärbar dator innefattar ett helt datorsystem, inklusive display, vilket medför flera fördelar. Ur en ren displaysynvinkel är datorns skärm delvis inkapslad, men med stor sannolikhet inte vattentät. Detta kan avhjälpas genom att tillverka en genomskinlig panel eller liknande. De absolut största fördelarna är dock att datorn har gränssnitt för in- och utsignaler, stöd för en mängd olika kameror och internt lagringsutrymme om operatören vill spela in video. Det kan dock vara både till för- och nackdel att bygga in en bärbar dator eftersom kompabiliteten kan utebli vid ett eventuellt byte av dator.

Tabell 3.14 - För- och nackdelar med bärbar dator

Fördelar	Nackdelar
Enkelt	Dyr
Många anslutningar	Ej vattentålig
Kan spara video och bilder	Kan bli kompabilitetsproblem
Kan visa information från ROV:en på samma skärm som video	
Begränsar inte valet av kamera	
Programvaran kan flyttas till en annan dator	

Bildskärm

En liten bildskärm likt de som används i bilar och båtar är smidig och relativt lätt att bygga in i ett skal som kan göras vattentätt. Om bildskärmen skulle krångla eller gå sönder går det att byta ut den utan att byta ut andra komponenter i ytstationen. Dessa bildskärmar har generellt få anslutningar och kräver i många fall någon form av omvandling av videosignalen innan den kan anslutas. Därför används de med fördel tillsammans med en analog kamera eftersom videosignalen då kan kopplas rakt in i enheten. De kan inte spara media utan ytterligare komponenter.

Tabell 3.15 - För- och nackdelar med bildskärm

Fördelar	Nackdelar
Enkel att byta	Kan inte byta videoformat
Behöver ingen codec	Få ingångar
Enkel att skydda mot vattenstänk	Begränsar valet av kamera
	Kräver fler delar vilket leder till mer komplex konstruktion
	Kan inte spara video eller bilder

TV

En viktig skillnad mellan en TV och en vanlig bildskärm är att TV:n har andra anslutningar och kan ta emot signaler via en ansluten antenn. Den kräver en analog videosignal in och används först och främst tillsammans med en analog kamera eftersom videosignalen annars måste konverteras.

Smartphone eller surfplatta

Att använda en smartphone eller surfplatta är ett alternativ som verkligen ligger i tiden. Det kräver dock en digital kameralösning.

Tabell 3.16 - För- och nackdelar med smartphone eller surfplatta

Fördelar	Nackdelar
Nyskapande	Dyrt
Tar lite utrymme	Svårt att hitta bra codec
	Få ingångar
	Låser valet av kamera

Slutsats

Utifrån detta anses en dator som ytstation för videouppspelningen vara det bästa valet. Då kan styrning och andra funktioner på ytan knytas samman i en komponent och operatören kommer kunna spara video eller stillbilder. En dator ger dessutom störst flexibilitet när det kommer till val av kamera eftersom både USB- och IP-kameror är gjorda för att användas med dator och koaxialkameror kan skicka video till datorn efter konvertering. Till projektet erhölls en bärbar dator. Därmed löstes detta gratis, där annars priset var den största nackdelen.

3.3.2 Styrdon

ROV:en kommer behöva kontroller för att köras framåt och bakåt, för att svänga och för att stiga eller dyka. Dessa kontroller skall helst vara steglösa. Dessutom behövs kontroller för att slå av eller på belysningen. Ytterligare funktioner som kan behövas är hantering av kameran, inspelning av video och kontroller för framtida verktyg till farkosten.

Efter genomgång av andra projekt och informationssamling har det konstaterats att det finns fyra relevanta metoder för att generera styrsignaler.

- En handkontroll från en Xbox 360 kopplad till en dator är ett enkelt sätt att generera styrsignaler. Kontrollen har tio knappar, två joysticks och ett styrkors.
- En mikroprocessor till vilken knappar och reglage ansluts. Antalet knappar och reglage som kan anslutas begränsas av mikroprocessorns ingångar.
- En handkontroll från ett Wii som ansluts till en mikroprocessor. Handkontrollen har en tre-kanals accelerometer, två knappar och ett joystick.
- Använda tangentbordet på ytenheten under förutsättningen att ytenheten är en dator.

De olika alternativens för- och nackdelar har studerats och utifrån det har ett beslut tagits.

Tabell 3.17 - Utvärdering av styrdon

Alternativ	Fördelar	Nackdelar
Xbox 360-kontroll	Ergonomisk, enkelt API, lätt att integrera med nätverkskamera, lätt att programmera	Kräver dator som ytstation
Mikroprocessor med knappar	Enkelt, billigt	Tar tid att tillverka, oergonomiskt, ytterligare enheter för videouppspelning
Wii-kontroll och mikroprocessor	Lätt att programmera, ergonomisk	Få knappar, ytterligare enhet för videouppspelning
Tangentbord till dator	Enkelt, lätt att integrera med nätverkskamera, många knappar	Kräver dator som ytenhet, inga analoga insignaler

Antalet analoga och digitala reglage såväl som möjlighet till framtida utbyggnad, samt komplexitet vid programmatisk integration, spelade in vid val av styrdon. Alternativen rangordnades utifrån nämnda egenskaper och det faktum att projektet tilldelats en bärbar dator som logisk enhet på ytan.

Tabell 3.18 - Rangordning av styrdon

Alternativ	Xbox 360-Kontroll	Mikroprocessor Med knappar	Mikroprocessor Med Wii-kontroll	Tangentbord till dator
Rank	1	2	4	3

4 Resultat

I detta kapitel beskrivs tillverkningen av ROV:ens ingående delar och tester som gjorts för att verifiera om kravspecifikationen uppfyllts.

4.1 Tillverkning

Till mittmötet som hölls halvvägs in i projektet lades alla lösningsförslag ihop till ett slutgiltigt koncept. Efter detta startade tillverkningen av ROV-systemet.

Tillverkningen av alla ROV:ens ingående delar har i största möjliga utsträckning skett i parallella spår. Det ena spåret har konstruerade och tillverkade de mekaniska delarna och det andra spåret tillverkade elsystemet och utvecklade programkoden. Majoriteten av arbetet har utförts i Chalmers Prototyplabb eller anslutande lokaler. Foton från tillverkningen finns i Appendix B.

4.1.1 Farkostens utsida

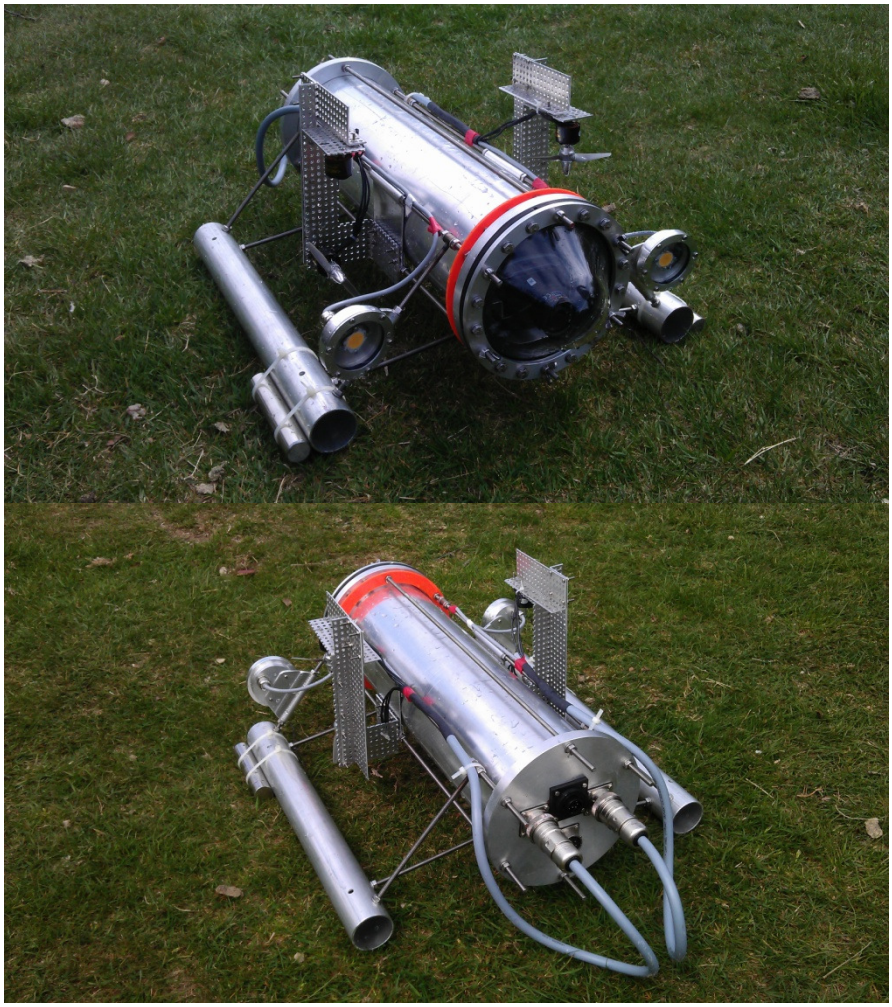
Som skrov valdes ett 500 mm långt aluminiumrör. Fram- och bakstycke svarvades ut ur ett massivt aluminiumstycke. I framstycket fästes en fem mm tjock, genomskinlig plastkupol med en aluminiumring och en gummipackning. Genom bakstycket går fyra kabelgenomförningar, en för kommunikation med ytstationen, två för motorer och lampor samt en som agerar startnyckel alternativt laddare. För att täta mot röret både fram och bak sitter två o-ringar, en mot rörets insida och en mot röränden. Startnyckeln är en vanlig kontakt med två ihopkopplade stift, se

Mellan framstycket och bakstycket går sex stycken M6-gängstänger. På dessa sitter motorfästen och medar som hålls på plats av muttrar. Alla gängstänger, skruvar, muttrar och brickor som sitter på ROV:ens utsida är syrafasta enligt klassificering A4. För att stabilisera och hålla ihop gängstängerna när ROV:en demonteras sitter dessutom en plastring precis bakom framstycket.

Medarna är två aluminiumrör som sitter på höger respektive vänster sida under ROV:en. Från varje fot går fyra gängstänger med en hylsa ytterst. Dessa sitter i sin tur fast i gängstängerna som går längs farkostkroppen där de är fästa med muttrar.

Mitt på gängstängerna på vänster och höger sida av ROV:en sitter aluminiumplåtar på vilka motorfästena är fastskruvade. Dessa är i sin tur tillverkade i perforerad aluminiumplåt. I dessa plåtar sitter motorerna med propellrar fastskruvade, en nedåtriktad och en framåtriktad på varje sida av ROV:en. På bak- och ovansidan av motorfästena sitter ytterligare perforerade plåtar vinkelrätt mot de andra, som fenor. Dessa finns till för att stabilisera konstruktionen så att inte motorfästena böjer sig. Kabeln till motorerna sitter fast längs gängstången och går sedan vidare fram till lamphusen.

Lamphusen är svarvade ur aluminium med en genomskinlig plastskiva ytterst som hålls på plats av en aluminiumring. I lamphuset ligger dessutom ett spår med en o-ring för att hålla dessa täta. I varje lamphus sitter en 800 lumen stark LED. Lamphusen sitter med hjälp av en aluminiumplatta fast i en av gängstängerna som går till medarna. Foton av ROV:ens exteriör finns i Figur 4.1.



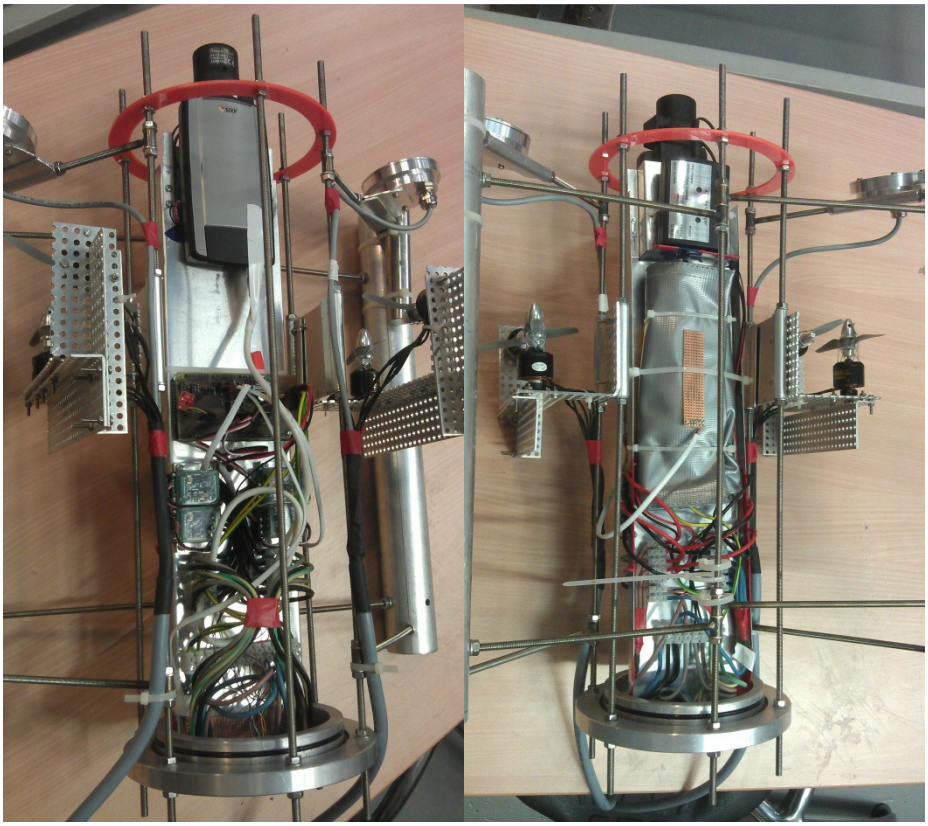
Figur 4.1 - ROV, färdig prototyp

Figur 4.2 - Startnyckeln som kopplas i en kontakt i bakstycket för att starta ROV:en

4.1.2 Farkostens insida

All elektronik i ROV:en sitter fast på en skiva av stålplåt som i sin tur är fästad i bakstycket. På denna skiva sitter kameran längst fram, därefter en huvudstyrenhet, fyra motorstyrenheter och längst bak kopplingsplintar från vilka kablar går till bakstycket och ut. ESC står för Electronic Speed Control och de styr således motorerna.

På undersidan av skivan sitter längst fram en batteriladdare. Efter denna sitter en nedböjd del av plåten för att hålla skivan horisontell. Bakom denna sitter batteriet och sedan sitter kopplingsplintar för strömförsörjningen.



Figur 4.3 – ROV:ens insida. Till vänster ovansidan av skivan med kamera, Arduino, ESC:er och kopplingsplintar. Till höger undersidan av skivan med laddare, batteri i skyddspåse och kopplingsplintar för strömförsörjning.

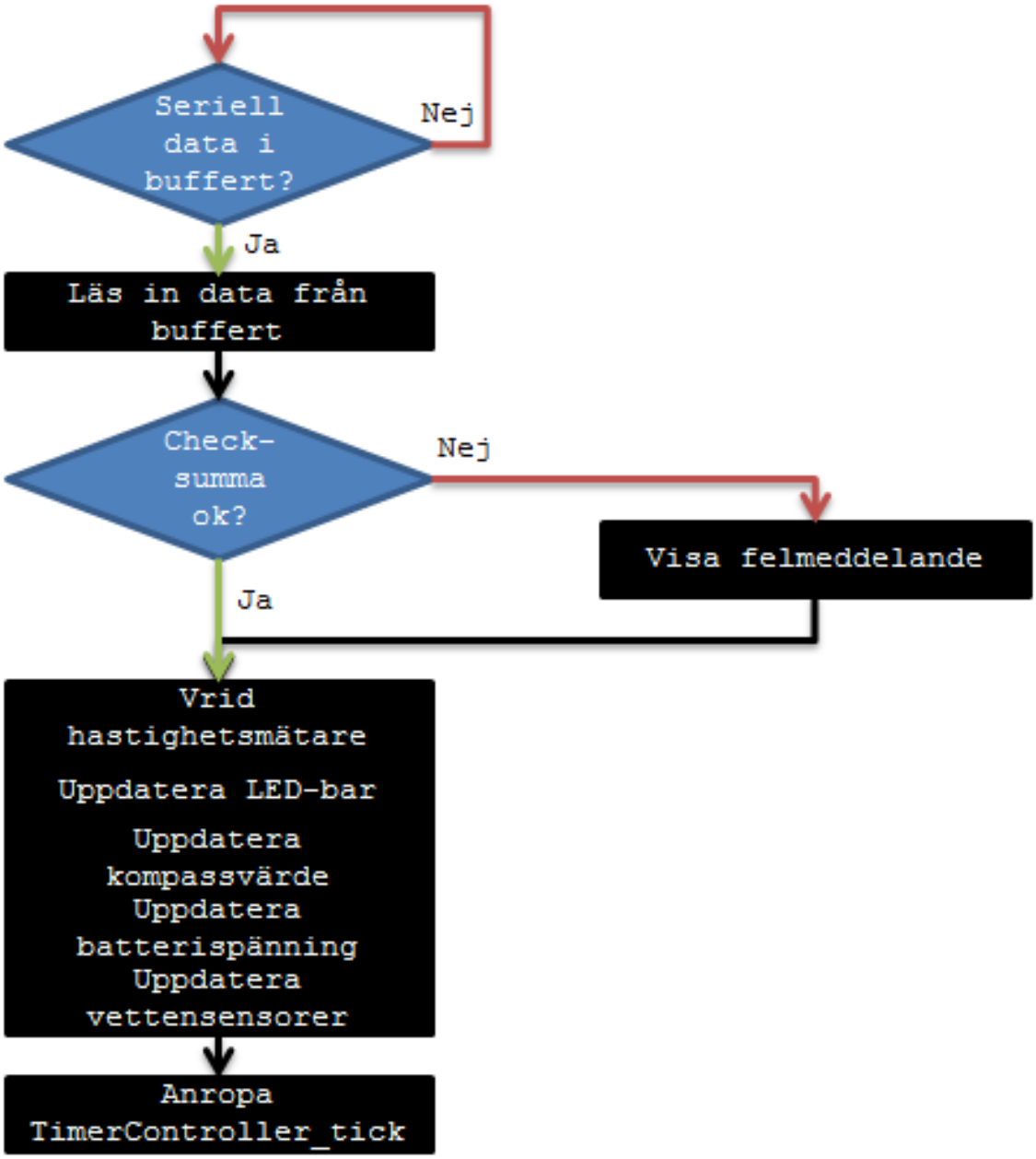
4.1.3 Ytenhet

Farkosten styrs med en Xbox 360-kontroll som är kopplad till en bärbar dator via USB. I datorn finns ett interface som visar bild från kameran och signaler från utrustningen i farkosten. Den skickar även styrsignaler till en transreciever som omvandlar USB-signalen till RS485 som skickas vidare till ROV:en över en cirka 40 m lång nätverkskabel. Ytenheten fraktas och förvaras i en skyddsväska. I denna väska förvaras även startnyckeln till ROV:en.

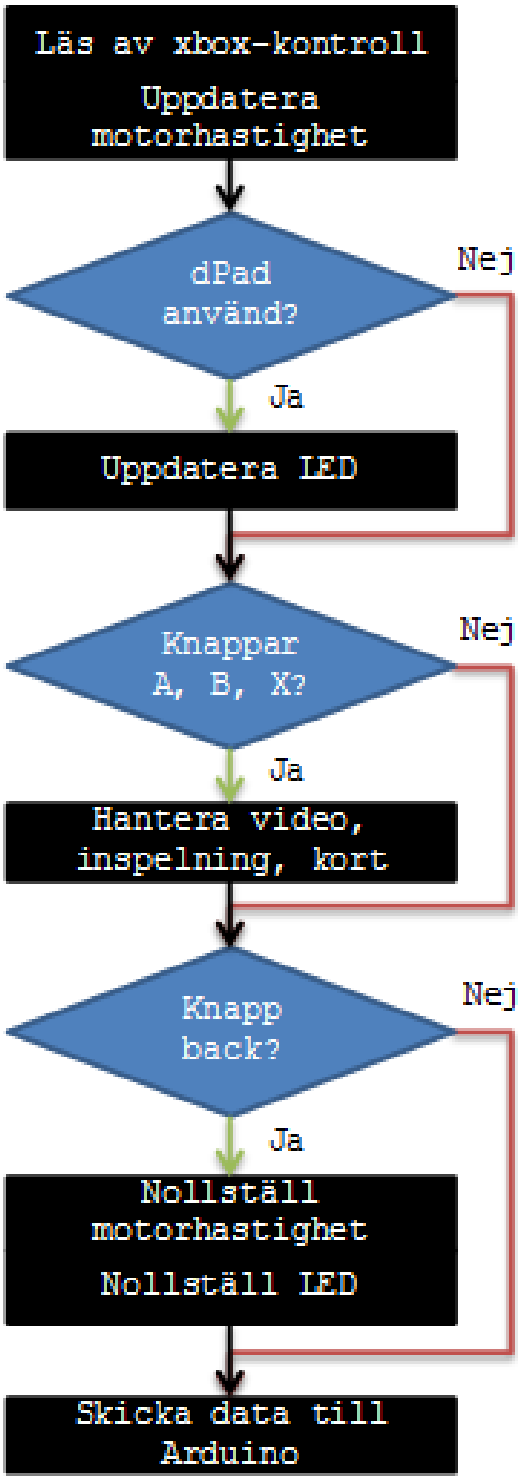
Då ytenheten innehåller en dator blev valet av språk svårare eftersom i stort sett vilket språk som helst hade kunnat lösa uppgiften. Dock visade det sig att kameratillverkaren har färdiga bibliotek för kameran till utvecklingsspråket C#. Detta i kombination med de redan existerande biblioteken för Xbox 360-kontrollen gjorde att valet föll på C#. Utvecklingsmiljön för datorn var Microsoft Visual Studio.

Figur 4.4 - Ytenheten med ROV-programmet igång. transrecievern ligger bakom datorskärmen och bredvid på bänken ligger navelsträngen

Då programmet till stor del bygger på grafiska element och användandet av timrar för att synkronisera olika delar av programmet sker ingen fullständig genomgång av programkoden i denna rapport. Nedan följer dock förenklade systemskisser av de två viktigaste looparna i programmet. För den intresserade återfinns den fullständiga programkoden i sin helhet i Appendix H – Datorkod.

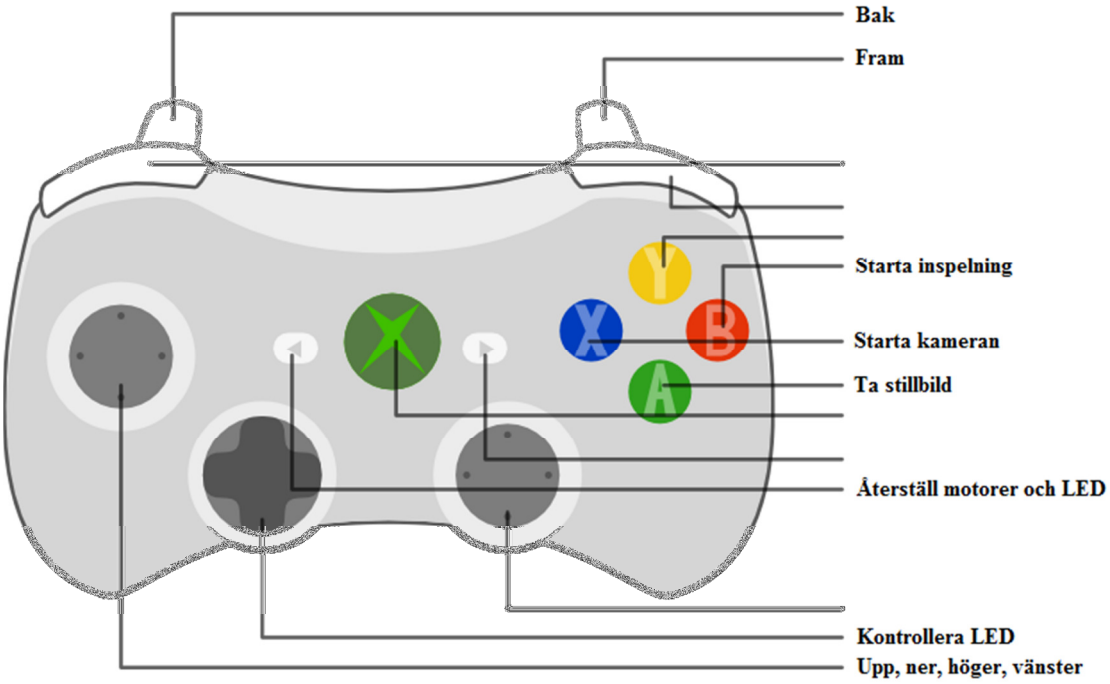


Figur 4.5 - Förenklad bild av timerRx_tick, mottagandet av data från styrenheten



Figur 4.6 - Förenklad bild av timerController_tick, hantering av kontroll samt skickande av data till styrenheten

Styrningen av ROV:en sker med en Xbox 360-kontroll enligt Figur 4.7 - Kontrollenheten till ROV:en med beskrivning av knapparnas funktion. Då kontrollen har fler knappar och spakar än vad som krävs för kontrollering av ROV:en finns det stora möjligheter till ytterligare påbyggnationer av systemet.



Figur 4.7 - Kontrollenheten till ROV:en med beskrivning av knapparnas funktion (Freepik, 2010)

Protokollet som används mellan ytenheten och styrenheten bygger på att skicka data seriellt med hjälp av paket bestående av nio byte. Samtliga delar av programmet är skrivet så att protokollet är utökningsbart till att innehålla fler bitar vid en eventuell utökning av ROV:ens funktioner.

Tabell 4.1 - Protokollet som används för att skicka data mellan ytenhet och styrenhet

Byte	Data	Omfång	Kommentar
0	Motor 1 (vänster)	10 – 170	90 = neutral
1	Motor 2 (höger)	10 – 170	90 = neutral
2	Motor 3 (upp)	10 – 170	90 = neutral
3	Motor 4 (upp)	10 – 170	90 = neutral
4	LED	0 – 255	Mjukvarubegränsad max 200 pga. nuvarande LED
5	Kompass	0 – 180	2 grader/bit
6	Vattensensorer	0 – 100	0 = torr
7	Batterispänning	0 – 245	0V – 24.5V
8	Checksumma	0 – 255	sum(byte1-7)%255

4.1.4 Elsystem

Motorer och motorstyrning

ROV:en styrs och drivs av fyra borstlösa motorer som sitter placerade på motorfästen ute i vattnet. Varje motor drivs av tre kablar som kopplas till en ESC, Electronic Speed Controller. Varje motor har en egen ESC och de styrs av servosignaler som genereras av ROV:ens styrenhet. Kablarna från motorerna förs in i ROV:en via vattentäta kontaktstycken.

Lampor

Längst fram på ROV:en, på vardera sida om kupolen, sitter två 800 lumen starka LED i vattentäta hus. Varje LED har sina poler kopplade till styrenheten via kontaktstyckena. Styrenheten jordar LED:arna med varsin mosfet, som i sin tur styrs med pulsbreddsmodulerade signaler. Genom att ändra pulsbredden så kan ljusstyrkan på lamporna styras.

Kopplingscentral

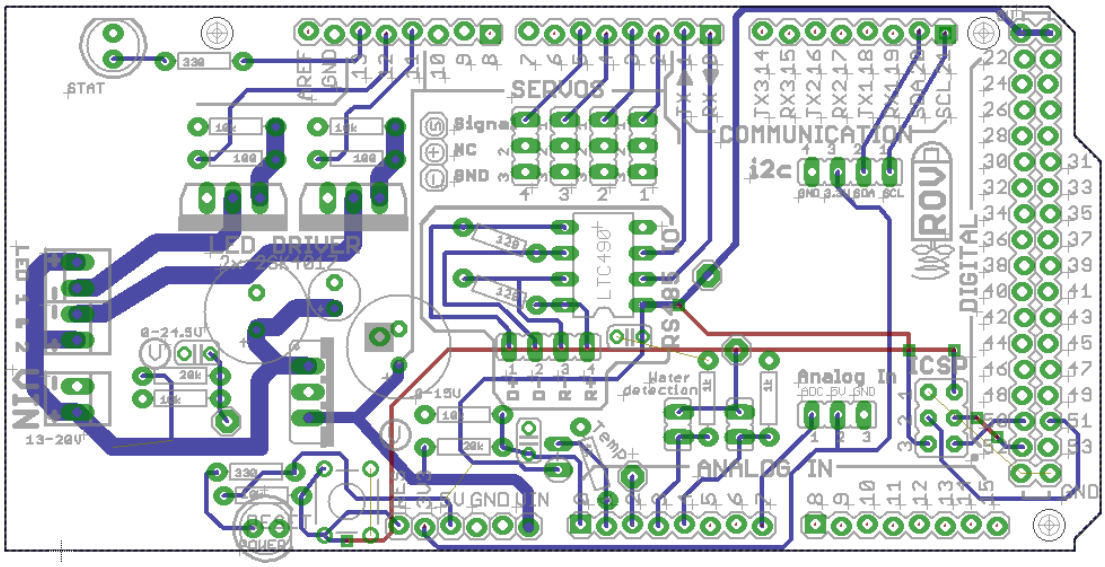
På insidan av ROV:en sitter fyra rader med kopplingsplintar där samtliga kablar från kontaktstyckena kopplas om mot de inre enheterna. Detta sätt att skarva kablarna gör att det i framtiden är lätt att byta ut, flytta eller lägga till utrustning.

Styrenhet

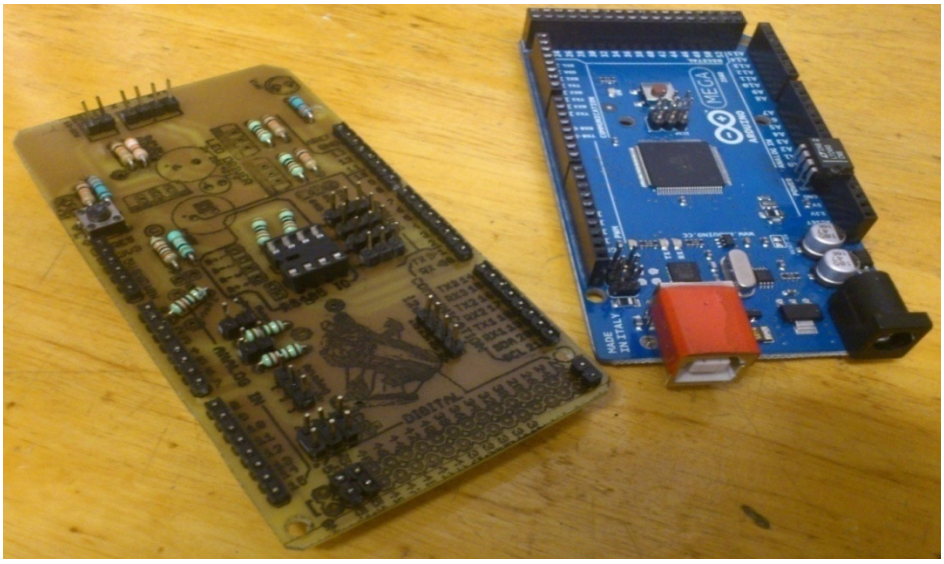
Styrenheten utgörs av en Arduino Mega 2560 och ett special designat påbyggnadskretskort, en sköld. På skölden sitter en spänningsregulator som reglerar ner batterispänningen från batteriet till 12V som leds till Vin på Arduinon. Den oreglerade batterispänningen leds till LED:arnas pluspoler och därmed måste MOSFET:arna som reglerar jorden styras så att de inte kan få för hög spänning.

Både den oreglerade och den reglerade batterispänningen delas och leds in till Arduinons analoga ingångar för att kunna mätas. Utöver spänningarna mäts också temperatur på kretskortet. Det sitter även två kontakter där vattensensorer kopplas på. De ger signal om vatten kopplar samman två ledare. Via I2C sitter en treaxlig magnetometer inkopplad. Den används för att beräkna kompassriktning. På kretskortet sitter även fyra servoportar som styr motorerna via ESC:erna. Arduinon har sin huvudseriellport kopplad till en RS485 transreciver som omvandlar den seriella kommunikationen till differentialsignaler.

Kretskorten till elektroniken har tillverkats genom att först skapa mönstret som önskas i CAD-programmet EAGLE. Mönstret har därefter skrivits ut och belysts med ultraviolett ljus ovanpå ett laminat som slutligen stoppas i en framkallningsvätska. Efter att kortet etsats, som hela den beskrivna processen kallas, har hål borrats och stiftlister monterats för att underlätta vid montering av kortet.



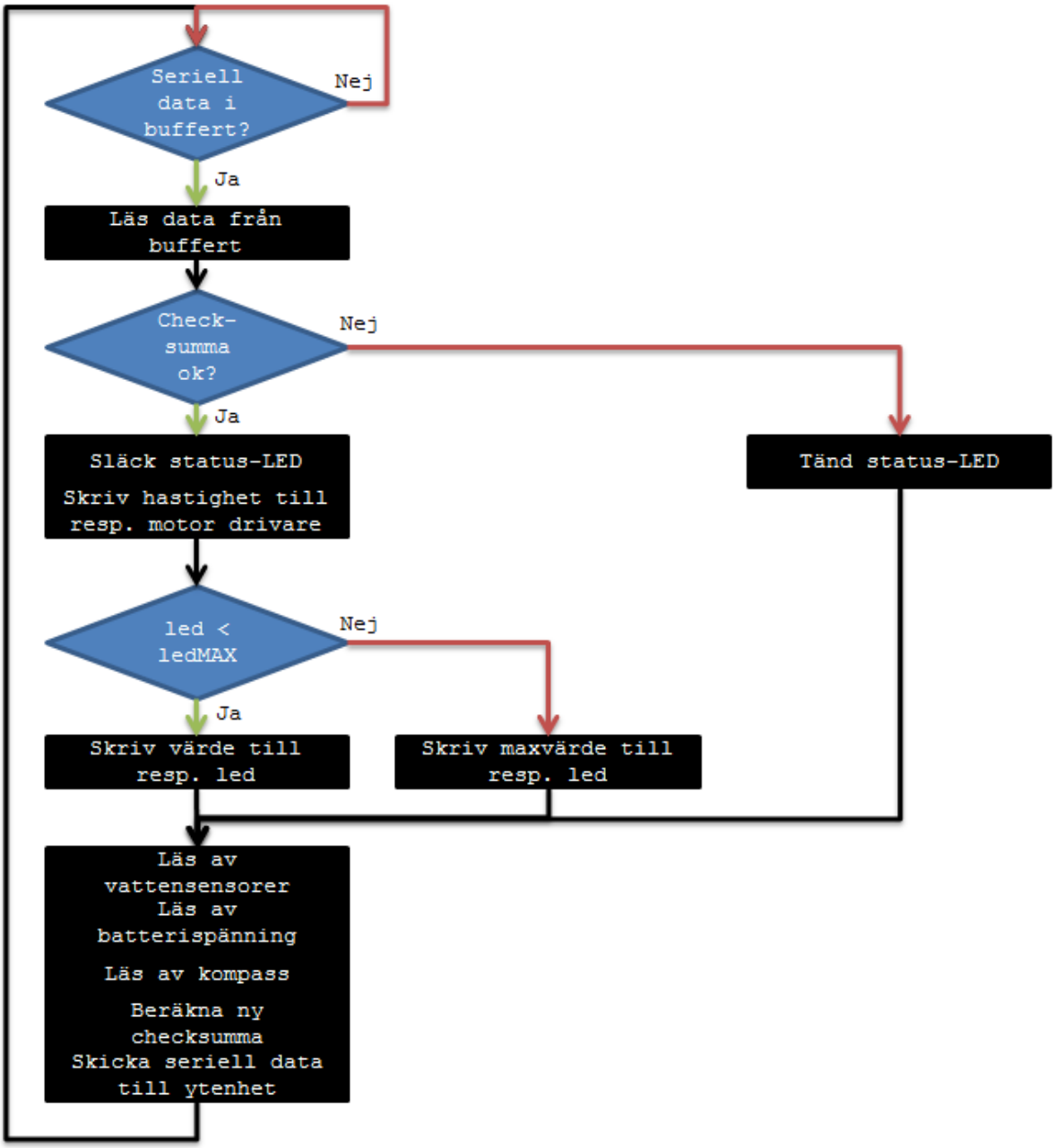
Figur 4.8 - Kretskortsdesign för skölden i programmet Eagle CAD



Figur 4.9 - Styrenheten, skölden till vänster och Arduino Mega 2560 till höger

Program till styrenhet

Då styrenheten utgörs av en Arduino Mega 2560 är programmet till den skrivit i Arduinos programvara i programmeringsspråket C. Överst i programmet definieras alla variabler och konstanter. Därefter följer initieringsfasen där kommunikationen startas. Sedan går programmet in i en evighets-loop som utför huvudprogrammet som väntar på att det skall komma kommandon från ytenheten, via seriell kommunikation. Kommandon skickas med paket och innehåller data i en förbestämd ordning enligt Tabell 4.1 - Protokollet som används för att skicka data mellan ytenhet och styrenhet. En förenklad systemskiss över hur programmet fungerar finns nedan i Figur 4.10 - Förenklad bild av programmet i styrenheten. Programmet i sin helhet återfinns i Appendix G – Arduinokod.



Figur 4.10 - Förenklad bild av programmet i styrenheten

4.2 Uppfyllande av kravspecifikation

I Tabell 4.2 och Tabell 4.3framgår hur väl ROV:en uppfyller de krav och önskemål som framställdes i början av projektet.

Tabell 4.2 - Uppfyllande av krav från kravspecifikation

Krav	Kriterier	Enhet	Antal	Uppmätt resultat	Uppfyllt
A	Minsta djup i saltvatten	m	10	Dykt 10 m i kanalen	Ja
B	Visa realtidsvideo på ytan	St	1	Videoöverföring fungerar	Ja
C	Maximal vikt	Kg	20	Cirka 12 kg	Ja
D	Minsta aktionsradie	m	20	Cirka 40 m	Ja
E	Lägsta operationstid vid maxfart framåt på 10m djup	min	10	ROV:en går att köra i 10 minuter under vatten	Ja
F	Minsta maxfart under vatten	m/s	0,5	Uppmätt 0,8 m/s	Ja
G	Farkosten skall vara tät	-	-	Farkosten är tät	Ja
H	Farkosten skall kunna manövreras av en person	-	-	Farkosten kan styras av en person	Ja
I	Information från farkosten skall visas i styrenheten	-	-	Information visas i styrenheten	Ja
J	Dragsäker kommunikationslänk	-	-	Lyckats dra loss ROV:en efter att den fastnat i botten	Ja

Tabell 4.3 - Uppfyllande av önskemål från kravspecifikation

Önskemål	Rang	Kriterier	Enhet	Antal	Uppmätt resultat	Uppfyllt
a	5	Minsta dykdjup	m	20	Ej testat	Nej
b	7	Maximal vikt	kg	10	Cirka 12 kg	Nej
c	4	Fungera upprepade gånger i saltvatten	-	-	Ej testat	Nej
d	1	Hållbar konstruktion	-	-	Konstruktionen är hållbar och robust	Ja
e	11	Minsta aktionsradie	m	100	Cirka 40 m	Nej
f	8	Minsta maxfart	m/s	1	Uppmätt 0,8 m/s, dock ej testat med fulladdat batteri	Nej
g	10	Tid att frilägga elektroniken på farkosten för en van användare	min	30	<5 min	Ja
h	2	Farkosten bör ha belysning	Lumen	>400	2 x 800 lumen LED	Ja
i	12	Farkosten bör vara genomskinlig			Farkosten är ej genomskinlig	Nej

J	3	Kostnad för farkosten bör ej överstiga	Kr	10000	< 10 000 kr	Ja
K	6	Tillverkad av standardkomponenter	-	-	I stor utsträckning tillverkad av standardkomponenter	Ja

5 Diskussion

Diskussionen ger en subjektiv bild av projektets gång och de resultat som uppnåtts.

5.1 Diskussion av resultat

Projektet

Planeringen i början av projektet var lyckad, framför allt då tidsplanen hölls. Tidigt delades ROV-systemet upp i olika delar som lösningar söktes till. Efter mittmötet påbörjades byggandet av prototypen och märktes att planeringen och lösningssökandet var väl genomfört eftersom byggfasen gick relativt problemfritt. En sak som saknades i byggfasen var dock att fler tester borde ha genomförts i förväg. Till exempel skulle en tryckkammare varit bra att konstruera för att testa kontaktstycken och tätningar. Under byggfasen erhölls stor hjälp av prototyplabbets anställda som gav tips och feedback på konstruktionsritningarna samt förverkligandet av dessa.

Testkörningen av prototypen var lyckad och det första testet som gjordes så fungerade allt som tänkt utan krångel, vilket var långt över förväntan.

Uppfyllande av kravspecifikation

Projektet ledde till en färdig, fungerande prototyp där alla krav i kravspecifikationen uppfylldes, se rubrik 4.2.

Vissa önskemål är dock ej uppfyllda. Detta gäller maxvikten, minsta aktionsradien och att farkosten bör vara genomskinlig. Ett önskemål var att vikten borde vara mindre än tio kg men i nuläget väger den ca tolv kg. Det har dock underlättat konstruktionen att inte behöva räkna allt för mycket på vikten och därmed uppfylldes inte detta önskemål. Vad gäller aktionsradien, som önskades vara 100 m begränsades detta av att den kabel som erhölls i projektet enbart var 40 m lång. Det ansågs då onödigt att köpa en längre kabel då det redan fanns en att använda och således uppfylldes ej det önskemålet. Det sista önskemålet som inte uppfyllts är att ROV:en borde vara genomskinlig, ur robusthetssynpunkt valdes dock detta bort.

Bland önskemålen finns även några punkter som ännu inte kunnat testas och dessa har således också satts som ouppfyllda. Anledningen till att några tester inte genomförts är då dessa varit lägre prioriterade än andra moment. ROV:ens maxfart under vatten har uppmätts till 0,8 m/s. Detta var dock inte med fulladdade batterier och med mycket stor sannolikhet kan den köra snabbare än en m/s under bra förutsättningar. Därför anses att detta inte testats ordentligt än. Ett annat otestat önskemål är att ROV:en bör kunna fungera upprepade gånger i saltvatten. Detta borde dock inte vara något problem alls så länge den sköljs av efteråt men det kan inte konstateras eftersom det inte testats. ROV:en har heller inte gått ned till 20 m eller djupare då det inte har funnits tillgång till båt eller plats med ett så stort djup nära land i Göteborgs närhet. Det finns däremot ingen anledning att tro att den inte skulle klara att gå ned till 20 m. Antagligen är den enda begränsande faktorn navelsträngens längd när det handlar om hur djupt den kan gå.

Prototyp

ROV:en fungerar mycket väl. Kameran ser mycket bättre i mörker än det mänskliga ögat och återger en bra och tydlig bild. Kamerans nackdel vid denna tillämpning var dock att autofokus är mycket begränsat och att det inte alls är anpassat för att kameran skall förflytta sig. ROV:en känns robust och mycket tät. Det till och med pyser när kontakterna dras ur efter att ha kört den i varmt vatten, exempelvis en pool. Motorfästena är stabila men inte särskilt estetiskt tilltalande, det saknas dessutom skydd mot propellrarna. Det går fort och lätt att frilägga elektroniken och sedan montera ihop ROV:en igen.

Kondens visade sig inte bli något problem alls då inte ens de mycket känsliga vattensensorerna givit utslag och då det dessutom är mycket små mängder rent vatten det handlar om. Navelsträngen skapade inga problem i ROV:ens framdrift genom att bromsa eller dra i den. Däremot sjönk navelsträngen och fastnade lätt.

ROV:en är väl avvägd och smidig att köra i vatten men det är svårt att navigera i sjöar eller hav där sikten är begränsad.

5.2 Förbättringsförslag

Drivsystemets nackdelar är att lagren i motorerna kommer korrodera sönder efter en tids användning och att motorerna är begränsade när det gäller rotationshastighet då motorerna ej går att ”smyga” med. Därför bör drivsystemet ses över och eventuellt uppgraderas till en mer robust lösning som ej korroderar sönder och som går att ”smygköra” med. Borstlösa elmotorer med sensorer löser ett av problemen men fortfarande så kommer dessa motorer att korrodera. Ett då bättre alternativ är att bygga täta kroppar till motorerna, denna lösning diskuterades tidigare i rapporten.

Det visade sig mycket svårt som förare att lokalisera sig då ROV:en framfördes i vattnet. Föraren behöver mer information i form av koordinater, djupmätning och liknande information. Kompassen i farkosten är ej tillförlitlig då ROV:en gör snabba manövrar, vilket skulle kunna förbättras. Dessutom borde navelsträngen göras flytande då den i nuläget lätt fastnar i diverse föremål på botten.

Ett system för att låta farkosten avgöra och motverka förflyttning i vattnet då ingen input från föraren ges kan vara önskvärt. Tillexempel accelerometrar och gyroskåp skulle kunna komplettera kompassmodulen för att avgöra hur farkosten förflyttar sig och då motverka detta.

I direkt soljus så är det svårt för föraren att se vad som visas på skärmen då den har för låg ljusstyrka. Detta skulle kunna lösas med en ljusstarkare skärm alternativt med avskärmning mot solljuset.

Designen av motorfästena är ej optimala, de är onödigt tunga och klumpiga. Om drivsystemets designas om så bör även motorfästena göra det. Dessutom borde någon form av skydd sättas runt propellrarna för att inte dessa skall skadas av skräp i vattnet, eller ännu värre ifall navelsträngen åker in i en propeller.

6 Slutsats

Projektets mål var att producera en fjärrmanövrerad undervattensfarkost, en så kallad ROV.

En prototyp av en ROV har konstruerats. Den är tät, klarar tio meters djupt och går att manövrera av en person från ytan som har tillgång till realtidsvideo från farkosten. Den är dessutom enkel att lyfta och ta med sig. Alla projektets krav är därmed uppfyllda, något som dock inte gäller för samtliga önskemål. Önskemålen om maxvikt, minsta aktionsradie och ett genomskinligt skrov är inte uppfyllda. Dessutom har inte önskemålen om minsta dykdjup och att fungera upprepade gånger i saltvatten kunnat testas och måste därför också anses som ej uppfyllda.

Projektet är därmed avklarat med mycket gott resultat.

7 Referenser

Bies, L. (1997). *RS-485 serial information*. Hämtat från www.lammertbies.nl: http://www.lammertbies.nl/comm/info/RS-485.html den 10 Maj 2012

Christ, R. (2007). *ROV Designs | The ROV manual: a user guide to observation-class remotely operated vehicles.*Amsterdam; Boston; London: Butterworth-Heinemann.

Freepik. (den 10 November 2010). *Xbox Controller Strange perspektiv ClipArt*. Hämtat från http://se.freepik.com: http://se.freepik.com/vektor-fritt/xbox-controller-strange-perspektiv-clipart_385544.htm den 16 Maj 2012

Jarvoll, J. (den 20 08 2009). *http://www.hobbytronik.se*. Hämtat från Hobbytronik: http://www.hobbytronik.se/blogg/index.php?m=08&y=09&entry=entry090820-223856 den 10 05 2012

Kinetics, U. (2012). *C8 eLED*. Hämtat från http://www.uwkinetics.com: http://www.uwkinetics.com/products/c8-eled den 16 Maj 2012

Matbase. (2009). *Aluminium 6060*. Hämtat från www.matbase.com: http://www.matbase.com/material/non-ferrous-metals/wrought-aluminium/almgsi0.5-6060/properties den 10 Maj 2012

Matbase. (2009). *PMMA*. Hämtat från www.matbase.com: http://www.matbase.com/material/polymers/commodity/pmma/properties den 10 Maj 2012

Matbase. (2009). *Stainless steel 304*. Hämtat från www.matbase.com: http://www.matbase.com/material/ferrous-metals/stainless-steel/x5crni-18-10-austenitic-304/properties den 10 Maj 2012

Moore, S. W., Bohm, H., & Jensen, V. (2010). *Underwater Robotics, Science, Design & Fabrication.* Monterey: Marine Advanced Technology Education Center.

Wikipedia. (den 12 03 2012). *1966 Palomares B-52 crash*. Hämtat från en.wikipedia.org: http://en.wikipedia.org/wiki/1966_Palomares_B-52_crash den 08 05 2012

Figurförteckning

Figur 2.1 - Schematisk bild av systemet ... 7
Figur 3.1 - Förslag till motorplacering ... 17
Figur 3.2 - Frihetsgrader för ROV ... 17
Figur 3.3 - Slutgiltigt koncept för farkostkroppen ... 20
Figur 4.1 - ROV, färdig prototyp ... 36
Figur 4.2 - Startnyckeln som kopplas i en kontakt i bakstycket för att starta ROV:en ... 36
Figur 4.3 – ROV:ens insida. Till vänster ovansidan av skivan med kamera, Arduino, ESC:er och kopplingsplintar. Till höger undersidan av skivan med laddare, batteri i skyddspåse och kopplingsplintar för strömförsörjning. ... 37
Figur 4.4 - Ytenheten med ROV-programmet igång. transrecievern ligger bakom datorskärmen och bredvid på bänken ligger navelsträngen ... 38
Figur 4.5 - Förenklad bild av timerRx_tick, mottagandet av data från styrenheten ... 39
Figur 4.6 - Förenklad bild av timerController_tick, hantering av kontroll samt skickande av data till styrenheten ... 40
Figur 4.7 - Kontrollenheten till ROV:en med beskrivning av knapparnas funktion (Freepik, 2010) ... 41
Figur 4.8 - Kretskortsdesign för skölden i programmet Eagle CAD ... 43
Figur 4.9 - Styrenheten, skölden till vänster och Arduino Mega 2560 till höger ... 43
Figur 4.10 - Förenklad bild av programmet i styrenheten ... 44
Figur A.1 – Ganttschema ... 54
Figur B.1 - Aluminiumröret för skrovet ... 55
Figur B.2 - Röret passas in för bakstycket ... 55
Figur B.3 - O-ringsspåren i bakstycket ... 55
Figur B.4 - Rörets kant ... 55
Figur B.5 - Hålen i bakstycket ... 55
Figur B.6 - Kupolen ... 55
Figur B.7 - Framstycket monterat i svarven ... 56
Figur B.8 - Skrovet monterad och stående ... 56
Figur B.9 - Inuti Skrovet ... 56
Figur B.10 - Bakstycket med hål för genomföringar ... 56
Figur B.11 - Medarna monterade ... 56
Figur B.12 - Lamphus ... 57
Figur B.13 - Färdigmonterat lamphus ... 57
Figur B.14– Motorfäste ... 57
Figur B.15 - Platta för att fästa motorfästena vid gängstängerna ... 57
Figur B.16 - Motorfästen med motorer samt lampfästen monterade ... 57
Figur B.17 - Motorfästen med motorer samt lampfästen monterade ... 58
Figur B.18 – ROV:en framifrån med lamphusen monterade ... 58
Figur B.19 - Montering av kabelgenomföringar i bakstycke ... 59
Figur B.20 - Tre kabelgenomföringar på plats i bakstycke ... 59
Figur B.21 - Alla fyra kabelgenomföringar på plats i bakstycket ... 59
Figur B.22 - De två typerna av kontakter, till vänster för navelsträngen och till höger för motorer, lampor, laddning och startnyckel ... 59

Figur B.23 - Två ESC:er på egentillverkad kylplatta i aluminium 60
Figur B.24 - Alla fyra ESC:er med kylplattor monterade på skivan med kopplingsplintar bakom 60
Figur B.25 - Tillverkning av kretskort till Arduinosköld 61
Figur B.26 - Tillverkning av kretskort till Arduinosköld 61
Figur B. 27 - Tillverkning av kretskort till Arduinosköld 62
Figur B.28 - Den egentillverkade skölden på plats på Arduinon 62
Figur B.29 - Kablarna till motorerna och lamporna, där kablarna går ihop till en är det tätat med epoxylim under krympslangen 62
Figur B.30 - Motorer och lampor monterade med kablar dragna 62
Figur B.31 - ESC:er, Arduinon och kopplingsplintar på plats på ovansidan av skivan i ROV:en. 62
Figur B.32 - Fler kablar dragna 62
Figur B.33 - Ring för att stabilisera gängstängerna då framstycket tas av 63
Figur B.34 - Ringen på plats på gängstängerna 63
Figur B.35 - Kameran på plats 63
Figur B.36 - ROV:en utan rör, sedd framifrån 63
Figur B.37- Motorfäste med vinkelräta, stabiliserande plåtar 63
Figur B.38 – 0,5 kg tunga vikter för att göra ROV:en neutral i vattnet, skall fästas på benen 63
Figur B.39 - Två vikter fastsatta på ett av benen 63
Figur B.40 - LiPo-batteriet som används i ROV:en. (foto taget från RC-flights hemsida) 64
Figur B.41 - LiPo-laddaren som sitter i ROV:en 64
Figur B.42 - Undersidan av skivan inne i ROV:en. Längst bak (vänster) sitter kopplingsplintar för strömförsörjning, i mitten sitter batteriet i en skyddspåse, på batteriet sitter även en läckagesensor och längst fram sitter laddaren till batteriet 64
Figur B.43 - Låda för transrecievern vid ytan tillverkas i 3D-skrivare 64
Figur B.44 - Lådan, här går kablar in från datorn 64
Figur B.45 - Lådan, här går navelsträngen ut till ROV:en 65
Figur B.46 - Väskan som innehåller ytenheterna. Där finns datorn, Xbox 360-kontroll, lådan med transrecievernn (ligger bakom datorskärmen på bilden) och startnyckeln (inte med på bilden) ... 65
Figur B.47 - Gränssnittet i datorn, till vänster hastighetsmätare, kompass, indikation på ljusstyrka och övrig sensordata 65
Figur B.48 - Färdig ROV 65
Figur E.1- Ritning över hela ROV:en (ej skalenlig) 68
Figur E.2 - Ritning över ringen utanpå kupolen (ej skalenlig) 68
Figur E.3 - Ritning framstycke (ej skalenlig) 69
Figur E.4 - Ritning bakstycke (ej skalenlig) 69
Figur E.5 - Ritning hålen i bakstycket (ej skalenlig) 70
Figur E.6 - Ritning lamphus, kropp (ej skalenlig) 70
Figur E.7 - Ritning lamphus, låsring (ej skalenlig) 71
Figur E.8 - Ritning kylblock för ESC (ej skalenlig) 71
Figur F.1 - Kopplingsschema över farkosten 72
Figur H.1 - Bild på programmet med förklarande texter till respektive objekt 79

Appendix A - Tidsplanering

Figur A.1 – Ganttschema

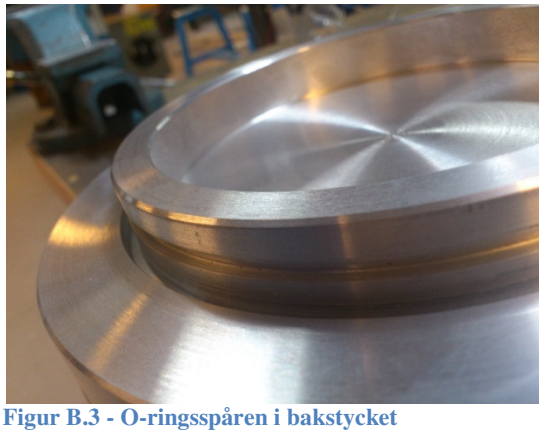
Tabell A.0.1 - Projektets milstolpar

Produkt	Deadline
Planeringsrapport	10-feb-2012
Sammanställning av lösningar till delproblem	28-feb-2012
Mittmöte	6-mar-2012
Färdig teoretisk produkt	15-mar-2012
Kompletta CAD-ritningar	15-mar-2012
Första vattentest av kropp	22-mar-2012
Färdiga fysiska lösningar	31-mar-2012
Färdig fysisk produkt	25-apr-2012
Slutrapport	30-apr-2012
Skriftlig opposition	16-maj-2012
Muntlig presentation	24-maj-2012

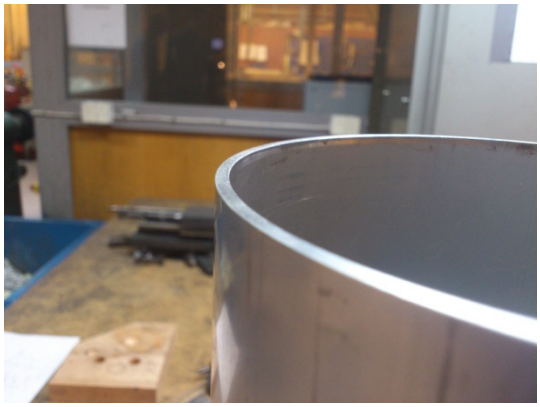
Appendix B – Foton från tillverkning

Figur B.1 - Aluminiumröret för skrovet

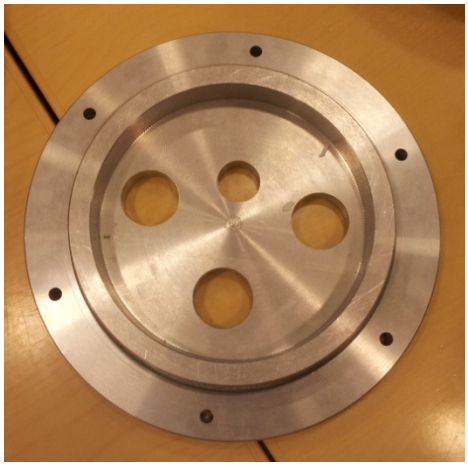
Figur B.2 - Röret passas in för bakstycket



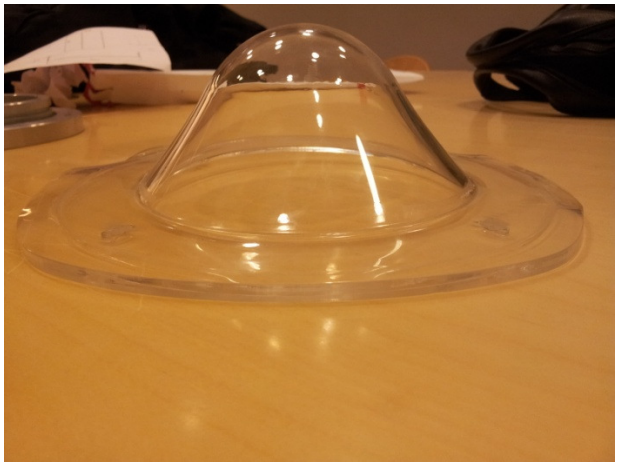
Figur B.3 - O-ringsspåren i bakstycket



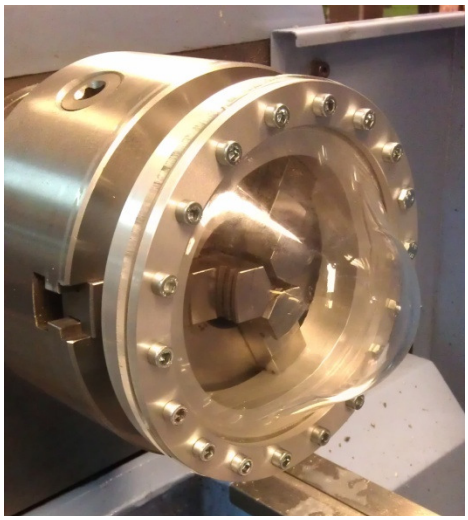
Figur B.4 - Rörets kant



Figur B.5 - Hålen i bakstycket

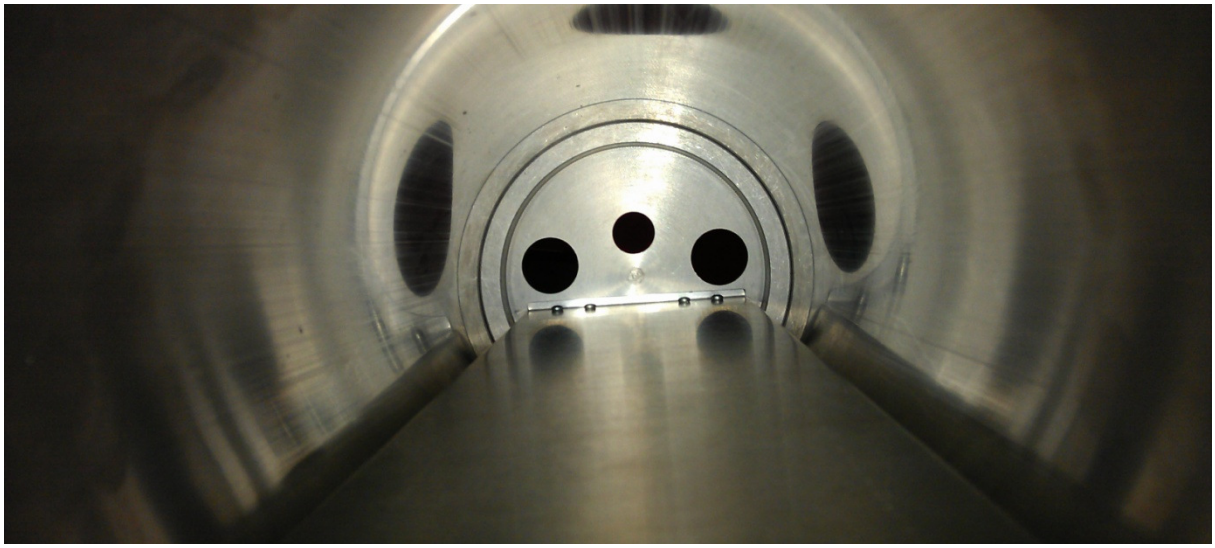


Figur B.6 - Kupolen

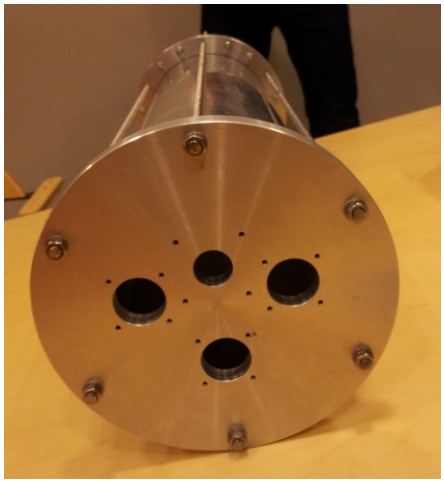


Figur B.7 - Framstycket monterat i svarven

Figur B.8 - Skrovet monterad och stående

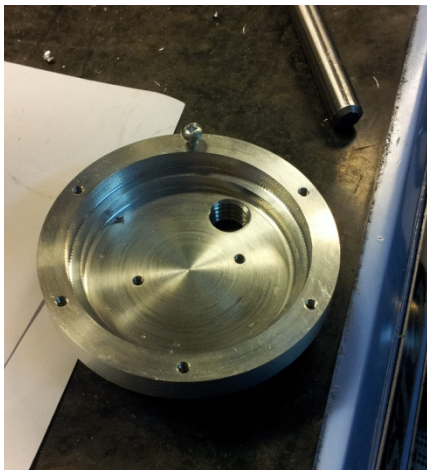


Figur B.9 - Inuti Skrovet

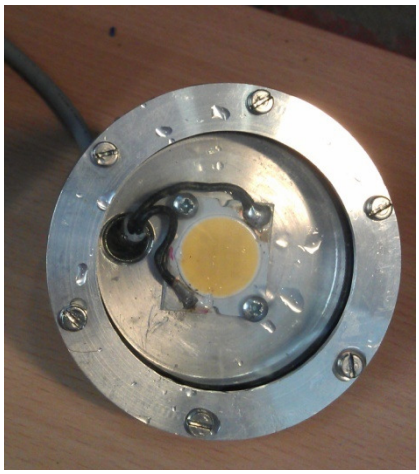


Figur B.10 - Bakstycket med hål för genomföringar

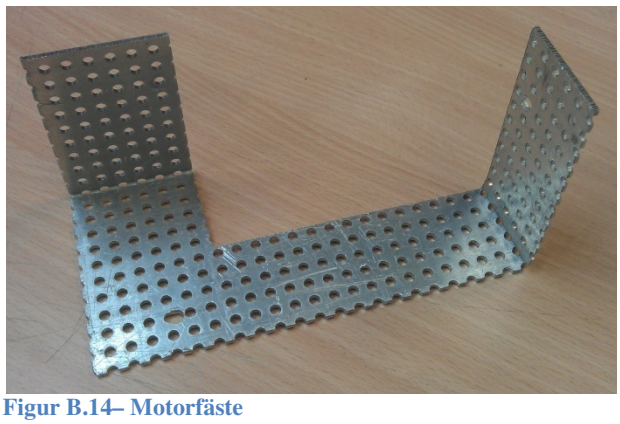
Figur B.11 - Medarna monterade



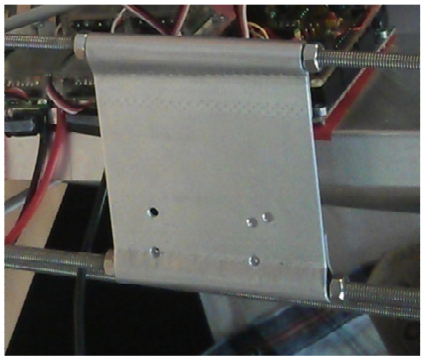
Figur B.12 - Lamphus



Figur B.13 - Färdigmonterat lamphus

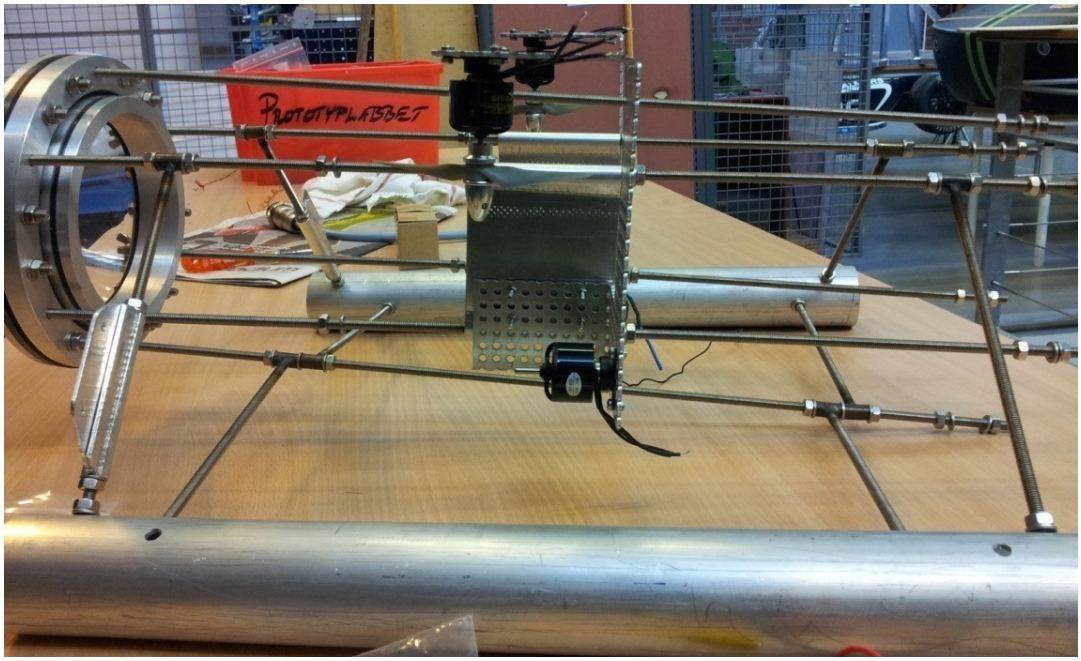


Figur B.14– Motorfäste

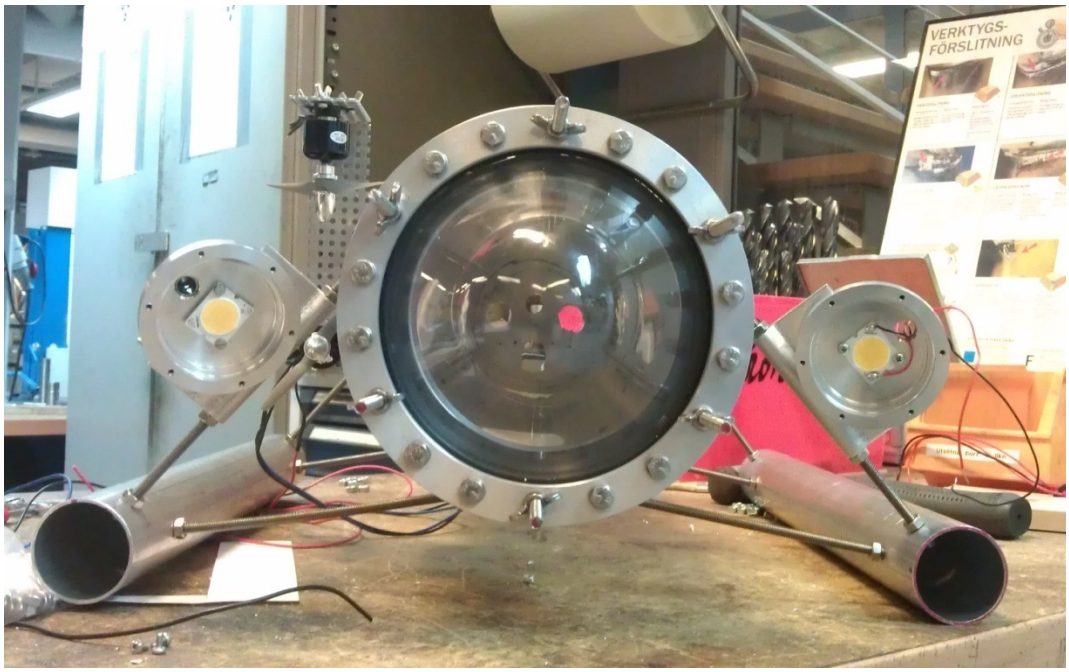


Figur B.15 - Platta för att fästa motorfästena vid gängstängerna

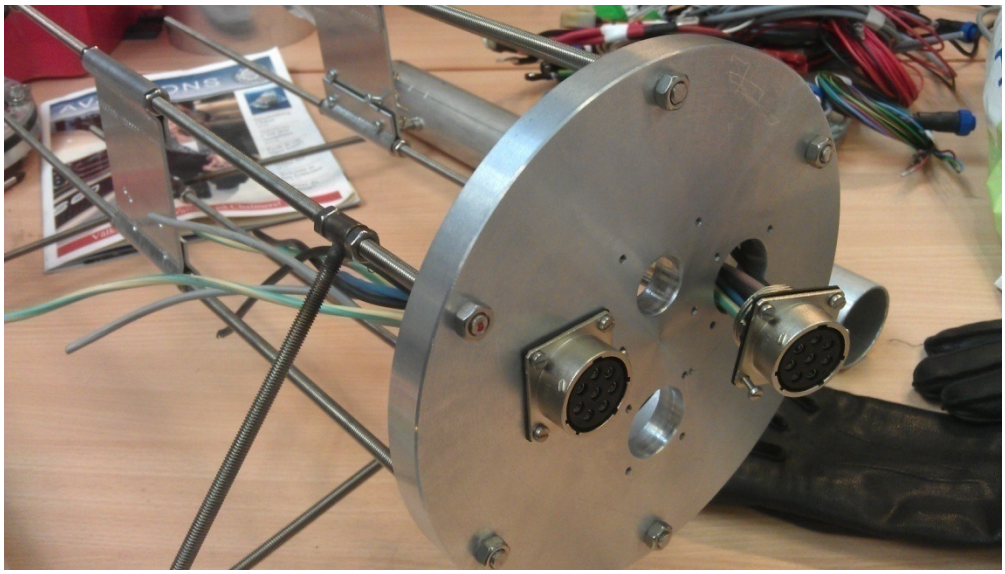
Figur B.16 - Motorfästen med motorer samt lampfästen monterade



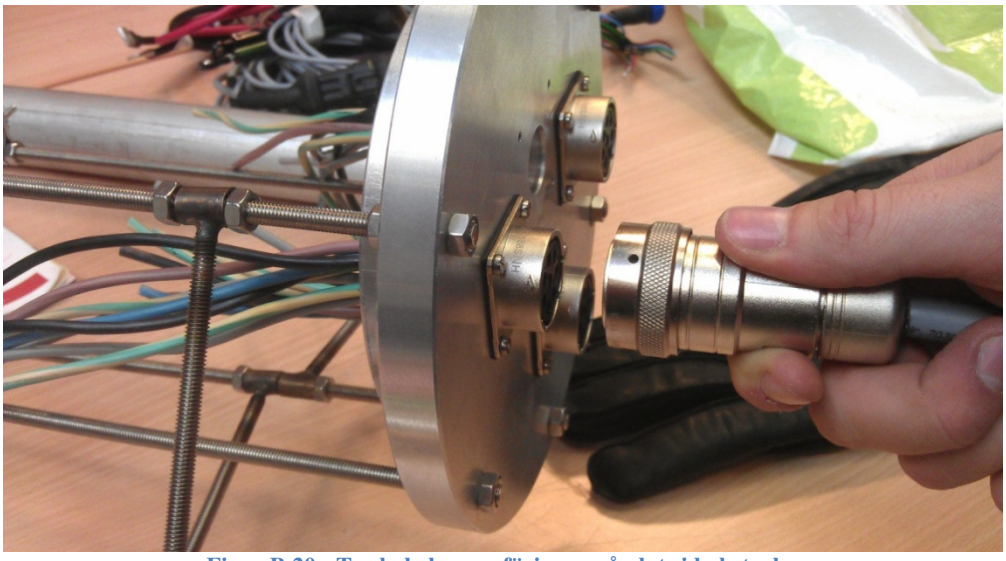
Figur B.17 - Motorfästen med motorer samt lampfästen monterade



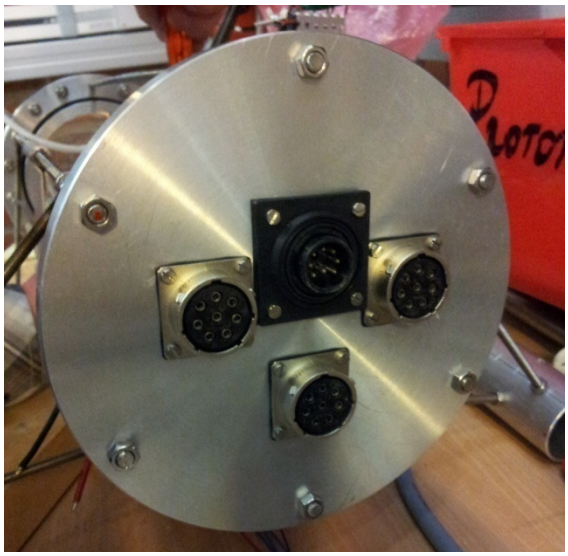
Figur B.18 – ROV:en framifrån med lamphusen monterade



Figur B.19 - Montering av kabelgenomföringar i bakstycke

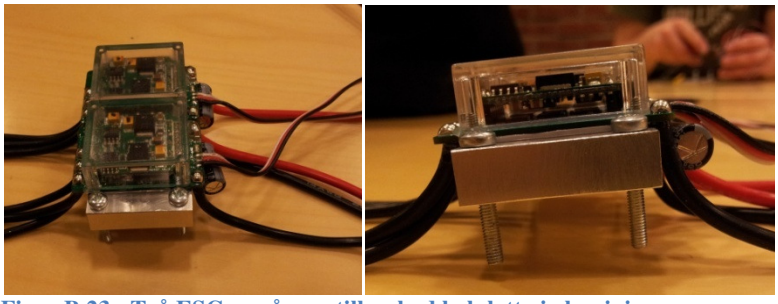


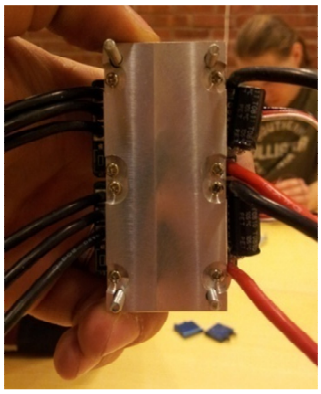
Figur B.20 - Tre kabelgenomföringar på plats i bakstycke



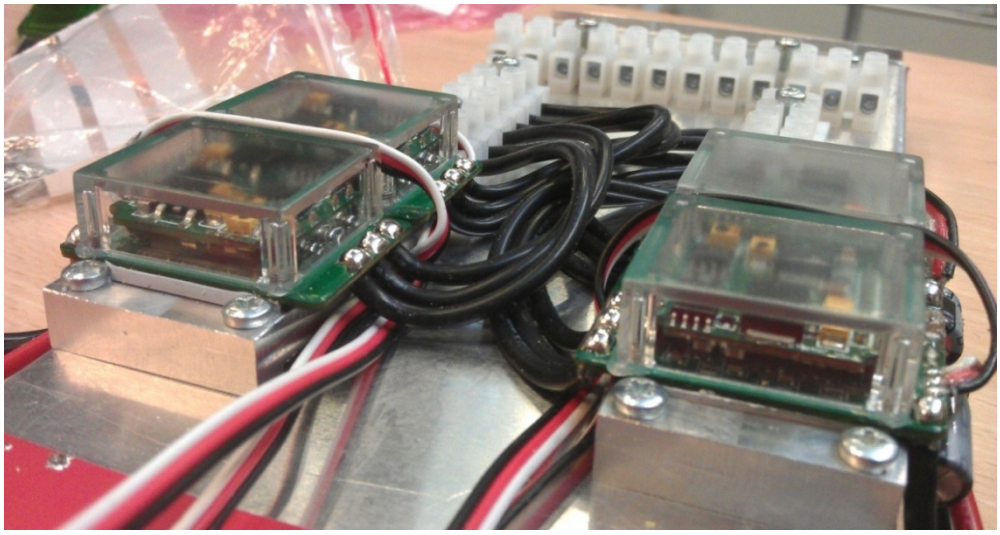
Figur B.21 - Alla fyra kabelgenomföringar på plats i bakstycket

Figur B.22 - De två typerna av kontakter, till vänster för navelsträngen och till höger för motorer, lampor, laddning och startnyckel

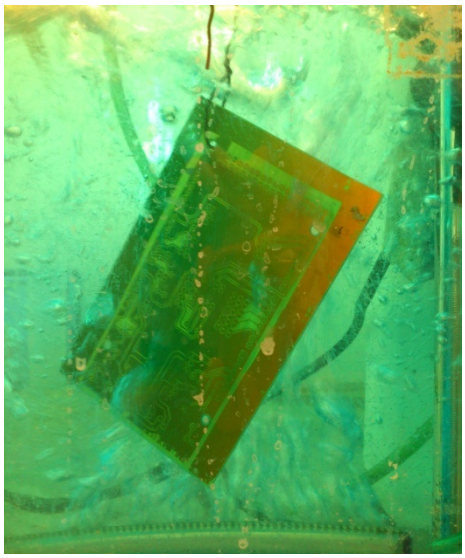




Figur B.23 - Två ESC:er på egentillverkad kylplatta i aluminium

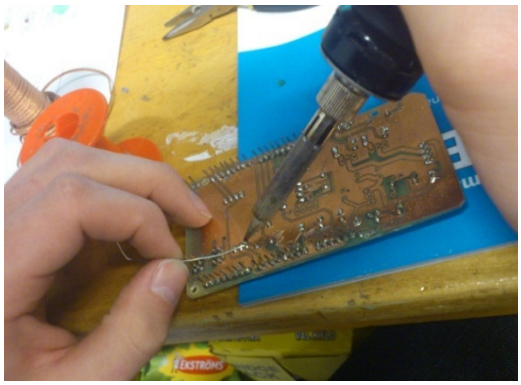


Figur B.24 - Alla fyra ESC:er med kylplattor monterade på skivan med kopplingsplintar bakom

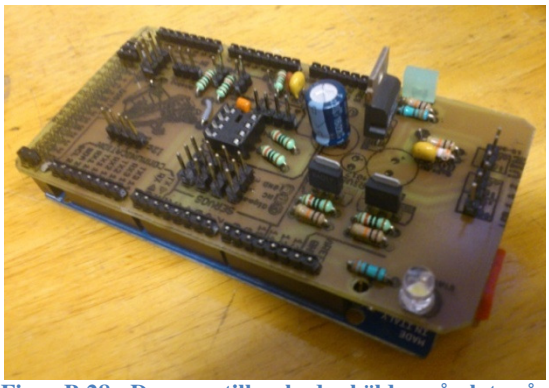


Figur B.25 - Tillverkning av kretskort till Arduinosköld

Figur B.26 - Tillverkning av kretskort till Arduinosköld

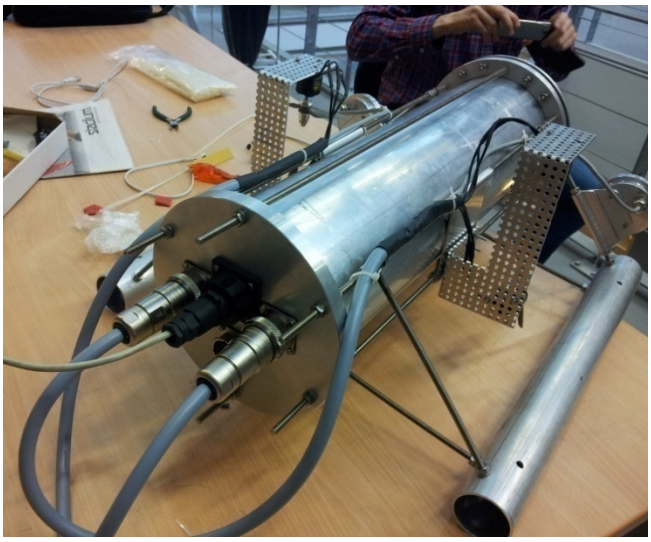


Figur B. 27 - Tillverkning av kretskort till Arduinosköld

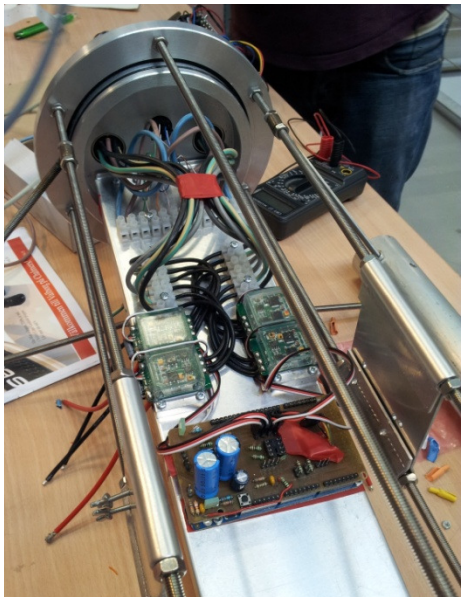


Figur B.28 - Den egentillverkade skölden på plats på Arduinon

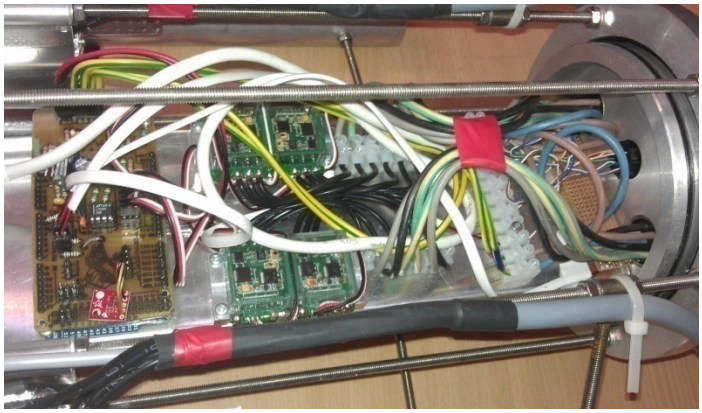
Figur B.29 - Kablarna till motorerna och lamporna, där kablarna går ihop till en är det tätat med epoxylim under krympslangen



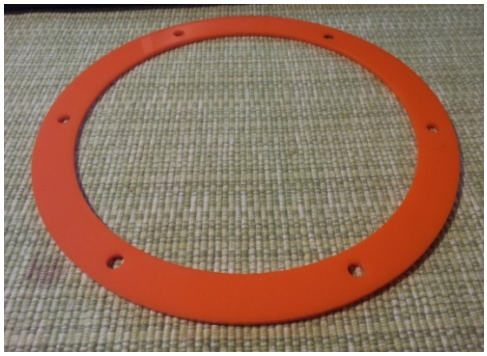
Figur B.30 - Motorer och lampor monterade med kablar dragna



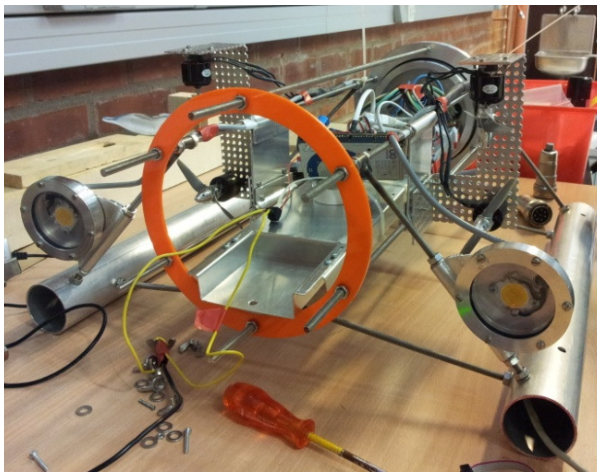
Figur B.31 - ESC:er, Arduinon och kopplingsplintar på plats på ovansidan av skivan i ROV:en



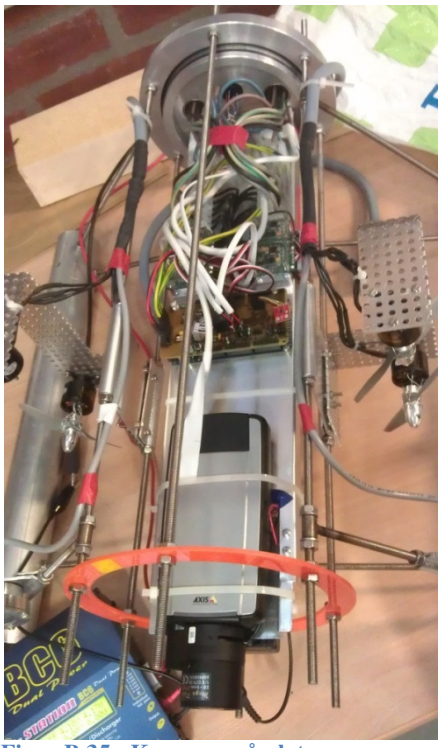
Figur B.32 - Fler kablar dragna



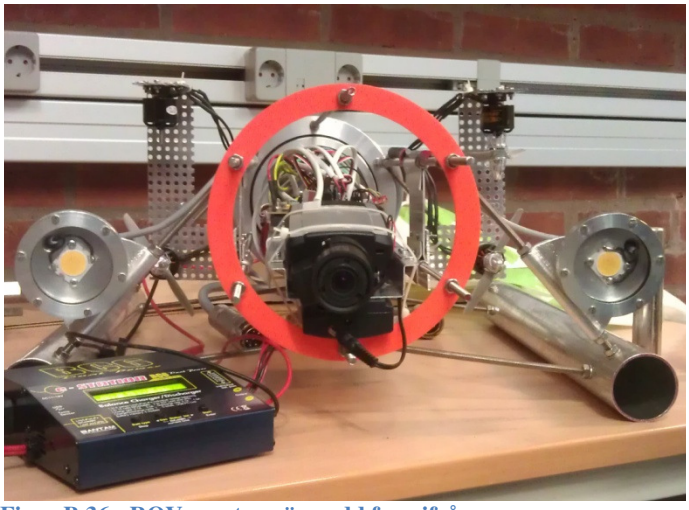
Figur B.33 - Ring för att stabilisera gängstängerna då framstycket tas av



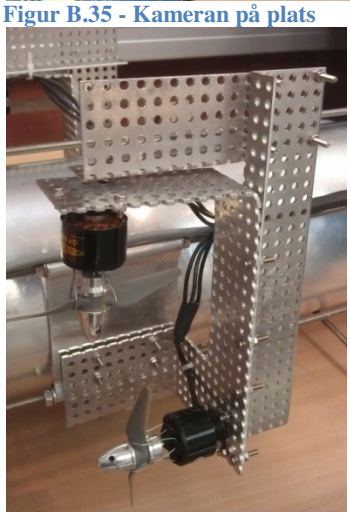
Figur B.34 - Ringen på plats på gängstängerna



Figur B.35 - Kameran på plats



Figur B.36 - ROV:en utan rör, sedd framifrån



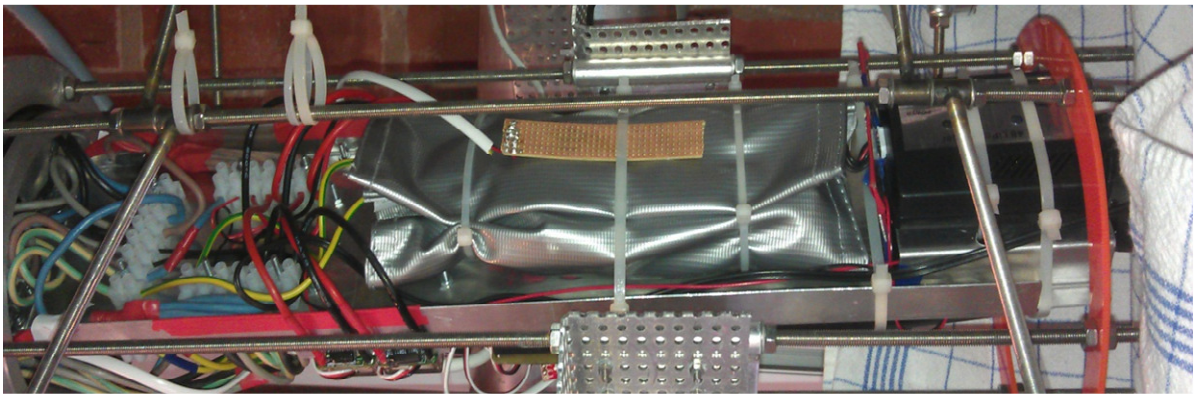
Figur B.37- Motorfäste med vinkelräta, stabiliserande plåtar

Figur B.38 – 0,5 kg tunga vikter för att göra ROV:en neutral i vattnet, skall fästas på benen

Figur B.39 - Två vikter fastsatta på ett av benen

Figur B.40 - LiPo-batteriet som används i ROV:en. (foto taget från RC-flights hemsida)

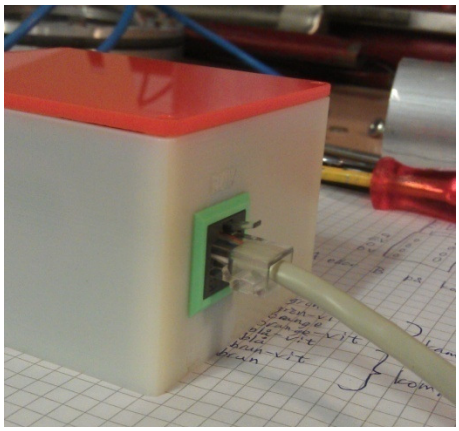
Figur B.41 - LiPo-laddaren som sitter i ROV:en



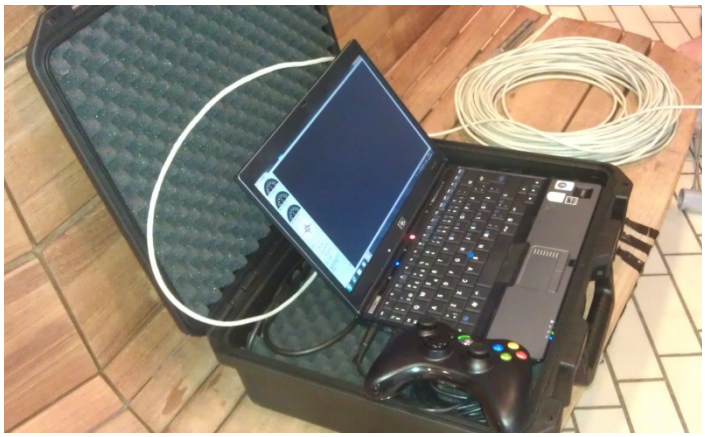
Figur B.42 - Undersidan av skivan inne i ROV:en. Längst bak (vänster) sitter kopplingsplintar för strömförsörjning, i mitten sitter batteriet i en skyddspåse, på batteriet sitter även en läckagesensor och längst fram sitter laddaren till batteriet

Figur B.43 - Låda för transrecievern vid ytan tillverkas i 3D-skrivare

Figur B.44 - Lådan, här går kablar in från datorn



Figur B.45 - Lådan, här går navelsträngen ut till ROV:en



Figur B.46 - Väskan som innehåller ytenheterna. Där finns datorn, Xbox 360-kontroll, lådan med transrecievernn (ligger bakom datorskärmen på bilden) och startnyckeln (inte med på bilden)

Figur B.47 - Gränssnittet i datorn, till vänster hastighetsmätare, kompass, indikation på ljusstyrka och övrig sensordata

Figur B.48 - Färdig ROV

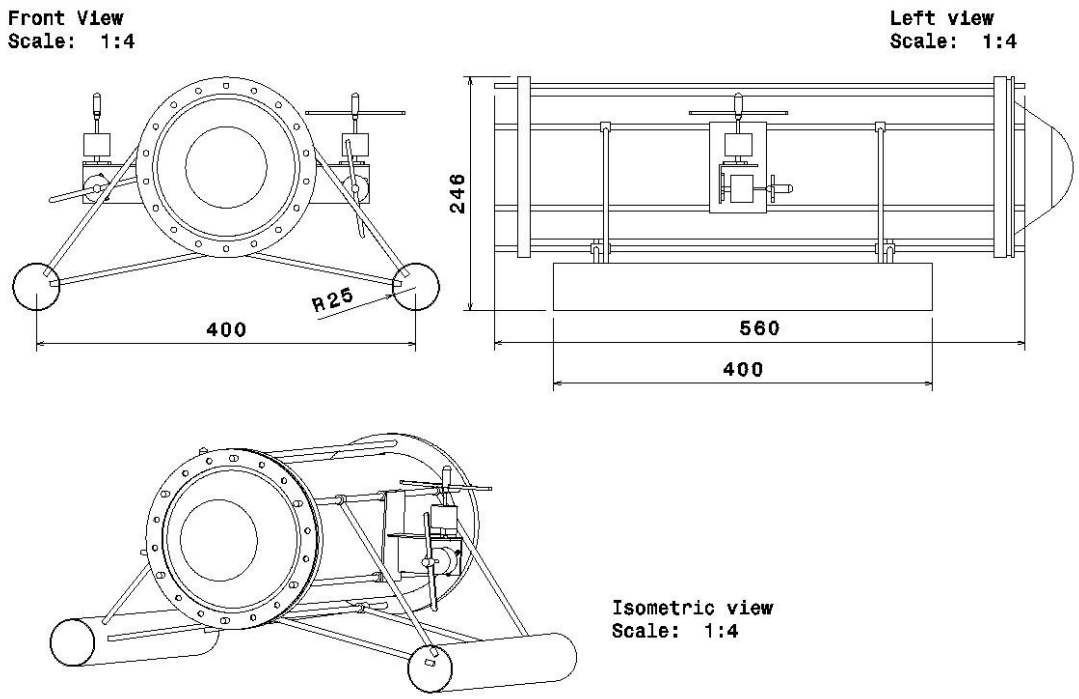
Appendix C - Energiförsörjning

Alternativ	48V, 300W	Förklaring	48V, 10W	Förklaring	Ingen överföring	Förklaring	Viktfaktor	Förklaring
Vikt	4	Tung kabel, märks inte på land.	3	Används två trådar i TP-kabeln kan kommunikation störas.	3	Ingen extra vikt.	0,6	Den måste vara lätthanterlig.
Pris	3	Krävs kablage och transformatorer.	4	Billigare och mindre transformatorer.	5	Ingen extra kostnad.	0,8	Begränsad budget.
Platsbesparing	2	Inget batteri i farkost.	2	Krävs batterier i farkost.	1	Alla energikällor i farkost.	0,6	En liten farkost är lättare att hantera.
Dragsäkerhet	3	Kabeln är relativt drakstark.	1	Ingen extra dragsäkring.	1	Ingen extra dragsäkring.	0,2	Inte jätteviktigt då det kan lösas på annat sätt.
Laddning	3	Problematik med laddning försvinner.	1	Måste ladda batterier i farkost.	1	Måste ladda batteri i farkost.	0,6	Fördel med batterier på land då de är lättare att ladda.
Komplexitet	3	Transformatorer kan bli svåra.	4	Krävs en liten transformator.	5	Måste bara vara stabil spänning.	1	Uppgiften är att konstruera enkelt.
Summa	11,4		11		12,2			
Rank	**2**		**3**		**1**			

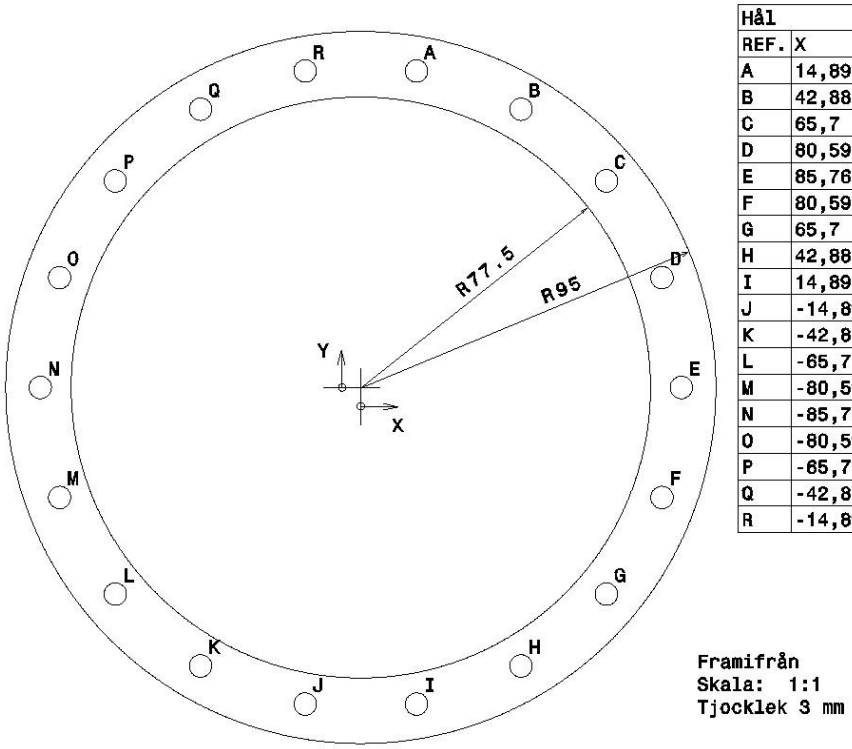
Appendix D - Batterialternativ

Egenskaper	Bly	Förklaring	Ni-Cd	Förklaring	Li-Po	Förklaring	Ni-Mh	Förklaring	Li-Ion	Förklaring	Vikt	Förklaring
Pris/Wh	5	0,26 Wh/sek	3	0,045 Wh/sek	4	0,056 Wh/sek	2	0,036 Wh/sek	1	0,033 Wh/sek	1	Pris är viktigt.
energi/Vikt	1	30-40 Wh/kg	2	40-60 Wh/kg	5	130-200 Wh/kg	3	60-120 Wh/kg	5	100-250 Wh/kg	0,8	Viktigt i farkosten
Energitäthet	1	60-75 Wh/l	2	50-150 Wh/l	5	300 Wh/l	3	140-300 Wh/l	5	250-620 Wh/l	0,6	Viktigt i farkosten.
Säkerhet	3	Gasbildning vid laddning över 2,4V/cell.	5	Stabila batterier.	1	Kan börja brinna vid felaktig hantering.	4	Överladdning är enda risken.	4	Överladdning får batteriet att brinna.	0,3	Relativt viktigt för att undvika skador.
Laddning	5	Laddas med +0,2V/cell, kan laddas i serie eller parallell.	4	Laddas med 0,25V/cell. Känslig för höga temperaturer.	2	Cellerna måste laddas individuellt.	3	Smart laddning bör användas.	3	Kräver speciell laddningsteknik och laddare krävs.	0,6	Laddning måste vara möjlig och enkel.
Tålighet	3	Tålig för yttre påverkan.	3	Tålig för yttre påverkan.	1	Känsligt för yttre påverkan.	3	Tålig för yttre påverkan.	3	Tålig för yttre påverkan.	0,2	Utsätts ej för större stötar.
Resultat	10,9		10,3		12,7		9,8		11,6			
Rank	**3**		**4**		**1**		**5**		**2**			

Appendix E – Ritningar

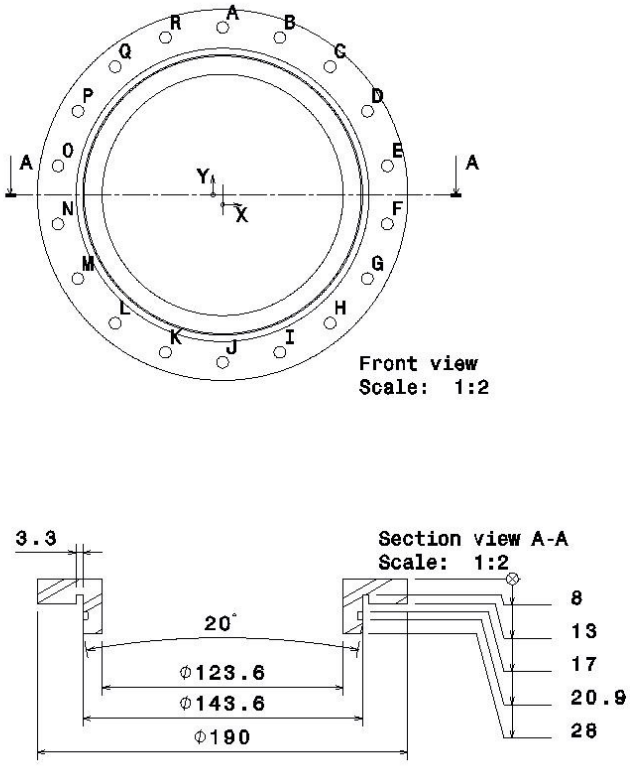


Figur E.1- Ritning över hela ROV:en (ej skalenlig)



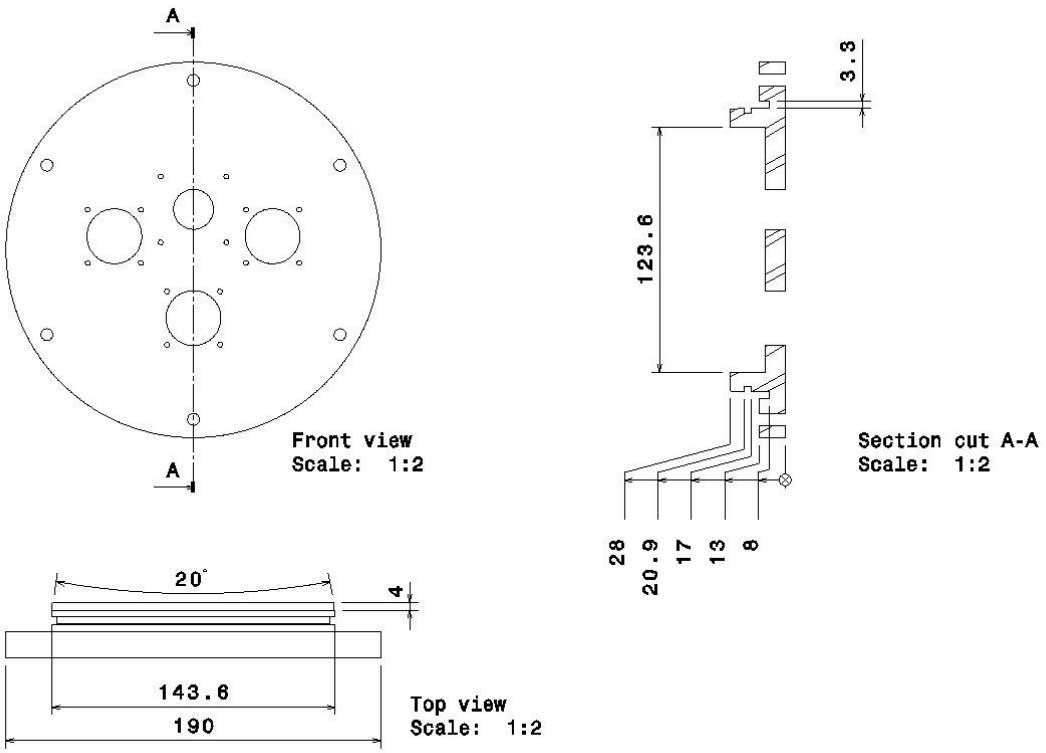
Hål			
REF.	X	Y	Diameter
A	14,89	84,46	6
B	42,88	74,27	6
C	65,7	55,13	6
D	80,59	29,33	6
E	85,76	0	6
F	80,59	-29,33	6
G	65,7	-55,13	6
H	42,88	-74,27	6
I	14,89	-84,46	6
J	-14,89	-84,46	6
K	-42,88	-74,27	6
L	-65,7	-55,13	6
M	-80,59	-29,33	6
N	-85,76	0	6
O	-80,59	29,33	6
P	-65,7	55,13	6
Q	-42,88	74,27	6
R	-14,89	84,46	6

Figur E.2 - Ritning över ringen utanpå kupolen (ej skalenlig)

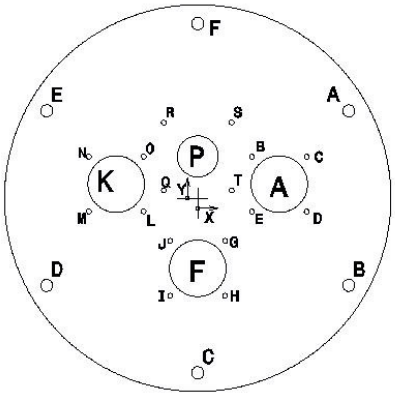


Hål			
REF.	X	Y	Diameter
A	0	85,76	6
B	29,33	80,59	6
C	55,13	65,7	6
D	74,27	42,88	6
E	84,46	14,89	6
F	84,46	-14,89	6
G	74,27	-42,88	6
H	55,13	-65,7	6
I	29,33	-80,59	6
J	0	-85,76	6
K	-29,33	-80,59	6
L	-55,13	-65,7	6
M	-74,27	-42,88	6
N	-84,46	-14,89	6
O	-84,46	14,89	6
P	-74,27	42,88	6
Q	-55,13	65,7	6
R	-29,33	80,59	6

Figur E.3 - Ritning framstycke (ej skalenlig)



Figur E.4 - Ritning bakstycke (ej skalenlig)

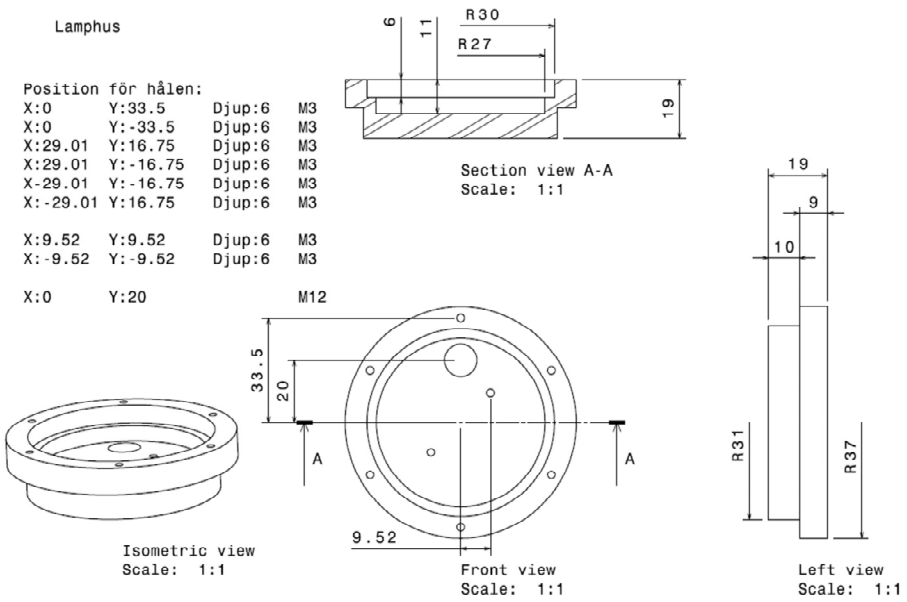


Front view
Scale: 1:2

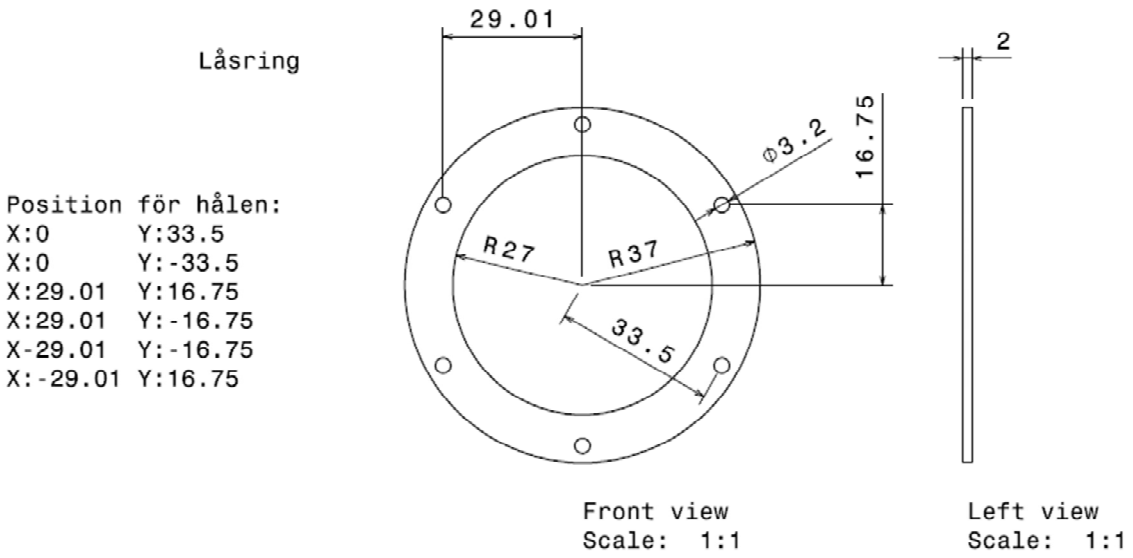
Innerhål			
REF.	X	Y	Diameter
A	40,1	6,89	27,8
B	26,63	20,37	2,46
C	53,58	20,37	2,46
D	53,58	-6,58	2,46
E	26,63	-6,58	2,46
F	0	-34,5	27,8
G	13,48	-21,02	2,46
H	13,48	-47,97	2,46
I	-13,47	-47,98	2,46
J	-13,47	-21,03	2,46
K	-40,1	6,89	27,8
L	-26,63	-6,58	2,46
M	-53,58	-6,58	2,46
N	-53,58	20,37	2,46
O	-26,63	20,37	2,46
P	0	20,5	20,2
Q	-16,65	3,85	2,46
R	-16,65	37,15	2,46
S	16,65	37,15	2,46
T	16,65	3,85	2,46

Ytterhål			
REF.	X	Y	Diameter
A	74,27	42,88	6
B	74,27	-42,88	6
C	0	-85,76	6
D	-74,27	-42,88	6
E	-74,27	42,88	6
F	0	85,76	6

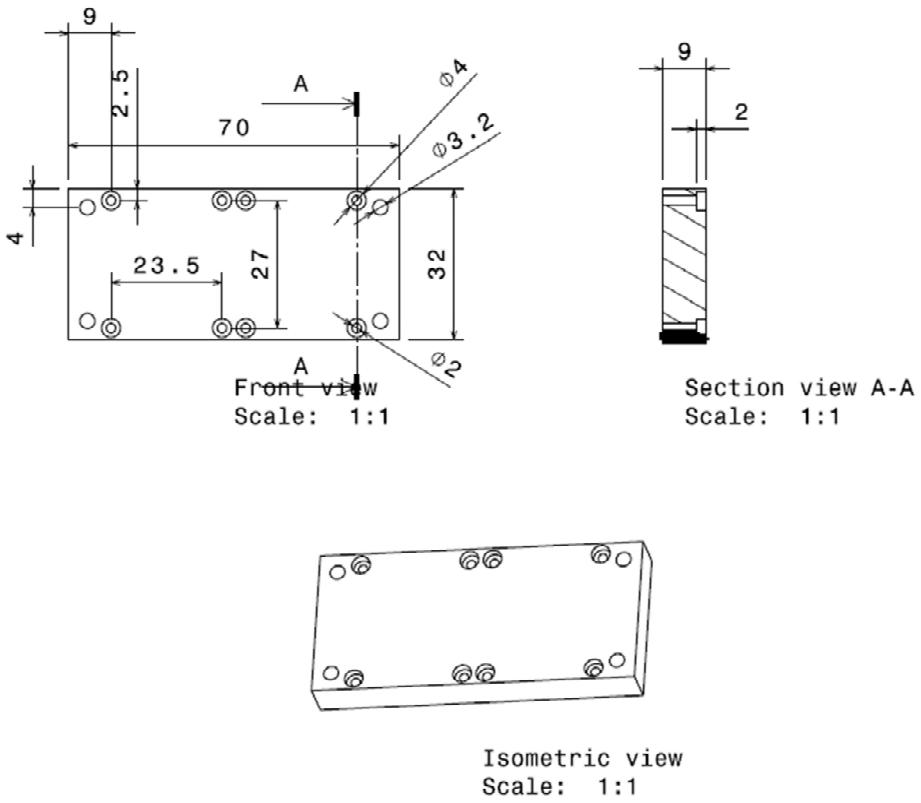
Figur E.5 - Ritning hålen i bakstycket (ej skalenlig)



Figur E.6 - Ritning lamphus, kropp (ej skalenlig)

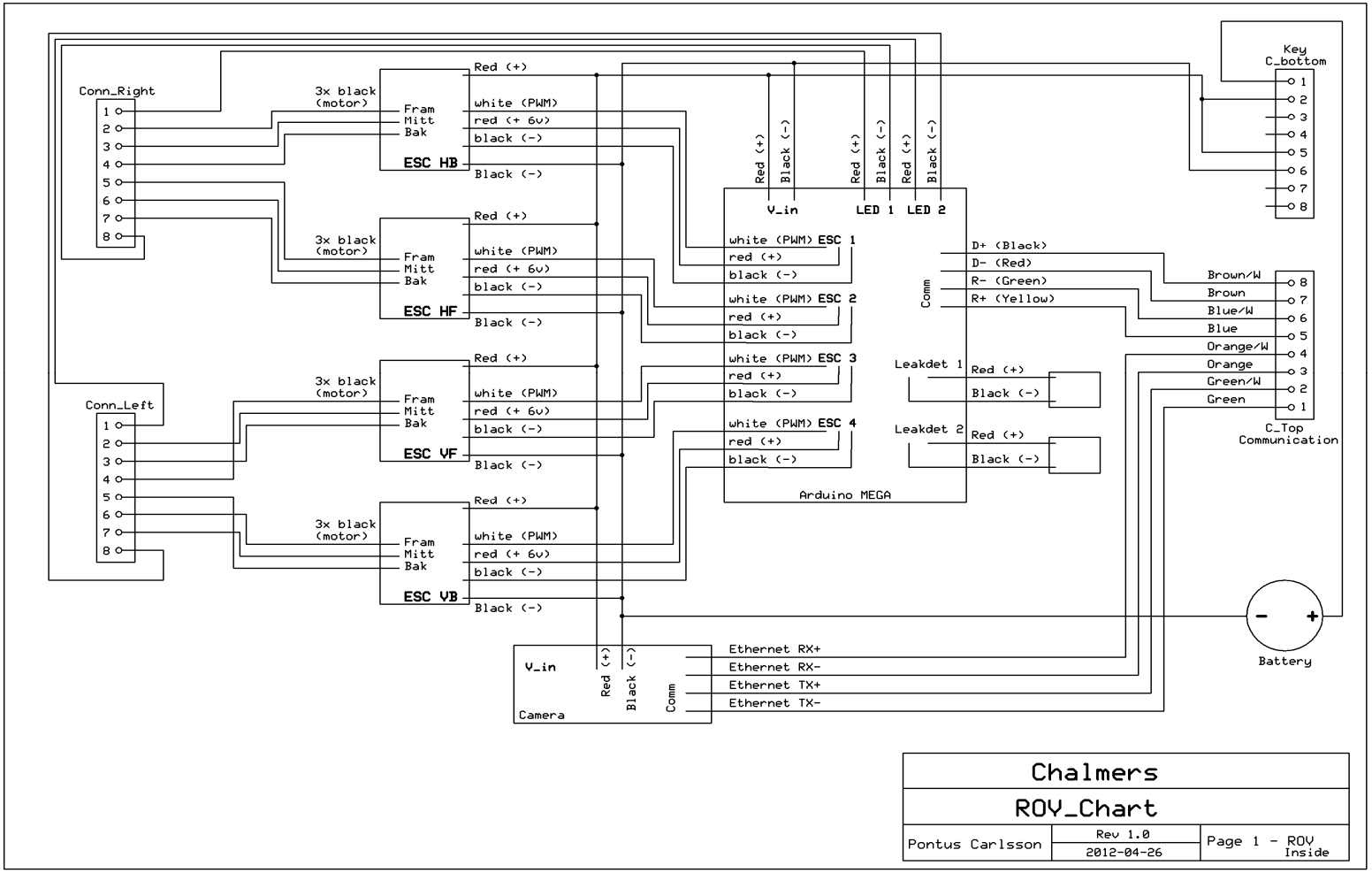


Figur E.7 - Ritning lamphus, låsring (ej skalenlig)



Figur E.8 - Ritning kylblock för ESC (ej skalenlig)

Appendix F – Kopplingsschema



Figur F.1 - Kopplingsschema över farkosten

Appendix G – Arduinokod

```
//ROV version 4
//Copyright Anders Markoff
#include <Servo.h>

//Kompassparametrar
#include <Wire.h>
#define MAG_ADDR  0x0E //7-bit address for the MAG3110, doesn't change
#define X_OFFSET  1469 // Offset of X-axis
#define Y_OFFSET  -1783 // Offset of Y-axis
#define Z_OFFSET  -1331 // Offset of Z-axis
int pitch_offset, roll_offset, heading_offset = 305;

//Kommunikationsparametrar
#define serialSpeed 57600
#define DATA_LENGTH 9
byte RxData[DATA_LENGTH] = {0};
byte TxData[DATA_LENGTH] = {0};
#define motor1 0
#define motor2 1
#define motor3 2
#define motor4 3
#define led 4
#define compass 5
#define water 6
#define battery 7
#define checksum 8

//Motorparametrar
Servo motorServo[4];  //Ett servoobjekt per motor
#define motorPin1 2  //Vilka pinnar motorerna är kopplade på
#define motorPin2 3
#define motorPin3 4
#define motorPin4 5
#define motorMidPos 90  //Värde för motorns mittläge = stillastående
#define motorMaxPos 170
#define motorMinPos 20

//LEDparametrar
#define ledPin1 11
#define ledPin2 12
#define ledMax 200
#define ledPinStatus 13

//Wattenparametrar
#define waterPin1 A3
#define waterPin2 A4

//Voltmeterparametrar
#define vinPin A2

//Temperaturparametrar
#define tempPin A1

void setup() {
//Säkerställ att motorerna är avstängda vid start
motorServo[motor1].write(motorMidPos);
  motorServo[motor2].write(motorMidPos);
  motorServo[motor3].write(motorMidPos);
```

```
  motorServo[motor4].write(motorMidPos);
//Anslut motorer på korrekt port
  motorServo[motor1].attach(motorPin1);
  motorServo[motor2].attach(motorPin2);
  motorServo[motor3].attach(motorPin3);
  motorServo[motor4].attach(motorPin4);

  setup_compass();

//Sätter PWM-frekvens till max
  TCCR1B = TCCR1B & 0b11111000 | 0x01;

//Initiera kommunikation
Serial.begin(serialSpeed);
}

void loop () {
if (Serial.available() >= DATA_LENGTH) {
//Tar emot data i ett array av längden DATA_LENGTH
for(int i = 0; i < DATA_LENGTH; i++){
RxData[i] = Serial.read();
    }

//Beräknar checksumma
int checksumRx = 0;
for (int i = 0; i < DATA_LENGTH - 1; i++) {
      checksumRx += RxData[i];
    }
    checksumRx = (byte)(checksumRx % 255);

//Jämför checksummorna så att korrekta data skickats
if (checksumRx == RxData[checksum]) {
//Släcker statusLED om checksumma ok
      digitalWrite(ledPinStatus, LOW);

//Ställer in korrekta värden på motorerna
      motorServo[motor1].write(RxData[motor1]);
      TxData[motor1] = RxData[motor1];
      motorServo[motor2].write(RxData[motor2]);
      TxData[motor2] = RxData[motor2];
      motorServo[motor3].write(RxData[motor3]);
      TxData[motor3] = RxData[motor3];
      motorServo[motor4].write(RxData[motor4]);
      TxData[motor4] = RxData[motor4];

//Ställer in LED
if (RxData[led] <= ledMax) {
        analogWrite(ledPin1, RxData[led]);
        analogWrite(ledPin2, RxData[led]);
        TxData[led] = RxData[led];
      }
else {
        analogWrite(ledPin1, ledMax);
analogWrite(ledPin2, ledMax);
        TxData[led] = ledMax;
      }
    }
else {
//Tänder statusLED om checksumma fel
      digitalWrite(ledPinStatus, HIGH);
    }
```

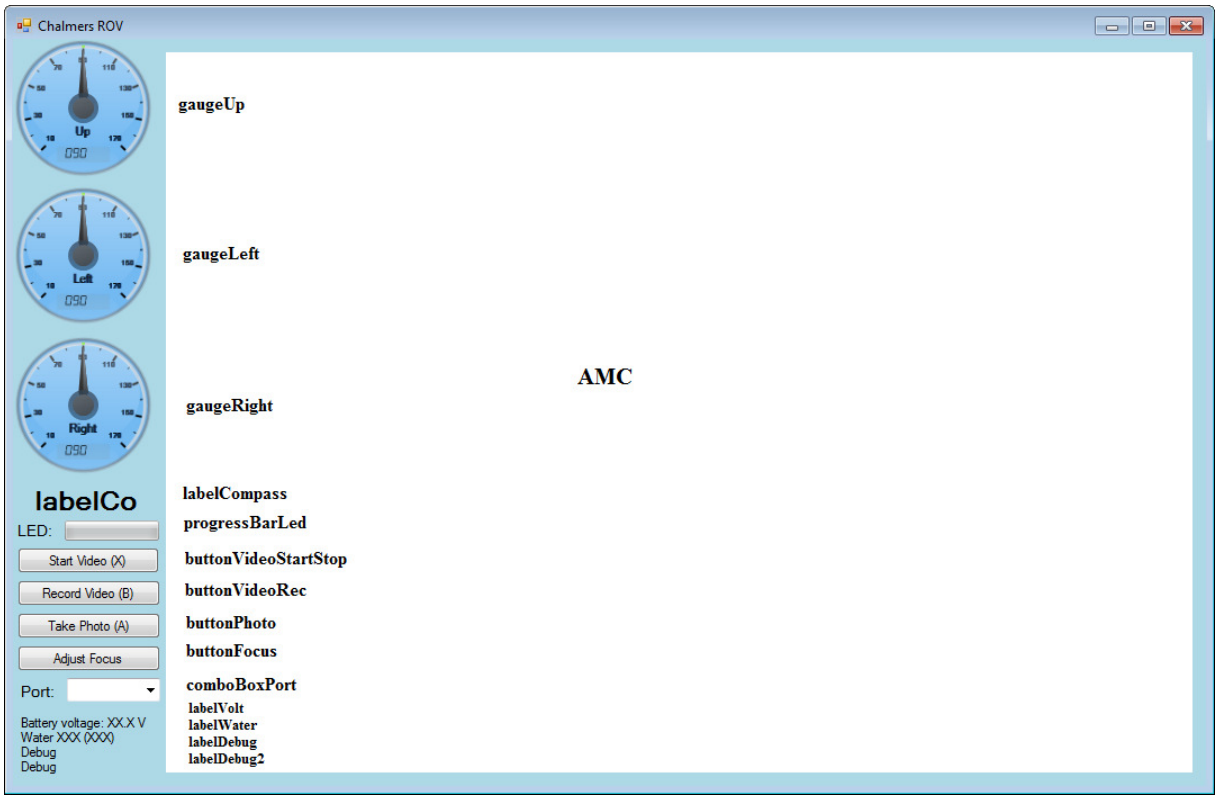
```
//Läser av vattensensorer
TxData[water] = map(max(analogRead(waterPin1), analogRead(waterPin2)), 0, 1023, 0,
100);

//Läser av batterispänningen
    TxData[battery] = map(analogRead(vinPin), 0, 1023, 0, 250);

//Läser av kompassen
    TxData[compass] = ((360-getHeading(readx(), ready(),  readz()))%360)/2;

//Beräknar checksumma
int checksumTx = 0;
for (int i = 0; i < DATA_LENGTH - 1; i++) {
      checksumTx += TxData[i];
    }
    TxData[checksum] = (byte)(checksumTx % 255);

//Skickar tillbaka data som ställts in
Serial.write(TxData, sizeof(TxData));
  }
}

/*
  MAG3110 Breakout Example Code

  by: Aaron Weiss, aaron at sparkfun dot com
      SparkFun Electronics 2011
  date: 9/6/11
  license: beerware, if you use this code and happen to meet me, you
           can by me a beer

  The code reads the raw 16-bit x, y, and z values and prints them
  out. This sketch does not use the INT1 pin, nor does it poll for
  new data.

*/



int x_min, x_max, y_min, y_max, z_min, z_max ;
//int pitch_offset, roll_offset, heading_offset ;

void setup_compass()
{
  config();            // turn the MAG3110 on
//heading_offset = getHeading(readx(), ready(),  readz());
//roll_offset = getRoll(readx(), ready(),  readz());
//pitch_offset = getPitch(readx(), ready(),  readz());
}

void config(void)
{
  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x11);              // cntrl register2
  Wire.write(0x80);              // write 0x80, enable auto resets
  Wire.endTransmission();       // stop transmitting

  delay(15);

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x10);              // cntrl register1
  Wire.write(1);                 // write 0x01, active mode
```

```
  Wire.endTransmission();       // stop transmitting
}

void print_heading(void)
{
int heading = getHeading(readx(), ready(),  readz());
//int roll = getRoll(readx(), ready(),  readz());
//int pitch = getPitch(readx(), ready(),  readz());

  Serial.print(heading);
  Serial.println("c");
//Serial.print(roll);
//Serial.println("x");
//Serial.print(pitch);
//Serial.println("z");
}

int readx(void)
{
int xl, xh;  //define the MSB and LSB

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x01);              // x MSB reg
  Wire.endTransmission();       // stop transmitting

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
while(Wire.available())    // slave may write less than requested
  {
    xh = Wire.read(); // read the byte
  }

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x02);              // x LSB reg
  Wire.endTransmission();       // stop transmitting

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
while(Wire.available())    // slave may write less than requested
  {
    xl = Wire.read(); // read the byte
  }

int xout = (xl|(xh << 8)); //concatenate the MSB and LSB
  xout -= X_OFFSET;
return xout;
}

int ready(void)
{
int yl, yh;  //define the MSB and LSB

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x03);              // y MSB reg
  Wire.endTransmission();       // stop transmitting

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop
```

```
  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
while(Wire.available())    // slave may write less than requested
  {
    yh = Wire.read(); // read the byte
  }

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x04);              // y LSB reg
  Wire.endTransmission();       // stop transmitting

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
while(Wire.available())    // slave may write less than requested
  {
    yl = Wire.read(); // read the byte
  }

int yout = (yl|(yh << 8)); //concatenate the MSB and LSB
  yout -= Y_OFFSET;
return yout;
}

int readz(void)
{
int zl, zh;  //define the MSB and LSB

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x05);              // z MSB reg
  Wire.endTransmission();       // stop transmitting

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
while(Wire.available())    // slave may write less than requested
  {
    zh = Wire.read(); // read the byte
  }

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x06);              // z LSB reg
Wire.endTransmission();       // stop transmitting

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
while(Wire.available())    // slave may write less than requested
  {
    zl = Wire.read(); // read the byte
  }

int zout = (zl|(zh << 8)); //concatenate the MSB and LSB
  zout -= Z_OFFSET;
return zout;
}

int getHeading(float x, float y, float z){
float heading=0;
```

```
if ((x == 0)&&(y < 0))
    heading= PI/2.0;
if ((x == 0)&&(y > 0))
    heading=3.0*PI/2.0;
if (x < 0)
    heading = PI - atan(y/x);
if ((x > 0)&&(y < 0))
    heading = -atan(y/x);
if ((x > 0)&&(y > 0))
    heading = 2.0*PI - atan(y/x);
  heading = degrees(heading) - heading_offset;
if(heading<0) heading +=360;
returnint(heading);
}

int getPitch(float x, float y, float z){
float pitch=0;
if ((z == 0)&&(y < 0))
    pitch= PI/2.0;
if ((z == 0)&&(y > 0))
    pitch=3.0*PI/2.0;
if (z < 0)
    pitch = PI - atan(y/z);
if ((z > 0)&&(y < 0))
    pitch = -atan(y/z);
if ((z > 0)&&(y > 0))
    pitch = 2.0*PI - atan(y/z);
  pitch = degrees(pitch) - pitch_offset;
if(pitch<0) pitch +=360;
returnint(pitch);
}
int getRoll(float x, float y, float z){
float roll=0;
if ((x == 0)&&(z < 0))
    roll= PI/2.0;
if ((x == 0)&&(z > 0))
    roll=3.0*PI/2.0;
if (x < 0)
    roll = PI - atan(z/x);
if ((x > 0)&&(z < 0))
roll = -atan(z/x);
if ((x > 0)&&(z > 0))
    roll = 2.0*PI - atan(z/x);
roll = degrees(roll) - roll_offset;
if(roll<0) roll +=360;
returnint(roll);
}
```

Appendix H – Datorkod



Figur H.1 - Bild på programmet med förklarande texter till respektive objekt

```
using System;
using System.Drawing;
using System.IO;
using System.IO.Ports;
using System.Windows.Forms;
//Lägger till biblioteken för xbox kontrollen
using Microsoft.Xna.Framework;
using Microsoft.Xna.Framework.Input;
usingColor = System.Drawing.Color;
using Input = Microsoft.Xna.Framework.Input; //Denna rad är för att slippa skriva
långa saker vid Button.Pressed
usingPoint = System.Drawing.Point;
using System.Collections.Generic;

namespace ROV_test1
{
publicpartialclassForm1 : Form
    {
//http://www.ucancode.net/Visual_C_MFC_Example/CSharp_Example_Free_DOTNET_Gauge_Contro
l_Draw_Source_Code.htm

//Statiska variabler för motorhastighet
staticbyte motorMidPos = 90;
staticbyte motorMinPos = 20;
staticbyte motorMaxPos = 170;

//Kommunikationsport
```

```
SerialPort port;
bool comOpen = false;
staticint serialSpeed = 57600;
staticbyte DATA_LENGTH = 9;
byte[] TxData = newbyte[DATA_LENGTH];
byte[] RxData = newbyte[DATA_LENGTH];
byte[] RxDataBuffer = newbyte[DATA_LENGTH];
int checksumError = 0;
int packages = 0;

//Till hjälp för protokollet
staticbyte motor1 = 0;
staticbyte motor2 = 1;
staticbyte motor3 = 2;
staticbyte motor4 = 3;
staticbyte led = 4;
staticbyte compass = 5;
staticbyte water = 6;
staticbyte battery = 7;
staticbyte checksum = 8;

//Värdet på hela kontrollen
GamePadState currentGamePadState;

//Variabler för de olika knapparna ändras i UpdateControllerInput()
bool A = false;
bool B = false;
bool X = false;
bool Y = false;
bool dPadUp = false;
bool dPadDown = false;
bool dPadLeft = false;
bool dPadRight = false;
bool start = false;
bool back = false;
bool LB = false;
bool RB = false;
bool leftStick = false;
bool rightStick = false;
float LT = 0; //Varierar mellan 0-10
float RT = 0;
float leftStickX = 0; //Varierar mellan -10-10
float leftStickY = 0;
float rightStickX = 0;
float rightStickY = 0;

//Variabler för LED
byte ledStep = 10;
byte ledMax = 200;

//Maximal vattennivå
byte waterMax = 0;
staticbyte waterTrigger = 20;

//Variabel för video
bool playVideo = false;
bool recVideo = false;
bool enableABX = true;

//Variabel för kompassen
List<int> compassList = newList<int>();
int compassRegulator = 40;  //Ju högre desto stabilare men långsammare uppdatering
```

```
//Variabel för voltmätare
List<int> voltageList = newList<int>();
int voltageRegulator = 40;  //Ju högre desto stabilare men långsammare uppdatering

public Form1()
        {
            InitializeComponent();
        }

privatevoid Form1_Load(object sender, EventArgs e) //Initierar lista med com-portar
och mätare
        {
foreach (string s in System.IO.Ports.SerialPort.GetPortNames())
comboBoxPort.Items.Add(s);
if (comboBoxPort.Items.Count == 1)
                comboBoxPort.SelectedIndex = 0;

            progressBarLed.Maximum = int.Parse(ledMax.ToString());

//Fyller kompasslistan med nollor
for (int i = 0; i < compassRegulator; i++)
                compassList.Add(0);

//Fyller voltmätarlistan med nollor
for (int i = 0; i < voltageRegulator; i++)
                voltageList.Add(0);

AMC.BackgroundColor = Color.Black.ToArgb();
}

privatevoid UpdateControllerInput()    //Uppdaterar alla kontrollvariablerna
        {
//Läser av hela kontrollen
            currentGamePadState = GamePad.GetState(PlayerIndex.One);

A = (currentGamePadState.Buttons.A == Input.ButtonState.Pressed);
            B = (currentGamePadState.Buttons.B == Input.ButtonState.Pressed);
            X = (currentGamePadState.Buttons.X == Input.ButtonState.Pressed);
            Y = (currentGamePadState.Buttons.Y == Input.ButtonState.Pressed);

            dPadUp = (currentGamePadState.DPad.Up == Input.ButtonState.Pressed);
            dPadDown = (currentGamePadState.DPad.Down == Input.ButtonState.Pressed);
            dPadLeft = (currentGamePadState.DPad.Left == Input.ButtonState.Pressed);
            dPadRight = (currentGamePadState.DPad.Right == Input.ButtonState.Pressed);

            start = (currentGamePadState.Buttons.Start == Input.ButtonState.Pressed);
            back = (currentGamePadState.Buttons.Back == Input.ButtonState.Pressed);

            LB = (currentGamePadState.Buttons.LeftShoulder ==
Input.ButtonState.Pressed);
            RB = (currentGamePadState.Buttons.RightShoulder ==
Input.ButtonState.Pressed);

            leftStick = (currentGamePadState.Buttons.LeftStick ==
Input.ButtonState.Pressed);
            rightStick = (currentGamePadState.Buttons.RightStick ==
Input.ButtonState.Pressed);

            LT = currentGamePadState.Triggers.Left * 10;
            RT = currentGamePadState.Triggers.Right * 10;
```

```
            leftStickX = currentGamePadState.ThumbSticks.Left.X * 10;
            leftStickY = currentGamePadState.ThumbSticks.Left.Y * 10;
            rightStickX = currentGamePadState.ThumbSticks.Right.X * 10;
            rightStickY = currentGamePadState.ThumbSticks.Right.Y * 10;
}

privatevoid UpdateMotorSpeed()  //Beräknar motorhastigheterna
        {
int motorSpeedLeft = (int)Math.Round((RT - LT + leftStickX) * 9 + motorMidPos);
int motorSpeedRight = (int)Math.Round((RT - LT - leftStickX) * 9 + motorMidPos);
int motorSpeedUp = (int)Math.Round(leftStickY * 9 + motorMidPos);

if (motorSpeedLeft < motorMinPos)
                TxData[motor1] = motorMinPos;
elseif (motorSpeedLeft > motorMaxPos)
                TxData[motor1] = motorMaxPos;
else
                TxData[motor1] = (byte)motorSpeedLeft;

if (motorSpeedRight < motorMinPos)
                TxData[motor2] = motorMinPos;
elseif (motorSpeedRight > motorMaxPos)
                TxData[motor2] = motorMaxPos;
else
                TxData[motor2] = (byte)motorSpeedRight;

if (motorSpeedUp < motorMinPos)
            {
                TxData[motor3] = motorMinPos;
TxData[motor4] = TxData[motor3];
            }
elseif (motorSpeedUp > motorMaxPos)
            {
                TxData[motor3] = motorMaxPos;
                TxData[motor4] = TxData[motor3];
            }
else
            {
                TxData[motor3] = (byte)motorSpeedUp;
                TxData[motor4] = TxData[motor3];
            }
        }

privatevoid timerController_Tick(object sender, EventArgs e)   //Huvudloopen
{
            timerController.Stop();
            UpdateControllerInput();  //Uppdatera kontrollen
            UpdateMotorSpeed();  //Uppdatera motorhastigheterna

//Kolla kontroll och skicka led tills strängen
if (dPadUp)
            {
if ((RxData[led] + ledStep) > ledMax)
                    TxData[led] = ledMax;
else
                    TxData[led] = (byte)(RxData[led] + ledStep);
            }
elseif (dPadDown)
            {
if ((RxData[led] - ledStep) < 0)
                    TxData[led] = 0;
else
```

```
                    TxData[led] = (byte)(RxData[led] - ledStep);
            }
elseif (dPadRight)
            {
                TxData[led] = ledMax;
            }
elseif (dPadLeft)
            {
                TxData[led] = 0;
            }
else
            {
TxData[led] = RxData[led];
}

//Kontrollera kontrollknapparna och aktivera video
if (X && enableABX)
            {
                buttonVideoStartStop.PerformClick();
enableABX = false;  //För att inte dubbelklicka av misstag
}
if (B && enableABX)
            {
                buttonVideoRec.PerformClick();
enableABX = false;  //För att inte dubbelklicka av misstag
}
if (A && enableABX)
            {
                buttonPhoto.PerformClick();
enableABX = false;  //För att inte dubbelklicka av misstag
            }

//Kontrollera bak-knappen och döda allt (måste ligga sist i loopen)
if (back)
            {
                TxData[motor1] = 0;
                TxData[motor2] = 0;
                TxData[motor3] = 0;
                TxData[motor4] = 0;
                TxData[led] = 0;
            }

            SendSerial();
        }

privatevoid SendSerial()   //Skickar datan på vald serieport
{
if (comOpen)
            {
int checksumTx = 0;
for (int i = 0; i < DATA_LENGTH - 1; i++)
                {
                    checksumTx += TxData[i];
                }
                TxData[checksum] = (byte)(checksumTx % 255);
                port.Write(TxData, 0, DATA_LENGTH);
            }
        }

privatevoid buttonVideoStartStop_Click(object sender, EventArgs e)  //Hanterar
Videoknappen
        {
```

```
            timerABX.Enabled = true;
if (!playVideo)
            {
                AMC.Play();
                playVideo = true;
                buttonVideoStartStop.Text = "Stop Video (X)";
            }
else
            {
                AMC.Stop();
                playVideo = false;
                buttonVideoStartStop.Text = "Start Video (X)";
            }
        }

privatevoid AMC_OnError(object sender,
AxAXISMEDIACONTROLLib._IAxisMediaControlEvents_OnErrorEvent e)
        {

        }

privatevoid timerRx_Tick(object sender, EventArgs e)
        {
if (comOpen && port.BytesToRead >= DATA_LENGTH)
{
//Läser in alla data i en buffer
port.Read(RxDataBuffer, 0, DATA_LENGTH);

//Kontrollerar checksumma
int checksumRx = 0;
for (int i = 0; i < DATA_LENGTH - 1; i++)
                {
                    checksumRx += RxDataBuffer[i];
                }
                checksumRx = (byte)(checksumRx % 255);

if (checksumRx == RxDataBuffer[checksum])
                    RxDataBuffer.CopyTo(RxData, 0);
else
                    labelDebug.Text = "Checksum error: " +
(++checksumError).ToString();

                labelDebug2.Text = "Packages: " + (++packages).ToString();

//Vrider hastighetsmätarna
                gaugeLeft.Value = RxData[motor1];
                gaugeRight.Value = RxData[motor2];
                gaugeUp.Value = RxData[motor3];

//Ställer in LED-bar
                progressBarLed.Value = RxData[led];

//Visar kompassen
                compassList.RemoveAt(0);
compassList.Add(RxData[compass]*2);
int compassavg = 0;
foreach (int i in compassList)
compassavg += i;
                labelCompass.Text = (compassavg/compassList.Count).ToString();
```

```
//Batterispänningen
                voltageList.RemoveAt(0);
                voltageList.Add(RxData[battery]);
int voltageavg = 0;
foreach (int i in voltageList)
                    voltageavg += i;
                labelVolt.Text = "Battery voltage: " + String.Format("{0:0.0}",
((float)(voltageavg / voltageList.Count) / 10)) + " V";

//Vattensensorerna
                labelWater.Text = "Water: " + RxData[water] + " (" +
waterMax.ToString() + ")";
if (RxData[water] > waterMax)
                    waterMax = RxData[water];
if (RxData[water] > waterTrigger)
labelWater.BackColor = Color.Red;
elseif (waterMax > waterTrigger)
                    labelWater.BackColor = Color.Pink;

//Uppdatera kontrollen och skicka nya data till arduinon
timerController.Start();
            }
        }

privatevoid comboBoxPort_SelectedIndexChanged(object sender, EventArgs e)  //Hanterar
porten
{
if (comOpen)
                port.Close();
            port = newSerialPort(comboBoxPort.SelectedItem.ToString(), serialSpeed);
            port.Open();
            comOpen = true;
            SendSerial();
}

privatevoid buttonVideoRec_Click(object sender, EventArgs e)   //Hanterar recknappen
        {
            timerABX.Enabled = true;

if (!playVideo)
                buttonVideoStartStop.PerformClick();


string directoryName = "C:\\Users\\ROV\\Documents\\ROV\\" +
DateTime.Now.ToString("yyyy-MM-dd");
if (!Directory.Exists(directoryName))
Directory.CreateDirectory(directoryName);


String videoURI = directoryName + "\\ROV_" + DateTime.Now.ToString("yyyyMMdd_hhmmss")
+ ".asf";

if (!recVideo)
            {
                AMC.StartRecord(videoURI);
                recVideo = true;
                buttonVideoRec.Text = "Stop recording (B)";
                buttonVideoRec.BackColor = Color.Red;
            }
else
            {
                AMC.StopRecord();
```

```
                recVideo = false;
                buttonVideoRec.Text = "Record video (B)";
buttonVideoRec.BackColor = SystemColors.Control;
            }
        }

privatevoid timerAB_Tick(object sender, EventArgs e)   //Hanterar ofrivilliga
dubbelklick på knapparna
{
            timerABX.Enabled = false;
            enableABX = true;
            buttonPhoto.BackColor = SystemColors.Control;
        }

privatevoid buttonPhoto_Click(object sender, EventArgs e)  //Hanterar fotoknappen
        {
            timerABX.Enabled = true;

if (!playVideo)
                buttonVideoStartStop.PerformClick();

string directoryName = "C:\\Users\\ROV\\Documents\\ROV\\" +
DateTime.Now.ToString("yyyy-MM-dd");
if (!Directory.Exists(directoryName))
Directory.CreateDirectory(directoryName);

String photoURI = directoryName + "\\ROV_" + DateTime.Now.ToString("yyyyMMdd_hhmmss")
+ ".jpg";
            AMC.SaveCurrentImage(0, photoURI);
            buttonPhoto.BackColor = Color.CornflowerBlue;
}

privatevoid buttonFocus_Click(object sender, EventArgs e)  //Öppnar internet explorer
och fokuserar kameran
{
            System.Diagnostics.Process proc = new System.Diagnostics.Process();
            proc.StartInfo.FileName = "iexplore";
//proc.StartInfo.Arguments = "https://192.168.0.90/axis-
cgi/opticssetup.cgi?autofocus=perform";
proc.StartInfo.Arguments = "http://192.168.0.90/operator/focus.shtml?basic=yes&id=11";
            proc.Start();
        }
    }
}
```