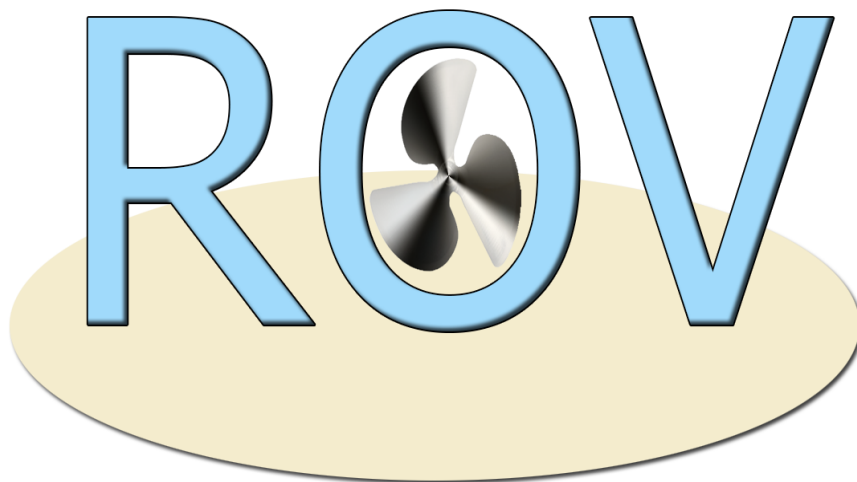


Designspecifikation  
Remotely Operated Underwater Vehicle  
Version 1.0

Simon Lindblom

9 oktober 2014



**Status**

Granskad	JA, OW	2014-10-08
Godkänd	Isak Nielsen	2014-10-06

## Projektidentitet

**E-post:** tsrt10\_rov2014@googlegroups.com

**Hemsida:** <http://www.isy.liu.se/edu/projekt/reglerteknik/2014/rov/>

**Beställare:** Isak Nielsen, ISY, Linköpings universitet  
**Telefon:** +46(0) 13 282804  
**E-post:** isak.nielsen@liu.se

**Kund:** Micael Derelöv, Saab Dynamics, Underwater Systems  
**Telefon:** +46(0) 13 281165  
**E-post:** micael.derelov1@saabgroup.com

**Kursansvarig:** Daniel Axehill, ISY, Linköpings universitet  
**Telefon:** +46(0) 13 284042  
**E-post:** daniel@isy.liu.se

**Projektledare:** Oscar Wyckman  
**Telefon:** +46(0) 73 7338744  
**E-post:** oscwy416@student.liu.se

**Handledare:** Jonas Linder, ISY, Linköpings universitet  
**Telefon:** +46(0) 13 282804  
**E-post:** jonas.linder@liu.se

## Gruppmedlemmar

Namn	Roll	Telefon	E-post (@student.liu.se)
Oscar Wyckman (OW)	Projektledare	073 - 733 87 44	oscwy416
Simon Lindblom (SL)	Dokumentansvarig	070 - 576 26 64	simli427
Dennis Forsberg (DF)	Regleringsansvarig	076 - 029 08 35	denfo765
Oscar Gunnarsson (OG)	Designansvarig	073 - 837 41 92	oscu132
Elias Nilsson (EN)	Testansvarig	073 - 729 62 47	elini289
Johan Andersson (JA)	Simuleringsansvarig	070 - 332 92 12	johan712
Sofia Larsson Cahlin (SLC)	Projektgruppssamordnare	076 - 881 40 06	sofla266
Marcus Johansson (MJ)	Mjukvaruansvarig	070 - 315 73 77	marma906

## Dokumenthistorik

<b>Version</b>	<b>Datum</b>	<b>Ändringar</b>	<b>Utförda av</b>	<b>Granskad</b>
0.1	2014-09-29	Första utkastet	Samtliga	OW, DF
0.2	2014-10-02	Andra utkastet	Samtliga	SL, DF, OG
0.3	2014-10-04	Tredje utkastet	Samtliga	JA
1.0	2014-10-09	Första versionen	Samtliga	JA, OW

# Innehåll

<b>1</b>	<b>Inledning</b>	<b>1</b>
1.1	Syfte och mål . . . . .	1
1.2	Användning . . . . .	1
1.3	Notation . . . . .	2
<b>2</b>	<b>Översikt av systemet</b>	<b>2</b>
2.1	Produktkomponenter . . . . .	3
2.2	Beroenden till andra system . . . . .	3
2.3	Ingående delsystem . . . . .	4
2.4	Avgränsningar . . . . .	4
2.5	Designfilosofi . . . . .	4
2.6	Intern PC . . . . .	4
2.7	Extern PC . . . . .	4
<b>3</b>	<b>Modell</b>	<b>5</b>
3.1	Ingående koordinatsystem . . . . .	5
3.2	Modellparametrar och koefficienter . . . . .	6
3.3	Förenklingar i modellen . . . . .	7
3.4	Framtagning av modell . . . . .	8
<b>4</b>	<b>Hårdvara</b>	<b>10</b>
4.1	Styr- och mätkort . . . . .	10
4.2	Magnetometer . . . . .	12
4.3	Trycksensorer . . . . .	12
4.4	Motorer . . . . .	13
4.5	Inertial Measurement Unit . . . . .	14
4.6	Intern PC . . . . .	14
4.7	Övrig hårdvara . . . . .	15
<b>5</b>	<b>Mjukvara</b>	<b>15</b>
5.1	Robot Operating System . . . . .	15
5.2	Styr- och mätkort . . . . .	15
5.3	Grafiskt gränssnitt . . . . .	16
<b>6</b>	<b>Delsystem Sensorfusion</b>	<b>17</b>
6.1	Kommunikation med andra delsystem . . . . .	17
6.2	Observatör . . . . .	18
6.2.1	Mätekvationer trycksensor . . . . .	19
6.2.2	Mätekvationer accelerometer . . . . .	19
6.2.3	Magnetometer . . . . .	20
6.2.4	Gyro . . . . .	20
<b>7</b>	<b>Delsystem Reglering</b>	<b>20</b>
7.1	Kommunikation med andra delsystem . . . . .	21
7.2	Decentraliserad regulator . . . . .	21
7.3	LQ-regulator . . . . .	23
7.3.1	Linjärisering av ROV:en . . . . .	24
7.3.2	Regulatorparametrar . . . . .	26
7.3.3	Trimning av LQ-regulatorn . . . . .	27
<b>8</b>	<b>Delsystem Simulering</b>	<b>28</b>

8.1	Kommunikation med andra delsystem . . . . .	29
<b>9</b>	<b>Delsystem Kommunikation</b>	<b>30</b>
9.1	Kommunikation med externa enheter . . . . .	32
9.2	Kommunikation med interna enheter . . . . .	32
<b>10</b>	<b>Delsystem Referensgränssnitt</b>	<b>33</b>
10.1	Kommunikation med andra delsystem . . . . .	33
<b>A</b>	<b>Topics och messages</b>	<b>36</b>

# 1 Inledning

Inom både militära och civila tillämpningar växer intresset och behovet av autonoma farkoster som kan utföra diverse uppdrag i luften, på land och till sjöss utan kontakt med en operatör. Exempel på sådana uppdrag för en undervattensfarkost kan vara övervakning, kartering eller reparationsarbeten.

Detta projekts undervattensfarkost (ROV) är ca 1,85 m lång, fjärrstyrd och torpedliknande. Den är utrustad med styrsystem och sensorer. Den har designats, konstruerats samt vidareutvecklats i tidigare projekt och examensarbeten på Linköpings universitet.

I detta dokument presenteras ROV-systemet med den funktionalitet som det avses innehålla vid projektets slut.

## 1.1 Syfte och mål

Långsiktigt är projektets mål att utveckla en helt autonom farkost som kan delta i den europeiska tävlingen för autonoma undervattensfarkoster, SAUC-E. I tävlingen ska farkosterna utföra vissa givna uppdrag på kortast möjliga tid. För att uppnå detta mål behöver den befintliga ROV:en vidareutvecklas till en helt autonom farkost som kan orientera sig i sin omgivning. Därutöver måste den vara utrustad med hård- och mjukvara som klarar av de uppgifter som tilldelas de tävlande.

Målet för nuvarande projektgrupp är att utveckla ett robust styrsystem för en väl fungerande reglering och navigering. Detta ska uppnås genom utveckling av systemet inom reglering och sensorfusion, simulering, samt hårdvaruintegrering. Befintlig reglering och parameterskattningar skall förbättras och regulatorprestandan ska utvärderas. Vid leverans ska även skattning och reglering av attityd, vinkelhastighet och djup vara implementerade. Modellen för ROV:en ska vidareutvecklas och testas i en simuleringsmiljö som byggs upp. Dessutom ska denna simuleringsmiljö utökas med funktionalitet för "hardware-in-the-loop" och SONAR-simuleringar. Vidare ska styr- och mätkort, trycksensorer för djupmätning och en extern magnetometer integreras i ROV:en.

Kraven på projektet finns beskrivna i kravspecifikationen [4].

## 1.2 Användning

Hittills har ROV:en varit ett utvecklingsprojekt inom projektarbetskurser och examensarbeten vid Linköpings universitet. Det kommer den fortsätta att vara några år framöver. Målet är att ROV:en på lång sikt ska kunna delta i tävlingen SAUC-E. Kunskapen som samlas genom utvecklingen kan komma att nyttjas av SAAB Dynamics, Underwater Systems.

## 1.3 Notation

AUV	Autonomous Underwater Vehicle
EKF	Extenden Kalman Filter
IMU	Inertial Measurement Unit
ISY	Institutionen för systemteknik
LQ	Linear Quadratic
PWM	Pulse-Width Modulation
ROS	Robot Operating System
ROV	Remotely Operated Vehicle
SAUC-E	Student AUV Challenge Europe
SONAR	Sound Navigation and Ranging

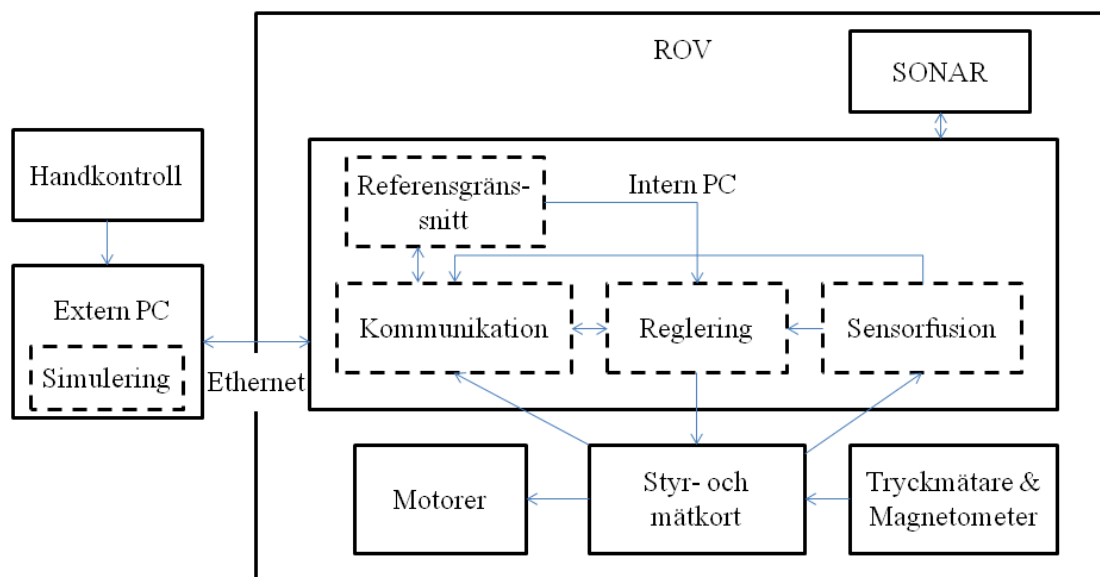
## 2 Översikt av systemet

Den produkt som ska vidareutvecklas i detta projekt är en ROV som lämnats över från tidigare års projektkurser och examensarbeten. Den är ca 1,85 m lång och består av flera moduler. ROV:en har en intern och en extern PC för styrsystem och manuell manövrering via en Xbox-kontroll. Man kan från den externa PC:n välja att köra ROV:en i två lägen, manuellt eller autonomt. I det manuella läget kan man köra varje motor separat. I det autonoma läget kan man välja att använda en av två regulatorer, en LQ-regulator eller en decentraliserad regulator. Förutom den interna PC:n finns olika styrkort och sensorer monterade på farkosten för att möjliggöra orientering och reglering. För framdrivning finns motorer och propellrar.

När projektet startar är mjukvaran indelad i fyra delsystem. Dessa sköter uppgifter kopplade till reglering, sensorfusion, kommunikation och simulering av ROV:en. Delsystemen Kommunikation, Reglering och Sensorfusion finns implementerade i den interna PC:n och delsystemet Simulering på den externa PC:n. Både interna och externa PC:n kör operativsystemet Linux och ovanpå det körs operativsystemet ROS för att sköta kommunikationen mellan de olika delsystemen.

Referenssignalerna till regulatorerna skickas i dagsläget från den externa PC:n via delsystemet Kommunikation direkt till delsystemet för reglering. För att anpassa mjukvaran mot en framtida autonom farkost kommer ett nytt delsystem kallat Referensgränssnitt införas under detta projekt. Detta är tänkt att hantera alla referensvärden då dessa måste beräknas internt på ROV:en vid framtida autonom körning. Under detta projekt kommer dock bara strukturen införas som ett första steg och referenserna kommer fortsätta skickas från den externa PC:n men via delsystemet Kommunikation till Referensgränssnitt som i sin tur skickar vidare referenserna till delsystemet Reglering.

En översikt över ROV:ens mjuk- och hårdvara som den ska se ut vid projektets slut kan ses i Figur 1.



Figur 1: En översikt över systemet där pilarna indikerar informationens flödesriktning. Streckade block motsvarar mjukvara och heldragna block motsvarar hårdvara.

## 2.1 Produktkomponenter

De komponenter som finns att tillgå i projektet är ROV:en i befintligt skick med monterade komponenter, den externa PC:n och en Xbox-kontroll. Dessutom finns nya komponenter i form av ett nytt styr- och mätkort, nya trycksensorer, en magnetometer samt en ny extern dator.

Den hårdvara som finns monterad på ROV:en i dagsläget är en intern PC, en mikrokontroller, ett arduinokort, fem styrkort, en trycksensor, en IMU med inbyggd magnetometer, en läckagesensor, fem propellrar, två lampor, en kamera, en SONAR och ett antal batterier [3].

Under projektet ska det nuvarande styr- och mätkortet i form av arduinokortet med mikrokontrollern och den nuvarande IMU:n bytas ut mot det nya styr- och mätkortet som har en intern IMU. Då den gamla magnetometern var intern i den IMU som ska bytas ut kommer en ny, extern magnetometer kopplas in istället. Den befintliga trycksensorn kommer dessutom att bytas ut och eventuellt kommer ytterligare en trycksensor att integreras. Den nuvarande externa datorn ska ersättas helt och all mjukvara kommer att föras över från den gamla externa datorn till den nya.

## 2.2 Beroenden till andra system

ROV:en kommer att vara beroende av den externa PC:n eftersom det är därigenom all kommunikation med ROV:en sker. Ifall ROV:en förlorar kontakten med den externa PC:n ska den stanna upp och med horisontell orientering stiga mot ytan. ROV:en kommer också att vara beroende av handkontrollen, utifrån vilken man styr ROV:en. I år kommer även en extern undervattensfarkost att utvecklas i ett annat projekt. ROV:en ska framöver kunna kommunicera med den externa undervattensfarkosten och den externa undervattensfarkosten ska kunna docka vid ROV:en. Även den externa PC:n ska kunna kommunicera med den externa undervattensfarkosten.



## 2.3 Ingående delsystem

ROV:ens delsystem är Reglering, Sensorfusion, Kommunikation, Referensgränssnitt och Simulering. Delsystemen Reglering, Sensorfusion, Kommunikation och Referensgränssnitt kommer att finnas implementerade på den interna datorn på ROV:en medan delsystemet Simulering kommer att vara implementerat på den externa datorn. Delsystemet Sensorfusion har i uppgift att utifrån signaler från flera sensorer skapa en skattning av ROV:ens orientering. Mer information om sensorerna hittas i Kapitel 6. Delsystemet Reglering har i uppgift att utifrån data från delsystemet Sensorfusion reglera ROV:en efter en given referens. Referensen erhålls i dagsläget från den externa datorn och kommunikationen där emellan sker över Ethernet, vilket delsystemet Kommunikation ansvarar för. Referensen från den externa datorn ska efter projektet skickas in i ett referensgränssnitt som skickar den vidare till delsystemet Reglering. För att få ROV:en autonom är det i framtiden tänkt att referenserna ska kunna skapas ombord på den interna PC:n i delsystemet Referensgränssnitt. Delsystemet Kommunikation ska erhålla data från styr- och mätkortet och delsystemen Reglering, Referensgränssnitt och Sensorfusion, se Figur 1. Delsystemet Simulering har i uppgift att simulera ROV:en så att man inte behöver lika mycket tid för tester i bassäng. Det ska även vara möjligt att använda ROV:ens hårdvara för att styra simuleringsmodellen av ROV:en. Mer om detta i Kapitel 8.

## 2.4 Avgränsningar

ROV:en är en gemensam resurs med det parallella projektet som har ansvar för dess mekanik och utformning. Förutom kommunikation mellan de båda projekten och frågor rörande placering av nya sensorer omfattas inte detta projekt av den fysiska utformningen av ROV:en.

## 2.5 Designfilosofi

Eftersom ROV:en är en gemensam resurs är den konstruerad i moduler för att möjliggöra parallell utformning och testning av ny funktionalitet. Det är därför viktigt att ha ett aktivt modultänk genom hela projektet. För att underlätta i överlämningen till kommande projekt och examensarbeten fyller detta modultänk en viktig funktion även i mjukvaruutvecklingen.

## 2.6 Intern PC

På den interna PC:n som sitter på ROV:en implementeras mjukvaran för referensgränssnittet och delsystemen Kommunikation, Reglering och Sensorfusion. PC:n använder operativsystemet Linux. I Linuxmiljön körs programvaran Robot Operating System (ROS) som används för kommunikation mellan delsystemen. För mer information om ROS, se Kapitel 5.

## 2.7 Extern PC

Den externa PC:n är en laptop vars anslutning till ROV:en sker via en Ethernetkabel. PC:n används i nuläget för att kontrollera och styra ROV:en. ROV:ens användare kan styra ROV:en med ett GUI på den externa PC:n eller genom att ansluta en Xbox-kontroll till PC:n och styra ROV:en med den. I den externa PC:ns GUI kan användaren bland annat välja regulator, sätta ROV:ens referensvärden och ladda in parametrar. Dessutom kan man avläsa ROV:ens position, orientering och styrsignaler ur GUI:t. För mer information om ROV:ens GUI, se Kapitel 5.3.

Den externa PC:n kommer att bytas ut mot en ny under projektets gång. Den mjukvara som används till ROV:en måste alltså flyttas över från den gamla PC:n till den nya.

### 3 Modell

Den fysikaliska modell som ligger till grund för ROV:en var först framtagen av Patricia Sundin m.fl. [7], men har på grund av att dess rimlighet ej var undersökt modifierats enligt en modell utarbetad av Fossen [1]. Denna modifikation gjordes för att kunna ta hänsyn till tillagd massa från corioliseffektens inverkan vilket inte gjordes tidigare, samt för att bidra med en mer vedertagen notation för modellparametrarna. Tanken var att göra modellen mer strukturerad och lättförstådd.

Modellen bygger på att teckna Newtons rörelseekvationer och Newtons momentekvation i ett kroppsfixt (relativt ROV:en) koordinatsystem och sedan lägga till bidragande krafter. Dessa ekvationer beskriver sedan ROV:ens hastighet och vinkelhastighet i det kroppsfixa koordinatsystemet.

Genom en rotationstransform kan ROV:ens hastigheter beskrivas i ett jordfixt koordinatsystem. I delsystem Sensorfusion används kvaternioner  $\mathbf{q} = (q_0, q_1, q_2, q_3)^T$  istället för Eulervinklar för att undvika singularitet då pitch-vinkeln  $\theta = \pm 90^\circ$ . Däremot föredras användning av Eulervinklar i delsystem Reglering på grund av att det möjliggör för ett mer intuitivt tillvägagångssätt vid design av reglersystemet. Vid normal användning anses detta inte vara ett betydande problem.

I detta avsnitt används notationen  $s\theta = \sin(\theta)$  och  $c\theta = \cos(\theta)$  för de två trigonometriska funktionerna sin och cos för att underlätta nedtecknandet av längre trigonometriska uttryck.

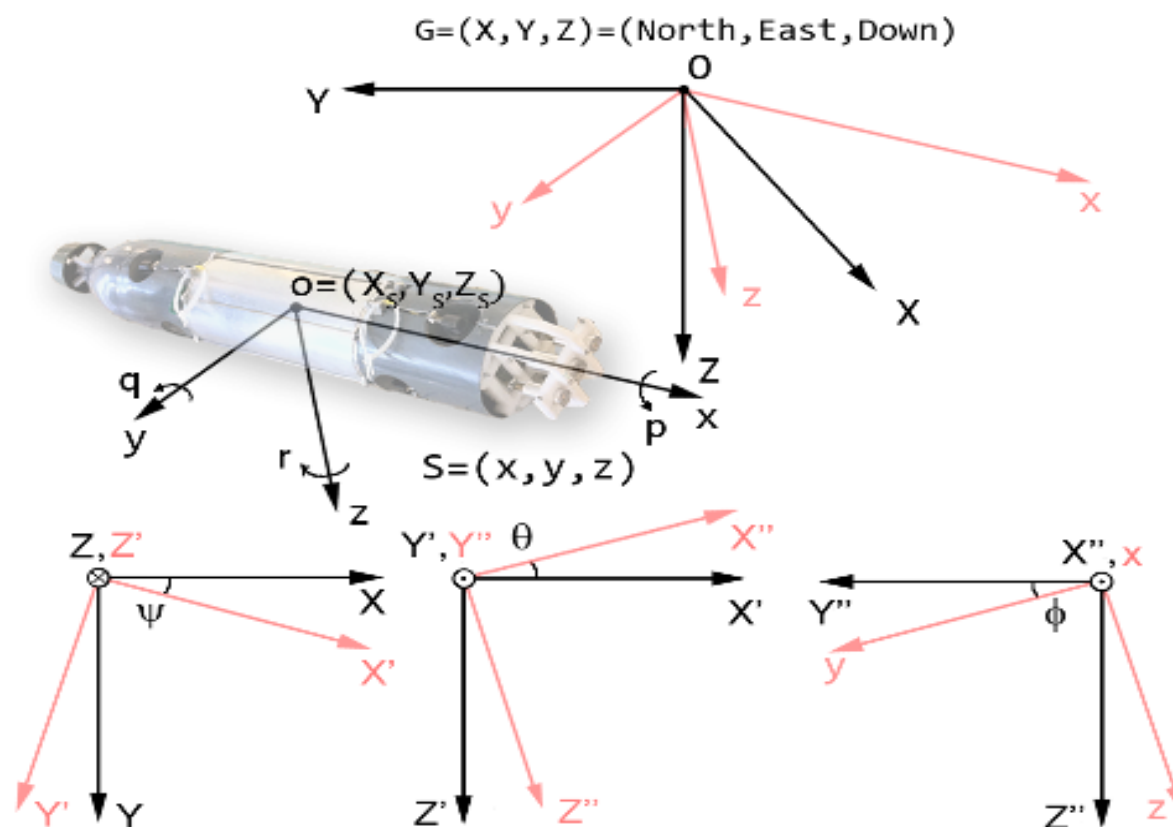
#### 3.1 Ingående koordinatsystem

I modellen har två koordinatsystem införts, ett ROV-fixt  $(x, y, z)^T$  och ett jordfixt inertialsystem  $(X, Y, Z)^T$  där Newtons rörelseekvationer gäller. Det förstnämnda har sitt origo i ROV:ens masscentrum med x-axeln riktad framåt längs med ROV:en, y-axeln åt styrbord samt z-axeln nedåt under ROV:en. Det senare koordinatsystemet har sitt origo i vattenytan med X-axeln riktad i nordlig, Y-axeln i östlig riktning och Z-axeln mot jordens centrum. Se Figur 2.

En rotationsmatris  $\mathbf{R}$  används för övergång från jordfixa till ROV-fixa koordinater. Den är framtagen genom multiplikation av tre rotationsmatriser som var och en roterar med aktuell yaw- ( $\psi$ ), pitch- ( $\theta$ ) och roll-vinkel ( $\phi$ ) kring respektive jordfix Z-, Y- och X-axel i den ordningen. Se (3.1). En alternativ version finns för omvandling med hjälp av en kvaternion, se ekvation (3.2).

$$\mathbf{R}(\phi, \theta, \psi) = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\phi \\ s\phi s\theta c\psi - c\phi s\psi & s\phi s\theta s\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{bmatrix} \quad (3.1)$$

$$\mathbf{R} = \mathbf{R}(q_0, q_1, q_2, q_3) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2q_1q_2 - 2q_0q_3 & 2q_0q_2q_1q_3 \\ 2q_0q_3 + 2q_1q_2 & q_0^2 - q_1^2 + q_2^2 - q_3^2 & -2q_0q_1 + 2q_2q_3 \\ -2q_0q_2 + 2q_1q_3 & 2q_2q_3 + 2q_0q_1 & q_0^1 - q_1^2q_2^2q_3^2 \end{bmatrix} \quad (3.2)$$



Figur 2: Bilden visar hur det kroppsfixa koordinatsystemet förhåller sig till ROV:en, hur det globala koordinatsystemet är orienterat till det lokala samt hur rotation av dessa ser ut. Bilden är hämtad från 2012 års projektgrupp [7].

En matris  $\mathbf{T}$  används för att transformera de ROV-fixa vinkelhastigheterna till derivator av Eulervinklarna, se (3.3). Ett alternativ finns för en transform för hastigheter med hjälp av en kvaternion, se (3.4).

$$\mathbf{T}(\phi, \theta, \psi) = \begin{bmatrix} 1 & s\phi\theta & c\phi\theta \\ 0 & c\phi & -s\phi \\ 0 & \frac{s\phi}{c\theta} & \frac{c\phi}{c\theta} \end{bmatrix} \quad (3.3)$$

$$\mathbf{T} = \mathbf{T}(q_0, q_1, q_2, q_3) = -\frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & q_3 & -q_2 \\ -q_3 & q_0 & q_1 \\ q_2 & -q_1 & q_0 \end{bmatrix} \quad (3.4)$$

### 3.2 Modellparametrar och koefficienter

Nedan i Tabell 1 och 2 tecknas och beskrivs de ingående parametrarna och koefficienterna i modellen. Modellkoefficienterna har skattats med hjälp av systemidentifiering och med information från en mass-

fördelningsmässigt verklighetstrogen CAD-modell av ROV:en. Ett fåtal koefficienters värden har tagits fram analytiskt.

Under detta projekt ska dessa skattningar utvärderas och eventuellt omprövas. Detta kommer främst att ske genom fysikaliska tester och System Identification ToolBox i MatLab.

Parametersymbol(er)	Beskrivning
$n, e, d$	Globala koordinater i $X$ -, $Y$ - och $Z$ -led för ROV:en.
$\phi, \theta, \psi$	Eulervinklar för roll, pitch och yaw.
$q_0, q_1, q_2, q_3$	Kvaternioner som beskriver en transformation mellan det globala koordinatsystemet och det ROV-fixa koordinatsystemet.
$u, v, w$	Hastigheter i det ROV-fixa koordinatsystemet längs $x$ -, $y$ - och $z$ -axlarna.
$p, q, r$	Rotationshastigheter i det ROV-fixa koordinatsystemet längs $x$ -, $y$ - och $z$ -axlarna.
$u_T, u_{yf}, u_{yr}, u_{zf}, u_{zr}$	Pålagd spänning för respektive motor.

Tabell 1: Tabell som täcker de ingående parametrar i modellen med symbolförklaring och kort beskrivning. Parametrar för ROV:ens position och attityd tillsammans med motorspänningar ingår.

Koefficientsymbol(er)	Beskrivning
$K_p, M_q, N_r$	Linjära motståndskoefficienter för rotation i vatten.
$K_{p p }, M_{q q }, N_{r r }$	Kvadratiska motståndskoefficienter för rotation i vatten.
$m$	ROV:ens massa.
$X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}$	Tillagd massa i $x$ -, $y$ -, $z$ -led på grund av translation i vatten.
$I_x, I_y, I_z$	Tröghetsmoment runt $x$ -, $y$ -, $z$ -axlarna.
$K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}$	Ökat tröghetsmoment runt $x$ -, $y$ -, $z$ -axlarna på grund av rotation i vatten.
$z_B$	Avståndet längs $z$ -axeln från masscentrum till flytkraftens angreppspunkt.
$x_{yf}, x_{yr}, x_{zf}, x_{zr}$	Avståndet längs $x$ -axeln från masscentrum till respektive motor.
$X_u, Y_v, Z_w$	Avståndet längs $z$ -axeln från masscentrum till flytkraftens angreppspunkt.
$X_{ u }, Y_{ v }, Z_{ w }$	Kvadratiska motståndskoefficienter för translation i vatten.
$C_T, C_{M_T}, C_{y_r}, C_{y_f}, C_{z_r}, C_{z_f}$	Motorkoefficienter för respektive motor, där $y$ och $z$ står för motorerna i $y$ -respektive $z$ -planet $r$ och $f$ står för rear (bakre) respektive front (främre).
$V$	ROV:ens undanträngningsvolym.
$\rho$	Vattnets densitet.
$g$	Tyngdacceleration.

Tabell 2: Tabell över de koefficienter som ingår i modellen med beskrivning.

### 3.3 Förenklingar i modellen

För att göra modellen mer hanterbar har en del förenklingar gjorts. Tröghetsmatrisen har kommit att bli diagonal sedan symmetri för ROV:ens kropp antagits i  $xy$ -,  $xz$ - och  $yz$ -planen. ROV:en är inte helt symmetrisk i dessa plan, och eftersom tröghetsmomenten påverkar ekvationerna för vinkelaccelerationerna blir modellen mer osäker. Detta korrigeras dock genom mätning av vinkelhastigheterna.

Rörelserna från ROV:en har antagits vara frikopplade från det hydrodynamiska systemet i vattnet för att förenkla beräkningen av dämpningen. Detta anses vara godtagbart i och med ROV:ens relativt låga hastigheter.

### 3.4 Framtagning av modell

Utgångspunkten för modellframställandet är att sammanfatta Newtons rörelseekvationer och momentekvation med endast ett uttryck där hänsyn tas till tillagd massa för kompenserad tröghetsmatris och corioliseffekt. Därför låts ROV:ens position och attityd betecknas i jordfixa koordinater med  $\boldsymbol{\eta} = (n, e, d, \phi, \theta, \psi)^T$  för användning av Eulervinklar, eller  $\boldsymbol{\eta} = (n, e, d, q_0, q_1, q_2, q_3)^T$  för användning av kvaternioner, samt ROV:ens kroppsfixa hastigheter och vinkelhastigheter betecknas av  $\boldsymbol{\nu} = (u, v, w, p, q, r)^T$ .

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \quad (3.5)$$

I Ekvation (3.5) betecknar:

- $\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A$  ROV:ens tröghetsmatris för stel kropp samt tillagd massa
- $\mathbf{C}(\boldsymbol{\nu}) = \mathbf{C}_{RB}(\boldsymbol{\nu}) + \mathbf{C}_A(\boldsymbol{\nu})$  matrisen som beskriver corioliseffekten för stel kropp och tillagd massa
- $\mathbf{D}(\boldsymbol{\nu})$  matris för den hydrodynamiska dämpningen
- $\mathbf{g}(\boldsymbol{\eta})$  gravitationskraften i vektorform som funktion av ROV:ens position och attityd

Tröghetsmatrisen  $\mathbf{M}$  kan, enligt Fossen [1], förenklas om ROV:en betraktas som symmetrisk kring xy-, xz- och yz-planen och skrivs:

$$\begin{aligned} \mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A &= \text{diag}[m, m, m, I_x, I_y, I_z] + \text{diag}[X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}] \\ &= \text{diag}[m + X_{\dot{u}}, m + Y_{\dot{v}}, m + Z_{\dot{w}}, I_x + K_{\dot{p}}, I_y + M_{\dot{q}}, I_z + N_{\dot{r}}] \end{aligned} \quad (3.6)$$

Den stela kroppens bidrag till corioliseffekten,  $\mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu}$ , och den tillagda massans bidrag,  $\mathbf{C}_A(\boldsymbol{\nu})$ , kan enligt Fossen [1] skrivas om med hjälp av omskrivningarna  $\mathbf{v} = (u, v, w)^T$ ,  $\boldsymbol{\omega} = (p, q, r)^T$  samt  $\mathbf{S}(s\mathbf{A})\mathbf{B} = a\mathbf{A} \times \mathbf{B}$ . Se Ekvation (3.7) och (3.8).

$$\mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} = \begin{bmatrix} m\mathbf{S}(\boldsymbol{\omega}) & 0_{3 \times 3} \\ 0_{3 \times 3} & -\mathbf{S}(I\boldsymbol{\omega}) \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} m\boldsymbol{\omega} \times \mathbf{v} \\ -(\mathbf{I}\boldsymbol{\omega}) \times \boldsymbol{\omega} \end{bmatrix} \quad (3.7)$$

$$\mathbf{C}_A(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{bmatrix} \quad (3.8)$$

Matrisen som beskriver den hydrauliska dämpningen,  $\mathbf{D}(\boldsymbol{\nu})$ , får enligt Fossen [1] en diagonal form om ROV:en antas ha de tre ortogonala symmetriplanen som nämndes i resonemanget om tröghetsmatrisen och om dess sex frihetsgrader i  $\boldsymbol{\nu}$  antas vara frikopplade. Om andra gradens termer eller högre försummas kan  $\mathbf{D}(\boldsymbol{\nu})$  skrivas:

$$\begin{aligned} \mathbf{D}(\boldsymbol{\nu}) &= -\text{diag}[X_u, Y_u, Z_w, K_p, M_q, N_r] \\ &\quad -\text{diag}[X_{u|u}|u|, Y_{u|u}|u|, Z_{w|w}|w|, K_{p|p}|p|, M_{q|q}|q|, N_{r|r}|r|] \end{aligned} \quad (3.9)$$

Tyngdkraftens inverkan,  $\mathbf{g}(\boldsymbol{\eta})$ , på systemet kan tecknas som i (3.10) om hänsyn tas till att ROV:ens flytpunkt är placerad rakt ovanför dess masscentrum i det kroppsfixa koordinatsystemet.

$$\mathbf{g}(\boldsymbol{\eta}) = \begin{bmatrix} (W - B)s\theta \\ -(W - B)c\theta s\phi \\ -(W - B)c\theta c\phi \\ -z_b B c\theta s\phi \\ -z_b B s\theta \\ 0 \end{bmatrix} = \begin{bmatrix} g(m - \rho V)s\theta \\ -g(m - \rho V)c\theta s\phi \\ -g(m - \rho V)c\theta c\phi \\ -z_b \rho g c\theta s\phi \\ -z_b \rho g s\theta \\ 0 \end{bmatrix} \quad (3.10)$$

Vad gäller thrustrarnas kraftbidrag,  $\boldsymbol{\tau}$ , så har motorernas krafter och moment modellerats som fluidkrafter där kraften är proportionell mot kvadraten på thrusters rotationshastighet. Vidare är thrustrarna placerade så att deras kraftbidrag verkar ortogonalt mot x-axeln vilket resulterar i moment kring y- och z-axeln. Sidothrustrarna är placerade på en axel som går genom ROV:ens geometriska centrum och är parallell med x-axeln. Eftersom masscentrumet befinner sig under ROV:ens geometriska centrum bidrar sidothrustrarna även med ett vridande moment i roll-led. Detta moment antas vara försumbart. Därför kan kraftbidraget från thrustrarna modelleras som i ekvation (3.11).

$$\boldsymbol{\tau} = \begin{bmatrix} C_T u_T |u_T| \\ C_{yr} u_{yr} |u_{yr}| + C_{yf} u_{yf} |u_{yf}| \\ C_{zr} u_{zr} |u_{zr}| + C_{zf} u_{zf} |u_{zf}| \\ C_{M_t} u_T |u_T| \\ C_{zr} u_{zr} |u_{zr}| - C_{zf} u_{zf} |u_{zf}| \\ C_{yr} u_{yr} |u_{yr}| - C_{yf} u_{yf} |u_{yf}| \end{bmatrix} \quad (3.11)$$

Med utgångspunkt från de förenklingar som gjorts i ekvation (3.6), (3.7), (3.8), (3.9), (3.10) samt (3.11) kan ekvation (3.5)  $\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau}$  nu skrivas:

$$\dot{u} = \frac{C_T}{m+X_{\dot{u}}} u_T |u_T| + \frac{X_u + X_u |u|}{m+X_{\dot{u}}} u - \frac{g(m-\rho V)}{m+X_{\dot{u}}} s\theta - \frac{m-Z_{\dot{u}}}{m+X_{\dot{u}}} wq + \frac{m-Y_{\dot{u}}}{m+X_{\dot{u}}} vr \quad (3.12)$$

$$\dot{v} = \frac{C_{yr}}{m+Y_{\dot{v}}} u_{yr} |u_{yr}| + \frac{C_{yf}}{m+Y_{\dot{v}}} u_{yf} |u_{yf}| + \frac{Y_v + Y_v |v|}{m+Y_{\dot{v}}} v + \frac{g(m-\rho V)}{m+Y_{\dot{v}}} c\theta s\phi + \frac{m-Z_{\dot{v}}}{m+Y_{\dot{v}}} wp - \frac{m-X_{\dot{v}}}{m+Y_{\dot{v}}} ur \quad (3.13)$$

$$\dot{w} = \frac{C_{zr}}{m+Z_{\dot{w}}} u_{zr} |u_{zr}| + \frac{C_{zf}}{m+Z_{\dot{w}}} u_{zf} |u_{zf}| + \frac{Z_w + Z_w |w|}{m+Z_{\dot{w}}} w + \frac{g(m-\rho V)}{m+Z_{\dot{w}}} c\theta c\phi + \frac{m-X_{\dot{w}}}{m+Z_{\dot{w}}} uq - \frac{m-Y_{\dot{w}}}{m+Z_{\dot{w}}} vp \quad (3.14)$$

$$\dot{p} = \frac{C_{M_t}}{I_x + K_{\dot{p}}} u_T |u_T| + \frac{K_p + K_p |p|}{I_x + K_{\dot{p}}} p + \frac{z_B \rho V g}{I_x + K_{\dot{p}}} c\theta s\phi - \frac{(I_x - N_r) - (I_y - M_{\dot{q}})}{I_x + K_{\dot{p}}} qr - \frac{Y_{\dot{v}} - Z_{\dot{v}}}{I_x + K_{\dot{p}}} vw \quad (3.15)$$

$$\dot{q} = \frac{C_{zr} x_{zr}}{I_y + M_{\dot{q}}} u_{zr} |u_{zr}| - \frac{C_{zf} x_{zf}}{I_y + M_{\dot{q}}} u_{zf} |u_{zf}| + \frac{M_q + M_q |q|}{I_y + M_{\dot{q}}} q + \frac{z_B \rho V g}{I_y + M_{\dot{q}}} s\theta - \frac{(I_x - K_{\dot{p}}) - (I_x - N_r)}{I_y + M_{\dot{q}}} pr - \frac{Z_{\dot{w}} - X_{\dot{w}}}{I_y + M_{\dot{q}}} uw \quad (3.16)$$

$$\dot{r} = \frac{C_{yr} x_{yr}}{I_z + N_{\dot{r}}} u_{yr} |u_{yr}| - \frac{C_{yf} x_{yf}}{I_z + N_{\dot{r}}} u_{yf} |u_{yf}| + \frac{N_r + N_r |r|}{I_z + N_{\dot{r}}} r - \frac{(I_y - M_{\dot{q}}) - (I_x - K_{\dot{p}})}{I_z + N_{\dot{r}}} pq - \frac{X_{\dot{u}} - Y_{\dot{u}}}{I_z + N_{\dot{r}}} uv \quad (3.17)$$

Ett samband mellan de kroppsfixa och jordfixa hastighetsvektorerna kan nu tecknas som  $\dot{\boldsymbol{\eta}} = \mathbf{J}\boldsymbol{\nu}$  där  $\mathbf{J}$  där  $\mathbf{J}$  är transformationsmatrisen

$$\mathbf{J} = \mathbf{J}(\phi, \theta, \psi) = \begin{bmatrix} \mathbf{R}(\phi, \theta, \psi) & \mathbf{0} \\ \mathbf{0} & \mathbf{T}(\phi, \theta, \psi) \end{bmatrix}, \quad (3.18)$$

alternativt som

$$\mathbf{J} = \mathbf{J}(q_0, q_1, q_2, q_3) = \begin{bmatrix} \mathbf{R}(q_0, q_1, q_2, q_3) & \mathbf{0} \\ \mathbf{0} & \mathbf{T}(q_0, q_1, q_2, q_3) \end{bmatrix}. \quad (3.19)$$

Den slutgiltiga fullständiga modellen tecknas nu nedan med hjälp av omskrivningarna  $x = (\boldsymbol{\eta}^T, \boldsymbol{\eta}^T)$  och  $u = (u_T, u_{yf}, u_{yr}, u_{zf}, u_{zr})^T$ .

$$\dot{x} = f(x, u) = \begin{bmatrix} \mathbf{J}\boldsymbol{\nu} \\ f_b \end{bmatrix} = \begin{bmatrix} \mathbf{R}\boldsymbol{\nu} \\ \mathbf{T}\boldsymbol{w} \\ f_b(\boldsymbol{\nu}, \boldsymbol{\eta}) \end{bmatrix}, \quad (3.20)$$

## 4 Hårdvara

Man har under tidigare års projekt insett att viss hårdvara som har använts i ROV:en inte har haft tillräcklig prestanda för att uppnå samtliga krav som ställts på ROV:en. Placering av sensorerna kan även ha orsakat störningar som påverkar prestandan i en sådan utsträckning att systemets prestanda försämras för mycket. Därför ska hårdvaran under detta projekt modifieras. I detta kapitel beskrivs de hårdvarukomponenter som används eller ska användas i ROV:en i detalj.

### 4.1 Styr- och mätkort

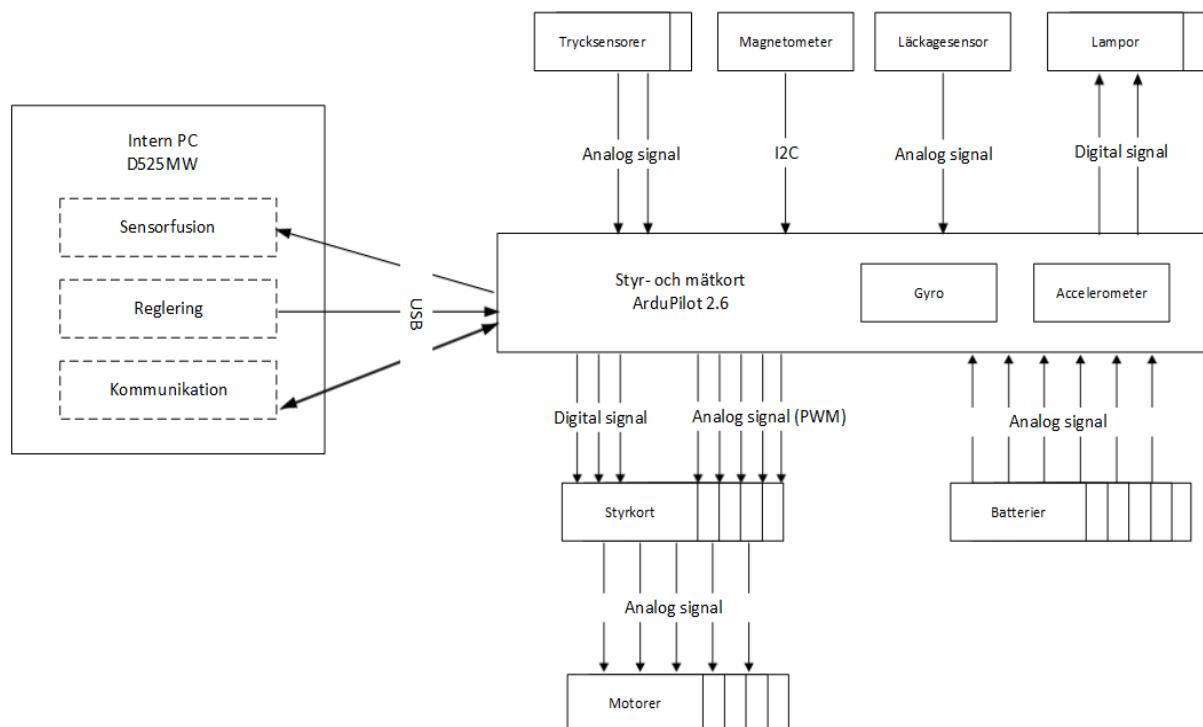
Tidigare år har en Arduino Mega2560 med en ATmega2560 mikrokontroller använts som styr- och mätkort men på grund av överbelastning ska kortet bytas ut [3]. Istället ska en Ardupilot APM 2.6 från 3D Robotics användas som styr- och mätkort. I det nya kortet finns en inbyggd IMU, vilket gör att den externa IMU:n inte längre behövs. Magnetometern som var integrerad med den gamla IMU:n ska istället ersättas med en extern magnetometer. På så vis kan magnetometern placeras längre ifrån störningskällor och därmed ge mer tillförlitliga mätningar.

Det är styr- och mätkortets uppgift att samla in sensordata från den externa magnetometern, trycksensorerna, batterispänningarna, läckagesensorn samt den interna accelerometern och gyrot. All sensordata förutom batterispänningarna och läckagesensorn skickas därefter vidare till delsystemet Sensorfusion. Sensordatan från batterispänningarna och läckagesensorn behöver inte behandlas av delsystemet Sensorfusion utan ska istället skickas till delsystemet Kommunikation för att sedan skickas vidare till GUI:t på den externa PC:n. Gyrot och accelerometern mäter vinkelhastigheter kring varje rotationsaxel och acceleration i  $x$ -,  $y$ - och  $z$ -led.

Styr- och mätkortet erhåller regelbundet styrsignaler för ROV:ens motorer från delsystemet Reglering vilka sedan görs om till en analog signal med PWM som skickas till respektive motors styrkort. Ty varje motor har var sitt individuellt anpassat styrkort som har i uppgift att omvandla den analoga PWM signalen som fås från ROV:ens styr- och mätkort till en högre spänning för att driva motorerna. Hur hög spänning som skickas till motorerna beror på pulsbredden i den analoga PWM signalen. Det sitter även tre reläer för att strömföra ROV:ens motorer och dessa styrs med tre digitala signaler från styr- och mätkortet.

Eftersom det ska vara möjligt att från GUI:t på den externa datorn kunna tända och släcka ROV:ens två lampor krävs även att styr- och mätkortet ska kunna ta emot styrsignaler från delsystemet Kommunikation. Styr- och mätkortet gör sedan om dessa till en digital signal som skickas till lampan för att tända eller släcka den. En översikt över vilka komponenter som är inkopplade till styr- och mätkortet kan ses i Figur 3. Både det nya och det gamla styr- och mätkortet bygger på öppen mjukvara från Arduino.

Därför kommer den tidigare mjukvaran att kunna implementeras till det nya styr- och mätkortet med några mindre justeringar.



Figur 3: En översikt över kopplingsschemat till styr- och mätkortet samt hur styr- och mätkortet kommunicerar med den interna PC:n. Heldragna rektanglar representerar hårdvara och streckade rektanglar representerar mjukvaran i den interna PC:n. Styr- och mätkortet kommunicerar med den interna PC:n via USB, med magnetometern via I2C, för resterande komponenter sker kommunikationen med digitala och analoga signaler. De digitala signalerna som skickas till motorernas styrkort styr reläerna som strömsätter motorerna.

Sensor	Signal
Trycksensorer	Analog signal
Magnetometer	I2C
Läckagesensor	Analog signal
Lampor	Digital signal
Styrkort (Reläer)	Digital signal
Styrkort (styrsignaler)	Analog signal (PWM)
Batterier	Analog signal
Intern PC	USB

Tabell 3: Tabell över signalerna mellan sensorer, aktuatorer och ROV:ens styr- och mätkort.



## 4.2 Magnetometer

I tidigare projekt har det funnits en magnetometer som varit en del av IMU:n vilket gjort att valmöjligheterna för placeringen av denna varit begränsade. Detta är ett problem då magnetometern är känslig för störningar från ROV:ens elektronik [3]. I detta projekt ska dessa störningar utvärderas och magnetometern placeras för att minimera påverkan av dessa. För att få större flexibilitet i denna placering ska en ny, extern magnetometer användas och integreras med det övriga systemet. Denna är en HMC5883L - Triple Axis Magnetometer från 3D Robotics.

Utvärdering av störningar på magnetometern gjordes även under projektet 2013 och den tekniska rapporten från detta år visar att den placering som gav minst störningar på magnetometern var 4 cm ovanför ROV:en längst fram. Den placering som gav näst minst störningar var då magnetometern placerades längst fram i ROV:en [3].

De potentiella störningar som påverkar magnetometern kommer främst från säkerhetsmagneten, batteripaketet, kablage och motorer. Säkerhetsmagneten sitter i bakre delen av ROV:en medan det mesta av kablaget och batterierna sitter i nedre delen av ROV:en långt fram. Tidigare års resultat tyder alltså på att säkerhetsmagneten skapar stora störningar på magnetometern som bör placeras långt från denna.

För att minimera störningarna från kablaget ska alla sladdar hållas så korta som möjligt, eftersom magnetfältets styrka växer med strömmens omcirklade area enligt Biot-Savarts lag. Av samma anledning kan man minska den omcirklade arean genom att tvinna samman sladdarna. På så vis hålls även återströmmen nära strömmens väg vilket även det minskar den omcirklade arean. Eftersom ny hårdvara ska integreras i detta projekt finns stora möjligheter att även se över kablaget och ta hänsyn till detta i integreringen av den nya hårdvaran för att minimera störningarna på magnetometern. Det kablage som ger upphov till störst störningar är mellan styrkort och motorer, men detta är dock redan monterat och möjligheten att tvinna detta ska undersökas.

## 4.3 Trycksensorer

Vid projektets start sitter en trycksensor, från serien PX2 tillverkad av Honeywell, monterad på ROV:en som mäter tryck mellan 0 kPa och 689 kPa, vilket motsvarar vattendjup på upp till 59 m. För att uppnå en bättre upplösning i mätningarna kommer denna bytas ut mot en trycksensor vars maximala utsignal svarar mot 250 kPa, vilket innebär 15 m djup. Eventuellt kommer ytterligare en trycksensor att integreras på en annan del av ROV:en för att ge ytterligare bättre noggrannhet samt möjlighet till skattning av ROV:ens pitchvinkel. Möjligheten att sätta in en extra trycksensor kommer diskuteras med gruppen i det parallella maskinteknikprojektet. De två största anledningarna till att det kanske inte är möjligt att integrera den extra trycksensorn är tidsbrist för gruppen i maskinteknikprojektet och platsbrist/brist på ställen att fästa sensorn på i ROV:en.

Ifall ytterligare en trycksensor integreras på ROV:en kommer detta att ske i fören på ROV:en. Eftersom det redan finns ett hål i fören som är oanvänt så är det högst troligt att detta kommer att användas för att slippa borra nya hål.

Trycksensorn kommer att omvandla trycket till en spänning som mäts av en A/D-omvandlare på styr- och mätkortet. Denna har en upplösning på 10 bitar, och därmed kan trycksensorns mätintervall delas upp i 1024 mätsteg.

Med den befintliga trycksensorn har en referensspänning på 5V använts i A/D-omvandlaren, och trycksensorn ger 0.5 V vid 0 kPa och 4,5 V vid 689 kPa. Eftersom trycket vid ytan är ca 100 kPa kommer intervallet att bli 100-689 kPa vilket motsvarar ca 1-4,5 V. Detta gav en upplösning på 844 Pa per bit. Dock vore det här fördelaktigt att sänka referensspänningen på A/D-omvandlaren åtminstone till 4,5V

då sensorn inte ger mer än 4,5 V. Det skulle även gå sänka referensspänningen ytterligare för att ge bättre upplösning men då minskar arbetsområdet.

Med antagande om att samma matningsspänning och spänningsintervall skulle gälla på en ny trycksensor med mätområde 0-250 kPa och med A/D-omvandlarens referensspänning på 5V skulle upplösningen istället bli 305 Pa, vilket motsvarar ungefär 3.1 cm. Detta att jämföra med 844 Pa på den nuvarande trycksensorn som motsvarar ungefär 8.6 cm.

Det tryck som sensorn mäter är totaltrycket, som består av statiskt och dynamiskt tryck. Det statiska trycket är givet för varje vattendjup och det är därför det som man vill mäta. Enligt Bernoullis ekvation gäller

$$p_{tot} = p_{stat} + p_{dyn} = (p + \rho gh) + \frac{\rho}{2}v^2 \quad (4.1)$$

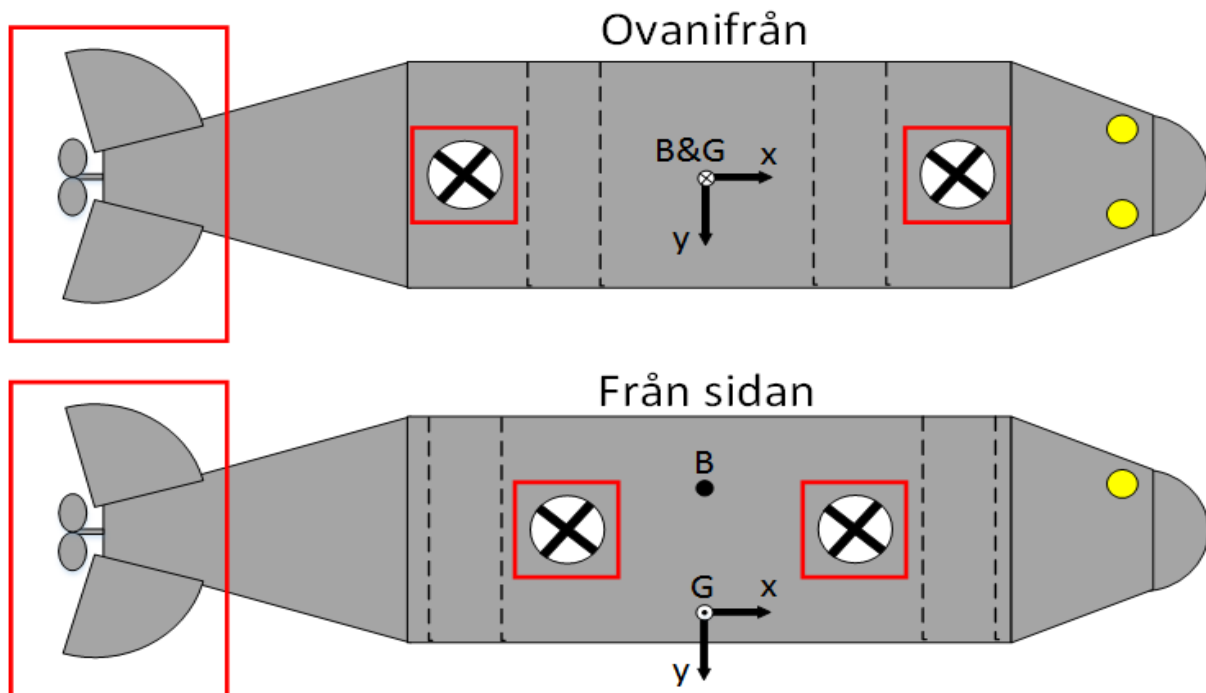
där  $p$  är lufttrycket vid vattenytan vilket kalibreras innan användning,  $\rho$  är vattnets densitet (998 kg/m<sup>3</sup> vid 20 °C),  $g$  tyngdaccelerationen (9,81 m/s<sup>2</sup>),  $h$  djupet och  $v$  vattenströmningens hastighet vid trycksensorn. I (4.1) ses att det dynamiska trycket kan utgöra en betydande del av totaltrycket vid höga hastigheter och/eller grunda djup.

Ett exempel för ROV:en skulle kunna vara att den ligger och kör på 1 m djup och med en strömningshastighet av 2 m/s. Då kommer det totala trycket att vara 111,590 kPa. Det dynamiska trycket står för cirka 2,0 kPa och det statiska djupberoende trycket står för cirka 9,8 kPa av det totala trycket. Detta ger cirka 20 % fel i djupskattningen. På grund av detta så kommer en lösning för att avskärma trycksensorn från stora strömningar att ses över.

## 4.4 Motorer

ROV:en har fem motorer varav fyra av dessa driver varsin thrusterpropeller och den femte motorn driver bakpropellern. Motorernas placering på ROV:en kan ses i Figur 4. Huvudmotorn sitter längst bak och styr ROV:ens translation i  $x$ -led. Två mindre motorer styr ROV:en i  $y$ -led och två motorer hanterar rörelser i  $z$ -led. Med dessa motorer och propellrar erhålls då fem frihetsgrader: rörelser i  $x$ -,  $y$ - och  $z$ -led samt rotationer i yaw och pitch. Rollvinkeln är den frihetsgrad som inte är möjlig att styra eftersom thrusterpropellrarnas kraftvektorer skär  $x$ -axeln vilket medför att det kan inte skapas något moment kring  $x$ -axeln.

De olika motorerna kräver olika mycket ström. Därför är varje motor kopplad till ett individuellt anpassat styrkort, vars uppgift är att översätta styrsignalen från styr- och mätkortet till en pulsbreddsmodulerad signal för respektive motor.



Figur 4: En översikt över motorernas placering på ROV:en. Motorernas placering visas med röd rektangel. Överst i figuren kan ROV:en ses ovanifrån och underst syns ROV:en från sidan. ROV:ens flytkraft påverkar ROV:en i punkt B och ROV:ens tyngdpunkt anges i punkten G. Observera att figuren inte är skalening.

#### 4.5 Inertial Measurement Unit

IMU:n, av modell MPU-6000, mäter vinkelhastigheter kring varje rotationsaxel och acceleration i  $x$ -,  $y$ - och  $z$ -led. Till denna hör även en extern magnetometer. Dessa mätningar utnyttjas sedan för att skatta ROV:ens orientering och vinkelhastighet.

#### 4.6 Intern PC

Den interna PC:n är placerad inuti ROV:en och i denna är delsystemen Kommunikation, Referensgränssnitt, Reglering och Sensorfusion implementerade. Hårdvaran i den interna PC:n presenteras nedan.

- Typ: Intel Desktop Board D525MW.
- Processor: Intel Atom processor D525.
- Internminne: 2x2 GB Kingston DDR3 PC3-8500 1066 MHz.
- Hårddisk: OCZ 60 GB vertex II E-series.

## 4.7 Övrig hårdvara

Utöver ovan nämnda hårdvara finns även följande hårdvara på ROV:en:

- En webbkamera som är placerad framtill på ROV:en och som ansluts via USB till den externa PC:n.
- Lampor som styrs av styr- och mätkortet och sitter framtill på ROV:en. Deras syfte är att lysa upp för webbkameran.
- 6 st batteripaket på 24 volt som spänningssätter hela ROV:en.
- En läckagesensor för att indikera eventuellt läckage in i ROV:ens hårdvarubehållare.
- En säkerhetsmagnet som bryter strömmen till ROV:en. Denna ska användas som nödstopp ifall något går fel.

## 5 Mjukvara

Den mjukvara som hanterar kommunikation och reglering ligger på den interna PC:n. På den här PC:n körs i grunden ett Linuxsystem och över detta körs delsystemen i operativsystemet ROS (Robot Operating System). En mer grundlig beskrivning av ROS finns i Kapitel 5.1 och en utförlig beskrivning av de olika delsystemen finns i Kapitel 6, 7, 8, 9 och 10.

### 5.1 Robot Operating System

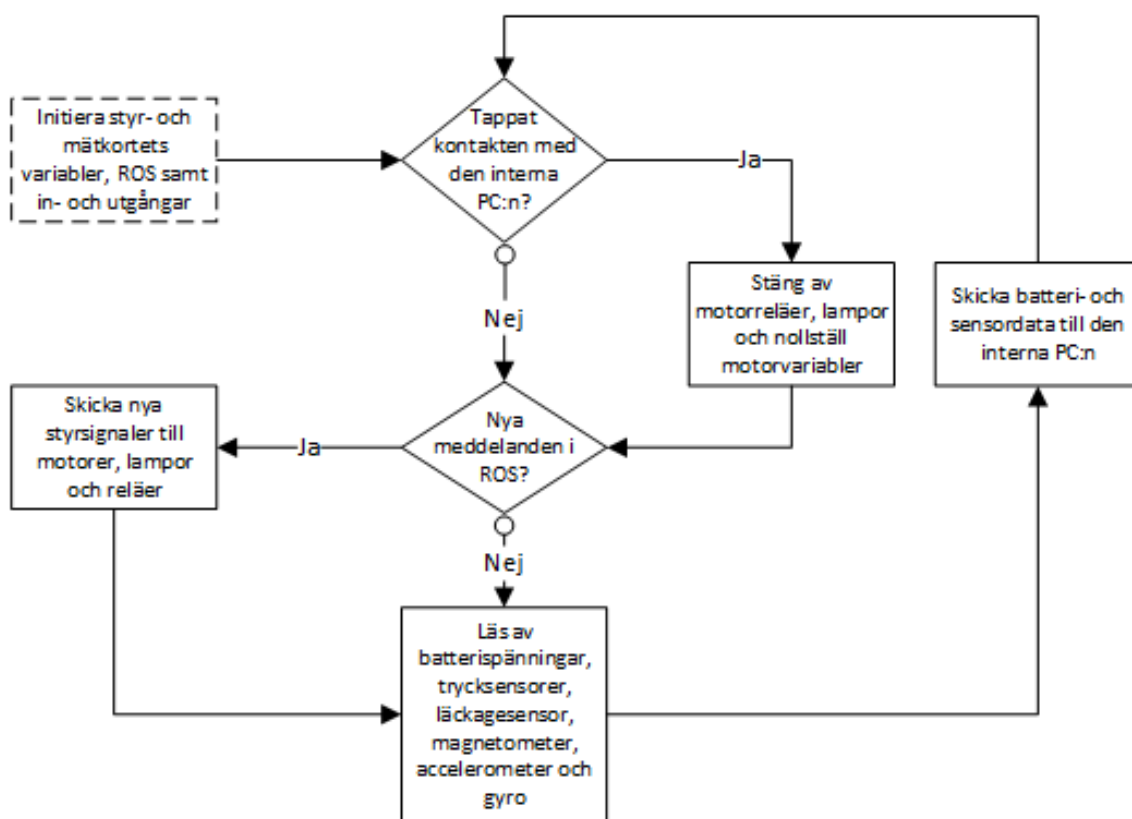
ROS används för hårdvaruabstraktion och kommunikation mellan de olika processerna i systemet. Processerna kan köras parallellt och kallas för noder. Kommunikationen sker genom så kallade "topics" där varje nod kan vara "publisher" eller "subscriber" till denna topic. De noder som ska ta emot informationen är alltså subscribers och de som sänder information är publishers. Då en nod ska skicka ett meddelande till en annan nod läggs meddelandet i buffern för varje subscriber. Dessa subscribers kan sedan välja själv när de vill läsa av det nya meddelandet. För att skicka ett meddelande krävs ett topic som beskriver meddelandets struktur gällande heltal, flyttal etc. Topicsen fungerar sedan som en buss för att överföra det specifika meddelandet till andra processer. För att se vilka topics och meddelanden som kan sändas, se Appendix A.

### 5.2 Styr- och mätkort

ROV:ens styr- och mätkort bygger på en öppen mjukvara från Arduino. Mjukvaran för en arduino är uppbyggt på ett C/C++ bibliotek tillsammans med ett bibliotek för att interagera med hårdvaran. Fördelen med detta är att man på ett enkelt sätt kan skriva och läsa av de in- och utgångar som finns på styr- och mätkortet. Koden till styr- och mätkortet är därför förhållandevis simpel och ett flödesschema för denna kan ses i Figur 5. Till en början initieras styr- och mätkortets in- och utgångar. När in- och utgångarna har konfigurerats definieras ett antal variabler och ROV:en sätts i tomgång. Därefter initieras ROS så att styr- och mätkortet kan skicka och ta emot meddelanden från andra noder i systemet. Styr- och mätkortet väntar sedan på meddelanden från den interna PC:n. Då styr- och mätkortet får ett meddelande från den interna PC:n så skickar styr- och mätkortet vidare styrsignalerna på digital eller analog form till motorer, lampor och reläer. Därefter läses alla sensorer av, dvs. batterispänningar, trycksensorer, läckagesensorer, magnetometer, accelerometer samt gyro. Sensordata och batterispänningar skickas sedan till den interna PC:n i två ROS topics. När meddelandena skickats kontrolleras ifall styr- och mätkortet tagit emot något meddelande de senaste 20 looparna. Om inget meddelande tagits emot de senaste

20 looparna så har kommunikationen mellan interna PC:n och styr- och mätkortet förlorats. Styr- och mätkortet stänger då av motorreläerna, lamporna och nollställer motorvariablerna.

Motorerna styrs enligt Kapitel 4.1 och pulsbredden för den analoga PWM signalen kan programmeringsmässigt väljas i intervallet 0-255 där 0 representerar att signalen alltid är låg och 255 representerar att signalen alltid är hög. Då pulsbredden väljs till värdet 127 står motorerna stilla. Värdet över 127 motsvarar att motorerna körs framåt och vice versa.



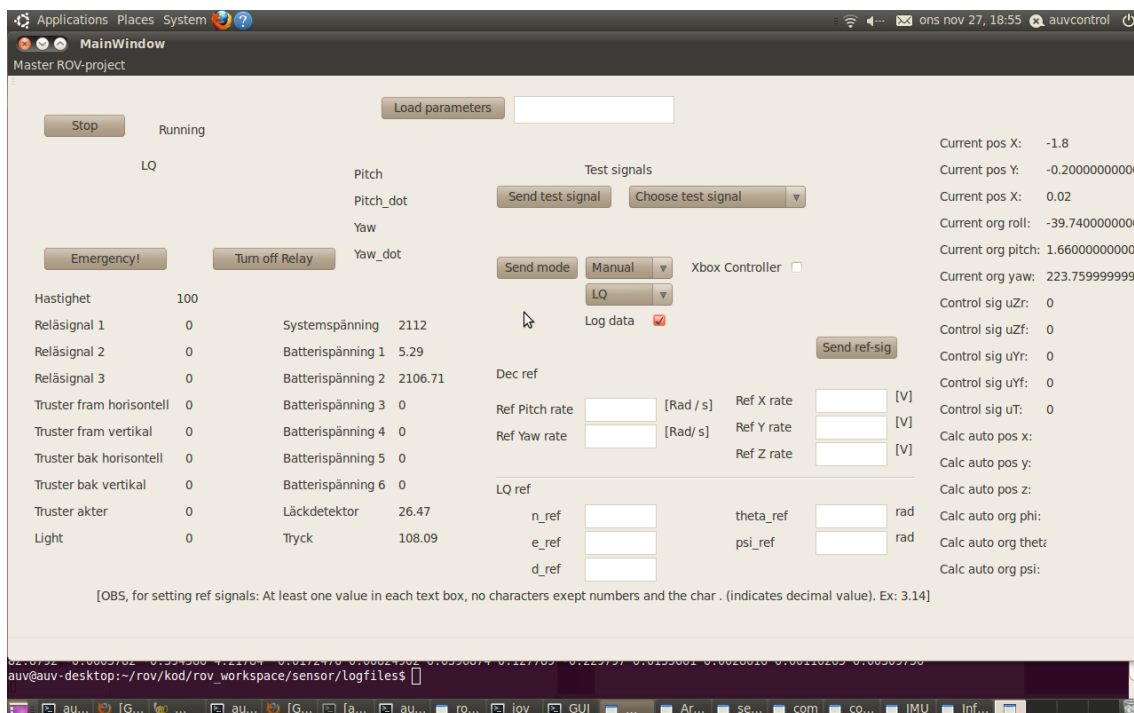
Figur 5: Ett flödesschema över kodstrukturen för ROV:ens styr- och mätkort. I figuren representerar rektanglar det som ska utföras och romber de beslut som ska tas. Pilarna indikerar händelseförloppet och texten "Ja" eller "Nej" förtydligar utfallet från romberna.

### 5.3 Grafiskt gränssnitt

På den externa datorn finns ett grafiskt gränssnitt (GUI) för att enkelt kunna sköta och övervaka kommunikationen med ROV:en. En översikt över det grafiska gränssnittet kan ses i Figur 6. Det är i detta GUI som användaren kan välja vilket mode ROV:en ska köras i, manuellt eller stabiliserat, och vilken regulator som ska användas. Användaren kan också sätta referensvärden och övervaka ROV:ens status via gränssnittet. Den information kring ROV:ens status som kan avläsas i GUI:t är position, orientering, batterispänning och larm. Position, orientering och batterispänning kan avläsas i det ordinarie fönstret medan ett nytt fönster öppnas vid larm.

GUI:t är skrivet i Qt och finns definierat i en xml-fil. Under detta projekt kommer det grafiska gränssnittet att utvecklas för att kunna hantera två farkoster. Det kommer skapas en andra flik för den nya ROV:en.

Den nya ROV:en ska utvecklas av den parallella projektgruppen på M-programmet. Denna flik kommer innehålla samma funktionalitet för reglering och övervakning som den som redan finns i det existerande GUI:t. I framtiden kan man dock tänka sig att man vill ha andra funktioner i den nya fliken. Det är dock inget som detta projekt ska tänka på eftersom den nya ROV:en inte kommer att vara färdigutvecklad innan projektets slut.



Figur 6: En skärmdump av befintligt grafiskt gränssnitt som i realtid visar mätdata och styrsignaler från styr- och mätkortet. GUI:t ger även möjlighet för användaren att interagera med ROV:en vad gäller nödstopp, val av regulator och referensvärden. Bilden är hämtad från föregående års projekt [8].

## 6 Delsystem Sensorfusion

Delsystemet Sensorfusion har till uppgift att skatta ROV:ens djup och orientering med hjälp av ett antal olika sensorer. Utöver detta ska även tillstånden för vinkelhastighet i pitch-, roll- och yawed skattas.

### 6.1 Kommunikation med andra delsystem

Delsystemet Sensorfusion kommer att hämta sensordata från styr- och mätkortets interna IMU. Även data från magnetometern och trycksensorerna hämtas via styr- och mätkortet. Med denna data utför delsystemet Sensorfusion skattningar av ROV:ens orientering och djup och skickar vidare dessa till delsystemen Reglering och Kommunikation. En beskrivning av signalerna som skickas till och från delsystemet Sensorfusion kan ses i Tabell 4 och 5.

Mottagare	Signal	Enhet
Kommunikation, Reglering	Skattade värden av pitch, yaw och roll.	°
Kommunikation, Reglering	Skattade värden av vinkelhastigheten i pitch-, yaw- och rolled.	°/s
Kommunikation, Reglering	Skattat värde av djup.	m

Tabell 4: Tabell över vad som skickas från delsystemet Sensorfusion.

Avsändare	Signal	Enhet
Styr- och mätkort	Tryck	V
Styr- och mätkort	Rådata från magnetometer.	°
Styr- och mätkort	Rådata från IMU-signaler.	°/s och °/s <sup>2</sup>

Tabell 5: Tabell över vad som mottages i delsystemet Sensorfusion.

## 6.2 Observatör

Tidigare år har man en använt EKF-observatör för att skatta ROV:ens orientering. Observatören är implementerad som en EKF1, d.v.s. att den olinjära tillståndsekvationen utvecklats med första ordningens taylorutveckling.

Ur den olinjära modellen

$$x_{k+1} = f(x_k, u_k) + v_k, \quad \text{där } v_k \sim N(0, Q_k) \quad (6.1a)$$

$$y_k = h(x_k, u_k) + e_k, \quad \text{där } e_k \sim N(0, R_k) \quad (6.1b)$$

fås algoritmen för EKF1 enligt

$$S_k = R_k + h'(\hat{x}_{k|k-1})P_{k|k-1}(h'(\hat{x}_{k|k-1}))^T \quad (6.2a)$$

$$K_k = P_{k|k-1}(h'(\hat{x}_{k|k-1}))^T S_k^{-1} \quad (6.2b)$$

$$\epsilon_k = y_k - h'(\hat{x}_{k|k-1}) \quad (6.2c)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \epsilon_k \quad (6.2d)$$

$$P_{k|k} = P_{k|k-1} - P_{k|k-1}(h'(\hat{x}_{k|k-1}))^T S_k^{-1} h'(\hat{x}_{k|k-1}) P_{k|k-1} \quad (6.2e)$$

$$\hat{x}_{k+1|k} = f(\hat{x}_{k|k}) \quad (6.2f)$$

$$P_{k+1|k} = Q_k + f'(\hat{x}_{k|k})P_{k|k}(f'(\hat{x}_{k|k}))^T \quad (6.2g)$$

Tillståndsmodellen  $f(x_k, u_k)$  representeras av (3.20). Alla trigonometriska funktioner försummas i derivatan av tillstånden. Då ROV:en körs på land sätts  $f_b(\boldsymbol{\nu}, \boldsymbol{\eta})$  i (3.20) = 0. Därutöver gäller accelerometer-dödräkningsmodellen enligt Singer-approximationen

$$f_b(\boldsymbol{\nu}, \boldsymbol{\eta}) = \begin{bmatrix} y_a \\ 0 \end{bmatrix} - \beta \boldsymbol{\nu} \quad (6.3)$$

i vilken  $\beta$  är en designparameter. [3]

Denna EKF som togs fram förra året kommer användas även detta år då den fungerade bra.

### 6.2.1 Mättekvationer trycksensor

Det kommer att krävas mättekvationer från respektive sensor vilka kommer att behandlas nedan. Varje sensors mättekvation har additivt brus och bruset antas okorrelerat med andra sensors brus. Från ROV:ens trycksensorer erhålls en spänning som tolkas som ett absolut tryck. Utifrån det absoluta trycket är man intresserad av att beräkna ROV:ens djup vilket kan beräknas enligt (6.4).

$$p_{tot} = p_{stat} + p_{dyn} = (p + \rho gh) + v_d \Rightarrow h = \frac{p_{tot} - p - v_d}{\rho g} \quad (6.4)$$

där  $p$  är lufttrycket vid vattenytan vilket kalibreras innan användning,  $\rho$  är vattnets densitet ( $998 \text{ kg/m}^3$  vid  $20 \text{ }^\circ\text{C}$ ),  $g$  tyngdaccelerationen ( $9,81 \text{ m/s}^2$ ),  $h$  djupet och  $v_d$  är sensorbrus. Enligt föregående års projekt [3] behövs även en konversionsfaktor  $\alpha_d$  för att korrigera felaktig skalning i A/D-omvandlaren men eftersom att ROV:en detta år har en ny trycksensor och A/D-omvandlare behövs inte denna konversionsfaktor längre. Eftersom trycksensorn inte är placerad i ROV:ens geometriska centrum måste man även kompensera för ROV:ens vinkel i pitchled genom att minska djupet med  $l \sin \theta$ , där  $l$  är avståndet från trycksensorn till ROV:ens geometriska centrum längs  $x$ -axeln och  $\theta$  är pitchvinkeln. Mättekvationen för en trycksensor på befintlig position (bak på ROV:ens trycksäkra sektion) följer då enligt (6.5).

$$h = \alpha_d \frac{p_{tot} - p - v_d}{\rho g} - l \sin \theta \quad (6.5)$$

Som beskrivs i Kapitel 4.3 så finns ett borrarat hål för trycksensorn i ROV:ens för. Det hålet ligger dock lite längre ner än ROV:ens geometriska centrum och man måste därför kompensera för detta genom att lägga till en term  $\beta_d$  som beskriver vinkeln mellan  $x$ -axeln och hypotenusan från trycksensorn till ROV:ens geometriska centrum. Detta för att kunna skatta ROV:ens geometriska centrum istället för trycksensorns position.

$$h = \alpha_d \frac{p_{tot} - p - v_d}{\rho g} - l \sin(\theta - \beta_d) \quad (6.6)$$

### 6.2.2 Mättekvationer accelerometer

På det nya styrkortet är IMU:n integrerad med styr- och mätkortet. Mättekvationerna kommer se ut på samma sätt som i tidigare års projekt [3] enligt (6.7).

$$\mathbf{a} = \mathbf{R}^T(\mathbf{q})(\mathbf{a}_{kropp} + g\hat{\mathbf{d}}) + \mathbf{v}_a \quad (6.7)$$

Man vill använda denna mättekvation för små värden på  $\mathbf{a}_{kropp}$ . Då mäts enbart gravitationen och man förkastar då mätvärden:

$$\|\mathbf{a}\| - g > \epsilon_a \quad (6.8)$$



där  $\epsilon_a$  är designparameter,  $\mathbf{R}(\mathbf{q})$  är en rotationsmatris (3.2) från globala koordinater till kroppsfixa,  $\mathbf{a}_{kropp}$  är den kroppsfixa accelerationen och  $\mathbf{a}$  är acceleration i det globala koordinatsystemet.  $\mathbf{v}_a$  är mätstörningar och  $g\hat{\mathbf{d}}$  svarar mot tyngdaccelerationen.

### 6.2.3 Magnetometer

Det nya mätkortet använder en ny extern magnetometer som kan mäta magnetfältstyrkan i tre ortogonala riktningar. Mättekvationen för magnetometern följer enligt (6.9) från föregående års projekt [3]

$$\mathbf{m} = \mathbf{R}(\mathbf{q})\mathbf{y}_m \quad (6.9a)$$

$$0 = m_2 + v_m \quad (6.9b)$$

där  $\mathbf{q}$  är kvaternionerna,  $\mathbf{R}$  är rotationsmatrisen enligt (3.2),  $\mathbf{y}_m$  är det uppmätta magnetfältet i det kroppfixa koordinatsystemet och  $\mathbf{m}$  är det uppmätta magnetfältet i det globala koordinatsystemet. Variabeln  $m_2$  är det andra elementet i vektorn  $\mathbf{m}$  d.v.s. magnetstyrkan i östlig riktning och  $v_m$  är mätbrus. Eftersom magnetfältet är konstant i det globala koordinatsystemet men man saknar information om initialläget hos farkosten måste man sätta ett av  $\mathbf{m}$ -matrisens element till 0 för att fixera det globala koordinatsystemet och skatta orienteringen korrekt. I detta fall har man valt att sätta öst-komponenten till 0 då detta orsakar mindre störningar än då man sätter nord-komponenten till 0. Det är ekvation (6.9b) som driver magnetfältets öst-komponent mot noll och  $v_m$  anger hur mycket man lutar på denna mättekvation.

Man kan få ett skattningsfel på  $180^\circ$  eftersom magnetfältet kan vara noll både om ROV:en är riktad mot öst eller väst. Detta har från föregående års projekt lösts genom att modifiera  $\mathbf{q}$  då  $m_1 < -0.1$ .

Då elektronik i ROV:en kan ge upphov till störningar så används inte mättekvationer då:

$$||\mathbf{m}| - b_{jord}| > \epsilon_m \quad (6.10)$$

där  $\epsilon_m$  är en designparameter och  $b_{jord}$  är jordens magnetfält.

### 6.2.4 Gyro

På det nya styr- och mätkortet finns ett gyro integrerat som mäter ROV:ens kroppsfixa vinkelhastigheter. Mättekvationen för gyrot följer enligt (6.11).

$$\boldsymbol{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \mathbf{v}_g \quad (6.11)$$

Där  $\mathbf{v}_g$  är mätbruset samt  $p, q$  och  $r$  är ROV:ens kroppsfixa vinkelhastigheter runt axlarna  $x, y$  och  $z$ .

## 7 Delsystem Reglering

Reglersystemet på ROV:en finns implementerat i mjukvara på den interna PC:n. ROV:en har två autonoma lägen: ett för användning av en decentraliserad regulator och ett för en LQ-regulator. Dessa två lägen kan väljas från den externa PC:n. I båda lägena kan en Xbox-kontroll användas för att ange referenser

alternativt kan referensvärden sättas i GUI:t på den externa PC:n. Med den decentraliserade regulatorn kan man styra vinkelhastigheterna och huvudmotorn hos ROV:en, medan LQ-regulatorn styr djup, orientering och huvudmotorn. Båda dessa regulatorer finns implementerade i styrsystemet men behöver förbättras för att få en mer pålitlig och robust reglering. Regulatorprestandan för de nya regulatorerna ska utvärderas.

## 7.1 Kommunikation med andra delsystem

Från delsystemet Referensgränssnitt ska delsystemet Reglering erhålla information om vilket reglerläge (manuellt eller stabiliserat) som ska användas, vilken regulator som ska användas, om ROV:en ska vara av- eller påslagen och ifall nödstopp ska initieras. Anledningen att alla signaler (även de som icke är referenser) går via referensgränssnittet är att man vill ha ROV:ens mjukvara så modulbaserad som möjligt. För att kunna reglera ROV:en behöver dock delsystemet Reglering även ta emot referensvärden och mätsignaler. Referensvärdena erhålls från referensgränssnittet. Dessutom fås mätsignaler i form av skattningar av ROV:ens orientering och djup från delsystemet Sensorfusion. Därefter beräknar delsystemet Reglering utsignaler som sedan skickas ut till motorerna via styr- och mätkortet. De signaler som skickas till och från delsystemet Reglering ses i Tabell 6 och 7.

Mottagare	Signal	Enhet
Kommunikation, Styr- och mätkort	Styrsignaler till motorerna.	Styrkod

Tabell 6: Tabell över vad som skickas från delsystemet Reglering.

Avsändare	Signal	Enhet
Referensgränssnitt	Nödläge (Då kontakten med den externa PC:n bryts).	–
Referensgränssnitt	Referensvärden för vinkelhastighet i pitch- och yawled (då decentraliserade regulatorn används).	$^{\circ}/s$
Referensgränssnitt	Referensvärden för vinklar i pitch- och yawled (då LQ-regulatorn används).	$^{\circ}$
Referensgränssnitt	Referensvärden för $z$ -led (då LQ-regulatorn används).	$m$
Referensgränssnitt	Hastigheter i $x$ -, $y$ - samt $z$ -led (då den decentraliserade regulatorn används).	<i>styrkod</i>
Referensgränssnitt	Hastigheter i $x$ samt $y$ -led (då LQ-regulatorn används).	<i>styrkod</i>
Referensgränssnitt	Om ROV:en ska operera i manuellt eller autonomt läge.	–
Referensgränssnitt	Om ROV:en ska använda decentraliserad eller LQ-regulator då ROV:en opererar i autonomt läge.	–

Tabell 7: Tabell över vad som mottages i delsystemet Reglering.

## 7.2 Decentraliserad regulator

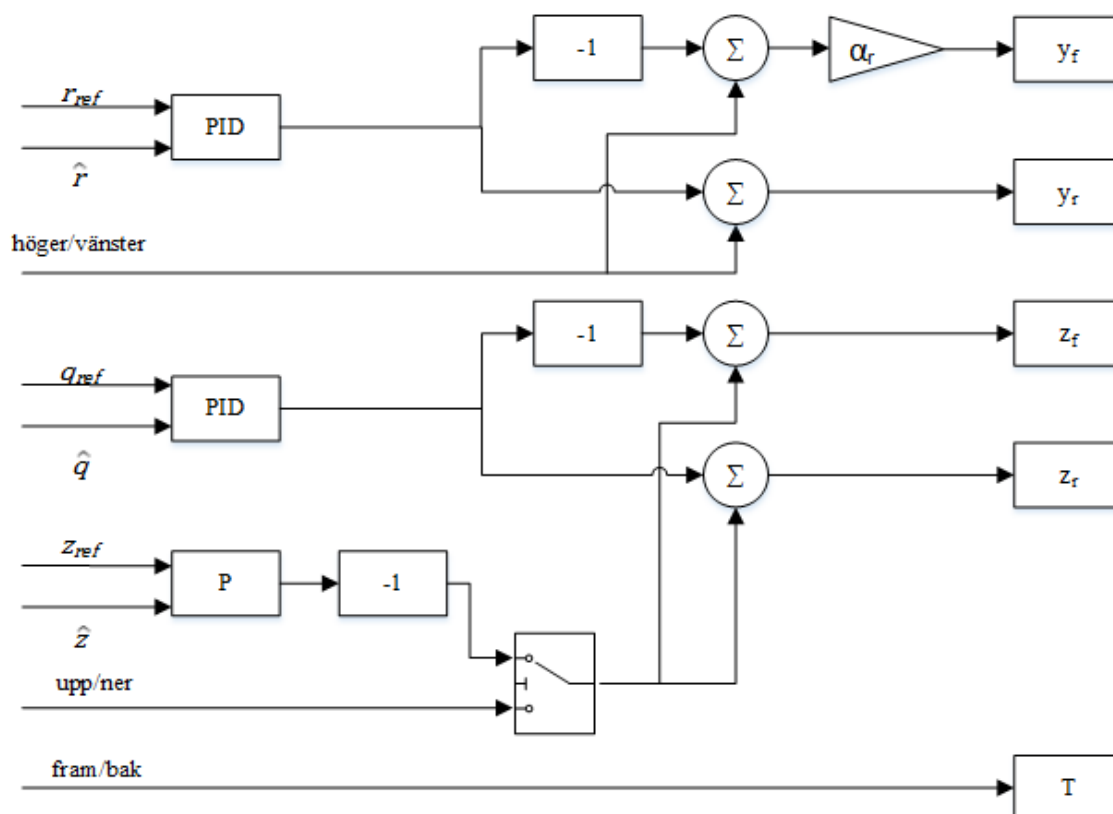
Instället för att behöva styra alla motorer var för sig så kan en decentraliserad regulator användas för att styra ROV:en. En sådan regulator innebär att man styr varje utsignal med en enskild insignal. Regulatorn som förra årets projektgrupp skapade består av två PID-regulatorer samt en P-regulator. De båda PID-regulatorerna används för att reglera ROV:ens skattade vinkelhastigheter efter referensvinkelhastigheter

i yaw- och pitchled. Regulatorns utsignal summeras med önskad translationsrörelse och den resulterande signalen skickas vidare till motorerna i  $y$ - och  $z$ -led. Variabeln  $y_f$  är signalen till den främre motorn i  $y$ -led och  $y_r$  är signalen till den bakre.  $z_f$  och  $z_r$  är motsvarande signaler i  $z$ -led. P-regulatorn används för att reglera ROV:ens djup. Eftersom att det ändå ska vara möjligt att styra ROV:en uppåt och neråt används en switch som stänger av P-regulatorn då operatören önskar stiga eller sjunka i djupled. Då operatören önskar att translatera ROV:en i  $x$ -,  $y$ -, eller  $z$ -led skickas styrsignaler direkt in i ROV:en. På grund av att motorerna har olika avstånd till rotationscentrum lades en skalfaktor  $\alpha_r$  till för att undvika drift. Föregående års projekt använde  $\alpha_r = 0,95$ .

Föregående års projektgrupp använde position i  $z$ -led, det vill säga  $z_{ref}$  och  $\hat{z}$ , som insignaler till P-regulatorn. Man kan dock använda  $\dot{z}_{ref}$  och  $\dot{\hat{z}}$  som insignaler istället och huruvida denna förändring förbättrar den decentraliserade regulatorns körkänsla eller inte ska undersökas under detta projekt.

Om ROV:en ska ligga på en given position och samtidigt utföra ett steg i pitchled kommer ROV:en troligtvis börja driva i  $x$ -led om inte huvudmotorn som styr bakpropellern används för att bibehålla positionen under rörelsen. Om detta beteende bekräftas vid testkörning av ROV:en ska problemet undersökas närmare.

Regulatorparametrarna  $K_p$ ,  $K_i$  och  $K_d$  för PID-regulatorerna och P-regulatorn ska utförligt utvärderas och trimmas för att öka ROV:ens prestanda. Även precisionen och tillförlitligheten för  $\alpha_r$  ska förbättras under projektets gång. Detta ska ske genom testning av regulatorn för olika värden på  $\alpha_r$ . En översiktlig bild över den decentraliserade regulatorn finns i Figur 7.



Figur 7: En översikt över strukturen för den decentraliserade regulatorn. Insignalerna  $r$ ,  $q$ ,  $z$  är referenssignaler och  $\hat{r}$ ,  $\hat{q}$  och  $\hat{z}$  är skattningar för vinkelhastighet i yaw- och pitchled respektive ändring i djup. Höger/vänster, upp/ner och fram/bakåt är hastigheter i  $y$ -,  $z$ - och  $x$ -led skickade från den externa PC:n. Variablerna  $y_f$ ,  $y_r$ ,  $z_f$  och  $z_r$  är signaler till motorer riktade i  $y$ - respektive  $z$ -led. Index "f" står för front och "r" står för rear. Den sista utsignalen  $T$  styr bakpropellerns motor.

### 7.3 LQ-regulator

Linjärvadratisk reglering är en reglerstrategi där man gör avvägningar för att minimera specifika designkriterier. Till exempel kan man välja en avvägning mellan reglerhastighet och små styrsignaler för systemet i (7.1) genom att vikta reglerfelet,  $e = z - r$ , och styrsignalernas,  $u$ , storlek och minimera integralen enligt (7.2) där  $Q_1$  och  $Q_2$  är viktmatriser [9].

$$\dot{x} = Ax + Bu + Nv_1 \quad (7.1a)$$

$$z = Mx \quad (7.1b)$$

$$y = Cx + v_2 \quad (7.1c)$$

$$\min(\|e\|_{Q_1}^2 + \|u\|_{Q_2}^2) = \min \int e^T(t)Q_1 e(t) + u^T(t)Q_2 u(t) dt \quad (7.2)$$

En LQ-regulator finns implementerad i ROV:en i dagsläget. I framtiden ska denna kunna användas för att ROV:en ska kunna utföra uppdrag helt autonomt men idag kan den användas för att reglera vinklar och djup. Tidigare års projektgrupper har skapat LQ-regulatorn men de hade inte tiden som krävdes för att utvärdera den. I detta projekt ska därför den befintliga LQ-regulatorn utvärderas. Dessutom ska regulatorparametrarna som använts utvärderas och förbättras.

För en LQ-regulator krävs att man har en linjär modell av systemet. Eftersom detta system är olinjärt har man under tidigare år valt att använda exakt linjärisering. Detta innebär att man väljer en styrsignal på ett sådant sätt att olinjäriteterna kompenseras bort och ett linjärt system erhålls. Under detta projekt ska den exakta linjäriseringen utredas och utvärderas.

Följande två delkapitel beskriver hur LQ-regulatorn vid projektets start är framtagen och Kapitel 7.3.3 beskriver vad som ska utföras under projektet gällande LQ-regulatorn.

### 7.3.1 Linjärisering av ROV:en

Då LQ-reglering används måste en linjär tillståndsbeskrivning användas. I tidigare projekt har man använt exakt linjärisering vilket innebär att de olinjäriteter som finns kompenseras bort, vilket resulterar i ett linjärt system. Följande ekvationer har använts och är hämtade från Kapitel 3.

$$\dot{\eta} = J\nu \quad (7.3)$$

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) = \tau, \quad (7.4)$$

där  $\eta$  och  $\nu$  estimeras med hjälp av delsystem Sensorfusion. Genom att derivera (7.3) med avseende på tiden fås

$$\dot{\nu} = J^{-1}[\dot{\eta} - \dot{J}\nu]. \quad (7.5)$$

Då  $J$  endast är singularär då pitchvinkeln  $\theta$  är  $\pm 90^\circ$ , har det inte tidigare orsakat några problem för ROV:en. Då dessa vinklar inte uppkommer vid normal körning. Om man väljer följande olinjära styrlag

$$\tau = M\mathbf{a}^b + C(\nu)\nu + D(\nu)\nu + g(\eta) \quad (7.6)$$

där  $\mathbf{a}^b$  används för att beskriva accelerationen i det kroppsfixa koordinatsystemet. Med hjälp av denna och (7.4) kan man få

$$M(\dot{\nu} - \mathbf{a}^b) = MJ^{-1}[\dot{\eta} - \dot{J}\nu - J\mathbf{a}^b] = 0 \quad (7.7)$$

Då man studerar

$$\ddot{\eta} = \mathbf{a}^n \quad (7.8)$$

där  $\mathbf{a}^n$  beskriver den önskade accelerationen i det jordfixa koordinatsystemet. Då man väljer

$$\mathbf{a}^n = \dot{J}\nu + J\mathbf{a}^b, \quad (7.9)$$

och använda (7.7) erhålls det linjära systemet

$$\mathbf{M}^*(\ddot{\boldsymbol{\eta}} - \mathbf{a}^n) = 0, \quad (7.10)$$

där  $\mathbf{M}^* = \mathbf{J}^{-T} \mathbf{M} \mathbf{J} > \mathbf{0}$ . Då är alltså (7.10) ekvivalent med ekvationen

$$\ddot{\boldsymbol{\eta}} - \mathbf{a}^n = 0. \quad (7.11)$$

Utifrån (7.9) kan man få ekvationen

$$\mathbf{a}^b = \mathbf{J}^{-1}[\mathbf{a}^n - \dot{\mathbf{J}}\boldsymbol{\nu}] = f(\mathbf{a}^n), \quad (7.12)$$

där  $\mathbf{a}^n$  är styrsignalen till det linjäriserade systemet. Styrsignalen i det icke-linjäriserade systemet är  $\mathbf{u} = (u_T, u_{yf}, u_{yr}, u_{zf}, u_{zr})^T$ . Vidare används formlerna (7.13) och (7.14) för att konvertera  $\boldsymbol{\tau}$  till styrsignaler och tvärtom.

$$\boldsymbol{\tau} = \mathbf{C}_u \tilde{\mathbf{u}} \quad (7.13)$$

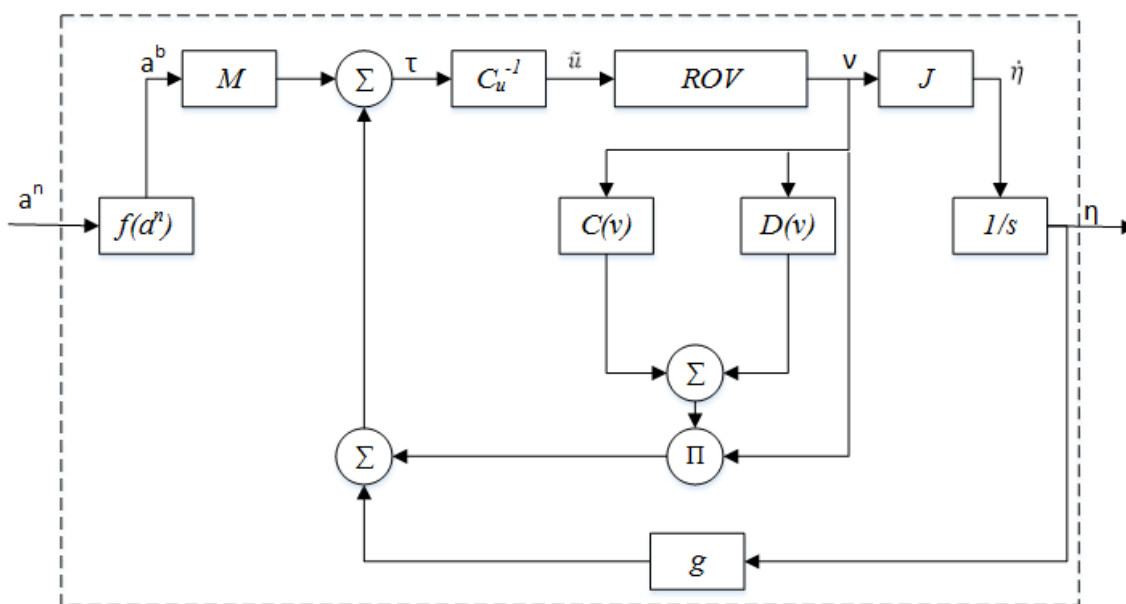
$$\tilde{\mathbf{u}} = \mathbf{C}_u^{-1} \boldsymbol{\tau} \quad (7.14)$$

där  $\boldsymbol{\tau}$  konverteras till  $\tilde{\mathbf{u}} = (u_T|u_T|, u_{yf}|u_{yf}|, u_{yr}|u_{yr}|, u_{zf}|u_{zf}|, u_{zr}|u_{zr}|)^T$  som är styrsignalerna som går till motorerna och

$$\mathbf{C}_u = \begin{bmatrix} C_T & 0 & 0 & 0 & 0 \\ 0 & C_{yf} & C_{yr} & 0 & 0 \\ 0 & 0 & 0 & C_{zf} & C_{zr} \\ 0 & 0 & 0 & -C_{zf}x_{zf} & C_{zr}x_{zr} \\ 0 & -C_{yf}x_{yf} & C_{yr}x_{yr} & 0 & 0 \end{bmatrix} \quad (7.15)$$

där  $C_T, C_{yf}, C_{yr}, C_{zf}, C_{zr}$  förklaras i Tabell 2.

Med hjälp av dessa ekvationer är det möjligt att beräkna de styrsignaler som ska skickas in i ROV:en baserat på ett  $\boldsymbol{\tau}$  som linjäriserar modellen. Strukturen för hur linjärisering ser ut kan man se i Figur 8.



Figur 8: En översikt av den exakta linjäriseringen. En matematisk beskrivning av systemet kan ses i (7.4). Metoden har hämtats från projekt under tidigare år [3]. Insignalen  $\mathbf{a}^n$  är den önskade accelerationen för ROV:en. Utsignalen  $\boldsymbol{\eta} = (n, e, d, \phi, \theta, \psi)^T$  används i de kvadratiske kostnadsfunktionerna.

### 7.3.2 Regulatorparametrar

När linjäriseringen av ROV:en är gjord har man erhållit ett frikopplat och linjärt system. Varje enskild insignal till detta system motsvarar varsin önskad acceleration i det jordfixa koordinatsystemet. För att kunna reglera detta har man valt att lösa ett LQ-problem, det vill säga minimera en viktfunktion enligt (7.2). I detta fall har man valt att vikta referensföljningen och styrsignalernas storlek men har även adderat en viktning av dynamiken för att straffa brus. Den kostnadsfunktion som ska minimeras blir då

$$J = \min \int_0^{\infty} \mathbf{e}^T(t) \mathbf{Q}_1 \mathbf{e}(t) + \mathbf{u}^T(t) \mathbf{Q}_2 \mathbf{u}(t) + \dot{\mathbf{z}}^T \mathbf{Q}_3 \dot{\mathbf{z}} dt \quad (7.16)$$

där  $\mathbf{e} = \mathbf{z} - \mathbf{r}$  är reglerfelen,  $\mathbf{z}$  är tillstånden och  $\mathbf{u}$  är styrsignalerna. Viktningen sker med hjälp av straffmatriserna  $\mathbf{Q}_1$ ,  $\mathbf{Q}_2$  och  $\mathbf{Q}_3$ .

Styrsignalen beräknas enligt tillståndsåterkoppling till

$$\mathbf{u} = -\mathbf{L}\mathbf{x}, \text{ där } \mathbf{L} = \mathbf{Q}_2^{-1}(\mathbf{B}^T \mathbf{S} + \mathbf{Q}_{12}^T) \quad (7.17)$$

där  $\mathbf{x}$  är systemets tillstånd och  $\mathbf{u}$  är styrsignalerna. Matrisen  $\mathbf{S}$  bestäms genom att lösa Riccati ekvationen för det linjära systemet och  $\mathbf{Q}_{12}^T$  beräknas utifrån  $\mathbf{Q}_3$  genom en omskrivning av  $\dot{\mathbf{z}}$  till en korsterm mellan  $\mathbf{x}$  och  $\mathbf{u}$ .

I tidigare projekt har LQ-problemet lösts genom att minimera följande kostnadsfunktioner, en för varje regleringsbar frihetsgrad. Den totala kostnadsfunktionen är då summan av dessa fem.

$$J_i = \frac{1}{2} \int_0^\infty ((\eta_{r_i} - \eta_i)^2 + \lambda_{1_i} \dot{\eta}_i^2 + \lambda_{2_i} (a_i^n)^2) d\tau, \quad i = 1, \dots, 5 \quad (7.18)$$

där  $\mathbf{a}^n$  relaterar till styrsignalen  $\tilde{\mathbf{u}}$  enligt följande samband

$$\begin{aligned} \tilde{\mathbf{u}} &= \mathbf{C}_u^{-1} \boldsymbol{\tau} = \mathbf{C}_u^{-1} (\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta})) = \mathbf{C}_u^{-1} (\mathbf{M}\mathbf{a}^b + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta})) = \\ &= \mathbf{C}_u^{-1} (\mathbf{M}\mathbf{f}(\mathbf{a}^n) + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta})) \end{aligned} \quad (7.19)$$

I (7.18) ovan är  $\eta_i$  och  $\eta_{r_i}$  tillstånden i systemets tillståndsbeskrivning och den önskade accelerationen och  $\mathbf{a}^n$  är insignalen till det linjäriserade systemet.

Följande val av önskad acceleration  $\mathbf{a}^n$  erhålls, enligt Fossen [1], efter minimering av kostnadsfunktionerna.

$$a_i^n = K_{p_i}(\eta_{r_i} - \eta_i) - K_{d_i}\dot{\eta}_i, \quad i = 1, \dots, 5 \quad (7.20)$$

där

$$K_{p_i} = \left( \frac{1}{\lambda_{2_i}} \right)^{1/2}; \quad K_{d_i} = \left( 2 \left( \frac{1}{\lambda_{2_i}} \right)^{1/2} + \frac{\lambda_{1_i}}{\lambda_{2_i}} \right)^{1/2} \quad i = 1, \dots, 5 \quad (7.21)$$

Genom att lägga till en term för integralverkan fås slutligen den önskade accelerationen  $\mathbf{a}^n$  som

$$a_i^n = K_{p_i}(\eta_{r_i} - \eta_i) - K_{d_i}\dot{\eta}_i + K_{I_i} \int_0^T (\eta_{r_i} - \eta_i) d\tau, \quad i = 1, \dots, 5 \quad (7.22)$$

där  $K_{I_i}$  väljs enligt (7.23). Enligt [3] motsvarar detta tumregeln  $T_{I_i} = 5T_{d_i}$  i en PID-regulator.

$$K_{I_i} = \frac{K_{p_i}}{5K_{d_i}} \quad i = 1, \dots, 5 \quad (7.23)$$

Alltså består den nuvarande regulatoren av fem oberoende PID-regulatorer, vars konstanter bestäms enligt (7.21) och (7.23). Straffkonstanterna  $\lambda_1$  och  $\lambda_2$  bestäms genom att variera storleken på båda konstanterna tills önskat resultat uppnås [3]. För att undvika integratoruppvridning i PID-regulatorerna låter man endast accelerationen integreras ifall denna ligger inom ett bestämt intervall.

### 7.3.3 Trimning av LQ-regulatoren

Under detta projekt ska LQ-regulatoren utvärderas och parametrarna ska förbättras. För att förbättra regulatoren ska ROV:en testas och utifrån resultaten av dessa tester ska förbättringar utföras. Testerna ska utföras torrt alternativt i bassäng. Detta itereras tills dess att ett bra resultat uppnås.

Koden från förra årets projekt ska utvärderas och implementationen av denna ska noggrant ses över.

Projektgruppen kunde varken finna tumregeln som använts vid val av  $K_{I_i}$  med hjälp av enkla internetsökningar eller i reglerteknisk kurslitteratur vid LiU. Däremot finns en vanligt använd tumregel som lyder  $T_i = 4T_d$  [2]. Därför ska bestämmandet av konstanten  $K_{I_i}$  undersökas närmare under projektet.



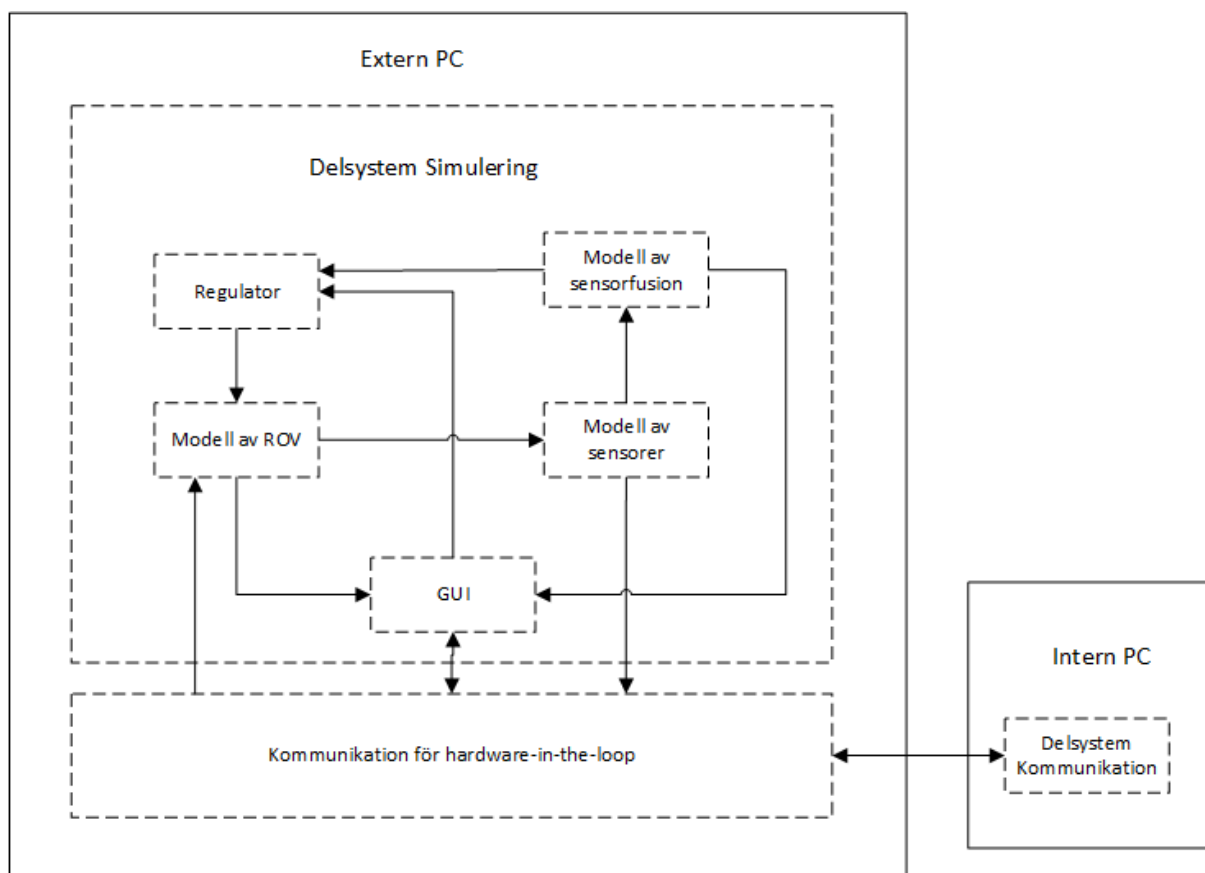
Tillståndsåterkopplingen som används i LQ-regulatorn kommer från den tillståndsskattning som beskrivs i Kapitel 6. Insignalerna, eller referenssignalerna, till regulatorn kommer att vara djup samt orienteringen i yaw- och pitchled och utsignalerna kommer vara styrsignaler till de fem motorerna. För att förhindra drift så behöver LQ-regulatorn ha referenssignaler för  $X$ - och  $Y$ -led. Dessa koordinater skattas inte i nuläget så om problem med drift uppkommer så kommer detta försöka lösas.

## 8 Delsystem Simulering

Simuleringsmiljön kommer utvecklas i Simulink där modellen för ROV:en implementeras. Denna modell ska ge en bra bild av ROV:ens dynamik. Modellen ska användas i en hardware-in-the-loop applikation och en applikation där även styrsystemet byggs upp och körs lokalt via ett grafiskt gränssnitt. Den del av simuleringsmiljön som endast ska köras lokalt ska gå att använda på godtycklig PC med den mjukvara som krävs.

Till simuleringsmiljön ska det kopplas ett grafiskt gränssnitt för ett enklare och mer intuitivt användande. Det grafiska gränssnittet ska innehålla referenssignaler samt data för ROV:ens djup och attityd. Simuleringen ska vara en realtidssimulering och ska kunna styras direkt genom GUI:t eller annan kontroll. Vid körning med hardware-in-the-loop ska Xbox-kontrollen kunna användas för att styra systemet. I det grafiska gränssnittet ska data visas i realtid både i sifferfält och grafiskt genom till exempel grafer.

För att ge en så bra bild som möjligt av styrsystemet ska samma algoritmer användas i simuleringsmiljön som i hårdvaran i så stor utsträckning som möjligt. I simuleringsmiljön ska även en enklare modell av SONAR finnas. Denna SONAR-modell ska kunna användas för att skatta avstånd till olika objekt i omgivningen.



Figur 9: En enkel skiss över simuleringsmiljön och dess kommunikation med hårdvaran. Streckade rektanglar representerar mjukvara och heldragna rektanglar representerar hårdvara. Pilarna visar informationsflöde både inom delsystemet simulering och med omgivande delsystem. GUI:t är tänkt att styra simuleringsmiljön och visa intressant data.

## 8.1 Kommunikation med andra delsystem

Kommunikation med andra delsystem sker endast i fallet hardware-in-the-loop. Data ska då skickas och läsas av i realtid för att hardware-in-the-loop-applikationen ska fungera på önskvärt sätt. Detta innebär att beräkningar i modellen måste vara utförda inom den tidsram som krävs för att datan ska nå hårdvaran i rätt tid. För att kommunicera med hårdvaran, som i detta fall är ROV:ens interna PC, används ROS och data skickas över Ethernet till ROV:ens interna PC:n. Simuleringsmiljön simulerar alltså den mekaniska ROV:en medan observatören och regulatorn som används är det som ligger i hårdvaran på den interna PC:n. Detta innebär att indata till modellen ska vara signaler till motorstyrkortet och utdata ska motsvara rådata från IMU, trycksensorer och magnetometer.

Mottagare	Signal	Enhet
Kommunikation	Simulerad rådata från trycksensor.	V
Kommunikation	Simulerad rådata från magnetometer.	°
Kommunikation	Simulerad rådata från IMU.	°/s och °/s <sup>2</sup>
Kommunikation	Referensvärden för vinkelhastighet i pitch- och yawled (då den decentraliserade regulatorn används).	°/s
Kommunikation	Referensvärden för vinklar i pitch- och yawled (då LQ-regulatorn används).	°
Kommunikation	Referensvärden för z-led (då LQ-regulatorn används).	m
Kommunikation	Om ROV:en ska operera i manuellt eller stabilt läge.	–
Kommunikation	Om ROV:en ska använda den decentraliserade regulatorn eller LQ-regulatorn då ROV:en opererar i stabilt läge.	–

Tabell 8: Tabell över vad som skickas från delsystemet Simulering.

Avsändare	Signal	Enhet
Kommunikation	Styrkoder för huvudmotor.	<i>styrkod</i>
Kommunikation	Styrkoder för övriga fyra motorer.	<i>styrkod</i>
Kommunikation	Av delsystemet Sensorfusion skattade tillstånd.	<i>styrkod</i>

Tabell 9: Tabell över data och datatyper som skickas till delsystemet Simulering.

## 9 Delsystem Kommunikation

Kommunikationen mellan en extern PC och de interna delsystemen Sensorfusion och Reglering kommer skötas av delsystemet Kommunikation. Delsystemet är även en viktig komponent i säkerheten på ROV:en eftersom det detekterar ifall kontakten mellan den externa PC:n och ROV:ens interna PC bryts. Samtliga signaler som skickas till och från delsystemet Kommunikation finns i Tabell 10 och 11.

Mottagare	Signal	Enhet
Externa PC:n	Skattade värden av pitch, yaw och roll.	$^{\circ}$
Externa PC:n	Skattade värden vinkelhastigheter pitch-, yaw- och rolled.	$^{\circ}/s$
Externa PC:n	ROV:ens djup.	$m$
Externa PC:n	Batterispänning	$V$
Externa PC:n	Läckagesensorsignal	$V$
Externa PC:n	Tryck	$V$
Externa PC:n	Rådata från magnetometer.	$^{\circ}$
Externa PC:n	Rådata från IMU-signalerna.	$^{\circ}/s$ och $^{\circ}/s^2$
Externa PC:n	Styr signaler för motorerna.	<i>styrkod</i>
Externa PC:n	Referensvärden för $q$ och $r$ .	$^{\circ}$
Externa PC:n	Referensvärden för pitch och yaw.	$^{\circ}$
Externa PC:n	Referensvärden för djup.	$m$
Referensgränssnitt	Referensvärden för vinkelhastighet i pitch- och yawled (då decentraliserade regulatorn används).	$^{\circ}/s$
Referensgränssnitt	Referensvärden för vinklar i pitch- och yawled (då LQ-regulatorn används).	$^{\circ}$
Referensgränssnitt	Referensvärden för $z$ -led (då LQ-regulatorn används).	$m$
Referensgränssnitt	Hastigheter i $x$ -, $y$ - samt $z$ -led (då den decentraliserade regulatorn används).	$m/s$
Referensgränssnitt	Hastigheter i $x$ samt $y$ -led (då LQ-regulatorn används).	$m/s$
Referensgränssnitt	Om ROV:en ska operera i manuellt eller stabilt läge.	–
Referensgränssnitt	Om ROV:en ska använda decentraliserad eller LQ-regulator då ROV:en opererar i stabilt läge.	–
Reglering	Nödläge (Då kontakten med den externa PC:n bryts).	–

Tabell 10: Tabell över vad som skickas från delsystemet Kommunikation.

Avsändare	Signal	Enhet
Externa PC:n	Referensvärden för vinkelhastighet i pitch- och yawled (då decentraliserade regulatorn används).	$^{\circ}/s$
Externa PC:n	Referensvärden för vinklar i pitch- och yawled (då LQ-regulatorn används).	$^{\circ}$
Externa PC:n	Referensvärden för $z$ -led (då LQ-regulatorn används).	$m$
Externa PC:n	Hastigheter i $x$ -, $y$ - samt $z$ -led (då den decentraliserade regulatorn används).	$m/s$
Externa PC:n	Hastigheter i $x$ samt $y$ -led (då LQ-regulatorn används).	$m/s$
Externa PC:n	Om ROV:en ska operera i manuellt eller stabilt läge.	–
Externa PC:n	Om ROV:en ska använda decentraliserad eller LQ-regulator då ROV:en opererar i stabilt läge.	–
Referensgränssnitt	Referensvärden för $q$ och $r$ .	$^{\circ}$
Referensgränssnitt	Referensvärden för pitch och yaw	$^{\circ}$
Referensgränssnitt	Referensvärden för djup	$m$
Reglering	Styr signaler till motorerna.	PWM
Sensorfusion	ROV:ens djup.	$m$
Sensorfusion	Skattade värden av pitch, yaw och roll.	$^{\circ}$
Sensorfusion	Skattade värden vinkelhastigheter pitch-, yaw- och rolled.	$^{\circ}/s$
Styr- och mätkort	Batterispänning	$V$
Styr- och mätkort	Läckagesensorsignal	$V$
Styr- och mätkort	Tryck	$V$
Styr- och mätkort	Rådata från magnetometer.	$^{\circ}$
Styr- och mätkort	Rådata från IMU-signaler.	$^{\circ}/s$ och $^{\circ}/s^2$

Tabell 11: Tabell över vad som mottages i delsystemet Kommunikation.

## 9.1 Kommunikation med externa enheter

Från den externa PC:n kommer information om vilket reglerläge (manuellt eller stabiliserat) som ska användas, vilken regulator som ska användas, om ROV:en ska vara av- eller påslagen, ifall nödstopp ska initieras och vilka referensvärden som ska följas skickas till delsystem Kommunikation. Alternativt ska delsystemet ta emot styr signaler från den externa PC:n ifall handkontrollen används.

Till den externa PC:n kommer delsystemet Kommunikation att skicka data med styr signaler, tillståndsskattningar och sensorvärden. Dessutom ska referensgränssnittets innehåll kunna skickas till den externa PC:n.

## 9.2 Kommunikation med interna enheter

Delsystemet Kommunikation ska skicka referensvärden till delsystemet Referensgränssnitt. Dessutom skickas information om vilket reglerläge (manuellt eller stabiliserat) som ska användas, vilken regulator som ska användas och om ROV:en ska vara av- eller påslagen till det delsystemet. Ett viktigt scenario att behandla är om kontakten med den externa PC:n av någon okänd anledning upphör. Om detta inträffar eller om den externa PC:n meddelar att nödläge ska initieras ska delsystemet Kommunikation meddela delsystemet Reglering att det ska gå in i nödläge.

Från styr- och mätkortet erhåller delsystemet Kommunikation sensorvärden samt motorernas styr signaler. Därtill erhåller delsystemet tillståndsskattningar från delsystemet Sensorfusion. Ibland kan det

även vara nödvändigt att kunna läsa av vad som finns lagrat i referensgränssnittet och därför ska även delsystemet Kommunikation kunna erhålla data därifrån.

## 10 Delsystem Referensgränssnitt

Syftet med delsystemet Referensgränssnitt är att det i framtiden ska ta fram en referenssignal och trajektoria för ROV:en utifrån skattningar från delsystemet Sensorfusion och en given uppgift. Detta är ett steg mot att utveckla ROV:en till en Autonomous Underwater Vehicle (AUV) och ska utföras av en framtida projektgrupp då budgeten inte räcker för årets projekt. Vid slutet av detta projekt ska detta delsystem ta emot referensvärde för samtliga styrbara frihetsgrader från delsystemet Kommunikation för att sedan skicka dessa vidare till delsystemet Reglering. Delsystemet Referensgränssnitt ska även kunna skicka tillbaka referensvärden till den externa PC:n via delsystemet Kommunikation.

### 10.1 Kommunikation med andra delsystem

I detta projekt ska delsystemet Referensgränssnitt verka som en kommunikationslänk mellan delsystemen Kommunikation och Reglering och har utöver det ingen uppgift. I Tabell 12 och 13 finns de signaler som skickas till och från delsystemet Referensgränssnitt.

Mottagare	Signal	Enhet
Reglering	Referensvärden för vinkelhastighet i pitch- och yawled (då den decentraliserade regulatorn används).	$^{\circ}/s$
Reglering	Referensvärden för vinklar i pitch- och yawled (då LQ-regulatorn används).	$^{\circ}$
Reglering	Referensvärden för $z$ -led (då LQ-regulatorn används).	$m$
Reglering	Hastigheter i $x$ -, $y$ - samt $z$ -led (då den decentraliserade regulatorn används).	$m/s$
Reglering	Hastigheter i $x$ samt $y$ -led (då LQ-regulatorn används).	$m/s$
Reglering	Om ROV:en ska operera i manuellt eller stabilt läge.	–
Reglering	Om ROV:en ska använda den decentraliserade regulatorn eller LQ-regulator då ROV:en opererar i stabilt läge.	–

Tabell 12: Tabell över vad som skickas från delsystemet Referensgränssnitt.

Avsändare	Signal	Enhet
Kommunikation	Referensvärden för vinkelhastighet i pitch- och yawled (då den decentraliserade regulatorn används).	$^{\circ}/s$
Kommunikation	Referensvärden för vinklar i pitch- och yawled (då LQ-regulatorn används).	$^{\circ}$
Kommunikation	Referensvärden för $z$ -led (då LQ-regulatorn används).	$m$
Kommunikation	Hastigheter i $x$ -, $y$ - samt $z$ -led (då den decentraliserade regulatorn används).	$m/s$
Kommunikation	Hastigheter i $x$ samt $y$ -led (då LQ-regulatorn används).	$m/s$
Kommunikation	Om ROV:en ska operera i manuellt eller stabilt läge.	–
Kommunikation	Om ROV:en ska använda den decentraliserade regulatorn eller LQ-regulator då ROV:en opererar i stabilt läge.	–

Tabell 13: Tabell över vad som mottages i delsystemet Referensgränssnitt.

## Referenser

- [1] Thor Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, 2011
- [2] Reglerteknik ISY, *Industriell reglerteknik - Kurskompendium*, 2014
- [3] Martin Lindfors, *Teknisk rapport, Remotely Operated Underwater Vehicle*. Institutionen för systemteknik (ISY) LiU, 2013.
- [4] TSRT10 Projektgrupp ROV, Oscar Wyckman m.fl. (2014) *Kravspecifikation Remotely Operated Underwater Vehicle*.
- [5] M. Eriksson *Utvärdering och vidareutveckling av undervattensfarkost*. Institutionen för industriell och ekonomisk utveckling (IEI) LiU, Examensarbete, 2011.
- [6] J. Bernhard, P.Johansson, *Advanced control of a remotely operated underwater vehicle*. Institutionen för systemteknik (ISY) LiU, Examensarbete, 2012.
- [7] Projektgrupp ROV 2012, <http://www.isy.liu.se/edu/projekt/tsrt10/2012/rov/> (2014-10-03)
- [8] Projektgrupp ROV 2013, <http://www.isy.liu.se/edu/projekt/tsrt10/2013/rov/> (2014-10-03)
- [9] Torkel Glad, Lennart Ljung, *Reglerteori - Flervariabla och olinjära metoder*, Studentlitteratur 2003.



## A Topics och messages

Topic	Innehåll	Publisher	Subscriber
mode	Systemets status, nödläge, manuell-, stabiliserad- eller autonom drift, LQ eller decentraliserad regulator.	GUI	ref,sensor
ref/mode	Systemets status, nödläge, manuell-, stabiliserad- eller autonom drift, LQ eller decentraliserad regulator.	ref	control
sensor/states	Systemets skattade tillstånd.	sensor	ref, GUI
ref/states	Systemets skattade tillstånd.	ref	control
gui/pidParameters	Parameterar till PID regulatorerna.	GUI	ref
gui/pidParameters	Parameterar till PID regulatorerna.	ref	control
gui/sensorParameters	Parametrar.	GUI	sensor
gui/modelParameters	Modellparametrar.	GUI	sensor, ref
ref/modelParameters	Modellparametrar.	ref	control
ard_batteries	Batterinivåer, läckage- och tryckinformation från Arduinon.	arduino	GUI, com
ard_sensors	Sensordata från Arduinon.	arduino	GUI, com, sensor
gui/leakage	Information om läckage har inträffat eller ej. Skickas internt inom GUI-noden.	GUI\src\mainwindow.cpp	GUI\src\warn.cpp
control/controlSignals	Styr signaler till motorerna (Stab. läge: från control till motorerna, manuellt läge: från GUI:t).	GUI, control	GUI, com, sensor
gui/decRef	Referenssignaler till den decentraliserade regulatorn	GUI	ref
ref/decRef	Referenssignaler till den decentraliserade regulatorn	ref	control
gui/LQRef	Referenssignaler till LQ-regulatorn	GUI	ref
ref/LQRef	Referenssignaler till LQ-regulatorn	ref	control
gui/ping	Skickas till interna PC:n så att bruten kontakt med GUI:t ska kunna detekteras.	GUI	com
com/arduinoData	Skickar motordata och batterispänning	com	GUI

---

gui/relayAndLight	Skickar relä- och ljusdata.	GUI	com
updateRequest	En förfrågan till GUI:t om att skicka referenssignaler när xbox-kontrollen används.	GUI	ref
ref/updateRequest	En förfrågan till GUI:t om att skicka referenssignaler när xbox-kontrollen används.	ref	control

Tabell 14: Översikt över de topics som finns i ROS och mellan vilka noder kommunikationen sker. Många topics har hämtats från föregående års projekt [3].

Topic	Meddelanden	Innehåll
mode	msg_Mode	Systemets status, nödläge, manuell, stabiliserad eller autonom drift, LQ eller decentraliserad regulator.
ref/mode	msg_Mode	Systemets status, nödläge, manuell-, stabiliserad- eller autonom drift, LQ eller decentraliserad regulator.
sensor/states	msg_States	<i>msg_Orientation</i> , <i>msg_Position</i> (se nedan), stamped.
	msg_Orientation	Två vektorer med tre flyttal vardera $(\phi, \theta, \psi)$ och $(p, q, r)$ .
	msg_Position	Två vektorer med tre flyttal vardera $((n, e, d)$ och $(u, v, w))$ .
ref/states	msg_States	<i>msg_Orientation</i> , <i>msg_Position</i> (se nedan), stamped.
	msg_Orientation	Två vektorer med tre flyttal vardera $(\phi, \theta, \psi)$ och $(p, q, r)$ .
	msg_Position	Två vektorer med tre flyttal vardera $((n, e, d)$ och $(u, v, w))$ .
gui/pidParameters	msg_PIDParameters	Sju flyttal $(K_\theta, Ti_\theta, Td_\theta, K_\psi, Ti_\psi, Td_\psi, K_z)$ , stamped.
gui/modelParameters	msg_ModelParameters	Sex flyttal.
ref/modelParameters	msg_ModelParameters	Sex flyttal.
ard_sensors	UInt16MultiArray	Ett heltal, (0 - 1023), i binär form
gui/leakage	String	Sträng 'leakage' om läckage, annars 'no_leakage'.
control/controlSignals	msg_ControlSignals	En vektor med 5 flyttal $(u_{yf}, u_{yr}, u_{zf}, u_{zr}, u_T)$ .
gui/decRef	msg_DecRef	5 flyttal $(\theta_{ref}, \psi_{ref}, \dot{n}_{ref}, \dot{e}_{ref}, \dot{d}_{ref})$ .
ref/decRef	msg_DecRef	5 flyttal $(\theta_{ref}, \psi_{ref}, \dot{n}_{ref}, \dot{e}_{ref}, \dot{d}_{ref})$ .
gui/LQRef	msg_LQRef	5 flyttal $(n_{ref}, e_{ref}, d_{ref}, \theta_{ref}, \psi_{ref})$ .
ref/LQRef	msg_LQRef	5 flyttal $(n_{ref}, e_{ref}, d_{ref}, \theta_{ref}, \psi_{ref})$ .
gui/ping	String	Sträng 'ping'.
com/arduinoData	msg_ArduinoData	6 flyttal för batterispänningar, ett flyttal för trycket, ett flyttal för läckagedetektorns spänning, tre heltal för reläspänningarna, stamped
gui/relayAndLight	msg_RelayAndLight	Tre heltal för reläspänningar, ett heltal för ljus, stamped
updateRequest	String	Sträng 'send'.
ref/updateRequest	String	Sträng 'send'.

Tabell 15: Förklaring av alla messages som kan skickas mellan noderna i ROS. Många meddelanden är hämtade från föregående års projekt [3].