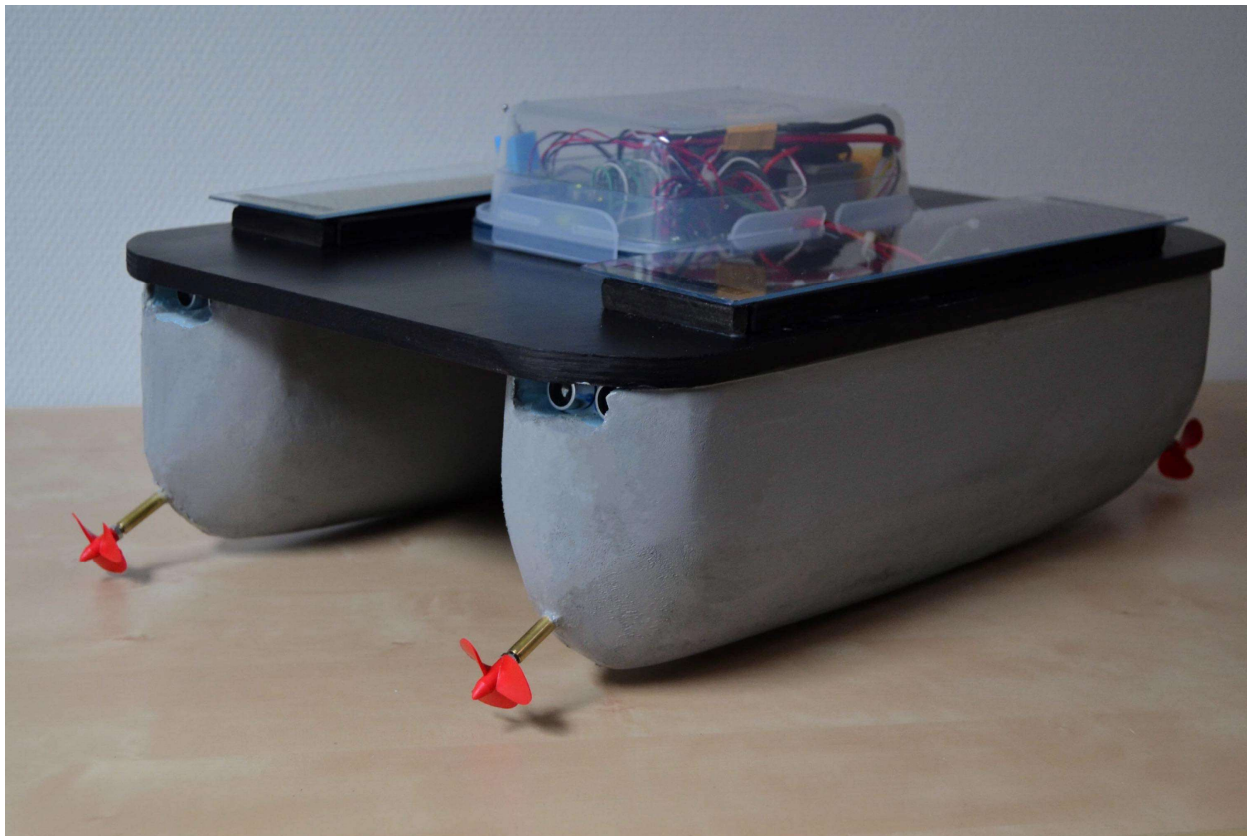




**CHALMERS**  
UNIVERSITY OF TECHNOLOGY



## Utveckling av autonom personfärja

Framtagning av modell samt resonemang kring verklig produkt

Examensarbete inom högskoleingenjörsprogrammet Mekanik

Daniella Rusanen

Kristin Blomdahl

Institutionen för Elektroteknik  
CHALMERS TEKNISKA HÖGSKOLA  
Göteborg, Sverige 2020



EXAMENSARBETE INOM MEKATRONIK

**Utveckling av autonom personfärja**  
Framtagning av modell samt resonemang kring verklig produkt

Daniella Rusanen  
Kristin Blomdahl



Institution för Elektroteknik  
CHALMERS TEKNISKA HÖGSKOLA  
Göteborg, Sverige 2020

## **Utveckling av autonom personfärja**

Framtagning av modell samt resonemang kring verklig produkt

Daniella Rusanen  
Kristin Blomdahl

© DANIELLA RUSANEN, KRISTIN BLOMDAHL, 2020

Examinator: Bertil Thomas, Institutionen för elektroteknik

Institutionen för Elektroteknik  
Chalmers tekniska högskola  
SE-412 96 Göteborg Sverige  
Telefon: +46 (0)31-772 1000

Framsida: Den slutgiltiga versionen av den självkörande färjan.  
Göteborg, 2020

## Sammanfattning

Automatisering sker i en ökande takt ute i samhället bland annat för att få en större säkerhet för individer. Här utreds frågan om möjligheten att automatisera passagerarfärjor med hjälp av att framställa en modell. Modellen av färjan har en automatiserad styrning samt drivning vilket gör att den kan ta sig från en punkt till en annan genom att hålla en så rak kurs som möjligt samt att den riktar upp och stannar motorer vid närmande av kaj. Arbetet har utförts till stor del på distans och provkörningen har skett i Broccoli Engineering ABs kontor i Göteborg. Material till projektet har finansierats av Broccoli Engineering AB. Det som tillämpas i modellen är styrning med hjälp av mikrokontroller Arduino och avståndssensorer med ultraljud. Modellen av färjan tar inte hänsyn till gällande lagar och regler. Skrovet på modellen är byggd till största delen av cellplast och har formen som en katamaran. Testerna som skedde i en mindre pool visar att det går utmärkt att använda sig utav fyra motorer utan att det blir motstånd i vattnet, hur detta skulle fungera i praktiken med större dimensioner är dock svårt att förutspå och kräver mer omfattande testkörningar. Dock hade den aktuella testningen varit mer optimal i en större anläggning för att få fram mer data. Det som testningen visade gav en diskussion om en autonom färja är möjligt och det gav ett positivt svar.

Nyckelord: Autonom, passagerarfärja, Arduino

## **Abstract**

Automation occurs at an increasing rate in the society to raise the security for individuals. A model is constructed to see if it is possible to automate passenger ferries. The model has an automated control and propulsion which means that it can get from one point to another by keeping a straight course by adjusting itself and it can also stop the engines while approaching the dock. The work has carried out mainly remotely and the test runs took place at Broccoli Engineering AB's office in Gothenburg. The materials of the project have been financed by Broccoli Engineering AB. What is applied in the model is a steering controlled by a microcontroller called Arduino with ultrasonic sensors measuring the distance. The model of the ferry does not take laws and regulations into account. The hull is formed as a catamaran and is mostly made by cellular plastic. The test showed that it works fine to have four engines without getting to much water resistance, however it is difficult to predict how it would work with a real size of a ferry and it requires more extensive test runs. It would have been ideal to have a larger testing facility to obtain more data but the current testing still gave a discussion about whether an autonomous ferry is possible and it gave a positive response.

Keywords: Autonomous, passenger ferry, Arduino

## Förord

Detta är ett examensarbete på Chalmers Tekniska Högskola, Mekanikprogrammet för Högskoleingenjör hos institutionen för elektroteknik. Omfattningen är 15 hp och har utförts på hösten år 2020 under 15 veckor.

Examensarbetet utfördes på Broccoli Engineering AB. Projektet i sig innefattar de viktiga delarna inom mekatronikprogrammet som elektronik, programmering, materialval, mekanik samt att kunna anordna och utföra ett projekt.

Vi vill tacka Broccoli Engineering AB för deras varma välkomnande och särskilt tack till Björn Bergholm för att han gav oss tilliten att utföra detta projekt och framförallt möjligheten att förverkliga vår idé.

Tack till examinator Bertil Thomas och även ett tack till David Wüstenhagen och Niclas Jernberg från Chalmers Tekniska Högskola för stöttning och goda råd under projektets gång.

Daniella Rusanen, Kristin Blomdahl

# Innehållsförteckning

Sammanfattning .....	i
Abstract .....	ii
Förord.....	iii
Terminologi.....	1
1. Inledning .....	2
1.1 Bakgrund .....	2
1.2 Syfte .....	2
1.3 Avgränsningar.....	2
1.4 Precisering av frågeställningen.....	2
2. Teknisk bakgrund.....	4
2.1 Arduino Uno Rev. 3 SMD .....	4
2.2 Ultraljudssensor, HC-SR05 .....	4
2.3 DC motor, Mabuchi .....	5
2.4 ESC – Electronic speed controller.....	6
2.5 Batteri Zippy Compact .....	6
2.6 Toggle Switch DQ11 .....	7
3. Metod .....	8
4. Genomförande.....	9
4.1 Tidsplan .....	9
4.2 Elektronik.....	9
4.2.1 Kopplingschema och blockschema.....	9
4.2.2 Elektronikkomponenter .....	10
4.2.3 Strömförsörjning .....	11
4.3 Mjukvara.....	12
4.3.1 Val av Mikrokontroller.....	12
4.3.2 Flödesschema.....	13
4.3.3 Funktioner.....	14
4.3.4 Styrning .....	14
4.4 Skrov.....	15
4.4.1 Material .....	16
4.4.2 Motorfäste .....	17
4.4.3 Batterihållare .....	18
4.4.4 Skydd.....	18
4.4.5 Drivaxel och propeller.....	19

4.4.6 ESC Hylla.....	19
<b>5. Resultat .....</b>	<b>20</b>
<b>6. Diskussion.....</b>	<b>21</b>
<b>6.1 Diskussion av färdig modell.....</b>	<b>21</b>
<b>6.2 Modell till verklig produkt .....</b>	<b>21</b>
6.2.1 Skrov .....	21
6.2.2 Drivning samt styrning .....	21
6.2.3 Sensorer .....	21
6.2.4 Passagerarvänlighet .....	22
6.2.5 Materialval.....	22
6.2.6 Lagar och säkerhet.....	22
<b>6.3 Förslag på vidareutveckling på modell .....</b>	<b>22</b>
<b>6.4 Miljö och hållbar utveckling.....</b>	<b>23</b>
<b>6.5 Etik och moral .....</b>	<b>23</b>
<b>7. Slutsats .....</b>	<b>24</b>
<b>Referenser.....</b>	<b>25</b>
<b>A Bilaga Gantt-schema</b>	
<b>A.1 Gantt-schema vid start</b>	
<b>A.2 Bilaga Gantt-schema omgjord</b>	
<b>B Bilaga Kopplingsschema</b>	
<b>C Bilaga 3D ritningar</b>	
<b>C.1 Bilaga Batterihållare</b>	
<b>C.2 Bilaga Hölje Toggle Switch</b>	
<b>D Bilaga programkod</b>	

## **Terminologi**

CAD – Computer-aided Design

DC – Direct current (Svenska: Likström)

ESC – Electronic Speed Controller (Svenska: Hastighetsregulator)

GND – Ground (Svenska: Jord)

LiPo – Lithium-polymer battery (Svenska: Litiumpolymerbatteri)

PWM - Pulse Width Modulation (Svenska: Pulsbreddsmodulering)

USB – Universal Serial Bus

# 1. Inledning

I det här avsnittet presenteras projektets bakgrund, syfte, avgränsningar samt mål och frågeställningar.

## 1.1 Bakgrund

Projektet är en egen idé där ändamålet är att utforska och se möjligheterna att automatisera dagens fartyg, inriktat på passagerarfärjor, för transport inom det marina. Om färjorna istället automatiseras kan den självkörande färjan ta passagerare över vattnet på ett smidigt och effektivt sätt med en minimering av eventuella handhavandefel samt minska energiförbrukningen. Även Broccoli Engineering AB är intresserad av just det här området vilket gjorde det optimalt att ta reda på mer inom ämnet. Autonoma fordon är just nu i stort fokus vilket medför en nyfikenhet till att automatisera inom det marina.

## 1.2 Syfte

Syftet med projektet är att skapa en småskalig modell av en självkörande färja och där genom utforska möjligheten om automatisering av en verklig passagerarfärja är genomförbart. Vilket i sin tur öppnar upp för en diskussion kring tillämpning av modellens autonoma lösning i framtidens färja.

Den automatiserade modellen av färjan har syftet att kunna ta sig från en punkt till en annan genom att hålla sin kurs med hjälp av sensorer.

## 1.3 Avgränsningar

Avgränsningar inom projektet för färjan är att den kommer att ha en elektrisk drivning dock kommer inte laddstationer för batteri att behandlas. Vid styrning av färjan kommer GPS-system eller koordinatsystem inte att användas. Det finns fall då det finns ett inbyggt system på fartyg som stannar motorerna om fel uppstår, om t.ex. något skulle fastna i propellern. Detta system finns inte med. Modellfärjan tar heller inte hänsyn till lagar och regler som krävs ute till havs t.ex. i form av lanternor/ljus eller nödutrustning m.m.

## 1.4 Precisering av frågeställningen

Huvudmålet i projektet är att ta fram en modell av en färja som ska ta sig från en punkt till en annan per automatik med hjälp av ett styrsystem, sensorer samt motorer. Ett diskussionsavsnitt finns med för att undersöka hur man skulle kunna gå från modell till verklig färja.

Projektet delas även in i delmål eftersom att det finns en begränsad tid att ta hänsyn till samtidigt som det även underlättar arbetet genom att dela upp huvudmålet till mindre delmål. Delmålen som finns prioriteras enligt nedanstående ordning:

1. Driva färjan på ett effektivt sätt från kaj A till kaj B och samtidigt kunna hinna bromsa innan färjan når kajen.
2. Fokusera på att driva färjan tillbaka från kaj B till ursprungskajen A och även här hinna bromsa innan färjan når kajen.
3. Få färjan att åka rakt fram med hjälp av lämplig styrning.
4. Undvika föremål i form av andra båtar m.m. med hjälp av sensorer/givare.
5. Ett mindre prioriterat delmål är att kunna lägga till vid respektive kaj.

De frågeställningar för modellen som tagits hänsyn till är följande:

- Vilken motor är den optimala för ändamålet d.v.s. har rätt hastighet för det motstånd som skapas av vattnet?
- Vilket är mest optimalt för ändamålet, Arduino eller Raspberry pi?
- Vilka typer av sensorer eller givare skall användas?
- När ska båten stanna för att inte köra in i kajen?
- Vilken typ av lösning skall användas för styrning av färja?
- Vilket skrov är att föredra för optimal körning samt vad för material skall användas?
- Vilka sensorer/givare skall användas för att undvika andra föremål?

## 2. Teknisk bakgrund

Kapitlet innehåller den tekniska bakgrunden kring de komponenter som används och dess kommunikation för att ge en djupare förståelse till projektet.

### 2.1 Arduino Uno Rev. 3 SMD

Arduino är ett utvecklingskort (mikrokontroller) som är baserat på ett så kallat ATmega328 med öppen källkod och förinstallerad bootloader. En bootloader kan förklaras som en programvara vilket efter BIOS (Binary input/output system) laddar in operativsystemet från sekundärminnet till primärminnet. Arduinon innehåller 14 digitala in- och utgångar varav 6 av dessa har PWM-funktion (Pulse Width Modulation). Den innehåller dessutom 6 stycken analoga ingångar. Arduinon programmeras via en USB-B-kabel som även kan fungera som strömkälla. Utöver drivning från USB-B-kabel kan den drivas av en DC-kontakt (med en spänning på 7-12V) eller ett batteri [1]. Den har även en klockfrekvens på 16 MHz och ett flash-minne på 32 kB varav 0,5 kB tas upp av bootloader.

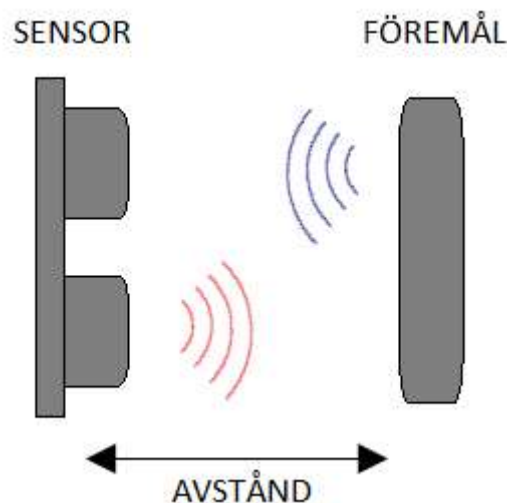
### 2.2 Ultraljudssensor, HC-SR05

HC-SR05 är en avståndsmätare med ultraljud som används för att mäta avstånd med hjälp av t.ex. en Arduino. Den har en räckvidd på 4,5 m och en noggrannhet på 3 mm. Drivspänning är på 5V. Sensorn fungerar genom att en trigger-puls skickar iväg en ljudsignal som sedan studsar på ett föremål och återvänder till sensorns ECHO-signal, detta visas i figur 2.1. Den tiden det tar att skicka iväg signalen och få tillbaka den avgör hur långt avståndet är [2]. Värdet på hur långt avståndet är beräknas med hjälp av ekvationen 2.1.

$$distance = \frac{traveltime}{2 * speed\ of\ sound}$$

$$speed\ of\ sound = 343\ [m/s] = 0,0343[cm/\mu s] = \frac{1}{29,1}\ [cm/\mu s]$$

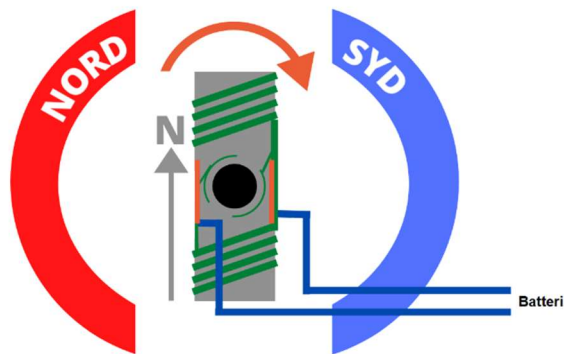
*Ekvation 2.1, visar hur distansen för ultraljudsgivaren beräknas från dess värde till ett avstånd i cm.*



*Figur 2.1, Visar hur ultraljudet studsar hos givaren.*

### 2.3 DC motor, Mabuchi

DC står för “direct current”, likspänning på svenska. Denna motor är en sort som är med kol, engelska “brushed” vilket menas med att motorn har en roterande del med trådlindade spolar som skapar ett magnetfält med två poler enligt figur 2.2. För varje varv den roterar så växlar en mekanisk kontakt magnetfältets riktning [3]. Under drift slits kolet ner då den ligger emot den roterande delen inuti motorn, detta medför att motorn i sig har en bestämd livslängd men det finns fall där kolet enkelt går att byta ut till ett nytt kol.



*Figur 2.2, Borstad DC motor.*

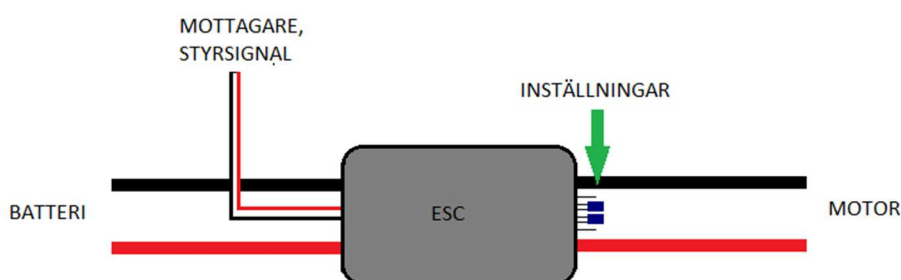
Motorn är på 90 W och ett varvtal på 20040 rpm vid en maximal verkningsgrad. Denna motor väger 160 g och har måtten 36x58 mm.

## 2.4 ESC – Electronic speed controller

Electronic speed controller, på svenska hastighetsreglage, är en sorts motorkontroller som används för att styra det bestämda varvtalet på en motor, vilket sker helt elektroniskt. Fyra ESC av samma sort används till färjan och är av märket Turnigy. Hastighetsreglaget är på 30 A och är av typen borstad. Om man använder sig utav en borstad motor så krävs det att även ESC:n är av borstad typ.

Hastighetsreglaget har sju stycken kablar där två av dem ger strömförsörjning från batteriet, två kablar ansluts till vardera tillhörande motor och de tre resterande kablarna kopplas till Arduinon. En av de sistnämnda är en styrsignal som krävs för att styra ESC som i sin tur styr motorn.

På en ESC kan inställningar ändras för att få den att passa för olika ändamål, till exempel den första inställningen är om ett LiPo-batteri eller ett NiCd/NiMh-batteri ansluts till ESC:n. Den andra inställningen bestämmer om ESC:n ska bromsa motorn eller ej. Dessa inställningar görs manuellt direkt på ESC genom att flytta på två byglar till respektive läge [4]. I figur 2.3 nedan visas en enkel översikt på hur hastighetsreglaget ser ut, var kablar ska anslutas samt var inställningarna görs.



Figur 2.3, En överblick på ESC och dess anslutningar

## 2.5 Batteri Zippy Compact

Den strömkälla som används till färjan är batteri. I tabell 2.1 nedan visas batteriets tekniska egenskaper. Valet föll på ett 3 celligt LiPo-batteri av märket Zippy med en spänning på 11,1 V vilket ger 3,7 V per cell. Batteriet är på 2200 mAh och har en laddning- samt urladdningshastighet på 40C. C-värdet talar om hur mycket ström batteriet klarar av utan att gå sönder.

Eftersom batteriet är av modell Compact betyder det att de har en hög prestanda samt kvalitet men i ett kompakt format, vilket även kan ses på måtten i tabellen. LiPo står för Litiumpolymer.

Märke/modell	Zippy Compact
Amperetimmor	2200 mAh
Antal celler	3
Typ	LiPo
Laddning-/urladdningskonstant	40C
Spänning	11,1 V
Mått	109x34x27 mm

Tabell: 2.1 Batteriets tekniska egenskaper.

## 2.6 Toggle Switch DQ11

En Toggle Switch, på svenska vippströmbrytare, används med ett 3D-utskrivet hölje för att skydda elektroniken. Switchen placeras utanför det vattensäkra skyddshöljet och har ett utseende enligt figur 2.4. CAD-ritningar för 3D-utskriften, som hittas i bilaga C2, finns på hemsidan Thingiverse och är gjord av Nils R. [5]. Materialet är utav plast som används för 3D-utskrifter.



*Figur 2.4, 3D-utskrivet hölje för vippströmbrytaren*

### 3. Metod

För att nå fram till huvudmålet som är att bygga en fungerande modell av en automatiserad färja krävs det ett tillvägagångssätt som ger önskad slutprodukt. Delmål för färjan sattes upp och för att effektivisera arbetets gång utvecklades mjukvara samt hårdvara parallellt.

Testkörningar för sensorer och motorer gjordes för undersökning och justering för stopptid samt styrning. Körningarna skedde på land för test av sensorer och den slutgiltiga testkörningen genomfördes i en vattenfylld pool med måtten 1,8x3 m. Vattennivån styrdes av att få propellrarna under vattenytan (ca 10 cm). Testkörningar gjordes upprepade gånger och värdena ändrades i programkoden för att ge en tydlig reaktion på sensorerna och motorerna.

Valet att skapa en modell var för att se hur möjligheterna för automatisering av en färja skulle kunna tillämpas i praktiken. Även öppna upp för en diskussion kring bl.a. problem som kan dyka upp samt vad man bör tänka på.

Programmeringen sker i språket C och programmet byggdes upp genom mindre program som testades var för sig och sedan sattes ihop till det slutgiltiga programmet. Detta för att kunna provköra under arbetets gång och för att få en enklare felsökningsmetod vilket sparar på tidsresurser.

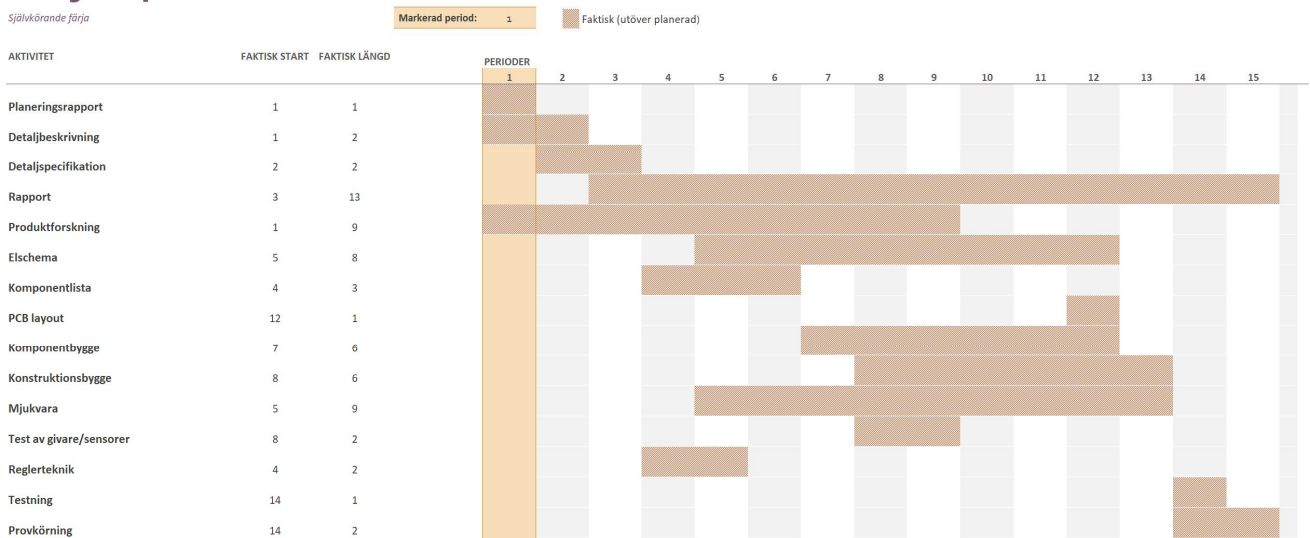
## 4. Genomförande

I det här avsnittet förklaras mer ingripande hur var och en av de mindre komponenterna och programmering har sammanställts för att uppnå målen.

### 4.1 Tidsplan

Projektet pågick i sammanlagt 15 veckor. Vid start utformades en tidsplan i form av ett Gantt-schema för att enkelt kunna följa upp och därmed nå målen i tid. Vissa ändringar skedde under projektets gång vilket kan läsas mer om i diskussionen, avsnitt 6.3. Figur 4.1 nedan visar Gantt-schemat som är baserat på den tidsomfattning, metod samt de delmål som sattes upp. I bilaga A.1 visas schemat i större format.

#### Projektplanerare



Figur 4.1, Version ett över Gantt-schema för projektets gång.

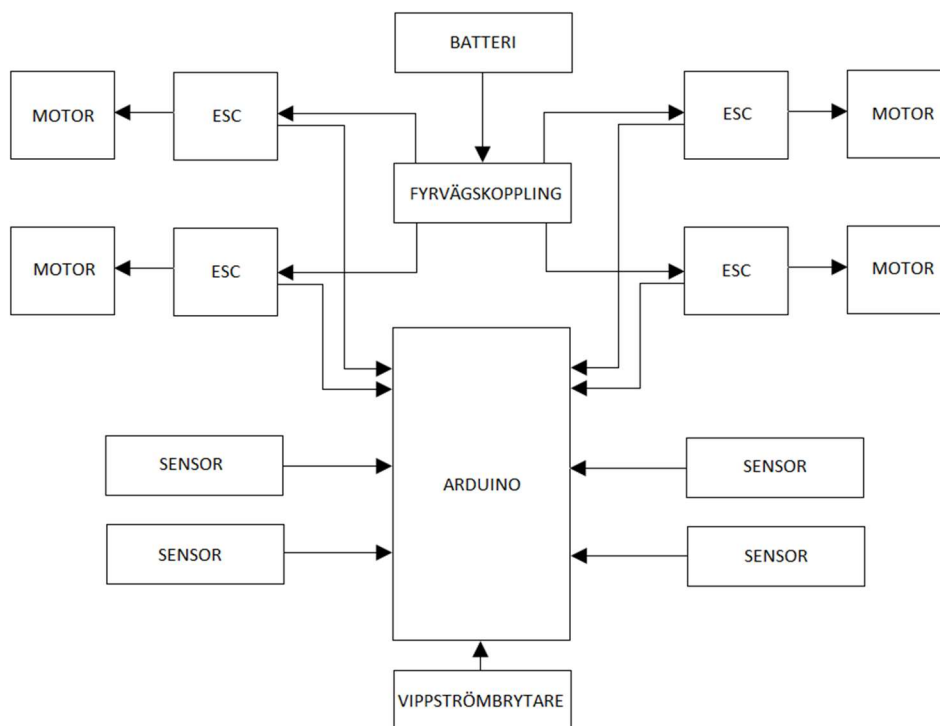
### 4.2 Elektronik

I det här avsnittet förklaras det mer djupgående hur de olika elektroniska komponenterna fungerar och används för modellfärjan.

#### 4.2.1 Kopplingschema och blockschema

I nedanstående figur 4.2 visas ett blockschema som ger en förenklad bild på hur elektroniken är ansluten med varandra, dock visas inte vilka specifika kablar från komponenterna som ansluts till varandra utan enbart att de är anslutna. Det detaljrika kopplingschemat går att finna i bilaga B.

Blockschemat i figur 4.2 visar att det är batteriet som ger strömförsörjning till resterande komponenter med hjälp av en fyrvägskoppling som går ut till ESC och motorer. ESC ger i sin tur spänning till Arduinon för att förse sensorerna med spänning.



Figur 4.2, Blockschema för att enkelt beskriva färjans kopplingar.

#### 4.2.2 Elektronikkomponenter

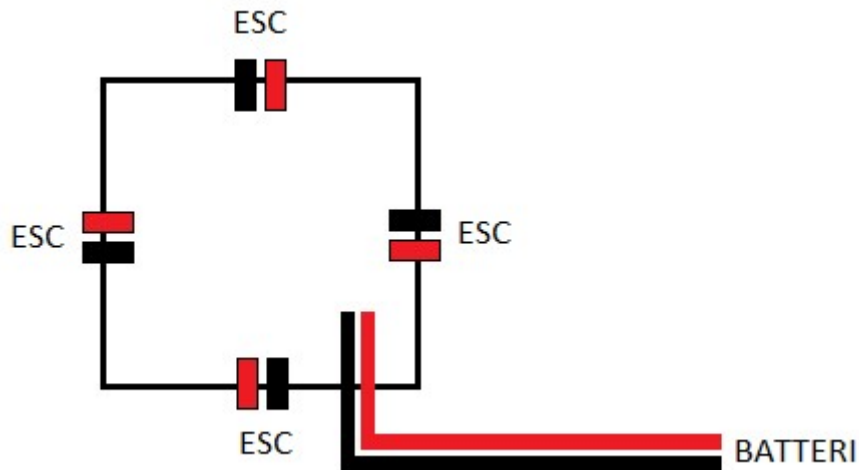
I figuren 4.3 visas alla de elektronikkomponenter som använts och dess position i färjan. Allt har anslutits samman i en låda och från höger till vänster visas batteriet, fyra stycken ESC med fyrvägskopplingen under, Arduino samt experimentkort. Bredvid behållaren kan vippströmbrytaren hittas. I utrymmena ovan och under som visas i figur 4.3 sitter motorerna samt sensorerna vilket kan läsas mer om i avsnitt 4.4 Skrov.



Figur 4.3, Färja ovanifrån utan dess skydd, visar elektronikkomponenterna.

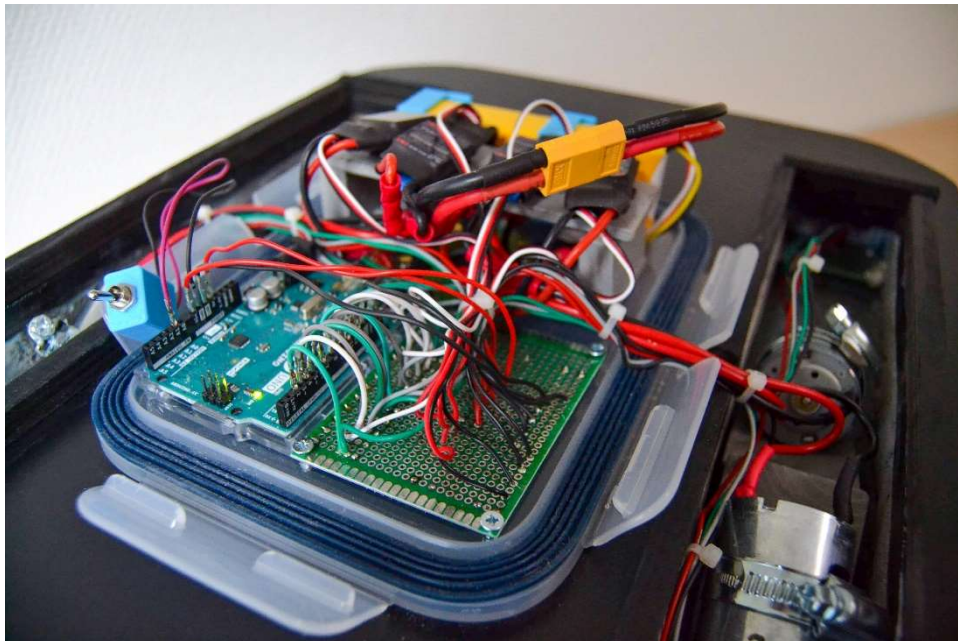
### 4.2.3 Strömförsörjning

Systemet drivs av ett 11,1 V batteri som är anslutet med en fyrvägskoppling. Funktionen för en fyrvägskoppling är att fungera som en fördelare genom att kunna ansluta ett batteri till fyra ESC på samma gång. Figur 4.4 visar en förenkling av fyrvägskopplingen vars kablar är anslutna till respektive komponent och där rött innebär plus medan svart för minus.



Figur 4.4, Fyrvägskoppling som användes för att koppla samman ESC och batteri.

Ett experimentkort används för att ansluta samman allt som förbinds med Arduino vilket är ingångar och utgångar från ESC, vippströmbrytaren samt sensorerna. Även GND, 5V samt  $V_{in}$  finns med på experimentkortet vilket visas i figur 4.5.  $V_{in}$  på arduinon är ansluten till experimentkortet tillsammans med strömförsörjningskabeln från ESC enligt kopplingschemat som hittas i bilaga B.



Figur 4.5, Bild på elektronik och framförallt experimentkortet samt Arduino.

Experimentkortet har kopplats enligt följande färgkod:

Röd: +5V samt  $V_{in}$

Svart: GND

Vit: styrkabel ESC

Grön: TRIG till sensor

Grå: ECHO till sensor

Lila: Kabel till vippströmbrytare

### **4.3 Mjukvara**

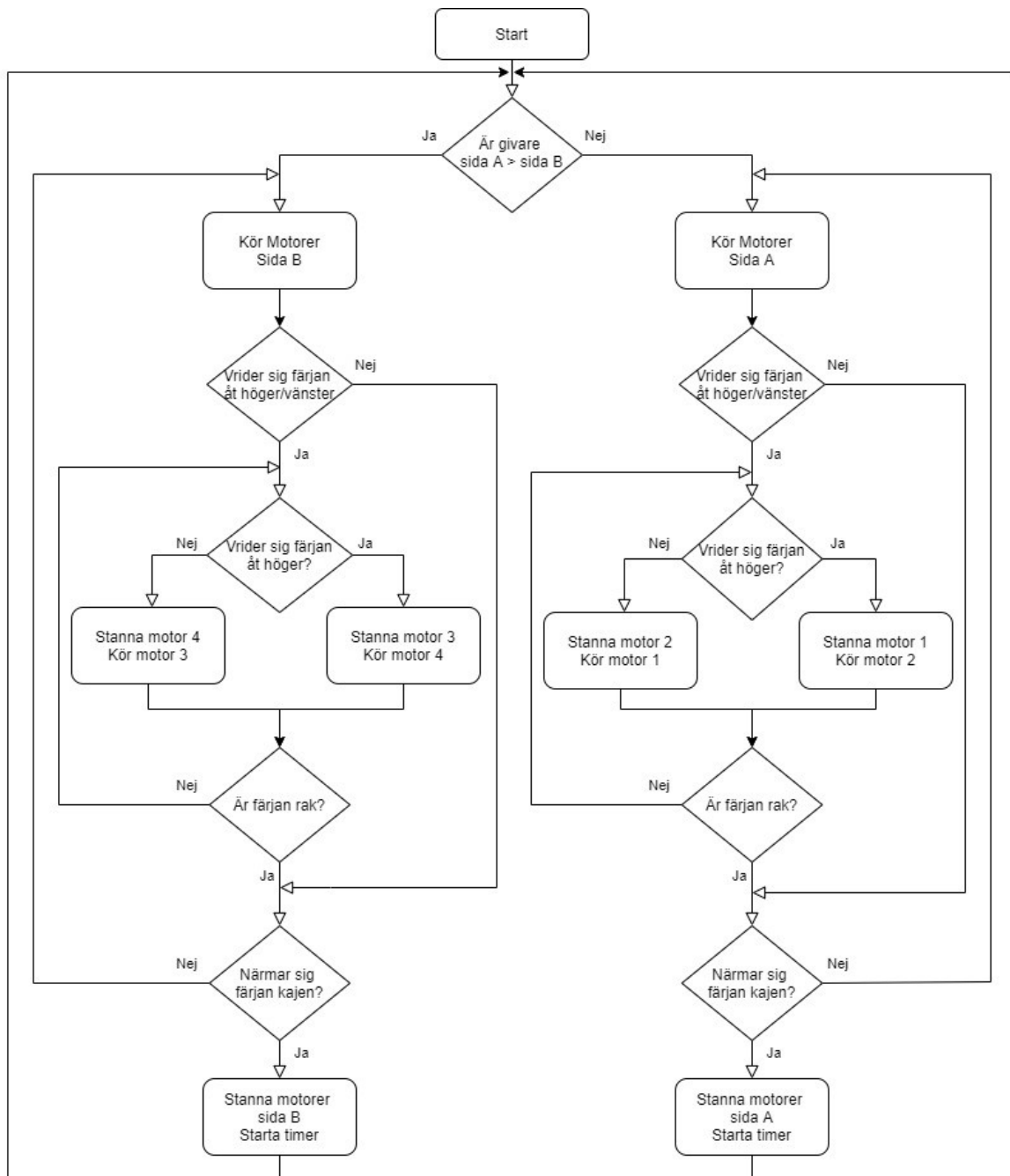
Här presenteras programuppbyggnaden genom flödesschema samt en beskrivning och även valet av mikrokontroller. Den färdiga programkoden går att finna i bilaga D.

#### **4.3.1 Val av Mikrokontroller**

Vid valet av mikrokontroller stod det mellan en Arduino och en Raspberry Pi. Den stora skillnaden mellan dessa två komponenter är att en Arduino är en mikrokontroller som beskrivits mer om under avsnitt 2.1 medan Raspberry Pi är en mikrodator med ett eget operativsystem. Eftersom det inte krävdes ett större avancerat system som en Raspberry Pi för programmering av färjan valdes Arduinon. En större nackdel med Raspberry Pi är att den är mer kraftfull och avancerad vilket i sig resulterar i att priset blir högre samt att förbrukningen på batteriet ökar. Andra fördelar med en Arduino är att de är lätta att arbeta med när det kommer till att ansluta komponenter som sensorer. Ytterligare en fördel med Arduino är att om den plötsligt förlorar sin spänning, stängs den endast av utan att det uppstår andra problem [6].

### 4.3.2 Flödesschema

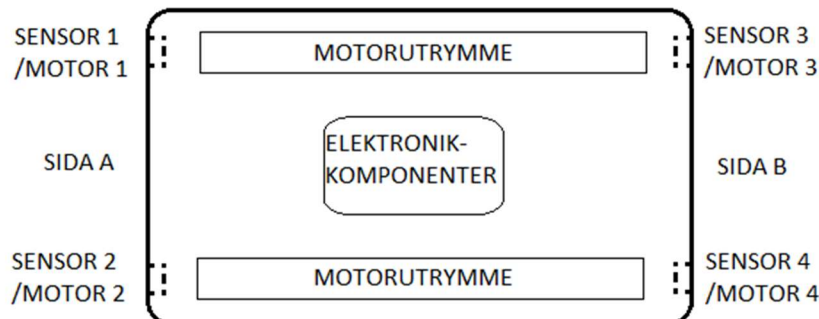
För att få en överblick över hur programmet utformats skapades ett flödesschema vilket visas i figur 4.6 nedan. Det är en gren för vardera riktning och det som avgör vilken gren programmet tar i flödesschemat beror på vilken sida på färjan som har längst till kajen, på detta sätt kan sensorerna känna av om färjan befinner sig vid kaj A eller kaj B. Därefter är de två riktningarna likartade. Den enda skillnaden är vilka motorer som ska starta och stanna för att rikta upp färjan om den vrider sig, eftersom färjan skall hålla en rak kurs fram till dess att den är framme.



Figur 4.6, Flödesschema över programkod.

### 4.3.3 Funktioner

I figur 4.7 syns en förenklad översyn av färjan ovanifrån som visar placering av sensorer samt övrig elektronik. Färjan är indelad i två sidor, A respektive B, samt har numrering för motorer och sensorer för att ge en enklare överblick på beskrivningen av båtens funktioner, vilket kan läsas mer av i avsnittet 4.3.4 Styrning.



Figur 4.7, Enklare ritning av färjan.

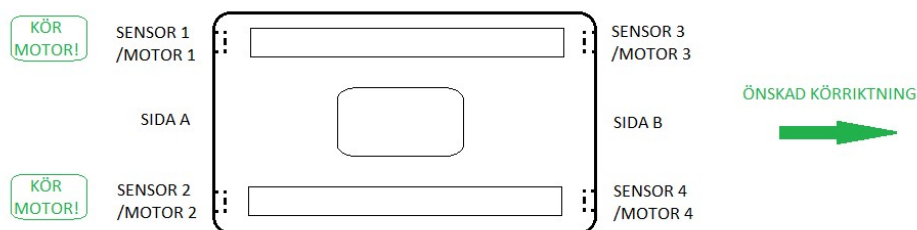
Funktionerna i programmet sker enligt följande:

- Start med vippströmbrytaren som beskrivs i avsnitt 4.2.6.
- Färjan avgör vilken riktning den skall köra beroende på vilken sida som har lägst värde på sina sensorer.
- Styrning behandlas mer i avsnitt 4.3.4.
- När färjan närmar sig kajen skall dess motorer stoppas en bit innan för att kunna röra sig fram till kanten långsamt.
- Efter ett tidsintervall skall motsatta riktningens motorer starta.

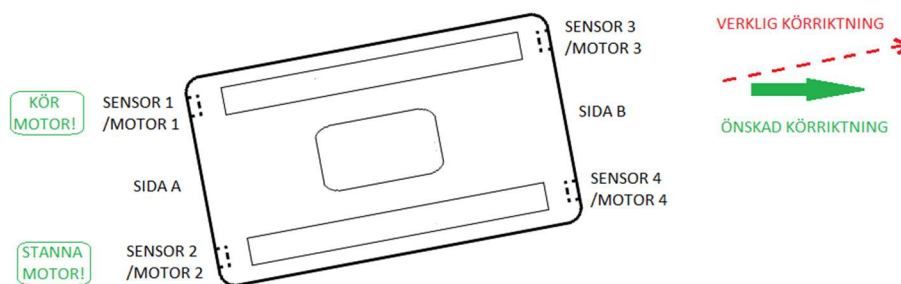
Detta kan tydligare ses i flödesschemat, figur 4.6.

### 4.3.4 Styrning

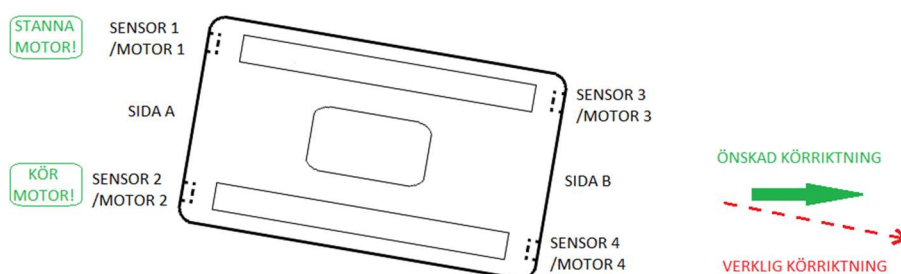
Styrning sker med hjälp av att stoppa och starta de fyra motorerna vid bestämd tidpunkt. När färjan kör till kajen på B sidan används motorerna på A sidan (d.v.s. motor 1 och motor 2 enligt figur 4.8). För att korrigera i det fall då färjan inte håller en rak kurs framåt åtgärdas detta genom att stanna den ena motorn och köra den andra tills de båda givarna har samma värde igen. Denna funktion har en tolerans på 8 cm vilket betyder att färjan inte börjar styra förrän den avviker mer än det angivna värdet gentemot varandra. Om färjan vrider sig för mycket åt vänster, skall motor 2 stanna och motor 1 köra, om färjan kör till kaj B som visas enligt figur 4.9. Om färjan däremot vrider sig åt höger skall motor 1 stanna och motor 2 köra som visas i figur 4.10. Det fungerar på samma sätt åt motsatt riktning fast då med motor 3 och 4. Motorerna som inte används för stunden är avstängda.



Figur 4.8, Färjan kör i den önskade riktningen därav driver båda motorerna på sida A.



Figur 4.9, Färjan har kommit ur kurs, starta motor 1 och stanna motor 2 för att åter komma till den önskade riktningen.



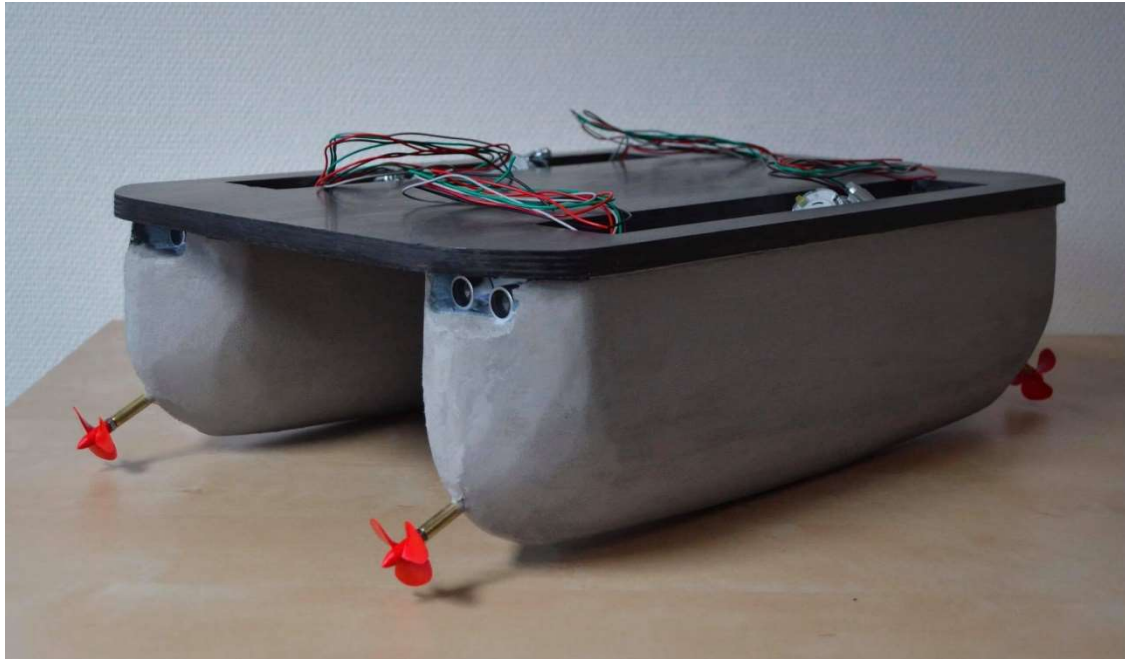
Figur 4.10, Färjan har kommit ur kurs, starta motor 2 och stanna motor 1 för att åter komma till den önskade riktningen.

Styrningen av en motor kan ske på olika sätt bland annat genom att ge en PWM- (Pulse Width Modulation) signal till motorn men det går även att använda sig utav en “throttle”-signal, en signal till ett hastighetsreglage som i det här fallet är förutbestämt. Det kan efterliknas med en potentiometer tack vare att man har ett värde från 0-100% där noll representerar värdet 1000 och 100% värdet 2000. Här valdes “throttle”-signalen med ett förutbestämt värde på 1300 vid start och 1000 vid stopp.

#### 4.4 Skrov

Skrovet som valdes var av typen katamaran. En katamaran är en typ av båt som har två parallella skrov, med en överbyggnad i mellan. Det finns ett flertal fördelar med denna typ av form på båt. En av dessa är att mer utrymme fås samt att båten blir mer stabil, då bredden är stor i förhållande till längden av skrovet. En annan fördel med en katamaran är att den är lättmanövrerad även i dåliga väderförhållanden som vid starka vindar. Skrovets utformning har även visats vara mer effektiv än ett monoskrov då motståndet minskas och det leder till ett lägre koldioxidutsläpp [7].

Dessa två skrov var från början fyra stycken cellplastskivor som limmades ihop två och två för att få den önskade formen, mer om materialet hittas i avsnitt 4.4.1. Efter montering krävdes arbete bestående av formning samt slipning både manuellt och med maskin.



*Figur 4.11, Färjans skrov innan montering av elektroniklådan.*

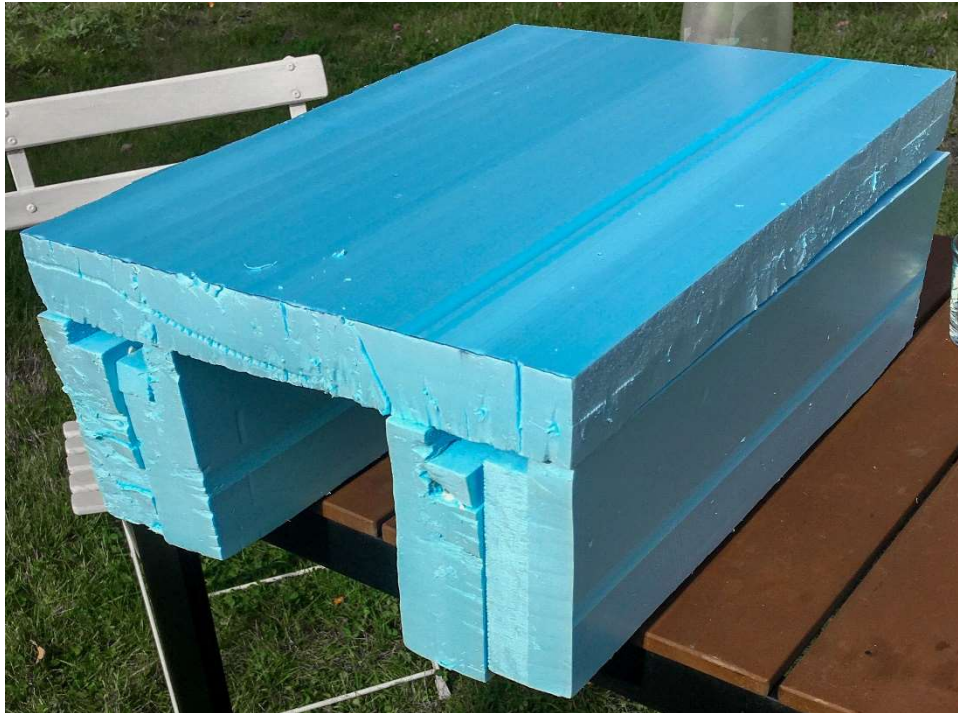
#### **4.4.1 Material**

Skrovet byggdes ihop genom att skivor av cellplast (XPS300) limmades samman för att få de egenskaper i form av bland annat den tjocklek som söktes. I figur 4.12 visas hur färjan såg ut från första början, efter att de limmats samman. På bilden kan man se en skiva cellplast ligga som en överbyggnad men den byttes därefter ut till en träskiva där luckor sågats ut för att enkelt kunna komma åt motorer samt andra elektronikkomponenter som placerades i skrovet. Varför just en träskiva användes var för att få tyngd på färjan för att kunna få den i önskat läge i vattnet d.v.s. propellrarna under vattenytan. En annan anledning till valet av skivan är att den är robust nog för att kunna såga luckor ur som nämnts tidigare.

Avståndssensorerna löddes fast på hållare i form av experimentkort och därefter monterades dessa fast på skrovet innan överbyggnationen monterades på plats.

När skrovet var färdigt gjordes färgtester för att ta reda på vad för färg som kunde användas för målning av skrovet. De egenskaper som krävdes var att det både skulle klara av vatten men den fick inte innehålla starka ämnen för att minimera risken att fräta cellplasten. Färgtesterna visade ett oväntat resultat och valet föll tillslut på en grå snickerifärg som uppfyllde kraven.

Justeringar gjordes efterhand som t.ex. extra lister runt luckorna på överbyggnaden som monterades fast för att kunna sätta dit skyddslock, mer om detta beskrivs i avsnitt 4.4.4 Skydd.



*Figur 4.12, Cellplasten innan arbete av form.*

#### **4.4.2 Motorfäste**

För att fästa motorerna i skrovet används en enklare träkonstruktion. Ur en träregel sågas bitar med rätt vinkel i förhållande till drivaxeln för att motorn ska kunna arbeta korrekt. För att få motorn att hamna i ett önskat läge slipas en form av motorn ut ur träet. Slangklämmor skruvas fast in i träfästet och används för att få motorn att sitta fast i det rätta läget vilket även ger en möjlighet att frigöra motorerna vid behov. I figur 4.13 nedan kan man se den urslipade formen samt slangklämman. Varje motor har var sitt motorfäste som är fastlimmat i skrovet. I figur 4.14 visas ett fäste där motorn satts i det rätta läget.



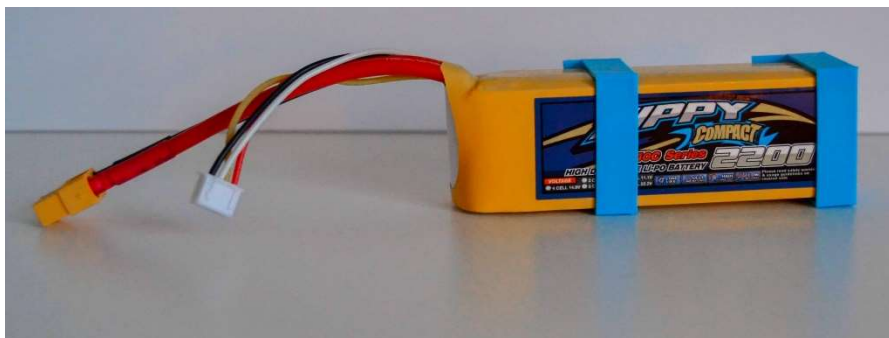
*Figur 4.13: Bild på motorfäste utan motor.*



*Figur 4.14: Bild på motorfäste med motor.*

#### **4.4.3 Batterihållare**

Batteriet som används är placerat i den behållare ovanpå färjan tillsammans med annan elektronik. Risker finns att batteriet rör sig och därmed kan skada övrig känslig elektronik vilket ger lösningen att ha en batterihållare enligt figur 4.15. Valet av fäste blev en 3D-printad hållare som ursprungligen är gjord för det aktuella batteriet. CAD-ritningar för 3D-utskriften, som hittas i bilaga C1, finns på hemsidan Thingiverse och är gjord av Harry Southworth [8].



*Figur 4.15, Batterihållare visas innan montering på skrov.*

#### **4.4.4 Skydd**

För att säkra de elektronikkomponenter som placeras ovanpå skrovet från vattenstänk så monterades en låda ovanpå skrovet. Komponenterna kunde därefter placeras på undersidan av locket. Ur lådan gjordes uttag för att kunna montera kablar genom från motorer samt sensorer in till de övriga komponenterna. När locket sattes fast på lådan blev komponenterna skyddade från vatten på effektivt sätt. Lådan stängdes samt öppnades enkelt med spännen enligt figur 4.5.

Även skydd i form av rektangulära plexiglas monterades för att förhindra att vattenstänk når motorerna, dessa visas i figur 4.16. De sattes fast med hjälp av kardborreband som fästes på

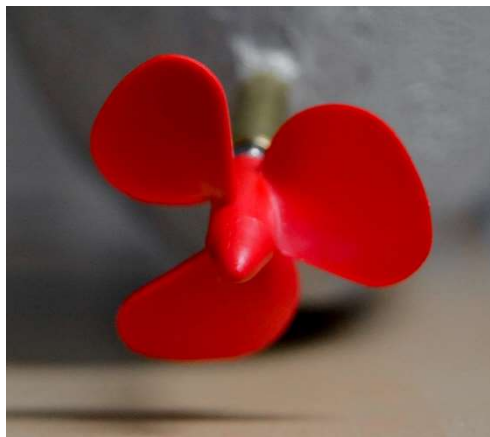
listerna på överbyggnadsskivan. Dessa skydd gjordes inte helt vattentäta utan fungerade mest som skydd mot vattenstänk.



*Figur 4.16, Skydd över elektronik samt motorutrymmen avplockad från skrovet*

#### **4.4.5 Drivaxel och propeller**

Till skrovet köptes färdiga paket innehållande drivaxel, propeller samt kardanknut. Drivaxel kapades i önskad längd för att överensstämja med skrov och motor. Propellerns blad har en diameter på 40 mm och de är gängade för enkel fastsättning i drivaxel. Kardanknuten fanns med för att skapa en säker koppling mellan motoraxel och drivaxel och är uppbyggd med en gummiled för att fungera även om det uppkommer viss rörelse samt vinkelskillnad. I figur 4.17 visas en av de fyra propellrarna som används.



*Figur 4.17, En av fyra propeller som sitter på färjan*

#### **4.4.6 ESC Hylla**

I figur 4.3 med elektronikkomponenter ovan kan även en trästruktur ses, vilket används som en hylla. För att få plats med alla komponenter i lådan byggdes hyllan för att kunna fästa ESC:na vilket gav dem sin egna plats, detta medförde att de inte var lösa och därmed inte riskerade att skada annan elektronik i komponentlådan. Hyllan byggdes av trä och monterades fast i lådan, med hjälp av skruv, över fyrvägskopplingen som nämnts tidigare. Hyllan hade även en funktion i att kunna vridas för att komma åt anslutningarna under.

## 5. Resultat

Resultatet blev en fungerande modell av en färja som uppfyllde huvudmålet att ta sig från en punkt till en annan per automatik. Även majoriteten av delmålen uppnåddes, däribland att få färjan att kunna gå åt båda håll men även delmålet med att få fram en lämplig styrning uppnåddes. Delmålet som fanns om att färjan med hjälp av sensorer skulle kunna undvika föremål uppnåddes delvis då sensorerna kan registrera om det kommer något framifrån och styra undan utifrån den informationen, men inte registrera om något kommer från sidan eller bakifrån. Det sista delmålet om tilläggning var från start ett mindre prioriterat mål som i slutänden prioriterades bort på grund utav tidsbrist.

Testkörningarna visade att färjan var stabil samt att den skulle ur stabilitetssynpunkt klara av en relativt kraftig sjögång i förhållande till modellens storlek. Vid den slutgiltiga testkörningen visade färjan att styrning fungerade enligt målen. Med hjälp av modellbyggandet och tester kunde en grund för diskussion om möjlig verklig färja inledas.

## 6. Diskussion

Här presenteras resultatet samt diskussion på hur man skulle kunna möjliggöra projektet i ett större format än prototypen. Även egna ord och åsikter finns med här.

### 6.1 Diskussion av färdig modell

Vid testkörningen observerades att motorerna var kraftfulla, vilket resulterade i att färjan fick en hög hastighet då fartreglaget redan var på ett lågt värde. I efterhand hade det varit optimalt att ha en större testanläggning för att få mer uppgifter som en grund att bygga en diskussion på. För att eliminera problemet med den höga hastigheten, utan att byta motorer, kan vissa justeringar göras som t.ex. byta ut propellrarna mot propellrar med mindre bladarea, ändra antalet blad eller ha blad med annan stigning. Vilket medför att motståndet samt propellerkraften ändras som i sin tur ändrar dragkraften och färjan driver framåt långsammare.

Ett antagande som gjordes innan modellen var färdigställd var att färjan skulle driva alldeles för långsamt eller att motorerna inte skulle vara kraftfulla nog att driva, vilket vid testkörning visade sig inte vara ett bekymmer. Skrovet samt drivaxel var täta nog för att inte läcka in vatten. Även styrningen samt sensorerna fungerade som önskat under testkörningen.

### 6.2 Modell till verklig produkt

För att få modellen till en verklig produkt så krävs det att se igenom varje del av färjan och dess förmåga att befinna sig ute på vatten.

#### 6.2.1 Skrov

Att använda sig utav ett katamaranskrov är något som fungerar bra och är även något som används inom färja i dagsläget, enligt Sea Transport Solutions [7]. En stor fördel är att modellen kan köra åt vardera riktning utan att behöva vända vilket fungerar för en storskalig personfärja. Fördelen med att ha två skrov är att det finns utrymme för motorer samt elektronik hos en större färja och att övrigt utrymme används genom att frakta passagerare. Det går även att bygga en skyddad plats till passagerarna, mot väder och vind, som liknar den låda som använts i modellen för att ha passagerare i istället för elektronik.

#### 6.2.2 Drivning samt styrning

Att driva en färja med fyra motorer som ligger i vattnet hela tiden fungerar utmärkt i modellen med tanke på att motorerna är väldigt kraftfulla. Vid större format hade testning fått göras, vilket hade visat om det är optimalt att ha fyra motorer i vattnet på samma gång eller om det skapar för mycket motstånd och att energi går åt i onödan. Detta hade kunnat åtgärdas genom att hissa upp de motorer som ej används eller ha propellrar som är effektivt åt båda riktningar, då hade man kunnat starta motorer sida A medan sida B "backar" under drift.

Vid den verkliga färjan hade man kunnat använda motorerna för att backa eller förflytta sig vid hinder. Genom att ha fyra motorer undviks det att vända färjan under drift, detta sparar på resurserna tid samt energi.

#### 6.2.3 Sensorer

Sensorer som använts under projektet är begränsade att nå upp till 4,5 m vilket betyder att dessa inte är gynnsamma att använda sig utav vid en större färja som skall färdas mellan längre sträckor. Även andra typer av sensorer bör användas för att detektera om något närmar

sig färjan och att den ska agera på detta genom att antingen köra eller stanna alternativt backa eller styra undan. Ett radarsystem eller liknande hade eventuellt fungerat till detta ändamål.

Eftersom modellen av färjan enbart körde kortare avstånd togs det inte hänsyn till om färjan hade hamnat rätt eller ej då det istället antogs. Vad som hade krävts vid en större färja är någon typ av anläggning, sensorer eller liknande som känner av om färjan håller rätt kurs i förhållande till den aktuella kajen.

#### **6.2.4 Passagerarvänlighet**

Om en personfärja automatiseras kan även en smidig och effektiv påstigning samt avstigning automatiseras. När färjan har lagt till vid kajen kan programmerade grindar finnas på färjan som öppnas och stängs vid på- och avstigning.

#### **6.2.5 Materialval**

Materialvalet för en verklig färja skiljer sig fullständigt från valet av material på modellfärjan. För att göra ett bra materialval krävs det att man vet vad som är viktigt och vad som söks. För modellen söktes ett material som förmår vatten, har rätt flytegenskaper och är enkel att arbeta med. Modellen i sig behövde inte klara stora påfrestningar som väder och långa stunder i vatten, modellen var även skyddad då den inte utsattes för situationer som riskerade att ge skador på skrovet vilket en verklig färja bör klara av på grund av säkerhetsskäl.

#### **6.2.6 Lagar och säkerhet**

Inga lagar togs hänsyn till under framtagning av modellfärjan. Vid utvecklandet av en verklig färja krävs det att lagar och regler följs på samma sätt som för andra fartyg. Lagar är till för att säkerheten inte ska äventyras, varken för passagerare i färjan eller de som kan påverkas runt omkring. De föreskrifter som finns kan vara allt från att färjan måste ha rätt ljus i form av lanternor eller lampor, till rätt nödutrustning vid en olycka. I dagsläget finns även föreskrifter på antal besättningsmän som krävs för olika fartyg vilket blir en intressant fråga vid automatisering. Mer om detta i avsnitt 6.5.

### **6.3 Förslag på vidareutveckling på modell**

Just nu används de fyra sensorerna som är till för att köra fram respektive bak även till att undvika föremål, vilket inte är det effektivaste om föremål skulle närma sig från sidan av båten i en långsammare fart. Detta kan eventuellt åtgärdas med fler sensorer eller givare som kan läsa av närgående föremål alternativt med radar som ser andra fartyg i vattnet.

Motorerna var något som oroade i början då det fanns farhågor att propellrarna som sitter i motsatt riktning till färdriktning skulle bromsa in färjan, vilket skulle leda till att färjan inte orkade drivas framåt. Detta var inget problem då motorerna var oerhört kraftfulla och istället gick alldeles för fort. Vilket enkelt går att åtgärda med antingen mindre kraftfulla motorer eller andra sorters propellrar som diskuterats i avsnitt 6.1.

Tidsplanen som gjordes i början av projektet ändrades aningen under tidens gång vilket går att se i bilaga A.2. Det fanns ett önskemål att göra mätningar inom reglerteknik med Simulink, Matlab. Men med tanke på rådande situation med pandemin fick vi prioritera annorlunda med de resurser vi själva hade tillgång till, därav hanns inte detta med. De mätningar som var tänkta att göra var att finna den rätta stopptiden för när färjan skulle stoppas till att den gled fram längs vattnet fram till kajen och till dess att den igen skulle starta för att köra åt motsatt håll efter att färjan hämtat upp tänkta passagerare. Signalen som

stoppar färjan är just nu förbestämd i programkoden. Detta är något som skulle kunna göras vid en vidareutveckling av modellen.

## 6.4 Miljö och hållbar utveckling

Möjligheten för att transporteras över vatten har och är ett vanligt sätt att ta sig från ett ställe till ett annat. Att fler och fler istället väljer att åka kommunalt än att ta bilen samt andra färdmedel är en fördel för koldioxidutsläppet (CO<sub>2</sub>). Detta medför även att man kan ta den närmaste vägen än att behöva ta omvägen med bilen, detta minskar även köerna samt att den trafik som åker på vägarna flyter på mer och därav, även dem, minskar sina utsläpp.

Färjan är elektronisk vilket medför att den går på batteri, detta bidrar till att möjligheten att konstruera laddstationer finns samt att ha solceller eller att välja att driva färjan på annat miljövänligt drivmedel som t.ex. biogas. En fördel med denna färja är att den inte kräver extra energi att vända på färja utan kan köra i rätt riktning oavsett var färjan ska köra.

Som avsnitt 4.4 nämnde är ett monoskrov mer effektkrävande med tanke på dess form, vilket leder till att formen av en katamaran ger mindre motstånd i vattnet som i sin tur leder till att motorerna använder sig utav mindre energi [7]. Detta är en fördel för koldioxidutsläppet.

Med hjälp av underhåll och inspektion, vilket även är ett krav enligt fartygssäkerhetslagen [9], minskar man de akuta felen som kan uppstå. Har man istället planerat stopp går det att beställa varor utifrån detta, dessa varor kan då transporteras på ett miljövänligare sätt än när det är akut. Man minskar även en oro för passagerarna då färjan har mindre risk att råka ut för fel vid planerat underhåll.

## 6.5 Etik och moral

I nuläget automatiseras allt fler fordon och även flygplan runt om i världen. Men vem är det som bär ansvaret när det går fel? Är det försäljaren, konsumenten eller ingetdera? Enligt Fartygssäkerhetslagen som hittas hos Sveriges Riksdag [9] står det att det krävs en befälhavare på båten som har ansvaret för sin egen men även andras säkerhet. I det här fallet med en självkörande passagerarfärja hade det varit gynnsamt att ha en befälhavare som ser till att allt går rätt till på färjan men också att passagerare är säkra under på-och avstigning. Befälhavaren hade även kunnat räkna antal passagerare som en säkerhetsåtgärd om inte räknarsystemet skulle fungera och maxlasten skulle uppnås, vilket finns lagat på enligt Fartygssäkerhetslagen [9].

Automatisering av ett fordon ger möjlighet att t.ex. få tätare avgångar då personal inte behövs tas hänsyn till. Dock finns en etisk fråga, även om en befälhavare skulle finnas på plats utgår andra arbeten som en kapten och eventuell styrman men samtidigt försvinner den mänskliga faktorn som gör att passagerare förmodas få en säkrare resa. Sedan kommer andra arbeten att finnas kvar som besiktning, underhåll och inspektion vilket också är ett krav enligt Fartygssäkerhetslagen [9].

## 7. Slutsats

Syftet med projektet var, som nämnt i avsnitt 1.2, att ta reda på hur möjligheterna ser ut att automatisera en färja genom att bygga en mindre modell av en självkörande passagerarfärja och därmed få fram ett underlag för en diskussion. Syftet har uppnåtts och även satts igång en diskussion kring ämnet efter att modellen har framställts.

Det finns fortfarande mer omfattande frågor samt testning som bör ske om man ska påbörja processen att producera en autonom färja vilket inte tagits hänsyn till här.

Frågeställningarna besvarades och diskussioner skapades för att koppla arbetet från modell till framtida möjligheter av en verklig autonom färja.

Huvudmålet har uppnåtts genom framtagning av en fungerande modellfärja och har öppnat upp för en diskussion med hjälp av att ha arbetat med de delmål som sattes upp.

Sammantaget har projektet blivit som planerat.

## Referenser

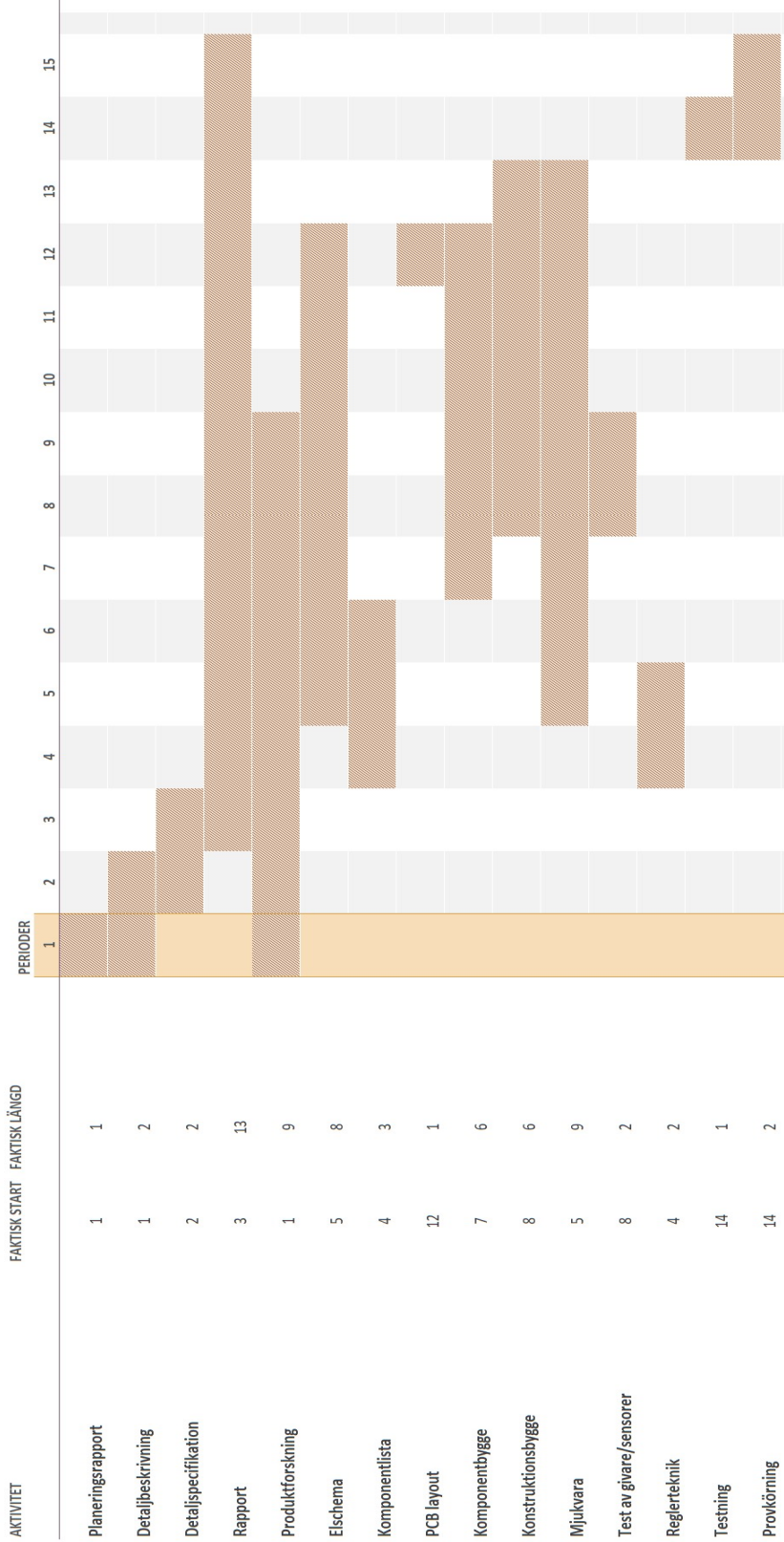
- [1] Arduino, 2020. Arduino uno rev3 SMD [online]. Tillgänglig: <https://store.arduino.cc/arduino-uno-rev3-smd> [2020-10-29]
- [2] A. Abdul Jaabar, 2019. Ultrasonic sensor HC-SR04 with Arduino tutorial. [online] Tillgänglig: <https://create.arduino.cc/projecthub/abdularbi17/ultrasonic-sensor-hc-sr04-with-arduino-tutorial-327ff6>, [2020-10-31]
- [3] Electro-mobile Scandinavia AB, 2017. Borstad eller borstlös motor. [online] Tillgänglig [https://www.electro-mobile.se/wp-content/uploads/2017/02/Borstad-eller-borstl%C3%B6s-motor\\_a.pdf](https://www.electro-mobile.se/wp-content/uploads/2017/02/Borstad-eller-borstl%C3%B6s-motor_a.pdf) [2020-10-29]
- [4] Turnigy, N/A. Manual of Speed Controller for Brushed Motor [online]. Tillgänglig: <https://cdn-global-hk.hobbyking.com/media/file/t/g/tgy30.jpg> [2020-12-04]
- [5] R, Nils, 2013. In-line Q11 Switch box [online]. Tillgänglig: <https://www.thingiverse.com/thing:127231> [2020-11-27]
- [6] Electronicshub, 2017. What are the differences between Raspberry Pi and Arduino? [online]. Tillgänglig: <https://www.electronicshub.org/raspberry-pi-vs-arduino/> [2020-12-04]
- [7] Sea Transport Solutions, 2018. Ferry Design: Monohull Versus Catamaran [online]. <https://www.seatransport.com/ferry-design-monohull-versus-catamaran/> [2020-12-05]
- [8] Southworth, H, 2015. FPV Ground Station Battery And Receiver Holder RC305. [online]. Tillgänglig: <https://www.thingiverse.com/thing:630841> [2020-11-19]
- [9] Infrastruktursdepartimentet, 2003. Fartygssäkerhetslag (2003:364) [online]. Tillgänglig: [https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/fartygssakerhetslag-2003364\\_sfs-2003-364](https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/fartygssakerhetslag-2003364_sfs-2003-364) [2020-12-06]

# Projektplanerare

Självvärnande fjärra

Markerad period: 1

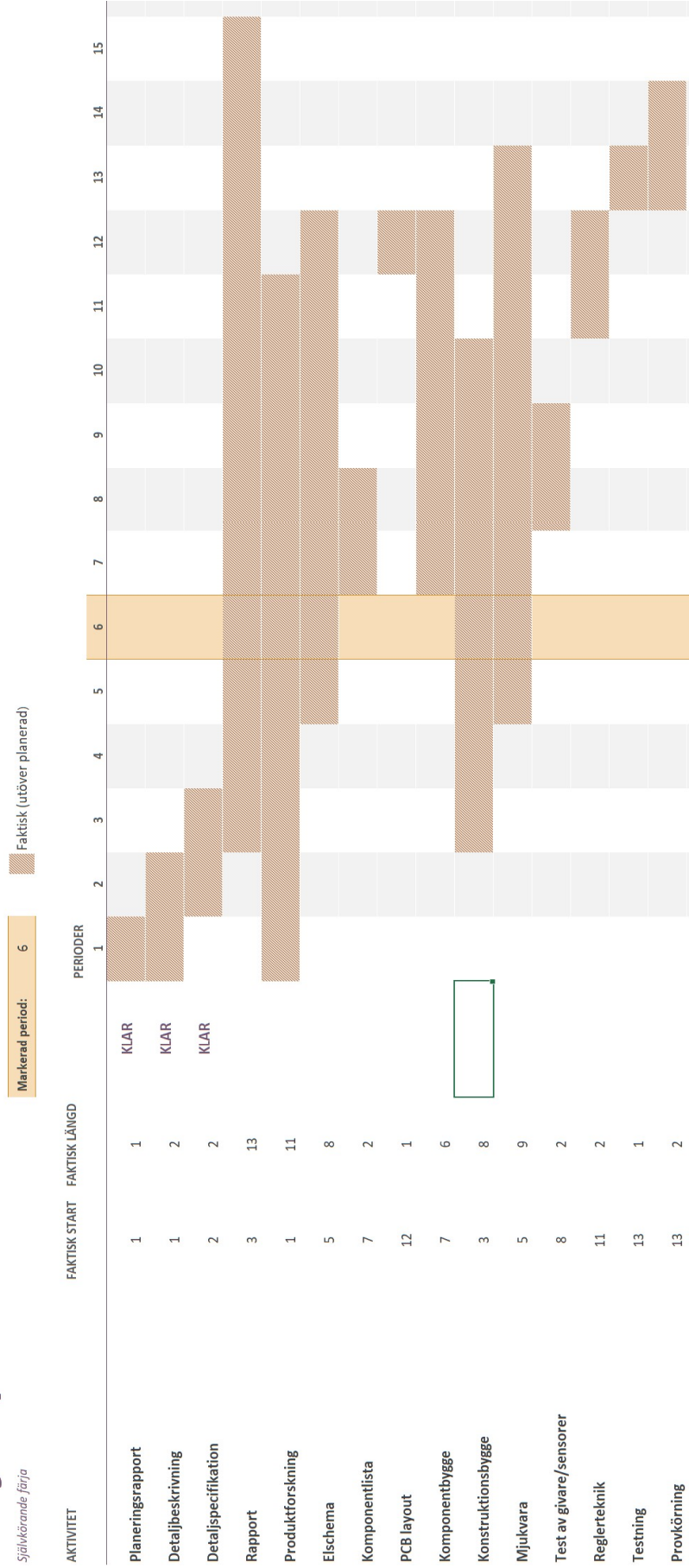
Faktisk (utöver planerad)



# Projektplanerare

*Självlärande färdiga*

## A.2 Bilaga Gantt-schema omgjord

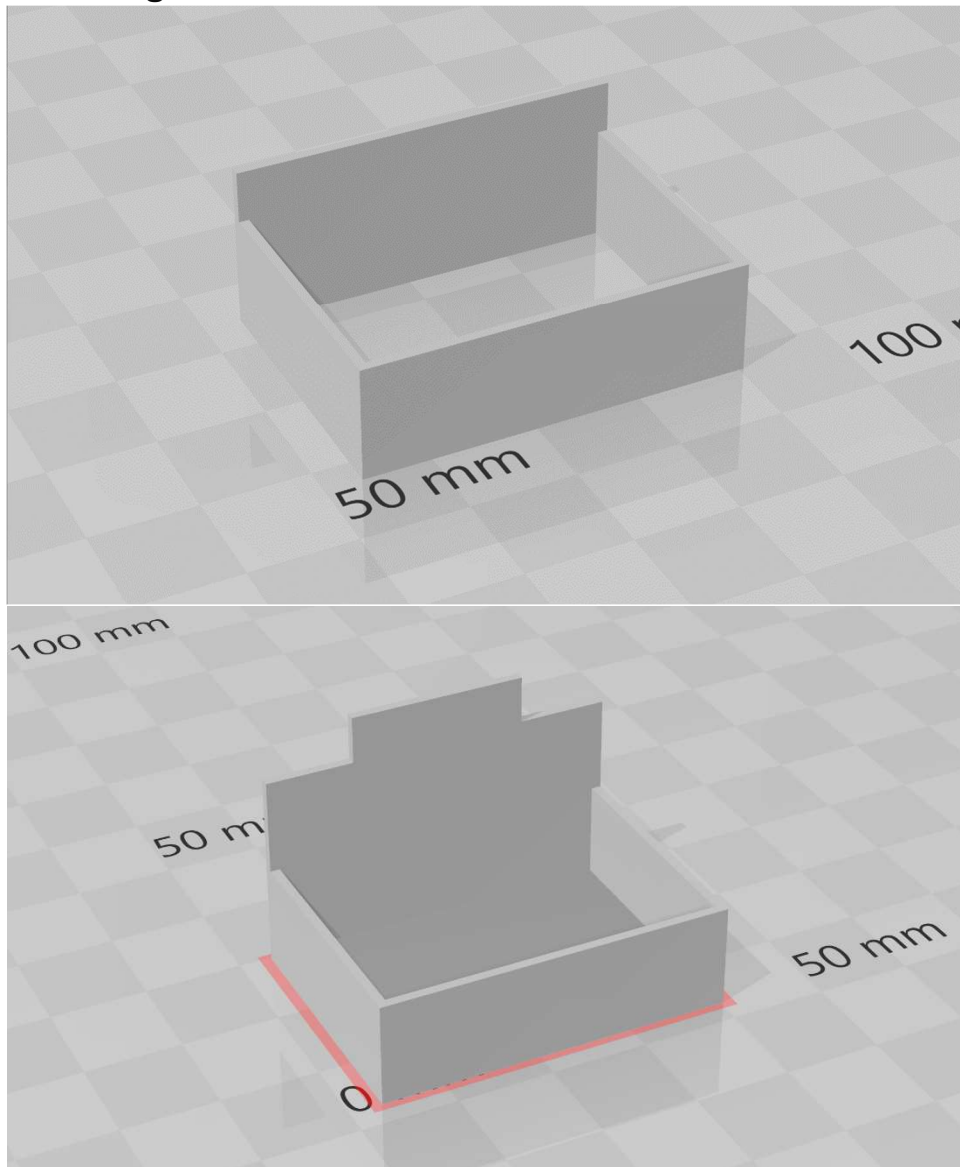




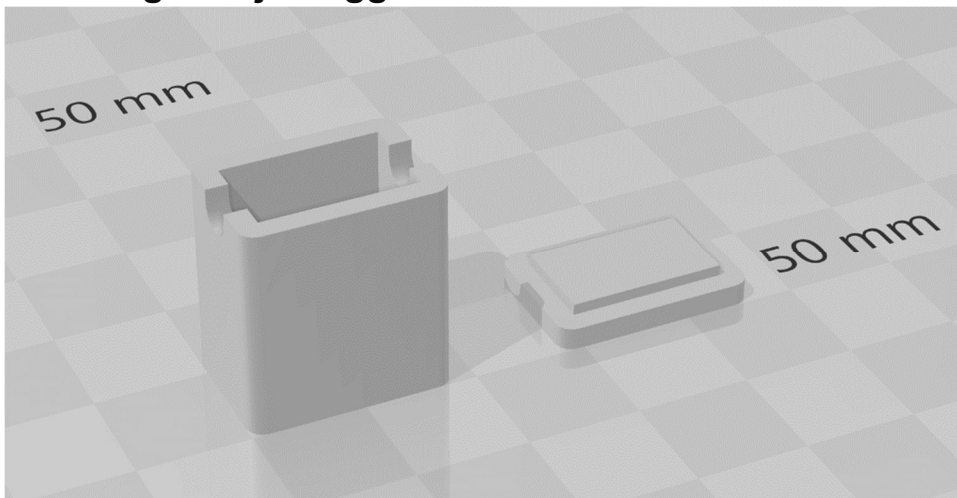
## C Bilaga 3D ritningar

Batterihållare samt hölje för vippströmbrytaren i 3D format.

### C.1 Bilaga Batterihållare



## C.2 Bilaga Hölje Toggle Switch



## D Bilaga programkod

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

const int TRIG_PIN1 = 2;      //Sensor1: TRIG D2, ECHO D13
const int ECHO_PIN1 = 13;    //Sensor1: TRIG D2, ECHO D13
const int TRIG_PIN2 = 12;    //Sensor2: TRIG D12, ECHO D8
const int ECHO_PIN2 = 8;     //Sensor2: TRIG D12, ECHO D8
const int TRIG_PIN3 = 7;     //Sensor3: TRIG D7, ECHO D6
const int ECHO_PIN3 = 6;     //Sensor3: TRIG D7, ECHO D6
const int TRIG_PIN4 = 5;     //Sensor4: TRIG D5, ECHO D4
const int ECHO_PIN4 = 4;     //Sensor4: TRIG D5, ECHO D4
const int BUTTON_PIN = A1;   //The number of the switch pin

long duration1, duration2, duration3, duration4;
int cm1, cm2, cm3, cm4;     //Value of the sensors
int currentState;          //Reading from the switch pin

Servo motor1;              //Servo library, Motor 1: PIN = 10
Servo motor2;              //Servo library, Motor 2: PIN = 3
Servo motor3;              //Servo library, Motor 3: PIN = 11
Servo motor4;              //Servo library, Motor 4: PIN = 9

void setup() {
  //Setup motor:
  motor1.attach(10);        //Motor 1: PIN = 10
  motor2.attach(3);         //Motor 2: PIN = 3
  motor3.attach(11);        //Motor 3: PIN = 11
  motor4.attach(9);         //Motor 4: PIN = 9

  //Setup sensor:
  pinMode(TRIG_PIN1,OUTPUT); //set pin 2 as output, sensor1
  pinMode(ECHO_PIN1,INPUT);  //set pin 13 as input, sensor1
  pinMode(TRIG_PIN2,OUTPUT); //set pin 12 as output, sensor2
  pinMode(ECHO_PIN2,INPUT);  //set pin 8 as input, sensor2
  pinMode(TRIG_PIN3,OUTPUT); //set pin 7 as output, sensor3
  pinMode(ECHO_PIN3,INPUT);  //set pin 6 as input, sensor3
  pinMode(TRIG_PIN4,OUTPUT); //set pin 5 as output, sensor4
  pinMode(ECHO_PIN4,INPUT);  //set pin 4 as input, sensor4

  //setup switch
  pinMode(BUTTON_PIN, INPUT_PULLUP); //Set pin A1 as input pull-up

  Serial.begin(9600);       //Opens serial port, sets data rate to 9600 bps
}
```

```

int sensor1() { //Function sensor1
  // The sensor is triggered by a HIGH pulse of 10 or more microseconds.
  // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
  digitalWrite(TRIG_PIN1, LOW); //Clear TRIG_PIN1
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN1, HIGH); //Sets TRIG_PIN1 to HIGH for 5 micro seconds
  delayMicroseconds(5);
  digitalWrite(TRIG_PIN1, LOW);
  duration1 = pulseIn(ECHO_PIN1, HIGH); //Reads ECHO_PIN1, returns sound wave travel time
  //(Divide by 29.1 or multiply by 0.0343)
  cm1 = ((duration1)/2) / 29.1; //Converts the value from the sensor into cm
  //Serial.print("Givare1: "); //Write "Sensor1: ", Used for trouble shooting
  //Serial.println(cm1); //Write cm value, Used for trouble shooting
  delay(50); //Delay 50 ms
  return cm1; //Return the value in cm1
}

int sensor2() { //Function sensor2
  // The sensor is triggered by a HIGH pulse of 10 or more microseconds.
  // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
  digitalWrite(TRIG_PIN2, LOW); //Clear TRIG_PIN2
  delayMicroseconds(5);
  digitalWrite(TRIG_PIN2, HIGH); //Sets TRIG_PIN2 to HIGH for 10 micro seconds
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN2, LOW);
  duration2 = pulseIn(ECHO_PIN2, HIGH);
  cm2 = ((duration2)/2) / 29.1; //Converts the value from the sensor into cm
  //Serial.print("Givare2: "); //write "Sensor2: ", Used for trouble shooting
  //Serial.println(cm2); //write cm value, Used for trouble shooting
  delay(150); //delay 150 ms
  return cm2; //Return the value in cm2
}

```

```

int sensor3() { //Function sensor3
  // The sensor is triggered by a HIGH pulse of 10 or more microseconds.
  // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
  digitalWrite(TRIG_PIN3, LOW); //Clear TRIG_PIN3
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN3, HIGH); //Sets TRIG_PIN3 to HIGH for 5 micro seconds
  delayMicroseconds(5);
  digitalWrite(TRIG_PIN3, LOW);
  duration3 = pulseIn(ECHO_PIN3, HIGH);
  cm3 = ((duration3)/2) / 29.1; //Converts the value from the sensor into cm
  //Serial.print("Sensor3: "); //write "Sensor3: ", Used for trouble shooting
  //Serial.println(cm3); //write cm value, Used for trouble shooting
  delay(50); //delay 50 ms
  return cm3; //Return the value in cm3
}

int sensor4() { //Function sensor4
  // The sensor is triggered by a HIGH pulse of 10 or more microseconds.
  // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
  digitalWrite(TRIG_PIN4, LOW); //Clear TRIG_PIN4
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN4, HIGH); //Sets TRIG_PIN4 to HIGH for 5 micro seconds
  delayMicroseconds(5);
  digitalWrite(TRIG_PIN4, LOW);
  duration4 = pulseIn(ECHO_PIN4, HIGH);
  cm4 = ((duration4)/2) / 29.1; //Converts the value from the sensor into cm
  //Serial.print("Sensor4: "); //write "Sensor4: ", Used for trouble shooting
  //Serial.println(cm4); //write cm value, Used for trouble shooting
  delay(50); //delay 50 ms
  return cm4; //Return the value in cm4
}

```

```

void loop() {
//Motors to 0%, stopp
  motor1.write(1000);
  motor2.write(1000);
  motor3.write(1000);
  motor4.write(1000);

while(1) {
  currentState = digitalRead(BUTTON_PIN); //Read switch
  if(currentState == LOW) { //If switch is on continue
  Serial.println("The button is on"); //Write "The button is on", used for trouble shooting
  while(!digitalRead(BUTTON_PIN)) { //Run while loop until the button is off
  //Start motors on B side, drive to dock A
  if(sensor1() > sensor3()) {
    while(!digitalRead(BUTTON_PIN)){ //Run while loop until the button is off
      currentState = digitalRead(BUTTON_PIN);
      //Sensor2 > sensor1 (tolerance 8): Stop motor 3, start motor 4. (drive to dock A, start motors B side)
      if(abs(sensor2()-sensor1()) >= 8) {
        motor3.write(1000); //Stop motor3
        motor4.write(1300); //Start motor4
        Serial.print("Stop motor 3, ");
        Serial.println("Start motor 4");
      }
      //Sensor 1 > sensor2 (tolerance 8): stop motor 4, start motor 3 (drive to dock A, start motors B side)
      else if(abs(sensor1()-sensor2()) >= 8) {
        motor3.write(1300); //start motor3
        motor4.write(1000); //stop motor4
        Serial.println("Start motor 3");
        Serial.print("stop motor 4, ");
      }
    }
  }
  else {
    Serial.println("Start motors on side B");
    motor3.write(1300); //start motor3
    motor4.write(1300); //start motor4
    if(sensor1() && sensor2() < 50) { //if it is less than 50 cm left to dock, stop motor 3 and 4.
      Serial.println("stop");
      motor3.write(1000); //stop motor3
      motor4.write(1000); //stop motor4
      delay(2000); //Delay, 2 seconds
      break;
    }
  }
}
}
}

```

