



Telepresence Robot

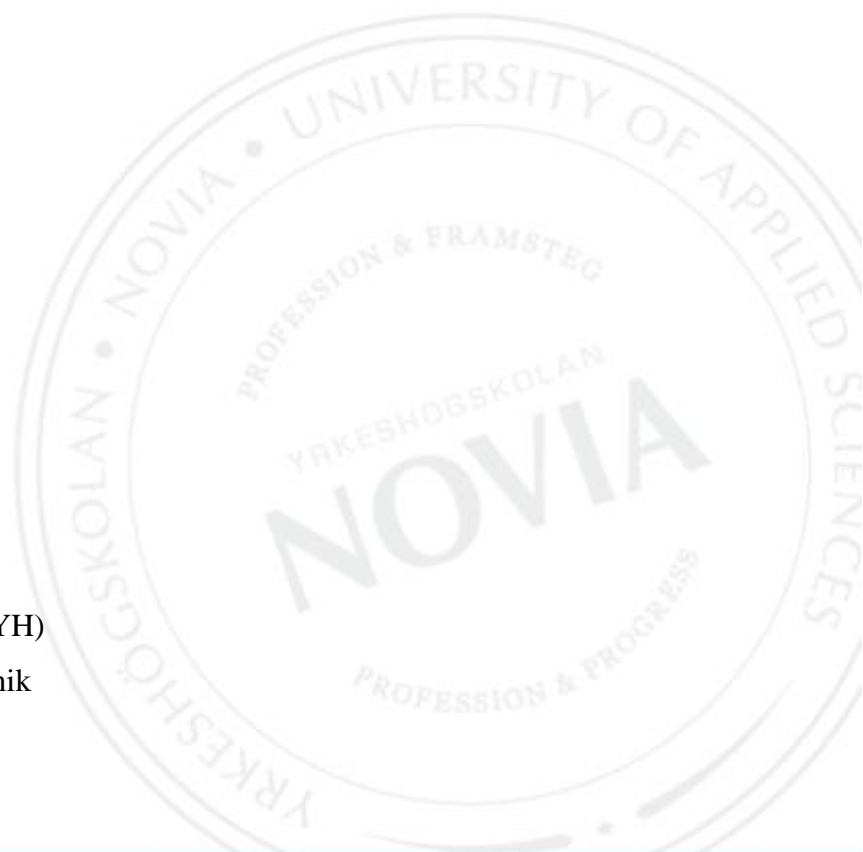
Jonatan Klemets

Sebastian Siffrén

Examensarbete för ingenjörsexamen (YH)

Utbildningsprogrammet för elektroteknik

Vasa, 2011



EXAMENSARBETE

Författare: Jonatan Klemets och Sebastian Siffrén
Utbildningsprogram och ort: Elektroteknik, Vasa
Inriktningsalternativ: Automationsteknik
Handledare: Jonas Waller

Titel: *Telepresence robot*

Datum: 12.12.2011

Sidantal: 71

Bilagor: 8

Abstrakt

Innan webbkonferenser var möjliga var det svårt att ha möten utan att någon av parterna måste resa. Eftersom resor är dyra och tar tid jobbar man hela tiden med att förbättra möjligheterna till webbkonferenser. Den senaste tekniken inom webbkonferenser är telepresence roboten. Telepresence roboten ger användaren möjlighet att vara närvarande fast han befinner sig på en avlägsen plats genom att kommunicera genom den. Användaren kan även förflytta sig med hjälp av roboten på fjärrplatsen.

Målet med detta examensarbete var att på Yrkeshögskolan Novias begäran utveckla en prototyp av en telepresence robot. Roboten ska kunna styras varifrån som helst i världen och ge användaren goda kommunikationsmöjligheter. För att uppnå målet gjordes en del forskning inom ämnet för att kunna göra de lämpligaste komponentvalen för just detta ändamål. Detta examensarbete resulterade i en fungerande prototyp av en telepresence robot.

Språk: svenska

Nyckelord: telepresence robot, webbkonferens, LabVIEW

Tillgängligt: I Theseus.fi eller vid Tritonia, Vasa vetenskapliga bibliotek.

BACHELOR'S THESIS

Author: Jonatan Klemets and Sebastian Siffrén
Degree Programme: Electrical Engineering
Specialization: Automation Technology
Supervisor: Jonas Waller

Title: *Telepresence robot*

Date: 12.12.2011

Number of pages: 71

Appendices: 8

Abstract

Before web conferencing was possible it was hard to have meetings to which none of the participants had to travel. Due to the fact that travels are expensive and time-consuming, better opportunities for web conferences are requested. The latest technology in web conferencing is the telepresence robot. A telepresence robot gives the user opportunity to attend a meeting although he is not located where the meeting takes place. Not only can the user of this robot communicate through it, but he can also move around on the remote site.

The goal with this thesis was to develop a prototype of a telepresence robot, a task which was commissioned by Novia University of Applied Sciences. From anywhere in the world it will be possible to control this robot, which will give the user good communication opportunities. To reach the goal some research on the topic was done to be able to make the most suitable choices of components serving this purpose. This thesis work resulted in a working prototype of a telepresence robot.

Language: Swedish Key words: telepresence robot, web conferencing, LabVIEW

Filed at: Theseus.fi or Tritonia, Academic library of Vaasa

Innehållsförteckning

1 Inledning	1
1.1 Uppdragsgivare	1
1.2 Syfte	1
2 Vad är en robot?	2
2.1 Stationära och mobila robotar	4
2.2 Autonoma och fjärrstyrda robotar.....	4
2.3 Telepresence robotar	6
3 Tillvägagångssätt	8
3.1 Plattformsväl	8
3.1.1 Robotdammsugare.....	9
3.1.2 Radiostyrdbil	9
3.1.3 Färdiga robotplattformer	9
3.1.4 Egen plattform.....	10
3.1.5 Slutsats	10
3.2 Framdrivningssätt	11
3.2.1 Differentialstyrning	12
3.2.2 Ackerman-styrning.....	13
3.2.3 Tricycle-styrning	14
3.2.4 Synkrostyrning	15
3.2.5 Mecanum-hjuldrivning.....	16
3.2.6 Slutsats	16
3.3 Design	17
3.4 Motorer	18
3.4.1 PMDC- motorer (med borstar).....	18
3.4.2 Borstlösa DC-motorer	20
3.4.3 Stegmotorer	21
3.4.4 Servomotor	22
3.4.5 Motordimensionering	24
3.4.6 De valda motorerna	24
3.4.7 PWM-styrning.....	26
3.5 Strömkälla	28
3.5.1 Zink	28
3.5.2 Alkaliska.....	28
3.5.3 Nickel-kadmium.....	29
3.5.4 Nickel-metallhydrid	29
3.5.5 Litium och litium-jon	29
3.5.6 Bly batteri.....	30
3.5.7 Det valda batteriet	30
3.6 Centralenhet	31
3.7 I/O- kort	31
3.7.1 LabJack U3-HV.....	32
3.8 Övrig utrustning	33

4 Elektronik.....	34
4.1 Drivkretsar för motorer och logikkrets	34
4.1.1 H-brygga.....	34
4.1.2 Komponenter	36
4.1.3 Design av H-brygga	37
4.2 Batterikapacitetskrets.....	39
4.3 Brytarkrets för huvudbrytare	40
4.4 Brytarkrets för datorladdare	40
4.5 f/V-krets	41
5 Programvara för styrning.....	45
5.1 LabVIEW	45
5.2 Programmet.....	46
5.2.1 Subrutiner	46
5.2.2 Tangentbordsstyrning.....	48
5.2.3 Huvudprogrammet	49
5.2.4 Frontpanelen.....	52
5.2.5 LabVIEW Web Server	53
6 Programvara för kommunikation.....	54
6.1 TeamViewer.....	55
7 Funktionssätt.....	56
8 Slutresultat	60
8.1 Svårkörd.....	60
8.2 Problem med framkomligheten	61
8.3 För många program i användning	61
8.4 Kåpa och basstation saknas.....	61
9 Diskussion.....	62
Klemets diskussion	62
Siffréns diskussion	64
Källförteckning	65
Bilagor	71

1 Inledning

Det finns ett stort behov idag för både privatpersoner och företag att kommunicera fritt med varandra oberoende av var i världen man befinner sig. En metod som används friskt för att kunna kommunicera är internetkonversationer, eller webbkonferenser.

En vidareutveckling på webbkonferenser är telepresence robotar. Telepresence robotar ger användarna både möjlighet att kommunicera med människor på en avlägsen plats och att förflytta sig, vilket ger en känsla av att verkligen befinna sig på fjärrplatsen.

Detta examensarbete går ut på att bygga en prototyp av en telepresence robot. I rapporten framkommer lite fakta om robotar och hur man gått tillväga under projektets olika arbetsmoment. Det framkommer även en del undersökningar som man var tvungen att göra innan man kunde börja tillverka roboten. Dessa undersökningar ska förhoppningsvis vara till hjälp i framtiden om man vidareutvecklar prototypen. I rapporten presenteras även olika alternativ av komponentval och design. Slutligen presenteras slutresultatet och förslag till förbättringar diskuteras.

1.1 Uppdragsgivare

Yrkeshögskolan Novia har fungerat som uppdragsgivare för detta examensarbete och har gett i uppdrag att utveckla en prototyp av en telepresence robot. Jonas Waller vid Yrkeshögskolan Novia har fungerat som handledare.

1.2 Syfte

Syftet med examensarbetet är att utveckla en prototyp av en telepresence robot. Målet är att roboten ska ha god mobilitet och goda kommunikationsmöjligheter. Roboten kommer inte att vara autonom i någon grad, dvs. intelligent, utan den kommer i stället att vara fjärrstyrd från en dator. Till projektet hör även undersökningar av komponentval.

2 Vad är en robot?

En människa kan ses som en nästan perfekt maskin. Människan är intelligent och har förmågan att förflytta sig och flytta på saker i sin omgivning. Människan har även förmågan att kunna uppfatta saker och agera därefter som t.ex. att äta när hon är hungrig eller att skydda sig om hon känner sig hotad. /6/

I årtal har forskare försökt skapa maskiner som efterliknar människans beteende och funktioner, men man har fortfarande ännu inte lyckats helt. En robot är med andra ord en maskin som på ett människolikt sätt utför sina uppgifter. /6/

Man kan fråga sig vad en maskin måste ha för egenskaper för att kunna kallas robot. Detta är en fråga som inte har något rätt eller fel svar eftersom det inte finns någon standarddefinition på en robots krav. Människor har även en mängd olika synpunkter på vilka egenskaper en robot bör ha. /6/

En del påstår att en robot är en helt och hållet fristående maskin som kan utföra sina uppgifter genom att enbart få några enstaka instruktioner av människan. Dessa är överens om att en robot kan beskrivas som en intelligent agent som är kapabel att utföra uppgifter på egen hand. Dessutom anser de att en robot bör vara mobil och ha egen strömförsörjning. /6/

Andra anser att en robot inte alls behöver vara en fristående maskin eller autonom, dvs. intelligent. De menar att en maskin även kan räknas som robot så länge den utför ett arbete på ett människolikt sätt. Det är då ingen skillnad om själva styrsystemet eller strömkällan till roboten är beläget på en annan plats än själva roboten. En robot av denna typ kan jämföras med en industrirobot. /6/

Eftersom det finns så många typer av robotar finns det organisationer som har egna standarder på vilka krav en robot bör uppfylla för att vara en viss typ av robot. Nedan listas tre olika organisationers klassificeringar på robotar. (Författarens egen översättning) /9/

JIRA (Japanese Industrial Robot Association) – Japanska Industrirobotföreningen

- Klass 1: Manuellt hanterad enhet: en enhet med hög grad av självständighet, styrd av en operatör.
- Klass 2: Fast sekvensrobot: en enhet som utför sin uppgift upprepade gånger utgående från en förbestämd metod, som är svår att modifiera.
- Klass 3: Variabel sekvensrobot: samma som i klass 2, men lätt att modifiera.
- Klass 4: Uppspelningsrobot: en mänsklig operator utför uppgiften manuellt genom att leda roboten, som spelar in sekvensen för att sedan kunna spelas upp; roboten upprepar rörelsen enligt den inspelade informationen.
- Klass 5: Numeriskt kontrollerad robot: operatören förser roboten med ett rörelseprogram i stället för att manuellt lära den uppgiften.
- Klass 6: Intelligent robot: En robot som förstår sin omgivning och har förmåga att framgångsrikt utföra en uppgift trots förändringar i omgivningen.

RIA (Robotics Institute of America) – Robotinstitutionen i Amerika anser endast klass 3-6 av ovanstående vara robotar. AFR (Association Francaise de Robotique) – Franska robotföreningen har följande klassificeringar:

- Typ A: Manuellt hanterad enhet.
- Typ B: Automatiskt hanterad enhet med förutbestämda cykler.
- Typ C: Programmerbara, servokontrollerade robotar med kontinuerlig eller punkt-till-punktbanor.
- Typ D: Samma som ovan (C), men med förmågan att samla information från dess omgivning.

2.1 Stationära och mobila robotar

En del robotar är inte gjorda för att köra runt, utan de kan ha uppgifter som inte kräver att de kan förflytta sig. Man kan därför indela robotar som mobila eller stationära. De flesta robotar som används inom industrin är stationära robotar och används t.ex. för att svetsa eller flytta objekt från ett ställe till ett annat. De stationära robotarna som används inom industrin består ofta av en arm som antingen är tre- eller fleraxlad och utrustade med ett lämpligt verktyg i ändan av armen för att kunna utföra sin uppgift. /6/

De mobila robotarna är gjorda för att kunna förflytta sig. Detta sker vanligen med hjälp av hjul, larvband eller ben. Dessa robotar kan även de vara utrustade med en rörlig arm för att hantera objekt. De mobila robotarna används till allt från att dammsuga till att utföra militära uppdrag. /6/

2.2 Autonoma och fjärrstyrda robotar

Den klassiska uppfattningen av en robot är att den är fullt autonom. Autonom betyder att en robot har förmågan att kunna sköta sig själv och endast genom ett par enstaka kommandon från människan kunna genomföra sina uppgifter. Dessa robotar har egenskaperna att kunna samla information om sin omgivning, fatta beslut och agera därefter. De är bra på att ta sig fram i ostrukturerade terrängar och de kan även undvika situationer som är farliga för dem själva eller för en människa. Autonoma robotar kan även ha egenskaperna att lära sig nya saker. Detta kan vara t.ex. att välja en ny rutt som går snabbare än den tidigare eller känna igen föremål. En fullt autonom robot har egenskaperna att:

- Samla information om dess omgivning.
- Arbeta helt - eller en längre tid utan mänsklig assistans.
- Förflytta sig genom okänd terräng utan mänsklig assistans.
- Undvika situationer som är skadliga för människor, egendom eller sig själv.

Robotar behöver inte alltid vara fullt autonoma. Det kan ibland vara endast en del av roboten som är autonom medan andra delar av roboten styrs av människan. Ett exempel på detta är marsbilarna som används av NASA. Där sköts lågnivåfunktionerna av mikroprocessorer ombord på bilen, medan information vart bilen ska köra kommer från människan. Dessa robotar är fjärrstyrda och tekniken kallas för telemanövrering, eller *eng. teleoperation*. Telemanövrerade robotar används t.ex. för patrullering och övervakning eller för militära uppdrag. I figur 1 ses en bild på en telemanövrerad robot. /6/ /35/



Figur 1. Telemanövrerad robot för militärt bruk. /41/

2.3 Telepresence robotar

Telepresence är en teknik som ger oss möjlighet att delta i möten utan att vara fysiskt närvarande. Tekniken ger användaren känslan av att vara närvarande fast han kan befinna sig på en helt annan plats. Skillnaden på vanlig videokonferens och telepresence teknik är att användaren får en känsla av att befinna sig på den plats där mötet äger rum samt att kunna påverka något på platsen, som t.ex. styra en kamera. /42/

Innan man hade tillgång till denna teknik var man tvungen att resa mycket för att ha möten. Orsakerna till att tekniken används friskt i dag är att det reducerar resor, vilket betyder att man sparar både tid, pengar och miljö. I figur 2 ses en bild på en telepresence konversation. /42/



Figur 2. Telepresence konversation. /18/

Telepresence robotar är en typ av robotar som använder en kombination av både telepresence och teleoperation teknikerna. De ger användaren möjlighet att kunna utföra arbete på en avlägsen plats och även kommunicera med människor som befinner sig där. /1/

Dessa robotar är utrustade med kamera, högtalare, mikrofon och i många fall även en monitor. Telepresence robotar är även utrustade med t.ex. hjul för att kunna ta sig fram, vilket ger användaren möjlighet att förflytta sig på fjärrplatsen. /1/

Denna typ av robot är fortfarande inte så långt utvecklad och det finns få telepresence robotar på marknaden. En telepresence robot som kan nämnas är Anybots QB, som kan ses i figur 3. Roboten är självbalanserande och tar sig fram med hjälp av hjul. Uppe på roboten sitter en liten bildskärm där man kan se operatören. Nedanför bildskärmen sitter kameran med vars hjälp operatören kan följa med omgivningen. /11/



Figur 3. En telepresence robot från Anybots. /11/

3 Tillvägagångssätt

Projektet att utveckla en telepresence robot omfattar många olika delmoment. Eftersom dessa robotar är relativt nya finns det lite information om hur motsvarande projekt har lösts. Därför var det nödvändigt att göra en del undersökningar för att komma fram till de bästa funktionssätten och val av komponenter. Delmomenten behandlas var för sig och i de kommande rubrikerna kan man se hur projektet genomfördes.

3.1 Plattformsval

Det finns en rad olika alternativ som man kan välja mellan vad beträffar valet av plattform vid tillverkning av en telepresence robot. En av de första sakerna som gjordes var att undersöka vilka alternativ som skulle lämpa sig bäst för just detta projekt, för att utgående från det kunna gå vidare och välja övriga komponenter.

De krav som plattformen bör uppfylla är att den ska ha bra manövrerbarhet, dvs. den ska vara lätt att köra och ha möjlighet att manövreras med god precision. Målet är även att roboten ska kunna köras fritt inomhus och detta förutsätter att den ska kunna ta sig över trösklar. Därför bör man i valet av plattform ta i beaktande plattformens framkomlighet.

Det är tänkt att robotens kamera ska placeras högt uppe för att ge en bra översikt när man kör. Den höga tyngdpunkten kommer att inverka på stabiliteten och därför är det viktigt att man tar även detta i beaktande och ser till att få en stabil plattform. Plattformen bör även vara tillräckligt stor för att ha rum med alla komponenter.

Till roboten är det även tänkt att ha en basstation dit roboten körs när den ska laddas. Detta behövs eftersom det är önskvärt att man utan en mänsklig assistans ska kunna använda roboten.

Flera olika plattformsalternativ undersöktes noggrant för att finna det bästa alternativet till projektet. Här togs i beaktande allt från drifttid till manövrerbarhet och stabilitet. Det som sågs som största problem var att hitta en tillräckligt stor plattform som skulle ha rum med alla komponenter samt tåla den höga vikten. Nedan presenteras de olika alternativen.

3.1.1 Robotdammsugare

Först ansågs en robotdammsugare vara det bästa alternativet, dels för att de har färdiga basstationer som roboten automatiskt kan docka till och dels för att robotdammsugare har god manövrerbarhet. Många andra liknande projekt har gjorts med dessa robotar. Robotdammsugare har även en del sensorer som kan tänkas vara användbara i detta projekt, som t.ex. ”väggberöringssensor”.

Det visade sig snabbt att robotdammsugare hade svagheter som gjorde dem oanvändbara. De hade bl.a. för liten markfrigång, vilket skulle resultera i att roboten skulle ha svårt att ta sig över trösklar. De var även för små, vilket innebär att det skulle vara svårt att få rum med alla komponenter.

3.1.2 Radiostyrbil

Ett annat alternativ som undersöktes var radiostyrda bilar. Det sågs till början ut som ett vettigt alternativ eftersom det finns många radiostyrda bilar ute på marknaden som är relativt billiga och har god framkomlighet. Det finns även tillräckligt stora radiostyrda bilar som skulle ha rum med alla komponenter.

Nackdelen med dessa är att de saknar basstation och har överlag mindre batterikapacitet än robotdammsugare. Den största nackdelen med detta alternativ är ändå att de går alldeles för fort för att kunna manövreras.

3.1.3 Färdiga robotplattformar

Det finns många färdiga robotplattformar som man kan utgå ifrån och som man kan bygga vidare på enligt egna önskemål. Dessa plattformar består i många fall inte av något annat än chassi och motorer, och därför skulle de passa bra till detta projekt.

Nackdelen med detta alternativ är att de flesta färdiga plattformar på marknaden är dyra i jämförelse vad man får. Dessutom var alla färdiga robotplattformar som undersökte alldeles för små för att kunna användas och även om man hittade en tillräckligt stor skulle den vara alltför dyr. Dessutom saknar de flesta även basstation.

3.1.4 Egen plattform

Det sista alternativet som undersöktes var en egen konstruerad plattform. Det finns många fördelar med detta val. Här har man möjlighet att själv bestämma storlek på plattformen, framdrivningssätt, val av motorer, etc. Man har även möjlighet att bestämma robotens hastighet eftersom man väljer motorer och hjul själv. En annan fördel med detta val är att eftersom man själv får välja storleken på plattformen, så har man även möjlighet att använda ett större batteri, vilket skulle vara begränsat i annat fall. Nackdelarna med en egen plattform är att den tar tid att bygga och att en basstation måste tillverkas.

3.1.5 Slutsats

Man kom slutligen fram till att det bästa alternativet var en egen konstruktion. Fördelen med detta är att man fritt får bestämma hur man vill att den ska se ut samt att man slipper problem med svåråtkomliga signaler som kan uppstå om man använder sig av t.ex. en robotdammsugare. Nackdelarna med en egen konstruktion är som tidigare nämnts att man är tvungen att lägga ner mera tid på tillverkningen samt att en basstation måste tillverkas.

3.2 Framdrivningssätt

För att få en robot att kunna röra sig krävs det någon typ av drivlina. Det finns olika framdrivningssätt att välja mellan när man konstruerar en robot, som ben, larvband eller hjul. Hjuldrivna robotar är de mest populära och även de enklaste att konstruera. Hjulantalet kan variera och styrsättet kan ske på olika sätt. /7/

En minoritet av robotar är konstruerade med ben. Detta bland annat för man måste övervinna en del svåra utmaningar för att få dem att fungera bra. Det är svårt att få en robot med ben att hålla balansen och det finns svårigheter i hur man får en sådan robot att svänga. Även om robotar med ben är svåra att konstruera har de fördelar. De är bra på att ta sig fram i terränger där robotar med hjul kan ha det svårt, som över diken och andra ojämnheter. /7/

Robotar med larvband har en ganska enkel konstruktion och de har fördelarna att ha bra manövrerbarhet och framkomlighet. Denna konstruktion består av två larvband, ett på var sida, som drivs av varsin motor. Larvbanden ger roboten möjlighet att ta sig fram i ojämn terräng och på hala underlag. /7/

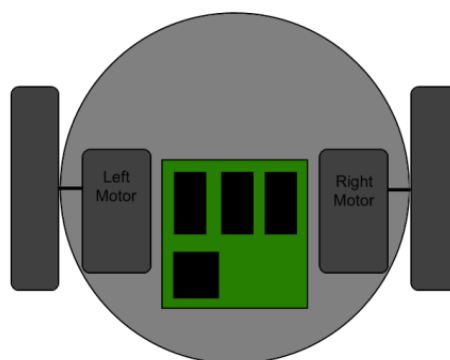
Det som ända från början sågs som en självklarhet var att konstruera en robot med hjul eftersom en sådan robot är lättast att bygga. Dessutom är kravet på att framkomligheten inte så stor eftersom roboten kommer att användas inomhus.

Det finns dock många olika alternativ att välja på när man använder hjuldrift. Alla alternativ har för- och nackdelar och nedan kommer en del olika alternativ att presenteras.

3.2.1 Differentialstyrning

Differentialstyrning består av två hjul som drivs av varsin motor. För att roboten ska hålla balansen placeras ofta ett tredje frirullande hjul bakom de drivande hjulen. Ibland förekommer även konstruktioner med två frirullande hjul för att öka stabiliteten. /2/ /3/

Differentialstyrning är ett av de vanligaste framdrivningssätten på grund av att det är okomplicerat att konstruera och styra roboten. Detta drivsätt ger en bra manövrerbarhet eftersom roboten kan svänga runt på stället om man kör hjulen i motsatt riktning. En nackdel med differential styrning är att det är svårt att få roboten att köra rakt framåt. Detta eftersom motorerna alltid går lite olika trögt, vilket resulterar i att ena motorn alltid kommer att ha ett lägre varvtal än den andra, vilket medför att roboten kommer att svänga lite åt det ena hållet. För att lösa detta problem är man tvungen att använda en återkoppling för att kunna reglera motorernas varvtal. I figur 4 kan ses principen för differentialstyrning. /2/ /3/ /14/



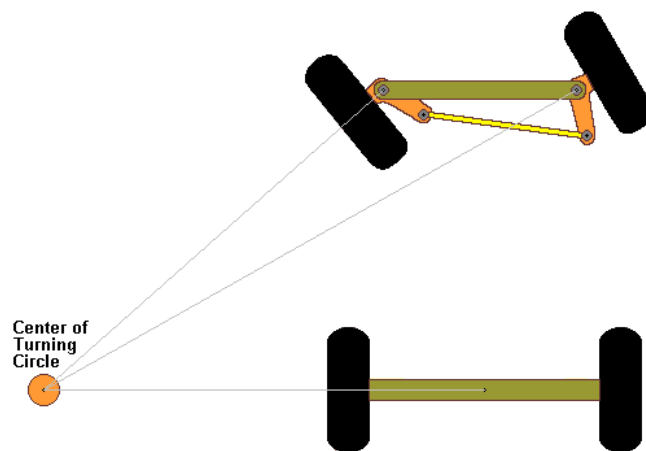
Figur 4. Differential styrning. /14/

3.2.2 Ackerman-styrning

Ackerman-styrning är den metod som används bland annat i bilar. Konstruktionen är utformad så att styrningens sker på det ena hjulparet och drivningen på det andra. Även drivning och styrningen på samma hjulpar förekommer. /2/ /3/

Konstruktionen ger en god stabilitet, men en något sämre manövrerbarhet eftersom man inte kan svänga runt på stället. Med denna konstruktion har man inga problem med att roboten inte skulle gå rakt. /2/ /3/

Konstruktionen blir i många fall även dyr eftersom det förutsätter en mer komplicerad konstruktion av styranordningen samt att en differentialaxel måste användas på drivhjulena för att få styrningen att fungera optimalt. I figur 5 ses funktionen hos ackerman-styrning. /2/ /3/

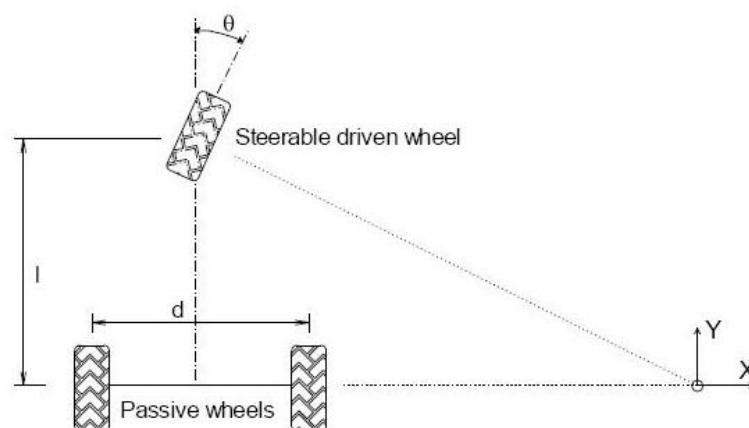


Figur 5. Ackerman-styrning. /12/

3.2.3 Tricycle-styrning

Tricycle-styrning har i princip samma funktion som Ackerman-styrning, förutom att man i stället för fyra hjul har tre. Både drivningen och styrningen sker på det hjul som befinner sig längst bak eller också längst fram, de andra två hjulen är passiva. /2/ /3/

Denna konstruktion ger en bra manövrerbarhet, men en något sämre stabilitet än Ackerman-styrningen. Konstruktionen har dock inte problem att inte gå rakt eftersom drivningen sker enbart på ett hjul som är centrerat. Funktionssättet ses i figur 6. /2/ /3/

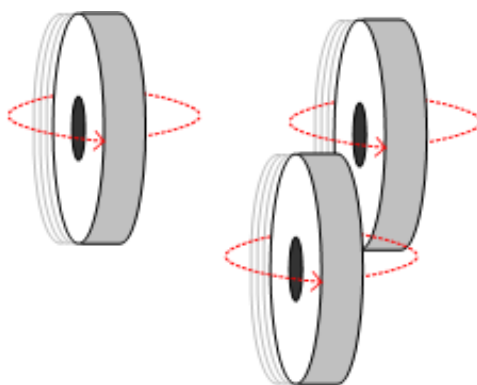


Figur 6. Tricycle-drivning. /16/

3.2.4 Synkrostyrning

I figur 7 illustreras hur synkrostyrning fungerar. Systemet består av tre eller flera hjul som alla både styr och driver. Vanligtvis styrs hjulen av en motor, som med hjälp av en kedja eller rem sammankopplar alla hjulen mekaniskt och styr dem i samma riktning. Likadant är det med drivningen. Alla hjul länkas ihop av en kedja eller rem. Detta gör att man får samma varvtal på alla hjul. Ibland kan konstruktionen även ha två motorer per hjul. En för drivning och en för styrning. I detta fall behövs ingen mekanisk sammankoppling. /2/ /3/

Fördelarna med denna drivning är att man uppnår en bra manövrerbarhet och god precision. Nackdelen är att det krävs lite mera av den mekaniska biten för att få det att fungera, vilket ofta leder till högre kostnader. /2/ /3/



Figur 7. Synkro drivning. /44/

3.2.5 Mecanum-hjuldrivning

Mecanum-hjuldrivning går ut på att man använder ett så kallat mecanum-hjul, som är en variant av hjul som består av flera rörliga rullor, vilket gör att man kan uppnå mycket bra manövrerbarhet eftersom man kan köra dessa hjul även i sidled. /2/

Konstruktionen består av tre eller flera hjul som kan drivas i båda riktningarna för att köra framåt, bakåt, i sidled eller snurra runt på stället. I figur 8 och 9 finns bilder på mecanum-hjulet. /2/ /37/

Fördelen med denna konstruktion är den goda manövrerbarheten. Nackdelarna är ökat slitage på hjulen, pga. att hjulen slirar vid körning och att det ibland uppstår problem med överstyrning vid svängar. /2/ /37/



Figur 8: Mecanum-hjuldrivning. /37/



Figur 9. Mecanum-hjuldrivning. /19/

3.2.6 Slutsats

Framdrivningsättet som valdes var differential styrning, alltså en variant av två motorer som driver varsitt hjul samt ett passivt hjul där bak som hjälper till att hålla balansen. Denna variant valdes eftersom det konstaterades att det var den lättaste och billigaste konstruktionen samt att robotens manövrerbarhet skulle bli tillräckligt bra med det drivsättet.

3.3 Design

Designen på roboten spelar en avgörande roll för framkomligheten. En rund plattform vore det bästa alternativet eftersom det då är en mindre risk att roboten fastnar i objekt som finns i dess omgivning /3/. Nackdelen är då att man i detta fall skulle vara tvungen att göra roboten högre för att få plats med alla komponenter och därmed skulle den kunna bli instabil. Det beslöts att göra plattformen mera kantig, med fördelarna att få en stabilare och rymligare plattform.

Roboten får inte vara konstruerad så att den kan skada inredning eller människor. Därför är det viktigt att runda av alla vassa kanter och helst även förse roboten med någon typ av gummilist för att undvika skador.

Till en början gjordes en prototyp av plattformen för att kunna provköra bl.a. motorerna och dess drivkretsar. Efter en del testande kunde man göra ritningar på den slutgiltiga plattformen. Där beaktades allt som tidigare kommit fram samt komponenternas storlek, vikt och placering. Under projektets gång byggdes plattformen på efterhand för att till slut bli en färdig robot. Bilder på plattformen kan ses i bilaga 8.

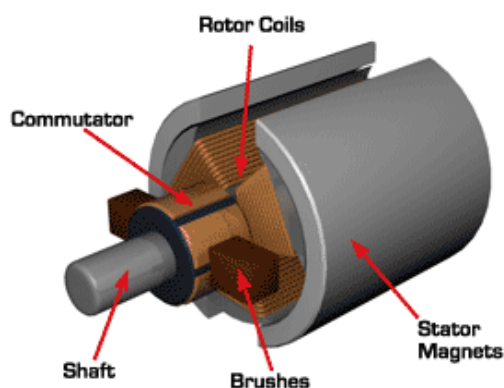
Eftersom motorernas axlar inte fick belastas hur mycket som helst valde man att placera batteriet längs bak eftersom detta var den komponent som vägde mest. På det viset kom dess tyngd främst på det passiva hjulet som tål mera vikt. Datorn placerades så att den skulle vara lättåtkomlig eftersom det är den komponent som man oftast måste ha tillgång till.

3.4 Motorer

En väsentlig del under projektet var att välja lämpliga motorer till roboten. Eftersom roboten skall få sin energi från ett batteri så är det naturligt att välja ett par DC-motorer. Det finns dock många olika typer av DC-motorer så därför undersöktes ett antal olika varianter för att kunna jämföra deras fördelar, nackdelar och styrmöjligheter.

3.4.1 PMDC- motorer (med borstar)

En DC-motor består av en roterande del som kallas rotor, en statisk del som kallas stator, en kommutator med borstar som används för att ändra strömmens riktning och slutligen en axel. Dessa delars placering kan ses i figur 10.



Figur 10. Permanent magnet DC-motor. /43/

Ett permanent magnetfält skapas i dess stator med permanenta magneter eller m.h.a. elektromagnetiska lindningar. I dagsläget och speciellt i mindre motorer (under ca 3hp) så är DC-motorer med permanent magnet betydligt vanligare pga. deras låga kostnad och pålitlighet. I större motorer så är lindningar vanligare eftersom det magnetfält som magneterna skapar inte är tillräckligt starkt /13/.

I motsatt riktning från statorfältet så finns armaturfältet. Det skapas genom ett varierande magnetfält som kommer från de lindningar som finns på rotorn. De magnetiska polerna som finns hos armatur fältet kommer att försöka ställa sig vid dess motsatta pol som genereras av statorfältet. Med andra ord så kommer motorn att rotera fram tills att de motsatta polerna står emot varandra /13/

Den del av rotorn där elektricitet går in i rotor lindningar kallas för kommutator. Strömmen överförs mellan rotorn och statorn genom ledande grafit - koppar borstar som finns monterade på rotorn. Motorn roterar mot dess motsatta pol och strax innan motorn kommer till den punkten så hoppar borstarna över ett gap i statorringarna. Momentet gör att motorn kör över detta gap och när borstarna kommer till andra sidan av gapet så har de kontakt med stator ringarna igen och polariteten på spänningen svängs om för ringuppsättningen. Motorn börjar accelerera igen och försöker att ta sig till dess motsatta pol /13/.

Så länge som rotorn roterar så kommer strömmen i lindningarna att kommuteras och ett kontinuerligt vridmoment kommer att skapas /26/.

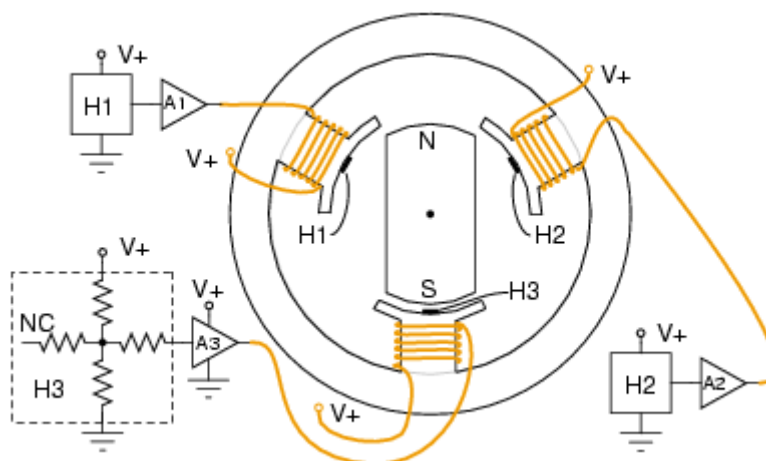
Det finns många fördelar med en permanent magnet DC-motor. De är oftast väldigt små och lätta i förhållande till den effekt de ger ut. Eftersom statorns magnetfält skapas m.h.a. permanenta magneter så blir effektförlusterna mindre än om man skapar magnetfältet med lindningar. En PMDC-motor kan också ge ut ett relativt högt vridmoment vid låga hastigheter.

Det finns också en del nackdelar, bl.a. är den svår att styra hastigheten vid låga varvtal och så är deras livslängd relativt kort pga. att borstarna slits relativt snabbt /26/.

För att ändra rotationsriktning i en normal DC-motor så svänger man på polariteten. Detta görs genom att antingen svänga polerna hos spänningskällan eller byta strömmens riktning i motorn. Byte av rotationsriktning görs vanligtvis med en s.k. ”H-brygga”(se kap. 4.1.1) /26/.

3.4.2 Borstlösa DC-motorer

En borstlös DC-motor fungerar ungefär på samma sätt som en vanlig DC-motor men som namnet antyder så finns det inga borstar. Den mekaniska kopplingen som vanligtvis genomförs av en borste ersätts med elektronisk växling i en borstlös motor. I en typisk borstlös motor så skapas det roterande magnetiska fältet m.h.a. permanenta magneter hos dess rotor. Statorns magnetfält genereras med ett antal elektromagneter som styrs m.h.a. elektroniska brytare (t.ex. transistorer eller FET:ar) i rätt ordning och hastighet. Det varierande statorfältet gör att motorn roterar. Problemet är att veta när den elektriska energin i lindningarna skall ändras för att skapa den roterande rörelsen. Detta löser man oftast genom att använda sig av någon slags återkoppling som ger information om magnetpolernas position i förhållande till lindningarna /26/. Detta kan man se i figur 11 där H1, H2, H3 är halleffektgivare som anger rotorns position.



Figur 11. 3-fas borstlös motor. /32/

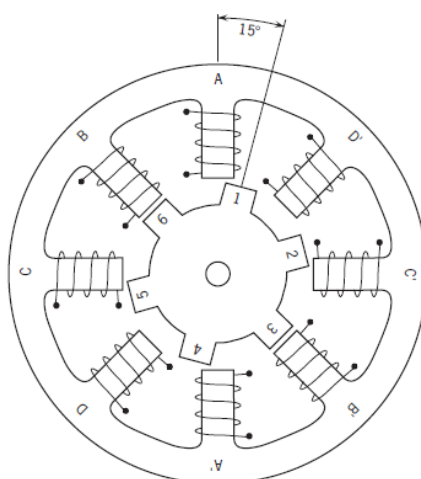
En borstlös motor brukar användas när hög pålitlighet, lång livslängd och hög hastighet är viktigt. Eftersom det ofta är borstarna som nöts ut i en vanlig DC-motor så är det bättre att använda en borstlös motor ifall man vill ha en motor med ett högt varvtal (ju högre varvtal desto mer slits borstarna) /26/.

Ett borstlöst motorpaket som innehåller motor och kontroller kostar betydligt mer än en vanlig DC-motor, men kostnaden jämnar vanligtvis ut sig i längden pga. dess långa livslängd/26/.

3.4.3 Stegmotorer

En stegmotor är en motor som omvandlar elektriska pulser till mekanisk rörelse. En stegmotor styrs m.h.a en PWM-signal där den vid varje puls roterar ett steg. Stegmotorns beteende är direkt beroende av PWM-signalens utseende och frekvens. Utseendet bestämmer rotationsriktning, medan frekvensen bestämmer rotationshastigheten /17/.

Det finns tre olika varianter av stegmotorer, varierbar reluktans (VR), permanent magnet (PM) och hybrid (HB) stegmotorer. VR-stegmotorn är äldst av dem, den består av en mjuk flertandad järnrotor och en shuntlindad stator som kan ses i figur 12. När statorlindningarna leder ström från en DC-spänning så blir polerna magnetiserade. Rotation uppstår när rotorns tänder dras till den strömförande statorns poler.



Figur 12. Tvärsnitt av en VR-stegmotor. /17/

PM-stegmotorn har en permanent magnet med nord- och sydpoler i stället för en tandad järnkärna som rotor. Den magnetiserade rotorn ger ett ökat magnetiskt flöde, vilket leder till ett bättre vridmoment. Däremot så är motorns noggrannhet lite sämre eftersom stegavstånden blir lite större /17/.

Hybridstegmotorn är dyrare än PM-stegmotorn men ger bättre vridmoment, hastighet och noggrannhet. Dess noggrannhet varierar mellan 100 till 400 steg per varv vilket är betydligt bättre än PM motorns 24 – 48 steg per varv. Hybridstegmotorn kombinerar PM- och VR-stegmotorernas bästa egenskaper. Rotorn är flertandad som hos VR-motorn men

också magnetiserad. Av dessa tre så PM och hybridvarianterna vanligast där PM-motorn oftast är flera gånger billigare än hybridmotorn men ger inte lika bra noggrannhet och vridmoment /17/.

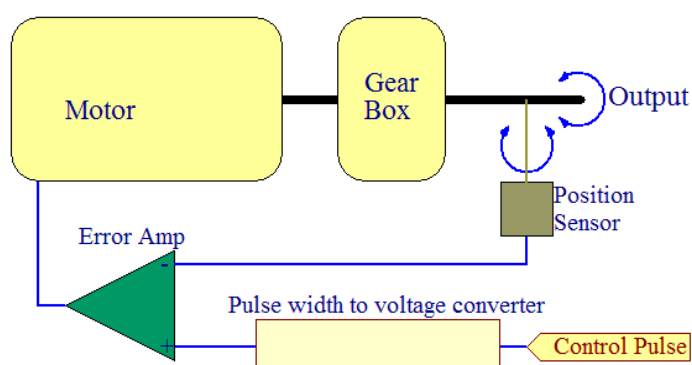
En av de största fördelarna med en stegmotor är att de kan styras genom ”open loop”, vilket innebär att ingen återkoppling från givare krävs för att få information om dess position och hastighet. Positionen och hastigheten fås genom att hålla reda på motorns ingångspulser /17/.

3.4.4 Servomotor

En servo är en liten enhet som har en utgående axel. Denna axel kan placeras på särskilda vinkellägen genom att skicka en pulssignal till servon. Så länge som den pulserande signalen finns så kommer servons vinkel att hållas konstant. Om längden på pulserna ändras så kommer också servons vinkel att ändras /29/.

Servon är mycket användbara inom robotik. Motorerna är små, har inbyggda kontrollkretsar och är kraftfulla i förhållande till deras storlek. Effektförbrukning är också direkt proportionell med den mekaniska belastningen, så en lätt belastad servo förbrukar väldigt lite energi /29/.

En servomotor består av flera delar, en motor med växellåda, en positionsgivare som är ansluten till den utgående axeln, en felförstärkare och en pulsbredd till spänningsomvandlare /15/. Ett block diagram av dessa delar kan ses i figur 13.



Figur 13. Blockdiagram över en servos uppbyggnad. /15/

Styrpulsens matas in till en pulsbredd till spänningsomvandlare. Denna krets laddar en kondensator så länge som pulsen är hög. När pulsen går låg så går kondensatorns laddning till en utgång genom en lämplig förstärkare. Detta leder till att en spänning produceras som är beroende av styrpulsens längd /15/. Servon förväntar sig vanligtvis en puls med 20 ms mellanrum, men eftersom utgångsspänningen lagras så påverkar inte små variationer i tiden mellan pulserna märkbart /29/ /15/. Det är längden på den pulsen som bestämmer hur och vart servomotorn ska rotera /29/. Pulsernas längd brukar vanligtvis ligga mellan 1 ms och 2 ms där 1,5 ms är servomotorns neutrala läge (90 grader). Om pulsen är kortare än 1,5 ms så kommer den att rotera mot 0 grader, medan en längre puls kommer att rotera den mot 180 grader /15/.

Servomotorns position avläses m.h.a en givare, vanligtvis ett varierbart motstånd som producerar en spänning beroende på axelns vinkel. Positionsgivaren sänder spänningen till en operationsförstärkare som jämför den spänningen med den spänning från pulsbredd till spänningsomvandlaren. Operationsförstärkaren har en negativ tillbakakoppling och kommer därför att skicka ut skillnaden mellan dessa två spänningar /15/. Ifall axeln är i rätt vinkel så är spänningarna lika stora och motorn stängs av men ifall axeln inte är i rätt vinkel så uppstår en spänningsskillnad /29/. Skillnaden är antingen negativ eller positiv, ifall den är positiv så kommer motorn att rotera åt en riktning och om den är negativ så kommer den att rotera åt den andra riktningen /15/. En servos vinkel kan vanligen variera mellan 0 - 180 grader p.g.a. ett mekaniskt stopp men det finns även servomotorer som kan rotera upp till 360 grader /29/.

Den effekt som motorn behöver är proportionell med det avstånd som motorn måste rotera. Så om axeln måste rotera en längre distans så kommer motorn att rotera med nästan full fart, men om den endast behöver rotera en kort sträcka så kommer den att rotera med en långsammare hastighet. Detta kallas även proportionell reglering /29/.

3.4.5 Motordimensionering

Vid dimensionering av motorerna så användes en kalkylator som fanns tillgänglig på <http://www.robotshop.com>. Med denna kalkylator så kunde man ange hjulens diameter, maximal lutning som roboten skall köra upp för, maximal hastighet och acceleration. Kalkylatorn räknar därefter ut motorns vridmoment och vinkelhastighet.

Eftersom roboten endast bör kunna färdas på ett plant golv så borde lutningen vara noll men eftersom det i skolan fanns väldigt höga dörrtrösklar så antogs lutningen motsvara ca 30 grader med ett par hjul som har diametern 20 cm. Robotens maximala hastighet bör motsvara en persons medelgångshastighet (ca 5 km/h). Accelerationen var svårare att uppskatta men ifall den uppnådde sin maximala hastighet på 2 s så borde det gott och väl räcka till. Robotens totala massa uppskattades ligga på ca 15 kg där batteriet är den del som kommer att väga mest, men även plattformen och datorn kommer att väga en del.

Resultatet från kalkylatorn var att motorn borde ha ett vridmoment på 8,4 Nm och en vinkelhastighet på 62 r/min.

3.4.6 De valda motorerna

Vid valet av motorer så användes information om hur de olika motorerna fungerade och hur de styrdes. Denna information togs i beaktande när motorernas kostnader undersöktes. Stegmotorn var den motor som troligtvis hade varit lättast att använda sig av. Med den så skulle det räcka att man skapade en varierbar PWM-signal som man sedan förstärkte innan den matades till motorn. Motorns hastighet skulle också vara enkel att styra genom att endast variera på PWM-signalens frekvens. Nackdelen med stegmotorn var dess höga kostnad, den kostar flera gånger mer än både en vanlig DC-motor och en borstlös DC-motor.

Den borstlösa motorn är lite dyrare än en vanlig DC-motor men kostnaden skulle möjligtvis jämnas ut sig i längden eftersom den borstlösa DC-motorns har en mycket längre livslängd. Problemet med den borstlösa motorn är att den är väldigt svår att styra, eftersom man måste avläsa rotns position och därefter sända en signal till rätt ingång under ett visst tidsintervall. För att kunna göra det så behövs ytterligare en styrkrets eftersom den

dator och det I/O-kort som användes troligtvis inte skulle vara tillräcklig snabb för att läsa in, processera och sända iväg en signal till motorn. Ett paket som innehåller de borstlösa DC-motorerna med styrkrets skulle ha en total kostnad som visserligen ligger under priset för ett par stegmotorer, men skillnaden skulle inte vara så stor att det skulle var värt priset.

Servomotorn är väldigt användbar inom robotiken p.g.a. dess pris, noggrannhet och så är den lätt att styra. Den är däremot inte lämplig att driva hjul eftersom den endast kan rotera 180 grader (vissa kan även rotera upp till 360 grader). Den passar därför bättre ifall man vill ha den att styra saker som en robots armar eller ben. Två servomotorer användes för att kunna styra en webbkamera horisontellt och vertikalt.

De motorer som valdes för att driva hjulen var två vanliga 12 V DC-motorer med permanenta magneter. Anledningen till att de valdes var främst p.g.a. deras låga kostnad men de antogs också inte behöva vara lika noggranna som t.ex. stegmotorer ifall man körde dem tillräckligt långsamt. För då borde det ändå gå att kontrollera och styra roboten genom att i styrprogrammet ställa in dem så att de roterade med ungefär samma hastighet. Ifall det ändå skulle uppstå problem att styra motorerna så kunde man sätta in ett par varvtalsgivare som reglerade hastigheten.

Efter en tidskrävande sökning så hittades ett par DC-motorer från www.robotshop.com som uppfyllde de krav som fåtts fram från motordimensioneringen till ett relativt lågt pris. De var av märket Banebots RS 540 och hade en växellåda med en utväxling på 64:1 (se figur 14). En fördel med dessa är att man kan montera fast hjulen direkt på drivaxeln.



Figur 14. Banebots RS 540. /28/

Med växellådan fastmonterad så fås följande data:

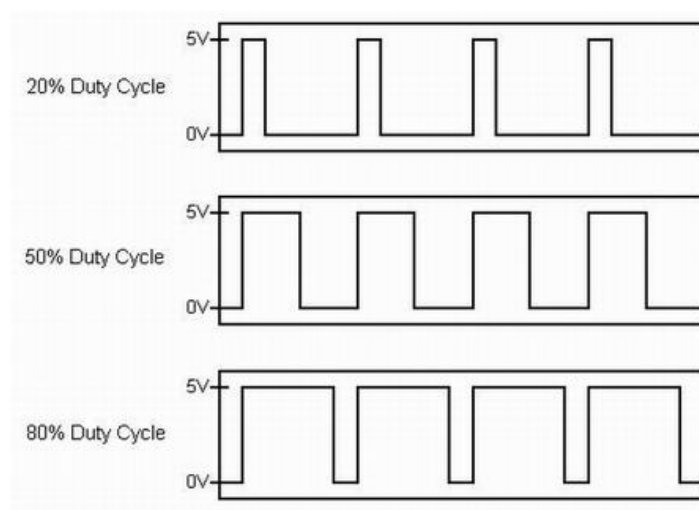
Operating Voltage	4,5 V – 12 V
Nominal Voltage	12 V
No Load RPM	263 r/min
No Load Current	1 A
Stall Current	42 A
Stall Torque	17,8 Nm
Efficiency	71 %

Figur 15. Banebots RS 540 märkdata med en 64:1 växellåda. /28/

3.4.7 PWM-styrning

För att styra en DC-motors hastighet så är det lättast att styra dess drivspänning. Ju högre spänningen över motorn är desto högre hastighet försöker motorn uppnå. Eftersom en vanlig spänningsregulator orsakar stora effektförluster så brukar man i stället använda sig av s.k. pulsbreddsmodulering (PWM) /26/.

Pulsbreddsmodulering är den vanligaste metoden att styra medelspänningen över en DC motorn. En PWM-signal är en signal som antingen är helt på eller helt av. Förhållandet mellan den tid som spänningen är på och den tid som spänningen är av kallas ”duty cycle”(se figur 16) där den vid 100 % är alltid på och vid 0 % är alltid av. /10/



Figur 16. PWM-signal. /24/

Det är ”duty cyclen” som avgör hur stor spänningen över motorn är, ifall man använder sig av en 12 V drivspänning och en ”duty cycle” på 50 % så kommer spänningen över motorn att vara 6 V. Anledningen till det är att motorn i princip fungerar som en stor spole där motorns induktans kommer att integrera eller jämna ut PWM-spänningen så den hålls jämn över motorn. /10/

När man väljer en PWM-frekvens så är det viktigt att tänka på att en effektförlust sker vid varje övergång. En 10 kHz PWM frekvens har tio gånger fler övergångar än en 1 kHz PWM-frekvens. Detta betyder att motorns växlings effektförlust blir tio gånger större vid en 10 kHz frekvens jämfört med en 1 kHz frekvens. Därför borde det vara bra att välja en så låg frekvens som möjligt. Problemet är att vid en för låg frekvens så hinner motorns magnetfält urladdas och spänningen över motorn sjunka så pass mycket att den börjar gå ojämnt. /10/

Efter att ha testat motorerna med ett antal olika PWM-signaler med varierande frekvens så konstaterades att en PWM-frekvens på 100 Hz är tillräcklig hög för att motorn skall hålla en jämn hastighet och en ”duty cycle” som ligger mellan 10 % – 20 % gav en lämplig hastighet.

3.5 Strömkälla

En mobil robot kräver en mobil energikälla som kan lagra energi. Energekällan måste kunna lagra tillräckligt med energi för att roboten skall klara av att utföra sina uppgifter. Den vanligaste energikällan för en mobil robot är ett batteri. Ett batteri omvandlar kemisk energi till elektrisk energi på begäran. Hur mycket energi som behövdes var på förhand svårt att förutspå, men målsättningen var att roboten bör kunna användas minst 2 h, vilket motsvarar en lektion. En grov uppskattning gjordes därför till att ett batteri med kapaciteten 10 Ah borde räcka. /4/

Det finns dock flera olika typer av batterier som har olika egenskaper, därför undersöktes olika alternativ för att veta vilket som var lämpligast.

3.5.1 Zink

Det finns två olika varianter av zinkbatterier, kol-zinkbatterier och zink-kloridbatterier. Kol-zinkbatterierna har sämst kapacitet av alla batterityper, vilket gör dem så gott som oanvändbara inom robotiken. Zink-kloridbatterierna är lite bättre de räcker 25 % till 50 % längre, men de är ändå inte tillräckligt kraftiga för att användas. /5/

3.5.2 Alkaliska

Alkaliska celler kostar ungefär dubbelt mer än zinkbatterier men räcker upp till 800 % längre. Däremot så är de svåra att ladda eftersom när alkaliska batterier laddas så genererar de en massa värme som i sin tur kan leda till att de exploderar eller börjar brinna. Därför slängs vanligen alkaliska batterier när de är slut. Även om det finns alkaliska batterier som går att ladda så kostar de mycket mer än vanliga alkaliska batterier. De kräver också en specielladdare som laddar batterierna väldigt långsamt med en låg ström för att inte skada batterierna. Så ifall man vill ha laddningsbara batterier så finns det bättre alternativ. /5/

3.5.3 Nickel-kadmium

Av laddningsbara batterier så är nickel-kadmiumbatterier en av de som är billigast och lättast att komma åt. De finns att få i de flesta storlekar men deras kapacitet är sämre än för både alkaliska batterier och zinkbatterier. Även om de går att ladda (ca 500 gånger) så lämpar de sig bäst i små och medelstora robotar /5/. Ni-kad batterier kan också råka ut för en s.k. ”minneseffekt”. Detta innebär att den mängd energi de kan lagra minskas efter varje laddning. För att undvika det så bör de endast laddas när de är helt urladdade. /31/

3.5.4 Nickel-metallhydrid

Nickel-metallhydridbatterier är väldigt bra billiga laddningsbara batterier. De kan laddas över 400 gånger och har en låg inre resistans, vilket innebär att de kan ge ut en väldigt hög ström. NIMH-batterier väger ungefär lika mycket som Ni-kadbatterier men de kan leverera upp till 50 procent mer ström. De kan heller inte drabbas av en minneseffekt så man behöver inte vänta tills de är helt urladdade innan man laddar dem /5/. Det kan dock ta väldigt lång tid att ladda dem, upp till 10 h /31/. När de används så avger de en ganska stor värme så man bör undvika att använda dem nära kretsar som är känsliga för värme. En nackdel med dem jämfört med andra laddningsbara batterier är att de inte kan hålla sin laddning väldigt länge. De kan ibland urladdas helt inom några veckor (ibland dagar) även om de inte används. Därför är det oftast bra att ladda dem ofta med jämna intervaller. /5/

3.5.5 Litium och litium-jon

Litium och laddningsbara litium-jonbatterier är populära i bärbara datorer och används bäst vid en jämn urladdning /5/. De har samma kapacitet som NIMH och samma ut effekt som Ni-kad men väger 20 % – 35 % mindre. De har heller inget minneseffektsproblem så de kan laddas när som helst /31/. Litiumbatterier har bäst energidensitet (mängd energi/voly) jämfört med övriga batterier och kan hållas sin laddning i flera månader/5/. Litiumbatterier är de mest avancerade inom bärbar energi, vilket också gör dem till de dyraste. Priset på dem sjunker dock hela tiden och de kommer troligtvis inom en snar framtid konkurrera ut både NIMH och Ni-kadbatterierna. De är också gjord av icke-giftigt material, men däremot så används de mycket lätt och bildar stora mängder väte ifall de kommer i kontakt med vatten. /31/

3.5.6 Bly batteri

Blybatterier är populära eftersom de är så enkla och billiga att tillverka. Laddningsbara blybatterier är också den mest använda batteritypen idag. De är väldigt kraftfulla och så har de en hög kapacitet. Det bästa med dem är att de är billiga och väldigt lättillgängliga. Nackdelen är att de är väldigt stora, tunga och så måste de alltid hållas laddade. Det finns flera saker som man bör tänka på när man hanterar blybatterier. För det första så skall de alltid hållas laddade. Detta betyder att man skall ladda dem som ofta som möjligt och när de laddas skall de alltid laddas fulla. Ifall de inte används så kommer de att sakta urladdas så därför är det bäst att ladda dem så ofta som möjligt. /31/

3.5.7 Det valda batteriet

Det batteri som valdes var ett 12 V blybatteri med en kapacitet på 12 Ah. Anledningen till att den valdes var främst p.g.a. av priset, den var betydligt billigare än övriga batterier med motsvarande spänning och kapacitet, även om Litium batterierna också var väldigt intressanta. En annan anledning till att blybatteriet valdes var att vikten inte hade så stor betydelse. Eftersom robotens konstruktion består av tre hjul så kan man sätta batteriet över det tredje hjulet (stödhjulet) och på så sätt inte belasta motorernas axlar samt så minskar det risken att roboten välter framåt.

3.6 Centralenhet

För att kunna kommunicera med och styra roboten krävs det någon typ av centralenhet som klarar av att hantera all data. Centralenheten ska klara av att skicka och ta emot bild och ljud samt att skicka och ta emot styrsignaler till och från roboten. Allt detta måste ske i realtid utan någon lång fördröjning.

På grund av dessa krav var det lättaste sättet att lösa detta genom att använda sig av en PC som är uppkopplad till internet via trådlöst nätverk. Valet av PC blev en mini-laptop, eftersom de är enkla att använda och man har möjlighet att använda sig av Windows och kan på så vis använda sig av mjukvaror som är anpassade till denna typ av uppgifter. En mini-laptop har även god batterikapacitet och är relativt små och lätta. Datorn som valdes var av märket ASUS, typbeteckning 1001-PX.

3.7 I/O- kort

En enhet som kan läsa signaler och generera styrsignaler användes i projektet. Exempel på dessa signaler är PWM-signaler till motorerna, digitala signaler för strömbrytaren samt analoga signaler för varvtalsmätning av hjulen. Eftersom man använder sig av en vanlig laptop i projektet var det lättaste sättet att åstadkomma detta att använda sig av ett USB-baserat I/O-kort.

Från början valdes att använda ett I/O-kort från National Instruments. Kortet fungerade bra med LabVIEW och det hade alla in- och utgångsportar som behövdes. Efter ett tag blev man dock tvungen att söka efter en annan typ av I/O-kort eftersom stora problem med att få ut en stabil PWM-signal uppstod. Detta var en orsak av att det saknades inbyggda räknare.

Det nya kortet som valdes var ett kort från LabJack, typbeteckning U3-HV. Kortet var lite svårare att få att fungera ihop med LabVIEW än vad National Instruments I/O-kort var, men det hade fördelen att ha inbyggda räknare. Detta gjorde att PWM-signalerna kunde hållas stabila, vilket var ett måste för att få motorerna att gå jämt.

3.7.1 LabJack U3-HV

I/O-kortet som användes har 16 stycken så kallade Flexible I/O-anlutningar som antingen kan användas som digitala ingångar, digitala utgångar eller som analoga ingångar. Kortet har även två stycken analoga utgångar som kan ge ut en spänning på 0–5 V. Vidare kan nämnas att I/O-kortet har inbyggda räknare som kan generera PWM-sigaler med en frekvens upp till ca 730 Hz. I figur 16 kan ses en bild på I/O-kortet. /20/



Figur 17. Labjack U3-HV. /21/

3.8 Övrig utrustning

För att få en så bra kommunikation som möjligt finns det några saker som en telepresence robot bör uppfylla. Det första som påverkar funktionaliteten hos en telepresence robot är den visuella biten. För att få en känsla av att vara närvarande krävs det att bilden som man ser är av god kvalitet. Ljudet bör även vara av god kvalitet för att det ska vara möjligt att föra en diskussion. /27/

Dessa krav uppfyllades genom att skaffa externa högtalare som monterades på roboten. För ljudupptagningen användes en mikrofon som var inbyggd i webbkameran. Webbkameran som valdes var av märket Logitech, typbeteckning Webcam Pro 9000. För att ge användaren möjlighet att få en bättre uppsikt över fjärrplatsen så monterades kameran högt uppe på roboten. Därtill monterades kameran på två servomotorer, vilket gav möjlighet att kunna styra kameran både vertikalt och horisontellt.

För att ha möjlighet att ladda datorns batteri under drift och på så vis förlänga drifttiden på roboten utrustades den med en datorladdare. För att öka drifttiden ytterligare finns även möjligheten att bryta strömmen till laddaren när datorn inte är i behov av att laddas.

Eftersom det var önskvärt att kunna använda roboten utan att vara beroende av mänsklig assistans så förutsätter det att man kan ladda roboten automatiskt. Detta var som tidigare nämnt tänkt att genomföras genom att bygga en basstation till roboten, dit roboten kan köras för laddning. Denna funktion hann aldrig påbörjas, vilket innebär att man i stället är tvungen att manuellt koppla in roboten till laddning.

4 Elektronik

Elektroniken utgjorde en stor del av projektet och tog väldigt mycket tid. Redan i ett tidigt skede var man medveten om att någon typ av drivkretsar till motorerna måste utvecklas för att inte vara tvungen att köpa färdiga drivkretsar, vilket skulle betyda en större finansiell satsning på projektet.

Utvecklingen och tillverkningen av drivkretsarna har i princip pågått under hela projektets gång och studeranden som deltagit i kursen Elektronik 2 har även bidragit till utvecklingen av dessa kretsar.

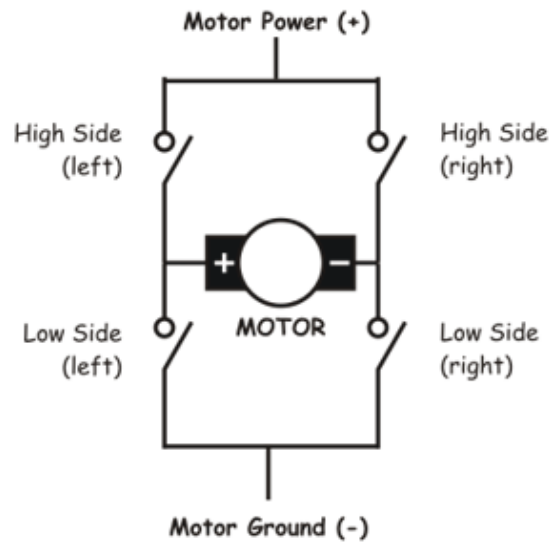
Förutom drivkretsarna tillverkades även en del andra kretsar. I detta kapitel kommer alla elektronikkretsar som finns på roboten att tas upp. Vid utvecklingen och tillverkningen av alla kretskort har det tagits i beaktande allmänna rekommendationer som gäller vid kretskortstillverkning. Man har även försökt dela in kretsarna så att man på ena sidan av kretskorten har placerat styrelektroniken och på den andra sidan drivelektroniken (12 V). Man har också använt optokopplare där det är möjligt för att skydda I/O-kortet.

4.1 Drivkretsar för motorer och logikkrets

I och med att två vanliga DC-motorer valdes så behövdes en krets tillverkas som kan styras med en PWM-signal för att variera hastigheten men man bör också kunna byta motors polaritet för att på så sätt kunna köra den i båda riktningarna. Den krets som oftast används för att åstadkomma detta kallas för en H-brygga.

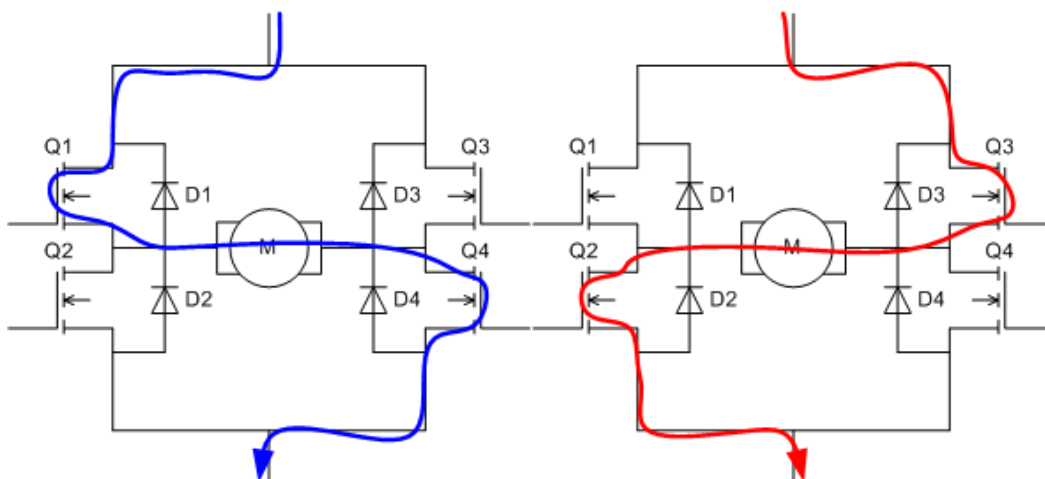
4.1.1 H-brygga

En H-brygga har fått sitt namn från att den består av fyra brytare som tillsammans med en motor brukar ritas så att den formar bokstaven ”H” (se figur 18). /22/ Som brytare används vanligtvis FET-transistorer men vid höga spänningar kan även IGBT förekomma./ 33/



Figur 18. En H-bryggas principskiss. /22/

En H-bryggas funktionsprincip är relativt enkel. Ifall brytaren uppe till vänster och brytaren nere till höger slås på så kommer den högra motorledaren vara ansluten till jord medan den vänstra motorledaren kommer vara ansluten till matningsspänningen. Ström åker igenom motorn och den börjar rotera. Om brytaren uppe till höger och brytaren nere till vänster slås på så kommer det omvända att hända, dvs. den kommer att rotera åt det andra hållet, vilket kan ses i figur 19 och figur 20.



Figur 19. Ström leds medurs. /33/

Figur 20. Ström leds moturs. /33/

Man kan också antingen slå på de två brytarna ovan eller de två brytarna nedan, då kortsluts motorn, vilket gör att motorns generatoreffekt kommer att arbeta emot sig själv. Den roterande motorn genererar en spänning som kommer att tvinga motorn att rotera åt det motsatta hållet och på så sätt bromsa motorn. /22/ Ifall man önskar en annan rotationshastighet än den maximala så åstadkoms det genom att styra brytarna med en PWM-signal. Spänningen över motorn kommer att bestämmas av hur länge PWM-signalen är på i förhållande till hur länge den är av. /33/ Problemet är att ifall båda brytarna på samma sida eller alla fyra brytarna slås på, så leder det till att kretsen kortsluts. /22/

4.1.2 Komponenter

De viktigaste komponenterna i en H-brygga är dess brytare. När man väljer brytare så bör man ta i beaktande saker som driftström, driftspänning och PWM-frekvens. Ett bra val är oftast MOSFET transistorer, de klarar stora strömmar samtidigt som de är väldigt snabba. I en H-brygga så brukar vanligtvis de två MOSFET:ar ovan vara av P-channel typ medan de där nere är av N-channel typ. /33/

När man använder en MOSFET så kan den antingen vara på eller av. När den är på så beter den sig ungefär som ett litet motstånd, så ju större motståndet är desto större kommer effektförlusterna att vara. Även om verkningsgraden sällan är något problem för en H-brygga så kan värme vara det. Därför kan den vara bra att använda sig av någon form av kylning t.ex. kylfläns för MOSFET:arna. /33/

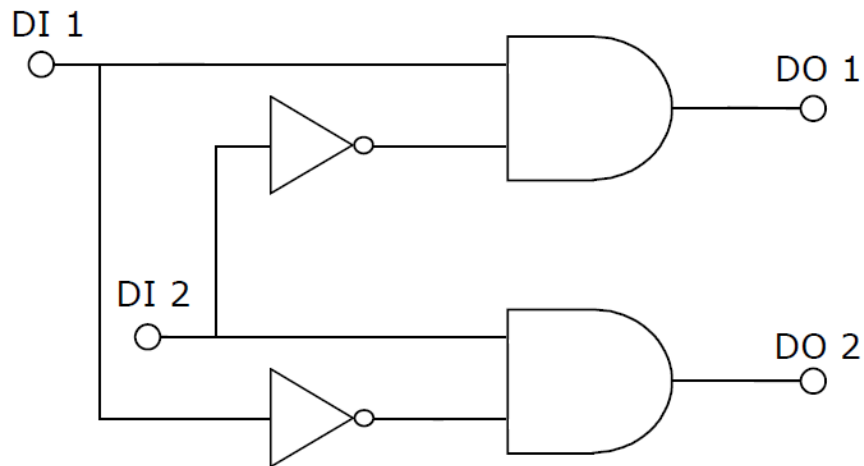
Förutom brytarna så brukar man sätta in en s.k. flyback diod parallellt med varje brytare. Detta främst för att skydda de brytare som används. Eftersom en DC-motor används som är induktiv så kommer den att så länge som den körs skapa ett elektromagnetiskt fält. När brytarna stängs så måste det elektromagnetiska fältet försvinna och för att det skall kunna ske så måste den generera ström. Eftersom brytarna är slutna så leder de inte ström men strömmen måste ändå på något sätt ta sig igenom, vilket kan förstöra brytarna. Flyback dioder används för att strömmen skall kunna ledas bort utan att förstöra brytarna. En bra egenskap hos MOSFET:ar är att de innehåller en inbyggd diod mellan ”drain” och ”source” vilken kan användas som en flyback diod i en H-brygga. /33/

4.1.3 Design av H-brygga

När H-bryggan skulle planeras så fanns det vissa krav och problem som måste tas i beaktande. Den skulle klara av en kontinuerlig ström på ca 5 Ampere och så måste den kunna slå av och på brytaren med minst 1 ms mellan rum dvs. 1 kHz. Från I/O-kortet så fanns det endast två st. PWM-signaler tillgängliga per H-brygga som hade en amplitud på ca 3,3 V, vilket innebar att den ena signalen måste användas för att köra motorn framåt och den andra för att köra motorn bakåt. Det största problemet var dock att designa en H-brygga som inte kunde kortslutas ifall den råkade få en hög signal från båda signalerna samtidigt. Även om man i programvaran kunde göra så att detta inte kunde inträffa så var det något som alltid inträffade när spänning anslöts till I/O-kortet.

För att lösa det problemet så testades och simulerades flera olika kretsar bl.a. med dioder och komparatorer som försökte förhindra att alla MOSFET:ar ledde samtidigt och på så sätt kortslöt kretsen. Även om de fungerade när båda signalerna antingen var höga eller låga så uppstod det alltid en stor strömpik när PWM-pulsen växlade från hög till låg eller vice versa. Anledningen till det är pga. att en MOSFET ursprungligen är uppbyggd av kondensatorer. Detta innebär att när spänningen på gaten överstiger tröskelspänningen så kommer det att ta en liten tid innan kondensatorn laddas upp och transistorn börjar leda. Samma princip gäller när spänningen på gaten går under tröskelspänningen så krävs en viss tid för kondensatorn att urladdas, vilket innebär att den under tiden kommer att fortsätta leda ström. Så under det korta tidsintervallet som den ena signalen går från hög till låg så kommer MOSFET:en att fortsätta leda, vilket kan resultera i att kretsen kortsluts.
/33/

För att få en H-brygga som inte kunde kortslutas så konstaterades det att det var bättre att i stället för att stoppa MOSFET:arna från att inte öppnas samtidigt, så är det lättare att förhindra insignalerna att gå höga samtidigt. Detta gjordes genom att tillverka en separat logikkrets som bestod av två "AND" grindar och två "NOT" grindar. Kretsen är uppbyggd enligt figur 21 och gör att så länge den ena signalen är hög så kommer den att förhindra den andra att gå hög.

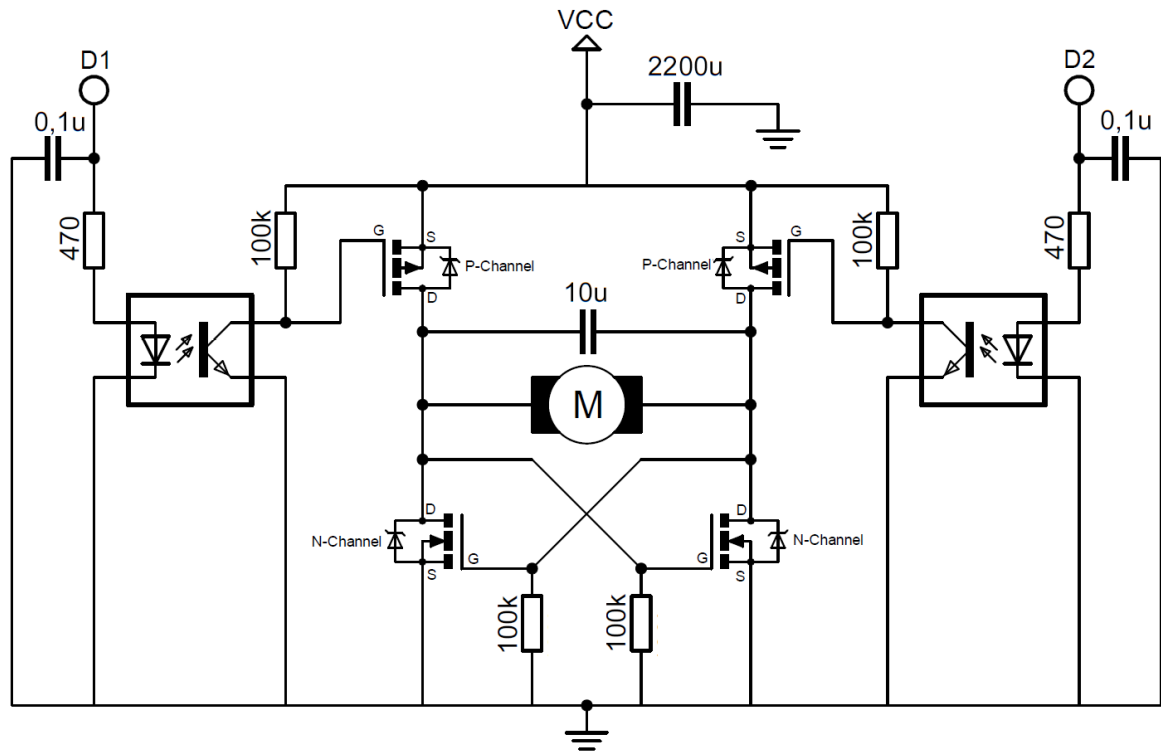


Figur 21. Schema på logikkrets.

Eftersom det endast fanns två PWM-signaler tillgängliga per H-brygga så gjordes det så att man bara kan använda dem till att köra motorn framåt eller bakåt. Motorbromsning genom att kortsluta motorn användes inte, eftersom det inte ansågs behövas i och med att den hastighet som roboten skulle köras i var så långsam.

Därför så användes PWM-signalen för att öppna P-channel MOSFET:en. När den öppnas så uppstår en spänning över motorn. Den spänningen kommer att användas för att styra N-channel MOSFET:en på den motsatta sidan. För att öppna P-channel MOSFET:en så användes en optokopplare för att galvaniskt isolera matningsspänningen från styrspänningarna. När optokopplaren öppnas så börjar den leda ström, vilket innebär att spänningen över det motstånd som finns mellan matningsspänningen och optokopplaren kommer att vara nästan lika stor som matningsspänningen. Detta resulterar i att spänningen mellan P-MOSFET:ens ”gate” och ”source” kommer att bli negativ och öppna den.

För att få bort störningar så sattes även kondensatorer in före matningsspänningen och före styrspänningen samt en parallellt med motorn. Kretsschemat kan ses i figur 22.



Figur 22. Krettschema över H-brygga.

4.2 Batterikapacitetskrets

En viktig funktion i roboten är att kunna se hur mycket energi det finns kvar i batteriet. För att mäta hur mycket energi det finns kvar i batteriet mäter man polspänningen. När batteriet är fulladdat ligger polspänningen på ca 12,7 V och när det är urladdat på ca 11,9 V.

För att kunna mäta denna polspänning kan man använda I/O-kortets analoga ingång genom att koppla batteriets poler till den. Eftersom de kan mäta en spänning på maximalt 10 V, så bör man få ner spänningen, eftersom batteriets maximala spänning är ca 12,7 V. För att åstadkomma detta tillverkade man ett kretskort som bestod av en spänningsdelare som tar ner spänningen till en lämplig storlek.

4.3 Brytarkrets för huvudbrytare

Förutom en mekanisk brytare ville man även ha en elektronisk brytare så att även användaren kan bryta strömmen fast han befinner sig på en annan plats. Denna funktion kunde man lätt lösa med hjälp av ett vanligt relä som man kunde styra med hjälp av I/O-kortet.

Det som visade sig bli ett problem var att i några enstaka fall blev alla utgångar på I/O-kortet höga, vilket ledde till att strömmen skulle bli påslagen okontrollerat, vilket i sin tur kunde leda till att roboten skulle kunna köra av sig själv. Man beslöt då att i stället använda sig av två brytare, en som var normalt öppen, NO, och en som var normalt stängd, NC. Denna konstruktion skulle inte göra det möjligt att strömmen blev påslagen om alla utgångarna skulle bli höga.

Det som då visade sig bli problem var att det inte fanns tillräckligt med analoga utgångar på I/O-kortet. De digitala utgångarna kunde inte användas för att dra ett relä pga. deras låga spänning. Man var då i stället tvungen att använda MOSFET:ar som brytare.

Kretsen består av en P-MOSFET och en N-MOSFET som styrs av I/O-kortets digitala utgångar. I/O-kortet är galvaniskt isolerat med optokopplare för att förhindra skador.

Kretsschema på denna krets kan ses i bilaga 4.

4.4 Brytarkrets för datorladdare

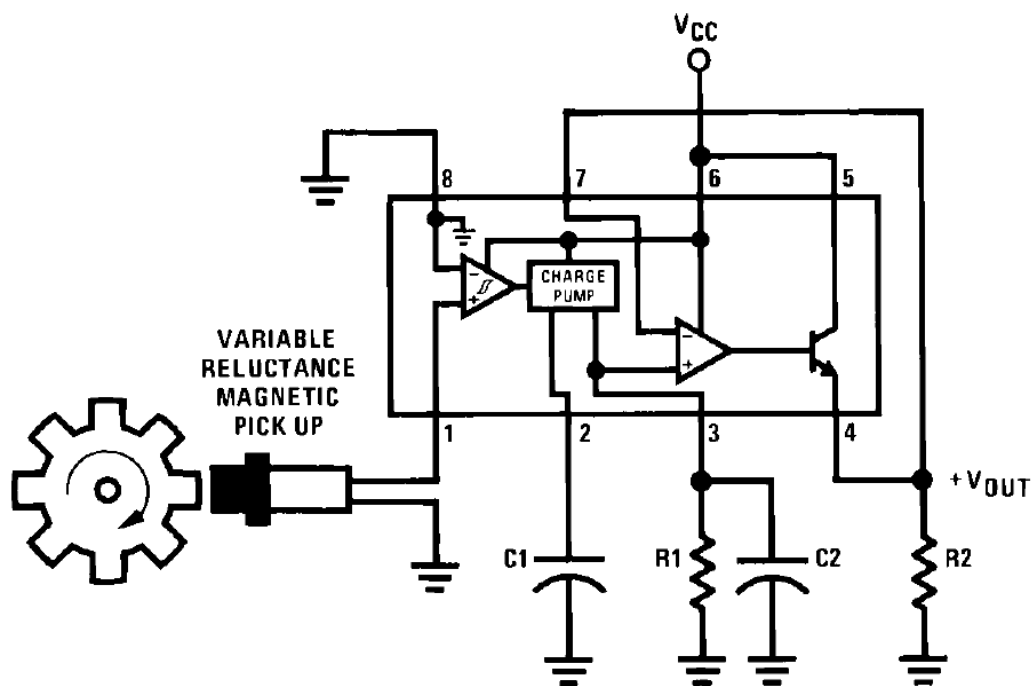
En funktion som man ville ha i roboten var att man skulle kunna ladda den bärbara datorn under drift, eftersom standby-tiden på den reduceras avsevärt pga. alla komponenter som är inkopplade (kamera, högtalare, I/O-kort, monitor, USB-hub). En vanlig bildatorladdare användes till laddningen. Eftersom dessa även tar en lite ström fast datorns batteri är fulladdat ville man ha möjlighet att kunna slå av den. Detta löste man med ett vanligt relä som man styr med hjälp av en analog utgång på I/O-kortets. Orsaken att man använde en analog utgång i stället för en digital var att de digitala utgångarna har en spänning på 3,3 V, vilket är för lite för att orka dra reläet. Den analoga utgången har en maxspänning på 5,2 V, vilket räckte till.

4.5 f/V-krets

När roboten testkördes så konstaterades det att den ena motorns tröghet var mycket större än den andra. Att det skulle vara en viss skillnad på motorerna var väntat och försöktes först lösas genom att ge den ena motorn en PWM-signal som hade en större ”duty cycle” än den andra. På så sätt så skulle man kunna räkna ut förhållandet mellan hur stor ”duty cycle” den ena motorn behövde i förhållande till den andra. Det visades inte fungera eftersom skillnaden var så mycket större under starten än den var när den kommit upp i önskat varvtal. Det resulterade i att den började dra åt ett håll så mycket att den helt släppte på det ena hjulet. Därför konstaterades det att den behöver någon form av regulator som styr hur snabbt hjulen snurrar.

För att kunna använda sig av en regulator så krävs det först att man mäter hur snabbt hjulen snurrar. Det gjordes genom att sätta fast skruvar med jämna mellanrum på hjulen som sedan med en induktiv givare avläser hur snabbt hjulen roterar. Först så testades ifall den bärbara datorn kunde avläsa tiden mellan varje puls som givarna ger ut. Det visades inte fungera eftersom den ibland inte hann med utan hoppade över vissa pulser, vilket i sin tur innebar att den ibland kommer att tro att hjulen roterar långsammare än vad de gör och regulatorn kommer därför att öka på hastigheten. För att lösa det problemet så valdes det att i stället tillverka 2 st. frekvenser till spänningsomvandlare (en för varje hjul). Eftersom en spänning motsvarar en viss frekvens så kan datorn läsa av den spänningen när den hinner och reglerar rotationshastigheten under det tidsintervallet.

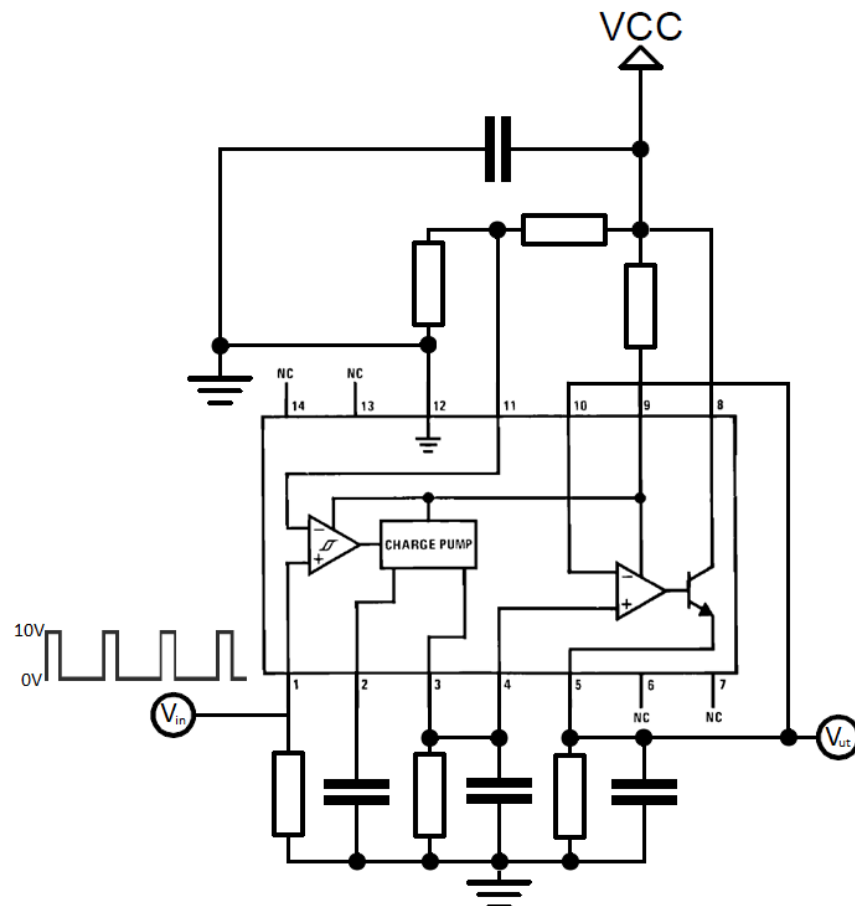
Det finns färdiga IC-kretsar som används för att göra f/V-omvandlare eller liknade kretsar. Den första IC-krets som användes var LM2907N-8 från National Instrument, det fanns också färdiga instruktioner hur man skulle göra för att använda den som frekvens till spänningsomvandlare, vilket kan ses i figur 23.



Figur 23. Grundläggande f/V -konverterare för LM2907N-8. /25/

Efter att ha undersökt den IC-kretsen så kan man se att signalen från givaren kommer att åka in till en komparator vars tröskelspänning är kopplad till jord. Detta innebär att denna krets inte kommer att fungera eftersom signalen från givaren aldrig kommer att gå under 0 V. Därför sattes en MOSFET in mellan ingången till komparatorm och jord som stängdes och öppnades med signalen från givaren. Det fungerade, för då kommer spänningen under en kort tid gå lite under 0 V, vilket gör att komparatorm kommer att variera utspänningen. Denna lösning fungerade men var inte optimal eftersom komparatorms tröskelspänning låg på 0 V och den spänning som styr komparatorm låg väldigt nära 0 V. Det innebar att den var väldigt känslig mot små störningar. Därför användes istället en annan IC-krets.

LM2907N från National Instrument användes därför i stället som IC-krets eftersom i den så behövs inte tröskelspänningen jordas, utan man kan välja den spänningen man vill genom t.ex. en spänningsdelare. Den kunde därför kopplas enligt figur 24.



Figur 24. f/V -konverterare för LM2907N.

Problemet med både Lm 2907N-8 och lm 2907N var att dimensionera de kondensatorer och motstånd som på figur 24 finns under IC-kretsen. Beroende på hur de dimensioneras så kan man bestämma hur stor ut spänningen tillåts vara men också hur snabbt den skall reagera och hur stort ”ripple” tillåts vara. Ifall man vill ha ett litet ripple, dvs. en så jämn signal som möjligt, så kommer det innebära att den reagerar väldigt långsamt på förändringar i frekvensen. Ifall man vill att den skall reagera snabbare på frekvensförändringar så innebär det att signalen blir ojämnare, dvs. ett större ripple. Därför fick man testa sig fram tills man fick en krets som ansågs vara tillräckligt snabb och tillräckligt noggrann. Dessutom så sattes en kondensator parallellt med ut signalen för att ytterligare minska ripplet och för att få en mjukare signal. Slutligen så mättes förhållandet mellan frekvens och spänning för att se om förhållandet var tillräckligt linjärt.

Efter att kondensatorerna och motstånden hade valts så upptäcktes ytterligare ett problem när kretskorten kopplades in. Om motorn gick men hjulen stod stilla så kunde f/V-korten ändå ibland ge ut en spänning. Anledningen till detta var att när motorerna körs med en PWM-signal så kommer spänningen över spänningskällan att variera, t.ex. för ett 12 V batteri kan spänningen variera mellan exempelvis 12,5V och 11,5 V. Eftersom f/V korten tar sin spänning från samma spänningskälla så innebär det att utspänningen från den komparator som finns inne i IC-kretsen kommer att variera ifall hjulet befinner sig i ett sådant läge att givaren går hög. För att lösa problemet så sattes en stor kondensator (10 000 μ F) mellan f/V-kortens matningsspänning och jord. Det ledde till att spänningsförändringarna jämnades ut så pass mycket att de inte påverkade kretsen.

5 Programvara för styrning

För att kunna styra roboten så skapades ett styrprogram i Labview 8.6 som kunde kommunicera med I/O-kortet och på så sätt styra roboten. Det fanns även diskussioner om att göra programmet i Matlab eller att skriva det i C-kod, men valet föll på Labview eftersom det är lätt att skapa användargränssnittet samtidigt som man gör programmet. Dessutom så var det ett program som för oss båda var obekant och vi ville lära oss.

5.1 LabVIEW

LabVIEW(Laboratory Virtual Instrumentation Engineering Workbench) är ett program av National Instruments som oftast används vid datainsamling, instrumentstyrning och i industriell automation. Det använder sig av ett visuellt programmeringsspråk som kallas för "G", vilket oftast används vid automatisering av processer och processmätningar. /36/

Programmeringsspråket "G" är ett dataflödesprogrammeringsspråk. Programmets exekvering bestäms av ett grafiskt blockdiagram som programmeraren kan koppla olika funktioner till genom att dra trådar från ett block till ett annat. Blocken utför en funktion medan trådarna förflyttar variabler från block till block. Eftersom man ofta använder sig av flera olika variabler och funktioner, är "G" kapabel av parallell exekvering och använder sig automatisk av inbyggda "multi-processing" och "multi-threading" maskinvara. /36/ /23/

LabVIEW binder ihop skapandet av användargränssnittet (Front Panelen) med resten av programmeringen. Ett program eller en subrutin i LabVIEW kallas för "virtual instruments"(VIs). Varje VI består av tre komponenter: ett block diagram, en frontpanel och en anslutningspanel. Anslutningspanelen används för att kunna ansluta ett VI till ett annat. Så att ifall man gjort en VI subrutin så kan man importera det in i huvudprogrammet och ansluta det m.h.a. trådar. Frontpanelen fungerar som användargränssnitt och kan användas för att mata in värden, avläsa värden, utföra olika styråtgärder osv. Ifall man gör en subrutin så kommer frontpanelen att användas för att definiera de ingångar och utgångar som motsvarar vilken på anslutningspanelen. /36/

Fördelar med LabVIEW är främst dess användargränssnitt som går snabbt och enkelt att skapa. Det innehåller väldigt många färdiga funktioner som finns bra beskrivet. Det är därför relativt lätt för en oerfaren programmerare att tillverka enklare program. Programmet är dock väldigt stort och för att förstå sig på hur man använder LabVIEW optimalt krävs en stor erfarenhet och kunskap inom området. Det kan också vara svårt att få strukturerat blockdiagrammet på ett snyggt sätt så att det blir lättläst.

5.2 Programmet

För att få programmet mer lättläsligt så delades det upp i flera subrutiner som sedan importerades in i huvudprogrammet. Under kommande del så kommer de olika subrutinerna och huvudprogrammets uppbyggnad att beskrivas för att ge en bättre inblick i hur det är uppbyggt. Detta för underlätta ifall någon i framtiden vill vidareutveckla det.

5.2.1 Subrutiner

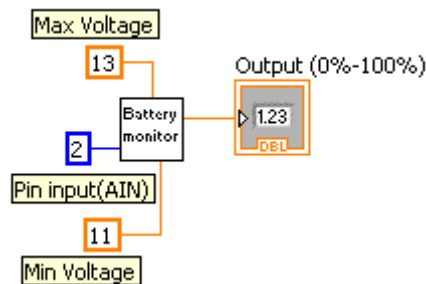
Eftersom ett I/O-kort från LabJack användes så krävdes det lite mer arbete att få det att fungera i LabVIEW än vad det skulle ha varit ifall man använde ett som var tillverkat av National Instruments. Ifall man använder sig av ett kort från NI så finns det en färdigt inbyggd funktion (DAQ Assistant) i LabVIEW, där man endast behöver ange vilken port på I/O-kortet som skall användas och vilken funktion den skall ha. Den funktionen fungerar inte ifall man använder sig av ett kort gjort av en annan tillverkare. Det kräver i stället att man skapar en VI som initialiserar I/O-kortet och använder drivrutiner från operativsystemet. På LabJacks hemsida så fanns det färdiga exempel program på LabJacks som man kunde använda och modifiera lite för att skapa subrutiner för I/O-kortet. De subrutiner som gjordes för I/O-kortet var:

- Analog input (läser in en analog spänning på vald port).
- Analog output (ger ut en analog spänning på vald port, 0 V – 5 V).
- Digital output (ger ut en digital spänning på vald port, 3,3 V).
- Enable 2 PWM (startar två PWM-signaler på valda portar med de valda frekvenserna och ”duty cyclerna”).

- Enable 1 PWM (startar en PWM-signal på den valda porten med den valda frekvensen och ”duty cyclen”).
- Update 2 PWM (uppdaterar de två PWM signalernas ”duty cycle”).
- Update 1 PWM (uppdaterar PWM signalens ”duty cycle”).
- Disable PWM (stoppas PWM signalen/signalerna).

Dessa subrutiner kunde därefter importeras in i huvudprogrammet eller in i andra subrutiner och man behövde bara ange önskad port, spänning, frekvens och ”duty cycle”.

Eftersom det var svårt att få en snygg och strukturerat program så gjordes en del mindre subrutiner för att minska på blocken och trådarna. Det gjordes bl.a. ett program som användes för att avläsa givarna och ett som räknade om spänningen över batteriet till procent som man kan importera in i huvudprogrammet.



Figur 25. Subrutinen Battery monitor insatt i huvudprogrammet.

5.2.2 Tangentbordsstyrning

För att styra roboten så kan man på frontpanelen skapa tryckknappar som man kan klicka på med musen. Det går lätt att åstadkomma i LabVIEW, men för att göra det smidigare att styra roboten så skulle det vara bra ifall det också fanns möjligheten att styra roboten med piltangenterna på tangentbordet.

Att använda tangentbordet för att styra saker kan göras på tre olika sätt i LabVIEW. Det första och lättaste sättet är att via frontpanelen tilldela en tangent från tangentbordet till en knapp på frontpanelen. Detta fungerar endast ifall man bara vill trycka en gång på knappen och inte ifall man vill hålla in knappen, för då kommer den att slås på och slås av med jämna och snabba mellanrum. Dessutom så är de tangenter från tangentborden som kan tilldelas väldigt begränsade (bl.a. piltangenterna finns inte).

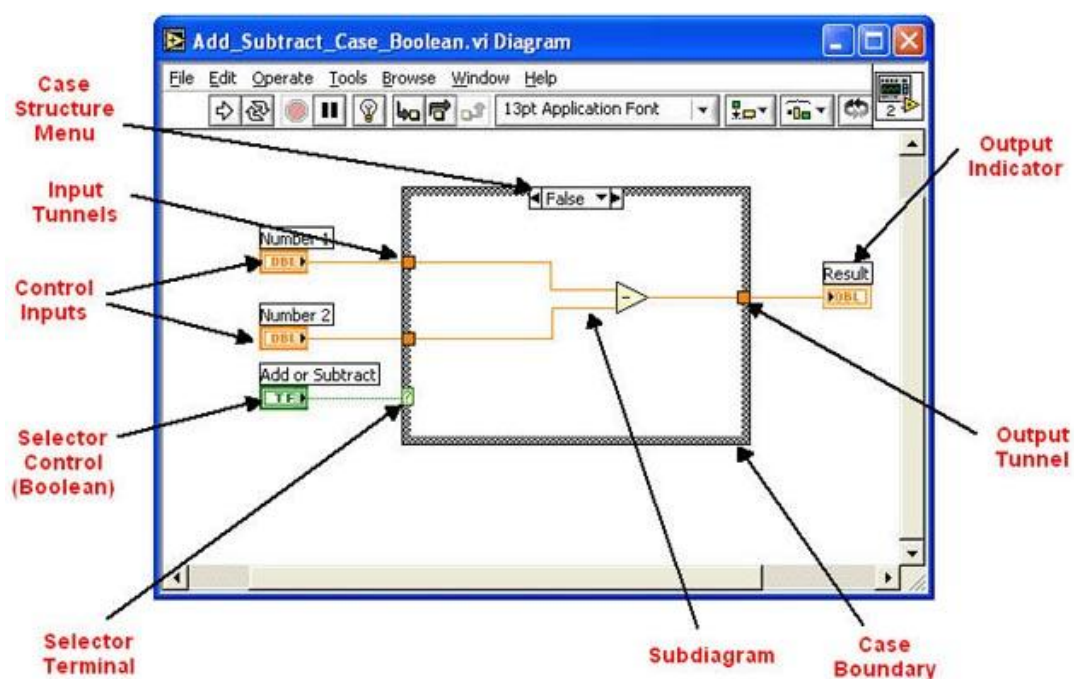
Det andra sättet är att använda sig av s.k. event-strukturer. Med dem så kan man ställa in så att ett "event" inträffar när en tangent på tangentbordet trycks in eller släpps av. Man kan därefter få fram vilken tangent som trycktes in och använda den informationen till att t.ex. starta en PWM-signal. När tangenten släpps av så skulle ett annat event inträffas som skulle kunna nollställa PWM-signalen. Risken med den metoden är att ifall man håller på att köra roboten samtidigt som t.ex. internetanslutningen avbryts så kommer den inte att registrera ifall en tangent släpps av och därför fortsätta köra. Därför torde det vara säkrare att i stället för event använda sig av s.k. "polling", dvs. man loopar hela tiden igenom programmet för att se ifall någon förändring skett.

Det tredje sättet och det sätt som användes var att skapa en subrutin (se bilaga 1) som först initialiserar datorns tangentbord och genom "polling" skaffar fram information om vilken eller vilka tangenter som trycks ned. Den informationen kommer att jämföras med alla de tangenter som används och ge ut en boolsk "1:a" ifall de är lika och en boolsk "0:a" ifall de är olika. Alla de boolska värdena kommer därefter att bilda en boolsk vektor som kommer att omvandlas till numerisk form t.ex. 0000 0100 kommer att omvandlas till talet fyra. Det numeriska värdet kommer därefter att användas för att huvudprogrammet ska veta vilken tangent som tryckts ner. Dessutom så kommer det föregående värdet att sparas via ett "shift register" som också kommer att användas i huvudprogrammet.

Ett problem som uppstod var att när programmet styrdes över Internet så kunde man endast använda sig av funktionstangenterna (t.ex. piltangenterna, shift, ctrl, alt osv.), ASCII-tecknen registrerade den inte. Anledningen till det är lite oklart men kan bero på att om man initialiserar det tangentbord som finns på roboten men använder sig av ett annat så kan den inte registrera vissa tangenter förutom funktionstangenterna. Det var ett problem eftersom man ville kunna styra roboten med piltangenterna och webbkameran med "W", "A", "S", "D". Därför gjordes det så att man kan styra webbkameran med musen istället. Programmet kommer att registrera muspekarens position och jämföra den med den föregående. Ifall de är olika så kommer den att ge ut en "1:a" som går in till den boolska vektorn som i huvudprogrammet kommer att styra webbkameran.

5.2.3 Huvudprogrammet

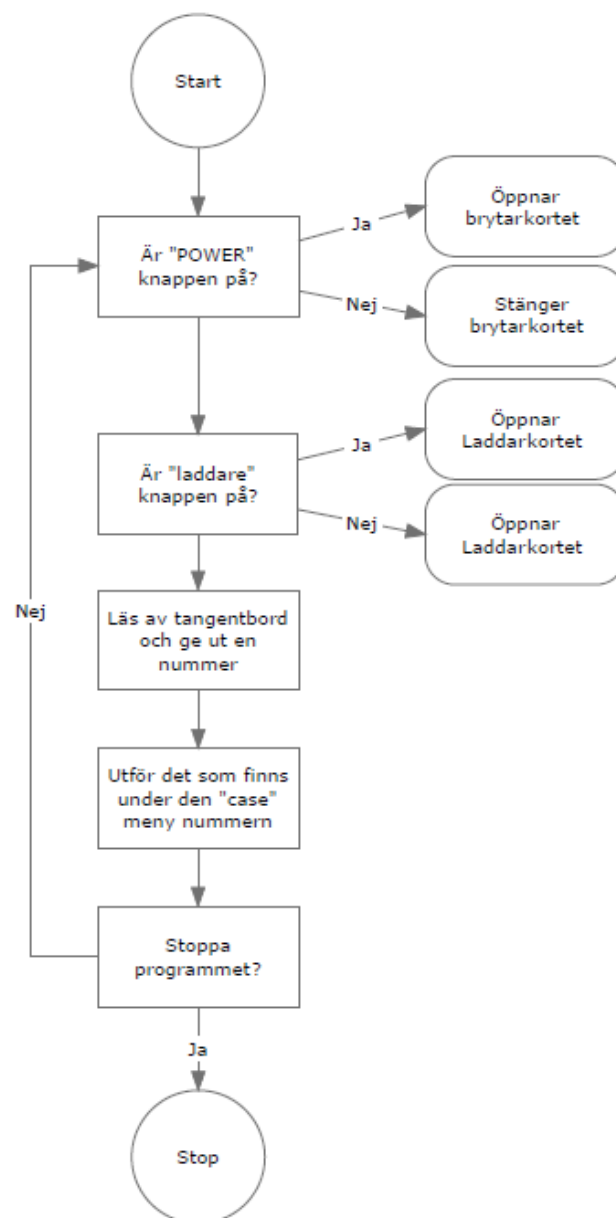
Huvudprogrammet är omringat av en "while loop" som kommer att polla igenom programmet tills man väljer att stanna det. Programmet är i stort sett uppbyggt av s.k. "case strukturer". I figur 26 beskrivs hur en "case struktur" är uppbyggd. Figuren är en boolsk "case struktur" som bara kan vara "true" eller "false" men den kan också var numerisk och då kommer den att gå till den case-strukturmeny som "selector terminalen" anger.



Figur 26. Exempel på en case-struktur. /30/

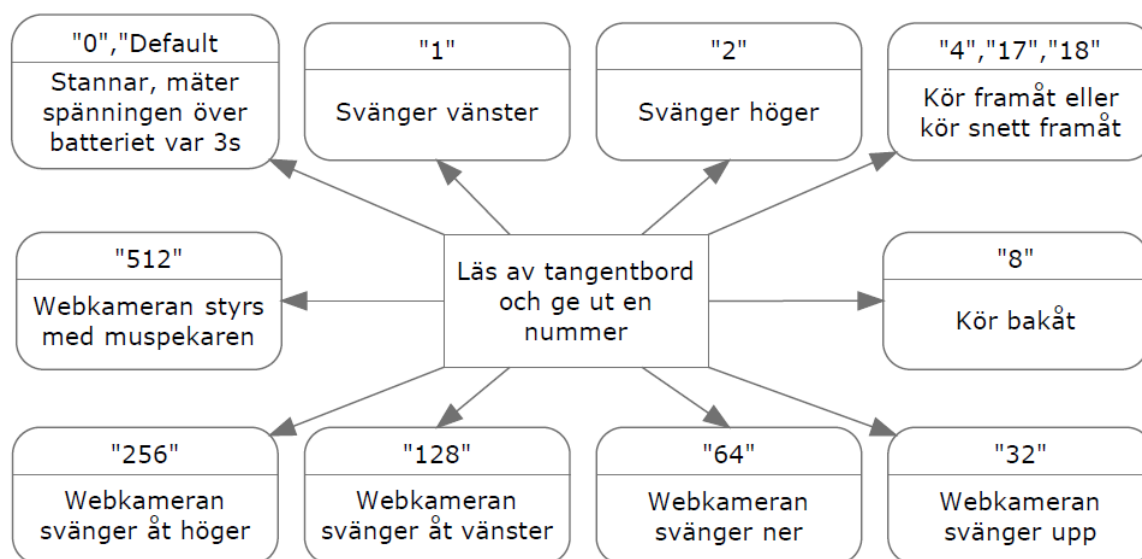
En "case struktur" fungerar på motsvarande sätt som en "switch case sats" i programmeringsspråket C. Det får in ett värde som det jämför med dess angivna värden och i fall det hittar ett värde som är lika så kommer den att hoppa dit och utföra den programkod som finns där /8/. Så ifall man trycker ner en viss tangent eller en knapp på frontpanelen så kommer tangentbordssubrutinen att ge ut ett tal till "case strukturen" och utföra den programkod som finns under det talet.

Efter att subrutinerna hade gjorts så sattes de in i huvudprogrammet och eftersom de är omringade av en "while loop" så kommer programmets uppbyggnad att se ut som i figur 27.



Figur 27. Programmets uppbyggnad.

Programmet kommer att loopa igenom blocken (=program koden) och kontrollera vilka knappar som är intryckta och utföra en åtgärd därefter. Efter att den utfört tangentbordssubrutinen så kommer den att ge ut en siffra. Den siffran kommer att bestämma vilken meny i "case strukturen" den skall hoppa till och utföra den kod som finns där. De olika åtgärderna som den kan utföra och vilken nummer det motsvarar kan ses i figur 28.



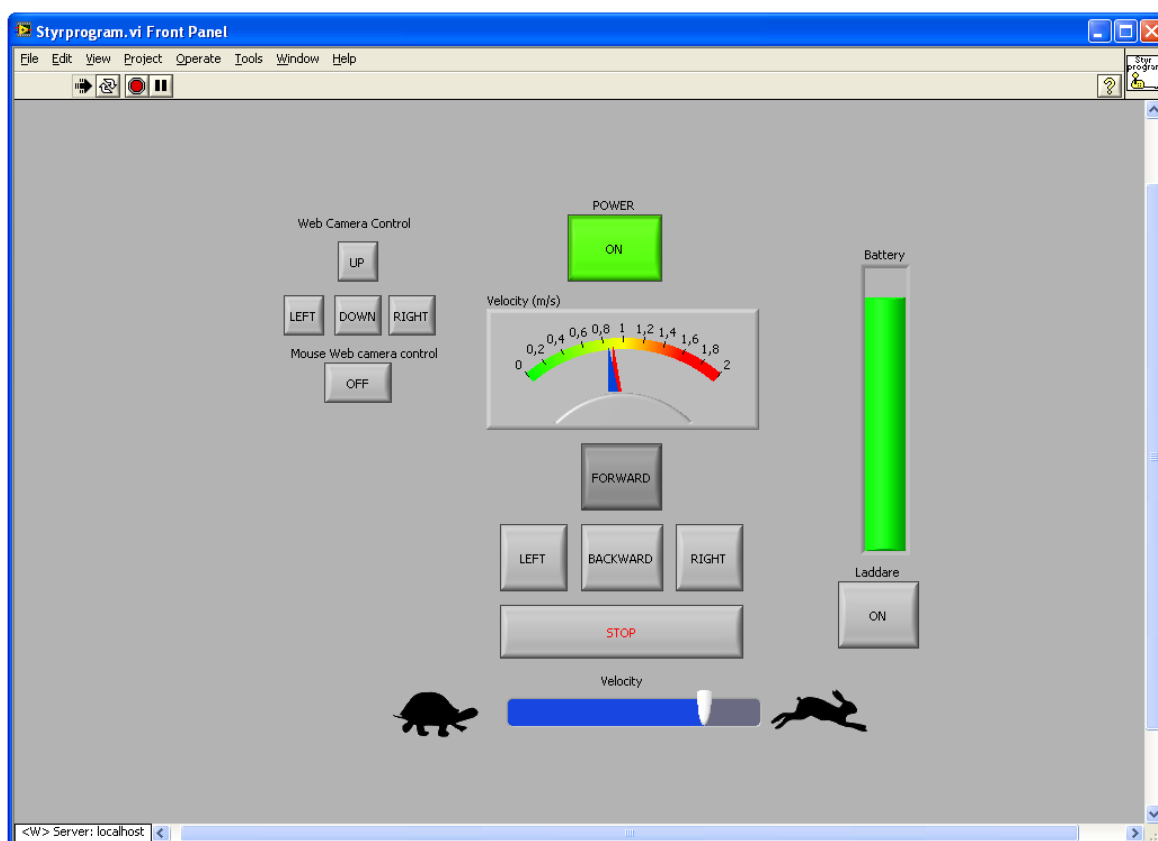
Figur 28. Blockschema över de olika menyerna som finns i "case strukturen".

Ifall något annat värde än de som finns ovan anges så kommer det att resultera att "case strukturen" förflyttar sig till "Default" och utför det som finns under den. I bilaga 2 kan man se hur de olika menyerna såg ut och en liten beskrivning av vad som sker i dem.

"Case strukturen" användes eftersom den passade väldigt bra in i sammanhanget. När en knapp trycks in så utförs en åtgärd. Nackdelen med den är att man endast kan utföra en sak åt gången. Så ifall man vill köra roboten och styra webbkameran samtidigt så går det inte. Detta gjorde inget eftersom det ändå inte skulle varit möjligt p.g.a. att I/O-kortet endast kan ge ut max två PWM-signaler åt gången. Detta innebär att om man kör framåt vilket kräver två PWM-signaler så finns det inte någon mer PWM-signal tillgänglig.

5.2.4 Frontpanelen

Frontpanelen består av ett antal tryckknappar uppe till vänster som styr webbkameran och i mitten som styr roboten. Till höger finns en mätare som visar hur mycket laddning som finns kvar i batteriet. Under det så finns också en knapp som man kan styra reläet till datorladdaren med. Under de knappar som styr roboten finns en slinga som man kan justera för att välja hastighet. Ovanför knapparna så finns en mätare som visar vilken hastighet de två hjulen roterar med. På så sätt kan man följa med hur bra regulatorerna fungerar. Högst upp finns ”POWER”-knappen som öppnar eller stänger brytarkortet och på så sätt styr spänningen till motorerna. En mer detaljerad beskrivning över hur man styr roboten finns i bilaga 3.



Figur 29. Frontpanelen

5.2.5 LabVIEW Web Server

I LabVIEW så finns en inbyggd webbserverfunktion som gör att man kan enkelt göra styrprogrammet till en webbserver. På så sätt så skulle man endast behöva öppna en webbläsare, ange adress och så skulle man kunna styra roboten. För att överföra bild och ljud så skulle Skype användas. Ett test program gjordes därför där man via LabVIEW kan ange ett namn och sedan trycka en knapp. Därefter så kommer LabVIEW att starta upp Skype leta upp det namn man angett och ringa åt den personen. På så sätt så skulle man vi LabVIEW styrprogrammet kunna ringa åt sig själv och få en Skype-videokonversation. Även om webbservern fungerade så gick den inte att använda. För att kunna använda den via skolans WLAN så krävdes det tillgång till routern för att kunna portforwarda en port, dessutom så måste den ställas in så att datorn som webbservern körs på har en fast IP-adress.

6 Programvara för kommunikation

En lämplig programvara måste väljas både för själva kommunikationen mellan datorerna och kommunikationen mellan användarna. Eftersom Labview kommer att tillämpas för styrningen, där man använder sig av ett grafiskt användargränssnitt för att styra roboten, krävs det att man har tillgång till skrivbordet på fjärrdatorn. Detta kan åstadkommas genom att använda ett program som möjliggör fjärrmanövrering av en dator, t.ex. Windows Remote Desktop.

Det testades att använda Windows Remote Desktop, men man kom fram till att det blev lite svårare än tänkt eftersom man var tvungen att göra portforwarding, vilket betyder att man öppnar den port i nätverket som Remote Desktop behöver ha öppen för att fungera.

Ett annat program som testades var TeamViewer som är ett program som är gjort både för fjärrmanövrering av datorer samt video- och ljudkommunikation. Eftersom TeamViewer innehöll alla de funktioner som behövdes till projektet, samt att det inte kräver portforwardning, så beslöts att använda det. Nackdelen med programmet var dock att det kräver ganska stor bandbredd. /38/ /40/

För att kommunicera mellan användarna prövades även Skype som är ett program som möjliggör webkonversationer. Skype är ett långt utvecklat program som fungerar väldigt bra till detta ändamål. Det visade sig att Skype fungerade bättre för ljud- och bildöverföringen än TeamViewers inbyggda ljud- och bildöverföring, vilket resulterade i att Skype användes i detta projekt. Nackdelen med Skype var att det blir ytterligare ett program som måste startas upp, vilket kan göra att starten blir onödigt invecklad på grund av att många program måste startas samt att robotens dator blir mera belastad och därmed slöare med ett program till i användning.

6.1 TeamViewer

TeamViewer är en enkel, snabb och säker applikation för webbkonversationer och fjärrmanövrering av PC. En fördel med TeamViewer som tidigare nämnts är att man kan kommunicera genom brandväggar och proxyn utan några speciella inställningar. Man kan använda Teamviewer till bland annat följande: /34/

- fjärrmanövrera en dator
- dela sitt skrivbord med andra
- filöverföring
- videokonversationer.

Programmet installeras på både den lokala datorn och fjärrdatorn. Sedan tilldelas datorn ett ID som klienten använder när den ansluter till datorn. Inloggningen är lösenordsskyddad för att förhindra obehöriga att logga in.

Vid inloggningen kan man antingen välja på *Remote control*, *File transfer* eller *VPN*-anslutning. Remote control används för att kunna fjärrmanövrera datorn, File transfer används för att kopiera filer mellan datorerna och VPN-anslutningen används för att t.ex. dela en printer och dylikt. Den typ av inloggning som används i detta fall är Remote control, vilket ger en tillgång till fjärrdatorns skrivbord. Väl ansluten till fjärrdatorn har man full kontroll och översikt över skrivbordet. Se bilaga 3 för en mera detaljerad beskrivning av hur man ansluter till roboten.

7 Funktionssätt

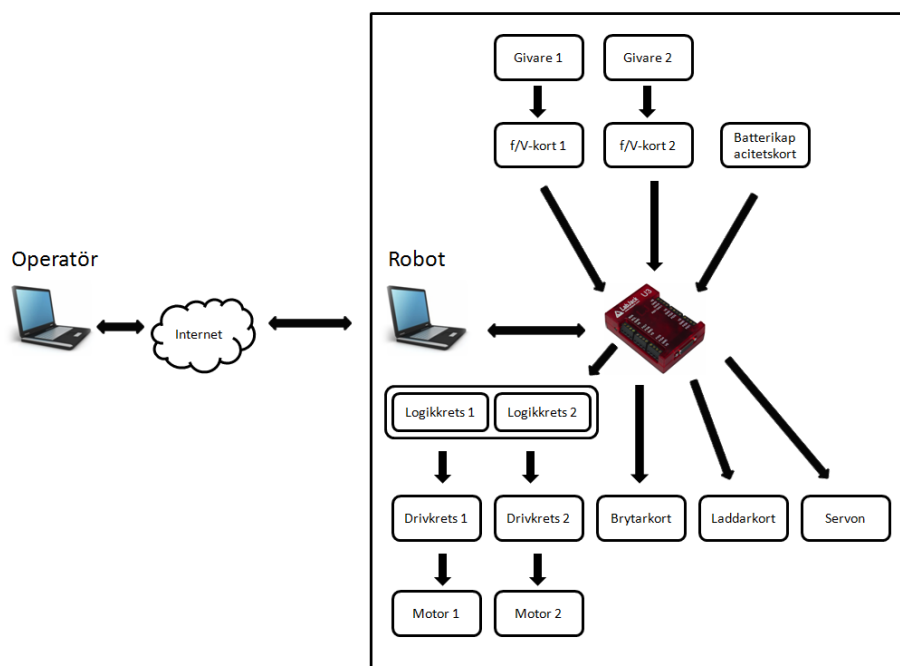
För att ge en bättre inblick i hur roboten fungerar och hur maskinvara och mjukvara samarbetar, kommer det att beskrivas i följande kapitel.

Motorerna som är kopplade till varsin drivkrets kommer att rotera när en PWM-puls öppnar och stänger drivkretsen med en viss ”duty cycle” och frekvens. Innan PWM-signalerna går till drivkretsarna så kommer de att passera ett kretskort bestående av två logikkretsar. Logikkretsarna används för att förhindra att en drivkrets kan få två styr signaler samtidigt. När motorerna roterar så kommer två induktiva givare att börja ge ut varsin PWM-signal. Ju snabbare hjulen roterar desto högre kommer PWM-signalernas frekvens vara. Givarnas PWM-signaler kommer att omvandlas till en spänning m.h.a. varsin frekvens till spänningsomvandlarkort. Den spänningen kommer i sin tur skickas in till I/O-kortet som ärvärde och motorernas hastighet kommer att regleras genom att variera dess PWM-signals ”duty cycle”. Två servomotorer användes för att kunna styra webbkameran horisontellt och vertikalt. De tar sin matningsspänning från I/O-kortet och styrs också m.h.a. en PWM-signal.

Brytarkortet hålls öppet så länge det får en hög signal på den ena ingången och en låg signal på den andra. Så ifall båda signalerna är höga eller låga så kommer kortet att vara stängt och ingen ström kan passera. På så sätt förhindras att övriga kretskort och främst motorerna får spänning när I/O-kortet får spänning (eftersom I/O-kortet alltid ger ut en kort spänningspuls på alla utgångar när matningsspänningen ansluts till det) och på så sätt förhindra en okontrollerad start. För att kunna öppna laddarkortet så användes en analog utgång från I/O-kortet. Anledningen till det är att det relä som det styr kräver en styrspänning på 5 V men de digitala utgångarna ger endast ut 3,3 V. Med en analog utgång kan man välja önskad utspänning från 0 V till 5 V. Den laddning som finns kvar i 12 V batteriet mäts genom att mäta spänningen över batteriet. Eftersom den spänningen är för stor för att avläsas av I/O-kortet så användes batterikapacitetskortet som består av en spänningsdelare. Spänningen från dess utgång läses av I/O-kortet och spänningsdelarens förhållande används för att i programmet beräkna vad den spänningen motsvarar.

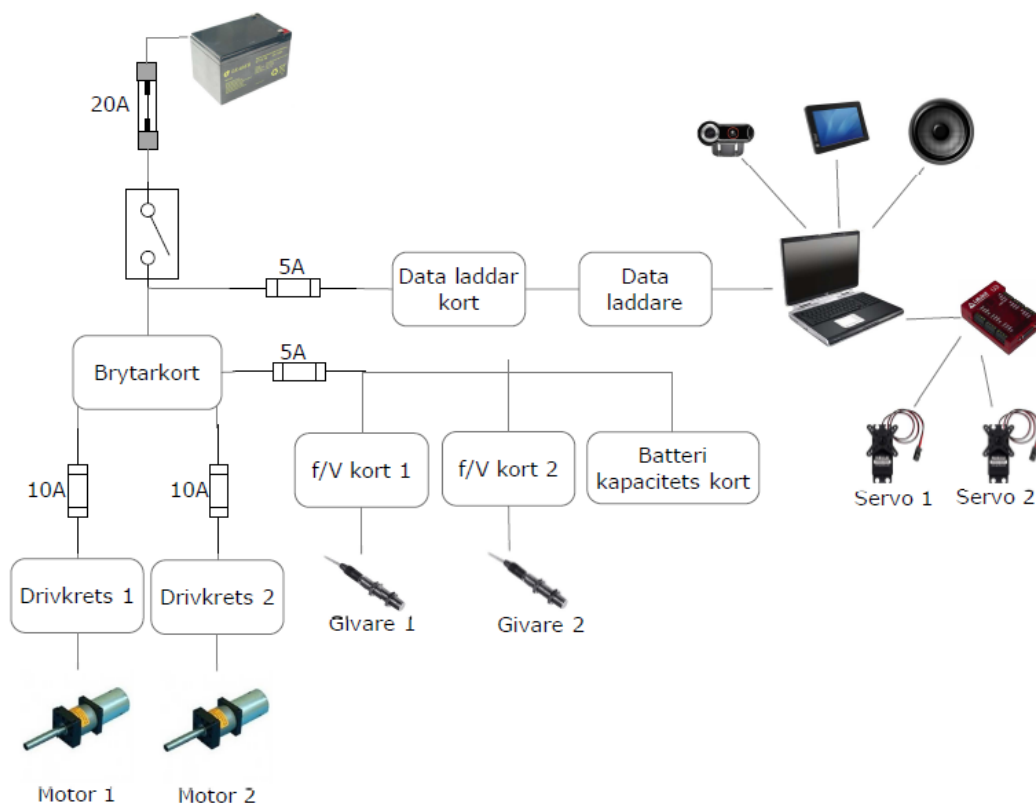
För att bearbeta och styra de signaler som I/O-kortet får och ska skicka ut så användes LabVIEW som styrprogram. LabVIEW finns installerat på en laptop som kommunicerar med I/O-kortet. Via LabVIEW:s frontpanel så kan man styra robotens motorer, servomotorer, övervaka batteriets kapacitet, styra brytarkortet och laddarkortet. För att kunna styra roboten så måste man först aktivera brytarkortet så att drivkretsarna får spänning. En eller två PWM-signaler (beroende på ifall man vill köra framåt/bakåt eller svänga) kommer att aktiveras på I/O-kortets utgång. En PID-regulator i programmet kommer att öka eller sänka på ”duty cyclen” tills f/V-kortens utspänning motsvarar börvärdet som angivits. Börvärdet justeras m.h.a. att ändra på hastighetslingan som finns på frontpanelen. När man styr servomotorerna för webbkameran så kommer programmet att minska eller öka på ”duty cyclen” för den PWM-signalen och på så sätt styra servomotorn.

Kommunikationen över Internet gjordes m.h.a. ett program som kallas Teamviewer. Det är ett ”remote desktop” liknande program som installeras på datorn och gör så att en operatör via Internet kan kontrollera en annan dator. På så sätt kan man kontrollera LabVIEW. För att kunna överföra bild via webbkameran och ljud via mikrofon och högtalare så finns det inbyggda funktioner i TeamViewer för att åstadkomma det. Det lönar sig dock att använda sig av Skype som är gjort för videokommunikation över Internet. I figur 30 kan man se ett blockdiagram över hur roboten styrs. För mer information om hur man styr och använder roboten se bilaga 3



Figur 30. Blockschemata för styrning av roboten.

Förutom styrsignalerna så måste även alla kretsar, motorer och givare m.m. få en matningsspänning. De flesta komponenter tar sin spänning från ett 12 V batteri men för vissa komponenter räcker det med att man tar ut en 5 V spänning från datorns USB-portar. Ett blockschema över varifrån de olika komponenterna får sin spänning och i vilken ordning de får den kan ses i figur 31.



Figur 31. Matningsspänningsblockschema.

Efter 12 V batteriet så finns en 20 A säkring och en mekanisk brytare. Från brytaren så kopplas en 5 A säkring och brytarkortet in. Brytarkortet som fungerar som en digital brytare öppnas och stängs m.h.a I/O-kortet. Ifall brytar kortet är öppet så leder den vidare spänningen till 2 st. 10 A säkringar som skyddar varsin drivkrets och motor och en 5 A säkring som skyddar f/V-korten och batterikapacitetskortet. De båda givarna tar sin matningsspänning från f/V-korten där deras matningsspänning bara byglas över till givarna.

Laddarkortet är inte kopplat via brytarkortet eftersom man vill ha möjlighet att ladda både 12 V batteriet och datorn även om datorn inte är påslagen. Ifall laddarkortet var kopplat via brytarkortet så kommer det inte att få någon spänning om datorn är avslagen, eftersom den förser I/O-kortet med spänning. Laddarkortet som består av ett relä stänger om det får en styrspänning. När det är öppet så ger det spänning åt en datorladdare. Den transformerar upp 12 V spänningen till 19 V och laddar datorn.

Från datorn så förses webbkameran, den externa skärmen, högtalarna och I/O-kortet med en 5 V spänning som de får via USB-portarna. I/O-kortet ger i sin tur också en 5 V spänning till servomotorerna som styr webbkameran.

8 Slutresultat

Roboten som började planeras i december 2010 var i maj 2011 färdig för provkörning. Roboten provkördes genom att användaren befann sig i Tyskland och roboten på Yrkeshögskolan Novia. Det visade sig fungera mycket bra. Både bilden och ljudet var av hög kvalitet och det gick bra att köra med roboten. Med andra ord resulterade projektet i en fungerande prototyp av en telepresence robot.

Stand-by tiden som var svår att bestämma eftersom man inte exakt visste hur mycket ström roboten skulle ta, visade sig vara helt enligt kraven. Vid provkörningar har man uppnått en stand-by tid på ända upp till fyra timmar.

Även om man konstruerade en fungerande prototyp så finns det en del saker som kan förbättras för att få en mera användarvänlig robot. Nedan kommer brister och förslag till förbättringar att diskuteras.

8.1 Svårkörd

Ett problem med roboten är att den är ganska svårkörd. Det finns flera orsaker till detta. En orsak är att det finns vissa fördröjningar i kommunikationen mellan datorerna, vilken kan variera mycket beroende på internetuppkoppling. Det största problemet med fördröjningen är att det leder till att bilden operatören ser blir hackig eller släpar. Även om bilden flyter på bra finns det vissa svårigheter i att se vart man kör eftersom kamerans synfält inte är alltför stort. Därför blir man tvungen att styra kamerans läge emellanåt för att se vart man kör för att undvika kollisioner.

En annan orsak till att den blir svårkörd är att den inte klarar av att köra helt rakt, utan drar åt båda hållen om vartannat. Detta beror på att PID-regulatorerna inte hinner med att kompensera. Orsaken till att regulatorerna inte hinner med att kompensera är att det finns en viss fördröjning av signalerna från f/V-omvandlarna.

8.2 Problem med framkomligheten

Ett av kraven på roboten var att den skulle ta sig över dörrtrösklar. Roboten konstruerades med stora hjul för att klara detta. Tyvärr visade det sig att den i alla fall inte uppfyllde detta krav. Orsaken till det är att hjulen är för små och i synnerhet stödhjulet längst bak. En annan orsak till att den har problem att komma över trösklar och andra hinder, såsom mattor och dylikt, är att motorerna styrs av PID-regulatorer. När roboten ska över ett hinder så krävs det större effekt på hjulen. Eftersom PID-regulatorerna är inställda på att vara långsamma, leder det till att det tar lång tid för dem att ge tillräckligt stort pådrag till motorerna för att roboten ska ta sig över hindren. Däremot om man kör över ett hinder med fart går det bättre.

8.3 För många program i användning

Även om kombinationen av TeamViewer, LabVIEW och Skype fungerar tillfredsställande, kunde man försöka utveckla en annan lösning på styrningen och kommunikationen. Detta eftersom datorn belastas för mycket och blir slö med alla program samtidigt i användning. För att lösa detta och kanske även få en snabbare kommunikation mellan datorerna kunde man i stället försöka få webbserverfunktionen i LabVIEW att fungera.

8.4 Kåpa och basstation saknas

På grund av tidsbrist hann man inte göra ett så bra utseende på roboten som det från början var tänkt. Resultatet blev en robot som ser lite halvfärdig ut eftersom ingen kåpa hann tillverkas till den. Möjligheten att montera en i efterhand finns dock. En annan sak som inte hann tillverkas var basstationen där det var meningen att roboten skulle laddas.

9 Diskussion

Det kan konstateras att det gått åt mycket mera tid än förväntat. Orsaken till detta var att projektet var mycket omfattande på grund av att det valdes att göra en egen plattform och det ledde till att mycket tid måste läggas på olika arbetsmoment, som man från början inte hade tänkt på. En annan orsak till varför projektet tog mycket tid var att det fanns väldigt lite information om telepresence robotar. Detta medförde att man var tvungen att göra egna tester och undersökningar, vilket från början även var väntat.

En av de stora utmaningarna var att få roboten att köra rakt fram. Det var en sak som man från början inte hade tänkt skulle bli ett så stort och tidskrävande problem som det slutligen blev. Mycket av tiden gick ut på att lösa detta, men det visade sig att man fick lön för mödan och fick en robot som kunde köra relativt rakt.

Slutligen vill vi tacka Yrkeshögskolan Novia och Jonas Waller som gett oss möjligheten att få jobba med detta projekt. Vi vill även rikta ett tack till Matts Nickull som hjälpt till med elektroniken.

Klemets diskussion

Målsättningen med projektet var som tidigare nämnt att tillverka en prototyp av en telepresence robot och en stor del av tiden gick i början åt att bestämma vilket typ av plattform som skulle användas. Valet att bygga en egen plattform tycker jag att var rätt eftersom man på så sätt kunde designa den som man ville, samtidigt fick man lära sig mer då man blev tvungen att fundera över alla komponenter som skulle användas.

Vid valet av motorerna så hade det varit intressant att använda sig av ett par stegmotorer i stället för de vanliga DC-motorerna. Det skulle ha varit tidsbesparande eftersom man då inte hade behövt sätta så mycket tid på drivkretsarna och dessutom skulle roboten troligtvis blivit lättare att styra.

Labview var ett intressant program att lära sig eftersom det är så mångsidigt och användbart i många olika sammanhang. Det tog ett tag att lära sig grunderna men gick

därefter bra att använda. Även om det främst används vid övervakning och mätning så fungerade det väldigt bra även till att styra roboten. Det är dock ett väldigt omfattande program och man hann endast lära sig en bråkdel av alla dess funktioner och kommandon.

Kommunikationen över Internet hade man till först hoppats på att kunna åtgärda genom att skapa en webbserver som man kunde ansluta via en webbläsare. Att skapa det var svårare än väntat eftersom det krävde bl.a. tillgång till skolans brandväggar. Möjlighet att kringgå brandväggarna valdes dock att inte undersökas vidare p.g.a. tidsbrist. I stället användes TeamViewer och Skype vilket också fungerade.

Att jobba två på detta projekt har varit positivt även om det funnits både för- och nackdelar. I början så övervägdes möjligheten att dela upp arbetet så att t.ex. den ena fokuserade på programvaran och den andra på maskinvaran. Det skulle möjligen varit lite mer tidsbesparande, men valdes att inte göras eftersom båda ville arbeta med lite av varje. Fördelen med det var att man hela tiden kunde bolla idéer mellan varandra eftersom man jobbade på samma problem, vilket man inte kunnat göra om vi jobbat med två olika saker. Nackdelen med att jobba tillsammans var skrivandet. Där delades arbetet upp så man fokuserade på skilda saker men det var ändå problematiskt att slå ihop arbetet så att man fick en sammanhängande text. En sak som skulle varit intressant är om man skulle ha varit ännu fler på detta projekt, men från olika avdelningar t.ex. en från informationsteknik, en från maskinteknik osv. Det skulle inneburit att varje person skulle kunna fokusera på sitt område så som det blir i arbetslivet, vilket också troligen hade gett ett ännu bättre resultat.

Detta projekt har trots att det varit tidskrävande varit väldigt intressant och lärorikt. Eftersom det innehållit så många olika delmoment som t.ex. elektronik, programmering, reglering och motordimensionering så har det inneburit att man kunnat bredda sina kunskaper inom dessa områden samtidigt som det varit väldigt varierande.

Siffréns diskussion

Från att projektet började tills att man kunde göra de första provkörningarna tog det ganska länge. Detta eftersom vi ville vara säkra på att vi faktiskt gjorde de rätta valen av komponenter och funktionssätt. Valen av komponenter och funktionssätt är jag personligen nöjd med, men visst kan man konstatera med facit i hand att det finns saker som vi kunde ha gjort annorlunda.

En sak som kan nämnas är valet av motorer. Motorerna som man valde fungerar visserligen bra, men med tanke på hur svårt det blev att få roboten att köra rakt fram hade man kunnat använda någon annan typ av motorer, t.ex. stegmotorer för att lättare kontrollera rotationshastigheten. Valet av dessa motorer var i och för sig aktuella en tid, men med tanke budgeten valdes i stället andra motorer.

Eftersom vi har varit två som skrivit denna rapport har det funnits en del svårigheter i att få en sammanhållande text. Det har även tagit ganska mycket tid att sammanfoga våra texter till ett fullständigt dokument. Samarbetet mellan mig och Jonatan tycker jag ändå fungerade bra och vi lyckades dela upp arbetet på ett bra sätt både gällande den praktiska delen och den skriftliga delen. En fördel att vara två var att vi kunde hjälpas åt och diskutera sinsemellan när det fanns saker som var oklara.

En annan sak som var svår var att göra en tidsplan för projektet. Detta eftersom man inte kunde veta hur länge ett arbetsmoment skulle ta och man blev ibland tvungen att gå tillbaka till ett arbetsmoment för att antingen göra förbättringar eller helt och hållet tänka om. Eftersom man jobbade lite fram och tillbaka med olika arbetsmoment och ändrade på några saker efterhand, var det även svårt att börja skriva rapporten parallellt med det praktiska arbetet.

Projektet har varit mycket intressant och jag har fått mycket ny kunskap inom många olika områden. Under projektets gång har ett intresse för forskning och utveckling väckts och jag skulle mycket gärna kunna tänka mig att arbeta med något liknande i framtiden.

Källförteckning

Litteratur

- /1/ Eklund, M. (2010). *Telepresence robotar – Intelligent system. Inlämningsrapport* Vasa: Yrkeshögskolan Novia.
- /2/ Everett, H.R. (1995). *Sensors for mobile robots – Theory and application*. Natick: A K Peters, Ltd. s. 48-61
- /3/ Jones, J.L., Flynn, A.M. & Seiger, B.A. (1998). *Mobile robots – Inspiration to Implementation*. Natick: A K Peters, Ltd. s. 169-178
- /4/ Jones, J.L., Flynn, A.M. & Seiger, B.A. (1998). *Mobile robots – Inspiration to Implementation*. Natick: A K Peters, Ltd. s. 265
- /5/ McComb, G. & Predko, M. (2000). *Robot Builders Bonanza*. New York: McGraw-Hill. s. 190-192
- /6/ McComb, G. & Predko, M. (2006). *Robot Builders Bonanza*. New York: McGraw-Hill. s. 10-12
- /7/ McComb, G. & Predko, M. (2006). *Robot Builders Bonanza*. New York: McGraw-Hill. s. 17-18
- /8/ National Instruments. (2007). *Introduction to LabVIEW and Computer-Based Measurements Hands-On Seminar*
- /9/ Niku, S.B. (2010). *Introduction to robotics – Analysis, control, applications*. Hoboken: John Wiley & Sons, Inc. s. 3-4
- /10/ Valentine, R. (1998). *Motor Control Electronics Handbook: Motor Control Overview*. New York: McGraw-Hill

Elektroniska källor

/11/ Anybots

www.anybots.com

(Hämtat: 3.4.2011)

/12/ Beam Wiki

Ackerman steering

http://www.beam-wiki.org/wiki/Image:Ackermann_Steering_Radius.GIF

(Hämtat: 14.2.2011)

/13/ Boley, L. Brian (1997)

Overview of Motor Types Tutorial

http://www.oddparts.com/acsi/motortut.htm#DC_MOTOR

(hämtat 15.02.2011)

/14/ Cognitoware

Differential drive

<http://cognitoware.com/tutorials/DifferentialDrive.htm>

(Hämtat: 14.2.2011)

/15/ Digital Nemesis Pty Ltd (senast uppdaterad 2008)

How R.C. Servo Motors Work

<http://www.digitalnemesis.com/info/docs/rcservo/>

(hämtat: 28.02.2011)

/16/ Electricly (Senast uppdaterad 2011)

Tricycle Drive

<http://electricly.com/tricycle-drive/>

(Hämtat: 14.2.2011)

- /17/ Ericsson (senast uppdaterad 2003)
Stepper Motor Basics
<http://www.solarbotics.net/library/pdflib/pdf/motorbas.pdf>
(hämtat: 28.02.2011)
- /18/ Global Crossing
Telepresence With Teliris
http://www.globalcrossing.com/NA/customer/collaboration/cust_telepresence_over.aspx
(Hämtat: 28.3.2011)
- /19/ Itechnews (Senast uppdaterad 2008)
WooWee Rovio Wifi Mobile Webcam
<http://www.itechnews.net/tag/wifi/>
(Hämtat: 14.2.2011)
- /20/ LabJack
U3 User's Guide
<http://labjack.com/support/u3/users-guide>
(Hämtat: 17.2.2011)
- /21/ LabJack
<http://labjack.com/u3>
(Hämtat: 16.4.2011)
- /22/ McMain, Chuck, Chuck's Robotics Notebook (senast uppdaterad 2006)
H-bridge Theory and Practice
<http://www.mcmanis.com/chuck/robotics/tutorial/h-bridge/>
(hämtat: 05.05.2011)
- /23/ National Instruments (senast uppdaterad 2011)
Multiple Programming Approaches in NI LabVIEW
<http://www.ni.com/labview/whatis/programming-approaches/>
(hämtat: 17.05.2011)

- /24/ National Instruments (senast uppdaterad 2010)
Pulse Width Modulation
<http://zone.ni.com/devzone/cda/tut/p/id/2991>
(hämtat: 15.05.2011)
- /25/ National Semiconductor (senast uppdaterad 2011)
LM2907 Tachometer/Speed Switch Building Block Applications
<http://www.national.com/profile/snip.cgi/openDS=LM2907/>
(hämtat: 18.05.2011)
- /26/ pc-control.co.uk (2008)
DC motors
<http://www.pc-control.co.uk/dc-motors.htm>
(hämtat: 27.02.2011)
- /27/ ProAV – Thom Mullins, CTS (Senast uppdaterad 2008)
Being There
<http://www.proavmagazine.com/Acoustics/being-there.aspx>
(Hämtat: 9.3.2011)
- /28/ Robotshop (senast uppdaterad 2011)
RS540 brushed DC motor
<http://www.robotshop.com/eu/banebots-p60-gearbox-rs540-brushed-motor-kit.html>
(hämtat: 01.03.2011)
- /29/ Ross, Kevin, Seattle Robotics Society (senast uppdaterad 2004)
Whats a Servo?
<http://www.seattlerobotics.org/guide/servos.html>
(hämtat: 28.02.2011)
- /30/ San Diego state University
LabView Tutorial 7 - Case Structures
<http://attila.sdsu.edu/me205/modules/1labview/07casestructs/2tutorial.html>
(hämtat: 19.09.2011)

- /31/ Society of Robots (senast uppdaterad 2010)
Robot Batteries
<http://www.societyofrobots.com/batteries.shtml>
(hämtat: 14.03.2011)
- /32/ Stutz, Michael (senast uppdaterad 2000)
Brushless DC motor
http://www.allaboutcircuits.com/vol_2/chpt_13/6.html
(hämtat: 28.02.2011)
- /33/ Tantos, Andras (senast uppdaterad 2007)
The H-bridge secrets
http://www.modularcircuits.com/h-bridge_secrets1.htm/
(hämtat: 06.05.2011)
- /34/ TeamViewer (Senast uppdaterad 2011)
Manual - TeamViewer 6.0
http://www.teamviewer.com/download/teamviewer_manual.pdf
(Hämtat: 23.3.2011)
- /35/ *Autonomous robot* (Senast uppdaterad 2011)
http://en.wikipedia.org/wiki/Autonomous_robot
(Hämtat: 3.4.2011)
- /36/ *LabVIEW* (senast uppdaterad 2011)
<http://en.wikipedia.org/wiki/LabVIEW/>
(hämtat: 17.05.2011)
- /37/ *Mecanum wheel* (Senast uppdaterad 2011)
http://en.wikipedia.org/wiki/Mecanum_wheel
(Hämtat: 15.1.2011)

- /38/ *Remote desktop software* (Senast uppdaterad 2011)
http://en.wikipedia.org/wiki/Remote_desktop_software
(Hämtat: 23.3.2011)
- /39/ *Robot* (Senast uppdaterad 2011)
<http://en.wikipedia.org/wiki/Robot>
(Hämtat: 2.2.2011)
- /40/ *TeamViewer* (Senast uppdaterad 2011)
<http://en.wikipedia.org/wiki/TeamViewer>
(Hämtat: 23.3.2011)
- /41/ *Teleoperation* (Senast uppdaterad 2010)
<http://en.wikipedia.org/wiki/Teleoperation>
(Hämtat: 3.4.2011)
- /42/ *Telepresence* (Senast uppdaterad 2011)
<http://en.wikipedia.org/wiki/Telepresence>
(Hämtat: 12.1.2011)
- /43/ Zero Emission Vehicles Australia, (senast uppdaterad 2011)
Tech Info: Motors
<http://www.zeva.com.au/tech.php?section=motors>
(hämtat: 16.02.2011)
- /44/ Zoltan, Eric (Senast uppdaterad 2005)
Wireless robotics: Mobile robots
http://www.egr.uh.edu/courses/ece/ece4437/support/notes_hardware/Wireless%20robotics%20%20Mobile%20robots.htm
(Hämtat: 14.2.2011)

Bilagor

Bilaga 1 - Tangentbordssubrutin

Bilaga 2 - Huvudprogram

Bilaga 3 - Användarmanual

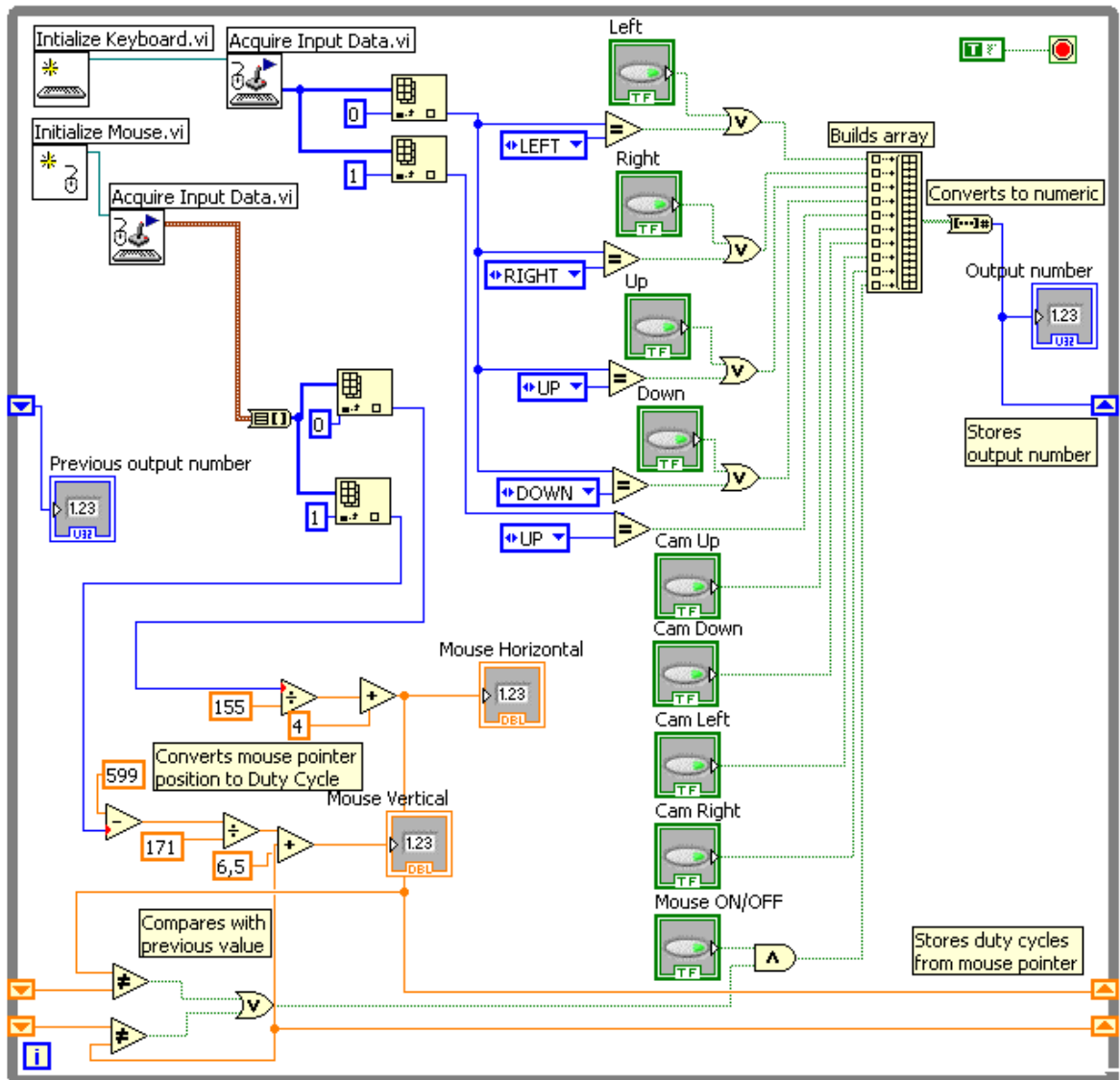
Bilaga 4 - Kretsschema på brytarkort

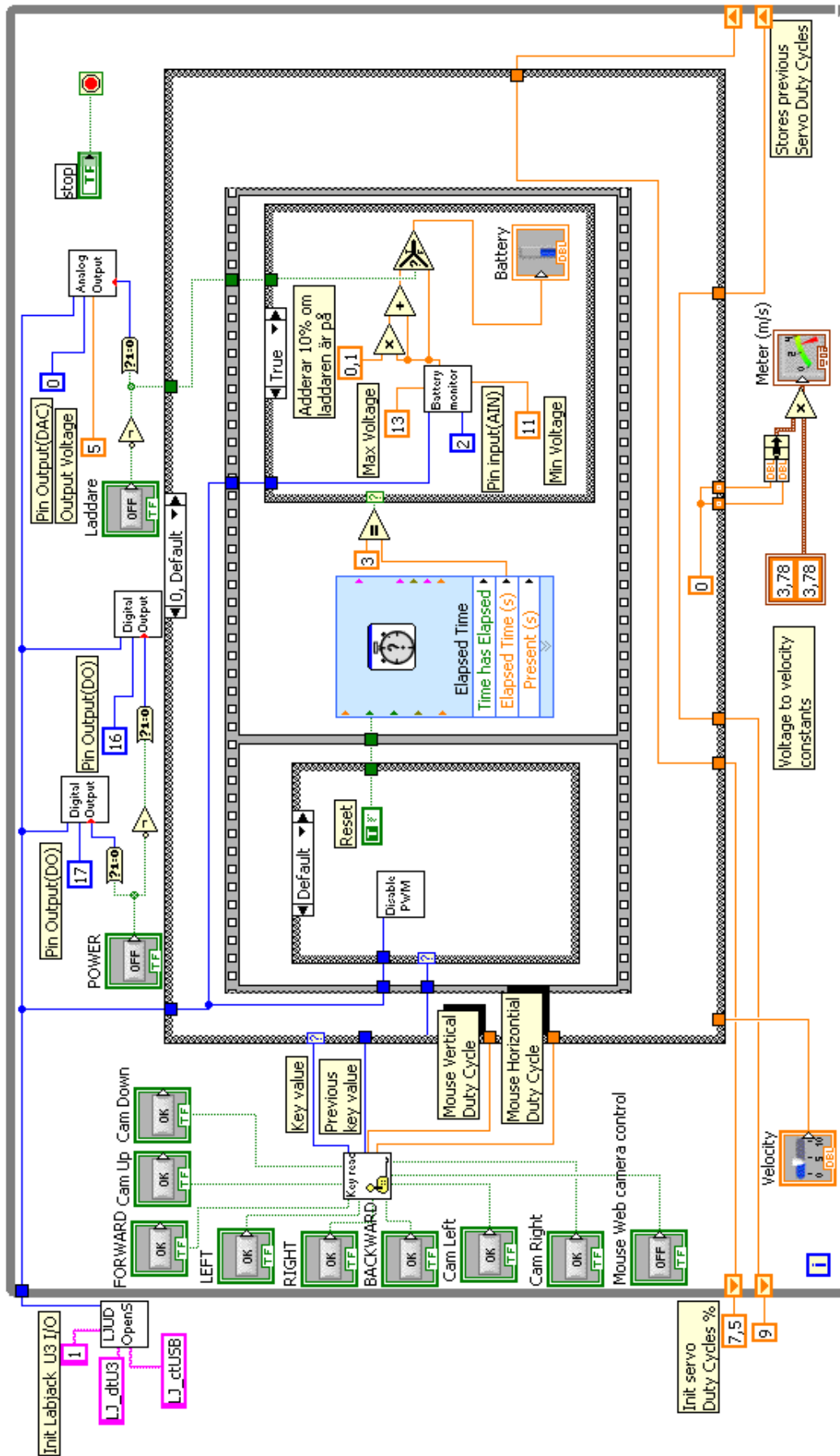
Bilaga 5 - Kretsschema på drivkrets

Bilaga 6 - Kretsschema på logikkrets

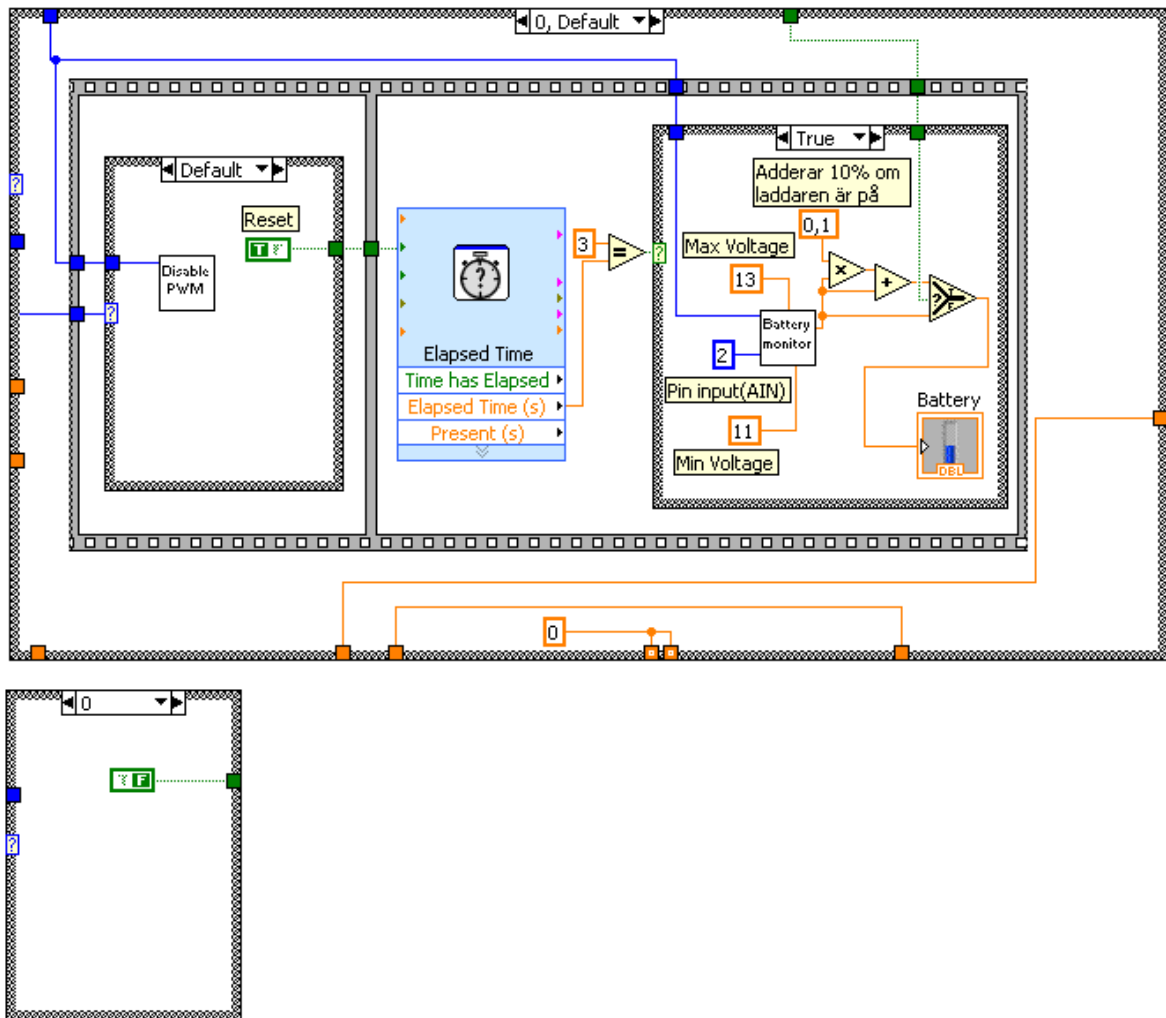
Bilaga 7 - Kretsschema - f/V-omvandlare

Bilaga 8 - Bilder på roboten



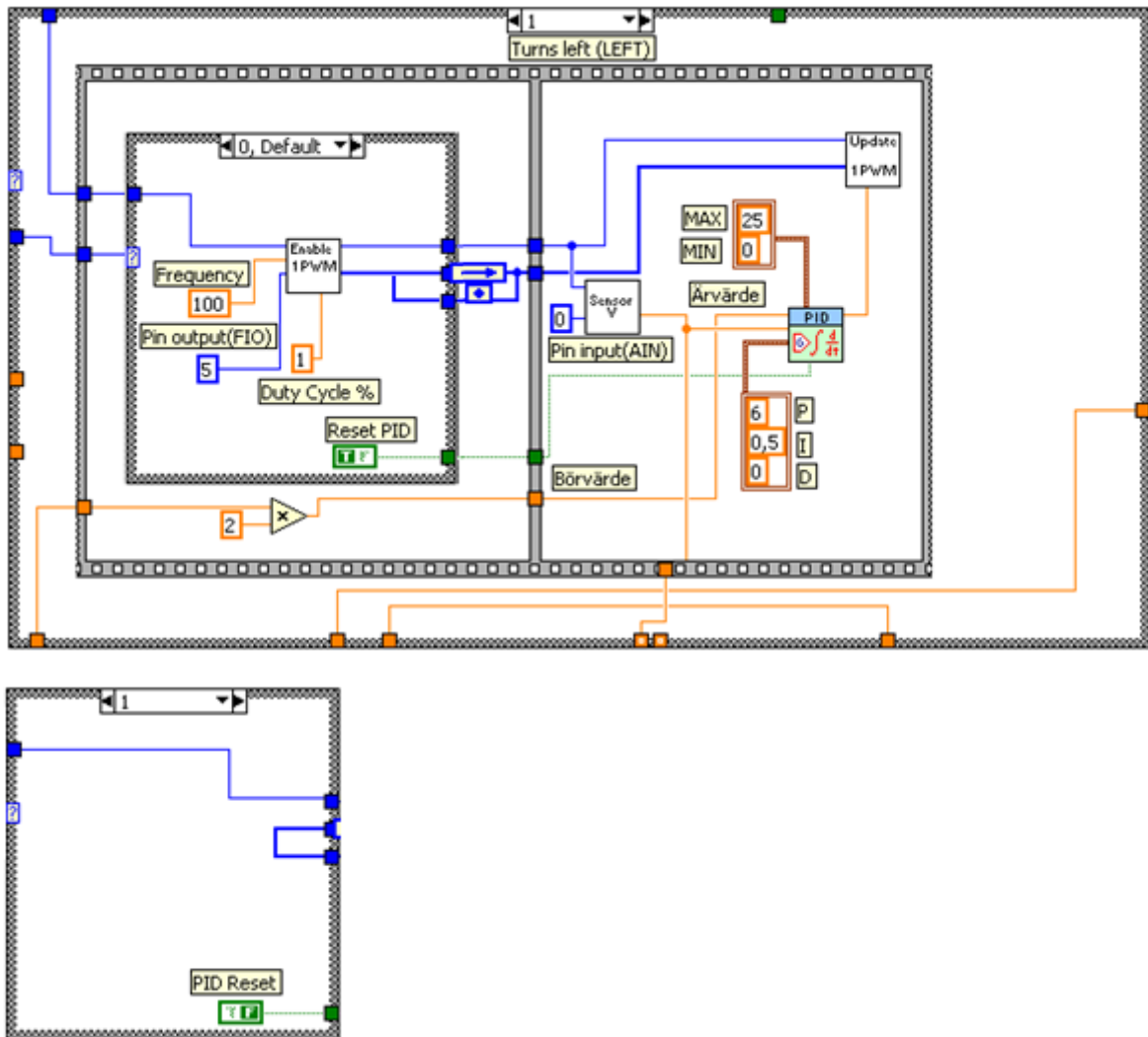


”0” Default



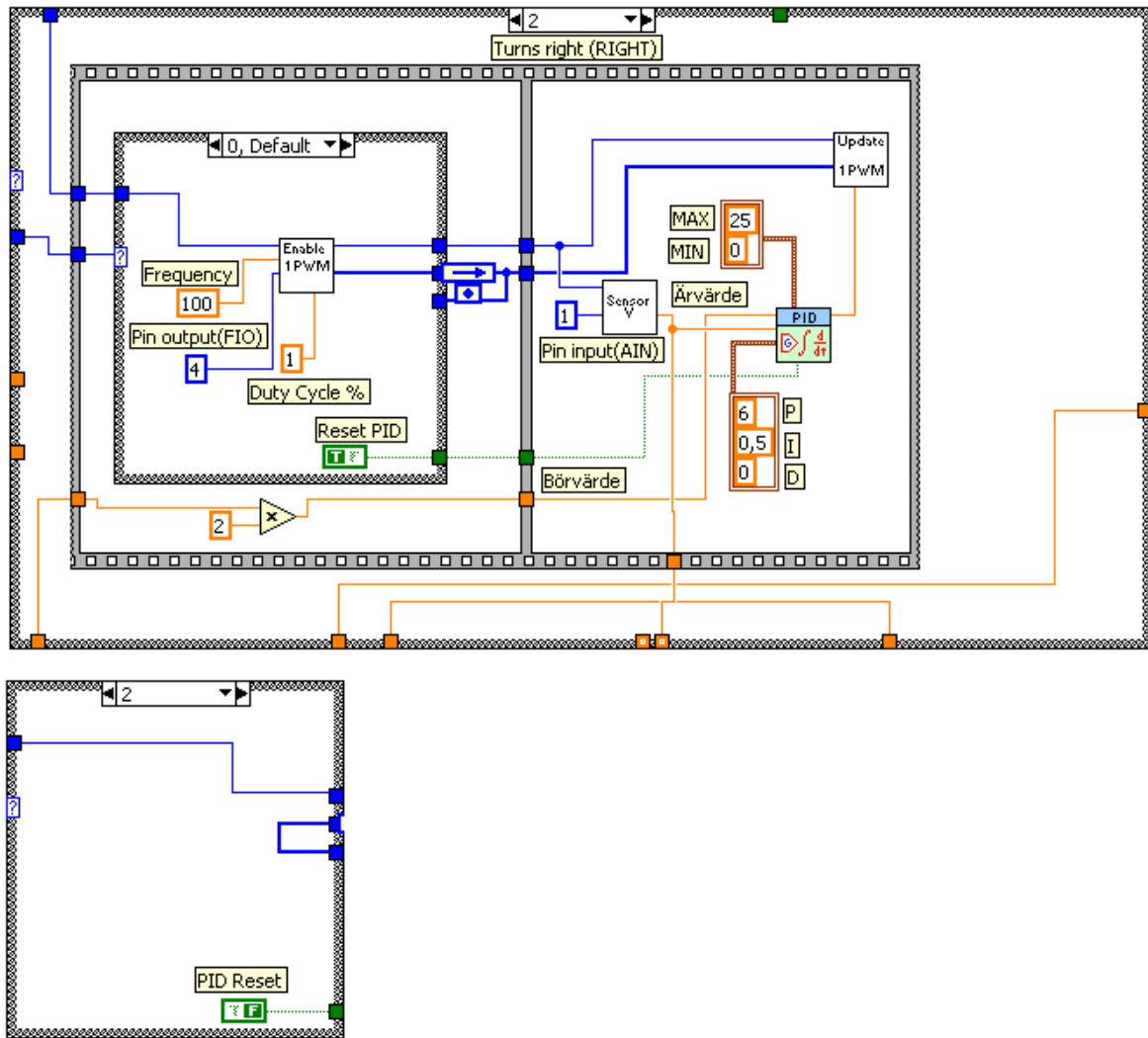
1. Nollställer PWM-signalerna.
2. Avläser spänningen över batteriet var 3:e sekund.

”1” Sväng vänster



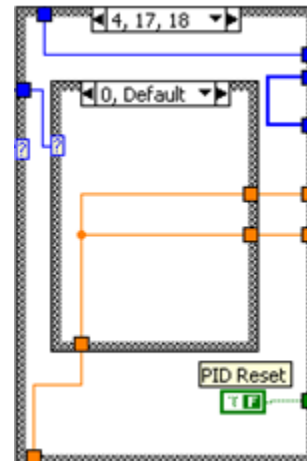
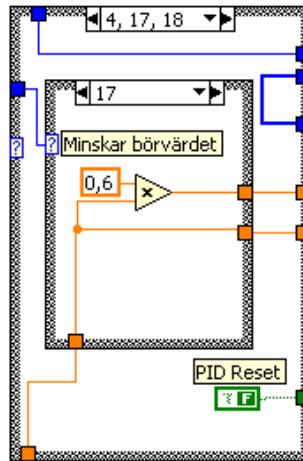
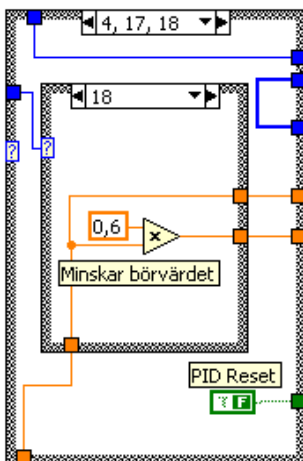
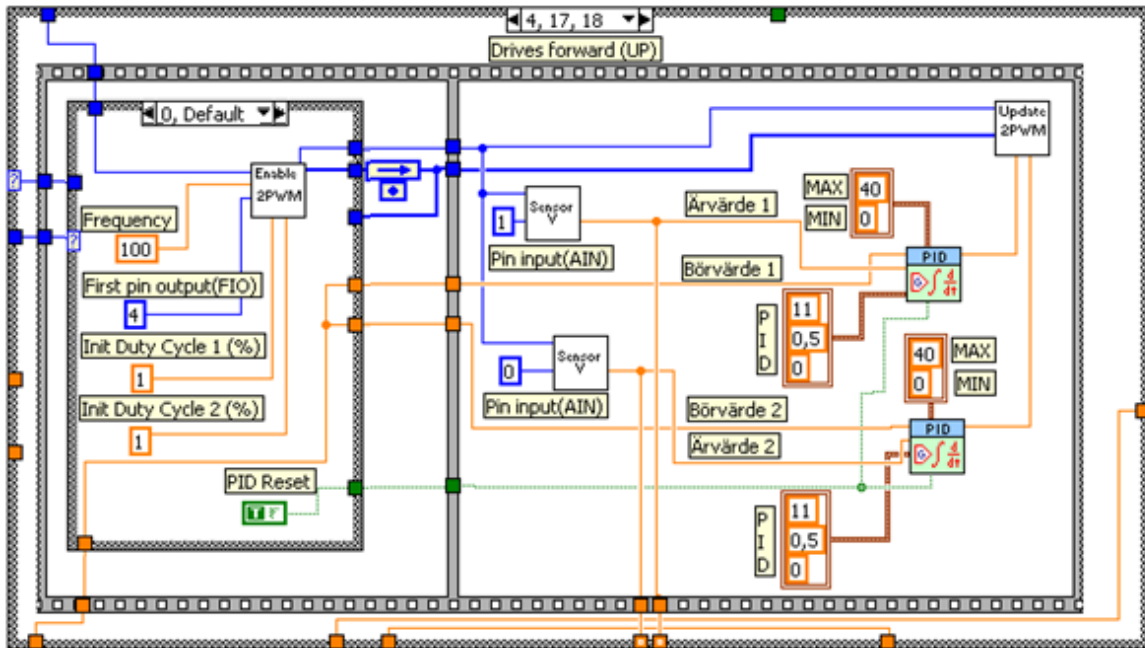
1. Initialiserar en PWM-signal som får den att svänga åt vänster ifall det föregående värdet inte var en 1:a.
2. Reglerar och uppdaterar ”duty cyclen” så att hjulets rotationshastighet motsvarar börvärdet.
3. Ifall föregående värde var en 1:a så hoppar den över PWM-initialiseringen och går direkt till PWM-regleringen.

”2” Sväng höger



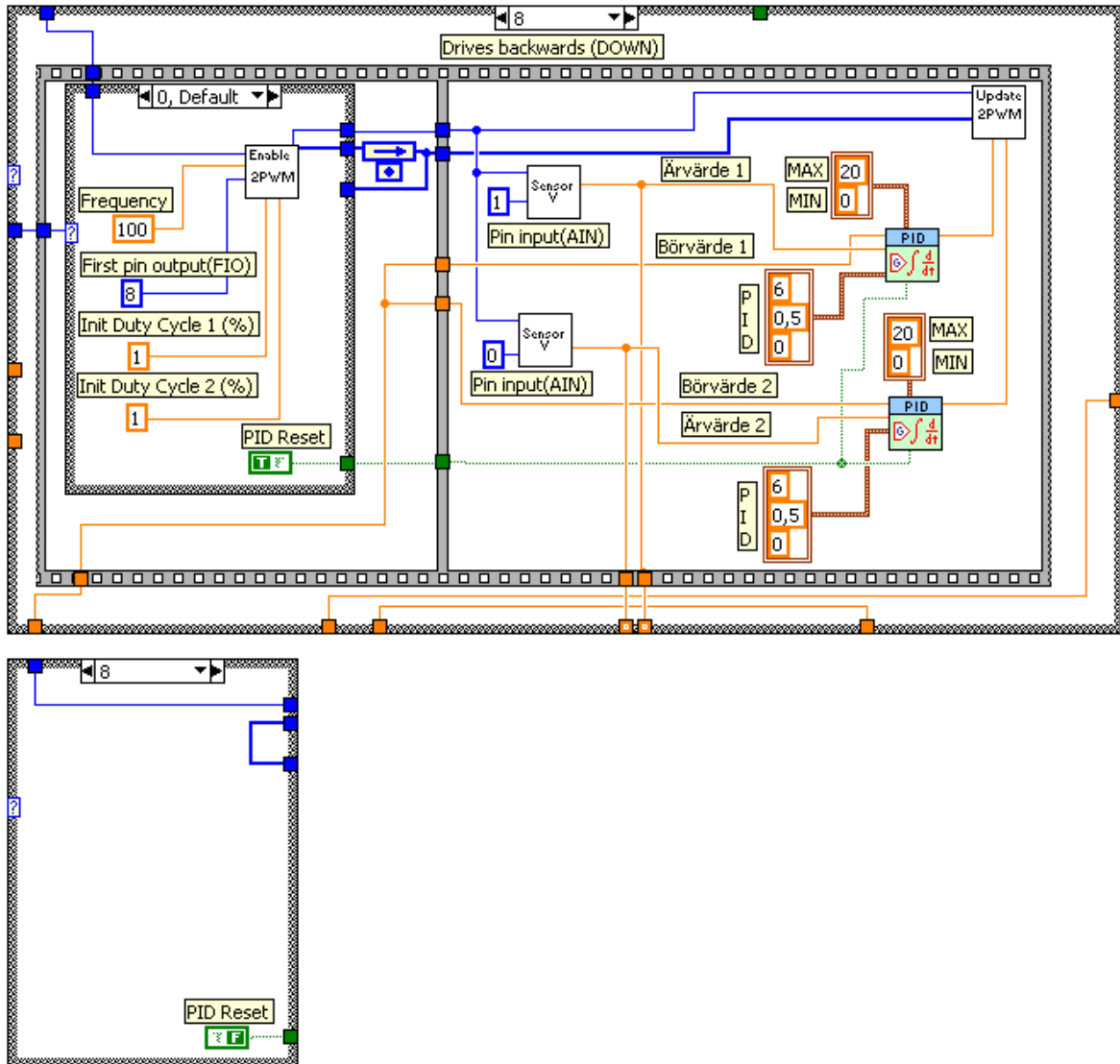
1. Initialiserar en PWM-signal som får den att svänga åt höger ifall det föregående värdet inte var en 2:a.
2. Reglerar och uppdaterar ”duty cyclen” så att hjulets rotationshastighet motsvarar börvärdet.
3. Ifall föregående värde var en 2:a så hoppar den över PWM-initialiseringen och går direkt till PWM-regleringen.

”4”, ”17”, ”18” Kör framåt/snett framåt



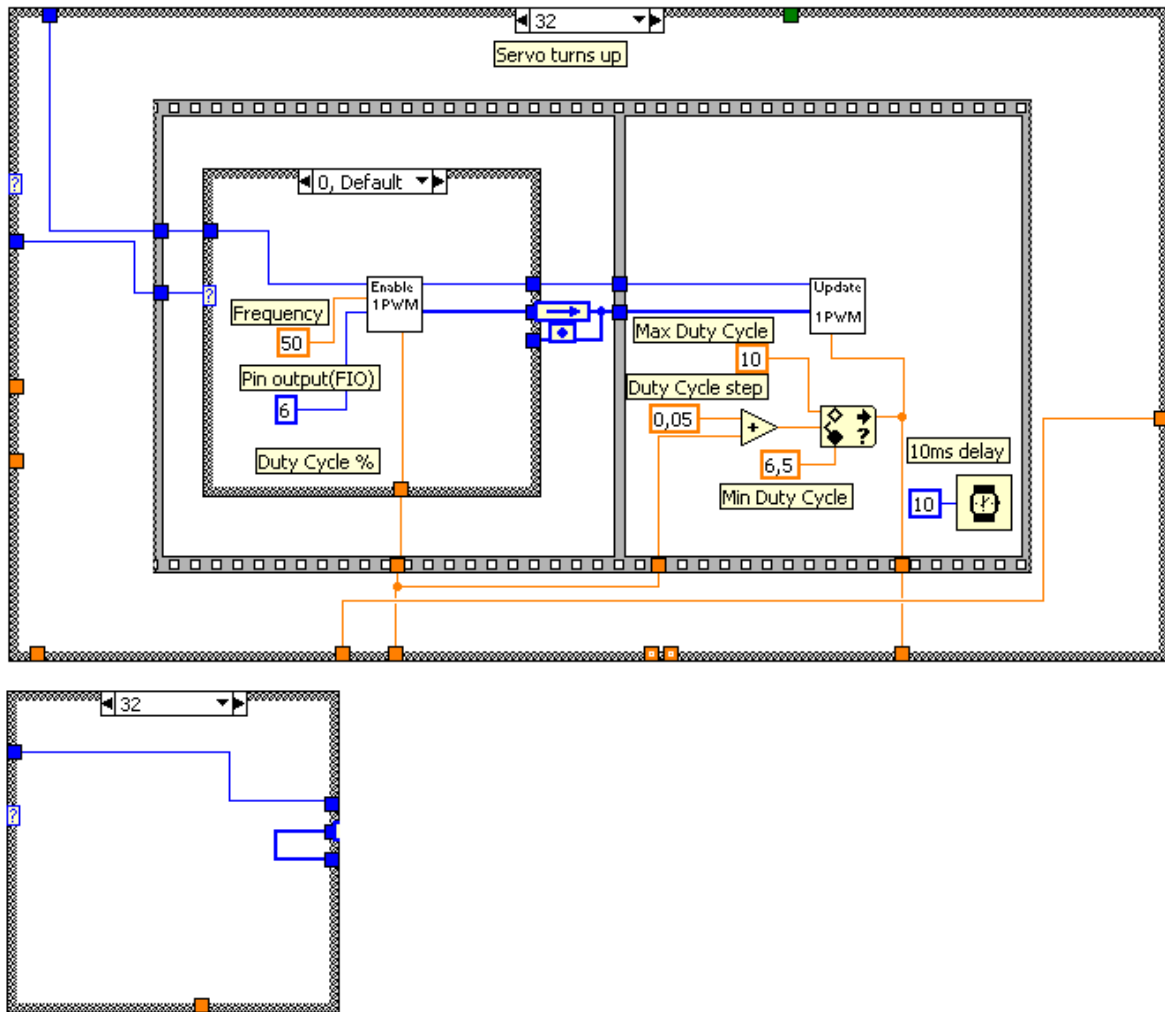
1. Initialiserar två PWM-sigaler som får den att köra fram ifall det föregående värdet inte var 4, 17 eller 18.
2. Ifall värdet är 4 så är båda hjulens börvärden lika stora men ifall värdet är 16 eller 32 så är det ena hjulets börvärde lite mindre vilket får roboten att köra ”snett” framåt.
3. Reglerar och uppdaterar ”duty cyclerna” så att hjulens rotations hastighet motsvarar börvärden.

”8” Kör bakåt



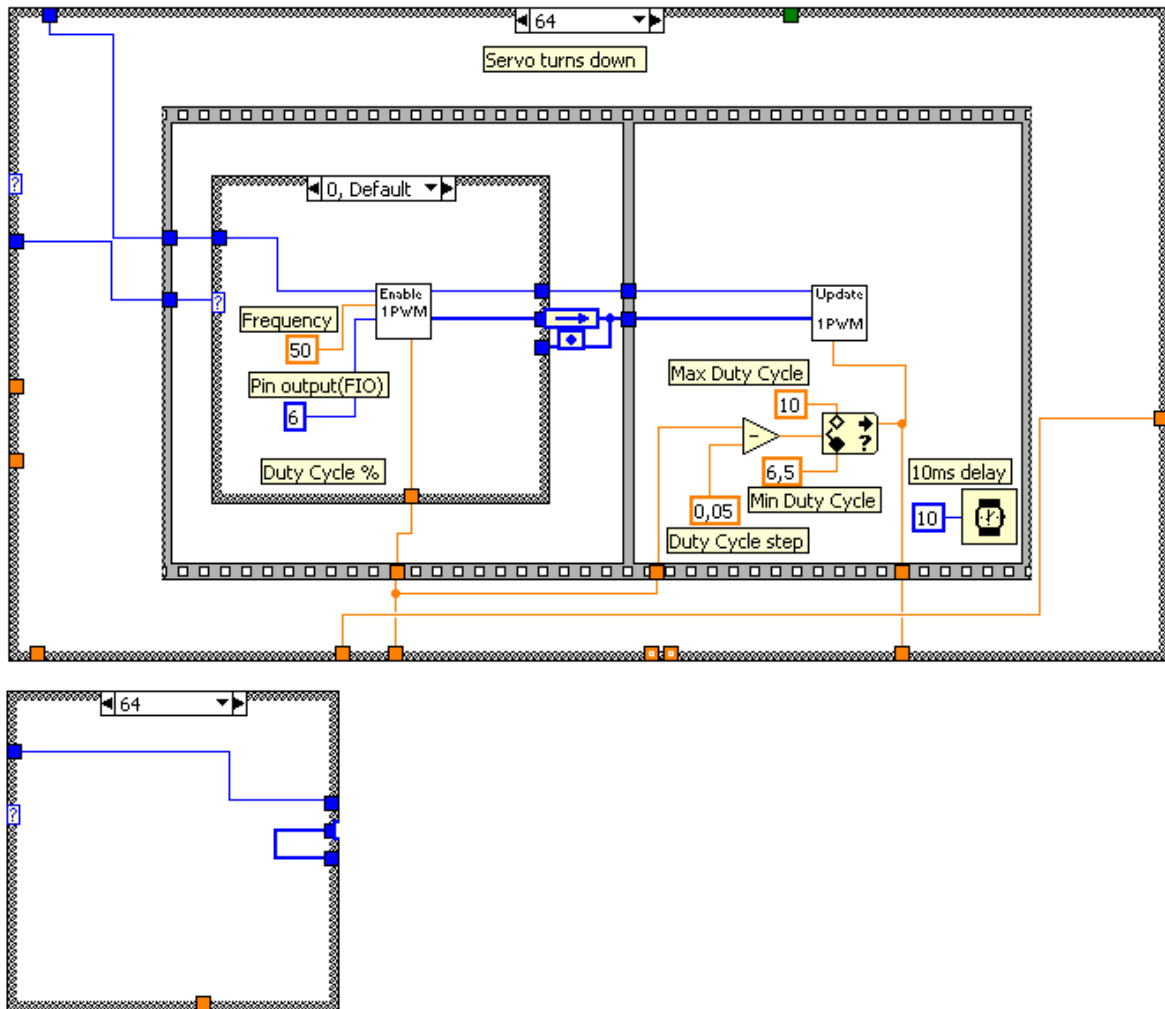
1. Initialiserar två PWM-signaler som får den att köra bakåt ifall det föregående värdet inte var 8.
2. Reglerar och uppdaterar ”duty cyclerna” så att hjulens rotations hastighet motsvarar börvärdet.
3. Ifall föregående värde var en 8:a så hoppar den över PWM-initialiseringen och går direkt till PWM-regleringen

”32” Webbkameran svänger uppåt



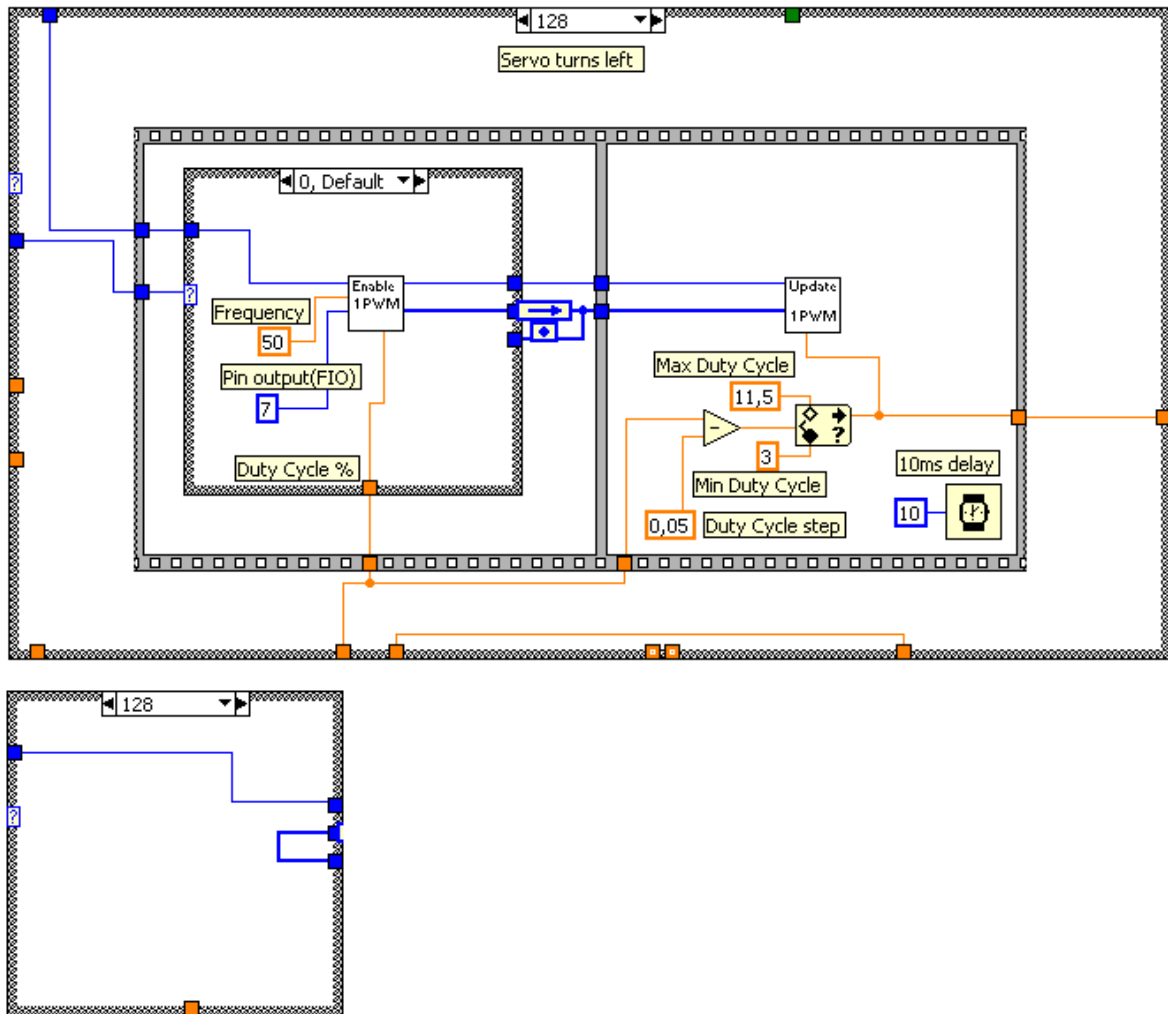
1. Initialiserar en PWM-signal som styr webbkameran lodrätt ifall föregående värde inte var 32.
2. Adderar 0,05 till den föregående ”duty cyclen” vilket får webbkameran att svänga uppåt.
3. Ger ut ”duty cyclens” värde som kommer att sparas i ett ”shift register”.

”64” Webbkameran svänger nedåt



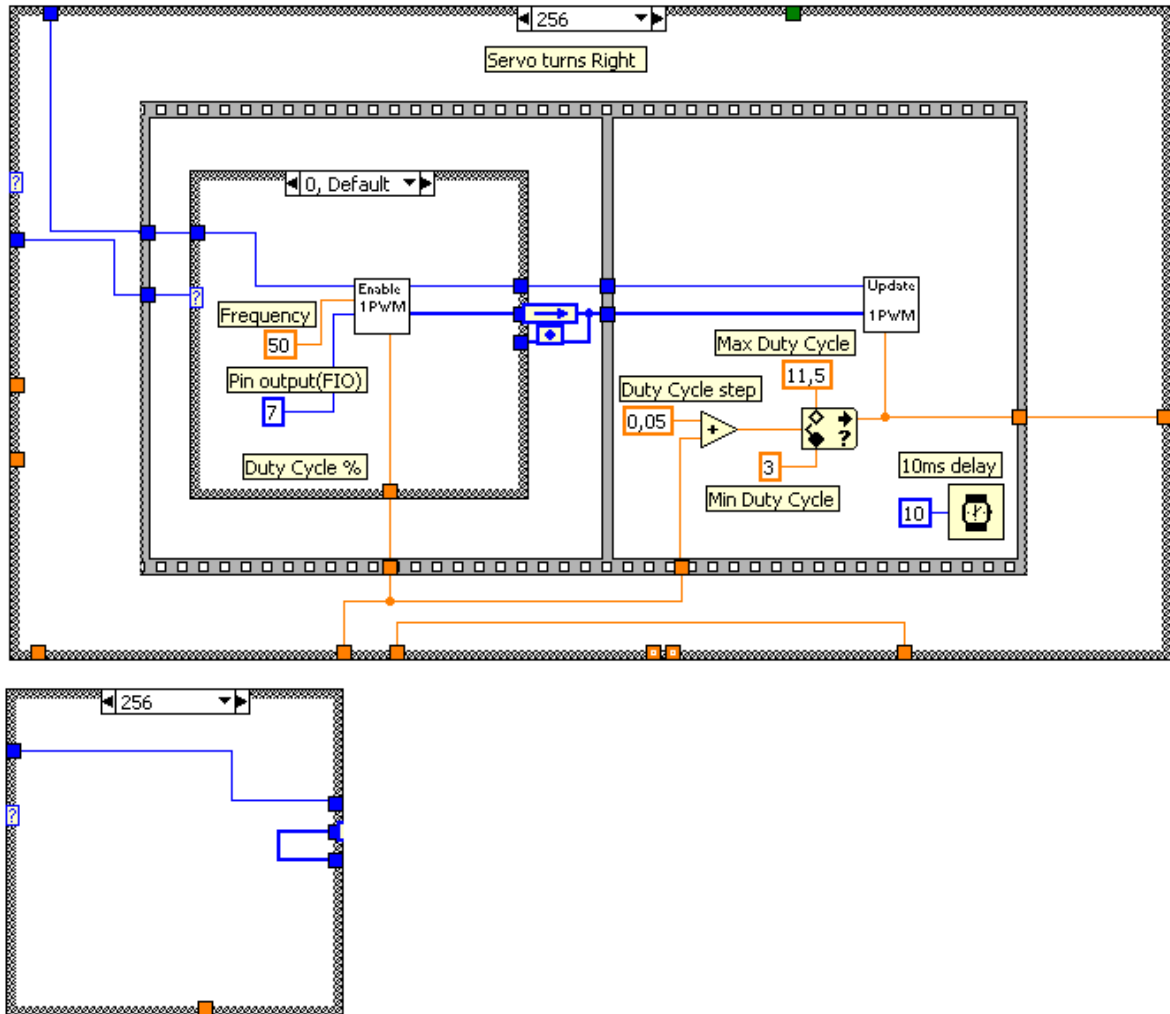
1. Initialiserar en PWM-signal som styr webbkameran lodrätt ifall föregående värde inte var 64.
2. Subtraherar 0,05 till den föregående ”duty cyclen” vilket får webbkameran att svänga nedåt.
3. Sparar ”duty cyclens” värde i ett ”shift register”.

”128” Webbkameran svänger åt vänster



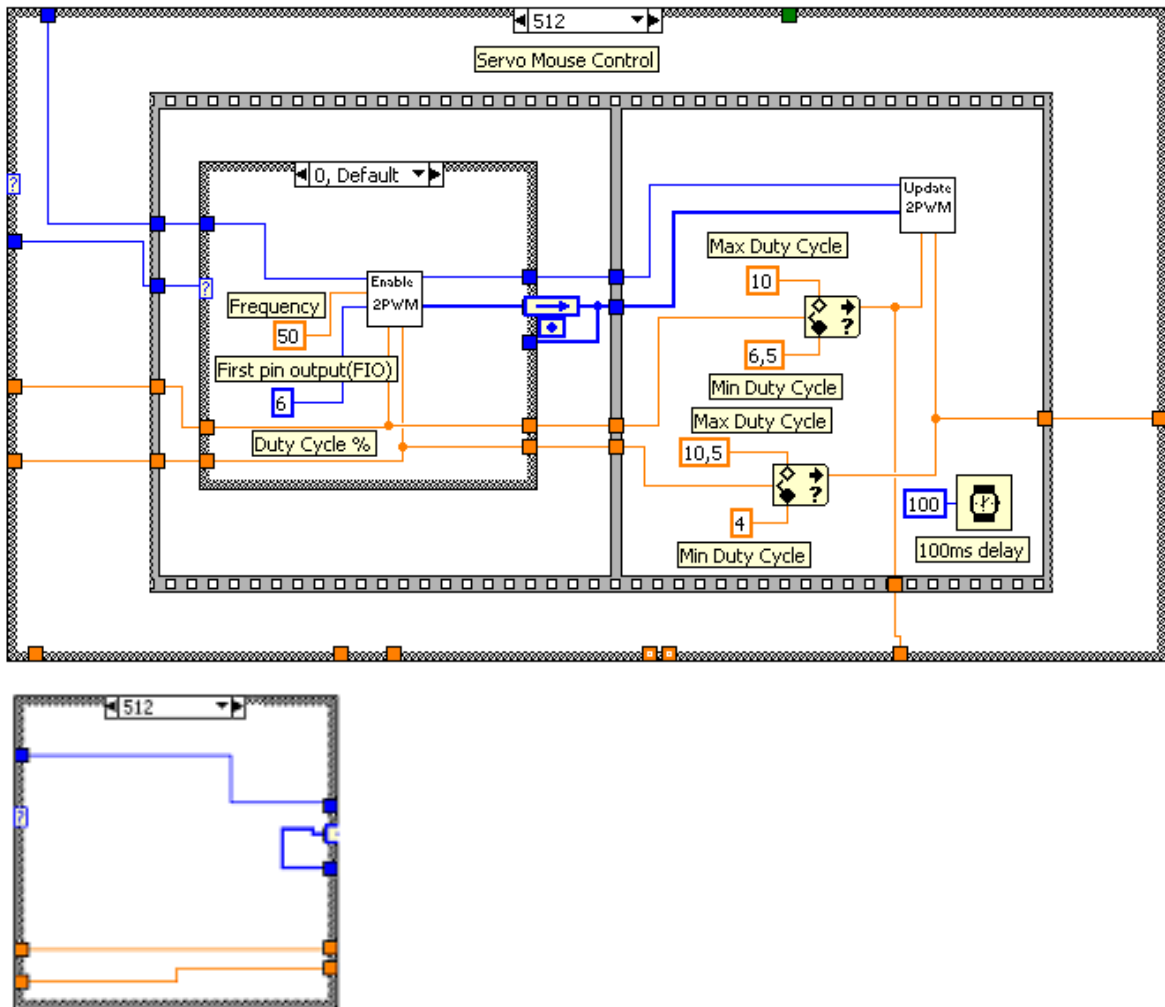
1. Initialiserar en PWM-signal styr webbkameran vågrätt ifall föregående värde inte var 128.
2. Subtraherar 0,05 från den föregående ”duty cyclen” vilket får webbkameran att svänga åt vänster.
3. Sparar ”duty cyclens” värde i ett ”shift register”.

”256” Webbkameran svänger åt höger



1. Initialiserar en PWM-signal styr webbkameran vågrätt ifall föregående värde inte var 256.
2. Adderar 0,05 till den föregående ”duty cyclen” vilket får webbkameran att svänga åt höger.
3. Sparar ”duty cyclens” värde i ett ”shift register”.

”512” Webbkameran styrs med musen



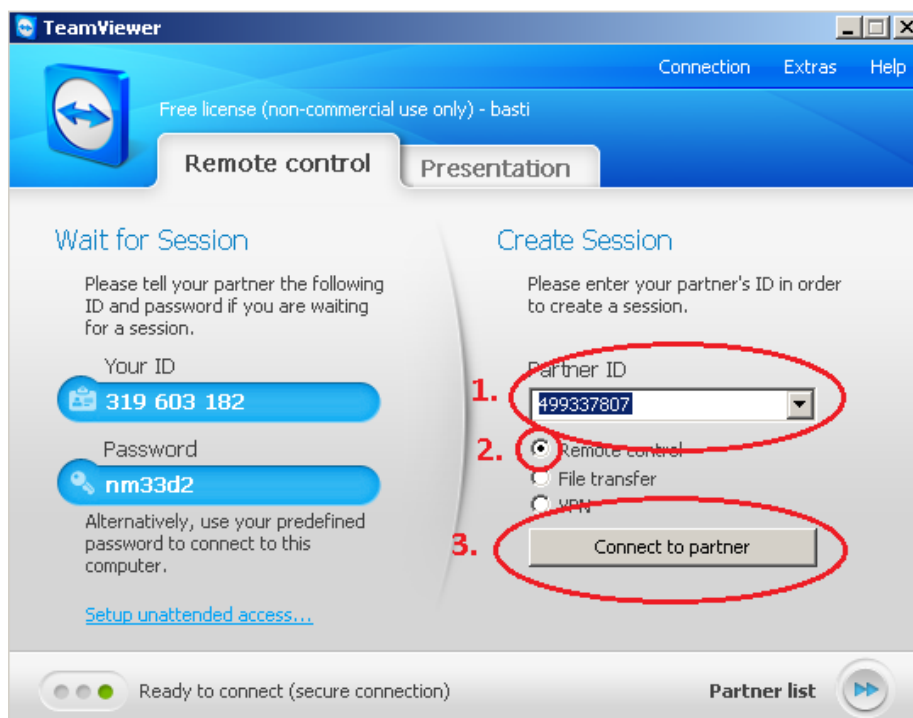
1. Initialiserar två PWM-sigaler som styr webbkameran vågrätt och lodrätt ifall föregående värde inte var 512.
2. Uppdaterar de båda ”duty cyclerna” till de värden som fås från muspekarens position.

Ladda ner och installera Teamviewer

- Ladda ner TeamViewer från www.teamviewer.com
- Installera!

Ansluta till Roboten

- Fyll i robotens ID-nummer
- Kryssa för "Remote control"
- Klicka på "Connect to partner"

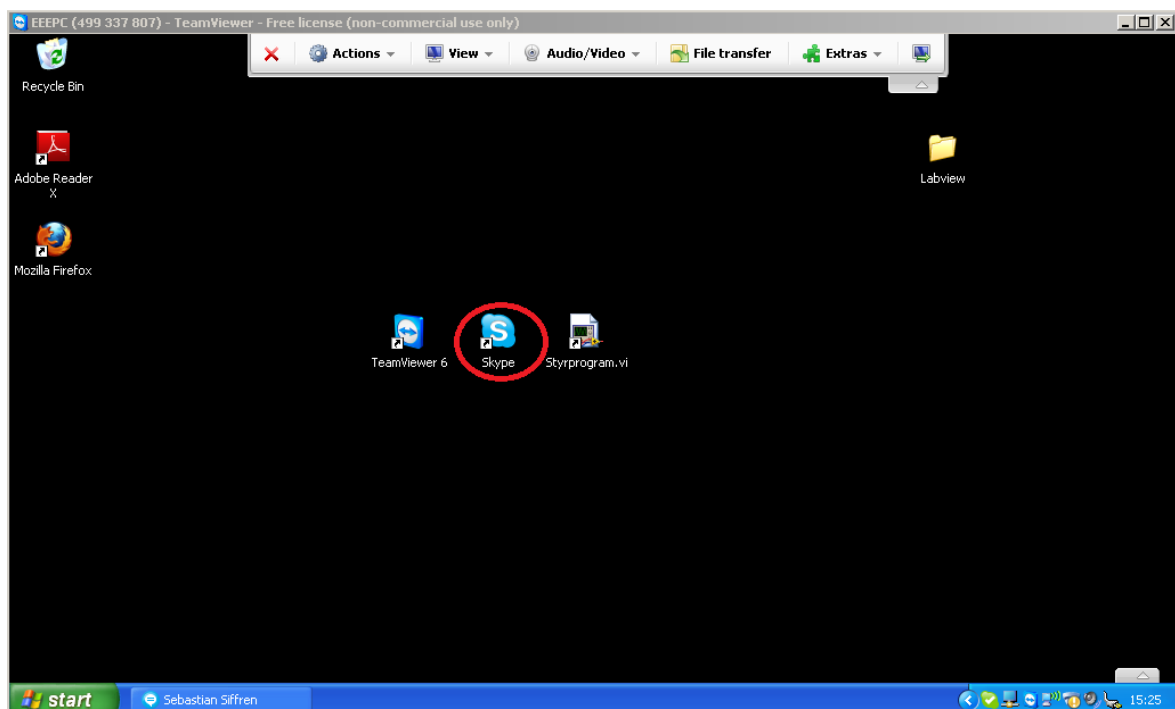


- Fyll i lösenordet
- Klicka på “Log On”

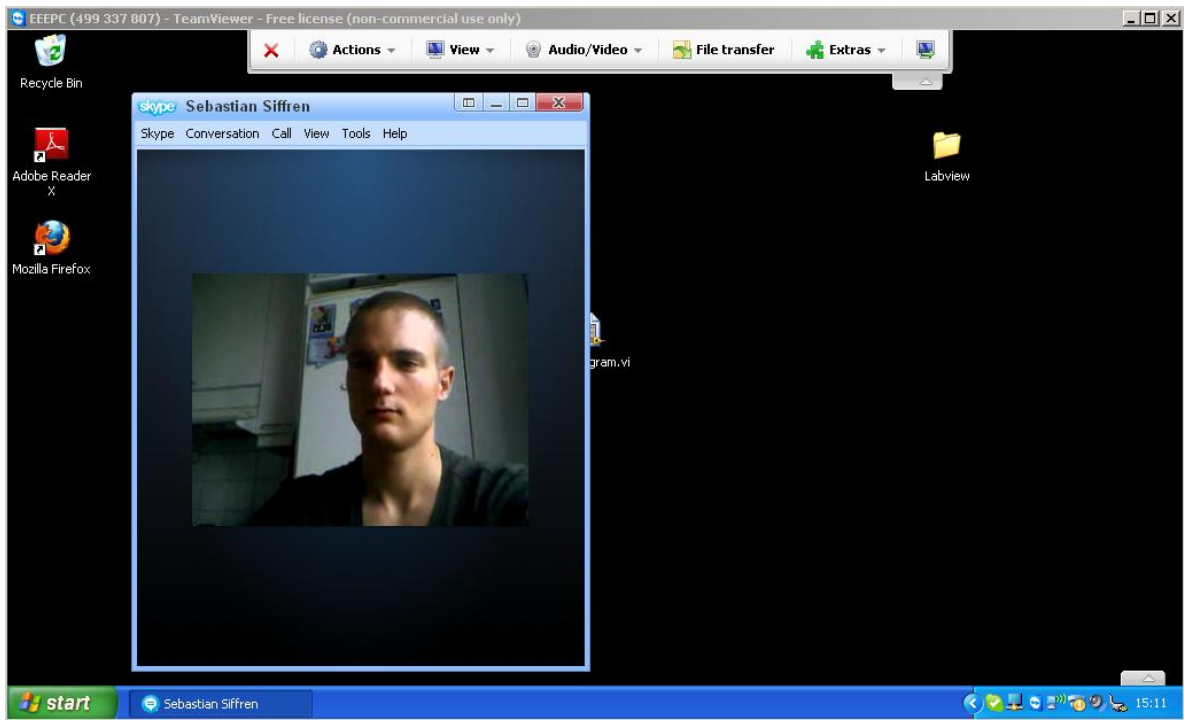


Uppstart av ljud- och bildöverföring

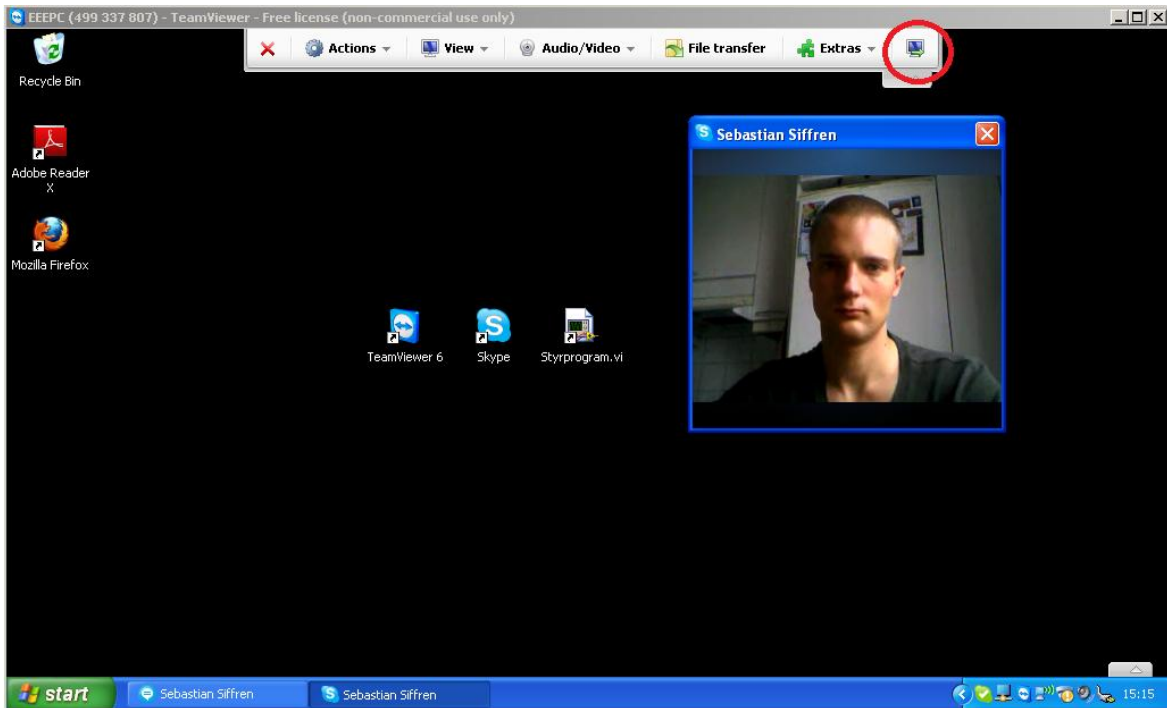
- Starta Skype på båda datorerna



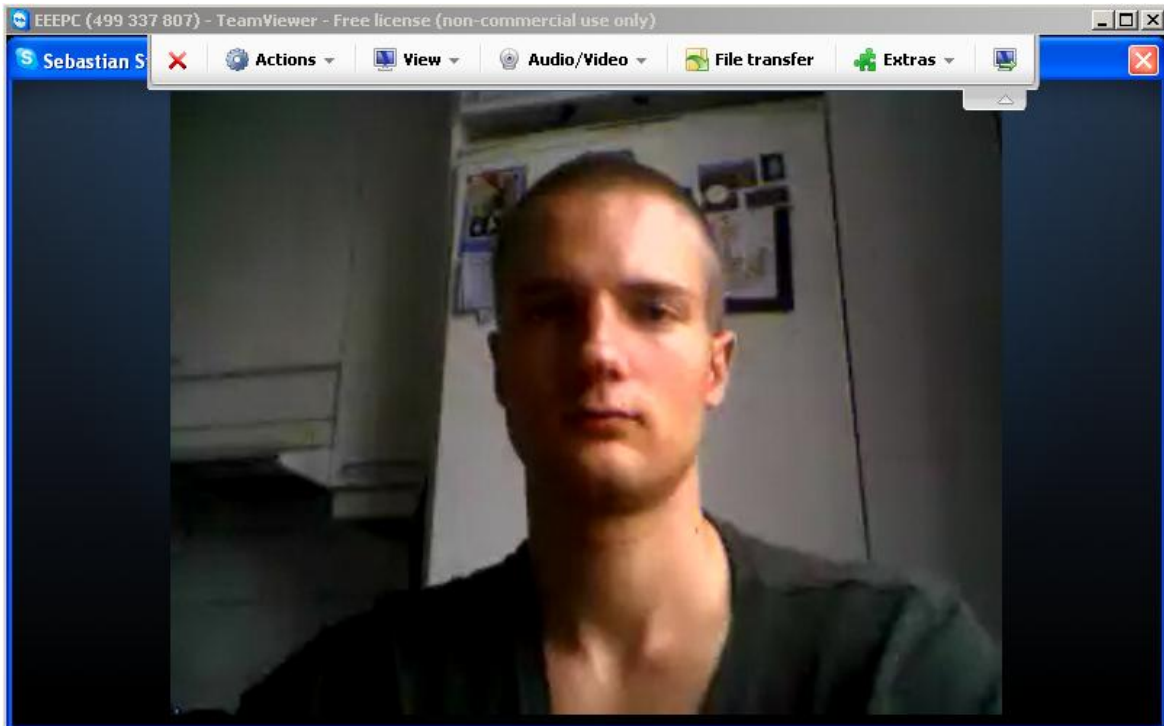
- Ring ett Skype-videosamtal till roboten från din egen dator.



- Klicka på ”pop-out” i Skype-konversationsfönstret för att få bilden att hamna i ett skilt fönster.
- Dra därefter fönstret till ”monitor-symbolen” som finns uppe i ”verktygsbalken” och släpp. (drag-and-drop)

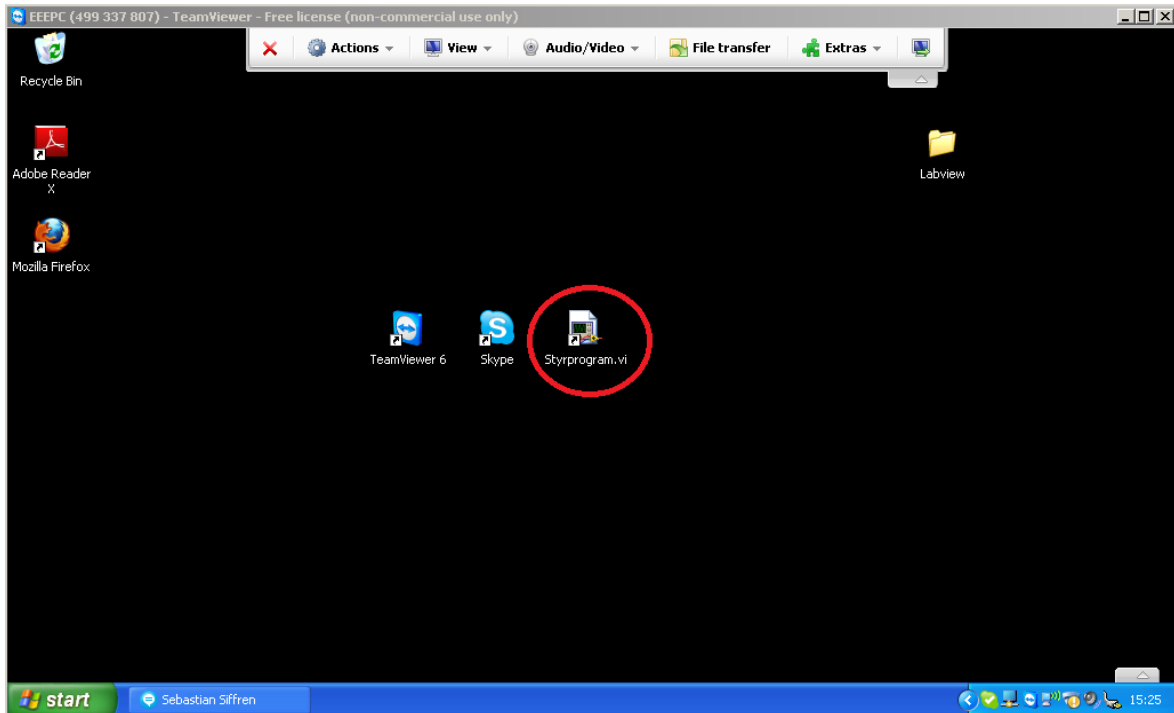


- Nu är fönstret flyttat till robotens externa monitor. Dra ut fönstret över hela skärmen för att få en större bild.
- Därefter återvänder du till ”monitor 1” genom att klicka på monitor-symbolen upp i verktygsbalken.

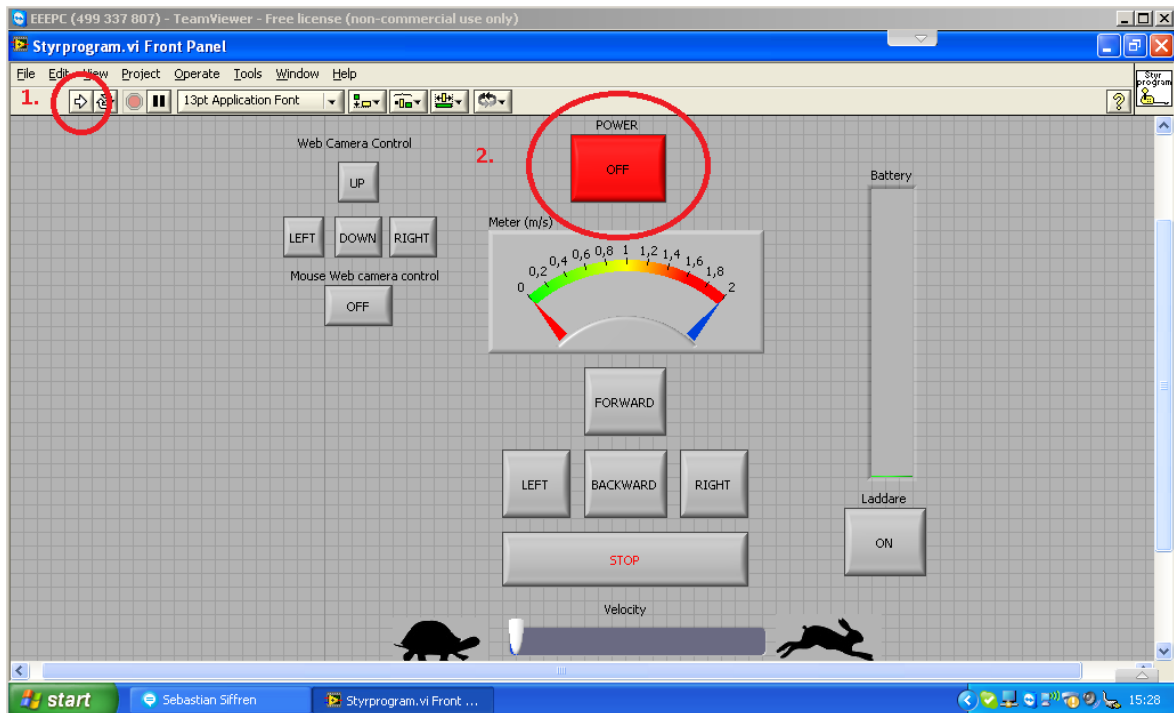


Uppstart av styrprogrammet

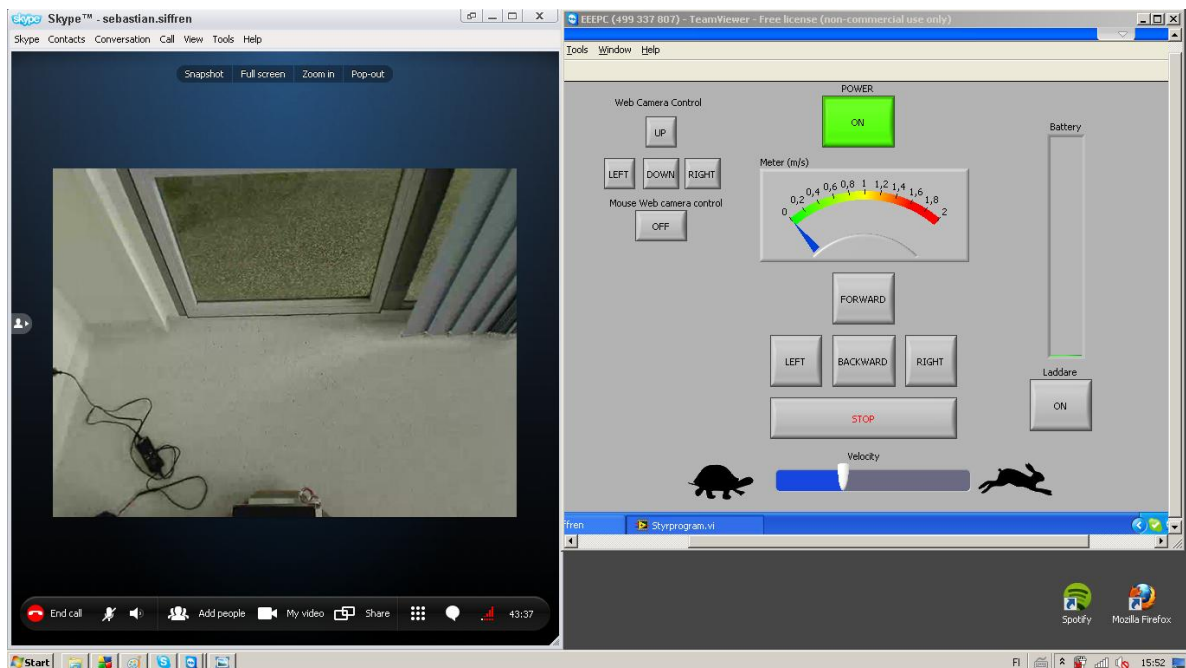
- Starta styrprogrammet genom att dubbelklicka på ”Styrprogram.vi” som finns på skrivbordet



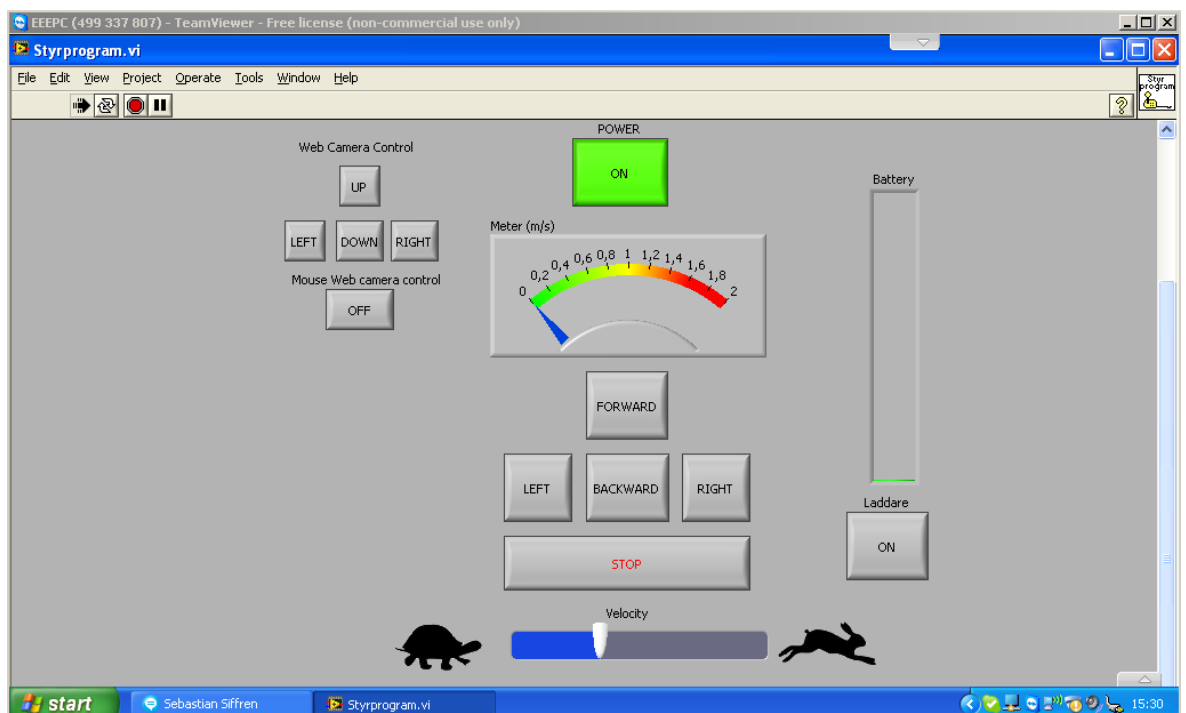
- För att starta programmet, klicka på pilen uppe i vänstra hörnet.
- För att slå på strömmen i roboten klicka på ON/OFF-knappen

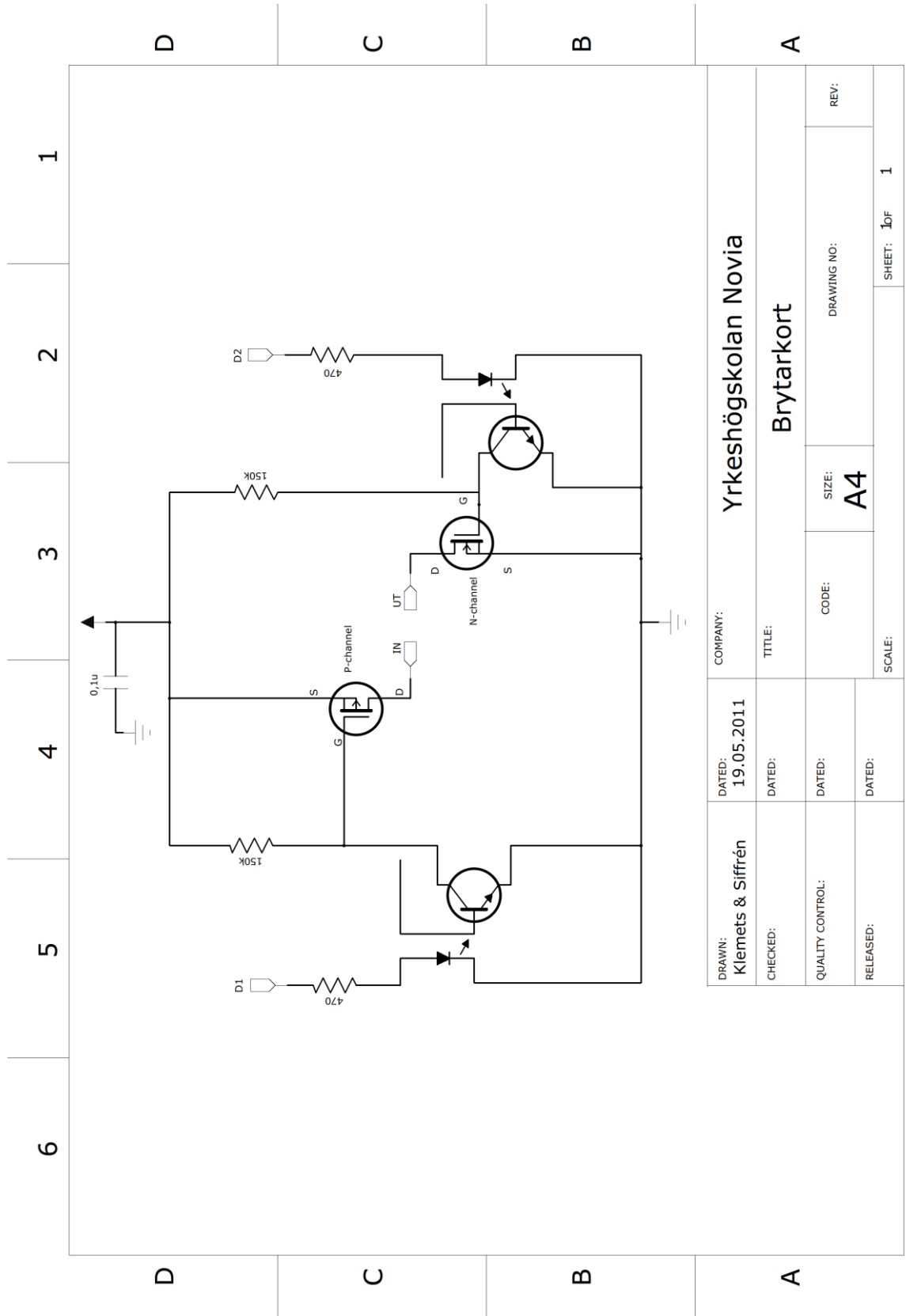


- Till sist flyttar man Skype-fönstret och LabVIEW-fönstret så att man ser lika mycket av båda två.

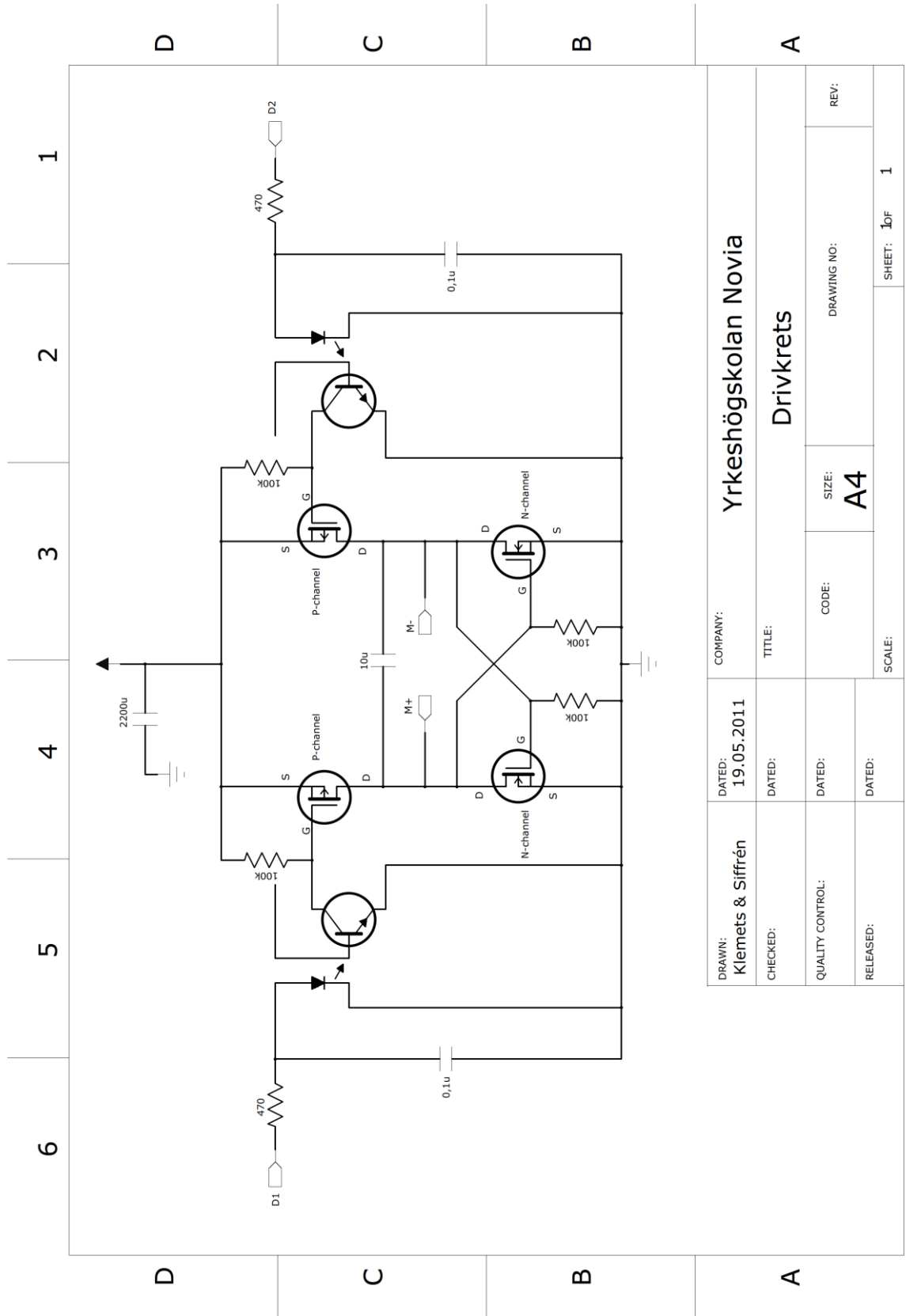


- Nu kan man börja köra. Ställ först in hur snabbt roboten ska köra på "velocity-slingan". Därefter styr man roboten och dess kamera antingen med hjälp av piltangenterna eller med hjälp av musen genom att klicka på de olika knapparna. Kameran kan även styras med musen om man aktiverar knappen "Mouse web camera control". Datorladdaren aktiveras på knappen "laddare" och återstående ström i batteriet kan ses i "Battery-figuren".

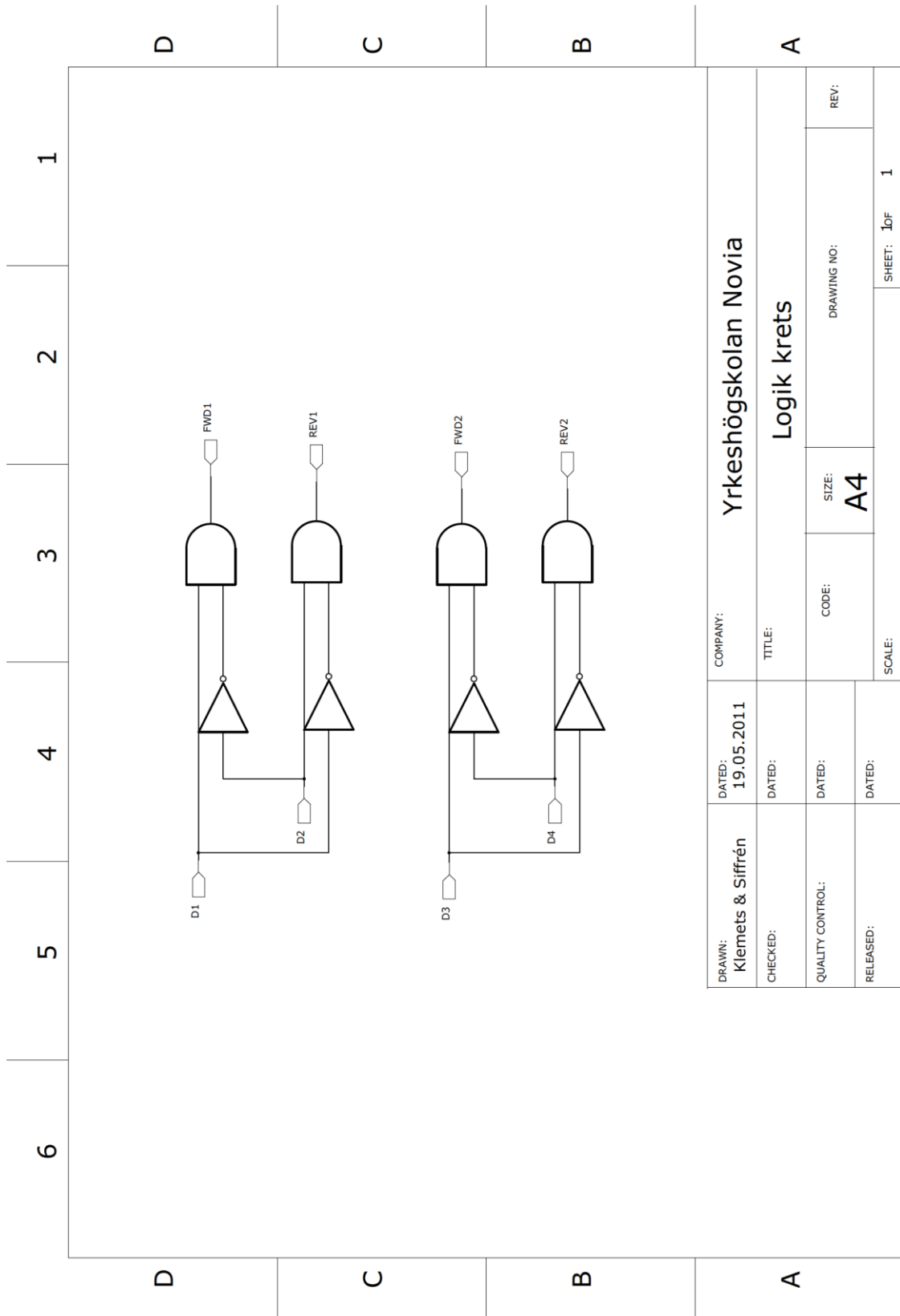




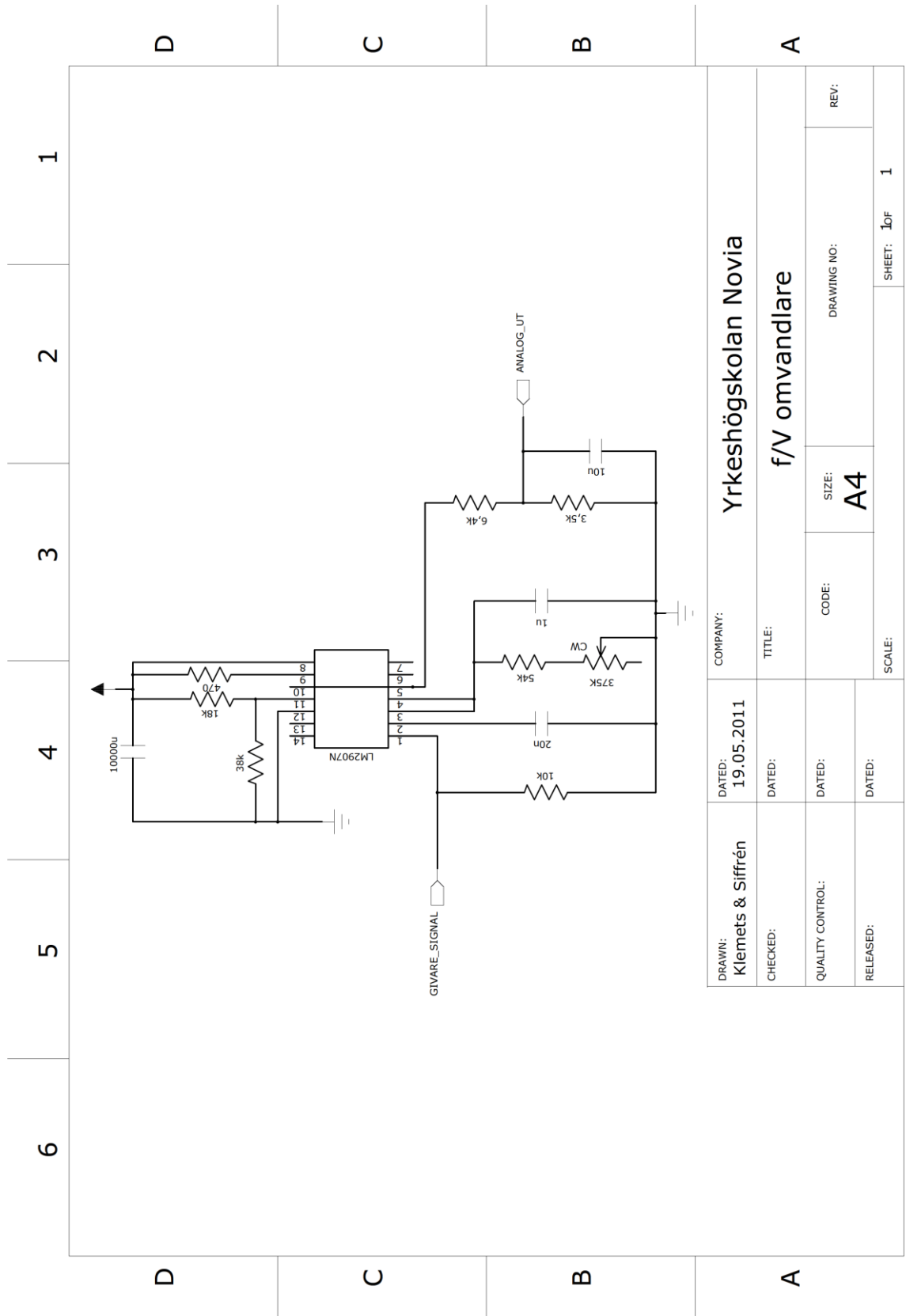
DRAWN: Klemets & Siffrén		COMPANY: Yrkes högskolan Novia	
DATED: 19.05.2011		TITLE: Brytarkort	
CHECKED:	DATED:	CODE:	DRAWING NO:
QUALITY CONTROL:	DATED:	SIZE: A4	REV:
RELEASED:	DATED:	SCALE:	SHEET: 1 of 1



DRAWN: Klemets & Siffrén		COMPANY: Yrkes högskolan Novia	
DATED: 19.05.2011		TITLE: Drivkrets	
CHECKED:	DATED:	CODE:	SIZE: A4
QUALITY CONTROL:	DATED:	DRAWING NO:	REV:
RELEASED:	DATED:	SCALE:	SHEET: 1 of 1



DRAWN: Klemets & Siffrén		DATED: 19.05.2011		COMPANY: Yrkehögskolan Novia	
CHECKED:		DATED:		TITLE: Logik krets	
QUALITY CONTROL:		DATED:		CODE:	DRAWING NO:
RELEASED:		DATED:		SIZE: A4	REV:
				SCALE:	SHEET: 1 of 1



DRAWN: Klemets & Siffrén		DATED: 19.05.2011		COMPANY: Yrkeshögskolan Novia	
CHECKED:		DATED:		TITLE: f/V omvandlare	REV:
QUALITY CONTROL:		DATED:		CODE:	DRAWING NO:
RELEASED:		DATED:		SIZE: A4	SCALE:
				SHEET: 1	of 1



