

Teknisk dokumentation för kartrobot

Patrik Sletmo

Version 1.0

Status

STATUS	Patrik Sletmo	2016-12-DD
--------	---------------	------------



Projektidentitet

Grupp 3, 16/HT, KarToffel
Linköpings tekniska högskola, ISY

Namn	Ansvar	Telefon	E-post
Patrik Sletmo	Projektledare	070 783 57 61	patsl736@student.liu.se
Rebecca Lindblom	Utvecklare	073 436 40 79	rebli156@student.liu.se
Matildha Sjöstedt	Utvecklare	070 515 84 11	matsj696@student.liu.se
Sebastian Callh	Utvecklare	073 820 46 64	sebca553@student.liu.se
Anton Dalgren	Utvecklare	076 836 51 56	antda685@student.liu.se
Matilda Dahlström	Utvecklare	070 636 33 52	matda715@student.liu.se

Hemsida: <https://github.com/SebastianCallh/kartoffel-tsea29>

Kund: Mattias Krysanter, 013 - 28 2198 , matkr@isy.liu.se

Kursansvarig: Tomas Svensson, 3B 528, +46 (0)13 28 1368,
tomas.svensson@liu.se

Handledare: Anders Nilsson, 3B 512, +46 (0)13 28 2635,
anders.p.nilsson@liu.se



Innehållsförteckning

1	Inledning	6
2	Systembeskrivning	6
2.1	Delsystem	6
2.2	Övergripande konstruktion	6
2.3	Komponenter	6
2.3.1	Beräkningsenheter	6
2.3.2	Sensorer	7
2.3.3	Ställdon	7
2.3.4	IC-kretsar	7
3	Delsystem	8
3.1	Huvudenhet	8
3.1.1	Delsystemets funktion	8
3.1.2	Kopplingsschema	9
3.1.3	Komponenter	10
3.1.4	Resurser	10
3.1.5	Programflöde	10
3.2	Sensorenhet	12
3.2.1	Delsystemets funktion	12
3.2.2	Kopplingsschema	13
3.2.3	Komponenter	14
3.2.4	Resurser	14
3.2.5	Programflöde	15
3.3	Styrenhet	15
3.3.1	Delsystemets funktion	15
3.3.2	Kopplingsschema	16
3.3.3	Komponenter	17
3.3.4	Resurser	17
3.3.5	Programflöde	18
3.4	Presentationsenhet	18
3.4.1	Delsystemets funktion	18
3.4.2	Blockschema	18
3.4.3	Komponenter	19
3.4.4	Resurser	19
3.4.5	Programflöde	19
3.5	Fjärrstyrningsenhet	19
3.5.1	Delsystemets funktion	19
3.5.2	Blockschema	19
3.5.3	Komponenter	20
3.5.4	Resurser	20
3.5.5	Programflöde	20



4	Kommunikation mellan delsystem	20
4.1	Huvudbuss	20
4.1.1	Uppbyggnad	20
4.1.2	Master-slave roller	21
4.2	Bluetooth	21
4.2.1	Handskakning	21
4.2.2	Master-slave roller	22
4.3	Informationsflöde	22
4.3.1	Kommunikation med sensorenhet	22
4.3.2	Kommunikation med mjukvaruklient	22



Dokumenthistorik

Version	Datum	Utförda ändringar	Utförd av	Granskad
VERSION	2016-12-DD	Första version	Grupp 3	Patrik Sletmo



1 Inledning

2 Systembeskrivning

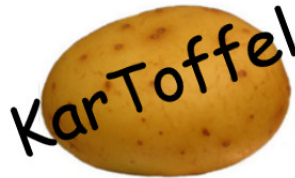
Systemet består av en fyrhjulig robot samt en mjukvaruklient som kommunicerar med varandra via bluetooth. Syftet med systemet är att roboten ska kunna kartlägga sin omgivning helt autonomt. För att göra det så har den diverse IR-sensorer, en lasersensor och ett gyroskop monterat på sig för att kunna navigera och mäta avstånd. Bluetooth-kopplingen låter mjukvaruklienten se kartläggningen i realtid, och ger möjligheten att fjärrstyra roboten.

2.1 Delsystem

Hela systemet är moduluppbyggt, och består av följande delsystem:

- Mjuvaruenhet
- Huvudenhet
- Sensorenhet
- Styrenhet

2.2 Övergripande konstruktion



Figur 1: Översikt över systemet

TODO:TEXT OM ÖVERGRIPANDE KONSTRUKTION OCH FIGUR

2.3 Komponenter

Här nedan listas alla komponenter som roboten består av.

2.3.1 Beräkningsenheter

Enhet	Antal
Raspberry Pi 3	1
ATMega 1284	2

Tabell 1: Tabell över beräkningsenheter.



2.3.2 Sensorer

Komponent	Antal
LIDAR-Lite v2	1
Knapp	1
GP2Y0A41SK IR-Sensor	3
GP2Y0A21 IR-Sensor	1
Adafruit 10-DOF IMU	1

Tabell 2: Tabell över sensorer.

2.3.3 Ställdon

Komponent	Antal
291RPM DC-motor	4

Tabell 3: Tabell över ställdon.

2.3.4 IC-kretsar

Komponent	Antal
BSS138	1

Tabell 4: Tabell över IC-kretsar.



3 Delsystem

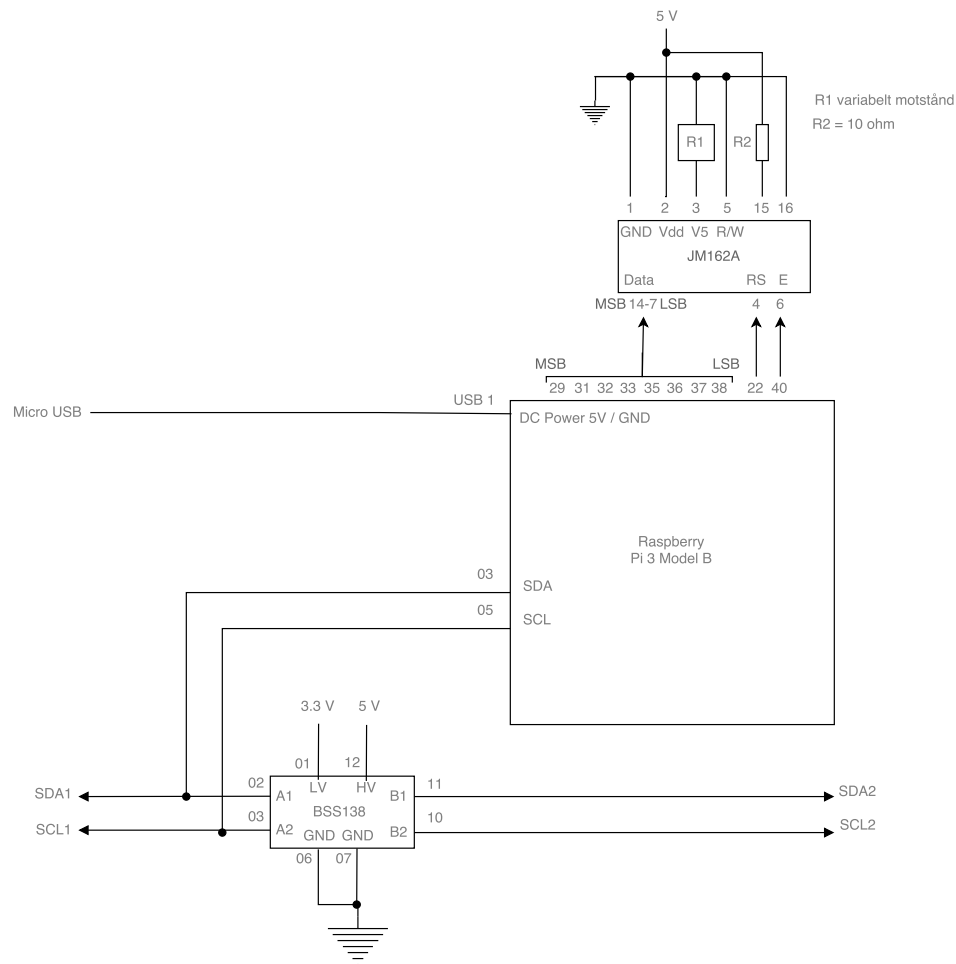
3.1 Huvudenhet

Huvudenheten består till hårdvaran av en Raspberry PI, på vilken operativsystemet Raspbian körs. Den representerar robotens beslutsfattande organ och sköter navigationsbeslut, kartläggning, kommunikation med mjukvaruklienten samt agerar mästare på robotens I2C-buss.

3.1.1 Delsystemets funktion

Huvudenheten är uppdelad i två undermoduler, logikenheten och kommunikationsenheten. Logikenheten analyserar data från sensorenheten för att avgöra hur roboten ska navigera och för att bestämma hur omvärlden ser ut så att den kan kartläggas. Roboten följer väggen på höger sida och genom att göra det kartlägger den hela den utomliggande väggen. När vi väl avgränsat rummet och området som köksön kan befinna sig i så vet roboten att de väggsegment den upptäckt som inte ingår i den utomliggande väggen är en del av köksön, och kan på så sätt navigera till dem och kartlägga därifrån. När hela kartan finns i robotens minne återvänder den till dess startposition. Logikenheten ansvarar även för att hålla koll på var någonstans i rutnätet roboten befinner sig. Kommunikationsenheten agerar som ett abstraktionslager för Raspberry Pi:ens Bluetoothmodul och ser till att sköta all översättning där i mellan så att systemet kan arbeta med interna dataformat i övriga delar.

3.1.2 Kopplingsschema



Figur 2: Kopplingsschema för huvudenheten.

Huvudenheten består av en Raspberry Pi 3 och en LCD-display av modellen JM162A. Utöver kopplingen till LCD:n så kopplas två pins till I2C-bussen. Eftersom Raspberry Pi drivs av 3.3 V matspänning och de övriga atmega-processorerna i roboten drivs av 5 V så behövs vi nivåskifta bussen så att de kan kommunicera med varandra, vilket görs med nivåskiftaren BSS138.



3.1.3 Komponenter

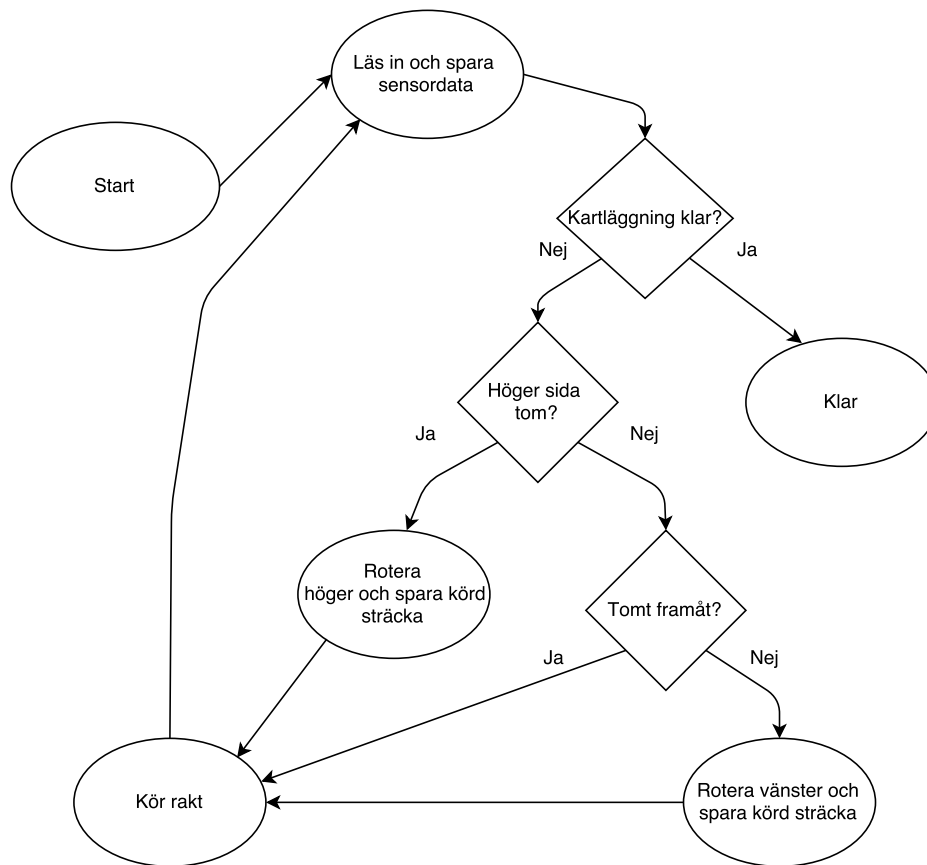
Komponent	Antal
Raspberry PI 3	1
JM162A	1

3.1.4 Resurser

Port	Antal	Används
GPIO pins	40	12

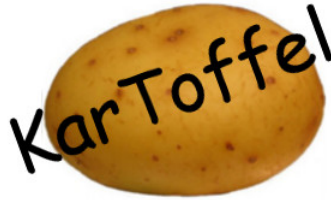
3.1.5 Programflöde

Programflödet i huvudenheten består i huvudsak av två delar: Den logiska delen som sköter navigeringsbesluten utifrån beräknad kartdata, samt delen som sköter kommunikation med resten av systemets delmoduler. Nedan beskrivs flödesscheman för navigering samt Bluetooth-kommunikation med huvudenheten. För detaljer kring kommunikation mellan huvud-, styr- och sensorenhet se avsnitt 4.



Figur 3: Flödesdiagram för navigering.

Roboten navigerar enligt flödesschemat i figur 3 till kartläggningen av rummets väggar är klara, det vill säga till det att den har cirkulerat hela rummet. Roboten vet med säkerhet att den under den här perioden kommer upptäcka väggsegment som ingår i köksön och inte i rummets väggar på sin västersida, och navigerar då till dem och påbörjar samma procedur som i flödesschemat igen varpå kartläggningen är klar.



Figur 4: Flödesschema över Bluetooth-kommunikationen i huvudenheten.

TODO:TEXT OM BLUETOOTH. UPPDATERA SÅ DEN REFLEKTERAR VÅR IMPLEMENTATION

3.2 Sensorenhet

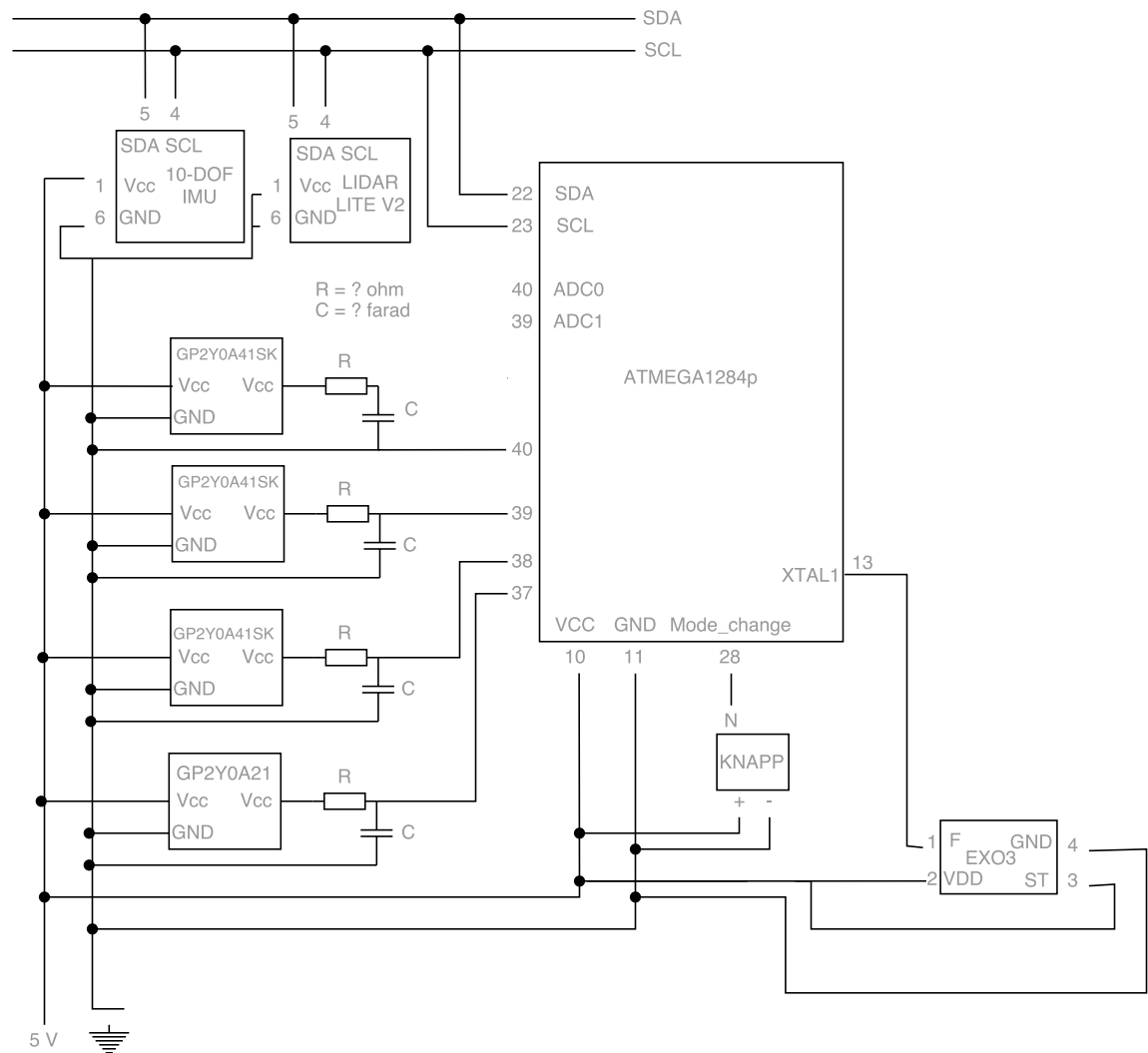
Sensorenheten har i uppgift att läsa in värden från sensorerna som finns placerade på roboten och rapportera värdena till huvudenheten. Den består av fyra IR-sensorer, ett gyroskop och en lasersensor.

3.2.1 Delsystemets funktion

Sensorenheten har sin egna beräkningsenhet, ATmega 1284, vilket låter den arbeta asynkront från våra andra enheter. Den ser på så sätt alltid till att ha data tillgänglig närhelst huvudenheten begär den. För IR-sensorerna sker det genom att processorn pollar information från varje sensor och sparar undan i minnet, medan gyroskop och laser håller koll på sina egna värden och är kopplade direkt på I2C-bussen. Huvudenheten kan sedan fråga sensorenheten efter ny IR-mätdata, eller fråga gyroskopet/lasern direkt varpå den får sensordatan levererad över I2C.

Avläsningarna från IR-sensorerna är analoga, och konverteras till digitalt med hjälp av ATMegans interna AD-omvandlare. För att sedan omvandla mätdata till en approximation i millimeter används en tabell sparad i ATMegans minne.

3.2.2 Kopplingsschema



Figur 5: Kopplingsschema för sensorenheten.

Som ses i figur 5 är IR-sensorerna kopplade till ATMegar via var sitt lågpassfilter för att reducera brus. Längst upp i kopplingsschemat ses den gemensamma I2C-bussen, dit gyroskop och laser är anslutna. Notera att EXO3-komponenten är delad mellan sensorenheten och styrenheten trots att den illustreras som separata komponenter i kopplingsschemat.



3.2.3 Komponenter

Komponent	Antal
ATMega 1284	1
LIDAR-Lite v2	1
Knapp	1
GP2Y0A41SK IR-Sensor	3
GP2Y0A21 IR-Sensor	1
Adafruit 10-DOF IMU	1
EXO3	1 (delad)

Tabell 5: Tabell över de komponenter som sensorenheten består av.

3.2.4 Resurser

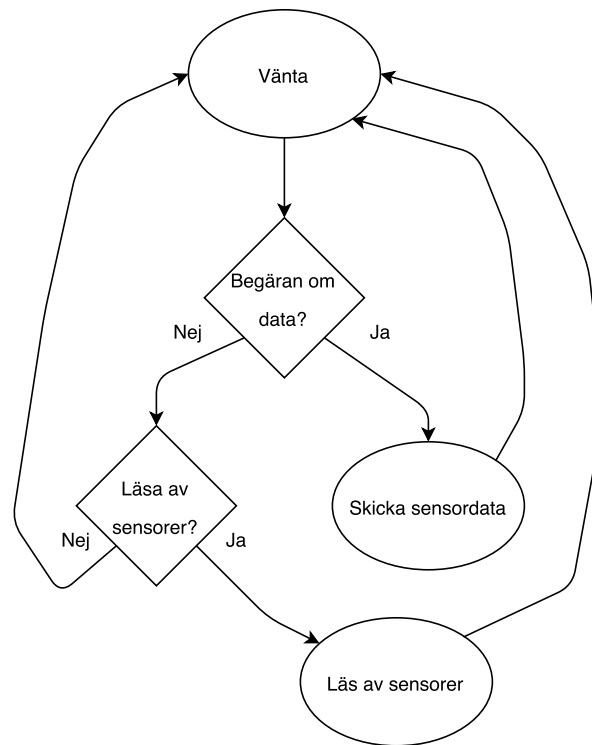
TODO:TABELLEN NEDAN EJ KORRIGERAD

Port	Antal	Krävs
SDA	1	1
SCL	1	1
PCINT	24	1
A/D	8	2
RESET	1	1
USART	4	2
JTAG	1	1
CLK	1	1

Tabell 6: Tabell över tillgängliga portar på processorn.

Den maximala hastigheten som sensordata kan läsas in i är begränsad av sensorernas rapporteringsfrekvens som är betydligt långsammare än en ATMega 1284:s klockfrekvens, därför lönar det sig inte med en snabbare processor för sensorenhetens syfte. Det i princip enda minnet sensorenheten använder sig av är för temporärlagring av alla IR-sensors data, vilket endast motsvarar fyra heltal. Utöver sensorinläsningen kommer enheten svara på kommandon över huvudbussen som inte heller kräver några höga hastigheter eller minneskrav.

3.2.5 Programflöde



Figur 6: Ett flödesdiagram över sensorenhetens tillstånd

För att kunna tillhandahålla sensordata från IR-sensorerna när huvudenheten ber den om det så läser sensorenheten in data med jämna mellanrum och sparar den lokalt för att sedan skicka datan på kommando från huvudenheten. Om den fått en begäran om sensordata från huvudenheten så tar det prioritet att leverera den över att läsa in ny data. Ett flödesdiagram över dess beteende ses i 6.

3.3 Styrenhet

Styrenhetens uppgift är att kontrollera alla robotens fysiska rörelser med hjälp av servon, baserat på kommandon ifrån huvudenheten. Det är genom styrenhetens funktionalitet som roboten förflyttar sig igenom rummet som ska kartläggas.

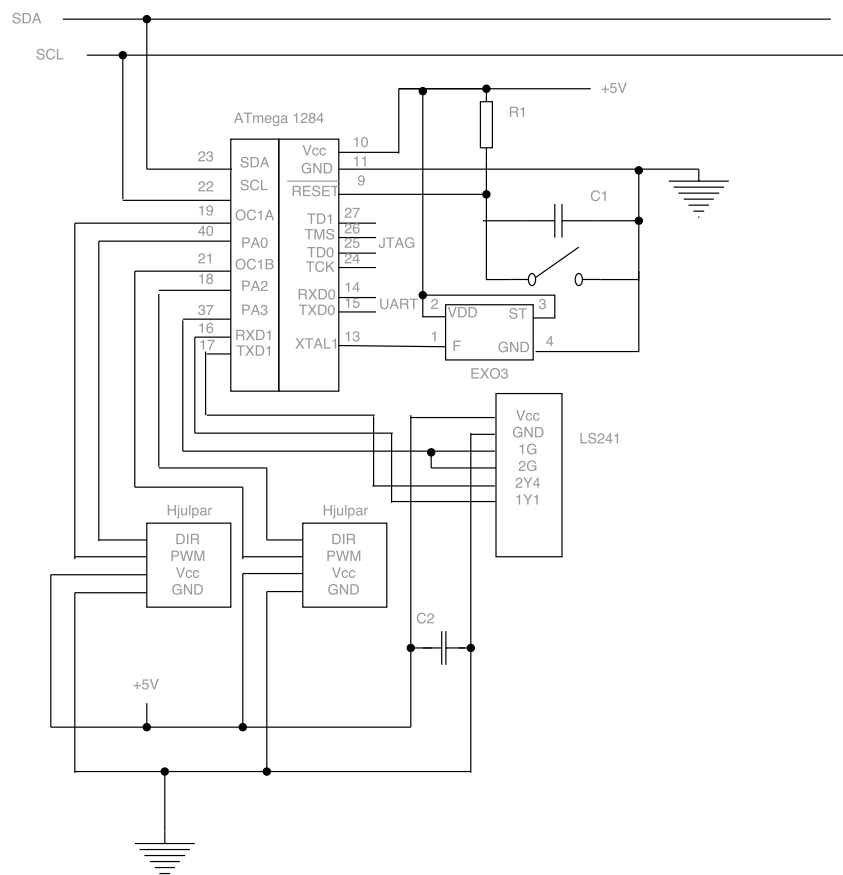
3.3.1 Delsystemets funktion

Styrenheten består av en ATmega 1284 processor som har robotens alla servon kopplade till sig. Den tar emot kommandon från huvudenheten över I2C-bussen

som säger åt roboten att exempelvis åka framåt eller rotera. Som svar skickar styrenheten ut korrekta signaler till relevanta servon för att kontrollera robotens rörelser enligt instruktioner.

För att styra robotens hjul används PWM, där styrenheten är förprogrammerad med lämplig pulskvot.

3.3.2 Kopplingsschema



Figur 7: Kopplingsschema för styrenheten.

TODO:TEXT OM KOPPLINGSSCHEMA



3.3.3 Komponenter

Komponent	Antal
ATMega 1284	1
Terminator (bas för fyrhjulingsrobot)	1
EXO3	1 (delad)
LS241	1

3.3.4 Resurser

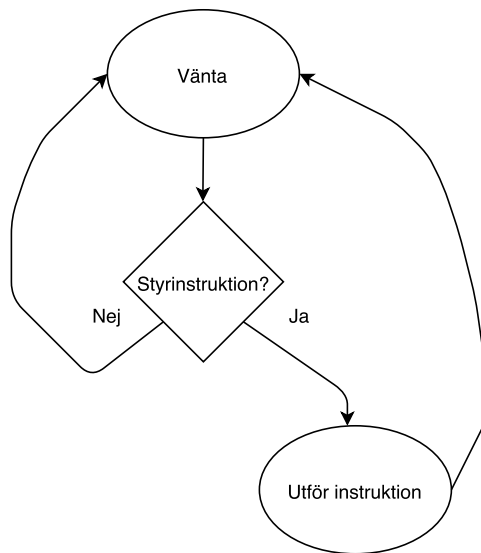
TODO:TABELL NEDAN EJ KORRIGERAD

Port	Antal	Krävs
SDA	1	1
SCL	1	1
RESET	1	1
IO	30	5
USART	4	4
JTAG	1	1
CLK	1	1

Tabell 7: Tabell över tillgängliga portar på processorn.

Omvandlingen av kommandon från huvudenheten och utskickning av signaler till servona uppskattas ej kräva mer prestanda eller minne än styrenhetens ATMega 1284 processor klarar av.

3.3.5 Programflöde



Figur 8: Ett flödesschema över styrenhetens tillstånd

Styrenheten står hela tiden och väntar på instruktioner från huvudenheten, och så snart den får några utför den dem så snabbt den kan. ett flödesschema över styrenhetens beteende kan ses i figur 8.

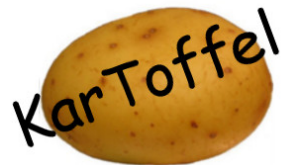
3.4 Presentationsenhet

TODO:UPPDATERA TEMPUS

3.4.1 Delsystemets funktion

TODO:UPPDATERA TEMPUS

3.4.2 Blockschema



Figur 9: Presentationsenheten i omgivning



TODO:TEXT OM BLOCKSCHEMA SOM REFLEKTERAR IMPLEMENTATION

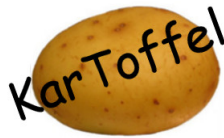
3.4.3 Komponenter

Mjukvaruklienten och därmed presentationsenheten kräver en bärbar dator med Bluetooth.

3.4.4 Resurser

TODO:VI ANVÄNDER INGEN TILLHANDAHÅLLEN BÄRBAR DATOR. KBRY? Presentationen och behandlingen av datan som presentationsenhet mot-tager uppskattas ej kräva mer minne eller prestanda än vad den tillhandahållna bärbara datorn klarar av.

3.4.5 Programflöde



Figur 10: Flödesschema över presentationsenheten

TODO:TEXT OM PROGRAMFLÖDE SOM REFLEKTERAR IMPLEMENTATION

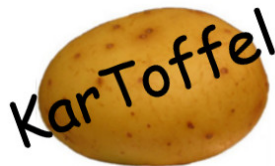
3.5 Fjärrstyrningsenhet

TODO:TEXT SOM SPEGLAR IMPLEMENTATION

3.5.1 Delsystemets funktion

TODO:TEXT SOM SPEGLAR IMPLEMENTATION

3.5.2 Blockschema



Figur 11: Fjärrstyrningsenheten i omgivning



TODO:TEXT OM BLOCKSCHEMA SOM REFLEKTERAR IMPLEMENTATION

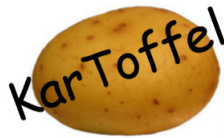
3.5.3 Komponenter

Fjärrstyrningsenheten kräver en dator med Bluetooth som kan köra mjukvaruklienten, samt tangentbord och mus/touchpad för interaktion med klienten.

3.5.4 Resurser

TODO:VI ANVÄNDER INGEN TILLHANDAHÅLLEN BÄRBAR DATOR. KBRY? Funktionaliteten för att ta emot användarinput och skicka iväg den via Bluetooth uppskattas ej kräva mer minne eller prestanda än vad den tillhandahållna datorn klarar av.

3.5.5 Programflöde



Figur 12: Flödesschema över fjärrstyrningssenheten

TODO:TEXT OM PROGRAMFLÖDE SOM REFLEKTERAR IMPLEMENTATION

4 Kommunikation mellan delsystem

TODO:UPPDATERA TEMPUS, KANSKE IMPLEMENTATION?

4.1 Huvudbuss

TODO: KORREKTURLÄS SÅ DET FORTFARANDE STÄMMER Systemets moduler på roboten kommer vara sammankopplade med en I2C-buss (även kallad Two-Wire-Interface) där huvudenheten skickar och tar emot kommandon från sensor- och styrenheten. Huvudbussen kommer även att användas för att skicka sensordata från lasersensorn samt den kombinerade accelerometern och gyrot till sensorenhetens processor.

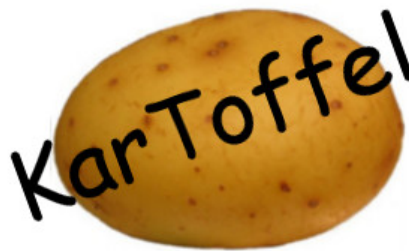
4.1.1 Uppbyggnad

TODO: KORREKTURLÄS SÅ DET FORTFARANDE STÄMMER Bussen består så som dess alternativa namn antyder av ett två-trådat gränssnitt och stödjer upp till 128 enheter. De två trådarna är SDA och SCL, där SDA överför



data medan SCL styr med vilken frekvens detta sker. SDA kommer skrivas till utav bussens alla inkopplade enheter och behöver kopplas in till ett pull-up-motstånd för att inte flyta. Uppbyggnaden av bussen och dess master-slave förhållande kan ses i figur 13.

4.1.2 Master-slave roller



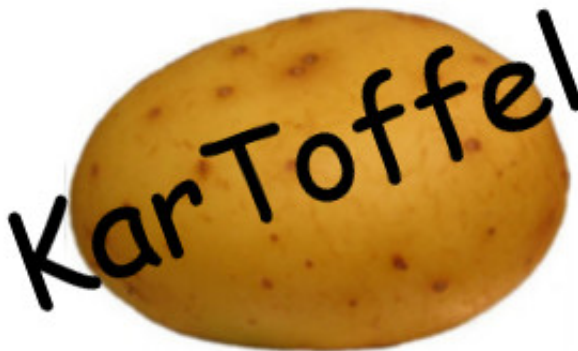
Figur 13: Huvudbussens uppbyggnad och master-slave förhållande

TODO:UPPDATERA TEMPUS

4.2 Bluetooth

TODO:UPPDATERA TEMPUS

4.2.1 Handskakning



Figur 14: Bluetooth mellan huvudenheten och mjukvaruklienten

TODO:TEXT OM HANDSKAKNING SOM REFLEKTERAR IMPLEMENTATION



4.2.2 Master-slave roller

TODO:UPDATERA TEMPUS, KANSKE IMPLEMENTATION?

4.3 Informationsflöde

TODO:UPPDATERA TEMPUS

4.3.1 Kommunikation med sensorenhet

TODO:UPPDATERA TEMPUS

4.3.2 Kommunikation med mjukvaruklient

TODO:UPPDATERA TEMPUS

TODO:SE TILL ATT NEDANSTÅENDE LISTA SPEGLAR IMPLEMENTATION Den data som skickas från mjukvaruklienten är följande

- Förfrågan om all ny kartdata sedan en viss tidpunkt t
- Kommandon för att styra roboten
- Byte av exekveringsläge

TODO:SE TILL ATT NEDANSTÅENDE LISTA SPEGLAR IMPLEMENTATION Den data som skickas till mjukvaruklienten är följande

- All ny kartdata sedan tidpunkt t
- Kontinuerlig kartdata
- Kontinuerlig mätdata
- Kontinuerlig styrdata
- Verifikation av kommandon

TODO:TEXT OM OVANSTÅENDE LISTOR

TODO:BYT UT ALLA BILDER FRÅN PLACEHOLDERN LOGO OCH SE ÖVER ATT DE FORTFARANDE ÄR AKTUELLA TODO:LCDn SKA IN NÅGONSTANS