



# Designspecifikation

## Version 1.1

### Status

Granskad	Sandy Klaff	2016-03-29
Godkänd	Olov Andersson	2016-03-29



## PROJEKTIDENTITET

2016/VT, Grupp 1

Linköpings Tekniska Högskola, ISY

### Gruppdeltagare

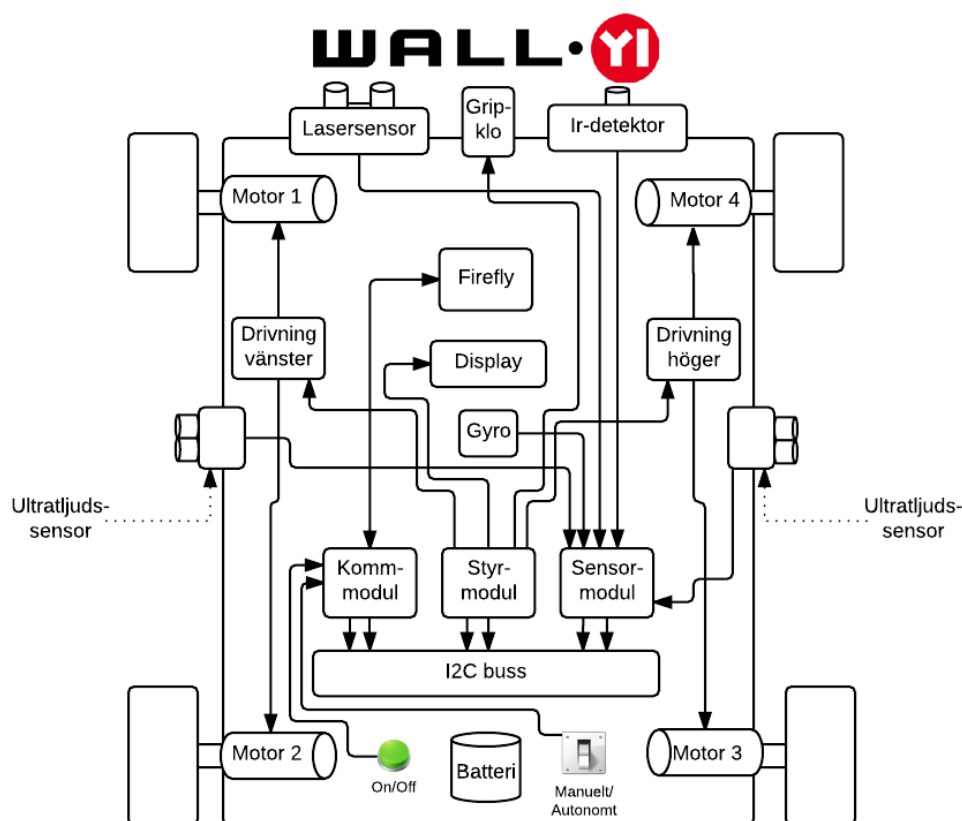
Namn	Ansvar	Telefon	E-post*
Albin Detterfelt	Kommunikationmodul och Blåtand	070-655 25 99	albde282
Klas Gudmundsson	Persondatoransvarig	072-714 06 66	klagu863
Martin Haugsbakk	Buss- och integrationsansvarig	070-270 63 36	marha996
Milton Johansson	Sensormodulsansvarig	073-037 01 64	miljo274
Sandy Klaff	Dokument- och mjukvaruansvarig	070-293 77 49	sankl660
Fredrik Olsson	Projektledare och testansvarig	072-315 38 68	freol454
Martin Prage	Styrmodulsansvarig	073-694 95 24	marpr146

\*alla e-postadresser slutar med @student.liu.se

**Kund:** ISY, Linköpings universitet, 581 83 Linköping**Kontaktperson hos kund:** Mattias Krysander, 013-282198, matkr@isy.liu.se**Kursansvarig:** Tomas Svensson, 013-281368, tomas.svensson@liu.se**Handledare:** Olov Andersson, 013-282658, olle.andersson@liu.se

## Sammanfattning

Systemet består av tre moduler; en sensormodul, en styrmodul samt en kommunikationsmodul (huvudenhet). Dessa tre moduler presenteras närmre nedan och beskrivs visuellt i figur 1. De sensorerna som ska användas är två ultraljudssensor, en lasersensor, en gyro och en IR-detektor. Sensorerna ska placeras enligt figur 1. Roboten använder fyra motorer kopplad till vardera hjul och respektive hjulpar styrs separat för att möjliggöra svängar.



Figur 1: Blockschemat över systemet

## Mikrodator Atmel ATmega16

Samtliga moduler styrs huvudsakligen av var sin mikrodator av typen Atmel ATmega16, en enchipsdator vilket innebär att CPU, RAM och ROM är på samma chip. Den består av fyra 8-bitars inportar med olika funktioner för olika pinnar



såsom parallell och seriell koppling, A/D-omvandling, timer/counter, PWM, I2C, SPI med mera.



# Innehåll

## Dokumenthistorik

<b>1</b>	<b>Inledning</b>	<b>1</b>
<b>2</b>	<b>Styrmodul</b>	<b>1</b>
2.1	Översikt av delsystem . . . . .	1
2.2	Framdrivning . . . . .	2
2.3	Motorkontroll . . . . .	3
2.4	Display . . . . .	3
2.5	Gripklo . . . . .	3
2.6	Hårdvara . . . . .	4
<b>3</b>	<b>Kommunikationsmodul</b>	<b>5</b>
3.1	Översikt av delsystem . . . . .	5
3.2	AVR-dator . . . . .	5
3.3	Buss . . . . .	5
3.4	Blåtandsmodul . . . . .	6
3.5	Labyrintkarta . . . . .	6
3.6	Kartläggningsalgoritmer . . . . .	7
3.6.1	Algoritmalternativ 1 . . . . .	7
3.6.2	Algoritmalternativ 2 . . . . .	8
3.6.3	Algoritmalternativ 3 . . . . .	8
3.7	Kortaste-vägen-algoritmen . . . . .	8
3.8	Manuell styrning . . . . .	8
<b>4</b>	<b>Sensormodul</b>	<b>9</b>
4.1	Översikt av delsystemet . . . . .	9
4.2	AVR-dator . . . . .	9
4.3	Gränssnitt . . . . .	9
4.3.1	Sensorer – AVR-datorn . . . . .	9
4.3.2	Sensormodulen – Huvudenheten . . . . .	12
4.4	Ultraljudssensor . . . . .	12
4.5	Lasersensor . . . . .	12
4.6	IR-detektor . . . . .	13
4.7	Gyro . . . . .	13
<b>5</b>	<b>Persondator</b>	<b>13</b>
5.1	Styrkommandon . . . . .	14
5.2	Grafisk modellering . . . . .	14



---

<b>6</b>	<b>Kommunikation mellan delsystem</b>	<b>14</b>
6.1	Kommunikation mellan processorerna . . . . .	14
6.2	Trådlös kommunikation mellan robot och persondator . . . . .	15
<b>7</b>	<b>Implementeringsstrategi</b>	<b>16</b>
7.1	Konstruktion . . . . .	16
7.2	Återkoppling från systemet . . . . .	16
7.3	Testning . . . . .	16
<b>A</b>	<b>Appendix</b>	<b>18</b>
A.1	Kopplingscheman för delsystemen . . . . .	18
	<b>Referenser</b>	<b>22</b>



Dokumenthistorik

Version	Datum	Utförda förändringar	Utförda av	Granskad
0.1	2016-03-11	Första utkast	Alla	FO
1.0	2016-03-29	Första version	SK,KG	SK
1.1	2016-03-29	Ändringar efter kommentarer från handledare.	SK	SK

# 1 Inledning

Projektets syfte är att utveckla och konstruera en undsättningsrobot. Hela systemet består i huvudsak av fyra delsystem som tillsammans ska kunna autonomt genomsöka, kartlägga miljön och leverera ett objekt till en funnen position på bästa möjliga sätt. De fyra delsystemen har olika ansvarsområden och består av kommunikationsmodul, styrmodul, sensormodul och persondator.

Kommunikationsmodulen är den centrala modulen i systemet och är där mest mjukvara kommer att skrivas och där de flesta beräkningar kommer att göras. Den kommer även att sköta integreringen mellan alla de andra modulerna och hålla reda på vad som händer i hela systemet.

Styrmodulen har som huvudsak att driva roboten framåt och reglera robotens rörelse och har även ansvar för gripklons drivning samt vad displayen visar.

Sensormodulen har som ansvar att ha koll på miljön runt omkring, hur långt har roboten färdats, hur långt är det till kringliggande väggar och finns den nödställda någonstans i närheten.

Sista delsystemet är persondatorn som skiljer sig lite då den inte behöver konstrueras eller sättas ihop, dock måste en trådlös kommunikation upprättas mellan persondatorn och kommunikationsmodulen för att kunna skicka komandon från persondatorn till systemet och kunna få information om vad som händer i systemet.

Hur delsystemen opererar och vilken hårdvara som ingår i respektive system beskrivs i detalj i avsnitt 2-5.

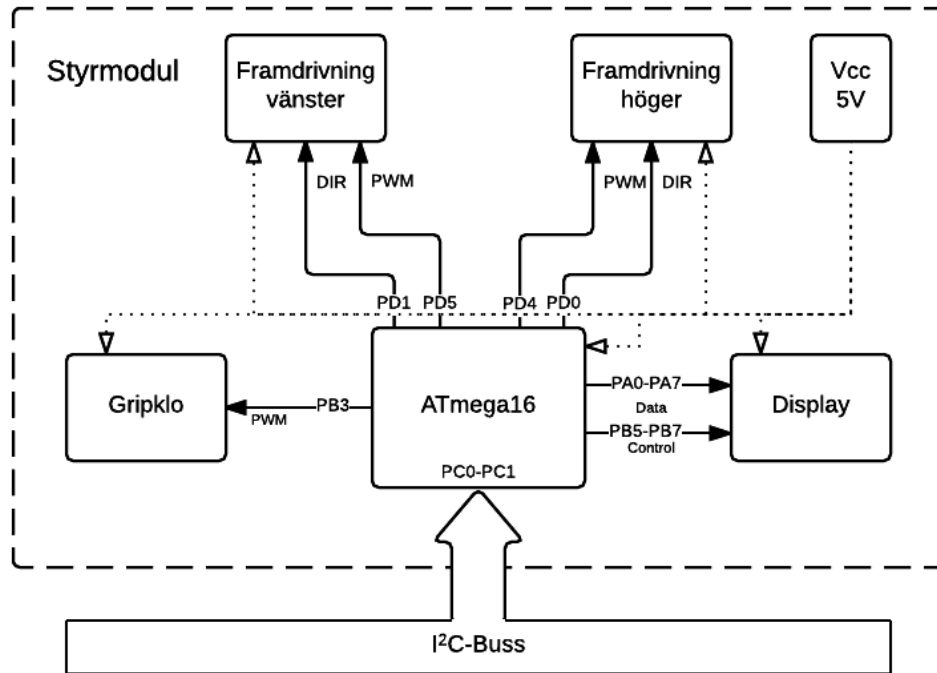
## 2 Styrmodul

Styrmodulen är den modul som utför framdrivningen av roboten genom motorer fram och bak, griper tag i nödproviant genom en gripklo samt presenterar data från sensorerna via en display. Ett blockschema över styrmodulen visas i Figur 2 nedan.

### 2.1 Översikt av delsystem

Styrmodulen består av en styrenhet, en gripklo och en display. I styrenheten ingår mikroprocessorn ATmega16 samt motorer till höger och vänster hjulpar för fram-



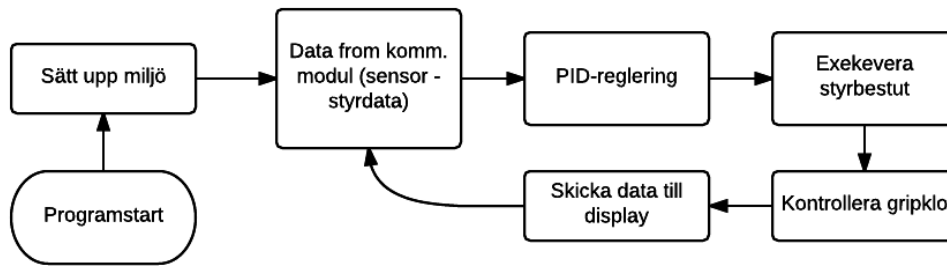


Figur 2: Blockschemat över styrmodulen, dess komponenter och signalvägar

drivning av roboten. Styrmodulen kommer endast att ta emot data från de övriga modulerna. Ett fullständigt kopplingschema över delsystemet återges i Appendix A.1. Figur 3 är ett flödesschema för styrmodulen.

## 2.2 Framdrivning

Microprocessorn tar emot navigationsbeslut från kommunikationsenheten. När enheten får kommandot kommer enheten ge signaler om rotationsriktning samt hastighet till motorenheterna. Vid körning kommer microprocessorn att använda data från sensorenheten och med hjälp av PID-reglering köra rakt så länge inte ett nytt styrbeslut måste tas. PID-regleringen sker alltså i styrmodulen, med hjälp av utvecklad mjukvara som tar in sensorvärden från sensorenheten. Då ett beslut behövs kommer styrenheten invänta kommando från kommunikationsmodulen som anger vad styrenheten ska göra. Energikällan för motorerna kommer att vara batterier på 7.2 V. De styrkommandon som kommer innebära en svängrörelse för roboten ska implementeras på så sätt att det skickas olika PWM-signaler med olika duty-cycle till hjulparen. Rotationsrörelsen kommer att implementeras genom



Figur 3: Flödesschema över styrmodulen.

att hjulparen blir signalerade olika riktningar men med samma fart. Exakt vilken duty-cycle som behövs för att åstadkomma dessa rörelser kommer att testas fram.

## 2.3 Motorkontroll

Motorerna kontrolleras genom att skicka två TTL/CMOS-kompatibla insignaler till respektive hjulpar från styrenhetens mikroprocessor. Den första signalen indikerar vilken rotationsriktning motorerna ska ha. Den andra signalen, en PWM signal, indikerar vilken hastighet motorerna ska ha. På ATmega16 ska register OCR1A/B och TCNT1 (vars värde bestäms av intern 16-bit räknare) användas för att åstadkomma en PWM-signal till respektive hjulpar, genom output på OC1A/B. TTL-signalen för att bestämma vilken riktning som respektive hjulpar ska rotera skickas från datapins PD0/1.

## 2.4 Display

Displayen ska vara av typen JM162A, en tvåraders alfanumerisk display. Displayen ska visa avståndet till högra, vänstra och främre vägg med information från sensorerna. Information om sensorvärdena skickas från microprocessorn till displayen genom åtta datapins, och kontroll av displayen sker genom ytterligare två. För att skriva värden till displayen kommer datapins PA0-PA7 (port A) användas, eftersom ingen A/D-omvandling behövs. Genom PB5, PB6 och PB7 kontrolleras displayens input Read/Write, Register Select och Enable.

## 2.5 Gripklo

Roboten kommer även att innehålla en gripklo som ska kunna gripa tag, hålla och släppa ett mindre objekt som ska framföras till de nödställda. Det är styrmodulen som ger besked om klon ska öppnas eller stängas. Gripklon styrs av ett

PIN	Funktion	signalnamn
4	OUT	PWM_GRIPKLO
6	OUT	DISPLAY_CONTROL_0
7	OUT	DISPLAY_CONTROL_1
8	OUT	DISPLAY_CONTROL_2
9	RESET	RST
10	VCC	VCC (+5V)
11	GND	GND
14	OUT	DIR_RIGHT
15	OUT	DIR_LEFT
18	OUT	PWM_DRIV_RIGHT
19	OUT	PWM_DRIV_LEFT
22	SCL	BUS_CLK
23	SDA	BUS_DATA
30	VCC	AVCC (+5V)
31	GND	GND
33	OUT	DISP_BIT_7
34	OUT	DISP_BIT_6
35	OUT	DISP_BIT_5
36	OUT	DISP_BIT_4
37	OUT	DISP_BIT_3
38	OUT	DISP_BIT_2
39	OUT	DISP_BIT_1

Tabell 1: PIN-konfiguration för styrmodulens AVR

servo som kontrolleras genom en PWM-signal på microcontrollern. Här används register OCR0 och TCNT0 (vars värde bestäms av intern 8-bit räknare) för att åstadkomma en PWM-signal till servot, genom output på PB3 (OC0).

## 2.6 Hårdvara

Det som behövs är följande: en AVR ATmega16, ett robotchassi med fyra motorer samt två drivkretsar, en gripklo samt servo till klon, en JM162A display, strömförsörjning, flertalet kablar för signalerna och ett virkort för montering. Portarna som används är färre än vad som finns på ATmega16 processorn och räcker därför till. Processorn anses ha den kapacitet i form av prestanda och minne som krävs för att lösa de uppgifter och krav som är ålagda styrmodulen.

PIN-konfigurationen för processorn återges i Tabell 1.

### 3 Kommunikationsmodul

Kommunikationsmodulen kommer att sköta intern och extern kommunikation. Modulen kommer att ta emot data från sensormodulen och styrmodulen samt kunna ta emot data från persondatorn via Blåtand för att kunna styras manuellt. Modulen ska också skicka bland annat sensordata/"kartdata" till persondatorn, detta för att persondatorn ska rita upp en karta och följa var roboten befinner sig och har varit. Denna data kommer sedan ligga till grund för de beslut kommunikationsenheten tar.

#### 3.1 Översikt av delsystem

Kommunikationsmodulen består av en kommunikationsenhet och en Blåtandsmodul. All kommunikation mellan styr- och sensormodulerna sker via bussen. Kommunikationen mellan kommunikationsmodulen och persondatorn sker via Blåtand. Flödesschema för roboten finns i figur 4. En övergripande bild av systemet ges i Figur 5. Ett fullständigt kopplingschema över delsystemet återges i Appendix A.1.

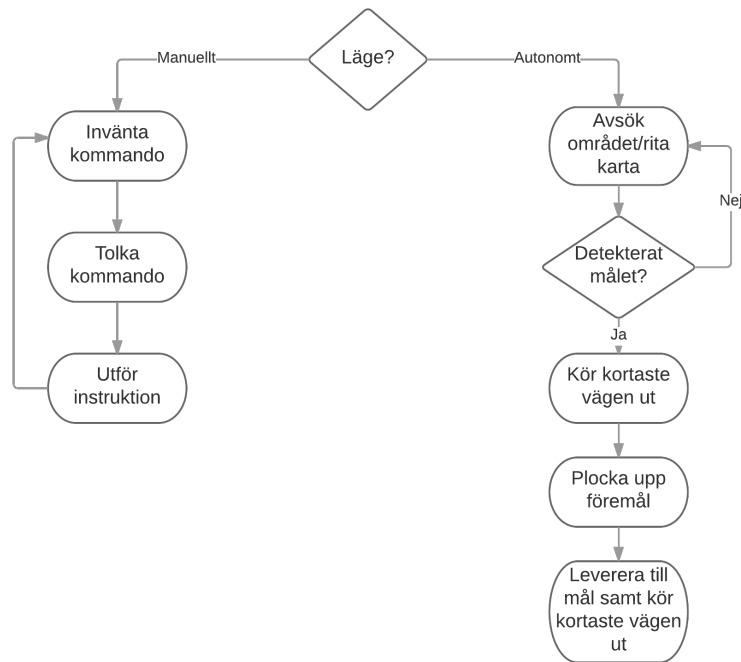
Kommunikationsenheten består av en mikroprocessor beskriven i avsnitt 3.2.

#### 3.2 AVR-dator

Huvudmodulen i kommunikationsmodulen utgörs av en AVR av typen ATmega16, se [1]. Mikroprocessorn är kopplad till en I<sup>2</sup>C-bus för att kunna ta emot mätvärden från sensormodul (se avsnitt 4 och 6). En blåtandsmodul, se 3.4 är kopplad till huvudprocessorn och kommer utgöra kommunikation mellan persondator och roboten. PIN-konfigurationen för processorn återges i Tabell 2.

#### 3.3 Buss

Det bussprotokoll som kommer kommunicera mellan de olika modulerna är I<sup>2</sup>C. I<sup>2</sup>C består utav två Kommunikationsmodulen kommer agera som "master" medan styr- och sensormodulerna är "slaves". Det innebär att det är kommunikationsmodulen som bestämmer bussens klockfrekvens. Kommunikationsenheten kommer alltså att bestämma hur och när data ska skickas över bussen.



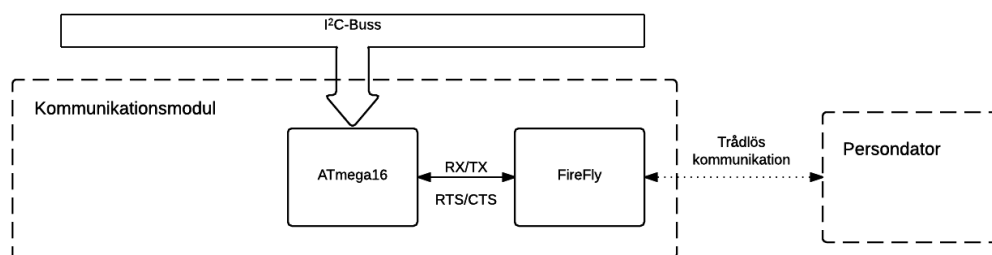
Figur 4: Flödesschema för roboten

### 3.4 Blåtandsmodul

Blåtandsmodulen som kommer att användas är en FireFly. RTS/CTS (ready to send/clear to send) ska hållas höga för att möjliggöra dataöverföring. När en flagga blir rest skickas informationen som TX/RX-signal (transmit/receive) till FireFly som sedan konverterar till informationssignaler som persondatorn kan ta emot.

### 3.5 Labyrintkarta

Kommunikationsenheten kommer att sköta de beslut rörande vilken väg roboten ska ta. Då sensorerna upptäcker en vägg kommer värden från avståndssensorerna att skickas till kommunikationsmodulen. Dessa värden kommer att ligga till grund för en matris som beskriver hur labyrinten ser ut. Enligt projektdirektiven och banspecifikation vet enheten om att labyrinten är 6x6 meter, i övrigt kommer labyrinten till en början att vara tom och efterhand fyllas på med väggar.



Figur 5: Blockschemata över kommunikationsmodulen

PIN	Funktion	signalnamn
9	RESET	RST
10	VCC	VCC (+5V)
11	GND	GND
14	RECEIVE_DATA	RXD
15	TRANSMIT_DATA	TXD
16	REQ_TO_SEND	INT0
17	CLR_TO_SEND	INT1
22	SCL	BUS_CLK
23	SDA	BUS_DATA
30	AVCC	AVCC (+5V)
31	GND	GND

Tabell 2: PIN-konfiguration för kommunikationsmodulens AVR

### 3.6 Kartläggningsalgoritmer

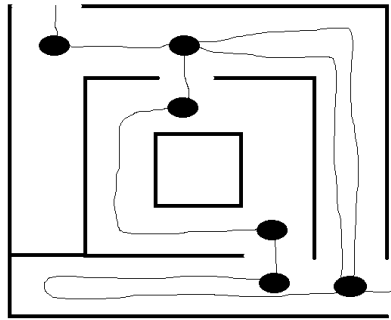
Ett av kraven på roboten är att den måste kartlägga tillräckligt mycket av labyrinten tills den hittar målet, för att sedan kunna ta den kortaste vägen fram och tillbaka till/från startpunkt och mål. Nedan listas olika alternativ för att ta beslut i kartläggningen.

Oavsett vilken algoritm som väljs för att kartlägga labyrinten så bör beslutsnoder placeras ut på kartan. Detta för att (kortaste vägen ut ur labyrinten) kunna ta beslut enklare vid uträkning av snabbaste väg.

#### 3.6.1 Algoritmalternativ 1

Trémaux-algoritmen är ett alternativ vilket illustreras i Figur 6. Den används främst för att lösa labyrinter. Den markerar varje ruta i labyrinten som den redan har be-

sökt, och hur många gånger den har besökt den. Den inleder kartläggningen med att välja en slumpmässig riktning, sedan markeras beslutsnoder. I noderna tar roboten beslut efter att välja den ruta som den inte har besökt/har besökt minst antal gånger. På så sätt kan hela labyrinten kartläggas. Denna algoritm hanterar dock inte att utesluta delar av labyrinten som den kan strunta i att undersöka.



Figur 6: En robot som avsökt en labyrint med Trémaux-algoritmen. Punkterna är beslutsnoder.

### 3.6.2 Algoritmalternativ 2

”Följ en vägg” är alternativ nummer 2. Algoritmen innebär att roboten följer en vägg och alltid tar vänster eller alltid tar höger vid beslutsfattning.

### 3.6.3 Algoritmalternativ 3

Alternativ 3 kallas för ”Den slumpmässiga musen” och fungerar så att den tar en slumpmässig riktning när den når en återvändsgränd. Innan dess går den bara rakt fram.

## 3.7 Kortaste-vägen-algoritmen

För att roboten ska kunna hitta den kortaste vägen så kommer kommunikationsmodulen att ta beslut om vilken väg som är kortast. Utifrån den kartmatris med beslutsnoder som hitills har genererats kommer roboten att välja en kortaste väg som ett vanligt optimeringsproblem. Det innebär att roboten kommer att räkna ut ”kostnaden” mellan varje nod och välja den billigaste vägen till startet och målet.

## 3.8 Manuell styrning

Ett krav är att roboten ska kunna styras manuellt genom en kontroll, tangentbord eller dylikt. Detta arbete kommer kommunikationsenheten att ta hand om.

Förutom kartdata så kommer också styrdata kunna skickas mellan roboten och en person dator vilket möjliggör manuell styrning i realtid. Den styrdata som kommer att skickas kommer att skickas via blåtandmodulen.

## 4 Sensormodul

Modulen utgörs av utbytbara sensorer som kopplas mot en mikroprocessor i vilken utsignalerna från sensorerna översätts till uppmätt storhet, exempelvis meter. Denna kan sedan skicka mätvärdena vidare till övriga delar av delsystemet över en I<sup>2</sup>C-buss.

### 4.1 Översikt av delsystemet

I modulen ingår ultraljuds och lasersensorer, IR-detektor, fototransistor samt ett gyro vilka beskrivs närmare i avsnitt 4.4 - 4.7. AVR-datorn utgörs av en Atmel ATmega16 och beskrivs utförligare i avsnitt 4.2. I Figur 7 ges en övergripande bild av delsystemet genom ett blockschema över modulen och Figur 8 ger en övergripande skiss över hur programmet i AVR-datorn är tänkt att exekveras. Ett fullständigt kopplingschema över delsystemet återges i Appendix A.1.

### 4.2 AVR-dator

Huvudmodulen i sensormodulen utgörs av en AVR av typen ATmega16, se [1]. Mikroprocessorn är kopplad till en I<sup>2</sup>C-bus för att kunna skicka mätvärden till kommunikationsmodulen (se avsnitt 3 och 6) samt så är systemets samtliga externa sensorer (se avsnitt 4.4 - 4.7) kopplade mot processorn. PIN-konfigurationen för processorn återges i Tabell 3.

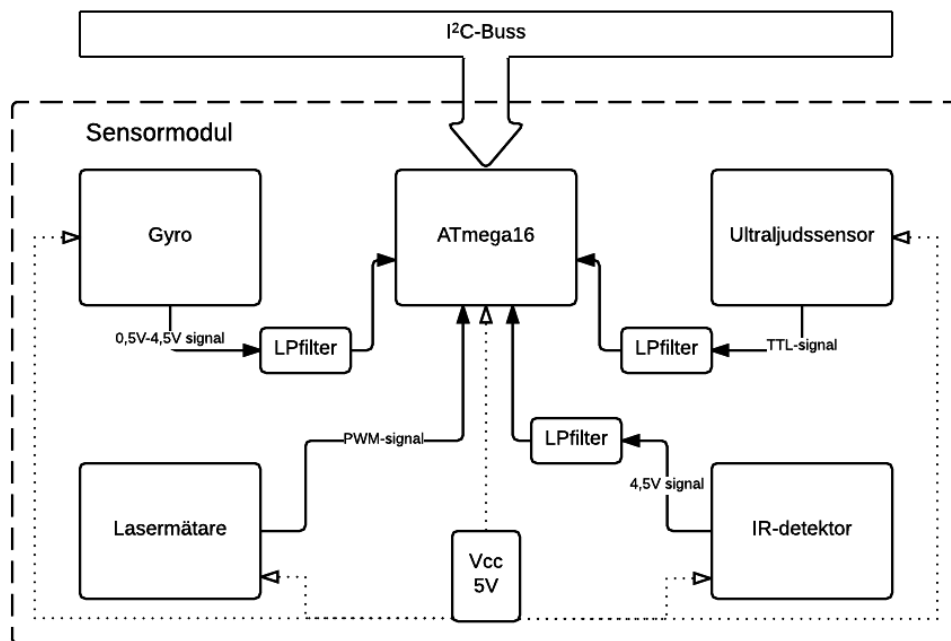
### 4.3 Gränssnitt

Gränssnittet delas upp i en intern del som beskriver kommunikationen mellan sensorerna och mikroprocessorn samt en extern del som behandlar kommunikationen mellan sensormodulen och övriga moduler i systemet, vilka beskrivs nedan.

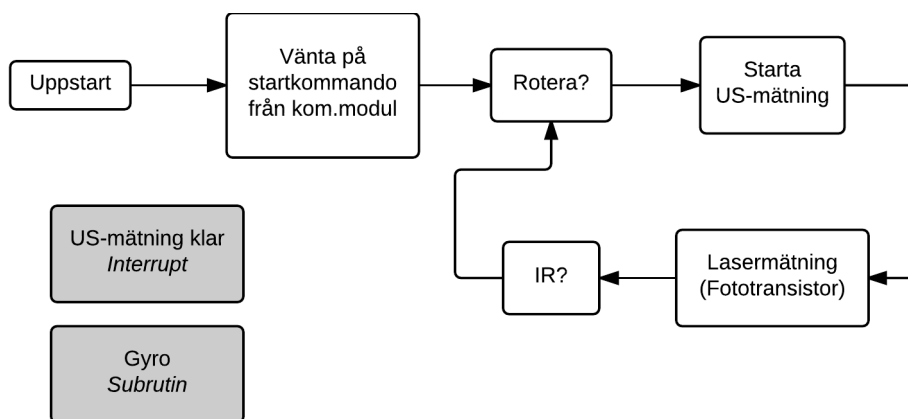
#### 4.3.1 Sensorer – AVR-datorn

Mätvärden från samtliga sensorer skickas till huvudmodulen, filtreras innan i de fall det behövs och i gryots fall så A/D-omvandlas signalen på huvudmodulen.





Figur 7: Blockschema över sensormodulen



Figur 8: Konceptuell skiss över sensormodulens mjukvara

PIN	Funktion	signalnamn
1	OUT	LASER_ENABLE
2	IN/OUT	LASER_TRIGGER_READ
3	IN	IR_DETECTION
4	OUT	IN_ENABLE
9	RESET	RST
10	VCC	VCC (+5V)
11	GND	GND
22	SCL	BUS_CLK
23	SDA	BUS_DATA
30	AVCC	AVCC (+5V)
31	GND	GND
32	AREF	AREF (+5V, ev 4.5V)
33	IN	PHOTO
34	IN	USL_ECHO
35	OUT	USL_TRIGGER
36	IN	USRR_ECHO
37	OUT	USRR_TRIGGER
38	IN	USFR_ECHO
39	OUT	USFR_TRIGGER
40	ADC	ANGLE_RATE

Tabell 3: PIN-konfiguration för sensormodulens AVR

### 4.3.2 Sensormodulen – Huvudenheten

Från sensorns huvudmodul skickas önskad information med en I<sup>2</sup>C-buss (se avsnitt 6) till kommunikationsmodulen där bussens klocka går på pin 22 och data skickas via pin 23.

## 4.4 Ultraljudssensor

Ultraljudsensorn är av typ SRF04, se [2], och mäter med hjälp av ultraljud avstånd från 3cm till 3m. En avståndsmätning görs genom att mäta tiden det tar för en utsänd ultraljudspuls att färdas till ett objekt samt att efter studs färdas tillbaka till mottagaren.

Sensorn kopplas till *VCC* (+5V) och *GND*. När en mätning skall startas går triggersignalen *USX\_TRIGGER* från AVRen från viloläget 'låg' till 'hög' under minst 10  $\mu s$ . När signalen återgår till 'låg' så skickas ljudpulsen från sensorn ut och tidräkningen i AVRen startas. Efter 100  $\mu s$  (i syfte att undvika brus från pingningen av *USX\_TRIGGER*) går *USX\_ECHO* 'hög' och när detektorn på ultraljudssensorn registrerar ljudpulsen, efter att den har färdats sträckan fram till ett objekt och tillbaka, så går signalen *USX\_ECHO* tillbaka 'låg' och på den negativa flanken avslutas tidräkningen i AVRen. Genom att dividera den uppmätta tiden angiven i  $\mu s$  med 58 fås det uppmätta avståndet i cm.

## 4.5 Lasersensor

Lasersensorn är av typ LIDAR-LITE, se [4] är en lite mer avancerad avståndsmätare som mäter avstånd 0 – 40 m med hjälp av reflekterat laserljus. Mätaren triggas genom en logisk nolla och gör då en mätning där den kollar på vinkeln från den reflekterade signalen och översätter vinkeln internt till ett avstånd och representerar detta avstånd med hjälp av en pulsbredds modulerad signal där det går 10 $\mu s/cm$ .

Lasersensorn kopplas till *VCC* (+5V) och *GND*. En mätning triggas då *LASER\_TRIGGER\_READ* går från positiv till negativ flank i en klockpuls och *LASER\_ENABLE* är en positiv etta. Efter trigging ställs dataregistret för in och utgångar på AVR:en om och *LASER\_TRIGGER\_READ* moduleras om till en insignal. Efter en viss tid efter trigging skickas en pulsbredds modulerad signal in på *LASER\_TRIGGER\_READ* som ska översättas till ett avstånd och kunna skickas ut på bussen.

## 4.6 IR-detektor

IR-detektorn är av typ IRM-8601S se [6] vilket fungerar som en fjärrkontroll där den letar efter en IR-signal i spannet runt 38 kHz. Detektorn letar efter en IR-signal med frekvens runt 38 kHz och skickar ut en logisk etta vid detektion.

IR-detektorn kopplas till  $VCC$  (+5V) och  $GND$ . Detektorn är på när modulen är igång och letar konstant efter en signal med rätt frekvens.  $IR\_DETECTION$  är en signal till AVR:en och kollar om den får in en positiv etta. För att detektorn ska fungera korrekt och inte bli mättad så måste de avlästa signalerna på  $IR\_DETECTION\_OUTPUT$  vara i intervallet  $100\mu s - 200\mu s$ . Detta löses av IR-sändaren.

## 4.7 Gyro

Gyrot är av typen MLX90609, se [3], som är en vinkelhastighetsmätare som kommer att användas för att kunna göra precisa rotationer. När roboten kommer fram till en korsning så ska en rotation påbörjas och avslutas då den integrerade vinkelhastigheten överensstämmer med den önskade slutvinkeln.

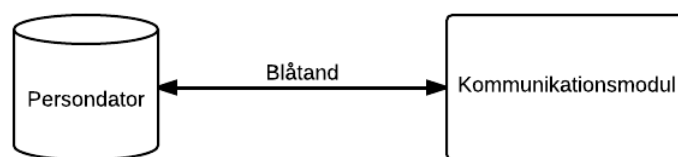
Gyrot kopplas till  $VCC$  (+5V) och  $GND$  och är igång när modulen är igång. Utsignalen från gyrot  $ANGULAR\_RATE$  är en analog signal som AD-omvandlas in på AVR:en och integreras sedan över tiden. Startvinkeln på gyrot behöver kalibreras enligt

$$V_{out} = Bias + Gain * AngularRate$$

där  $V_{out}$  är utsignalen,  $Bias$  och  $Gain$  är konstanter som behöver kalibreras efter olika förhållanden, samt  $AngularRate$  som är vinkelhastigheten. Då gyrot ger en analog utsignal så kommer den även behövas låpassfiltreras, detta kommer att göras på AVR:en.

## 5 Persondator

Persondatorn kommer sköta kommunikationen mellan användaren och roboten genom en bluetooth-länk läs mer i avsnitt 3.4 samt se Figur 9, den ska kunna ge order till roboten så man kan köra den helt manuellt från persondatorn. Ska även ta emot information från roboten i det autonoma läget och använda denna information för att få en bra bild av vad som händer kunna ge en grafisk model av det avsökte området.



Figur 9: Kommunikation persondator – system

## 5.1 Styrkommandon

För att kunna skicka kommandon från persondatorn och få roboten att reagera på dessa kommandon så kommer tangenter kopplas till key events och skickas via blåttandslänk till roboten och jämföra dessa kommandon med olika driv och styrutiner på kommunikationsmodulen. Till exempel pil-tangenten upp kopplas till ett key event Forward och triggat igång de båda drivarna på roboten under ett visst antal klockpulser. Och motsvarande för vänster och höger piltangent till key events Right\_turn , Left\_turn och triggat då bara den ena motordrivaren på roboten.

## 5.2 Grafisk modellering

För att kunna få en bra överblick av det avsökt området så bör det finnas en grafisk modellering av området som visas på persondatorn. Området kommer på något vis representeras av en matris av ettor och nollor, denna matris bör då översättas till en lättare visuell representation.

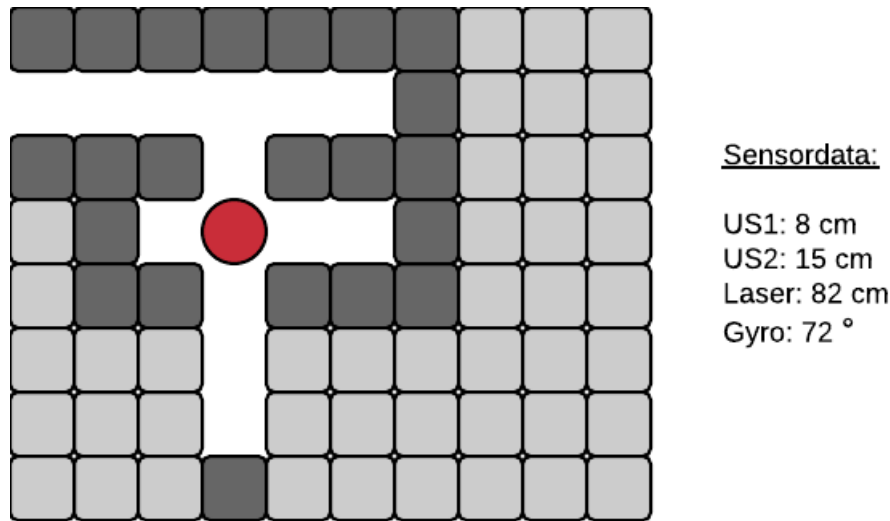
Med hjälp av ett externt bibliotek likt Qt i C++ kan ett gränssnitt utvecklas som översätter ovan nämnda matris till en grafisk representation där värdena istället representeras med kvadrater i olika färg för att skilja vägg från öppen yta vilket visas konceptuellt i Figur 10

# 6 Kommunikation mellan delsystem

Kommunikationen mellan delsystemen kan delas in i två olika kategorier, dels kommunikationen mellan AVR-datorerna som är monterade på roboten (se avsnitt 6.1) och dels mellan roboten och persondatorn (se avsnitt 6.2).

## 6.1 Kommunikation mellan processorerna

Processorerna ska kommunicera med varandra genom en I<sup>2</sup>C-buss. Kommunikationsmodulen (C-modulen) kommer agera master medan sensormodulen (S-Modulen) och styrmodulen (A-Modulen) agerar slavar. På så sätt förenklas de-



Figur 10: Konceptskiss över egenutvecklat gränssnitt

signen. A-Modulen ska endast ta emot data och behöver således inte skicka något över bussen. S-Modulen behöver inte ta emot något data utan ska bara skicka sensordata till C-Modulen. C-Modulen ska ta emot sensordata från S-modulen, skicka vidare denna data tillsammans med styrbeslut till A-Modulen. Detta ska ske sekvensiellt, C-Modulen ber S-Modulen om data. När överföringen är klar påbörjas överföringen till A-Modulen och när den sedan är klar påbörjas en ny överföring från S-Modulen.

## 6.2 Trådlös kommunikation mellan robot och persondator

Via en Blåtandslänk kommer styr signaler skickas från persondatorn till roboten och kartdata skickas från kommunikationsmodulen till persondatorn. För kommunikationsmodulen kan anslutningen till FireFly-modulen ses som en serieport och sköta kommunikationen därefter.

När roboten ska styras manuellt ska persondatorn skicka ett kommando som initierar ett nytt läge hos roboten där den i stället för att köra autonomt lyssnar efter styrkommandon från datorn. Dessutom ska ett kommando från datorn kunna starta robotens autonoma sökläge. När roboten är i autonomt sökläge ska i stället persondatorn hela tiden lyssna efter data från roboten för att kunna visualisera labyrinten med en karta.

## 7 Implementeringsstrategi

Systemets implementering genomsyras av filosofin *divide et impera*, d.v.s. att både konstruktion (avsnitt 7.1) och testning (avsnitt 7.3) av systemet skall göras i mindre delsystem, vilka i sig skall delas upp i mindre beståndsdelar där respektive dels funktionalitet säkerställs innan den integreras med övriga delar.

### 7.1 Konstruktion

Mycket av konstruktionen kommer att ske parallellt på grund av att gruppen består av sju personer. Detta medför att styrmodul, kommunikationsmodul samt buss kommer att påbörjas först. Under konstruktionen av delsystemen kommer även sensormodulen att börja konstrueras och konstruktionsarbetet kommer att pågå under en lite längre tid då alla sensorer inte behövs för de tidiga testerna. Under konstruktionen av varje delsystem så kommer det finnas extra AVR:er som kan användas för att testa respektive delsystem. Under konstruktionen av sensormodulen så kommer även en blåtandslänk upprättas så kommunikation mellan dator och robot är möjlig. När detta är klart så har en milstolpe uppnåtts då roboten kan styras manuellt. Efter detta kommer fokus ligga på mjukvara och att få roboten att utföra projektets autonoma uppdrag. Senare, närmare de avslutande milstolparna, så kommer gripkolon konstrueras och även IR-detektorn.

### 7.2 Återkoppling från systemet

För att kunna få bra koll på systemet medan det testas så behövs en återkoppling. När blåtandslänken är upprättad så kan data från kommunikationsmodulen skickas till en persondator och analys av sensorer samt vad som skickas ut på bussen göras. Vad som skickas ut på bussen kan även kontrolleras genom ett oscilloskop och då se vilka och när signaler skickas. När större förståelse för den interna kartan finns så ska det även vara möjligt att få en grafisk beskrivning om hur roboten uppfattar sin omgivning och se om det är rätt.

### 7.3 Testning

Testning av modulerna sker kontinuerligt under konstruktionen där varje del av respektive delsystem testas separat innan det monteras och integreras i respektive delsystem. Målsättningen är alltså att undvika så kallad "big bang"-testning genom att säkerställa respektive dels funktion innan de sammanförs.

När modulernas funktion testats och adekvat prestanda i den isolerade miljö de har

testats i kan säkerställas, så skall de monteras på roboten och integreras. Systemintegrationen medför ytterligare tester där säkerställandet av bussens funktion är av högsta prioritet. Integrationstesterna hämtar inspiration ur respektive delsystems testfall fast istället för simulerade testdata används data från de övriga delarna i systemet för att säkerställa att hela funktionskedjan är intakt.

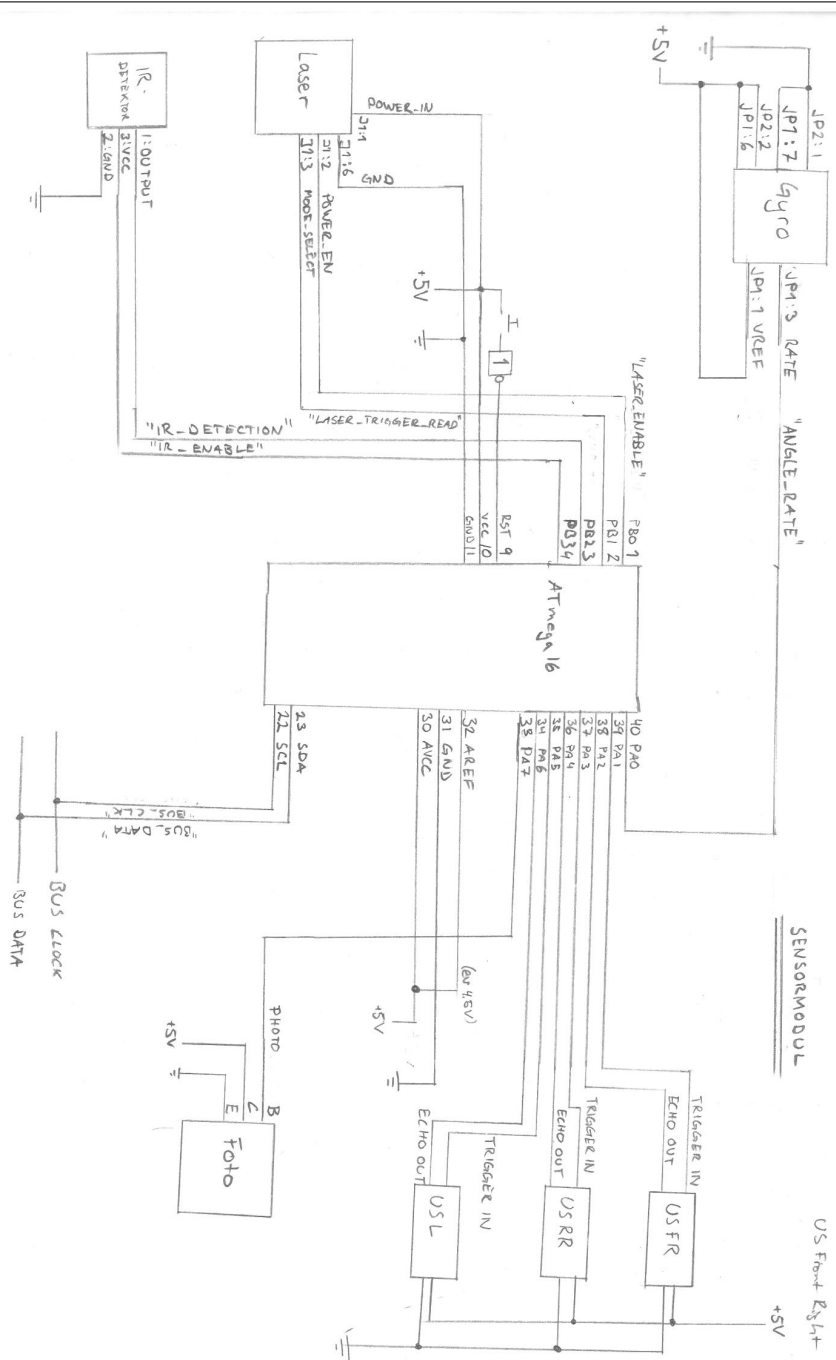
Systemet kommer att utsättas för omfattande sluttester som syftar till att säkerställa att roboten är redo för att tävla vid utsatt tid. Testerna kommer att genomföras genom att låta systemet köras i en bana av den typ som det är avsett att verka i. Dessa kommer att delas upp i testning av bana bestående enbart av korridorer respektive en bana där öppna ytor förekommer. Prioriteten ligger, i enlighet med kravspecifikationen, att först säkerställa att systemet presterar på en tillfredsställandenivå i de tester som avser en bana med enbart korridorer.

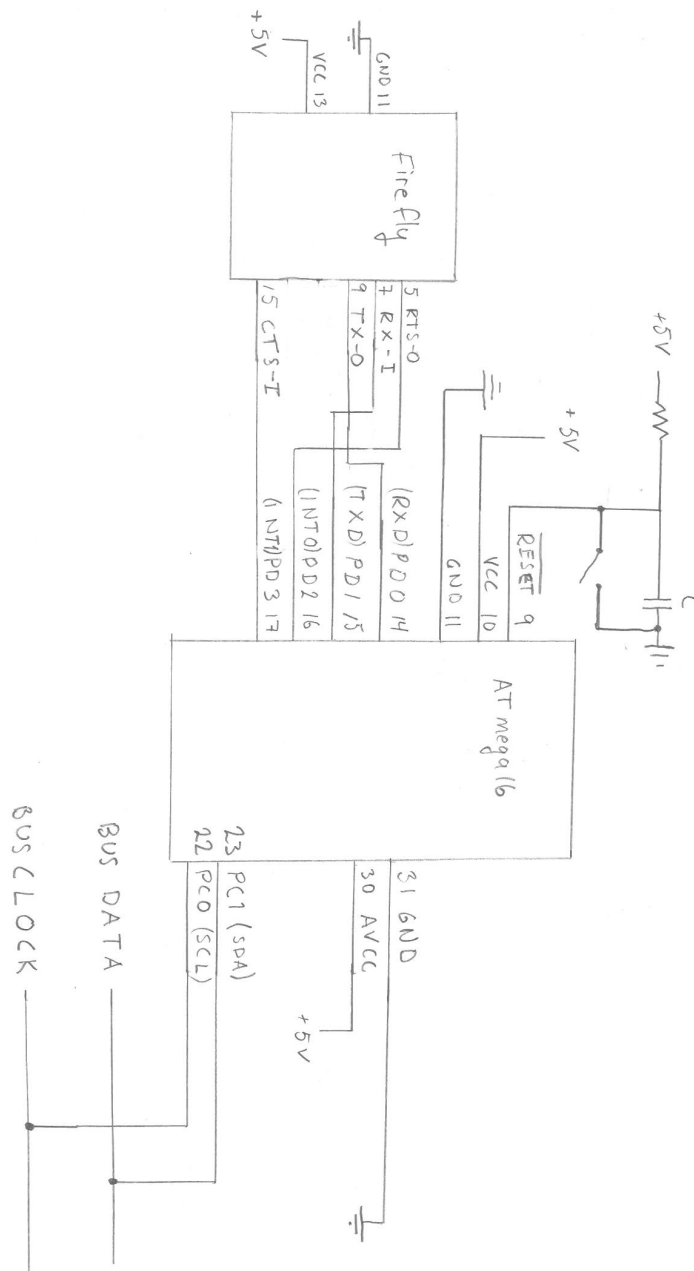




## **A Appendix**

### **A.1 Kopplingscheman för delsystemen**









## Referenser

- [1] Atmel, “8-bit AVR Microcontroller with 16K Bytes IN-System Programmable Flash” 2466D-AVR-09/02, Sep. 2002.
- [2] Devantec, “SRF04 Ultrasonic Range Finder”, Sep. 2003.
- [3] Melexis, “MLX90609 Angular Rate Sensor (Standard version)” 390109060901 datasheet, Feb. 2008.
- [4] PULSEDLEDLIGHT LLC, “LIDAR-LITE“, July.1014.
- [5] SIEMENS, “Silicon NPN Phototransistor”, BPX 43 Feb. 199.
- [6] Everlight, “Miniature type IRM Lateral, IRM8601S”