



LIPS Systemskiss

Version 1.0

Status

Granskad	Fredrik Olsson	2016-02-14
Godkänd	Mattias Krysanter	2016-02-18



PROJEKTIDENTITET

2016/VT, Grupp 1

Linköpings Tekniska Högskola, ISY

Gruppdeltagare

Namn	Ansvar	Telefon	E-post
Albin Detterfelt		070-655 25 99	albde282@student.liu.se
Klas Gudmundsson		072-714 06 66	klagu863@student.liu.se
Martin Haugsbakk		070-270 63 36	marha996@student.liu.se
Milton Johansson		073-037 01 64	miljo274@student.liu.se
Sandy Klaff	Dokumentansvarig	070-293 77 49	sankl660@student.liu.se
Fredrik Olsson	Projektledare	072-315 38 68	freol454@student.liu.se
Martin Prage		073-694 95 24	marpr146@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

Innehåll

Dokumenthistorik

1	Inledning	1
1.1	Konstruktionsskiss	1
2	Översikt över systemet	2
2.1	Mikrodator Atmel ATmega16	2
3	Styrmodul	3
3.1	Översikt av delsystem	3
3.2	Framdrivning	3
3.3	Motorkontroll	4
3.4	Display	4
3.5	Gripklo	5
3.6	Gränssnitt	5
4	Kommunikationsmodul	6
4.1	Översikt av delsystem	6
4.2	Buss	6
4.3	Blåtandsmodul	6
4.4	Labyrintkarta	7
4.5	Kartläggningsalgoritmer	7
4.5.1	Algoritmalternativ 1	7
4.5.2	Algoritmalternativ 2	7
4.5.3	Algoritmalternativ 3	8
4.6	Kortaste-vägen-algoritmen	8
4.7	Manuell styrning	8
5	Sensormodul	9
5.1	Översikt av delsystemet	9
5.2	AVR-dator	9
5.3	Gränssnitt	9
5.3.1	Sensorer – AVR-datorn	9
5.3.2	Sensormodulen – Huvudenheten	10
5.4	Ultraljudssensor	10
5.5	Lasersensor	10
5.6	IR-detektor	10
5.7	Gyro	10



6	Persondator	11
6.1	Grafisk modellering	11
6.1.1	Representation genom egenutvecklat gränssnitt	11
6.1.2	Representation genom MATLAB	11
	Referenser	12



Dokumenthistorik

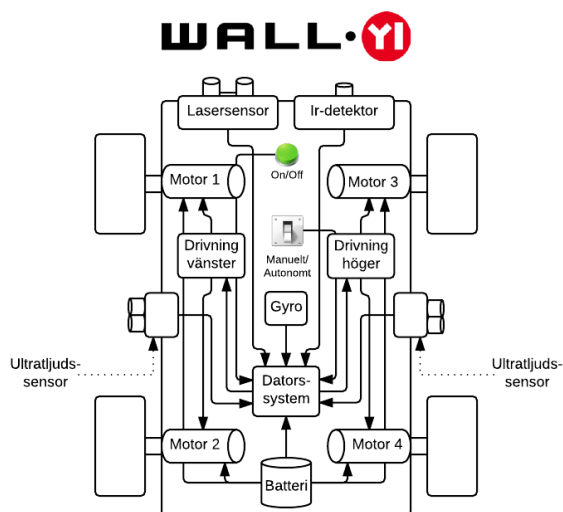
Version	Datum	Utförda förändringar	Utförda av	Granskad
0.1	2016-02-14	Första utkast	AD, FO, KG, MH, MJ, MP, SK	FO
1.0	2016-02-18	Första version	FO, MH, MJ, SK	FO

1 Inledning

Projektets syfte är att utveckla och konstruera en undsättningsrobot. Undsättningsroboten ska användas för att söka igenom en outforskad labyrint och hitta ett mål som ska representera en nödställd. När roboten har hitat sitt mål ska den ta sig ut och hämta ett föremål som sedan ska in till den nödställda. Detta ska roboten göra genom att ta den kortaste möjliga vägen. I projektet så ska mjukvara för de olika delsystemen konstrueras. Denna systemskiss kommer att ge en skiss av de olika delsystemen.

1.1 Konstruktionsskiss

Roboten kommer bestå av ett grundchassi med fyra hjul där alla moduler och dess delsystem kommer sitta fast. Det kommer fästas ultraljudssensorer på robotens båda sidor samt en lasersensor i fram som har som uppgift att skicka information om robotens omgivning. Framtill kommer även en IR-detektor fästas som ska leta efter ett IR-torn (den nödställda). Mitt på robotens rotations-axel, alltså runt den axel som roboten roterar när den svänger ska ett gyro fästas som ska hjälpa till med reglering av systemet så roboten åker rakt. Till varje hjul på roboten är fyra motorer och till varje sidas hjulpar är en drivare kopplad som ska föra roboten framåt och anpassas efter värden som kommer indirekt från sensormodulen. För att driva alla moduler krävs ett batteri som kopplas till alla moduler och placeras längst bak på roboten. Sist ska det även fästas två switchar, där den första är en på/av-knapp och den andra växlar mellan autonomt och manuellt läge.



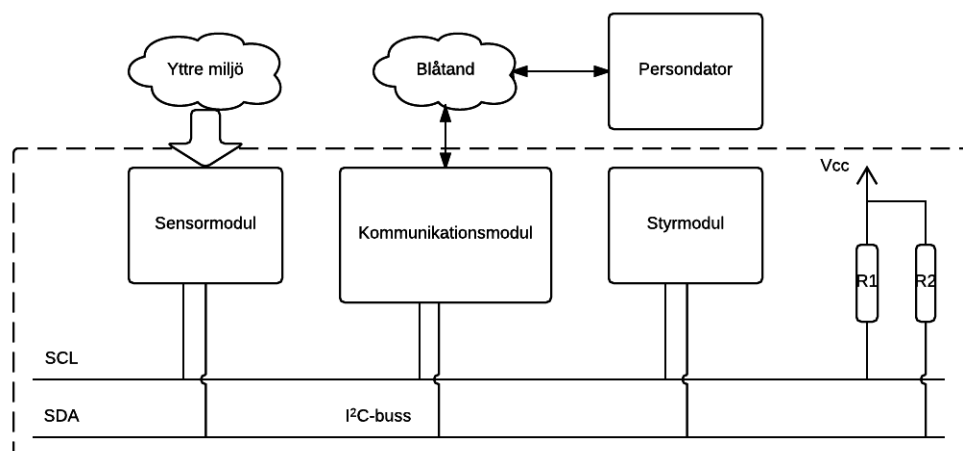
Figur 1: Översikt över roboten

2 Översikt över systemet

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.

2.1 Mikrodataor Atmel ATmega16

Samtliga moduler styrs huvudsakligen av var sin mikrodataor av typen Atmel ATmega16, en enchipsdataor 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.



Figur 2: Blockschemat över systemet

3 Styrmodul

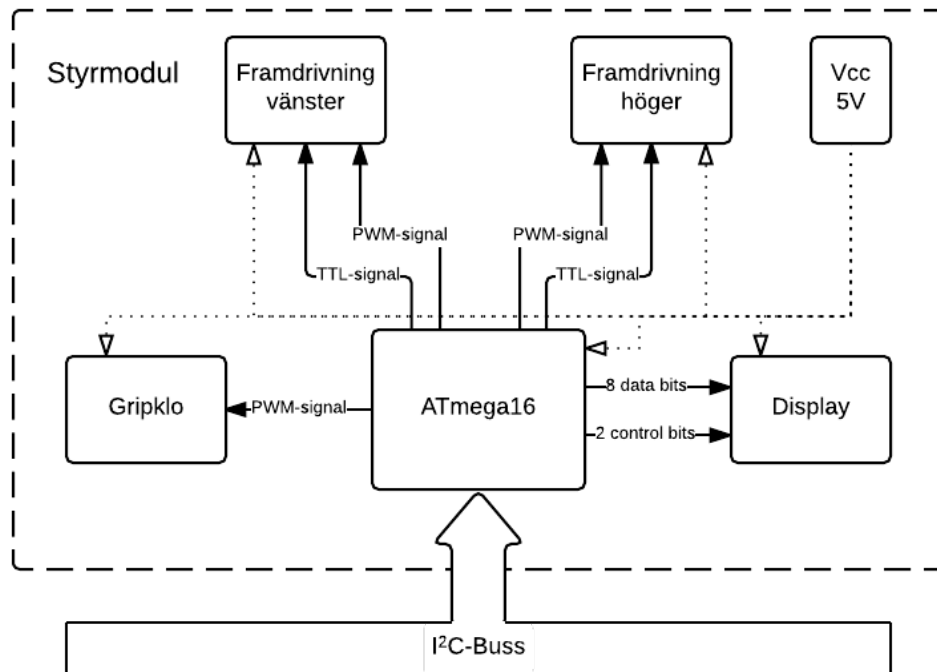
Styrmodulen är den modul som utför framdrivningen av roboten genom motorer fram och bak, griper tag i nödproviant genom en gripklo samt presenterar data från sensorerna via en display. Ett blockschema över styrmodulen visas nedan.

3.1 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 framdrivning av roboten.

3.2 Framdrivning

Microprocessorn tar emot navigationsbeslut från kommunikationsenheten. När enheten får kommandot kommer enheten ge signaler om rotationsriktning samt hastighet till motorenheterna. Vid körning kommer mikroprocessorn att använda data från sensorenheten och med hjälp av PID reglering köra rakt så länge inte ett nytt styrbeslut måste tas. Då ett beslut behövs kommer styrenheten invänta kommando från kommunikationsmodulen som anger vad styrenheten ska göra.



Figur 3: Blockschema över styrmodulen, dess komponenter och signalvägar

3.3 Motorkontroll

Motorerna kontrolleras genom att skicka två TTL/CMOS-kompatibla insignaler till respektive hjulpar. Den första signalen indikerar vilken rotationsriktning motorerna ska ha. Den andra signalen, en PWM signal, indikerar vilken hastighet motorerna ska ha.

3.4 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.

3.5 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 en servo som kontrolleras genom en PWM-signal på microcontrollern.

3.6 Gränssnitt

Genom en I2C-buss skickas sensorernas mätvärden från sensorenheten till kommunikationsenheten. Kommunikationsenheten ska därifrån skicka styrdata till styrenheten som reglerar framdrivningen, samt vidarebefordra mätvärden till persondatorn för uppritning av karta. Baserat på dessa tar kommunikationsenheten också beslut om navigering då sensorvärdena antyder nödgad kursändring.

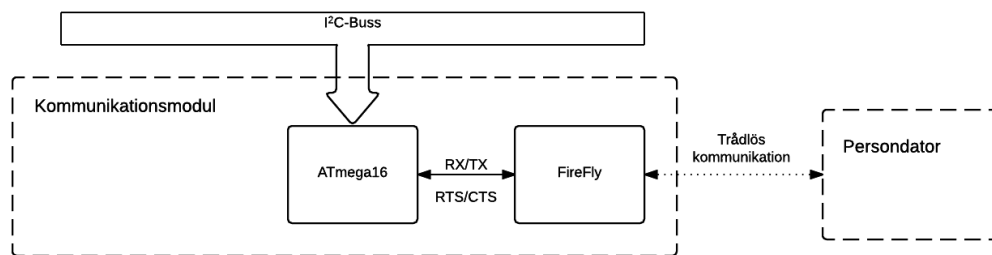
4 Kommunikationsmodul

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.

4.1 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.

Kommunikationsenheten består av en mikroprocessor beskriven i avsnitt 2.1.



Figur 4: Blockschema över kommunikationsmodulen

4.2 Buss

Det bussprotokoll som kommer kommunicera mellan de olika modulerna är I²C. Kommunikationsmodulen kommer agera som "master" medan styr- och sensormodulerna är "slaves". Det innebär att det är kommunikationsmodulen som bestämmer bussens klockfrekvens.

4.3 Blåtandsmodul

Blåtandsmodulen som kommer att användas är en FireFly. Flaggorna RTS/CTS (ready to send/clear to send) markerar när modulen är redo att skicka/ta emot data.

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.

4.4 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.

4.5 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.

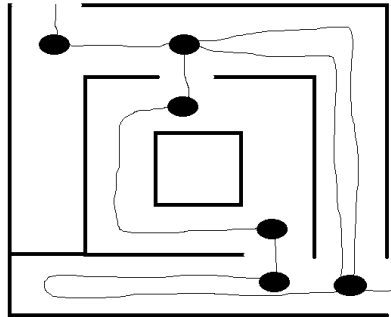
Oavsett vilken algoritm som väljs för att kartlägga labyrinten så bör beslutsnoder placeras ut på kartan. Detta för att (kortaste vägen ut ur labyrinten) kunna ta beslut enklare vid uträkning av snabbaste väg.

4.5.1 Algoritmalternativ 1

Trémaux-algoritmen är ett alternativ. Den används främst för att lösa labyrinter. Den markerar varje ruta i labyrinten som den redan har besökt, 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.

4.5.2 Algoritmalternativ 2

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



Figur 5: En robot som avsökt en labyrint med Trémaux-algoritmen. Punkterna är beslutsnoder.

4.5.3 Algorithmalternativ 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.

4.6 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.

4.7 Manuell styrning

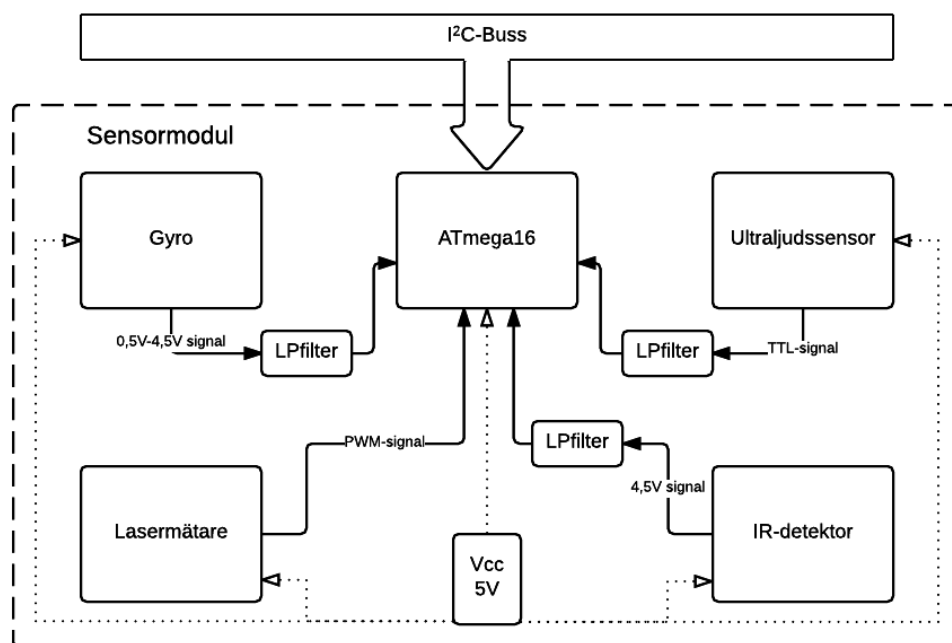
Ett krav är att roboten ska kunna styras manuellt genom en kontroll, tangentbord eller dylikt. Detta arbete kommer kommunikationsenheten att ta hand om. Förutom kartdata så kommer också styrdata kunna skickas mellan roboten och en personator vilket möjliggör manuell styrning i realtid. Den styrdata som kommer att skickas kommer att skickas via blåtandmodulen.

5 Sensormodul

Modulen utgörs av utbytbara sensorer som kopplas mot en mikroprocessor i vilken utsignalerna från sensorerna översätts till uppmätt storhet, exempelvis meter.

5.1 Översikt av delsystemet

I modulen ingår ultraljuds och lasersensorer, IR-detektor samt ett gyro vilka beskrivs närmare i avsnitt 5.4 - 5.7. AVR-datorn utgörs av en Atmel ATmega16 och beskrivs utförligare i avsnitt 2.1.



Figur 3. Blockschemat över sensormodulen

5.2 AVR-dator

Huvudmodulen i sensormodulen utgörs av en AVR av typen ATmega16 beskriven i avsnitt 2.1

5.3 Gränssnitt

5.3.1 Sensorer – AVR-datorn

Mätvärden från samtliga sensorer skickas till huvudmodulen, filtreras innan om det behövs och A/D-omvandlas på huvudmodulen.

5.3.2 Sensormodulen – Huvudenheten

Från sensorns huvudmodul skickas önskad information med en IC2-buss till kommunikationsmodulen.

5.4 Ultraljudssensor

Ultraljudsensorn mäter med hjälp av ultraljud avstånd från 3cm till 3m. Ultraljudsensorn kan då mäta avstånd till närliggande väggar och skicka de till sensorns huvudmodul för att lägga grund till avsökning, reglering och uppritande av miljö. Behöver en inspanning på 5 V och kommer ge ut en positiv TTL-signal

5.5 Lasersensor

Lasersensorn kan mäta avstånd med en lasersensor på 0-40 m med en säkerhet på ± 2.5 cm. Lasern kan då mäta avstånd till närliggande väggar och skicka de till sensorns huvudmodul för att lägga grund till avsökning, reglering och uppritande av miljö. Behöver en inspanning på 5 V och kommer ge ut en PMV-signal.

5.6 IR-detektor

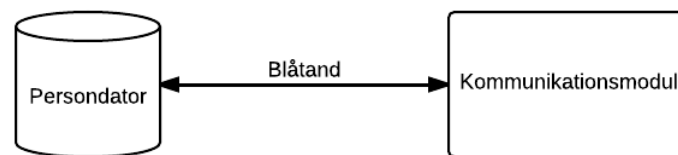
IR-detektorn kommer funka som en tv-kontroll och kommer att leta efter en tillräckligt stark IR-signal och då skicka ge ut en utsignal på ca 4.5 V. Detektorn ska leta efter en "IR-mast" och när den har funnit den, ska den med hjälp av tidigare information från sensorerna markera den på den utritade miljön. IR-detektorn behöver en inspanning på 5 V.

5.7 Gyro

Gyrot eller vinkelhastighetssensorn måste sitta på robotens mittpunkt, alltså den axeln som den roterar runt när den ändrar riktning. Kommer ge ut ett värde från 2.5 V till 4.5 V eller 2.5 V till 0.5 V beroende på vilket håll den roterar, värdet representerar vinkelförändringen från startläget. Den här informationen går att använda till en smidig reglering av roboten. Behöver en inspanning på 5 V.

6 Persondator

Persondatorn kommer sköta kommunikationen mellan användaren och roboten genom en bluetooth-länk läs mer i avsnitt 4.3 samt se Figur 6, 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 6: Kommunikation persondator – system

6.1 Grafisk modellering

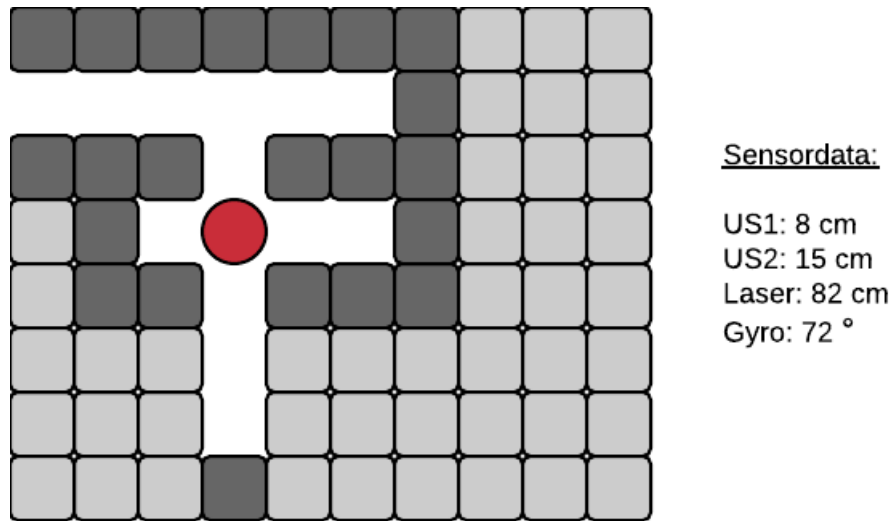
För att kunna få en bra överblick av det avsökte området så bör det finnas en grafisk modellering av området som visas på persondatorn. Området kommer på något vis representeras av en matris av ettor och nollor, denna matris bör då översättas till en lättare visuell representation.

6.1.1 Representation genom egenutvecklat gränssnitt

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 7

6.1.2 Representation genom MATLAB

MATLAB är ett väl använt program som har stöd för grafiska representationer genom till exempel SIMLINK, bygger även på matriser vilket gör det lätt implementerbart.



Figur 7: Konceptskiss över egenutvecklat gränssnitt

Referenser

- [1] *Projektmodellen LIPS* (2011), Tomas Svensson och Christian Kryssander, uppl. 1:1, Studentlitteratur AB, Lund. ISBN 978-91-44-07525-9.