
”Styrning av bilbana med en dator”
ett projekt för
ISY Fordonssystem

18 augusti 2002

Johan Dalin
xxxxxx-xxxx

Per Öberg
790611-3357
perob338

Innehåll

1	Inledning	5
2	Bilbanan	5
2.1	Konstruktionsskiss	5
2.2	Hårdvara	8
2.2.1	Sensorerna	8
2.2.2	Kopplingslåda	10
2.2.3	Drivsteg	11
2.2.4	Konnetktorblock	12
2.2.5	PCI6602	15
2.3	Mjukvara	15
2.3.1	Funktioner för autonom körning	15
2.3.2	Funktioner för manuell körning	16
2.3.3	Matlabprogram	17
3	Display	17
3.1	Hårdvara	17
3.2	Mjukvara	19
3.3	Begränsningar	19
	Referenser	19

1 Inledning

I en framtida kurs skall en bilbana som går att styra med en dator användas. Denna bilbana inköptes av **ISY** och datorkonstruktionen konstruerades av Johan Dalin då arbetande på **ISY Fordonssystem**. Konstruktionen hade dock några brister och därför gav man Per Öberg uppgiften att iordningställa bilbanan och utvärdera samt göra nödvändiga förändringar. Denna rapport syftar till att beskriva slutresultatet. En beskrivning av bilbanans konstruktion innan Pers modifieringar finns i [2]. Rapporten behandlar även den till bilbanan hörande LCD-displayen och dess programvara.

2 Bilbanan

2.1 Konstruktionsskiss

Bilbanan är av märket "Scalextric - Formula One"¹. Med en dator, programmet Matlab och ett I/O-kort kan bilarna köra autonomt. Dessutom kan man köra manuellt. Detta görs genom att ta in signaler ifrån manövreringshandtag till datorn via en seriell AD-omvandlare och sedan lägga ut en hastighet på skenor. Bilarna har en magnet i bakvagnen. Dessa gör att bilarna följer skenan bra i kurvorna. Magneterna utnyttjas också till viss del av denna tillämpning för att detektera när en bil passerar. Det finns två typer av sensorer till bilbanan.

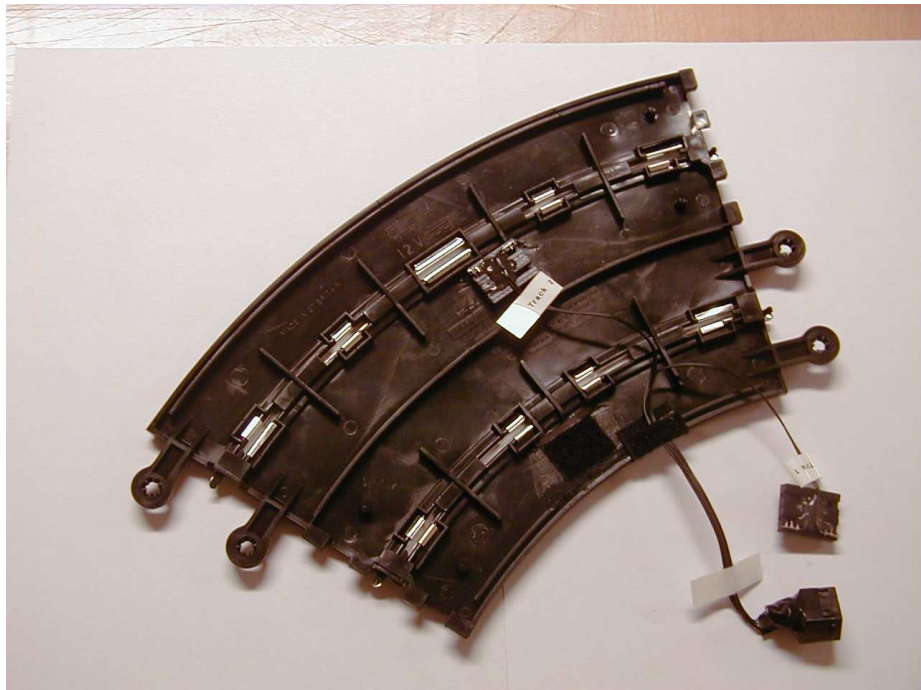
- Tungelementsensorer Sensorerna är tungelement av typ *RI-29AA*². När ett tungelement utsätts för ett magnetfält, av viss polaritet, sluts switchen. Tungelementen har en bandbegränsning på cirka 1 kHz, vilket är tillräckligt vid denna tillämpning. Tungelementen är placerade vid sidan om skenan eftersom detta ger bäst tillförlitlighet. Om man sätter tungelementet mellan skenorna störs magnetfältet lätt ut av metallen i skenorna. Konstruktionen finns avbildad i figur 1
- Optiska sensorer De optiska sensorerna är av typen *S4282-51* från Hamatsu³. S4282-51 är en s.k. störningsskyddad optoswitch som modulerar styrspänningen till en diod för att sedan söka efter just den signalen överlagrat i brus. Till optoswitchen används en IR-diod av typen HIR333 på 850 nm och med en spridningsvinkel på 8.5°. För en överblick av konstruktionen se figur 2 nedan.

Bilbanan har två spår. Varje spår har två olika typer av sensorer. Det finns en *varvräknare* och valfritt antal *checkpoints*. En extern transformator som genererar 13V är kopplad till en krets som kan *pulsa* ut den på banan. Hastigheten på bilen regleras genom ett Matlabinterface som kan variera "duty cyclen" på det pulståg som läggs ut på skenorna. Hela bilbanan finns avbildad i figur 3. Inkoppling sker genom att sensorerna kopplas in i valfri ordning med modularkontakter på sensorerna. Tungelementsensorerna kräver en kabelförgrening per sensor för inkoppling medan de optiska sensorerna redan har två ingångar som är likvärdiga. Ingen terminering behövs för inkopplingen av sista sensorn. En liten skiss över inkopplingsförfarandet finns nedan i figur 4.

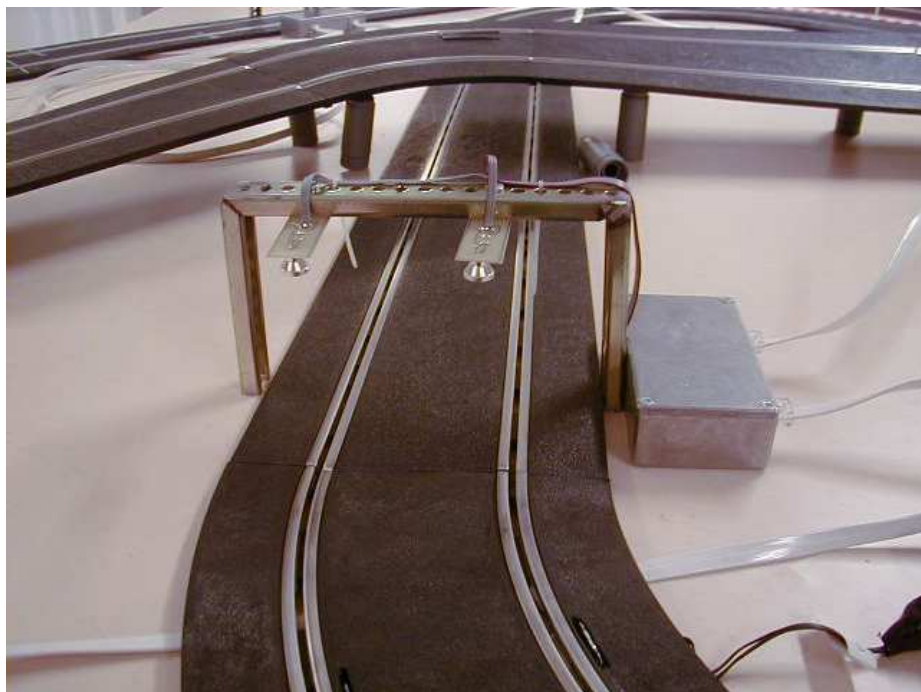
¹Inköpt på Sagolandet Torby, Linköping

²Inköpta från Elfa AB

³För mer information se <http://www.hpk.co.jp/eng/products/ssdproe.htm>



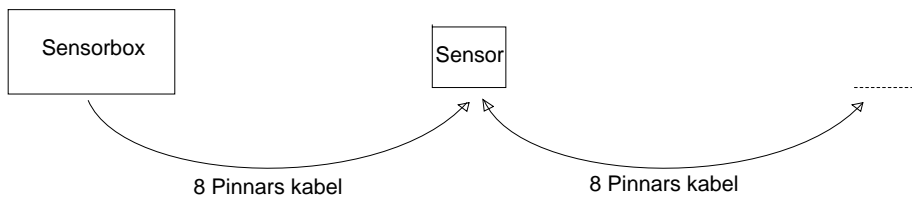
Figur 1: Tungelementsensorns utseende.



Figur 2: Optosensorns utseende.



Figur 3: Bilbanan i Fordonssystemens teknologlaboratorium.



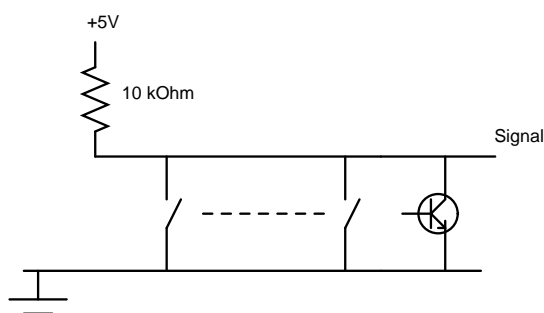
Figur 4: Kopplingsförfarande.

2.2 Hårdvara

I följande stycken beskrivs mer ingående den hårdvara som ingår i bilbanekonstruktionen.

2.2.1 Sensorerna

De två typer av sensorer som används är tungelementsensoter och optiska sensorer. Dessa kan samsas på bilbanan genom tungelementsensorererna är slutande brytare och de optiska har opencollectorutgång som fungerar som en brytare. (se figur 5 och 6)



Figur 5: Opencollector kretsar blandas med analoga switchar.

- Tungelementsensorn
Sensorn som används är som tidigare nämnts av typen *RI-29AA* och fungerar som slutare när den påverkas av ett magnetfält. Den är helt enkelt inkopplad mellan GND och respektive signalpinne så att den kan dra ned den till jordpotential när den signalerar.
- Optisk sensor
Den optiska sensorn är något mer avancerad i sitt utförande. Förutom att den känner av när bilen passerar har den även en inbyggd avstudsare/pulsformare som ser till att pulslängden hålls konstant oberoende av bilens hastighet samt att sensorn inte kan signalera igen under en bestämd tidsperiod. Sensorn bygger på en optoswitch av typen *S4282-51* från Hamatsu⁴. *S4282-51* är en s.k. störningsskyddad optoswitch som modulerar styrspanningen till en diod för att sedan söka efter just den signalen överlagrat i bruset. Till optoswitchen kopplades en IR-diod av typen *HIR333* på 850 nm och med en spridningsvinkel på 8.5 grader. Den optiska sensorn har även stöd för framtida inkoppling av mer avancerad buskommunikation. Detta har implementerats i form av plats lämnad på kretskortet för en stiftlist. Pinkonfigurationen för stiftlisten presenteras i figur 8 och 7 nedan.

Kretskortslayout, komponentplacering och kretsschema för sensorkorten finns för utskrift i katalogen "C:/Projekt_Y1/Kretskortslayout/".

⁴För mer information se <http://www.hp.k.co.jp/eng/products/ssdproe.htm>

1. GND
2. +5V
3. CHP Bana 2 (+5V 10 kOhm Pull Up)
4. CHP Bana 1 (+5V 10 kOhm Pull Up)
5. LAP Bana 2 (+5V 10 kOhm Pull Up)
6. LAP Bana 1 (+5V 10 kOhm Pull Up)
7. Future data 1
8. Future data 2

Figur 6: Pinkonfigurationen hos modularkontakterna till sensorerna



Figur 7: Numerering för stiftlist.

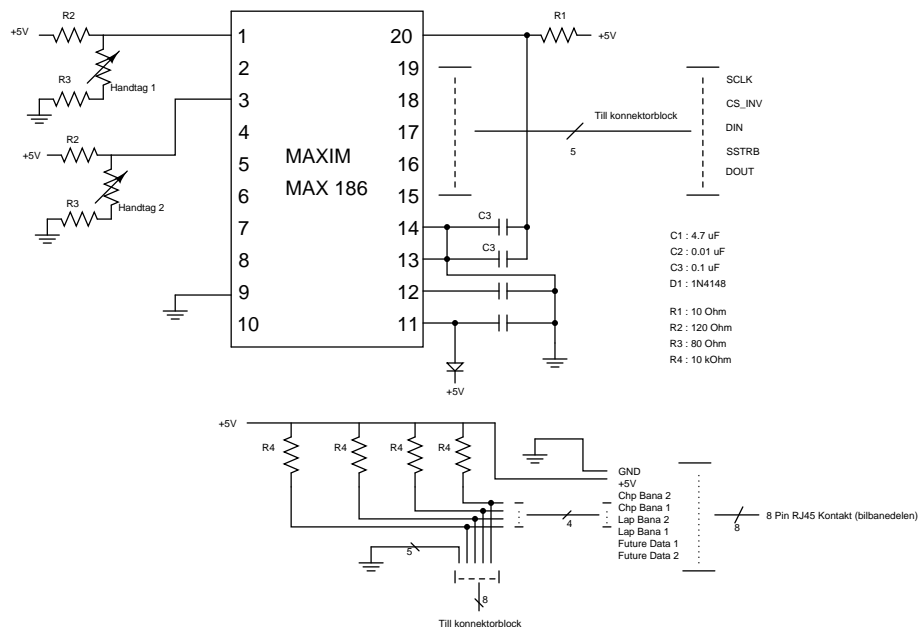
1. GND
2. Signal 2
3. Signal 1
4. FD1 (Future Data 1)
5. FD2 (Future Data 2)
6. VCC

Figur 8: Pinconfiguration för stiftlist.

Då tungelementsensornerna har s.k. kontaktstudsar finns f.n. en 200 ms delay i programvaran som hindrar sensorerna från att signalera mer än en gång. Då detta problem inte finns hos de optiska sensorerna kan delayen tas bort då tungelementsensornerna inte används.

2.2.2 Kopplingslåda

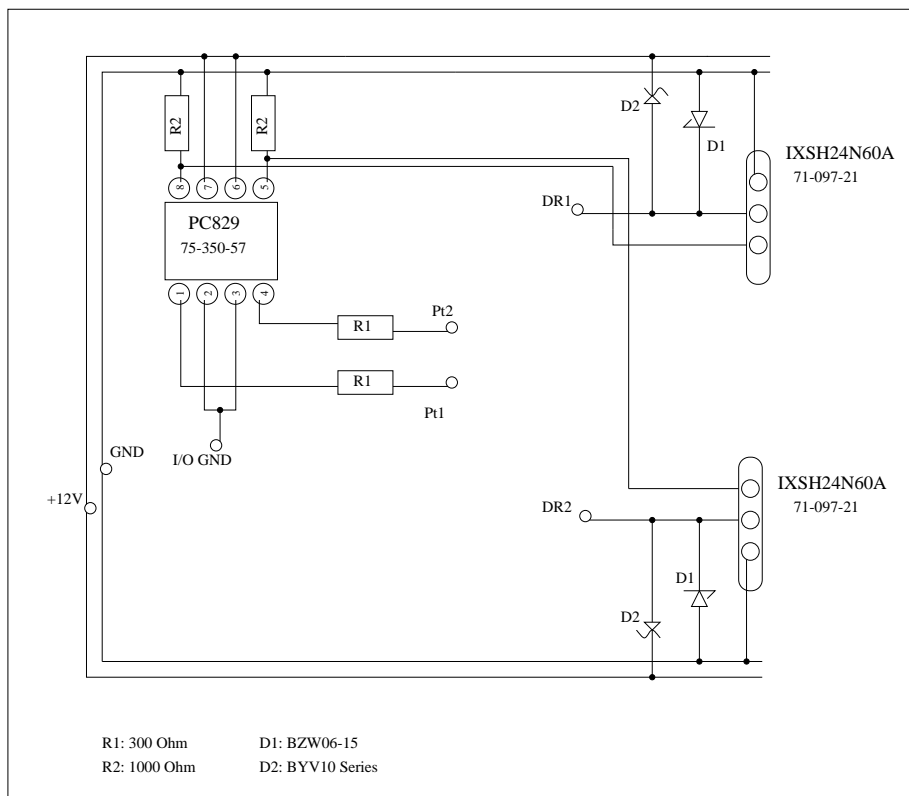
För att förenkla inkopplingen av sensorerna och körhandtagen finns en kopplingslåda innehållande en del styrelektronik. De manuella körhandtagen kan användas genom att en AD-omvandlare tar in signal och skickar värden till datorn. AD-omvandlaren har seriell dataöverföring. Kopplingschema för kortet i lådan finns i figur 9 nedan.



Figur 9: Kretsschema för Sensorlåda.

2.2.3 Drivsteg

Bilarnas hastighet styrs genom att lägga ut ett pulståg med en viss "duty cycle". Pulståg ifrån datorn går via en optokopplare och effekttransistor (IGBT i figur 10 nedan) ut till bilbanan.



Figur 10: Kretsschema för drivsteg.

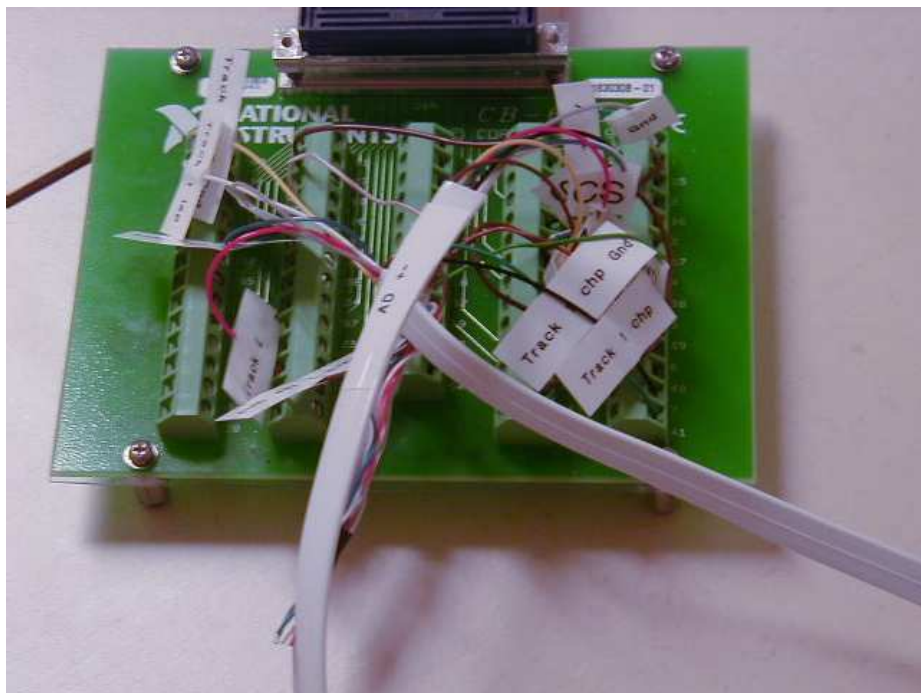
2.2.4 Konnektorblock

Signaler kopplas in på till kortet "PCI6602" via ett konnektorblock. Kopplingen till konnektorblocket är följande ⁵ :

Namn	Nummer
Bana 1 CHP	7
Bana 1 CHP GND	41
Bana 1 LAP	34
Bana 1 LAP GND	68
Bana 2 CHP	25
Bana 2 CHP GND	59
Bana 2 LAP	22
Bana 2 LAP GND	55
PT 1 (Pulståg 1)	5
PT 2 (Pulståg 2)	26
I/O GND	39,27

⁵Alla jordar ifrån AD-omvandlare, sensorpulser och jordning av pulståg in på optokopplare (i drivningsenheten) är sammankopplade i jorden på "PCI 6602", nummer 35 på konnektorblocket. Detta innebär att nummer 39, 27, 11, 46 och 14 på ska vara sammankopplad med nummer 35 på konnektorblocket. Övriga jordar (41, 68, 59 och 55) är redan kopplade, i enheten för sensorer och manuell styrning, till denna jord. Jordarna för IGBT:erna är kopplade, i drivningsenheten, till jorden för +12-voltspänningen.

SCLK	10
CS_INV	44
DIN	45
SSTRB	12
DOUT	13
SCLK GND	11
CS_INV GND	11
DIN GND	46
SSTRB GND	46
DOUT GND	14
GND	35
+5 V	1



Figur 11: Konnektorblocket.

2.2.5 PCI6602

Instickskortet "PCI6602" har åtta stycken räknare. Dessa räknare är programmerbara.

Vid autonom körning på båda banorna konfigureras räknarna enligt följande :

Räknare 0,4 : Drivning av bilen.

Räknare 1,5 : Detekterar pulser ifrån sensorer placerade längs med banan.

Räknare 2,6 : Detekterar pulser ifrån sensorn som indikerar nytt påbörjat varv.

Räknare 3,7 : Mäter aktuell sträcktid.

Räknare 0 upp till 3 gäller för bana 1 och räknare 4 upp till 7 gäller för bana 2.

Vid manuell körning på båda banorna konfigureras räknarna enligt följande :

Räknare 0,4 : Drivning av bilen.

Räknare 2,6 : Detekterar pulser ifrån sensorn som indikerar nytt påbörjat varv.

Räknare 3,7 : Mäter aktuell varvtid.

Räknare 0, 2 och 3 gäller för bana 1 och räknare 4, 6 och 7 gäller för bana 2.

2.3 Mjukvara

Program för att styra bilbanan från Matlab har utvecklats. De består av ett antal funktioner som kan anropas för att kommunicera med bilbanan och har implementerats i C (CVI/LabWindows). Listningar av programkoden finns i [1]

2.3.1 Funktioner för autonom körning

- `start_position(track,init_speed)`
Syfte : Köra bilen fram till startpositionen, samt konfigurerar räknare på kortet PCI6602.
Argumentet `track` : "1" för bana 1, "2" för bana 2. Argumentet `init_speed` : Antar värden mellan 0 och 100.
- `start_race(track)`
Syfte : Startar tidsräkningen, det vill säga alla räknarna börjar att räkna.
Argumentet `track` : '1' för bana 1, '2' för bana 2.
- `set_car_speed(track,speed)` Syfte : Sätta bilens hastighet.
Argumentet `track` : '1' för bana 1, '2' för bana 2.
Argumentet `speed` : Antar värden mellan 0 och 100.
- `add_lap add_check_point elapsed_time_check_point=get_car_position(track)`
Syfte : Ta reda på vilken sensor bilen passerade senast, samt hur länge sen det var.
Returvärde `add_lap` : Anger om nytt varv har påbörjats. Om så är fallet sätts parametern till '1', annars till '0'.
Returvärde `add_check_point` : Anger om en sensor längs med banan har passerats. Om så är fallet sätts parametern till '1', annars till '0'.

Returvärde `elapsed_time_check_point` : Anger aktuell sträcktid i millisekunder.

Argumentet `track` : '1' för bana 1, '2' för bana 2.

- `terminate(track)`
Syfte : Stoppa bilen och alla räknare.
Argumentet `track` : '1' för bana 1, '2' för bana 2.

2.3.2 Funktioner för manuell körning

- `activate_lap_sensor(track)`
Syfte : Konfigurera räknare för detektion för nytt påbörjat varv.
Argumentet `track` : '1' för bana 1, '2' för bana 2.
- `activate_lap_time(track)`
Syfte : Konfigurera räknare för tidtagning.
Argumentet `track` : '1' för bana 1, '2' för bana 2.
- `activate_pulse_train(track)`
Syfte : Konfigurera räknare för drivning av bilar.
Argumentet `track` : '1' för bana 1, '2' för bana 2.
- `config_IOs()`
Syfte : Konfiguera IO-pinnar.
- `manual_speed = get_manual_speed(track)`
Syfte : Erhålla manuell hastighet.
Returvärde `manual_speed` : Antar värden mellan 0 och 100.
Argumentet `track` : '1' för bana 1, '2' för bana 2.
- `add_lap elapsed_time = get_position(track)`
Syfte : Ta reda på när varvsensorn passerar.
Returvärde `add_lap` : Anger om nytt varv har påbörjats. Om så är fallet sätts parametern till 1 '1', annars till '0'.
Returvärde `elapsed_time` : Anger aktuell sträcktid i millisekunder.
Argumentet `track` : '1' för bana 1, '2' för bana 2.
- `set_manual_speed(track,manual_speed)`
Syfte : Sätta bilens manuella hastighet.
Argumentet `manual_speed` : Antar värden mellan 0 och 100.
Argumentet `track` : '1' för bana 1, '2' för bana 2.
- `terminate_manual(track)`
Syfte : Stoppa bilen och alla räknare.
Argumentet `track` : '1' för bana 1, '2' för bana 2.

2.3.3 Matlabprogram

Det finns två testprogram för Matlab

- demo.m
- person_vs_person.m

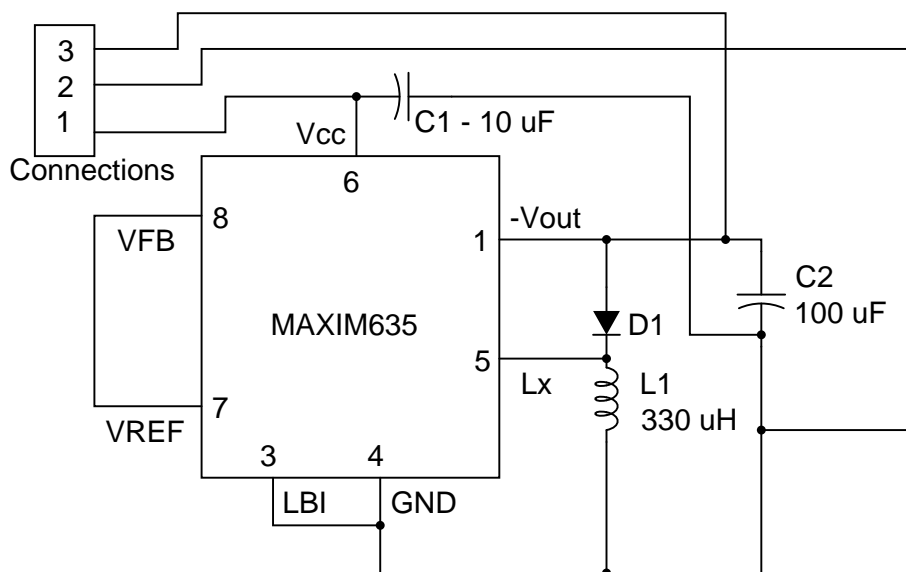
Deras syfte är att demonstrera hur Matlab kan användas för att styra bilbanan och de finns listade i [1]

3 Display

Till bilbanan finns en display av LCD typ. Dess funktion och tillhörande programkod beskrivs nedan.

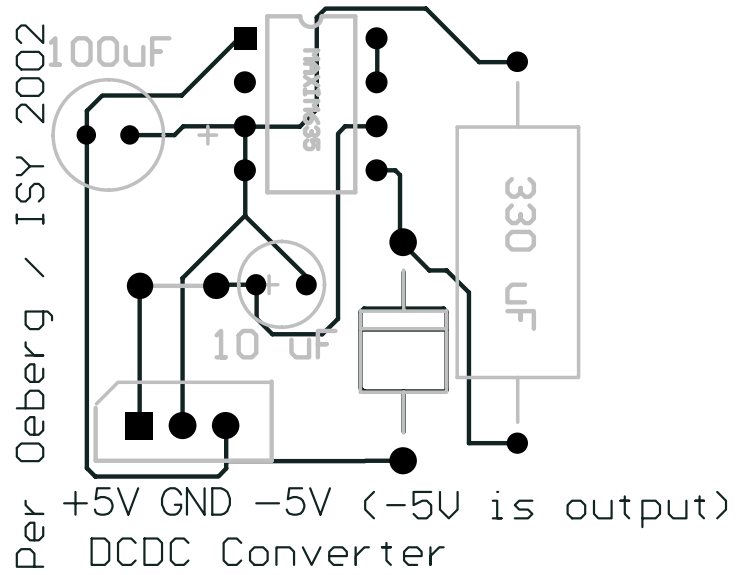
3.1 Hårdvara

Displayen behöver en negativ spänningsmatning (-5 V). Därför konstruerades en DCDC-konverter m.h.a. en krets från maxim⁶. Figur 12, 13 och 14 visar designen för DCDC-konverteraren.

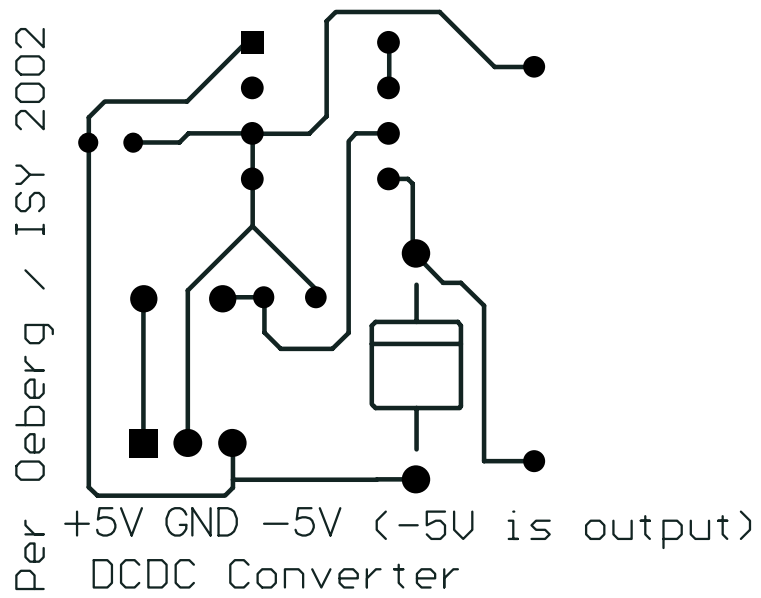


Figur 12: Krettschema för DCDC-konverterare

⁶För mer information se <http://www.maxim-ic.com>



Figur 13: Kretslayout för DCDC-koverterare



Figur 14: Etsmask för DCDC-koverterare

3.2 Mjukvara

För att styra displayen finns styrkod för Matlab. Koden finns att betrakta i [1] men nedan följer en kort redovisning av deras funktioner.

- `lcdinit.m`
Initierar LCD'n.
- `lcdclear.m`
Rensar vald halva av LCD'n.
- `lcdbacklight.m`
Ställer in bakgrundsljuset hos LCD'n.
- `lcdwrite.m`
Skriver ut en text på LCD'n.
- `lcdclose.m`
Stänger interfacet till LCD'n.

Alla funktioner har stöd för Matlabs "help" kommando för att underlätta deras användning.

3.3 Begränsningar

I nuläget finns några begränsningar hos displayen.

- LCD'n stödjer inte de svenska tecknen å,ä och ö. För att skriva dessa bokstäver kan man dock skicka deras bitmönster till LCD'n. Detta skulle innebära att man var tvungen att scanna igenom alla textsträngar efter de svenska tecknen och ersätta dem med kommandosekvenser för bitmönster.
- LCD'n stödjer inte mer än en storlek på teckensnitt. Detta går att ordna på samma sätt som med de svenska tecknen men då måste man tänka på att det inte kan bli lika snyggt eftersom det då kommer bli en glipa i skarven mellan två "rutor" som är storleken på standardteckensnittet.

Referenser

- [1] Per Öberg. Kodbilaga till 'styrning av bilbana med en dator'. ISY Fordonssystem, September 2002.
- [2] Johan Dalin. Styrning av bilbana med en dator. ISY Fordonssystem, September 2001.