

Rock Around Clock

Digitala Projekt LP3 2001
Grupp 07

Oscar Johansson, e96oj
Olof Larsson, e96ola
Erik Sylvan, e96esy

2 mars 2001



Abstract

We have designed a digital clock, displayed by a rotating row of seven Light Emitting Diodes. The reason why it's possible to experience a image of the clock is because the human brain and eyes are slow to interpret fast changes in light intensity. Therefore it seems like the clock is floating in the air.

The construction is based on a micro-controller, HC11, which is programmed to light the diodes at the correct time and in a given order. A sensor is used to give the micro-controller the lap-time and a start-position of the presentation for the clock.

Our prototype keeps the clock clear and stable. Using switches we can set and reset the current time. The clock is independent of the rotation-velocity.

Innehåll

1 Inledning	4
1.1 Bakgrund	4
1.2 Mål	4
1.3 Kravspecifikation	4
2 Utförande	5
2.1 Beskrivning	5
2.2 Blockstruktur	6
2.2.1 Varvpulsgenerator	6
2.2.2 Beräkningsmodul	6
2.2.3 Presentationsenhet	6
2.2.4 Spänningsmatning	7
2.2.5 Motor	7
3 Hårdvara	8
3.1 Enchipsdatorn HC11	8
3.2 Drivare för lysdioderna	8
3.3 IR givare/mottagare	8
3.4 Lysdioder	9
3.5 Spänningsregulator	9
4 Mjukvara	10
4.1 Programmering	10
4.2 Avbrott eller pollning?	10
4.3 Tidsinställning	10
4.4 Tid	10
4.5 Varvtid	11
4.6 Display	11
5 Resultat	12
5.1 Förbättringar	13
A Kretsschema	15
B Komponentlista	16

1 Inledning

Syftet med kursen Digitala Projekt är att varje gupp skall utveckla ett digitalt system, från en egen idé till en färdig och inte minst fungerande prototyp.

Studenten får fritt bestämma vad det digitala systemet skall utföra. Vårt första problem var just att bestämma vad vi ville bygga.

I denna rapport presenteras hur projektet har fortskridit. I avsnitt 1 ges bakgrund och mål med projektet. I avsnitt 2 ges en beskrivning om hur projektet är upplagt. En mera noggrann förklaring av projektets hårdvarudel återfinns i avsnitt 3 och mjukvarudelen finns beskriven i avsnitt 4. Sist diskuteras resultatet och möjliga förbättringar, avsnitt 5.

1.1 Bakgrund

Efter ett par öl på Stortorget här i Lund och efter tips från äldre teknologers hemsidor, bestämde vi oss för vad vi skulle ge oss på. Vi ville göra en digital klocka vilken presenteras med hjälp av en roterande kolonn av lysdioder. Nästa problem var att komma på ett bra namn på vår produkt; det blev *Rock Around Clock*, RAC.

1.2 Mål

Våra mål med projektet var att presentera en svävande ”virtuell” klocka med hjälp av en roterande motor. Klockan ska visa korrekt tid och det ska vara lätt att ändra inställningen av tiden. Bilden av klockan ska hållas stabil i sidled och ska ses tydligt.

1.3 Kravspecifikation

Detta är den kravspecifikation vi lämnade in i början av kursen:

Vårt prototyp ska kunna:

- visa aktuell tid med roterande dioder.
- presentera tiden i 24-timmarsvisning.
- nollställa tiden med en knapptryckning.
- sätta tiden som visas med två knappar.
- hålla bilden av klockan stabil i sidled.
- köras med en rotationshastighet som är över 25 rps.
- visa en skarp och flimmerfri klockupplevelse.

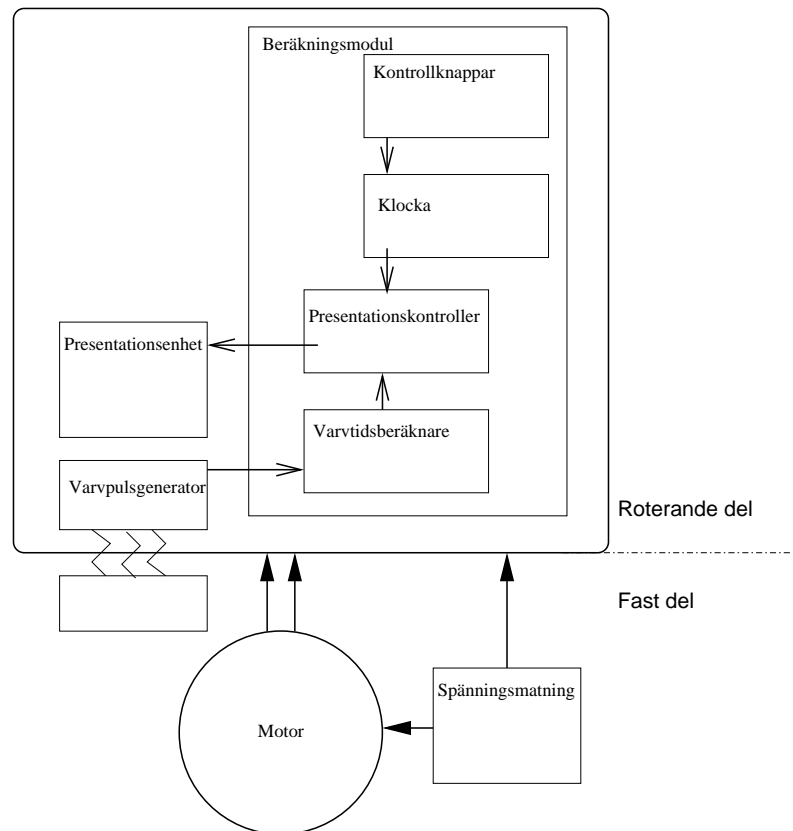
2 Utförande

2.1 Beskrivning

Displayen som skall presentera den aktuella tiden består endast av en sju lysdioder lång kolonn. Kolonnen med lysdioder sitter monterad på en motor. Dioderna tänds och släcks efter ett förutbestämt mönster när motorn roterar så att en betraktare ser en presentation av tiden när kolonnen far förbi. Att man ser alla siffror i klockslaget istället för bara en kolonn beror på en tröghet i det mänskliga synsinnets, vi uppfattar ljuset från en lysdiod efter att den släckts. För att kunna få större valmöjlighet hur presentationen skulle se ut valde vi att använda tvåfärgade lysdioder.

Enchipsdatorn som styr hela konstruktionen är en Motorola HC11. Den är liten och har många inbyggda enheter, t.ex. minne och timerkretsar, och passar därför bra till mindre inbyggda system.

För att lösa uppgiften som vi antagit oss i och med kravspecifikationen (avsnitt 1.3) var vi tvungna att bryta ner problemet i mindre delproblem. Vi tänkte igenom vilka delar av konstruktionen som gick att betrakta som enskilda enheter.



Figur 1: Blockstruktur för RAC.

2.2 Blockstruktur

RAC:en kan delas upp i fyra delar som tillsammans bildar en fungerande enhet. Delarna är; Varvpulsgenerator, Beräkningsmodul, Presentationsenhet, Spänningsmatning och motor. Beräkningsmodulen är väsentligen en enchipdator med mjukvara. Mjukvaran kan i sin tur delas upp i fyra delar; Klocka, Varvtidsberäknare, Presentationsberäknare och Kontrollknappar. Hur blocken samverkar kan ses i figur 1. Enheterna i blockstrukturen beskrivs nedan.

2.2.1 Varvpulsgenerator

Varvpulsgenerator är den del av konstruktionen som genererar en puls varje varv som motorn roterar. Pulsen behövs för två ändamål. För det första för att indikera var på varvet som presentationen av tiden skall börja. Detta är viktigt för att klockan skall vara stabil. För det andra för att kunna beräkna varvtiden för motorn. Med hjälp av varvtiden kan vi sedan beräkna under hur lång tid klockan skall visas så att klockan inte ändrar storlek beroende på hur fort motorn snurrar. Pulsen är insignal till beräkningsmodulen.

2.2.2 Beräkningsmodul

Beräkningsmodulen består väsentligen av en processor. Här sköts alla beräkningar som genererar presentationen av tiden. Beräkningsmodulen är konstruktionens hjärna.

Klocka: Klockan håller reda på den aktuella tiden. Den har insignaler från kontrollknapparna. Med dessa signaler kan klockan ställas. Minuter och timmar ställs var för sig. Utsignal är den aktuella tiden som vidarebefordras till presentationskontrollern. Klockan går även när motorn inte roterar.

Varvtidsberäknare: Varvtidsberäknarens uppgift är att beräkna varvtiden. Presentationskontrollern behöver varvtiden för att på rätt sätt kontrollera presentationen av tiden. Insignalen till varvtidsberäknaren kommer från varvpulsgeneratorn.

Presentationskontroller: Presentationskontrollern styr när lysdioderna skall vara tända eller släckta och vilken färg de skall ha. Presentationskontrollern behöver information från klockan, om vad som är den aktuella tiden, och från varvtidsberäknaren om hur lång den aktuella varvtiden är. Utsignalen går till presentationsenheten.

Kontrollknappar: Med kontrollknapparna kan man ställa klockan. Knapparna är tre till antalet, *reset*, *tim* och *min*. Varje knapp ger en positiv puls till klockan.

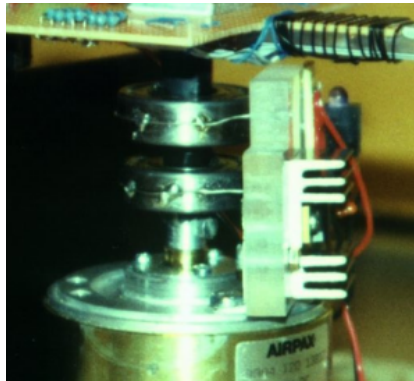
2.2.3 Presentationsenhet

Det är ju ingen större idé att ha byggt ihop allt detta utan att kunna visa upp klockan. Detta sköter en kolonn på sju stycken lysdioder. Vi valde att ha just sju på grund av att siffrorna som ska visas blir bra om de representeras av en 4×7 matris. Med 4 siffror med ett mellanrum samt ett kolon i mitten plus tre

kolonner på varje sida av klockan p.g.a. estetiska skäl blir det totalt en 27×7 matris.

2.2.4 Spänningsmatning

Ett svårt problem vi blev tvungna att lösa var spänningsförsörjning av det roterande kortet. Då en batterilösning skulle bli för tungt fick vi försöka lösa det med något annat. Lösningen blev kullager. Med dem kunde vi föra över spänning till kortet och det fungerar förvånansvärt bra. Funktionaliteten kan ses i figur 2.



Figur 2: Kullager i funktion som släpringar.

2.2.5 Motor

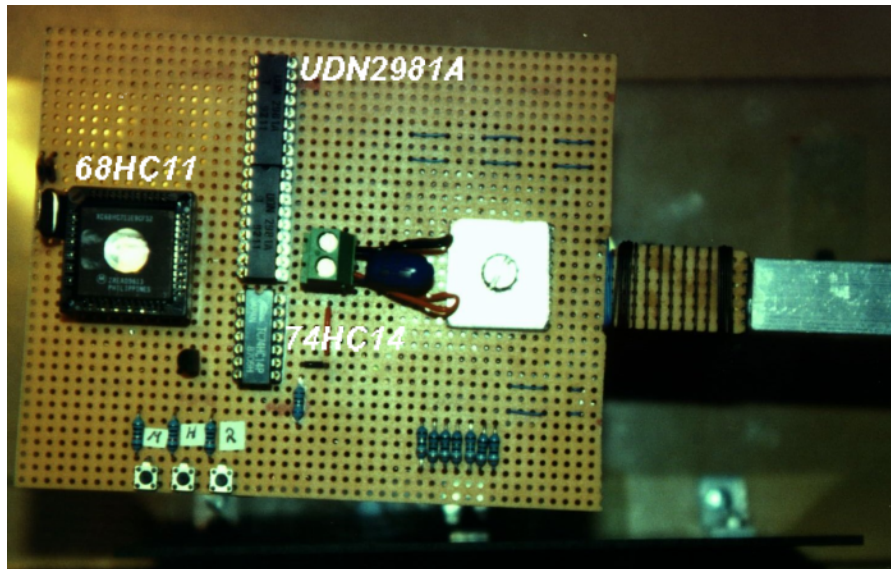
Till en början ville vi ha en motor som kunde snurra med 25 varv per sekund, eftersom vi trodde att denna höga hastighet skulle krävas för att klockan skulle synas tydligt. Detta visade sig vara väl tilltaget.

Ett annat krav på motorn är ju att den ska hålla en jämn gång. Vi hittade ingen motor på institutionen, men efter ett besök på Elektronikföreningen på LTH hittade vi en linjär, åttapolig liten likströmsmotor. Motorn är märkt med 24 V, men kan utan problem köras på lägre spänningar.

3 Hårdvara

Vi arbetade efter att använda så få komponenter som möjligt, då hela konstruktionen skall rotera. Komponenterna monterades på ett kort med hjälp av lödning och virning och hela konstruktionen fästes sedan på motorns axel.

Vissa komponenter, som t.ex. motstånd och kondensatorer, nämns inte nedan. Kretsschemat kan ses i sin helhet i appendix A.



Figur 3: Kretskortet från ovan.

3.1 Enchipsdatorn HC11

Enchipsdatorn Motorola XC68HC711E9CFS är hjärtat i vår konstruktion. Anledningen till att vi valde denna framför M68008 är att den är mindre till storleken om man jämför med alla sidokomponenter som den senare behöver.

3.2 Drivare för lysdioderna

HC11 är en mycket bra liten processor men den klarar inte av att leverera tillräckligt mycket ström till våra lysdioder. Detta löste vi med drivare, i vårt fall två UDN2981A. Kretsen har åtta ingångar, som kopplas till processorn, och åtta utgångar, vilka var och en kopplas till en lysdiod via ett motstånd.

3.3 IR givare/mottagare

I avsnitt 2.2.1 diskuteras vår varvpulsgenerator. Den består egentligen av en IR-sändare, SFH484-2, och en IR-mottagare, SFH203-FA. Sändaren är fast placerad på motorn och har en kontinuerlig ström genom sig, d.v.s. den sänder hela tiden. Mottagaren roterar med kortet, och varje varv då fotonerna från sändaren träffar mottagaren får vi en signal till processorn.

För att få en distinkt signal använder vi en schmidttrigger, TC74HC14P.

3.4 Lysdioder

Lysdioderna, TLUV5300, är tvåfärgade (röd/grön). En tvåfärgad diod är egentligen två ihopsatta dioder i en kapsel. Kapseln har tre ben, ett för jord och ett för varje färg.

3.5 Spänningsregulator

Vi tyckte efter ett tags utvecklande att det blev jobbigt att dra dubbla par sladdar till motorn och till kortet. Med en spänningsregulator, L7805CV, kunde vi styra motorn på högre spänning, men samtidigt ha konstant 5 V till kretskortet.

4 Mjukvara

Vårt mål var att i så stor utsträckning som möjligt låta mjukvaran ta hand om presentation och funktionalitet. Vi ställde oss tidigt frågan om vi skulle låta kolonnen med lysdioder tändas under ett förutbestämt tidsintervall. Fördelen med detta är att vi skulle slippa en hel del programmering, men förmodligen skulle inte displayen se bra ut. En ytterst liten spänningsändring över motorn hade medfört ett annat varvantal och därmed en display som varierar på bredden.

Med hjälp av en fast IR-sändare monterad på laborationsplattan och en IR-mottagare på rotern skulle det var möjligt att göra displayens bredd helt oberoende av varvtalet. Vi valde att göra på det senare sättet.

4.1 Programmering

Vi använde oss av Windowsprogrammet Embedded Workbench som innehåller C-kompilator, assemblerare och länkare. För att testa vår kod använde vi oss av C-SPY, en högnivådebugger som är integrerad med Embedded Workbench. C-SPY kommunicerade med vår konstruktion via en emulator. Eftersom hela vår konstruktion roterar, kunde vi bara göra enkla tester med hjälp av emulatoren.

För att kontrollera att varvtidsberäkningen gav en display med lagom bredd var vi tvungna att ladda ner all kod till en HC11. Därefter kunde vi provköra konstruktionen fullt ut.

Vi programmerade i en C-standard som heter ANSI-C. Det är en standard som är flitigt använd inom processorprogrammering. Vi behövde aldrig skriva assemblykod. Med tanke på att C-kompilatorn har en effektiv kodoptimerare är det föga troligt att vi med enbart assemblyprogrammering hade uppnått en snabbare och bättre kod än den vi skapade i C.

4.2 Avbrott eller pollning?

De enheter som kommunicerar med HC11:an är tre brytare och varvpulsgeneratorn. Brytarna sitter på den roterande delen, d.v.s. vi utnyttjar dem inte när motorn är igång. Därför vore det slöseri med exekveringstid att ta hand om brytarnas signaler med hjälp av pollning. Vi valde att ha en avbrottsrutin för var och en av dessa. Hur vi skulle ta hand om varvpulsgeneratorn signaler var inte lika självklart. Varvpulsgeneratorn genererar en puls en gång per varv. För att vara säkra på att inte missa några av dessa korta pulser krävs en intensiv pollning. Vi valde därför att även låta varvpulsgeneratorn genererar ett avbrott.

4.3 Tidsinställning

De tre brytarna, *reset*, *timme* och *minut* används för att ställa klockan. Avbrottsrutinen för reset nollställer de globala variablerna för Timme och Minut. Avbrottsrutinen för Timme respektive Minut ökar på den aktuella variabeln med ett.

4.4 Tid

Internt i HC11:an finns en räknare man inte kan påverka. Den stegar fram ett steg för varje klockpuls. Klockfrekvensen är 2 MHz. Genom att sätta ett ”Timer

Output Compare Register”, som gör jämförelser med den interna räknaren, till 20 000 får man ett avbrott var 20 000:e klockcykel vilket motsvarar 10 ms. Vi lät en variabel ökas på med ett varje gång detta avbrott genereras. På detta sätt håller vi reda på tiden.

4.5 Varvtid

Vi uppskattade varvtiden till ca 40 ms. De register som räknar klockcykler är 16 bitar långa vilket motsvarar ca. 32 ms i tid. Det räcker alltså inte med ett register för att beräkna varvtiden. För att lösa detta låter vi ”Timer Overflow” generera ett avbrott som ökar en 16-bitars variabel med ett varje gång den interna räknaren börjar om på noll. Varvpulsgeneratorn genererar ett avbrott som läser av ett ”Timer Input Capture Register”. Genom att mappa detta registers 8 MSB:s med 16-bitars variabelns 8 LSB:s får vi en tid med en upplösning av 128 μ s. Från denna tid drar vi av den tid vi fick föregående varv och en varvtid erhålles.

4.6 Display

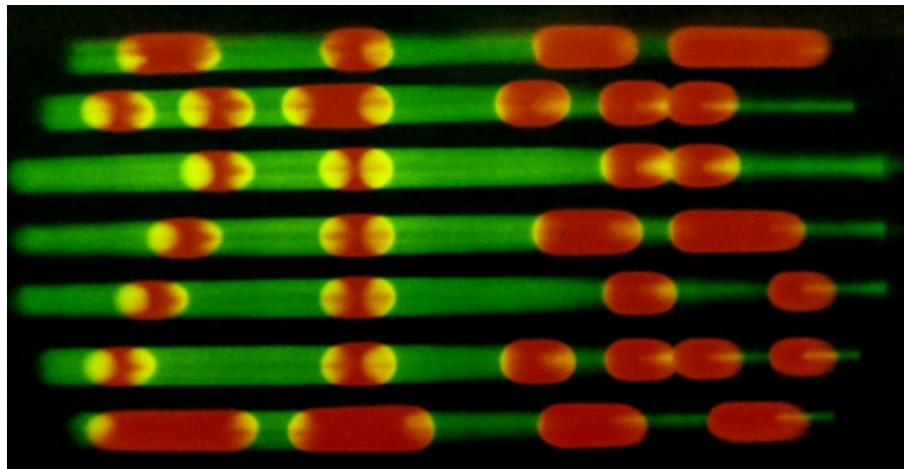
För att inte få en display som är alltför välvd, låter vi klockan presenteras på ca 30 grader utav varvet. Vi har, vilket nämnts i avsnitt 2.2.2 totalt 27 kolonner. Dessa 27 kolonner låter vi presenteras jämt över de 30 graderna av varvet. Presentationen börjar direkt efter att vi fått avbrottet från varvpulsgeneratorn. På så sätt hamnar displayen på samma ställ varje varv.

De dioder på displayen som inte visar rött sken sätts att visa grönt, för att få en annorlunda bakgrundseffekt. Vi använder oss av en vektor av bytes, där varje byte representerar en kolonn i en siffra. Vi låter varje kolonn visas under lika långa tidsintervall. Detta tidsintervall är direkt uträknat med hjälp av varvtiden.

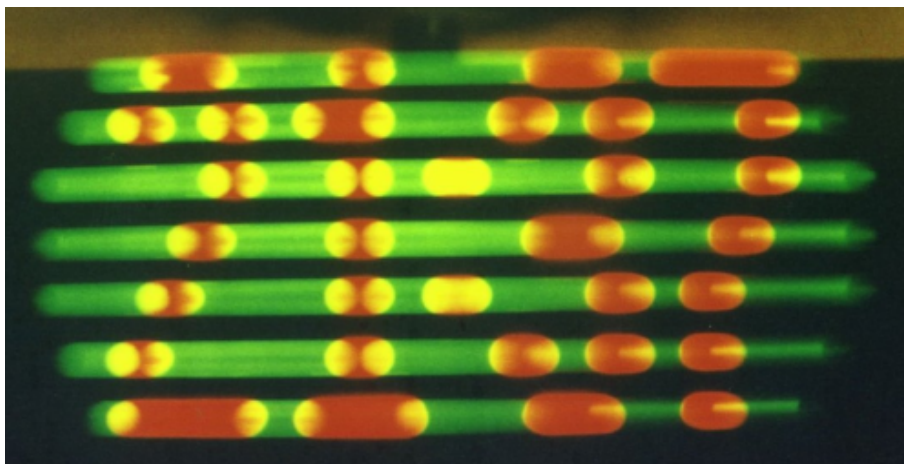
5 Resultat

Vi är mycket nöjda med vårt projekt, och är förvånade över att det gick att genomföra. Kursen "Digitala Projekt" är en trevlig och intressant kurs, då man får bryta ner ett eget valt projekt i småbitar och lösa dem. Vi gav oss på ett projekt som vissa herrar i Tekniska Gruppen tyckte var för svårt, med tanke på all den mekanik som spelar in. De problem som uppstått inom den mekaniska världen ha vi fått lösa helt själva, med undantag av viss fräsning.

Vårt antagande att 25 rps skulle krävas för en skarp bild av klockan, det räckte med ett lägre varvtal



Figur 4: RAC rocks!



Figur 5: RAC rocks II!

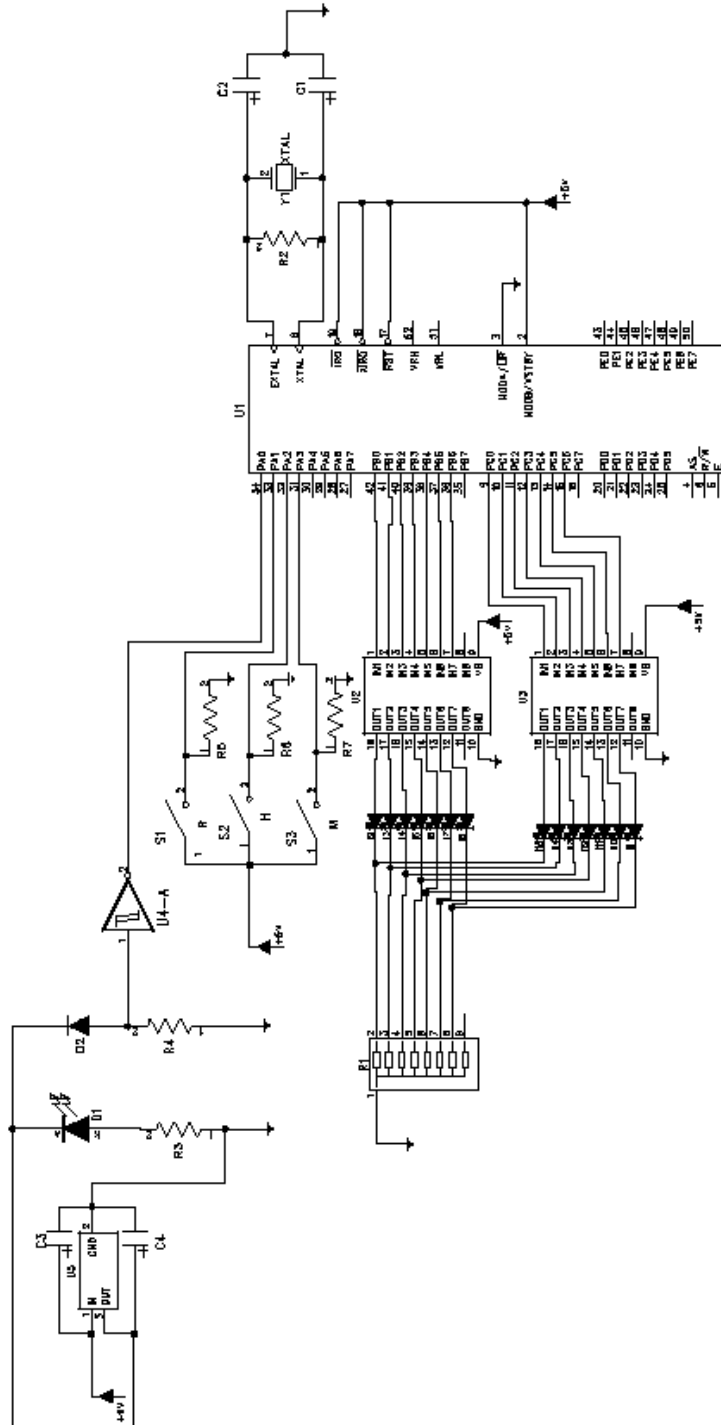
5.1 Förbättringar

- Den mekaniska konstruktionen skulle vi kunnat förbättra ytterligare. Eftersom vi har förlängt motoraxeln och den inte är optimal, leder detta till vibrationer i konstruktionen vid höga varvtal. En annan faktor som påverkar vibrationerna är kullagerna. De sitter inte centrerade på axeln.
- Klockan drar sig nån sekund per minut när motorn roterar på normal hastighet. Vi har inte lyckats lokalisera felet, men misstänker att det kan vara magnetfält från motorn som stör kristallen eller processorn. Detta har vi förgäves försökt åtgärda genom att jorda kristallens hölje och med diverse avstörningskondensatorer.
- Större intensitet på lysdioderna skulle ge oss en bättre display i dagsljus. Nu ses vår RAC bäst i mörker.

Referenser

- [1] Brian W. Kernighan, Dennis M. Ritchie. *The C Programming Language, Second Edition*. Rentice-Hall, 1988.
- [2] *M68HC11 Reference Manual*. Motorola, 1990.
- [3] *M68HC11 E Series Technical Data*. Motorola, 1995.
- [4] *M68HC11 E Series Programming Reference Guide*. Motorola.
- [5] *Digitala Projekt Datablad HC11*. IT.
- [6] *Emulator EM-11 Handledning*. IT, 1994.
- [7] Stefan Nyman. *Bygg och Programmera med HC11*. 2000.

A Krettschema



B Komponentlista

<i>Komponent</i>	<i>Modell</i>	<i>Värde</i>
C1	Kondensator	25 pF
C2	Kondensator	25 pF
C3	Kondensator	330 pF
C4	Kondensator	100 pF
D1	IR-sändare SFH484-2	
D2	IR-mottagare SFH203	
I1-I14	Lysdioder TLUV5300	
R1	Resistor $\times 7$	39 Ω
R2	Resistor	10 M Ω
R3	Resistor	22 Ω
R4	Resistor	6.8 k Ω
R5-R7	Resistor	220 Ω
S1	Brytare	
S2	Brytare	
S3	Brytare	
U1	Enchipsdator XC68HC711E9CFS	
U2	Drivare UDN2981A	
U3	Drivare UDN2981A	
U4	Schmitttrigger TC74HC14P	
U5	Spänningsregulator L7805CV	
Y1	Kristall	8.000 MHz