

# ELEKTOR Allt om **Elektronik**

Sveriges största specialtidning för tillämpad elektronik



**Programmerare  
för AT90S8535**



**Ljudtrycks-  
mätare**

**Tjuvlarm för  
motorcyklar**

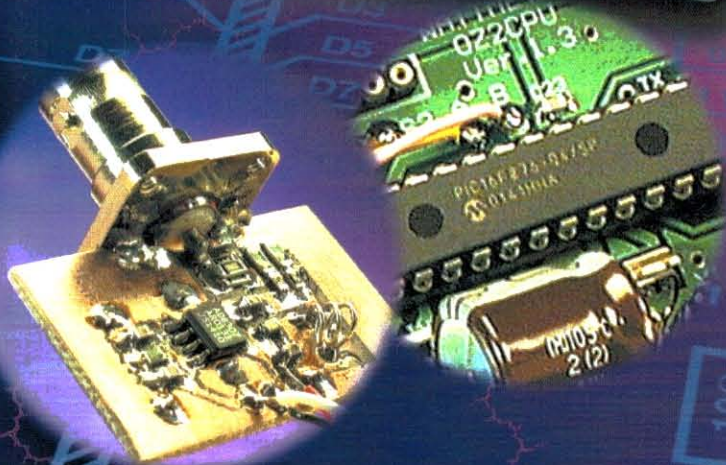
**Diagnossystem  
för bilar**

**Akustisk voltmeter**

**Programmering av  
USB-drivare**

## **1-GHz HF wattmeter med LCD-visning**

OZ2CPU DIGIWATTMETER  
Ver: 1.03 Jan. 2002



INTERPRESS 0173-09



7 388017 305909

RETURVECKA 52

**Chefredaktör och ansvarig utgivare**  
Bill Cedrum

## Redaktionen

Box 178  
444 22 Stenungsund  
Besöksadress  
Kristinedal 40  
444 47 Stenungsund

## Redaktionssekreterare

Asta Russell

## Utgivare

Electronic Press AB  
Box 178  
444 22 Stenungsund  
Tel: 0303-77 04 90  
Fax: 0303-77 04 90  
E-mail: red@alltomelektronik.com  
http://www.alltomelektronik.com

## Prenumerationer 2002

PressData AB  
Tel: 08-799 63 25  
Fax: 08-98 45 72  
Helår (11 nummer) 485:-  
Halvår (6 nummer) 285:-  
2 år (22 nummer) 875:-  
Norge helår 420:- NOK + porto  
Norge halvår 255:- NOK + porto  
Norge 2 år 755:- NOK + porto

## Annonser

Allt om Elektronik  
Annonsavdelningen  
Box 178  
444 22 Stenungsund  
Tel: 0303-77 04 90  
Fax: 0303-77 04 90  
E-mail: annons@alltomelektronik.com

## Tryck

NDB-Printing  
Zoeterwoude, Nederland 2002

ISSN 0280-8765

Samtliga artiklar, fotografier, kretskortslayouter och annat är copyrightskyddat och får endast återges efter utgivarens skriftliga tillstånd. Detta gäller även delar av materialet. Allt om Elektronik har, så långt som är möjligt, kontrollerat samtliga byggprojekt. Dock tar vi inget ansvar för fel som ligger utanför vår kontroll. Ej heller tar vi ansvar för byggen som våra läsare utför.

**Copyright:** Segment B.V.  
(Beek, the Netherlands) 2002  
P.O. Box 75  
6190 AB Beek (L), The Netherlands

## BYGGPROJEKT

### AT90C835-programmerare

Denna programmerare för Atmels processorer är lätt att bygga och mycket billig.

Inte av mycket mer än en ZIF-hållare, fyra kontakter och 32 LED. Laddarprogrammet kan hämtas gratis.



### Digital HF-wattmeter

Alla radioamatörer vet hur viktigt det är med en noggrann effektmätare för HF. En wattmeter kan

användas för att mäta förstärkning i förstärkare, bandbredd i filter, fältstyrka från antenner, sändareffekt, SWR, returförluster och mycket annat.



### Ljudtrycksmätare

Det mätinstrument som beskrivs här skiljer sig i ett avseende från alla andra instrument av samma sort, man kan utan överdrift säga att det inte går att missa. Detta öppnar för helt nya möjligheter för denna applikation.



## Tjuvlarm för motorcyklar

Detta larmsystem använder två PIC16F84 processorer och är ganska lätt att bygga. Priset är mycket lågt jämfört med färdiga system, men effektiviteten finns det absolut inget att anmärka på!



56

## INFORMATIVA ARTIKLAR

### Diagnossystem för fordon

Diagnossystem för fordon har fram till nu varit tillverkarspecifika och i allmänhet konstruerade på så sätt att endast auktoriserad verkstads-personal kunnat komma åt data.

Detta håller nu på att förändras med tillkomsten av det standardiserade OBD-2 systemet, som har blivit en EU-standard (EOBD) för alla nya bilar. I denna artikel beskriver vi detta interface i detalj och i en kommande artikel visar vi en EOBD diagnos-adapter som du kan bygga själv.



50

## Byggprojekt

- 6 Tjuvlarm för motorcykel
- 12 Ljudtrycksmätare
- 24 Akustisk voltmeter
- 32 Höghastighets  
kontrollerkort
- 40 Digital HF-wattmeter
- 47 Programmerare för  
Atmel AT90S8535

## Informativa artiklar

- 20 Programmering av  
USB-drivare
- 26 System för fordonsdiagnos

## Ovrigt

- 31 Läsarservice
- 50 Nästa nummer

# Tjuvlarm för motorcyklar

med dubbla sensorer, klåfingerskydd och fjärrkontroll

Design av G. Visschers

Detta larmsystem använder två PIC16F84 processorer och är ganska lätt att bygga. Priset är mycket lågt jämfört med färdiga system, men effektiviteten finns det absolut inget att anmärka på!

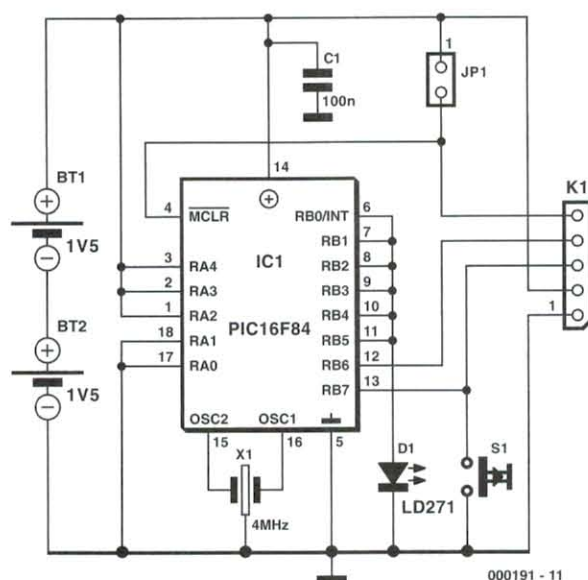


Motorcykel och scooterstölder ökar hela tiden, och speciellt utomlands arbetar stora ligan med att stjäla vissa märken på beställning. Högt upp på listorna ligger exklusiva modeller av Ducati och Harley-Davidson, men även de dyrare modellerna av BMW finns med samt 250 000 kronors Honda Goldwing.

Men inte heller ägare till inte fullt så exklusiva modeller är immuna mot stöld utan ett stort antal cyklar ur denna kategori stjäls också varje år. Det är sorgligt men sant. Polisen försöker självklart så gott de kan, men om du är riktigt rädd om

din tvåhjulning så är det nödvändigt att ha ett bra larmsystem.

Tyvärr är ett pålitligt larm allting annat än billigt. Ett bra larm kostar från 3000:- och uppåt, och till detta kommer installationskostnaden. Därför tvekar många mc-ägare och förlitar



**Figur 1. Fjärrkontrollen består inte av mycket mer än en programmerad PIC-kontroller och en IR LED.**

sig på att det räcker med ett kraftigt lås.

Naturligtvis går det att hitta system som kostar mindre, men det lägre priset betyder ofta lägre säkerhet. Och det är i detta läge som en elektronikingenjör börjar att tänka på ett hembyggt system. Ett välbeskrivet larm behöver inte vara speciellt komplext ur elektronisk synvinkel och om du bygger och installerar själv så har du den fördelen att larmet kan anpassas precis till motorcykeln eller scootern ifråga.

Den största fördelen är dock priset. Det är inte mer än en fjärdedel av de tidigare nämnda 3000:-!

## Krav

Vilka krav skall vi då ställa på ett tillförlitligt larm för en motorcykel eller en scooter?

I huvudsak skall följande fyra punkter uppfyllas:

- Kretsen skall ha en känslighets-sensor som reagerar för varje rörelse. De infraröda och ultraljudssensorer som används hemma och i billarm är inte bra för motorcyklar. Här behöver vi en typ av sensor som kan upptäcka rörelser eller vibrationer mekaniskt.
- En tydlig optisk och akustisk larm-signal hör självklart till. Detta är

inte heller speciellt svårt att implementera.

- Larmet skall naturligtvis vara väl skyddat mot klåfingrighet. Detta är inte så enkelt och kräver en del eftertanke.
- Och naturligtvis är det bara ägaren som skall kunna sätta på och stänga av larmet med dess rätta kod, helst med en fjärrkontroll.

## Designöversikt

Hur ser nu designen ut när vi nu gör praktiskt bruk av vår önskelista och hur implementeras den?

Till att börja med så har vi använt två olika typer av rörelsesensorer. Det kan kanske verka 'overkill' men det kostar ganska lite och det är bäst att vara på den säkra sidan. Ett par kvicksilverswitchar av standardtyp detekterar om motorcykeln ändrar läge och på toppen av detta har vi lagt till en modifierad piezotransducer som en vibrationssensor. Denna är speciellt effektiv och kan göras så känslig att det räcker med att knacka lite lätt med fingret på motorcykeln för att larmet skall utlösas. Och sedan finns då klåfingerskyddet. Detta består också av två sektioner. Ett LDR (ljusberoende motstånd) triggar larmet när någon med skumma avsiker lyfter på sadeln eller öppnar det utrymme som larmet

byggts in i. En extra säkring används så att larmet fungerar även om någon skulle kortsluta matningen till korriktionsvisarna (blinkersen).

När larmet triggas börjar blinkersen på motorcykeln att blinka och en kompakt, men väldigt ljudlig, piezosirén drar igång. En liten infraröd sändare styr larmet och behöver naturligtvis en matchande mottagarmodul som plockar upp signalen.

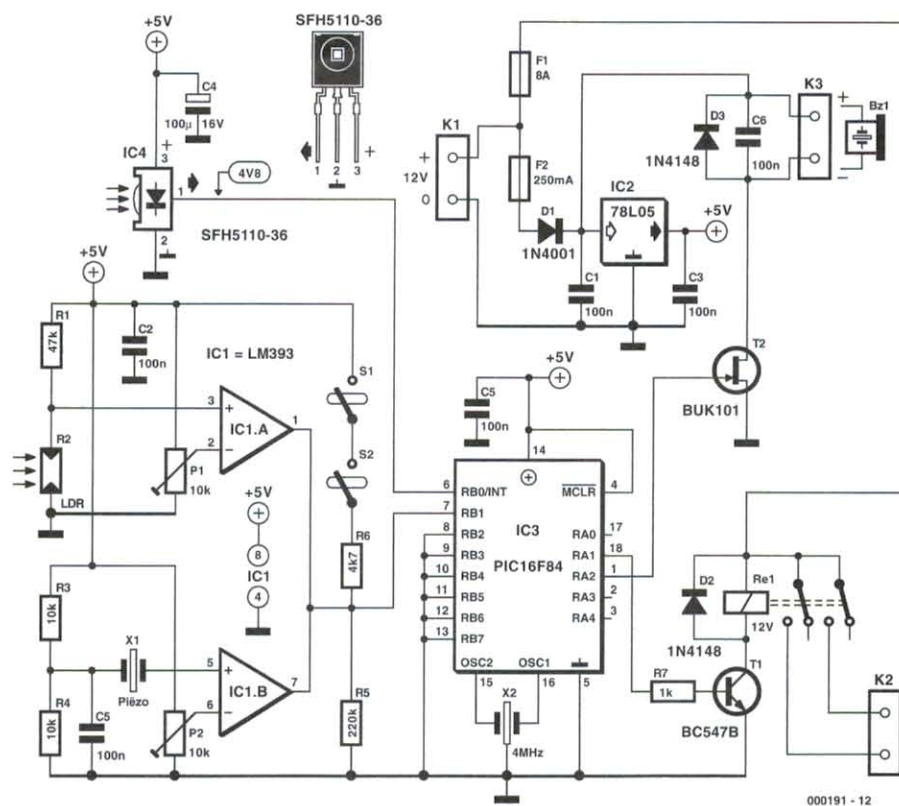
I hjärtat av kretsen sitter en PIC16F84 som innehåller ett enkelt program. Nu är det säkert flera läsare som undrar om inte denna krets kan klara sig utan en programmerad mikrokontroller. Och självklart hade detta varit möjligt. Men kretsen skall utföra många funktioner, signalerna från sensorerna skall tolkas rätt, IR-signalen från fjärrkontrollen skall tas emot och sätta på eller stänga av larmet, det skall genereras styrsignaler för blinkers och siréner och sedan skall larmet stängas av efter 20 sekunder för att vara lagligt. Det skall då gå tillbaka till larmläge. Och slutligen så får larmet bara gå igång tre gånger efter det att det aktiverats. Detta är en lång lista och skulle kräva omfattande elektronik om ingen mikrokontroller används. Och det finns inte mycket ledigt utrymme i en motorcykel för en massa extra elektronik.

Eftersom utrymmet är ännu viktigare när det gäller fjärrkontrollen så har vi här tacksamt använt oss av de tjänster som erbjuds av en PIC16F84, denna gång i sin ytmonterade version. Denna kontroller har ett lätt jobb då den endast skall producera den rätta pulssade signalen på 36 kHz för lysdioden i IR-sändaren. Båda PIC-kretsarna finns att köpa färdigprogrammerade från oss, men om någon skulle vilja programmera dem själva så finns käll- och hexfiler att hämta gratis från vår webbsida. Titta under Download, nummer 9/2002, 000191-11.

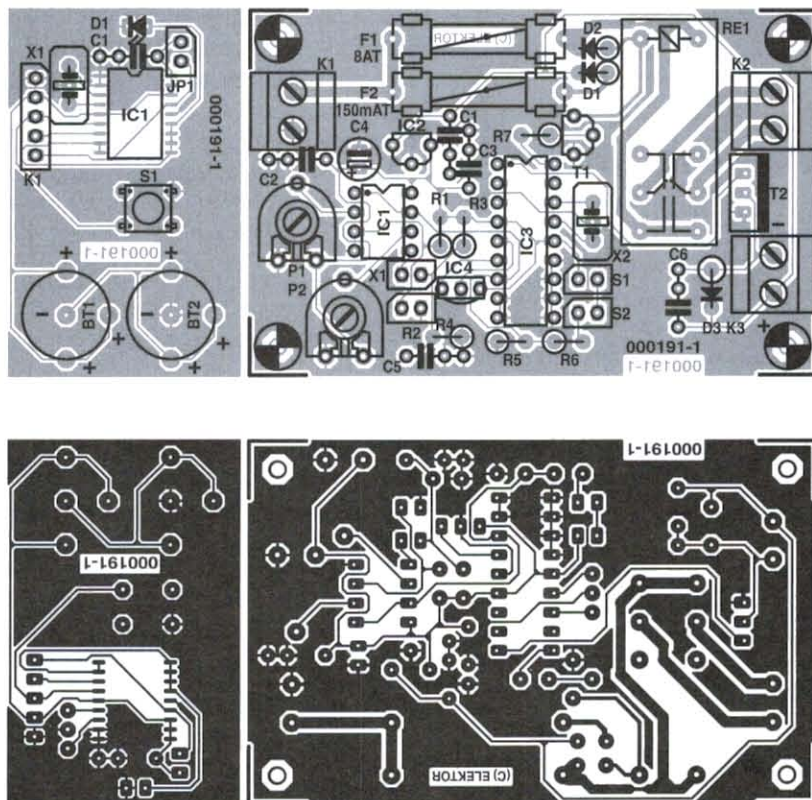
## IR-sändare i miniatyr

Nu är det dags att ta en närmare titt på kretsschemat. Vi börjar med fjärrkontrollen eftersom denna innehåller bara några få komponenter förutom den lilla PIC-kretsen. Schemat visas i figur 1. Resonatoren X1 tar hand om klocksignalen på 4 MHz, D1 är den infraröda sändardioden och S1 fungerar som startswitch. Kontakten K7 har lagts till för programmering av PIC-en eftersom detta annars kan vara ganska besvärligt med en SMD-kapsel. Innan PIC-en kan programmeras skall bygeln JP1 tas bort.

Funktionen hos det program som lagrats i IC1 är mycket enkelt. När S1 trycks in vaknar PIC-en ur sitt 'sovläge' och därefter skickar den ut koden som en serie pulser via utgångarna RB0 till RB5. PIC-en går sedan tillbaka till



Figur 2. En programmerad PIC hittas också i larmets hjärta. Till vänster är IR-mottagaren och sensorerna, till höger finns larmindikatorerna.



Figur 3. Sändaren och larmet kommer som ett enda kretskort som skall separeras innan du börjar bygga.

sovläget igen. IR-koden skickas till mottagaren i larmet av D1. Skälet till parallellkopplingen RB0-RB5 är att ge tillräckligt med ström för att driva LED-en.

Även om matningsspänningen till PIC16F84 skall vara minst 5 V enligt databladet, så verkar det i praktiken som om den kan fungera tillförlitligt ner till 2,8 V. Därför matas kretsen av två seriekopplade knappceller på vardera 1,5 V. Då strömförbrukningen i sovläge är försumbar räcker ett par knappceller minst ett år.

## Larmkretsen i detalj

Kretsschemat för det egentliga larmet visas i figur 2. Här ser man klart att användningen av PIC-kontrollern har gjort kretsen mycket enkel.

## KOMPONENTLISTA Sändare

### Kondensatorer:

C1 = 100nF

### Halvledare:

D1 = IR LED, t.ex., CQW13  
(Conrad Electronics # 184551-89)

IC1 = PIC16F84-04/SO,  
programmerad, ordernummer  
000191-41

### Övrigt:

JP1 = 2-pol stiftlist med bygel

K1 = 5-pol stiftlist

S1 = miniatyrtryckknapp

X1 = 3-bens 4-MHz keramisk  
resonator

BT1, BT2 = 1.5V knappcell med  
hållare för kretskort 16x1.6 mm  
(t.ex., CRI616)

Låda: Teko 11121 (Conrad  
Electronics # 543287)

Kretskort, se under 'Larm'

## KOMPONENTLISTA Larm

### Motstånd:

R1 = 47kΩ

R2 = LDR

R4, R4 = 10kΩ

R5 = 220kΩ

R6 = 4kΩ

R7 = 1kΩ

P1, P2 = 10kΩ

Mikrokontrollern hittar du i kretsens hjärta som IC3. Förutom den resonator på 4 MHz (X2) som ger klocksignalen så är det inte mycket att tillägga eftersom mjukvaran i kretsen utför alla uppgifter. Och vilka dessa uppgifter är har vi redan gått igenom. Värt att notera är att två (oanvända) utgångar kan användas för felsökning: RA0 visar larmets status ('0' i standby-läge, '1' i larmat läge) och RA3 blir en '1' när en IR-signal detekteras.

Den integrerade IR-mottagaren IC4, som hittas överst till vänster i kretsschemat, detekterar signalen från sändaren. Denna skall naturligtvis monteras på så sätt att den kan 'ses' av sändaren och skall anslutas till kretskortet via en kort bit kabel. Den mottagna signalen skickas till ingång RB0 hos PIC-en, som validerar

koden och växlar statusen hos larmet. 'Standby' ändras alltså till 'larmad' och nästa tryckning på sändarknappen växlar 'larmad' till 'standby'. Två blink på korriktionsvisarna indikerar varje förändring av statusen. När larmet går tillbaka till standby visar antalet extra blinkningar hur många gånger som larmet aktiverats under den larmade perioden.

Under IC4 i kretsschemat hittar vi sensorsektionen som består av kvicksilverswitcharna (S1/S2) och två komparatorer (IC1a/b), som tar utgångarna från LDR-en respektive piezotransducern. Kretsen har designats på så sätt att ingång RB1 går låg när det är ett larmaläge. När motorcykeln flyttas blir S1 och/eller S2 en öppen krets och RB1 dras låg av R6. När det faller ljus på LDR R2 eller när piezotransducern X1 producerar en spänning på grund av vibrationer börjar öppen-kollektorutgångarna hos IC1a respektive IC1b att leda, vilket också drar RB1 låg. Känsligheten hos LDR-en kan justeras med P1 och den hos piezotransducern med P2.

När larmet triggas kopplar utgången RA2 på FET T2, som i sin tur aktiverar den kraftfulla piezosirénen som är ansluten till K3. Den FET som används för T2 är en typ som kan drivas direkt av en utgång från en TTL-kompatibel IC och som kan switcha upp till 5 A. Eftersom sirénen endast behöver 140 mA arbetar FET-en klart inom sina gränser. Vi har valt en FET i stället för ett relä eftersom de kostar ungefär lika mycket

men en FET tar upp betydligt mindre plats.

I ett larmaläge aktiverar RA1 relät Re1 via T1, vilket ger ström till de blinkers som är kopplade till K2. Vid varje förändring av larmstatusen (från standby till larmad och tvärtom) skickar RTA1 ut en kort signal och blinkersen blinkar två gånger (plus det antal gånger larmet aktiverats). D2 och D3 undertrycker eventuella inducerade spänningar.

Strömförsörjningen består av bara en 5 V regulator (IC2) som får sin ingångsspänning från motorcykelbatteriet via kontakten K1. D1 förhindrar skador om matningen kopplas in med omvänd polaritet. Som vi redan nämnt har två säkringar använts för att förbättra det skydd som larmet ger. Om en företagsam tjuv skulle komma på att öppna en blinkers och kortsluta matningen kommer F1 att brinna men F2 påverkas inte. Kretsen fungerar fortfarande och larmet uppmärksammas tack vare sirénen.

Och nu en liten försäkring när det gäller strömförsörjningen. Hela kretsen drar inte mer än 12 mA, både i standby och larmaläge. Det finns därför ingen risk för att batteriet skall laddas ur på kort tid.

#### Kondensatorer:

C1, C2, C3, C5, C6 = 100nF  
C4 = 100µF 16V radial

#### Halvledare:

D1 = 1N4001  
D2, D3 = 1N4148  
T1 = BC547  
T2 = BUK101  
IC1 = LM393  
IC2 = 78L05  
IC3 = PIC16F84-04/P,  
programmerad, ordernummer  
000191-42  
IC4 = SFH5110-36

#### Övrigt:

K1, K2, K3 = 2-pol kopplingsplint  
för kretskort, raster 5mm  
RE1 = 12V relä, 2 slutande  
kontakter, 8A, (t.ex., Schrack  
RT424012)  
S1, S2 = kvaicksilverswitch  
(Conrad Electronics # 700452)  
BZ1 = 12V sirén, t.ex., Alecto  
X1 = piezoelement (Conrad  
Electronics # 712930)  
X2 = 3-bens 4-MHz resonator  
F1 = säkring, 8A(T) (trög), med  
hållare  
F2 = säkring, 250mA(T) (trög),  
med hållare  
Kretskort (larm & sändare):  
ordernummer 000191-I (se  
Läsarservice)  
Diskett, med källkodfiler,  
ordernummer 000191-II eller  
gratis nedladdning

## Två kretsar på ett kort

Eftersom larmkretsen och fjärrkontrollen alltid kompletterar varandra så tillhandahålls de på ett kretskort (figur 3). Detta skall delas i två delar innan du bestyckar dem. Det ger dig två kompakta kort, där sändaren är ungefär lika stor som en liten tändsticksask.

Sändarkortet omfattar bara några få kretsar och borde inte ta lång tid att bygga. Det är endast den ytmonterade IC-kretsen som kräver noggrann lödning och en lödpenna med mycket fin spets. För nybörjare är det enklast att placera kretsen exakt rätt på kor-

## Tillfälle

I nummer 9 och 10/2000 presenterade vi GBDSO, Gameboy Digital Sampling Oscilloscope, en digital oscilloskopmodul som kan användas tillsammans med ett Gameboy. Detta blev mycket populärt och de färdigbyggda modulerna såldes slut mycket fort. Vi har nu fått några stycken till av dessa och säljer dem för specialpriset 750:-/st, inkl. moms. Mjukvara ingår. Porto tillkommer. Vi skickar också med artikelkopior ur nummer 9 och 10/2000. Vi har bara ett begränsat antal av dessa moduler och först till kvarn...

Märk beställningen 'GBDSO'.

Allt om Elektronik  
Box 178  
444 22 Stenungsund  
Tel och fax: 0303-770 490  
email: order@alltomelektronik.com

tet och fästa den med en droppe lim. Löd därefter försiktigt ett ben, kolla läget igen, och löd sedan fast resten av benen. Överflödigt tenn tas bort med en avlödningsfläta. Kolla noga att det inte finns någon kortslutning mellan benen! Om du inte kan få tag på rätta hållare för battericellerna kan du göra egna hållare av kraftig förtent koppartråd. Detta fungerar lika bra. Knappcellerna skall placeras i sina hållare med plussidan uppåt. Bygget av huvudkortet borde inte heller vålla några problem. Komponentplaceringsritningen i figur 3 är tillräckligt tydlig och de olika anslutningarna är tydligt markerade. När det gäller de 'normala' komponenterna bör det inte vara några problem. Det mesta jobbet med huvudkortet handlar om de komponenter som inte sitter på kortet utan externt och som ansluts till kortet via korta kablar. Men detta kommer vi till lite senare.

## Installationen

Sändarkortet är så litet att det lätt skulle ha fått plats i en liten tändsticksask, inklusive knappcellerna. Men det finns bättre och stadigare lådor som har designats speciellt för fjärrkontroller. För prototypen använde vi en låda från Teko, figur 4. I framkanten skall du borra ett litet hål för D1. I lådans lock finns det redan ett uttag, med en knapp som passar exakt ovanför switchen S1. Att montera in kortet i denna låda tar inte mer än en kvart. Därefter kommer larmkortet. Hur det skall installeras beror till stor del på hur mycket plats det finns på motorcykeln. Normalt finns det plats under bönpallen, men detta utrymme varierar från modell till modell. Det är också möjligt att klara sig utan låda helt och hållet och i stället använda en stor bit krympslang runt hela kortet som isolering och sedan placera det direkt i cykelns verktygslåda. Vi har använt en standardlåda för vår prototyp (figur 5) och vi har några viktiga anmärkningar när det gäller bygget och ledningsdragnin- gen.

Det är självklart viktigt att kolla att polariteten hos matningsspänningen, IR-mottagaren (IC4) och sirénen är korrekt. I stället för den SFH5110-36 mottagare som används här är det också möjligt att använda SFH505. Den enda skillnaden är att anslutningarna är vridna 180 grader, så var uppmärksam! När du ansluter blinkersen till K2 skall du se till att vänster och höger blinker förblir isolerade från varandra. I annat fall kommer båda att blinka under normala förhållanden.

IC4 skall naturligtvis monteras på så sätt att IR-signalen från sändaren kan nå den. Du kan göra ett litet hål för detta bak på cykeln. Om mottagaren sitter något nedsänkt så 'ser' den inte signaler som inte är avsedda för den. En



Figur 4. Denna lilla låda passar perfekt för sändarkortet.

tvåtråders skärmad (audio) kabeln skall användas för att ansluta IC4 till kortet. Därmed minimeras eventuella störningar.

Kvicksilverkontakterna (S1 och S2) skall monteras på så sätt att de är slutna när motorcykeln är parkerad i sitt normala läge (på sidostöd eller



Figur 5. En titt på insidan av prototypen. Hur larmet skall installeras beror till stor del på typen av motorcykel eller scooter.

mittstöd). Det kan vara lite knepigt att hitta det bästa läget för dessa. LDR R2 har monterats på sidan av lådan i vår prototyp. Det är viktigt att det monteras på så sätt att det 'ser ljuset' när sadeln lyfts eller när det utrymme som larmet finns i öppnas. Och sedan är det bara monteringen av piezotransducern (X1) kvar.

## Vibrationssensor

Den (billiga) piezotransducer som används här har visat sig vara en mycket effektiv vibrationssensor, i så hög grad att du faktiskt skulle kunna utelämna kvicksilverswitcharna helt och hållet. Den behövde dock en liten modifiering. Transducern är limmad på lådans sida (limma bara efter kanten så att den aktiva delen kan röra sig fritt!). En liten vikt på ca 60 gram limmas sedan på andra sidan. Trögheten hos denna vikt ökar den deformation av piezokristallen som orsakas av vibrationerna. Detta ökar känsligheten så mycket att det kan anses vara den viktigaste delen av larmet. En 2,5 cm lång M10-skruv fungerade mycket bra för oss. Om du vill hålla allting prydligt så kan du först tillverka en plathållare för piezotransducern med hjälp av en borrh och en såg. Denna kan sedan monteras i larmlådan med två skruvar, se figur 6. Eftersom vi experi-

## Ändra koden?

I praktiken är det naturligtvis inte särskilt säkert om alla som bygger detta larm använder sig av samma infraröda kod för att sätta på eller stänga av larmet. För de av er som verkar hur man programmerar, eller har någon som kan hjälpa till, kan detta lätt ändras. Du kan lätt byta ut den fasta koden till en som är personlig. För detta betyder du en assembler och en PIC-programmerare.

Källkodsfilerna för båda PIC-kontrollerna kan du hämta gratis från vår websida vid adressen ([www.alltomelektronik.com](http://www.alltomelektronik.com)). Filen heter **000191-11.zip** och koderna för de färdigprogrammerade PIC-kretsarna är 000191-41 och 000191-42.

I båda filerna skall du leta efter de rader som innehåller 'code 1' och 'code 2'. De aktuella värdena på 'F2' och '45' kan sedan ändras till vilket värde som helt inom området '00' till 'FF'. Se till att båda kontrollerna innehåller samma värde! De modifierade programmen assembleras sedan och bränns in i PIC-kontrollern med en programmerare.

En lämplig gör-det-självp programmerare kan hittas på sidan 48 i nummer 11/2000 av Allt om Elektronik.

menterade en hel del med prototypen så bestämde vi oss för denna praktiska lösning.

## Justeringar

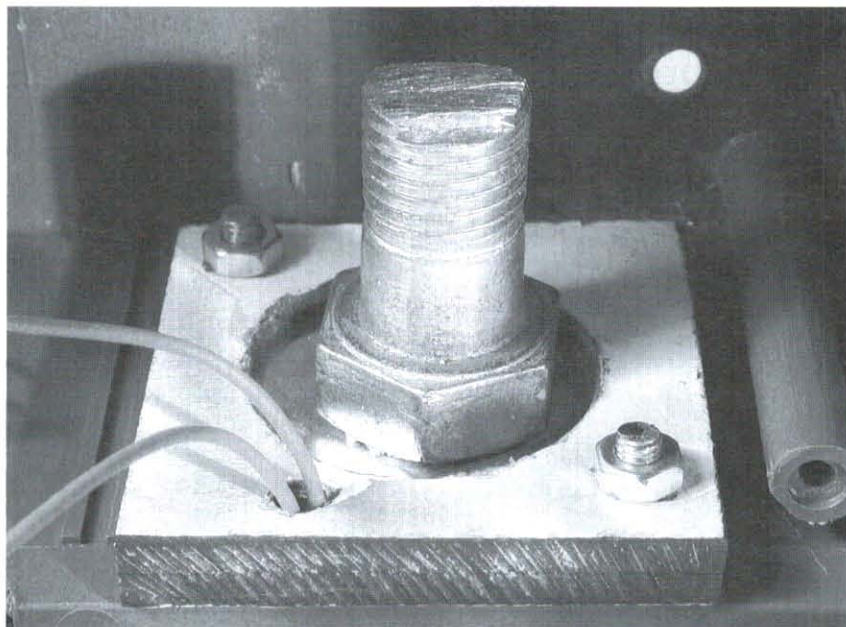
När du har byggt kretsen, och när sensorerna, körriktningsvisarna och sirénen har anslutits, skall du först vrida P1 och P2 helt moturs innan du lägger på strömmen från batteriet via K1.

När strömmen kopplas in första gången befinner sig larmsystemet i sitt

standbyläge. När korrekt kod från fjärrkontrollen tas emot av PIC16F84, och verifieras, blinkar körriktningsvisarna två gånger och larmet är på. Du kan verifiera detta tillstånd genom att kolla utgången RA0 hos IC3, denna skall nu vara hög.

Nu vrider du P1 och P2 försiktigt medurs tills den önskade känsligheten har ställts in. Se till att du inte vrider P2 för långt eftersom piezotransducern kan göras så känslig att larmet nästan triggas bara du tittar på motorcykeln eller scootern!

(000191-1)



**Figur 6.** Piezotransducern förses med den rekommenderade motvikten och limmas sedan in i en plastram.

# Ljudtrycksmätare (SPL)

## Modell King-size

Design av T. Giesberts

Det mätinstrument som beskrivs här skiljer sig i ett avseende från alla andra instrument av samma sort, man kan utan överdrift säga att det inte går att missa.

Detta öppnar för helt nya möjligheter för denna applikation.

Den mest framträdande egenskapen hos denna krets är att displayen består av en panel som tillverkats av tio glödlampor. Detta fångar omedelbart ögat och samtidigt står det helt klart att vi här har att göra med ett nytt slags mätinstrument. En typisk ljudtrycks-

eller VU-meter använder sig av betydligt mer beskedliga LED eller ett vridspoleinstrument.

Och vad är då meningen med denna enorma display? Är den bara till för skoj skull? Både ja och nej. Uppen-

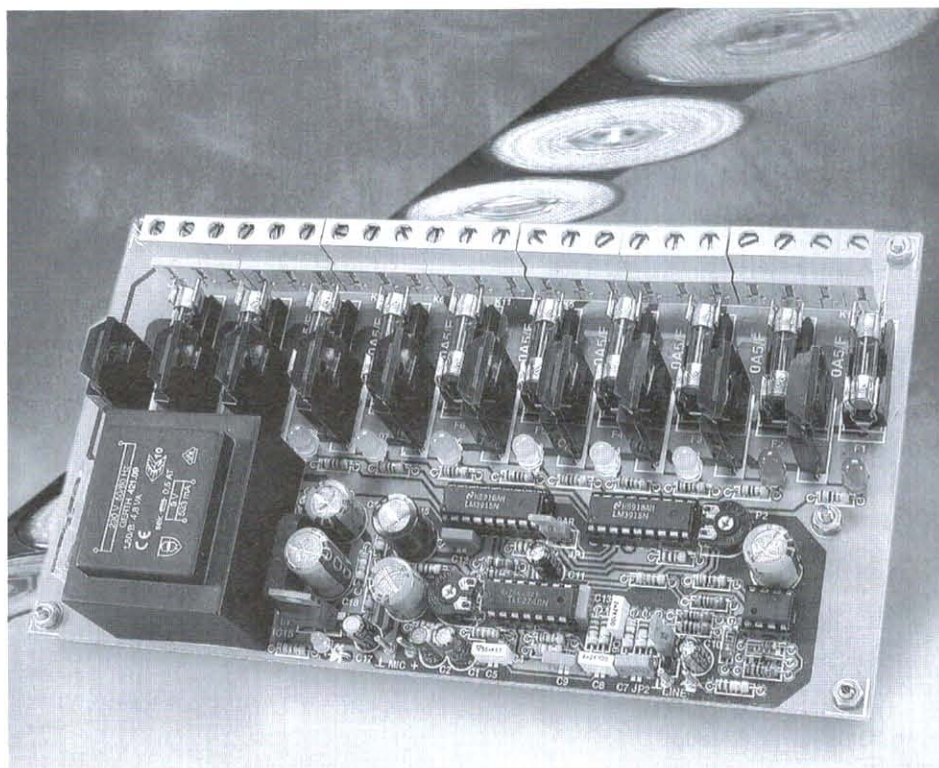
bart är att en sådan här display tigger om att få användas på ett disko eller ett party där den kan fungera som en ljusshow och samtidigt en optisk inställning av musiken. Vi tror att detta projekt kan bli mycket populärt i dessa kretsar.

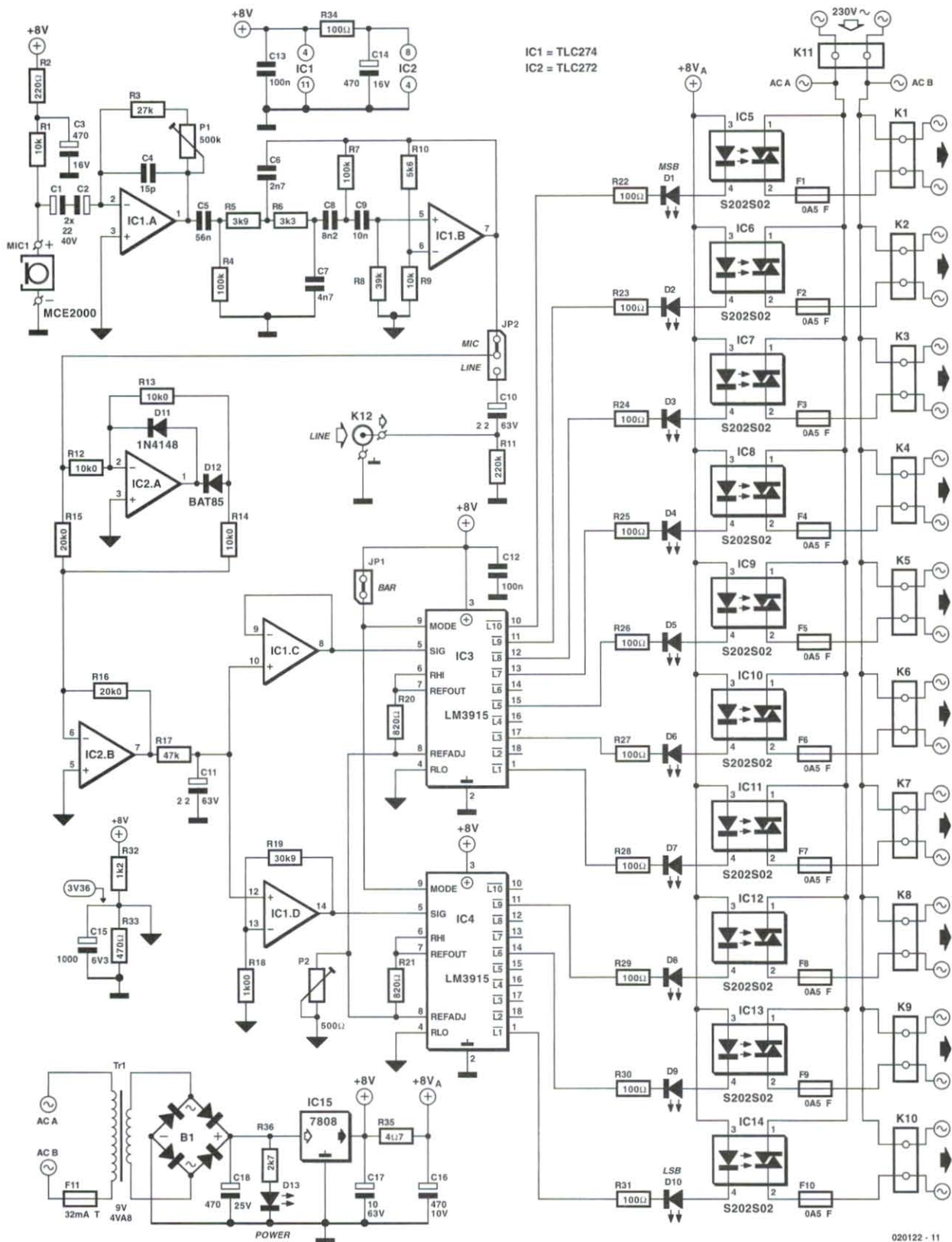
Men detta är inte för detta ändamål som detta stycke elektronik har designats från början. Den viktigaste applikationen som vi hade i åtanke var att ge en tydlig indikering av ljudtrycksnivån i verkstäder, diskotek, stora feshallar etc., alltså platser där man under längre tid kan utsättas för höga ljudnivåer, vilka kan leda till hörselskador.

### Designen

Designen av ljudtrycksmätaren (SPL, sound pressure level) är inte speciellt krånglig. Vid sidan av en mikrofon så finns det inte mycket mer än ett förstärkare, en likriktare och en display. Dessutom, för att anpassa displayen till området för den mänskliga hörseln är det önskvärt att även ha ett A-vägt filter, och i vårt fall måste vi också lägga till lampdrivare.

En snabb titt på kretsen i figur 1 visar tydligt hur kretsen designats. Ljudet plockas upp med hjälp av en elektret-





Figur 1. Kretsen omfattar en mikrofonförstärkare, ett filter, en likriktare och en display. Den vanliga LED-skalan har här här utökats med glödlampsdrivare.

mikrofon (MIC1), förstärks (IC1a) och filtreras (IC1b). Detta följs av en helvågslikriktare (IC2) varefter signalen genomsnittas (R17/C11) och buffras/förstärks (IC1c/IC1d). Signalen förs därefter till två displaydrivare av typ LM3915 (IC3/IC4), vilka har designats att driva LED-indikatorer direkt. Var och en av de totalt tio lysdioderna är seriekopplad med en optotriac (IC5 till IC14). Det är optotriacens uppgift att switcha de lampor som är anslutna till K1-K10. Känsligheten kan justeras med P1. Genom att flytta JP2 till den andra positionen är det möjligt att ersätta mikrofonsignalen med en linjesignal som kan anslutas till K12. Detta alternativ kan användas när kretsen används som ljusshow i PA-installationer. Om mätaren används på detta sätt kan mikrofonen, IC1 samt de omgivande komponenterna utelämnas.

Nu har vi spridit lite ljus över kretsens viktigaste delar. Men som vanligt finns det ett antal detaljer vilka är värda att nämnas. Så låt oss ta en närmare titt på de olika delarna i kretsen.

## Mikrofonförstärkare

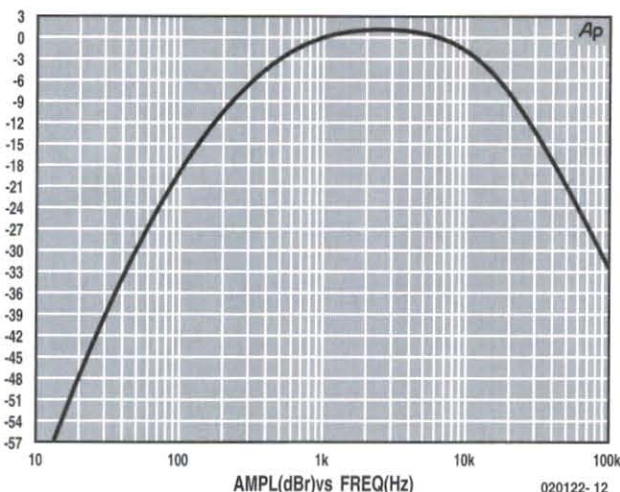
Den mikrofon som vi valt för detta projekt är en liten elektret från Monacor med typnummer MCE-2000. Andra jämförbara mikrofoner kommer antagligen att fungera utan problem de också.

Elektretmikrofonen tar sin matningsspänning via R1. R2/C3 undertrycker eventuella störningar som kan finnas på matningslinjen. Utan filtret skulle dessa störningar förstärkas avsevärt av IC1a. Förstärkningen av mikrofonsignalen definieras av förhållandet mellan utgångsimpedansen hos elektretkapseln och R1, såväl som av värdet på P1 plus R3. Värdet på motståndet R3 har valts på så sätt att justeringsområdet för P1 blir ungefär 26 dB.

När man gör absoluta ljudtrycksmätningar bär det viktigt att veta vilka kalibreringsnivåerna är. När förstärkningen hos IC1a är satt till max med P1, och P2 är satt till '0', motsvarar fullt skalutslag en nivå på ca 90 dB. Detta är ett mycket medvetet val eftersom detta ljudtryck markerar början på den farliga zonen för hörselskador.

Några ytterligare detaljer. Utgången (FET) från elektretkapseln hålls vid en konstant spänning via C1/C2. Detta förbättrar drivområdet och gör det möjligt att använda ett något högre värde på R1. Operationsförstärkaren (opampen) IC1a arbetar därför som en ström-till-spänningsomvandlare. C4 eliminerar eventuella HF-störningar vid högsta förstärkning och begränsar bandbredden till ungefär 20 kHz.

Alternativet att flytta bygeln JP2 till den andra positionen och ansluta en linjesignal till K12 har vi redan diskuterat. Mikrofonen och förstärkaren för denna ignoreras då.



**Figur 2.** Överföringsfunktionen hos det A-vägdta filtret har sitt max vid en frekvens runt 3 kHz för att likna den mänskliga hörseln.

## Filter

Det så kallade 'A-vägdta' filtret används både för ljudmätningar och inom audiotekniken när man mäter signal-till-brusförhållande. Huvuduppgiften för detta filter är att modifiera mätspektrat så att det motsvarar egenskaperna hos den mänskliga hörseln. Örats känslighet är som maximal runt 3 kHz och faller snabbt över och under denna frekvens. Det A-vägdta filtret är därför designat på så sätt att dess max också sker vid ca 3 kHz.

Figur 2 visar filtrets överföringsfunktion. För att förenkla konstruktionen av filtret har vi bara använt komponenter ur E12-serien. Trots detta har vi kunnat matcha överföringsfunktionerna till inom  $\pm 1$  dB av den teoretiska modellen.

## Likriktare

För att göra helvågsliktaren har vi använt en standardkrets med två opampar. Den 'negativa' signalen från halvågslikriktaren IC2a förstärks med en faktor två av IC2b och läggs ihop med ingångssignalen för att skapa en helvågslikriktare. En Schottkydiod har använts för D12 eftersom denna har en lägre tröskelspanning, vilket ger bättre linjäritet och en större bandbredd. Nackdelen är att dessa dioder har en högre läckström och av detta skäl har en normal diod använts för D11.

R17 och C11 tar ett genomsnitt av den likriktade signalen med en tids-

konstant på 0,1 sekund. Detta kan betyda att displayen reagerar lite för snabbt, men det ger en bättre indikering av topparna. Tidskonstanten kan ökas, om du så vill, genom att öka värdet på C11 (det finns förmodligen utrymme på kretskortet för en liten elektrolytkondensator med värdet 100  $\mu$ F/10 V).

## Displaydrivare

För att kunna driva displayen har vi använt två IC-kretsar av typ LM3915. Trogna läsare av *Allt om Elektronik* känner mycket väl igen denna krets. Om du inte känner till den kan vi berätta att LM3915 designades att visa en varierande ingångssignal på en LED-stapel. Internt har IC:n tio komparatorer vilkas inverterande ingångar är hopkopplade till utgången hos en intern buffert/förstärkare. Var och en av de icke-inverterande ingångarna är anslutna till en avtappning på en stegformad spänningsdelare, som genererar de olika referensspänningarna. Följande gäller för varje komparator: Så fort som spänningen på den inverterande ingången är högre än referensspänningen på den icke-inverterande ingången går utgången 'låg' och motsvarande LED tänds. Varför behöver vi då två LM3915 när det skulle ha räckt med en IC för att driva 10 LED? Det finns två skäl till detta. Det första är att en enkel likriktare inte är särskilt linjär i den nedre delen av området och detta kan korrigeras genom att välja specifika utgångar.

Det andra skälet är att vi vill att kretsen också skall visa ljudtryck med låga nivåer. Tabell 1 visar förhållandet mellan det teoretiska värdet och det verkliga uppmätta värdet på vår prototyp (mätt vid 1 kHz). Tabell visar också att ljudtrycksmätaren har ett område på 48 dB, vilket i de allra flesta fall är fullt tillräckligt.

Att sträcka ut skalan över två LM3915 är möjligt genom att förstärka det uppmätta värdet på C11 med extra 30 dB för IC4. När det gäller allting annat är justeringarna för IC3 och IC4 lika. En enkel virtuell jord har skapats med R32/R33/C15 för hela kretsen. Detta sänker också referensvärdet (RLO) för spänningsdelarna i båda IC-kretsarna. Denna spänningsnivå har valts på sätt att, vid full drivning, kommer utgången från IC2b att mättas symmetriskt. Toppen av spänningsdelarna (RHI) är ansluten till LM3915-ns interna referens (REFOUT). Det är möjligt att, på grund av små variationer mellan in-

dividuella komponenter (toleranser) att skalorna inte stämmer exakt överens. Det spelar dock ingen roll i denna applikation.

Känsligheten hos IC3 och IC4 är som störst när spänningsnivån vid REF-ADJ är lika med den virtuella jorden (P2 vid 0  $\Omega$ ). Detta ben har anslutits till båda IC-kretsarna så att denna nivå kan justeras med en potentiometer. Det är möjligt, till exempel, att kalibrera kretsen med P2 till en 2-V fullskalesignal (utgången från en CD-spelare eller liknande). Den interna referensspänningen är då typiskt 1,28 V.

Med en ren sinusvåg är detta som en känslighet på  $1.42 V_{\text{rms}} (1.28 \times \pi / 2\sqrt{2})$ .

## Lampdrivare

Triggerströmmen för de optotriacar som vi använt här är 8 mA minimum och 50 mA maximum. För att skapa en bekväm arbetsmarginal har vi satt triggströmmen till 16 mA och

den är därför lite högre än vad som är nödvändigt ( $10 \times 1,28 \text{ V/R20}$ ). För att göra det enklare att kolla så är en LED med ett seriemotstånd också ansluten till varje triggeringång och det är på så sätt möjligt att se om drivningen av optotriacarna är korrekt utan att behöva koppla in några lampor. Motstånden är i huvudsak avsedda för att begränsa förlusterna i LM3915, men de gör det också möjligt att kolla triggerströmmen.

En trevlig egenskap hos LM3915 är att du via ben 9 kan välja mellan stapel eller punktvisning. Detta gör du genom att växla JP1. När två LM3915 är kaskadkopplade, och punktmod har valts, är det normalt nödvändigt att använda ett trick för att stänga av MSB-LED (mest signifikanta bit) i den 'lägre' LM3915. Men eftersom denna utgång inte används så är det inte nödvändigt här.

## Praktiska saker

Det kretskort som tagits fram för ljudtrycksmätaren visas i figur 3. Eftersom det finns tio uppsättningar skruvplintar för lamporna och samma antal säkringar och optotriacar så är kortet ganska stort. Det är dock inte ohanterligt stort. Om man tittar närmare efter så finns det inte alltför många komponenter och eftersom layouten är mycket enkel så är detta ett projekt som lämpar sig att bygga även för nybörjare. Om du håller dig till komponentplaceringsritningen och komponentlistan så är det inte troligt att det blir några problem.

Du skall börja med att montera de lägre komponenterna och sedan fortsätta med de högre. Avsluta med att montera skruvplintarna, optotriacarna och transformatorn. IC-kretsarna skall monteras i hållare.

Kretskortslayouten har designats för elektrisk isolering av Klass-II och även optotriacarna klarar denna standard enkelt (4000 Vrms iso-

## Ljudtryck

Vi har nu ett fint mätinstrument, men vilka kriterier skall vi använda när vi gör mätningarna? Vad är normalt ljudtryck och när är det risk för hörselskador? Tabellen nedan visar ett antal typiska värden i dB(A). Blanda inte ihop smärtröskeln med den nivå där det kan uppstå hörselskador. Denna nivå ligger typiskt på 80 dB(A). Arbetsgivare är enligt lag skyldiga att tillhandahålla hörselskydd när ljudtrycket kontinuerligt ligger ovanför denna nivå på en arbetsplats. Vid högre nivåer ökar risken för hörselskador snabbt. Vid ett ljudtryck på 90 dB kan skador uppstå efter 1 timma, vid 95 dB redan efter 15 minuter. Tabellen visar tydligt att för de som besöker rockkonserter är det definitivt ingen lyx att förse sig med någon form av hörselskydd!

|           |                      |
|-----------|----------------------|
| 120 dB(A) | smärtröskel          |
| 110 dB(A) | rockkonsert live     |
| 100 dB(A) | cirkelsåg            |
| 90 dB(A)  | passerande tåg       |
| 80 dB(A)  | motorvägstrafik      |
| 70 dB(A)  | dammsugare           |
| 60 dB(A)  | normal konversation  |
| 50 dB(A)  | lugn gata            |
| 40 dB(A)  | tyst rum             |
| 30 dB(A)  | bibliotek            |
| 20 dB(A)  | prasslande löv       |
| 0 dB(A)   | tröskel för att höra |

Med den krets som beskrivs här motsvarar 'full skala' en nivå på 90 dB(A) när P1 är satt till maximal förstärkning och P2 satt till 0  $\Omega$ . Om du vill ändra detta, eller om du har en annan mikrofonskapsel, måste instrumentets känslighet justeras med hjälp av en kalibrerad ljudtrycksmätare.

## Tabell 1

Förhållande mellan teoretiska och verkliga ljudtrycksnivåer.

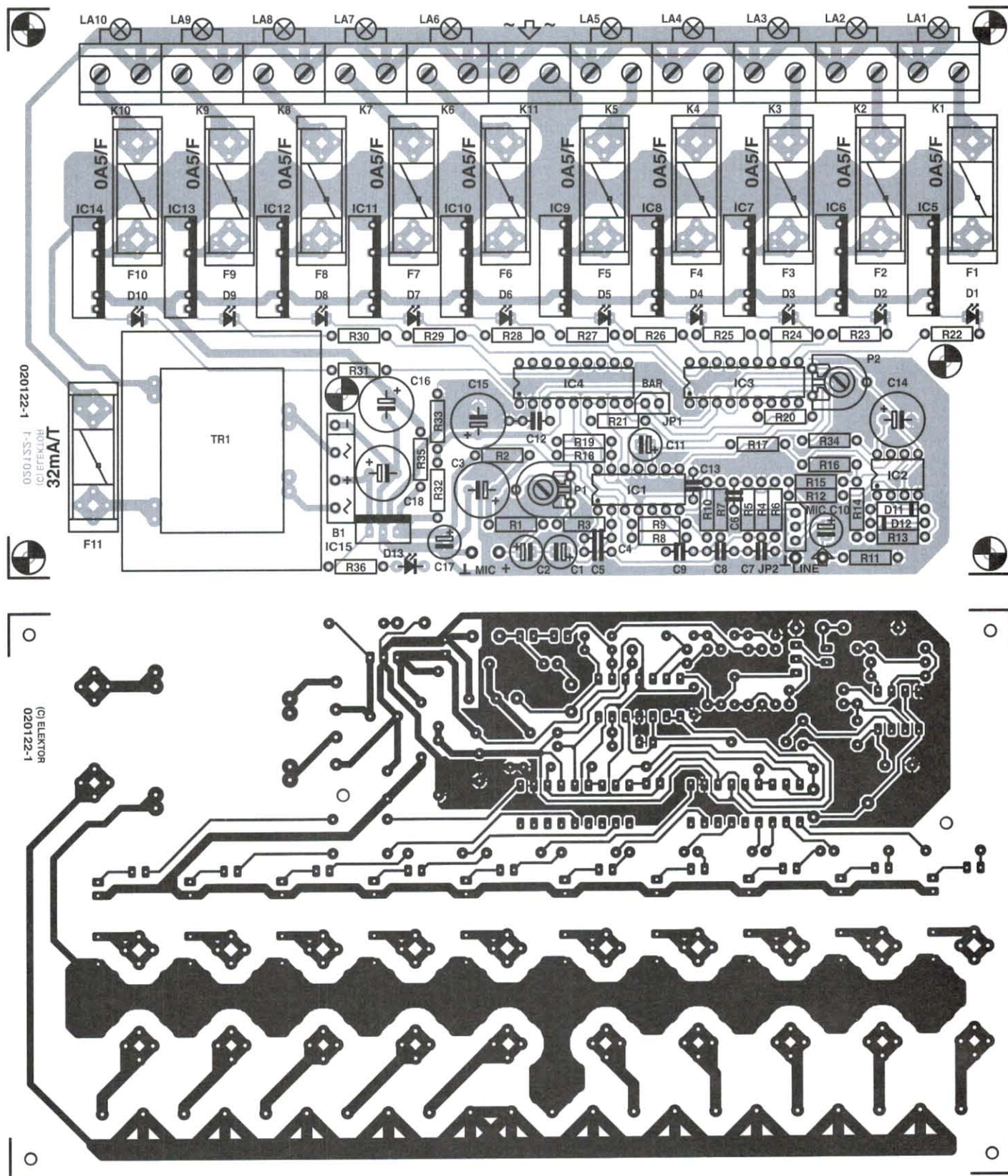
| LED | teoretisk<br>[dB] | uppmätt<br>[dB] |     |
|-----|-------------------|-----------------|-----|
| D1  | 0                 | 0               | MSB |
| D2  | -3                | -3              |     |
| D3  | -6                | -6              |     |
| D4  | -9                | -9              |     |
| D5  | -15               | -16             |     |
| D6  | -21               | -22             |     |
| D7  | -27               | -28             |     |
| D8  | -33               | -33             |     |
| D9  | -42               | -40             |     |
| D10 | -57               | -48             | LSB |

lering!).

För att förenkla bygget så mycket som möjligt så är kretsens nät-del, inklusive transformator, också monterad på kortet. Som man kan

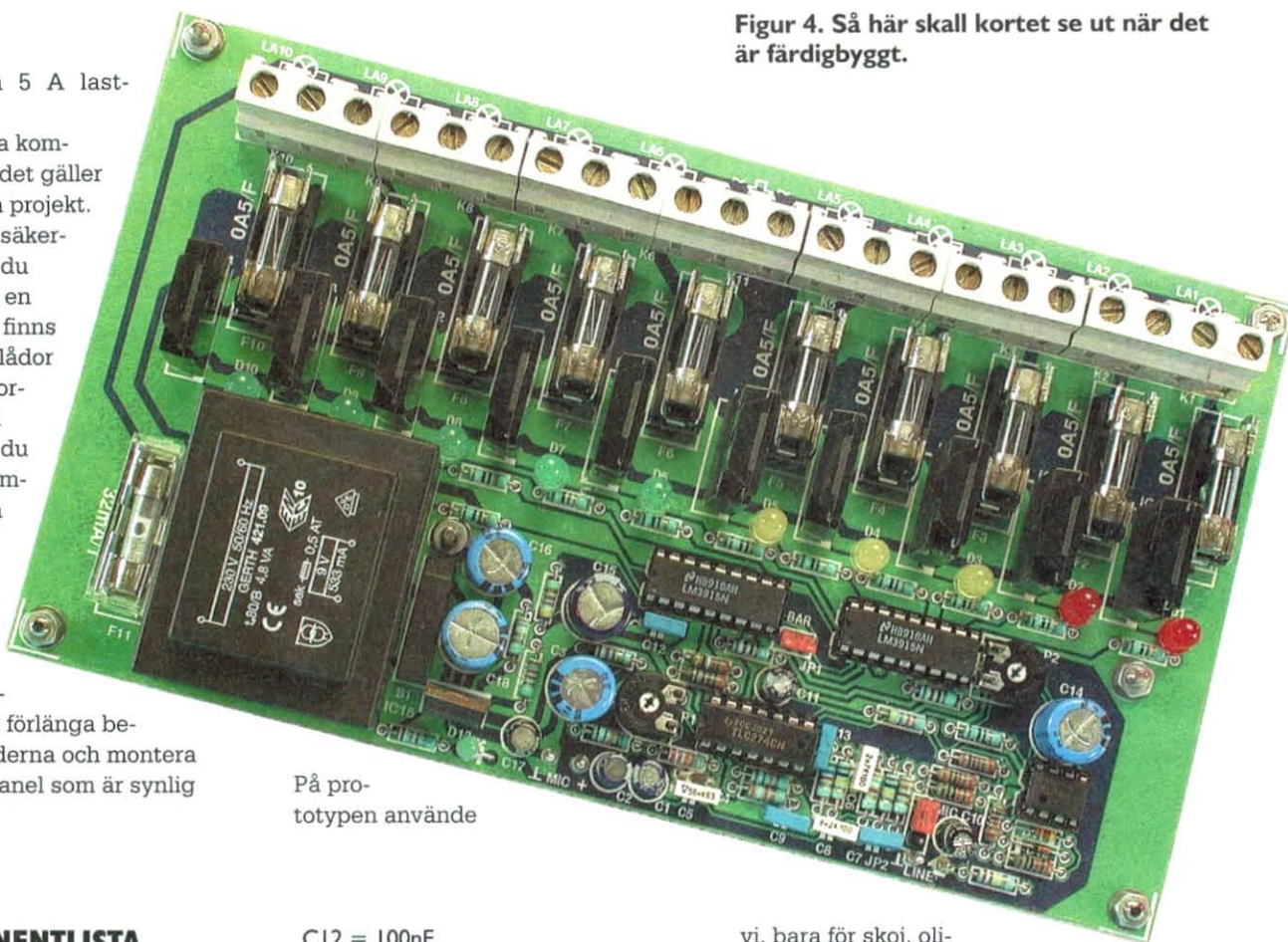
se av schemat i figur 1 följer nät-delen standardreceptet med en 4,8-VA transformator, likriktarbrygga, filterkondensator och en 8-V regulator,

IC15. LED D13 fungerar som till/frånindikator. För att förhindra eventuella EMI-problem så rekommenderar vi att du använder en stort nätfil-



Figur 3. Kretskortet domineras av de 10 optotriacarna, säkringarna och anslutningarna för lamporna.

ter (minimum 5 A lastmärke!). Slutligen några kommentarer när det gäller lådan för detta projekt. Med tanke på säkerheten så skall du helst använda en plastlåda. Det finns olika typer av lådor som passar storleken på detta kretskort. Om du väljer en genomskinlig låda så kan du se lysdioderna utifrån. Om du använder en låda som inte är genomskinlig kan du förlänga benen på lysdioderna och montera dessa på en panel som är synlig från utsidan.



Figur 4. Så här skall kortet se ut när det är färdigbyggt.

På prototypen använde

## KOMPONENTLISTA

### Motstånd:

R1, R9 = 10k $\Omega$   
 R2 = 220 $\Omega$   
 R3 = 27k $\Omega$   
 R4, R7 = 100k $\Omega$   
 R5 = 3k $\Omega$   
 R6 = 3k $\Omega$   
 R8 = 39k $\Omega$   
 R10 = 5k $\Omega$   
 R11 = 220k $\Omega$   
 R12, R13, R14 = 10k $\Omega$   
 R15, R16 = 20k $\Omega$   
 R17 = 47k $\Omega$   
 R18 = 1k $\Omega$   
 R19 = 30k $\Omega$   
 R20, R21 = 820 $\Omega$   
 R22-R31, R34 = 100 $\Omega$   
 R32 = 1k $\Omega$   
 R33 = 470 $\Omega$   
 R35 = 4 $\Omega$   
 R36 = 2k $\Omega$   
 P1 = 500k $\Omega$  trimpot  
 P2 = 500 $\Omega$  trimpot

### Kondensatorer:

C1, C2 = 22 $\mu$ F 40V radial  
 C3, C14 = 470 $\mu$ F 16V radial  
 C4 = 15pF  
 C5 = 56nF, raster 5mm  
 C6 = 2n7, raster 5mm  
 C7 = 4n7, raster 5mm  
 C8 = 8n2, raster 5mm  
 C9 = 10 n, raster 5mm  
 C10, C11 = 2 $\mu$ F2 63V radial

C12 = 100nF  
 C13 = 100nF, raster 5mm  
 C15 = 1000 $\mu$ F 6V3 radial  
 C16 = 470 $\mu$ F 10V radial  
 C17 = 10 $\mu$ F 63V radial  
 C18 = 470 $\mu$ F 25V radial

### Halvledare:

D1-D10 = LED, röd (ej typ dagsljus)  
 D11 = 1N4148  
 D12 = BAT85  
 D13 = LED, grön (dagsljus)  
 IC1 = TLC274  
 IC2 = TLC272  
 IC3, IC4 = LM3915  
 IC5...IC14 = S202S02 (Sharp)  
 IC15 = 7808

### Övrigt:

JP1 = 2-pol stiftlist med bygel  
 JP2 = 3-pol stiftlist med bygel  
 K1-K11 = 2-pol kopplingsplint för kretskort, raster 7.5 mm  
 K12 = cinch-kontakt för panelmontering  
 Tr1 = 9V 4VA8 t.ex., Gerth 421.09  
 B1 = B80C1500 rektangulär (80V piv, 1.5A topp)  
 F1-F10 = säkring, 500mA, snabb, med hållare för kretskort  
 MIC1 = MCE2000 (Monacor/Monarch) eller liknande  
 F11 = säkring, 32 mA, trög, med hållare för kretskort  
 Kretskort 020122-1 (se Läsaerservice

vi, bara för skoj, olika färger på lamporna, röda för D1 och D2, gula för D3 till D5 och gröna för D6 till D10.

Var uppmärksam när du monterar och ledningsdrar kretsen in i en låda så att det inte uppstår några kortslutningar. Förse också lådan med en lämplig ingångskontakt för nätkabeln eller förse denna med en dragavlastare. För att du inte skall behöva leta efter skruvplintarna för nätspänningen så kan vi tala om att dessa sitter precis i mitten bland lampkontaktarna.

Lådan måste vara väl ventilerad om kretsen skall användas kontinuerligt vid full signal och med max last ansluten till varje utgång (max 100 W för varje lampa!). Detta beror på att IC5-IC14 ger ifrån sig en hel del värme när de måste switcha större effekter.

(020122-1)

# Programmering av USB-drivare (I)

## en utvecklingsguide för gör-det-självare

Av M. Müller och C. Ehmer

Det finns många fördelar med att ansluta hårdvara via den universella seriebussen (USB), men det finns också många fallgropar att undvika. I denna artikelserie förser vi dig med en karta som hjälper dig att hitta den rätta vägen vid utvecklingen av USB-applikationer.

Hembyggd hårdvara för PC-datorer ansluts normalt via RS232-porten (serieporten). Efter att ha använts under flera decennier är detta interface både väl dokumenterat och enkelt att använda. Det finns naturligtvis också vis-

sa nackdelar. En av dessa är att du bara kan ansluta en enhet till porten innan du startar din PC och en annan nackdel är att strömförsörjning av en enhet som anslutits via serie-

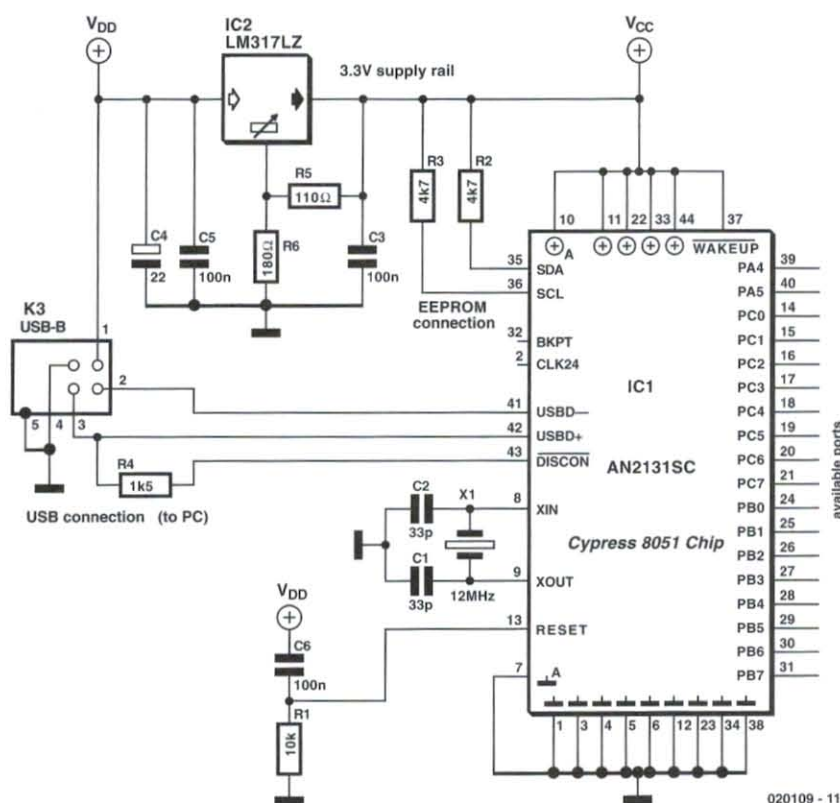
porten är krångligt (och du kan endast dra en liten effekt på detta sätt). USB-interfacet har ingen av dessa nackdelar. Emellertid vet de flesta användare, inklusive erfarna PC-hobbyister, mycket lite, eller ingenting, om USB-interfacet.

Avsikten med denna artikelserie är att du skall kunna ansluta hembyggda USB-enheter till en PC. Nyckelfaktorn här är att nästan allting du behöver för att kunna göra detta kan du ladda ner gratis från Internet. Det finns tre element vid utvecklingen av en perifer USB-krets:

1. Hårdvaran och ett program som körs i USB IC:n.
2. Ett program som kommunicerar med USB IC:n och som visar kommunikationen på en skärm.
3. En enhetsdrivare (SYS-fil) som ger kommunikationen mellan USB-porten och programmet.

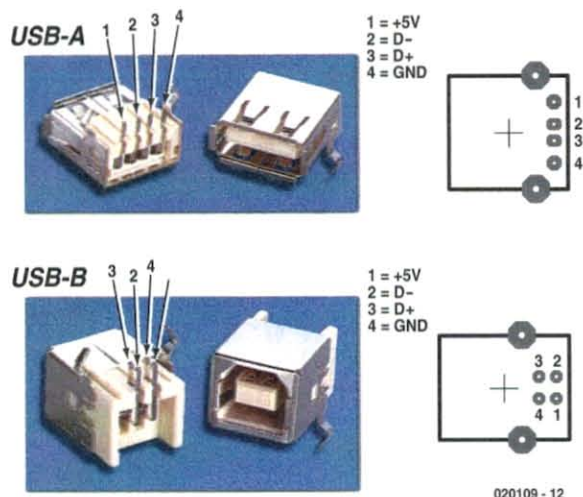
Allt detta kommer vi att gå igenom ett steg i taget, men först skall vi titta på vad som finns tillgängligt på marknaden. Många IC-tillverkare erbjuder USB-IC, men det är bara några få som erbjuder en bra support.

Ett utomordentligt exempel på detta är Cypress Semiconductor: ([www.cypress.com](http://www.cypress.com)).



020109 - 11

Figur 1. Minimal konfiguration för användning av Cypress AN2131SC IC.



Figur 2. Stifttilldelningen på USB-kontaktarna.

## Grundkrets för USB-utvecklare

Vår standard USB-kontrollern kommer

från Cypress och har beteckningen AN2131SC. Specifikationerna för denna mikrokontrollern ges i tabell 1, och dess datablad kan hämtas från

webben vid adressen [www.cypress.com/cf-uploads/img/products/AN2131SC.pdf](http://www.cypress.com/cf-uploads/img/products/AN2131SC.pdf).

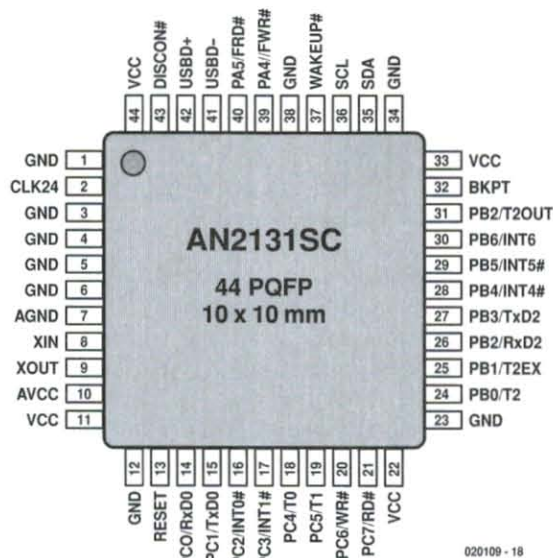
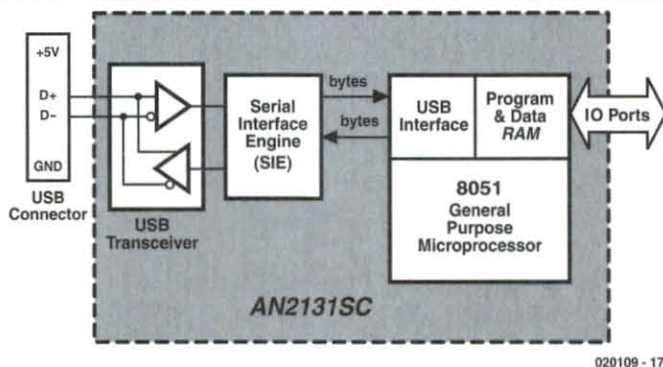
Som man kan se av figur 1 så finns det, förutom USB-mikrokontrollern, ytterligare en specialkomponent i grundkretsen, USB-kontakten. Det finns två versioner av denna som kallas 'typ A' och 'typ B'. Typ A versionen sitter alltid på PC eller hubsidan och typ B versionen sitter alltid på den perifera enhet som kopplas till PC-n. Ström kan dras från en typ A kontakt medan en typ B kontakt kan ta emot ström. Följaktligen kommer vi att enbart använda kontakter av typ B (figur 2) i vår krets. En USB-kabel är alltid kopplad 1:1.

Maximalt 500 mA kan dras via USB-kabeln, men Cypress-IC:n måste först rapportera nivån på strömförbrukningen, för om PC:n inte har så mycket reservkraft kvar kommer den att switcha från enheten automatiskt. IC-kretsen AN2131SC rapporterar automatiskt ett värde på 100 mA via sin interna USB kommunikationsmjukvara. Om hårdvaran behöver mer ström måste du skriva din egen mjukvara för USB-kommunikation.

Ett seriellt EEPROM, kan anslutas till SDA

## USB-kontrollern

AN2131SC enchips mikrokontrollern innehåller en 8051 processorkärna och ett fullfärdigt USB- interface. Mikrokontrollern har 8 kB RAM, tre räknare/timer, två seriella interface och interruptingångar. Det finns totalt 18 fritt programmerbara I/O-ben tillgängliga. Ett seriellt EEPROM kan anslutas och fungera som programminne, eller också kan IC:n köras från ett program som laddats in i dess RAM via USB. DRT behövs i stort sett inga andra externa komponenter. AN2131SC körs från 3,3 V och där USB-nets portklockfrekvens tas från 12-MHz oscillatorfrekvensen. En frekvensdubblare används för att generera en 24-MHz klocka för processorn. Tack vare ett optimerat instruktionsset utförs maskininstruktionerna fyra gånger så snabbt som vanligt. Timer 0 och Timer 1 är 8-bitars timers, medan Timer 2 är en 16-bitars. Den signal som läggs till ingångsporten för räknare/timer får inte ha högre frekvens än 2 MHz. Det seriella interfacet kan köras med 9600 baud med hj-Timer 0 eller Timer 1. Upp till 38400 baud är möjligt med Timer 2. För att uppnå ännu högre hastighet måste en annan frekvens läggas till timeringsgången. IC:n kan reagera individuellt för fem separata aktivt-låga eller aktivt-höga interrupter. I<sup>2</sup>C-benen måste vara anslutna till 3,3 V via motstånd på 4,7 kΩ, och Reset-ingången måste vara kopplad till ett POR-nät. Det enda som behöver göras efter detta är att ansluta allting till en USB-port så att PC:n kan installeras en enhetsdrivare. Mjukvaruuppdatering behövs inte längre med den andra versionen av mikrokontrollern eftersom med Actice Software så körs det lämpliga programmet alltid i USB-IC:n. Med hjälp av de inbyggda renumreringsegenskapen kan USB-interfacet logga in på operativsystemet med en ny VID och PID.



och SCL-linjerna hos Cypress-ICn. Antingen kan ID eller ID och ett program lagras i detta EEPROM. Om inget program är lagrat i EEPROM-et arbetar USB-delen och processorn helt självständigt. All kommunikation sker via RAM-et. Processorn kan placeras i Reset-mod med hjälp av USB drivkommandon. Efter detta kan ett program laddas in i RAM med början vid adress 0. Dessutom har USB-drivaren full åtkomst till hela RAM-regionen, även när 8051-kärnan körs. Det är inte nödvändigt att skriva en enda rad kod för detta eftersom USB-delen i mikrokontrollern klarar av detta själv.

Om du vill utveckla en krets för dig själv så kan denna grundkrets vara en bra utgångspunkt. Du skall dock noga överväga hur du vill använda portarna. Även om portbenen är fritt programmerbara kan de också användas som timerångar, interrupter och till och med som två serieinterface. Om du ansluter IC-kretsar som har interruptutgångar så skall du definitivt använda interrupterna. Detta gör programmeringen betydligt enklare, även om du använder C.

## Verktyg från Internet

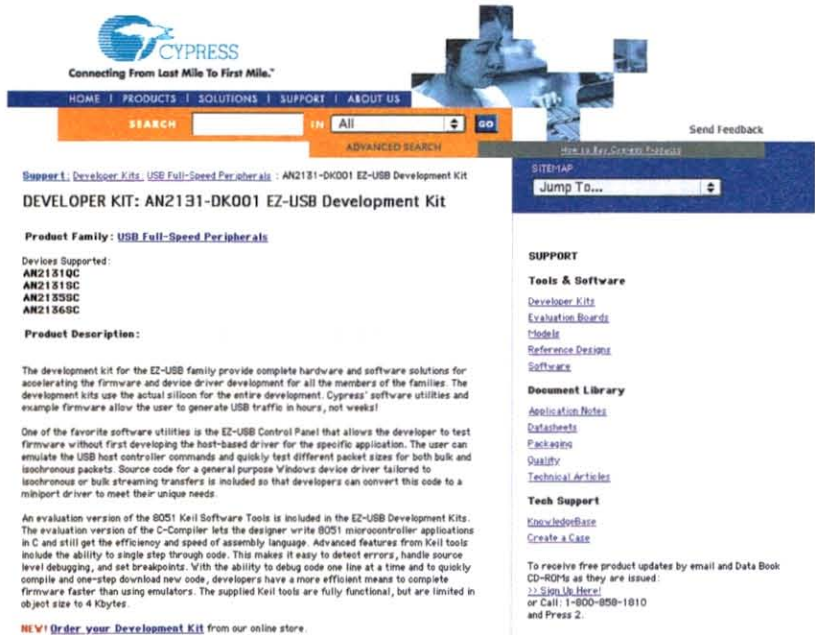
Innan du kan börja med den egentliga programmeringen behöver du några program och verktyg från Internet, som följande:

1. Cypress EZUSB Development Kit innehåller en C-kompilator från Kiel tillsammans med kodmallar för att generera drivare för USB-enheter. (Vi kommer att använda uttrycket drivare i stället för drivrutiner). I den lokala sökmotorn anger du IC-kretsens beteckning och går sedan till AN2131-DK001 Developer Kit (figur 3). Här kan du ladda ner *EZ-USB Family Complete Tools*. Den aktuella versionen är V2.52.701 (oktober 2002), som är 63 153 kB stor.

2. Hämta också *EZ-USB Technical Reference Manual* från samma plats.

3. Slutligen behöver du BinTerm, som finns att hämta gratis från utvecklarens websida ([www.mmvisual.de](http://www.mmvisual.de)). Du kan använda BinTerm för att ladda ner det genererade C-programmet till Cypress-ICn via USB-interfacet och för att styra hela RAM-regionen. De nödvändiga enhetsdrivarna är inkluderade i BinTerm.

När du installerar utvecklingssatsen så skall du helst inte ändra installationsvägen. Välj installationsalternativet 'Custom' eftersom Kiel C-kompilatorn endast installeras med detta alternativ. När du installerar Keil uVision2 C kompilatorn får du inte ändra någon av installationskatalogerna eftersom pro-



Figur 3. Downloadsida för Developer's Kit.

gram exempel (som använder absoluta sökvägar) annars inte kommer att fungera korrekt.

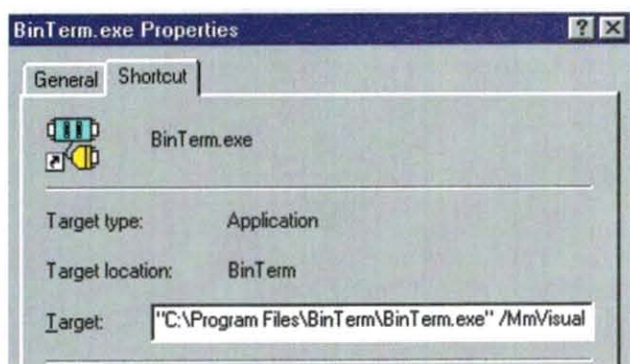
Du kan konsultera [www.keil.com](http://www.keil.com) för ytterligare tips om hur Kielprogrammet används. En viktig anmärkning är att utvärderingsdemot från Kiels websida inte fungerar korrekt med Cypress-ICn. Andra kompilatorer, som Rigel51, genererar kod som är sju gånger så stor som den som produceras av Kiel-kompilatorn. Många mjukvaruutvecklare svär över assemblerprogrammering. Ett program i full assembler är klumpigt att arbeta med, men det är lätt att inkorporera specifika assemblerrutiner i C-koden. Kod som är skriven i assemblyspråk kan endast användas med en specifik processortyp, medan C-kod kan användas för alla an-

dra processortyper med bara små modifieringar.

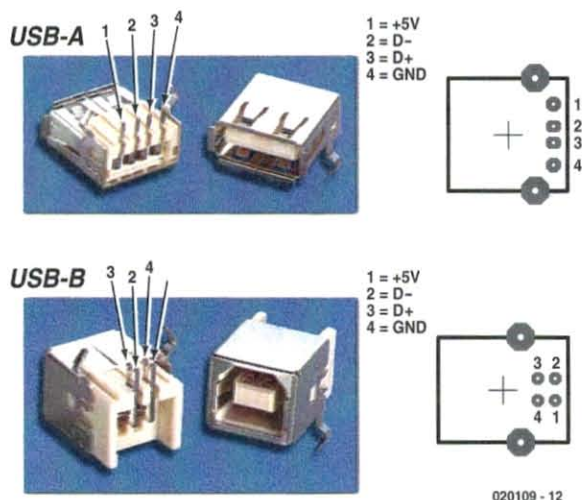
Det är inte nödvändigt att installera BinTerm, allt du behöver göra är att skapa en ny katalog (mapp) med namnet BinTerm under C:\Program och sedan helt enkelt kopiera alla filerna dit. Drag BinTerms programikon till skrivbordet för att skapa en genväg. Klicka sedan på genvägsikonen med höger musknapp, välj 'Egenskaper'. Modifiera länken genom att lägga parametern '/MmVisual' till Målfältet. Detta kommer att skapa en extra USB-Testflik i BinTermfönstret, se figur 4.

## C-kod för AN2131SC

Nu är du klar för lite riktigt arbete. Starta Keil uVision2, som är C-miljön



Figur 4. Så här anger du den extra uppstartsparametern.



Figur 2. Stifttilldelningen på USB-kontakterna.

## Grundkrets för USB-utvecklare

Vår standard USB-kontrollern kommer

från Cypress och har beteckningen AN2131SC. Specifikationerna för denna mikrokontroller ges i tabell 1, och dess datablad kan hämtas från

webben vid adressen [www.cypress.com/cf-uploads/img/products/AN2131SC.pdf](http://www.cypress.com/cf-uploads/img/products/AN2131SC.pdf).

Som man kan se av figur 1 så finns det, förutom USB-mikrokontrollern, ytterligare en specialkomponent i grundkretsen, USB-kontakten. Det finns två versioner av denna som kallas 'typ A' och 'typ B'. Typ A versionen sitter alltid på PC eller hubsidan och typ B versionen sitter alltid på den perifera enhet som kopplas till PC-n. Ström kan dras från en typ A kontakt medan en typ B kontakt kan ta emot ström. Följaktligen kommer vi att enbart använda kontakter av typ B (figur 2) i vår krets. En USB-kabel är alltid kopplad 1:1.

Maximalt 500 mA kan dras via USB-kabeln, men Cypress-IC:n måste först rapportera nivån på strömförbrukningen, för om PC-n inte har så mycket reservkraft kvar kommer den att switcha från enheten automatiskt. IC-kretsen AN2131SC rapporterar automatiskt ett värde på 100 mA via sin interna USB kommunikationsmjukvara. Om hårdvaran behöver mer ström måste du skriva din egen mjukvara för USB-kommunikation.

Ett seriellt EEPROM, kan anslutas till SDA

## USB-kontrollern

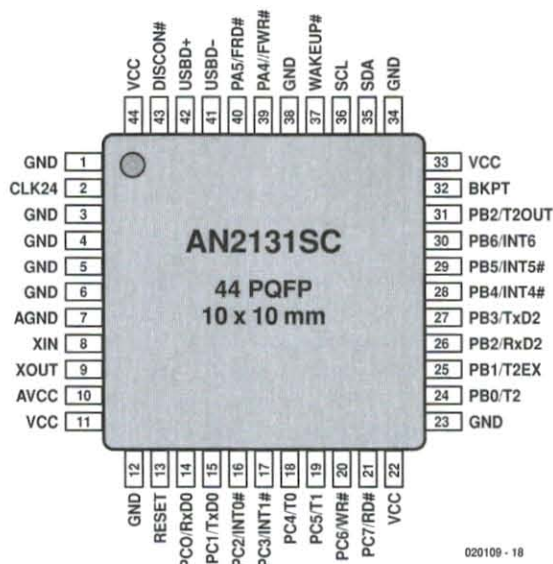
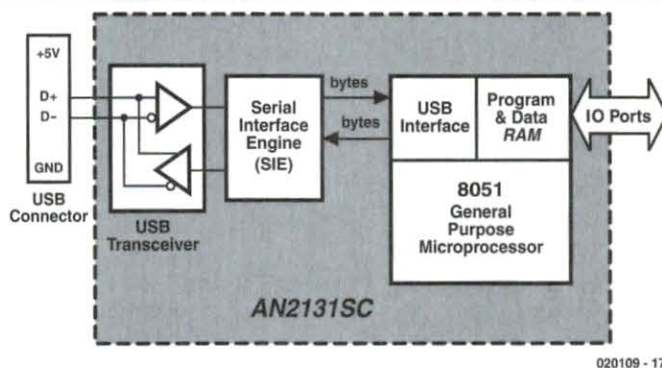
AN2131SC enchips mikrokontroller innehåller en 8051 processorkärna och ett fullfärts USB- interface. Mikrokontrollern har 8 kB RAM, tre räknare/timer, två seriella interface och interruptingångar. Det finns total 18 fritt programmerbara I/O-ben tillgängliga.

Ett seriellt EEPROM kan anslutas och fungera som programminne, eller också kan IC:n köras från ett program som laddats in i dess RAM via USB. Drt behövs i stort sett inga andra externa komponenter. AN2131SC körs från 3,3 V och där USB-n's portklockfrekvens tas från 12-MHz oscillatorfrekvensen. En frekvensdubblare används för att generera en 24-MHz klocka för processorn. Tack vare ett optimerat instruktionsset utförs maskininstruktionerna fyra gånger så snabbt som vanligt.

Timer 0 och Timer 1 är 8-bitars timers, medan Timer 2 är en 16-bitars. Den signal som läggs till ingångsporten för räknare/timer får inte ha högre frekvens än 2 MHz. Det seriella interfacet kan köras med 9600 baud med hjälp av Timer 0 eller Timer 1. Upp till 38400 baud är möjligt med Timer 2. För att uppnå ännu högre hastighet måste en annan frekvens läggas till timeringsgången.

IC:n kan reagera individuellt för fem separata aktivt-låga eller aktivt-höga interrupter. I<sup>2</sup>C-benen måste vara anslutna till 3,3 V via motstånd på 4,7 kΩ, och Reset-ingången måste vara kopplad till ett POR-nät. Det enda som behöver göras efter detta är att ansluta allting till en USB-port så att PC:n kan installeras en enhetsdrivare.

Mjukvaruuppdatering behövs inte längre med den andra versionen av mikrokontrollern eftersom med Actice Software så körs det lämpliga programmet alltid i USB-IC:n. Med hjälp av de inbyggda renumereringsegenskapen kan USB-interfacet logga in på operativsystemet med en ny VID och PID.



och SCL-linjerna hos Cypress-ICn. Antingen kan ID eller ID och ett program lagras i detta EEPROM. Om inget program är lagrat i EEPROM-et arbetar USB-delen och processorn helt självständigt. All kommunikation sker via RAM-et. Processorn kan placeras i Reset-mod med hjälp av USB drivkommandon. Efter detta kan ett program laddas in i RAM med början vid adress 0. Dessutom har USB-drivaren full åtkomst till hela RAM-regionen, även när 8051-kärnan körs. Det är inte nödvändigt att skriva en enda rad kod för detta eftersom USB-delen i mikrokontrollern klarar av detta själv.

Om du vill utveckla en krets för dig själv så kan denna grundkrets vara en bra utgångspunkt. Du skall dock noga överväga hur du vill använda portarna. Även om portbenen är fritt programmerbara kan de också användas som timerångar, interrupter och till och med som två serieinterface. Om du ansluter IC-kretsar som har interruptutgångar så skall du definitivt använda interrupterna. Detta gör programmeringen betydligt enklare, även om du använder C.

## Verktyg från Internet

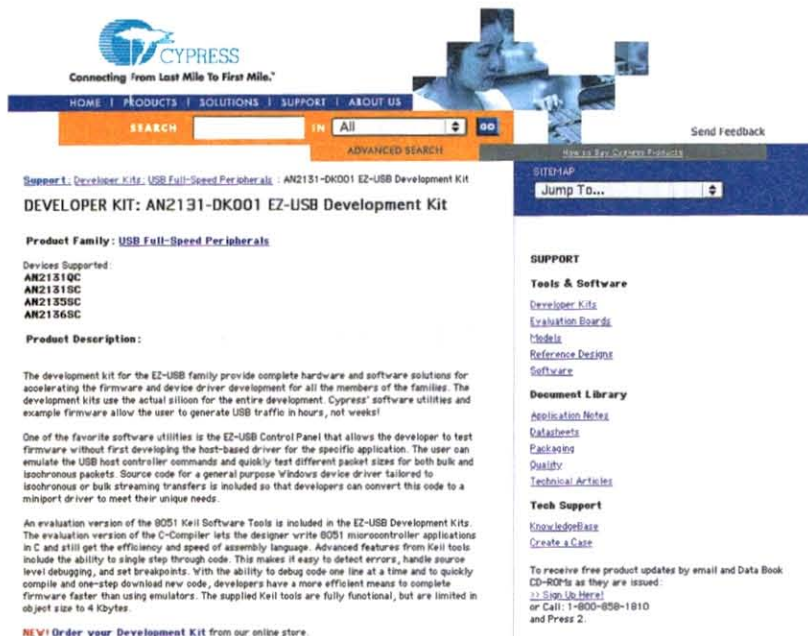
Innan du kan börja med den egentliga programmeringen behöver du några program och verktyg från Internet, som följande:

1. Cypress EZUSB Development Kit innehåller en C-kompilator från Kiel tillsammans med kodmallar för att generera drivare för USB-enheter. (Vi kommer att använda uttrycket drivare i stället för drivrutiner). I den lokala sökmotorn anger du IC-kretsens beteckning och går sedan till AN2131-DK001 Developer Kit (figur 3). Här kan du ladda ner *EZ-USB Family Complete Tools*. Den aktuella versionen är V2.52.701 (oktober 2002), som är 63 153 kB stor.

2. Hämta också *EZ-USB Technical Reference Manual* från samma plats.

3. Slutligen behöver du BinTerm, som finns att hämta gratis från utvecklarens websida ([www.mmvisual.de](http://www.mmvisual.de)). Du kan använda BinTerm för att ladda ner det genererade C-programmet till Cypress-ICn via USB-interfacet och för att styra hela RAM-regionen. De nödvändiga enhetsdrivarna är inkluderade i BinTerm.

När du installerar utvecklingssatsen så skall du helst inte ändra installationsvägen. Välj installationsalternativet 'Custom' eftersom Kiel C-kompilatorn endast installeras med detta alternativ. När du installerar Keil uVision2 C kompilatorn får du inte ändra någon av installationskatalogerna eftersom pro-



Figur 3. Downloadsida för Developer's Kit.

gram exempel (som använder absoluta sökvägar) annars inte kommer att fungera korrekt.

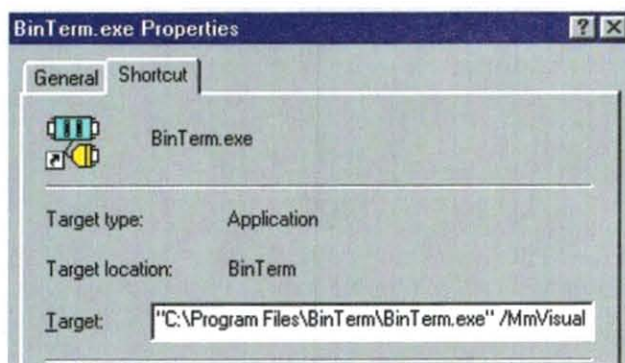
Du kan konsultera [www.keil.com](http://www.keil.com) för ytterligare tips om hur Kielprogrammet används. En viktig anmärkning är att utvärderingsdemot från Kiels websida inte fungerar korrekt med Cypress-ICn. Andra kompilatorer, som Rigel51, genererar kod som är sju gånger så stor som den som produceras av Kiel-kompilatorn. Många mjukvaruutvecklare svär över assemblerprogrammering. Ett program i full assembler är klumpigt att arbeta med, men det är lätt att inkorporera specifika assemblerrutiner i C-koden. Kod som är skriven i assemblyspråk kan endast användas med en specifik processortyp, medan C-kod kan användas för alla an-

dra processortyper med bara små modifieringar.

Det är inte nödvändigt att installera BinTerm, allt du behöver göra är att skapa en ny katalog (mapp) med namnet BinTerm under C:\Program och sedan helt enkelt kopiera alla filerna dit. Drag BinTerms programikon till skrivbordet för att skapa en genväg. Klicka sedan på genvägsikonen med höger musknapp, välj 'Egenskaper'. Modifiera länken genom att lägga parametern '/MmVisual' till Målfältet. Detta kommer att skapa en extra USB-Testflik i BinTermfönstret, se figur 4.

## C-kod för AN2131SC

Nu är du klar för lite riktigt arbete. Starta Keil uVision2, som är C-miljön



Figur 4. Så här anger du den extra uppstartsparametern.

för 8051-an. Skapa ett nytt projekt under Project -> New project. Ange filnamnet 'RamTest' och bekräfta med 'OK'. Nästa steg är att välja processor (Select Device for Target): Välj Cypress Semiconductor -> EZ-USB AN21XX -> OK. I projektfönstret (det vänstra fönstret) ser du att programmet skapat 'Target 1' med Source Group 1.

I Target 1 öppnar du popup-menyn genom att klicka med höger musknapp och sedan klicka på 'Options for Target Target 1'. Nu öppnas en dialogruta där du bekvämt kan ange alla inställningar. Välj fliken 'Output' och markera rutan 'Create Hex File' och avsluta sedan med OK.

För att kunna skriva kod måste du först skapa en ny, tom, fil med hjälp av menykommandot File -> New. Använd därefter File -> Save As för att lagra filen i den aktuella projektkatalogen med namnet 'eMain.c'. Använd sedan File -> Close för att stänga filen.

Filen finns nu i projektet, men den måste fortfarande läggas till. För att göra detta väljer du 'Source Group 1' i 'Project Window', klickar med höger musknapp och väljer 'Add files to Group Source Group 1'. Du kommer att föreslås filen eMain.c och du kan lägga denna till projektet med 'Add'. Stäng nu med 'Close'.

En ny post med namnet 'eMain.c' kommer nu att finnas under Source Group 1. Ett dubbelklick öppnar redigeraren (editorn) som nu vet att den har att göra med en C-källkod och aktiverar syntaxmarkering. En stor nackdel när det gäller C-språket är att kompileraren skiljer mellan stora och små bokstäver.

Nu kan du generera ett litet program för Cypress-ICn genom att använda editorn för att ange de kodrader som visas i Listning 1.

Tryck på F1 för att generera projektet. Kompileraren skall rapportera '0 Errors, 0 Warnings' om koden har

skrivits in korrekt.

## Hur programmet fungerar

De två första raderna genererar två variabler med fasta minnestilldelningar. Processorn startar programmet med hjälp av funktionen *main*. Den ändlösa loop *while(1) { }* körs hela tiden (tills nästa Reset). Den 'meningsfulla uppgiften' för denna rutin är att kopiera innehållet i minnesadress 0x100 till adress 0x101.

När projektet genereras med hjälp av F1 skapas flera filer i projektkatalogen. Filen med tillägget '.hex' kan laddas in i Cypress-ICn med BinTerm och sedan startas.

Starta BinTerm med alternativet '/MmVisual' med hjälp av den genvägsikon som du lagt på skrivbortet. Välj fliken USB-Test. Du kan styra hela Cypress-ICn med denna programfunktion. Plugga in en USB-enhet som använder Cypress-ICn. Enhetsnamnet för hårdvaran måste väljas utifrån om du använder en USB-dataspion eller en Cypress-IC utan något EEPROM.

Vid denna punkt kan BinTerm skapa en drivdiskett som innehåller enhetsdrivaren för USB-enheten. När den nya USB-enheten pluggas in kommer operativsystemet att försöka installera en enhetsdrivare som gör att BinTerm mjukvaran kan kommunicera med Cypress USB-IC.

Inbyggd kommunikation mellan BinTerm och USB-enheten måste kopplas bort genom att markera rutan 'stop query' som finns under 'controlling BinTerm'. Du kommer också att se en knapp med tre punkter intill den knapp som har beteckningen 'send new program'. Klicka på denna knapp för att öppna filen 'RamTxt.hex' som genererats av kompilatorn och ladda ner denna till USB-ICn. Klicka slutligen på 'start USB program' knappen så att Cypress-ICn börjar köra.

## Att testa 8051-processorn

Processorn kan testas genom att kolla minnesregionen. Detta gör man genom att ange och visa numeriska värden i hexadecimal form, såsom visas i figur 5. Alternativet 'write in the address' kan användas för att skriva ett byte direkt till RAM-et medan 8051-processorn körs. I det visade exemplet kommer bytet 0x02 att skrivas till adress 0x100. Bytet skrivs när knappen 'Send' aktiveras. Alternativet 'read from address' kan användas för att specificera startadressen för läsning av ett antal byte (i detta exempel tre byte som börjar vid adress 0x100). Alternativet 'autoread every second' får det visade värdet att uppdateras automatiskt.

Det lilla programexemplet har nu laddats in i Cypress-ICn och startats. Som du kan se fungerar IC-n perfekt. Bytet i adress 0x100 kopieras till adress 0x101 precis som vi vill. Men hur är det möjligt för MinTerm att kommunicera med Cypress-ICn då du inte skrivit en enda rad med kod för att driva USB-interfacet och inte heller ställt in några parametrar för någon typ av specialregister?

Svaret är att USB-delen i Cypress-ICn arbetar helt självständigt, tillsammans med Cypress enhetsdrivare, precis som om en andra mikrokontroller skulle ta hand om USB-interfacet. Dessutom har denna självständiga del full tillgång till RAM-regionen hos 8051-an hela tiden. Processorn kan till och med placeras i Reset-tillstånd och återstartas via register 0x7F92. Alla RAM-adresser kan skrivas till och läsas från hela tiden och vi har dragit fördel av denna egenskap. Även om du inte vet någonting om C och mikrokontrollers så har du precis programmerat en!

På Kiels hemsida finns det en hel del programexempel som kan laddas ner. Du kan också söka på Internet och du kan vara säker på att hitta någonting som du kan använda.

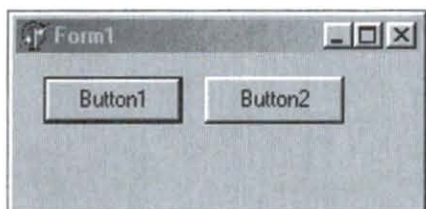
## Kan mitt eget program också vara så enkelt?

Programexemplen har genererats med hjälp av Delphi, som är den i särklass mest effektiva programmiljön för professionellt bruk och

### Listing 1.

```
xdata unsigned char BYTE_IN_at_    0x0100; // Byte arriving (RAM address)
xdata unsigned char BYTE_OUT_at_    0x0101; // Byte to be written (RAM address)
void main()
{ while (1)    // endless loop
  {   BYTE_OUT = BYTE_IN;    // Copy byte from BYTE_IN to BYTE_OUT
  }
}
```

som garanterar enkel användning av Visual Basic, klasser och C-typstestning. Delphi kan laddas ner gratis från [www.borland.com](http://www.borland.com). Nu kommer vi till den praktiska delen. Först måste du förstå hierarkin från USB-enheten till ditt program. Du har en enhet som du ansluter till USB-interfacet. Baserat på kodningen hos de motstånd som är anslutna till dataledarna D+ och D- så känner PC:n igen datahastigheten hos USB-enheten. Windows enhetsdrivare kommer nu att fråga enheten efter Vendor-ID och Product-ID. Om Windows redan känner igen enheten laddas en tillverkar-specifik enhetsdrivare. I annat fall letar Windows i sin databas efter en lämplig enhetsdrivare och erbjuder sig att installera denna. Den tillverkar-specifika enhetsdrivaren kommunicerar aldrig direkt med den egentliga USB-enheten, bara med Windowsdrivaren. För varje USB-interface kan Windows hantera en hub och flera enheter anslutna till denna hub. Applikationsprogrammet kan bara skapa en länk till USB-enheten via den tillverkar-specifika enhetsdrivaren. Windows tillhandahåller kommandona 'CreateFile' och 'DeviceIoControl' i Kernel32.dll för detta ändamål. För att producera ett program med Delphi genererar du först en ny applikation med File -> New -> Application. Nu får du upp en tom form. Placera två 'Tbutton' knappar i denna form. Lägg till den kod som visas i Listning 2 mellan slutet av den första Usersektionen och början av den första Typesektionen i programmet.



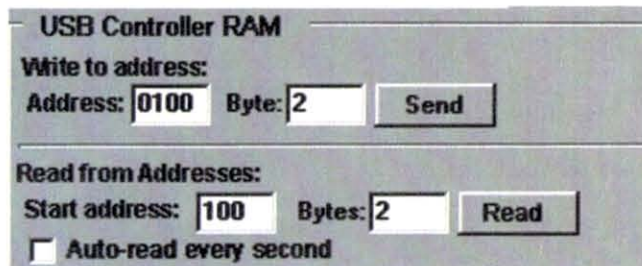
**Figur 6. En form med två kommandoknappar.**

## Listing 2.

```
type _VENDOR_REQUEST_IN = record
  bRequest : Byte;
  wValue : Word;
  wIndex : Word;
  wLength: Word;
  direction : Byte;
  bData : Byte;
end;
```

Insert the following code snippet at the line containing "implementation"

```
Function WrRAM(Adr: Word; Dat: Byte): Boolean; // Write byte to RAM address
```



**Figur 5. Kontroll av RAM-regionen.**

USB-enheten körs med hjälp av programmet RamTest.hex. Om enheten pluggas ur måste programmet laddas igen med BinTerm innan enheten kan användas igen. BinTerm kan köras i bakgrunden med hjälp av RAM query för att kolla funktionen hos enheten genom att övervaka dess RAM. BinTerm kommer inte att blockera USB-enheten. Starta Delphiprogrammet från Delphikompiletorn genom att trycka på F9. När kommandoknappen 'Button1' aktiveras skrivs Byte 1 till RAM-adress 0x100 i USB-enheten. Värdet i adress 0x101 kan läsas och visas med knappen 'Button 2' se figur 6.

## Data USB-enheten

Funktionen 'WrRAM' kan skriva data till RAM-et i Cypress-ICn. Funktionen 'RdRAM' läser ett byte från en specificerad adress. Enhetsnamnet (som är korrekt vid operation utan EEPROM) lagras i variabeln 'pGer-aet' ('pDevice'). Om en BinTerm USB dataspionadapter används måste du använda namnet '\\.\binterm-0' i stället för '\\.\ezusb-0'. Kommandot 'CreateFile' genererar en virtuell fil som skapar en länk till enhetsdrivaren.

Kommandot 'DeviceIoControl' tillåter direkt kommunikation med enhetsdrivaren. En post av typen `_VENDOR_REQUEST_IN` används för run-time övervakning. Kommandot 'CloseHandle' stänger länken till enhetsdrivaren. Eftersom länken stängs efter varje förfrågan (request) är det möjligt för flera program att få tillgång till hårdvaran, som det verkar samtidigt, utan att enheten är medveten om detta. Förresten, om du byter namn på alla ovan nämnda funktioner så fungerar detta också med Visual Basic eller C++.

(020109-1)

Den andra artikeln i denna serie kommer att handla om att producera enhetsdrivare för USB, vilket inte kräver att du har detaljerade kunskaper i C!

```

var USBDeviceHandle: THandle;
    USBTemplateHandle: THandle;
    nBytes: DWord;
    MyRequest: _VENDOR_REQUEST_IN;
    USBError: Boolean;
    pGeraet: PChar;
begin
    pGeraet := PChar('\\\\.\\ezusb-0'); // using BinTerm: '\\.\\binterm-0'
    USBError := False;
    myRequest.bRequest := $A0;
    myRequest.wValue := Adr;
    myRequest.wIndex := 0;
    myRequest.wLength := 1;
    myRequest.direction := 0;
    myRequest.bData := Dat;
    USBTemplateHandle := 0;
    USBDeviceHandle := CreateFile (pGeraet, Generic_write, File_Share_write, nil, open_existing,
                                   0, USBTemplateHandle);
    If USBDeviceHandle = INVALID_HANDLE_VALUE Then
        USBError := True;
    Result := DeviceIoControl(USBDeviceHandle, $00222014, @myRequest, 10, nil, 0, nBytes, nil);
    CloseHandle (USBDeviceHandle);
end;

```

function RdRAM(Adr: Word): Byte; // Read byte from RAM address

```

var USBDeviceHandle: THandle;
    USBTemplateHandle: THandle;
    nBytes: DWord;
    MyRequest: _VENDOR_REQUEST_IN;
    Buffer: Array [1..2] of Byte;
    USBError: Boolean;
    pGeraet: PChar;
begin
    pGeraet := PChar('\\\\.\\ezusb-0'); // using BinTerm: '\\.\\binterm-0'
    USBError := False;
    USBTemplateHandle := 0;
    myRequest.bRequest := $A0;
    myRequest.wValue := Adr;
    myRequest.wIndex := 0;
    myRequest.wLength := 1;
    myRequest.direction := 1; // Read
    myRequest.bData := $00;
    USBDeviceHandle := CreateFile (pGeraet, Generic_write, File_Share_write, nil, open_existing,
                                   0, USBTemplateHandle);
    If USBDeviceHandle = INVALID_HANDLE_VALUE Then
        USBError := True;
    DeviceIoControl(USBDeviceHandle, $00222014, @myRequest, 10, @Buffer, SizeOf(Buffer), nBytes, nil);
    CloseHandle (USBDeviceHandle);
    RdRAM := Buffer[1];
end;

```

Double-click on the 1st key and complete the code with this:

```

procedure TForm1.Button1Click(Sender: TObject);
begin
    WrrAM($100, 1);
end;

```

The same for the 2nd key:

```

procedure TForm1.Button2Click(Sender: TObject);
begin
    Button2.Caption := IntToHex(RdRAM($101), 2);
end;

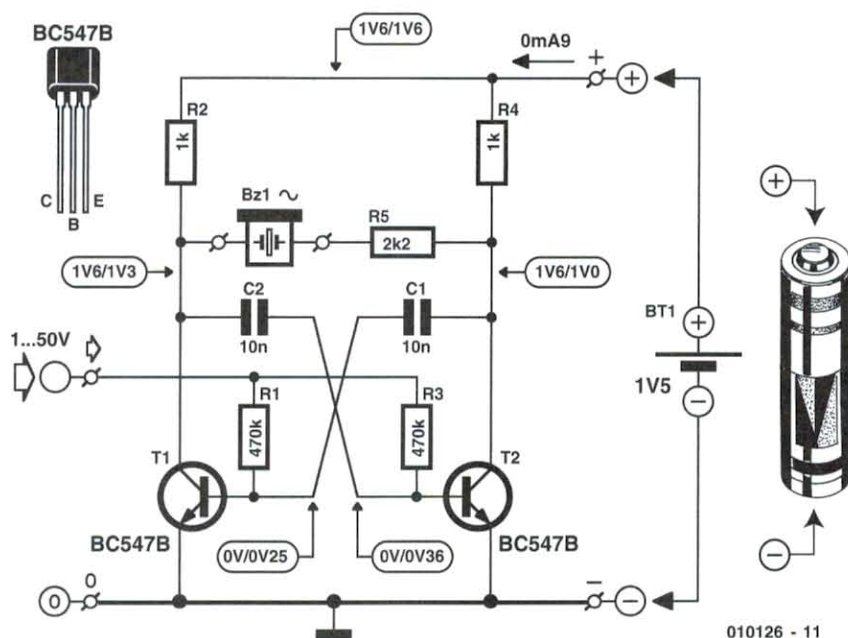
```

# Akustisk voltmeter

## hör vad du mäter

Design av B. Kainka

Med testproben säkert tryckt mot (hoppas du) motståndsbenet och sedan vrida på huvudet för att avläsa voltmeteren är inte alltid en så god idé. När dina ögon fokuserar på det visade värdet på mätarskalan kan proben glida åt sidan, orsaka en kortslutning och kanske förstöra en komponent. Inga sådana problem med denna månads Miniprojekt.



Figur 1. En enkel multivippa lämnar sin utgångssignal till en passive summer.

Den lilla krets som beskrivs i denna månads Miniprojekt omvandlar en elektrisk spänning till en akustisk (hörbar) signal. Ju högre spänning du mäter desto högre frekvens kommer du att höra. Även om denna typ av indikering inte är lika noggrann som den trogne voltmeteren med ett vridspoleinstrument, eller en digital visning, så är det en klar fördel att du kan koncentrera dig på mätpunkterna i kretsen utan att behöva riskera en titt på instru-

mentet. Men lite övning kommer du också att uppnå överraskande noggrannhet med hjälp av den 'akustiska' metoden.

### Multivibrator

Som du kan se av figur 1 består kretsen av en enkel multivibrator (multivippa) i vilken basströmmen

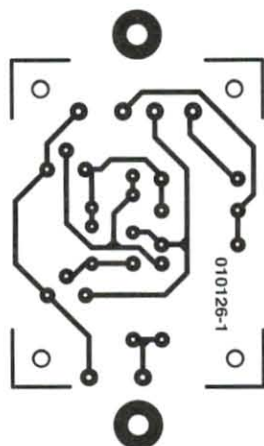
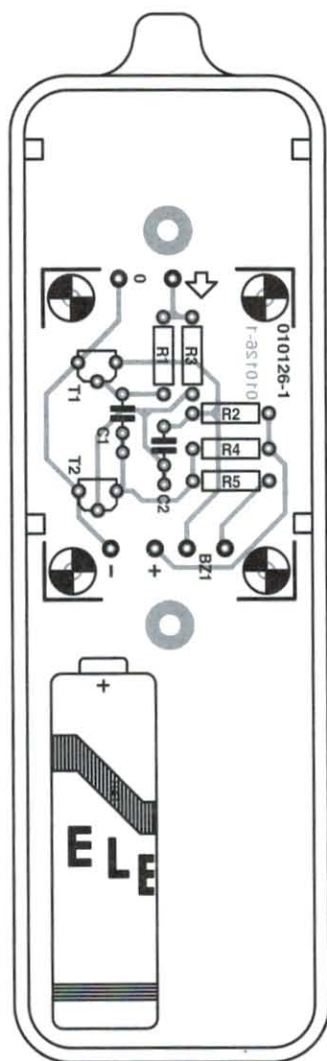
till de båda transistorerna kommer från den krets som du håller på att 'peta' i. Ett piezokeramisk element (passiv summer eller ac-summer) omvandlar den fyrkantsignal som produceras av den Akustiska voltmeteren till en hörbar signal. Vid en uppmätt spänning på ca 10 V är den producerade frekvensen ungefär 1 kHz. Från ca 2 V och framåt finns det ett linjärt förhållande mellan den uppmätta spänningen och den producerade frekvensen.

Mätarens inspänningsområde sträcker sig gott och väl över 50 V. Negativa spänningar producerar inte något ljud. De skadar inte kretsen heller då de låses på en säker nivå på ca -9 V av interna zenerdioder vid transistorernas basterminaler. En till/frånkopplare är överflödigt eftersom det inte finns något strömflöde om det inte finns någon ingångssignal.

Växelspänningar (VAC) kan också mätas utan problem. Dessa ger ett ganska speciellt utgångsljud då den producerade frekvensen moduleras av frekvensen hos den uppmätta växelspänningen. Detta gör det också enkelt att upptäcka ostadiga likspänningar, såväl som de som sakta fluktuerar eller har kraftigt rippel.

## Övning!

Den Akustiska voltmeteren går att bygga snabbt och enkelt om du använder kortlayouten som visas i fi-



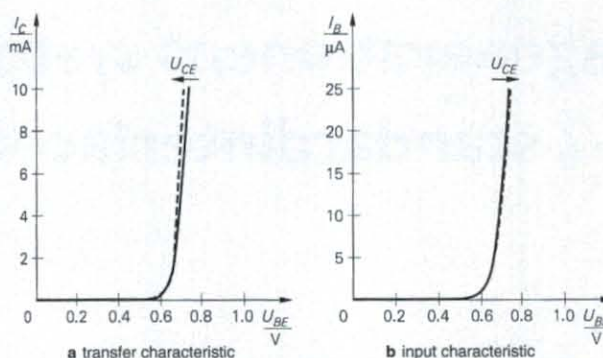
Figur 2. Kretskortslayouten. Kortet finns tyvärr inte att köpa färdigt.

## Det handlar om förstärkning

Teoretiskt skall spänningarna vid bas och kollektorterminalerna hos transistorerna i denna krets vara identiska. Även om detta är fallet när kretsen inte är aktiv (första värdena) så är värdena annorlunda när allting börjar att arbeta (andra värdena). En förklaring till denna konstighet ges nedan.

En småsignaltransistor som BC547 karakteriseras av sin strömförsörjning,  $H_{FE}$ , där förhållandet mellan kollektorströmmen,  $I_C$ , och basströmmen  $I_B$  uttrycks som

$$H_{FE} = I_C / I_B$$



Denna faktor är ett mått på transistorens typiska statiska strömförsörjning. I motsats till detta uttrycks den dynamiska förstärkningen (ett mått på strömförändringsförhållandet) som  $\beta$  eller  $h_{FE}$ . När det gäller de flesta småsignaltransistorer som BC547 är  $H_{FE}$  lika med  $h_{FE}$  under alla praktiska förhållanden.

Tyvärr är det inte möjligt för halvledartillverkare att producera transistorer med exakta förstärkningsspecifikationer. Databladet för en BC547 anger, till exempel, en förstärkning på '110-800' vid en konstant kollektorström. Tack och lov delas dock denna ganska klumpiga skala in i tre underklasser, som var och en identifieras av ett suffix i transistorbeteckningen. Med en BC547A kan vi vänta oss en förstärkning på 110-220, medan en BC547B och en BC547C har en förstärkning på 220-450 respektive 420-800.

Även när man använder två BC547B som i denna krets kan stora skillnader uppstå som ett resultat av det exakta förstärkningstalet i varje transistor. Trots detta är skillnaderna tolererbara även om de skulle resultera i skilda mätvärden.

gur 2. Förutom transistorerna och batteriet finns det inga polariserade komponenter. Även summern kan

placeras åt vilket håll som helst. Hela kretsen kan byggas in i en probliknande låda och kretskortet kan fästas med dubbelhäftande tejp eller limmas fast. Probspetsen kan anslutas till kortet via en kort bit kabel.

Summer låter inte speciellt mycket eftersom den arbetar från endast 1,5 V. Borra gärna ett par små hål i probens låda där summern sitter. Det tar lite tid att vänja sig vid detta lilla instrument. Men om du gör jämförande mätningar med en vanlig voltmeter kommer du snart att kunna avgöra en 'akustisk upplösning' på ca 1 volt. Växelspänningar kräver ytterligare övning innan du kan avgöra dess ungefärliga nivå och frekvens. Överkursen är när du även kan avgöra vågformen!

### KOMPONENTLISTA

#### Motstånd:

$R_1, R_3 = 470k\Omega$   
 $R_2, R_4 = 1k\Omega$   
 $R_5 = 2k\Omega$

#### Kondensatorer:

$C_1, C_2 = 10nF$

#### Halvledare:

$T_1, T_2 = BC547B$

#### Övrigt:

BZ1 = passiv (AC) summer  
 Bt1 = AA (R6/UM3) batteri med hållare  
 Problåda med probspets

(010126-1)

# Diagnossystem för fordon

## Andragenerationens system som använder OBD-2 standardinterfacet

av G. Müller

Diagnossystem för fordon har fram till nu varit tillverkarspecifika och i allmänhet konstruerade på så sätt att endast auktoriserad verkstadspersonal kunnat komma åt data. Detta håller nu på att förändras med tillkomsten av det standardiserade OBD-2 systemet, som har blivit en EU-standard (EOBD) för alla nya bilar. I denna artikel beskriver vi detta interface i detalj och i en kommande artikel visar vi en EOBD diagnosadapter som du kan bygga själv.

Faktorer som förbättrad säkerhet och ökad komfort, för att inte tala om miljövänlighet, har gjort moderna bilar mer och mer komplicerade. Ju oftare tillverkarna kommer fram med nya modeller, desto större blir trycket på servicekostnaderna och ju kortare tiden blir för utbildning av verkstadspersonal, desto viktigare blir diagnossystemen när det gäller att garantera snabb, säker och ekonomisk test, service och reparation.

Datorer, som ju har funnits i bilar under ganska lång tid nu, kan också användas för att diagnostisera de olika system och funktioner som de kontrollerar. För att kunna skydda sina auktoriserade verkstäder från de som inte är märkesspecifika, så har varje tillverkare fram till nu utvecklat sina egna system. En eller två standarder har tagit form, men bara som ett resultat av att försöka hålla kostnaderna nere. Detta har nu börjat förändras i takt med allt hårdare miljölagstiftning.

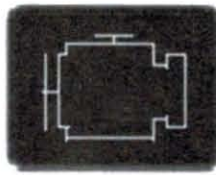
USA var det första landet som införde lagar mot luftföroreningar orsakade av avgaser och de spelar fortfarande en ledande roll. Speciellt

har delstaten Kalifornien varit framträdande när det gäller miljölagstiftningen. De europeiska biltillverkarna slogs länge mot kravet att förse alla sina bilar med katalysator, trots att de sedan länge hade tvingats att montera dem i bilar för den amerikanska marknaden. Tekniska framsteg, som mer kraftfulla datorsystem och utvecklingen av lämpliga sensorer, kombinerade med ännu tuffare krav på avgasrening, har lett fram till ett krav som inom en snar framtid skall genomföras i Kalifornien, en viss andel av bilarna i denna stat skall vara fordon med så kallat nollutsläpp.

Medan man i många andra länder har lagt kostnaden på bilföraren genom olika skatt på 'rena' och 'smutsiga' bilar så har man i USA lagt detta på biltillverkaren. Här har fabrikanterna tvingats att arrangera diagnossystemen i bilarna på så sätt

att föraren är tvingad att ta in bilen på verkstad även när tröskeln för avgasutsläpp överskrider kortvarigt mellan serviceintervallen. Detta system kallas 'On-Board Diagnosis' förkortat OBD. I händelse av för höga avgasutsläpp eller felaktigheter i fordonets säkerhetssystem skall en gul 'Kolla motorn' (Check Engine) lampa tändas, figur 1. Lampan har ibland etiketten 'Service Engine Soon' eller 'Malfunction Indicator Lamp', förkortat MIL. Detta system har varit ett krav i Kalifornien sedan 1988 och adopterades av alla andra delstater i USA 1994. 1996, efter ytterligare utveckling, ersattes systemet av det standardiserade OBD-2 (eller OBD-II) systemet, som nu används av i stort sett alla tillverkare.

I ett försök att uppmuntra öppen konkurrens, såväl som att tillmötesgå miljöintressena, har EU gett ut riktlinjer (98/69/EU, daterade 13/10/



**Figur 1. Lampsymbolen 'Kolla motorn'.**

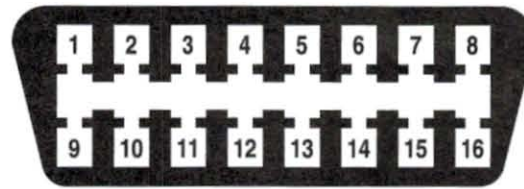
1998 [1], för ytterligare minskning av avgasutsläppen genom att kräva att alla biltillverkare installerar diagnossystem efter den amerikanska modellen. Tillverkarna kommer nu att för första gången tvingas att förse bilarna med standardiserade diagnosinterface och kontakter för anslutning av lämplig diagnosutrustning. Biltillverkarna kommer också att tvingas att publicera detaljer om sina diagnossystem, vilka hittills varit mer eller mindre hemliga. EU-s riktlinjer gäller tändstiftsmotorer (dvs bensinmotorer) registrerade år 2000 och senare, samt självtändande motorer (dvs dieselmotorer) registrerade år 2003 och senare.

## OBD-2 diagnoskontakt

Kraven stipulerar att diagnoskontakten för OBD-2, eller EOBD (European On-Board Diagnosis system) måste vara placerad i passagerarutrymmet nära förarplatsen. Normalt sitter kontakten under instrumentpanelen eller i mittkonsolen, ibland under en lucka och ibland bakom askkoppen.

OBD-2 använder en 16-polig kontakt (se figur 2) men det är inte alla stift som är inkopplade (se tabell 1)

Tabell 2 visar de olika protokollen i bussystemet. Vad som är helt tydligt är att det inte är någon större överenskommelse mellan tillverkarna. Bland de amerikanska företagen använder Ford ett PWM-protokoll (pulsbreddsmodulering) medan GM använder VPM-protokollet (variabel pulsbreddsmodulering, båda enligt SAEJ1850-standard, medan tillverkare i Europa och Asien i allmänhet använder KWP-protokollet (key word protokoll, nyckelordsprotokoll) enligt ISO 9141-2.



**Figur 2. Den 16-poliga diagnoskontakt som används i OBD-2.**

## Betydelse

Vad betyder nu det nya diagnossystemet för bilföraren? Där tidigare den mesta intelligensen fanns i den verkstadsdatorn som anslöts till diagnossystemet så är denna nu överförd till den allt kraftfullare datorn som finns i själva fordonet. Detta betyder att det enda som behövs för att leta efter det exakta felet i en bil är en billig avläsningsenhet. Detta kanske inte hjälper den genomsnittlige bilföraren eftersom han eller hon inte kan lösa problemet själv. Det betyder dock att han/hon kan köra till när-

## Tabell 1

### Standardiserade stift opå OBD-2-kontakten

|          |                         |
|----------|-------------------------|
| Stift 2  | J-1850 buss +           |
| Stift 4  | Chassijord              |
| Stift 5  | Signaljord              |
| Stift 6  | CAN hög (J-2284)        |
| Stift 7  | ISO 9141-2 K utgång     |
| Stift 10 | J-1850 buss             |
| Stift 14 | CAN låg (J-2284)        |
| Stift 15 | ISO 9141-2 L utgång     |
| Stift 16 | Batteriets positiva pol |

## Tabell 2

### Realtidsdata läsbara i mod 01 (alla data finns inte tillgängliga i alla bilmodeller):

|                                                                                             |
|---------------------------------------------------------------------------------------------|
| Status hos bränslesystem 1 och 2                                                            |
| Beräknat belastningsvärde                                                                   |
| Kylvattentemperatur                                                                         |
| Korttids bränsle-luftblandningsproportion bank 1                                            |
| Långtids bränsle-luftblandningsproportion bank 1                                            |
| Korttids bränsle-luftblandningsproportion bank 2                                            |
| Långtids bränsle-luftblandningsproportion bank 2                                            |
| Bränsletryck (uppmätt)                                                                      |
| Insugstryck                                                                                 |
| Motorhastighet                                                                              |
| Väghastighet                                                                                |
| Tändinställning för cylinder 1                                                              |
| Insugets lufttemperatur                                                                     |
| Sensoravläsning för Mass air flow (MAF)                                                     |
| Absolut läge för bränsleventil                                                              |
| Commanded secondary air status                                                              |
| Installed Lambdaprobcplaceringar där prob 1 är närmast motorn                               |
| Bank 1 sensor 1 Lambdaprobcavläsning och beräknad korttids bränsle-luftblandningsproportion |
| Bank 1 sensor 2 Lambdaprobcavläsning och beräknad korttids bränsle-luftblandningsproportion |
| Bank 1 sensor 3 Lambdaprobcavläsning och beräknad korttids bränsle-luftblandningsproportion |
| Bank 1 sensor 4 Lambdaprobcavläsning och beräknad korttids bränsle-luftblandningsproportion |
| Bank 2 sensor 1 Lambdaprobcavläsning och beräknad korttids bränsle-luftblandningsproportion |
| Bank 2 sensor 2 Lambdaprobcavläsning och beräknad korttids bränsle-luftblandningsproportion |
| Bank 2 sensor 3 Lambdaprobcavläsning och beräknad korttids bränsle-luftblandningsproportion |
| Bank 2 sensor 4 Lambdaprobcavläsning och beräknad korttids bränsle-luftblandningsproportion |
| OBD-requirements, modellberoende                                                            |
| Status hos extraingångar                                                                    |

## Dekodering von Fehlercodes nach SAE J2012



Figur 3. Felkoder enligt SAE 2012.

maste verkstad i stället för att först boka tid hos en märkesverkstad (och det vet vi ju alla hur lång tid det kan ta). Och han/hon behöver dessutom inte oroa sig för vad den tända servicelampen betyder, i handboken står det 'kör till n ä r -



Figur 4. Typisk handhållet instrument för avläsning av felkoder. (Källa: Actron Manufacturing Company, Cleveland, Ohio).

maste verkstad'.

## Felkonditioner

När, och varför, tänds felindikatorlamporna? Tillsammans med övriga indikatorlampor tänds Malfunction Indicator Lamp (från och med nu MIL) när startnyckeln vrids om och slocknar när motorn startar. Detta visar att lamporna i sig själva fungerar. Om en lampa tänds, och förblir tänd, under körning så är det något fel. Om lampan blinkar under färden visar detta att motorelektroniken har upptäckt att motorn misständer, vilket kan skada katalysatorn genom ofullständigt förbrända avgaser. De två MIL-signalerna betyder i allmänhet att motorelektroniken har kopplat över till felsäkert läge för att förhindra skador på motorn. Om, till exempel, lambdaproben detekterar en för mager bränsleblandning så kan detta allvarligt skada motorn. Felsäkerhetsläget kommer då att låta dig köra till närmaste verkstad utan att riskera ytterligare skador på motorn. Ingenting mer än detta är möjligt. I den första generationen av inbyggda diagnosystem kunde man ta bort ett visat fel på displayen genom att brygga två stift i diagnoskontakten eller genom att trycka in en knapp i mittkonsolen. Det är inte längre möjligt med OBD-2.

Med lite tur kommer MIL plötsligt att slockna när du har kört ett tag till. Standarden säger att tillfälliga fel, som inte återuppstår inom ett förinställt antal körningar, inte skall tända (eller skall nollställa) MIL. Men ett datablock med all information från de viktigaste sensorerna i det ögonblick som felet upptäcktes kommer att lagras i systemet, en så kallad 'freeze frame'. Dessa freeze-frame data kan sedan läsas vid nästa servicetillfälle för att kolla vad som orsakade felet, även om felet sedan inte uppstått på ett tag. Det kan vara fallet med fel som bara händer ibland eller under vissa förhållanden.

## Felkoder

Formatet för felkoderna (fem tecken) som specas av SAE J2012 och ISO DIM 15031-6 v-isas i figur 3. OBD-2 standarden definierar den speciella mening som visas för de felkoder som får MIL att tändas. Såväl som begynnelsebokstaven 'P' för 'Powertrain' (drivlina) så kan vi också ha 'B' för Body (ka-ross), 'C' för 'Chassi' och 'U' för nät-verksfel. Om det andra tecknet är '0' har vi ett standardfel. Siffrorna '1' och '2' används för tillverkardefinierade koder där tillverkaren måste hålla sig strikt till standarden. Siffran '3' är reserverad för framtida utbyggnad av systemet. Det tredje tecknet definierar det system som övervakas i enlighet med det schema som visas i figur 3. Slutligen ger de två sista tecknen information om felet, rena textförklaringar som motsvarar dessa siffror kan slås upp i standarden. Så betyder, till exempel, kod 'P0133' att lambdaprobsensor 1 i bank 1 reagerar för långsamt.

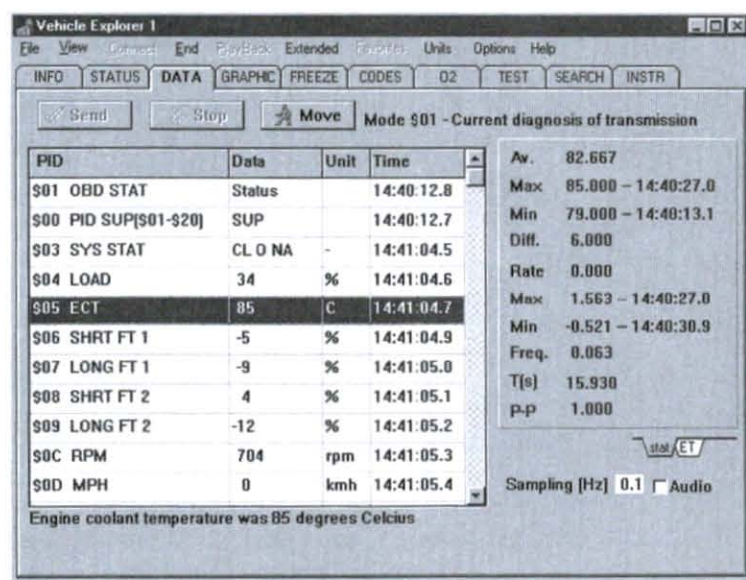
## Att läsa av data

Hur kan vi nu göra för att läsa ut data ur systemet? Som vi redan har sett i tabell 2 så är det inte bara tre olika kommunikationsprotokoll som kan finnas tillgängliga på OBD-2 kontakten beroende på vem som tillverkat fordonet, utan vi har också det problemet att inget av dessa protokoll är kompatibelt med de protokoll som man normalt hittar i en PC, som handskakningsprotokollet för serieinterfacet.

Förutom den dyrbara analysutrustning som biltillverkarna förser sina auktoriserade verkstäder med kommer allt fler och fler ekonomiska alternativ från tredjepartstillverkare. Dessa är antingen handhållna apparater som endast kan läsa felkoder eller en adapterkabel som erbjuder betydligt fler möjligheter via dedikerad PC-baserad mjukvara. OBD-2 kontakten har mycket mer att erbjuda än bara felkoder.

Ett representativt exempel på den handhållna typen är Kal KM9040 (figur 4). Som med de flesta apparater i denna kategori kan denna också radera felkoder från bilens dator. Priset i denna kategori varierar från några tusenlappar upp till över tjugo tusen kronor.

PC-baserade mjukvarulösningar, fi-



Figur 5. Det är möjligt att använda ett dedikerat program, som 'Vehicle Explorer' för att läsa felkoder med hjälp av en PC.

gur 5, som använder en adapterkabel, figur 6, finns också tillgängliga för palmtopdatorer. Det finns billiga system från drygt en tusenlapp. Det är således möjligt för de flesta att själva kunna böka runt inuti sin bildator. De PC-baserade lösningarna ger ofta tillgång till alla de data som finns tillgängliga från en OBD-2 kontakt. I enlighet med standarden arbetar systemet i nio olika mod:

- 01 ger realtidsdata
- 02 ger s.k. 'freeze frame' data
- 03 ger felkoder
- 04 raderar felkoder och lagrade värden
- 05 ger resultatet av Lambdaprobens självtest
- 06 ger självtestresultaten för system som inte övervakas kontinuerligt
- 07 ger självtestresultaten för system som övervakas kontinuerligt
- 08 dedikerat kontrollmod
- 09 fråga om fordonsdata: information såsom mjukvaruversion och VIN (Vehicle Identification Number, fordonets ID-nummer)

Det är inte alla fordon som stöder alla mod. Inom varje mod lämnar fordonsdatorn en parameteridentifiering (PID) som ett hexadecimalt tal. Om vi, till exempel, skickar bytesekvensen 01 00 ber vi att få veta vilka frågor

som fordonsdatorn kan reagera för och vilka data som kan lämnas. Efter det att själva modet har valts, i detta fall 01, kommer PID 00 att avgöra vilka ytterligare kommandon som förstås i detta mod och som kommer att besvaras.

Vi är nu redan inne i mitten av detaljerna för protokollet, som inte är viktigt för den som använder färdiga mjukvaruprodukter. Det relativt enkla

formatet hos OBD-2 protokollet, som vi har visat, betyder emellertid att det är möjligt för någon med programmeringsfärdighet att skriva egna program. Allt som behövs är att kommunikationsprotokollet för OBD-2 (41,6 kHz när det gäller PWM, 10,4 kHz för VPW och ISO 9141) översätts till ett format som är lämpligt för det seriella formatet på en PC. Lämpliga färdiga adapterkablar finns redan tillgängliga och även förprogrammerade mikrokontrollers (till exempel från PIC-familjen) som gör att man kan bygga sin egen adapter med en enkel hjärpkrets (se figur 7). Vi kommer att beskriva ett sådant projekt inom en nära framtid.

## Är det värt det?

Alla som någon gång har gått igenom den dyrbara processen att få en ny lambdapro (katalysator) monterad i sin bil har förmodligen undrat om problemet verkligen låg i katalysatorn eller om det kunde ha varit en lös kabel. Nästan varenda verkstadsundersökning som utförts av motororganisationer eller motorjournalister visar att kompletta sensorsystem har bytts ut när problemet helt enkelt varit en lös kabel. Ett diagnosverktyg kan således betala sig mycket snabbt.

Tabell 3 visar de realtidsdata som kan läsas i mod 01. Det är inte alla värden som finns tillgängliga i alla fordon, så har till exempel inte alla fordon lambdaprobar. Det beräknade belastningsvärdet är inte den effekt som är direkt tillgänglig från motorn utan snarare en dimensionslös storhet beräknad som den momentana luftflödeshasigheten



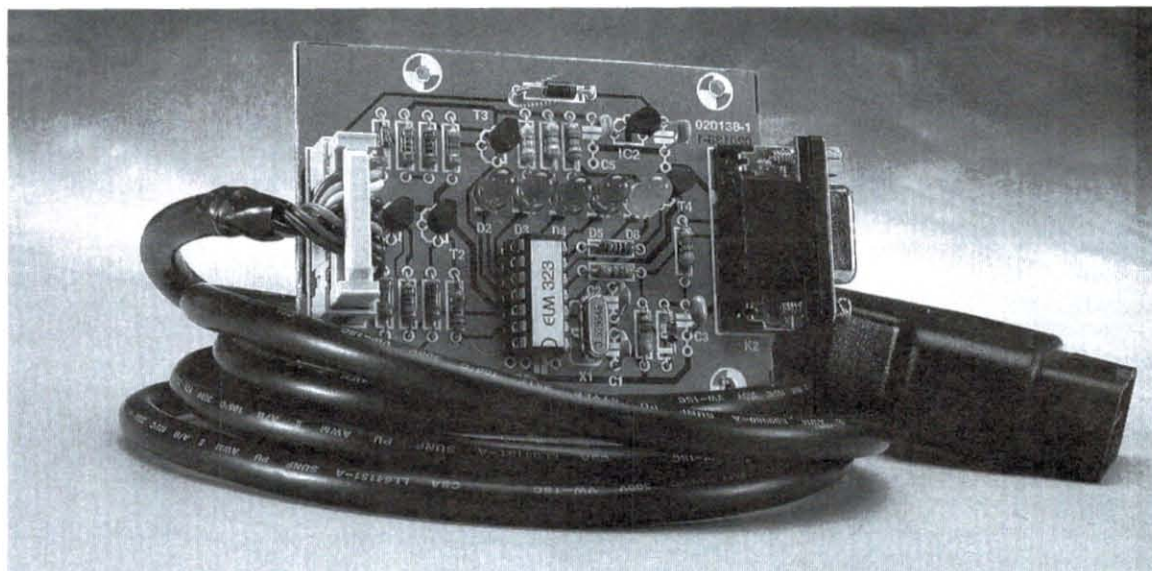
Figur 6. För att använda PC-baserad mjukvara behövs det en lämplig adapterkabel, som inkluderar en mikrokontroller som anpassar de två interfacen till varandra. (Källa: Onboard Diagnostics).

delad med den maximala luftflöde-  
shastigheten korrigerad för höjd  
över havet. Värdena kan lagras  
kontinuerligt i en bärbar dator  
och sedan visas. PC-datorn kan  
presentera data visuellt som kur-  
vor (figur 8) eller  
som instrumen-  
tavläsning. Ib-  
land är det  
möjligt att sätta  
triggers som får

dataregistreringen att starta eller stoppa  
när vissa värden går över eller under inställda  
tröskelvärden. Normalt kommer det  
freeze-frame datablock som vi redan nämnt  
(mod 02) att lagra samma information.  
I mod 04 kan en registrerad felkod raderas.  
Detta skall göras med försiktighet eftersom  
detta också raderar så kallad kalibreringsda-  
ta från fordonets dator. Kalibreringsdata är in-  
formation som samlats in av fordonets data-  
system själv för att kompensera för toleranser  
i sensorerna samt produktionsvariationer och  
förslitning i motorn. Om dessa datavärden  
raderas kommer de att automatiskt regener-  
eras när du kör igen. Detta betyder att bilens  
datavisning inte skall tas på fullt allvar omedelbart  
efter det att felkoden och kalibreringsdata raderats.

Slutligen några korta ord till alla er som hop-  
pas att kunna använda fordonsdatorn för att  
trimma bilen och få ut mer effekt ur den. Rikt-  
linjerna från EU stadgar att tillverkaren skall  
vidtaga de mått och steg som är nödvändiga  
för att 'programmerbara datorkoder eller ar-  
betsparametrar skall skyddas mot obehörig  
åtkomst'. Detta åstadkommer man normalt  
genom att använda kryptografisk processing  
och inkapsling av det relevanta chipen i mo-  
toreslektroniken. PC-baserade OBD-2 system  
inkluderar inte kryptografisk processing.

(020138-1)



**Figur 7. Vår version av en OBD-2-till-RS232 converter. Denna design utnyttjar en PIC mikrokontroller.**

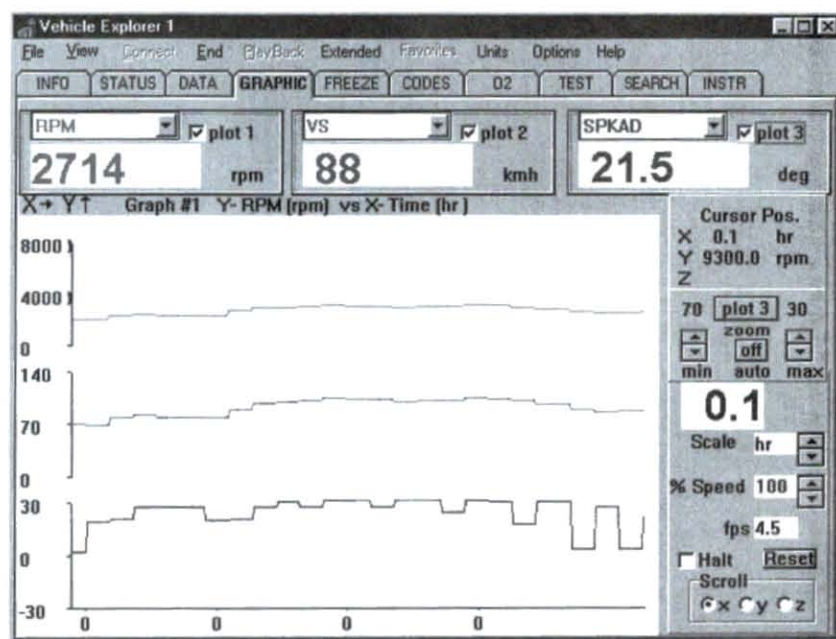
## Referenser och weblänkar:

[1] EU Guidelines EC70/220,  
EC98/69, EC99/102, ECE R83  
[http://europa.eu.int/index\\_de.htm](http://europa.eu.int/index_de.htm)

[2] ISO och SAE standarder  
[www.beuth.de](http://www.beuth.de)

[3] Tredjeparts OBD-2 tillverkare  
av diagnosutrustning

[www.actron.com](http://www.actron.com)  
[www.auterraweb.com](http://www.auterraweb.com)  
[www.autoxray.com](http://www.autoxray.com)  
[www.abcwc.net/accounts/quanta/](http://www.abcwc.net/accounts/quanta/)  
[www.eichstaedt-electronics.com](http://www.eichstaedt-electronics.com)  
[www.obd2.com](http://www.obd2.com)  
[www.obd-2.com](http://www.obd-2.com)  
[www.obd-2.de](http://www.obd-2.de)  
[www.kalequip.com](http://www.kalequip.com)  
[www.mavericktechnology.co.uk](http://www.mavericktechnology.co.uk)  
[www.snapon.com](http://www.snapon.com)



**Figur 8. De PC-baserade programmen kan presentera data visuellt som kurvor eller som instrumentvisning.**

# Höghastighets kontrollerkort (2)

## Del 2: Kretsbeskrivning och själva bygget

Design av Luc Lemmens

I förra numret presenterade vi er för DS89C420, den yngste medlemmen i den mikrokontrollerfamilj som produceras av våra mycket aktiva vänner hos Dallas Semiconductor. Denna kontroller, trots att den är fullt bakåtkompatibel med andra enheter ur 8051-serien, är mycket snabbare och erbjuder en del extra. Denna månad fortsätter vi med att presentera ett kort, som med hjälp av gratis mjukvara från Dallas bildar grunden för ett kraftfullt och ändå överkomligt utvecklingssystem.

80951-serien är fortfarande oerhört populär bland såväl hobbyister som proffs. Sedan introduktionen av denna mikrokontroller av Intel under slutet av 80-talet har en veritabel skattkammare med utvecklingsverktyg, program och subrutinsbibliotek öppnats upp, som är tillgänglig gratis för alla som har tillgång till Internet. Även om den ursprungliga kontrollern har byggts ut med perifera funktioner, och dess klockfrekvens har höjts av olika tillverkare, så har grundarkitekturen i 8051-kärnan behållits i stort sett oförändrad. I denna kontroller divideras frekvensen hos den externa klockoscillatorn med 12, helt enkelt därför att en maskincykel behöver detta antal klockcykler. Dallas Semiconductor har nu för första gången sett över den interna strukturen i processorkärnan för att eliminera 12 klockprescalern. I en DS89C420 är en klockcykel lika med en maskincykel och den nya kontrollern är därför tolv gånger snabbare än sina föregångare.

I del 1 av denna artikelserie har vi redan pekat på de extra egenskaper som erbjuds av C420 medan den förblir kompatibel med äldre applikationer. En 8051 eller en 8052 kan ersättas med en DS89C420 utan problem, den

nya IC:n är benkompatibel och kan köra äldre mjukvara, även om vissa timingaspekter måste tas hänsyn till tack vare den mycket högre hastigheten.

Förekomsten av ett Flashminne hos C420 gör chipet mycket lämpligt för utveckling, avbuggning och testning av ny applikationsmjukvara och ny hårdvara. Flashminnet kan programmeras många gånger medan ett monitorprogram (som finns i ett reserverat område i chipminnet) tillåter att ett nytt program kan laddas in i kontrollern via serieporten på en PC utan minsta problem. Det är nu hög tid att se hur vårt höghastighetskontrollerkort ser ut i verkligheten.

### Hårdvarubeskrivning

Kretsschemat för utvecklingskortet för DS89C420 visas i figur 1. Vid en första titt verkar det inte vara något ovanligt, ytterligare ett 8051-kort kanske du tänker. Frånvaron av ett EPROM kompenseras av ett (osyn-

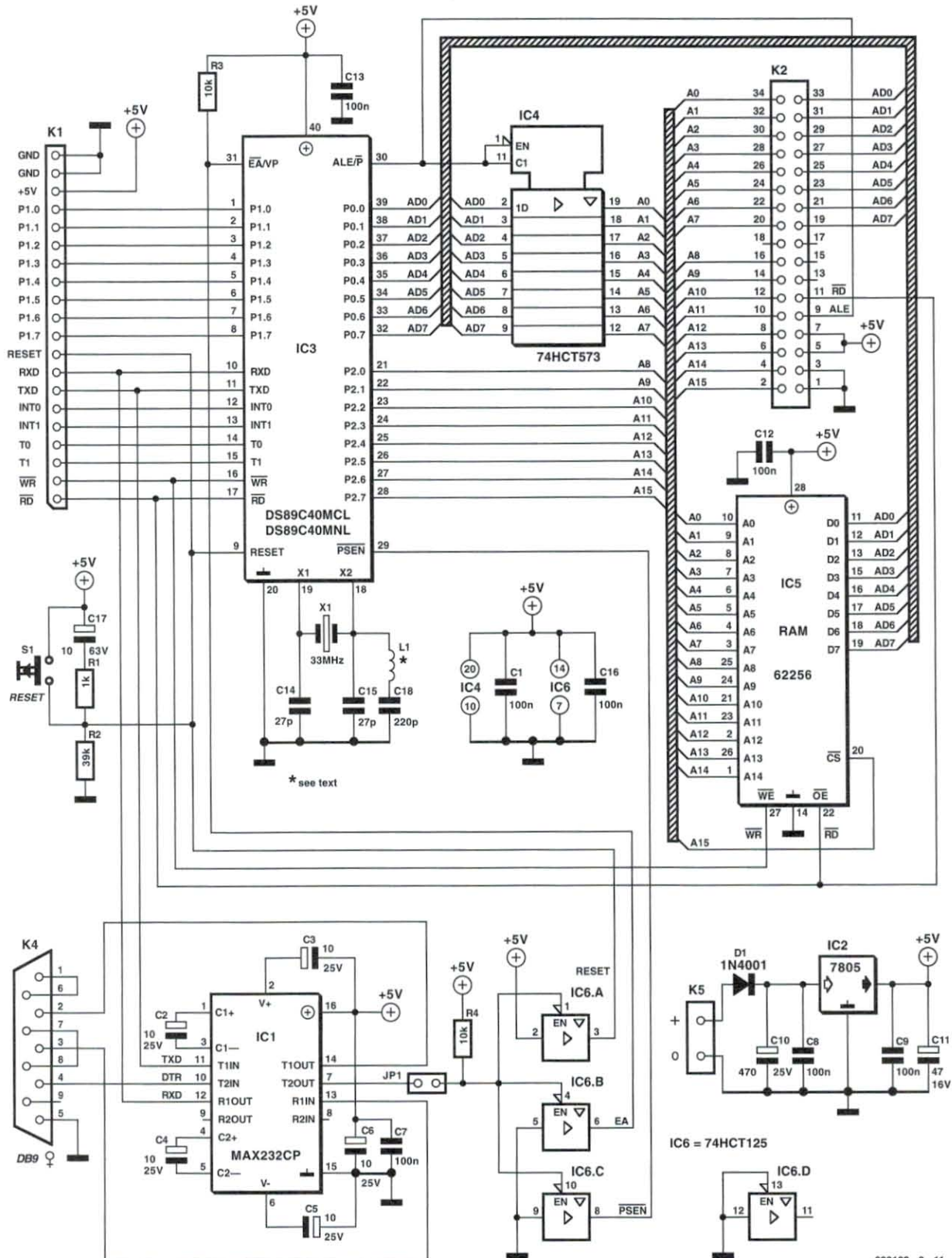
ligt) 16 kByte Flashminne inbyggt i C420. Även om det inte finns något som hindrar dig från att bygga ut kretsen med ett externt programminne till en maximal storlek på 64 kByte, så skall det sägas att programmeringen av kortet då blir betydligt svårare än med standardkonfigurationen. Det interna minnet har också den inbyggda fördelen av att köras med full kontrollerhastighet och du behöver inte heller bekymra dig för timingproblem om du använder dig av en minneskomponent som inte är så snabb, men mera om det längre fram.

DS89C420 är fullt benkompatibel med andra mikrokontroller ut 8051-serien och vi räknar därför med att bentilldelningen är välkänd för de av våra läsare som har sett (och byggt) några av våra tidigare projekt baserade på 8051-an (och dess avläggare). Det är endast klockoscillatorkretsen som skiljer sig något om kontrollern skall arbeta med relativt höga kristallfrekvenser. Som du kanske vet

arbetar kristaller upp till 25 MHz med sin grundfrekvens, medan 'tredje övertonen' är mer vanlig med kristaller för högre frekvenser. Den senare konfigurationen kräver att grund-

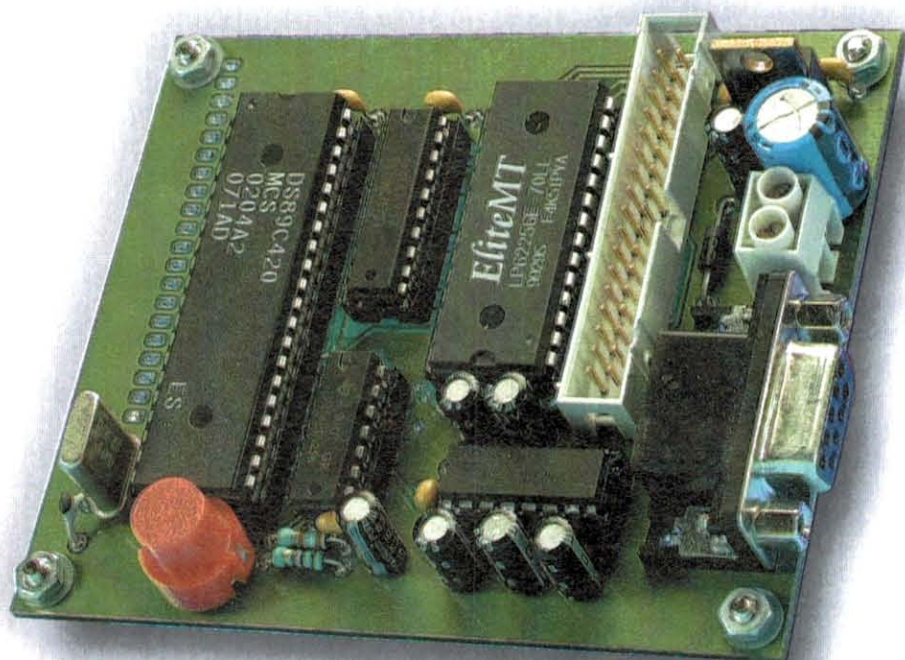
frekvensmodet undertrycks. Om man inte gör detta kommer en kristall som specificerats för, t.ex. 33 MHz, att glatt arbeta vid sin grundfrekvens, 11 MHz. Därför har en ex-

tra L/C-kombination (L1 och C18) adderats till den vanliga oscillatorlayouten. Det är riktigt, de två extra komponenterna tvingar kristallen att svänga vid sin tredje överton (och ditt program att köras med maximal hastighet).



020102 - 2 - 11

Figur 1. Kretsschemat över DS89C420 utvecklingskort.



Tabell 1 visar det maxvärde för L1 som kan användas vid några vanliga klockfrekvenser. Adresslatchen (IC4) och RAM (IC5) är standard för en applikationskrets med 8051. Du kan också se att de mest relevanta kontrollinjerna är åtkomliga för anslutning till världen utanför via K1 och K2. Dessa två kontakter gör det särskilt enkelt att ansluta extern hårdvara till utvecklingskortet.

En matningsspänning på mellan 9 och 15 VDC kan anslutas till kortet via K5. En billig nätdapter (batterieliminatör) är perfekt för detta jobb. Den råa inspanningen på 9-15 V tas ner och stabiliseras till en ren matning på +5 V av IC2. Denna spänning används av mikrokontrollern och den övriga hårdvaran. Dioden D1 fungerar som skydd mot omvänd polaritet och förhindrar att kretsen skadas om någon (självklart inte du) av misstag ansluter ingångsspänningen åt fel håll på K5.

Resetkretsen runt C17, S1, R1 och R2 är också klassisk. C17 och R2 ger en tillförlitlig reset-signal för mikrokontrollern när strömmen kopplas på eller när S1 trycks in. Även om R1 begränsar den ström som flyter genom C17 när

S1 är sluten så påverkar detta i stort sett inte resetsignalen.

Serieinterfacet med MAX232 (IC1) har använts i många andra mikrokontrollerkretsar. Det finns dock en skillnad den här gången. DTR-benet (Data Terminal Ready) på RS232-interfacet används för att driva tre 3-tillståndsbuffrar via IC1. När dessa buffrar är aktiverade (vilket sker när DTR på K4 går Hög medan ben 7 på IC1 följaktligen går Lågt) signalerar mikroprocessorn att RESET, EA (External Access) och PSEN (Program Storage Enable) är switchade till ett ovanligt tillstånd. Notera att PSEN normalt är en utgångssignal så att dra den till en fast logisk nivå (0 V i detta fall) är anmärkningsvärt.

Den resulterande kombinationen av logiska nivåer vid de tre kontrollbenen tvingar 89C420 att starta det interna monitorprogrammet som, bland annat, gör att du kan kommunicera med kontrollern via serieporten, radera programminnet, programmera det, samt läsa och verifiera det. Monitorprogrammet är lokaliserat i ett speciellt minnesområde inuti mikrokontrollern, som endast adresseras om C420 är switchat till 'monitormod'. Med andra ord, monitorn körs inte från det reguljära interna minnet. Det kan därför bäst beskrivas som ett 'boot ROM' för mikrokontrollern, som normalt är

osynligt men som förblir aktivt så länge som de tre buffrarna IC6a, IC6b och IC6c är aktiverade.

Så fort de tre buffertutgångarna återgår till hög impedans kommer 89C420 att uppföra sig som en vanlig mikrokontroller igen, som troget utför de instruktioner som hittas i programminnet.

Monitorn switchar från när kortslutningsbygeln JP1 tas bort. Detta kan vara nödvändigt om kontrollerkortet kommunicerar med en PC med hjälp av ett 'vanligt' terminalprogram, som HyperTerminalen som finns i Windows. Sådana program kan hålla DTR permanent Hög vilken gör att mikrokontrollern fastnar i monitor-mod och du kan inte köra dina applikationsprogram.

## Själva bygget

Med den målgrupp vi vänder oss till med detta projekt så kan vi vara rätt kortfattade när det gäller att bygga kontrollerkortet. Du kan inte göra mycket fel om du använder det dubbelsidiga, genompläterade kort som visas i figur 2 (best.nummer hos oss är 020102-1) tillsammans med komponentplaceringsritningen på kortet och komponentlistan. Vi rekommenderar att du monterar IC-kretsarna i hållare. Tänk på polariteten hos dioden och elektrolytkondensatorerna. Jp1 kan vara en kortslutningsbygel, men också en enkel till/från-omkopplare.

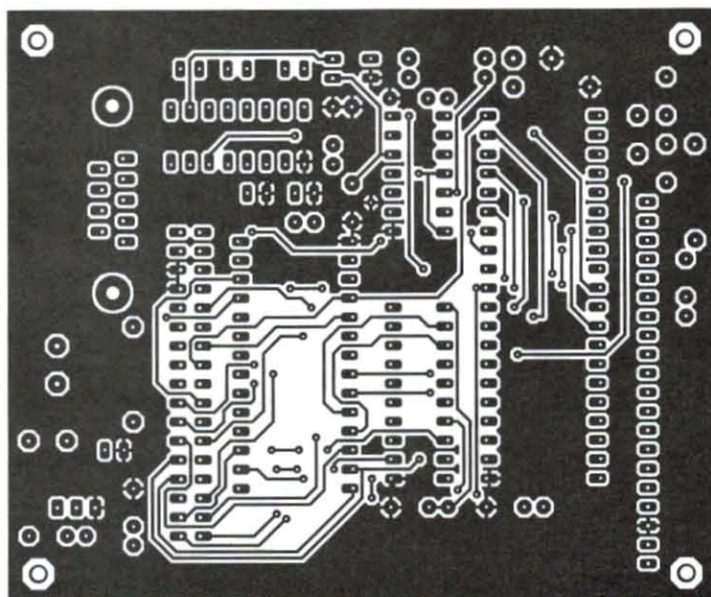
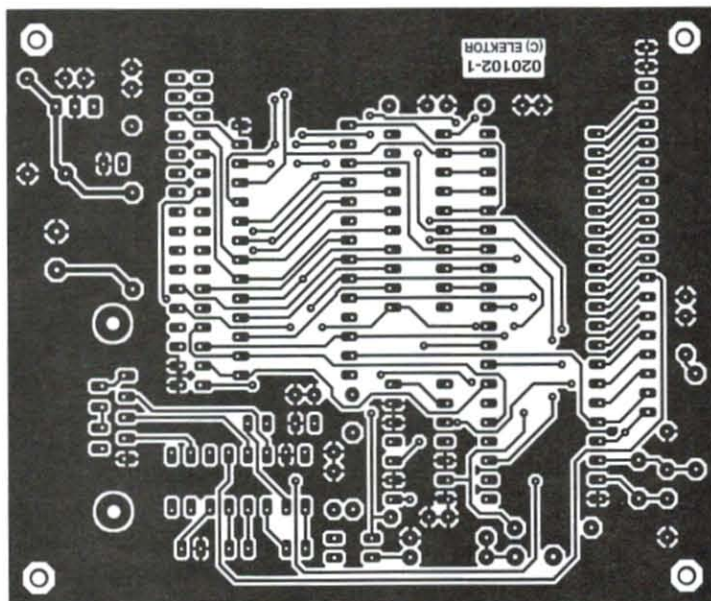
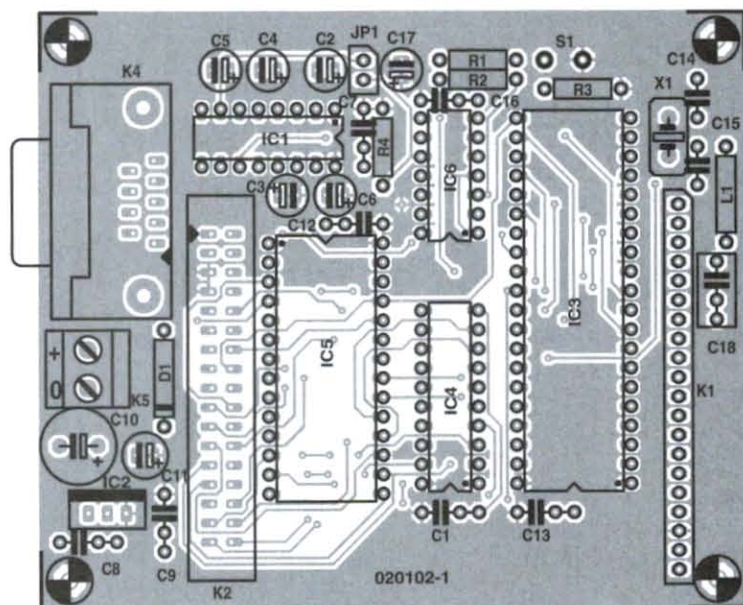
## Mjukvara

Monitorprogrammet gör att mikrokontrollern kan kommunicera med världen utanför. På PC-sidan behöver du också viss mjukvara för att kunna programmera eller läsa DS89C420. På Dallas ftp-sida vid [ftp://ftp.dalsemi.com/pub/microcontroller](http://ftp.dalsemi.com/pub/microcontroller) finns det två gratisprogram för detta ändamål **Microcontroller Toolkit** (MTKbeta.zip) och **Loader420** (loader420.zip). Även om figur 3 visar de båda programmen samtidigt så är det bara ett av dem som behövs i praktiken. Programmen är jämförbara vad gäller funktionalitet, men vi föredrar Loader420 eftersom detta har skrivits speciellt för vår mikrokontroller. MTK är också lämpligt för andra mikrokontrollers från Dallas Semiconductors. Det inne-

## Tabell 1.

Maximalvärde för L1 med olika kristallfrekvenser

| Kristallfrekvens | L1 (max.) |
|------------------|-----------|
| 20 MHz           | 12 mH     |
| 25 MHz           | 8.2 mH    |
| 32 MHz           | 4.7 mH    |



håller därför fler inställningar och alternativ och är därför inte lika lätt att använda jämfört med Loader420. MTK gav oss några 'blåskärmar' när scrollfönstren blev för överbelastade. Detta noterades med det programexempel vi kommer att diskutera längre fram, där ett externt RAM läses, och innehållet samt adressen för varje minnesplats visas på skärmen. När fönstret fylls med data kraschar till slut MTK och Windows poppar upp den ökända 'blå skärmen'. Inte speciellt uppmuntrande förstås och tillräckligt skäl för att kasta ut MKT och fortsätta arbeta med Loader420 i stället.

## Uppkoppling och initialtest

Anslut det färdigbyggda DS89C420-kortet till en ledig RS232-port på din PC med hjälp av en 9-polig icke-korsad D-sub expansionska-

## KOMPONENTLISTA

### Motstånd:

R1 = 1k $\Omega$   
R2 = 39k $\Omega$   
R3, R4 = 10k $\Omega$

### Kondensatorer:

C1, C7, C8, C9, C12, C13, C16 = 100nF  
C2-C6, C17 = 10 $\mu$ F 25V radial  
C10 = 470 $\mu$ F 25V radial  
C11 = 47 $\mu$ F 16V radial  
C14, C15 = 27pF  
C18 = 220pF

### Spolar:

L1 = se text

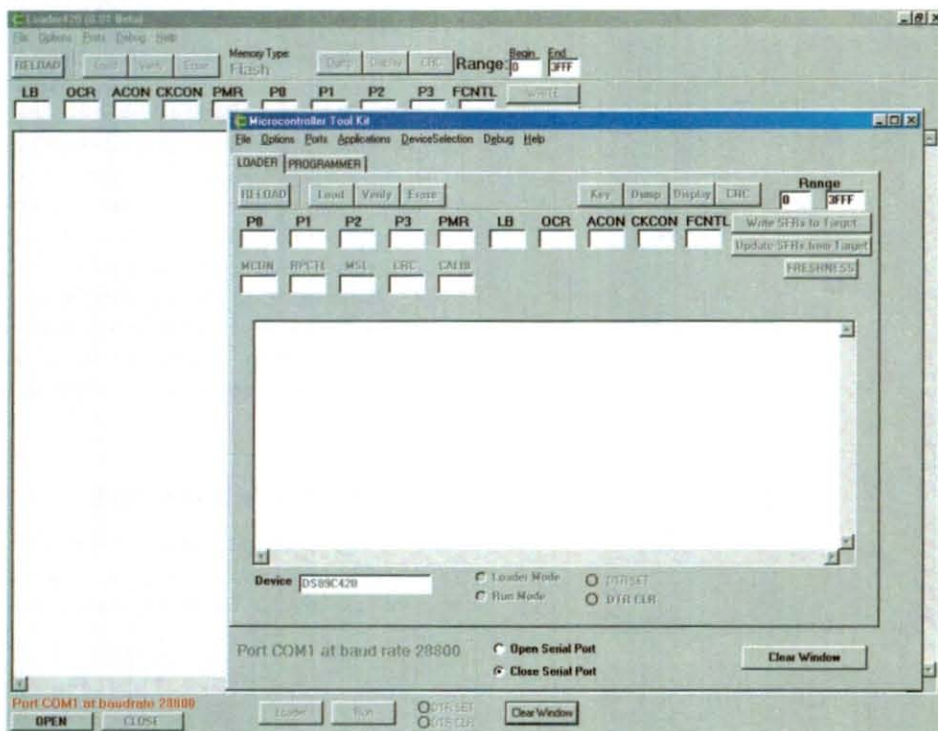
### Halvledare:

D1 = 1N4001  
IC1 = MAX232CP  
IC2 = 7805  
IC3 = DS89C420-MCL eller -MNL (DIP40, Dallas Semiconductor/Maxim)  
IC4 = 74HCT573  
IC5 = 62256 (8 x 32k, hast. minst 100ns)  
IC6 = 74HCT125

### Övrigt:

JP1 = kortslutningsbygel  
K1 = 20-pol SIL stiftlist  
K2 = 34-pol boxheader  
K4 = 9-pol sub-D kontakt (hona), vinklade stift, för kretskort  
K5 = 2-pol kopplingsplint för kretskort, raster 5mm  
S1 = slutande tryckknapp  
X1 = 33MHz kristall  
Kretskort 020102-1 (se Läsaerservice)

Figur 2. Dubbsidigt, genompläterat kort som finns att köpa från oss (se Läsaerservice).



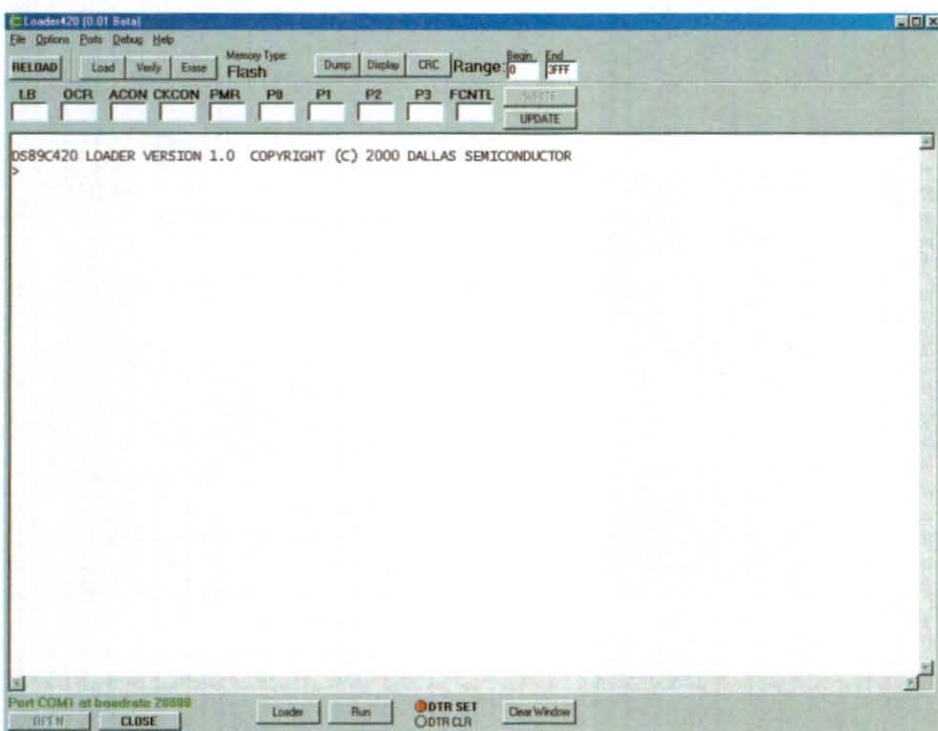
Figur 3. Båda PC-programmen på samma skärm (du behöver bara ett).

bel (inte en nollmodemkabel). Kontrollerkortet kan försörjas från vilken billig 12-VDC nätadapter som helst. Starta därefter Loader420 i din PC. Första gången programmet körs kommer Loader420 att välja COM1: som sin serieport. Detta kan du ändra om du vill i menyn *Ports/Select Port Settings*. Där kan du också ändra den baudrate som används på

serielänken. När du avslutar Loader420 sparas dina kommunikationsinställningar och du behöver därför inte ange dem igen nästa session. Det skall noteras att monitorprogrammet allmänt kallas för 'Serial Bootstrap Loader' i dokumentationen för DS89C420. För att undvika

sammanblandning med Loader420 som körs på PC-n, så kommer vi att fortsätta använda termen 'monitor'. Inte så överraskande så öppnas seriekanalen genom att klicka på knappen *Open*. Detta etablerar dock ännu inte någon kommunikation med mikrokontrollern. Faktum är att det inte händer någonting förrän du klickar på knappen *Loader*. DTR-linjen på serieporten dras då hög och den tidigare nämnda buffert-trippeln switchar kontrollern till monitormod. DS89C420 kommer att svara genom att producera en kommandoprompt såsom visas i figur 4. Mikrokontrollern kommer att automatiskt avgöra hastigheten (baudrate) vid kommunikationen med PC-n. Monitorprogrammet kommer faktiskt att försöka sig på ett antal olika baud-rate genom trial-and-error adaption av det förladdade värdet i Timer1. Om det inte kommer fram någon prompt på din skärm så fungerar uppenbart inte kommunikationen med kontrollern. I detta olyckliga fall så måste du kolla kortet för eventuella fel (kolla speciellt elektrolyterna runt MAX232), strömförsörjningen, den fysiska länken till PC-n och se till att du använder rätt kommunikationsport. Om allt verkar vara OK och DS89C420 fortfarande vägrar att kommunicera när du klickar på *Loader*-knappen, måste du använda en annan baudrate på din PC. Om du inte gillar att göra beräkningar så kan du helt enkelt prova olika inställningar tills DS89C420 hälsar dig välkommen med sin kommandoprompt. Denna trial-and-errorprocess är enkel, men den kräver att serieporten stängs med *Close*-knappen, ny baudrate ställs in och porten öppnas igen genom att klicka på *Loader*. Om du inte följer detta sätt kommer den gamla inställningen fortfarande att användas och du blir inte ett dugg klokare.

Läsare som kommer väl överens med sina fickräknare är kanske glada att veta att monitorprogrammet klarar av att detektera baudraten (inom vissa gränser) och 'justera' mikrokontrollern till motsvarande inställningar. Monitorn utnyttjar mikrokontrollerns serieport 0 i Mode 1 (asynkron kommunikation, 1 startbit, 8 databitar, ingen paritet, 1 stoppbit, full duplex) med Timer1 i 8-bitars autoreload mod och PCON.7 logiskt Hög.



Figur 4. Loader420 i action. Notera kommandoprompten.

Vad detta egentligen betyder är att förhållandet mellan baudrate, kristallfrekvens och Reloadvärde kan uttryckas på följande sätt:

$$\text{Serial\_Loader\_BaudRate} = \frac{\text{CrystalFrequency}}{192 \times (256 - \text{Timer Reload})}$$

(timer-reload värden försökta av Loader:

FF, FE, FD, FC, FB, FA, F8, F6, F5, F4, F3, F0, EC, EA, E8, E6, E0, DD, D8, D4, D0, CC, C0, BA, B0, A8, A0, 98, 80, 60, 40).

Monitorprogrammet kommer att försöka använda de indikerade talen för Timer1 Reloadvärden för att kunna finna ut vilket av dessa som ger kommunikation med PC:n.

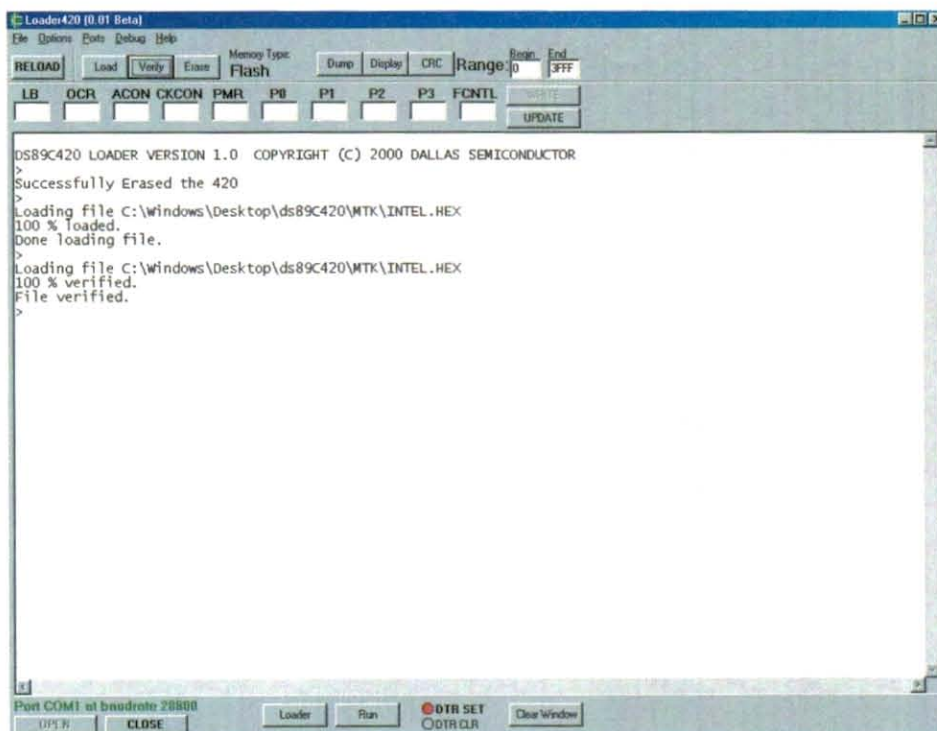
## Let's get cracking!

Om du nu har löst eventuella problem du kan ha stött på med den seriella kommunikationen så inbjuder den kommandoprompt som produceras av DS89C420 till lite seriös programmering. Även om monitorprogrammet känner igen kommandon som skrivits in efter prompten (se sektionen *Command Line Interface*, sida 183 ff i DS89C420 User's Manual), så är Loader420 ett bekvämt alternativ då det gör alla instruktioner åtkomliga via menyer och knappar. Enkelhet och lättanvändning börjar här!

Loader420 har ett antal funktioner för att inspektera och redigera minnesområden, register och allmänna DS89C420-inställningar. Dess grundfunktion är dock att programmera det inbyggda Flashminnet med det ultimata målet att få processorn att köra dina egna program.

Loader420 förväntar sig en fil som överensstämmer med industristandarden **Intel Hex** formatet, som stöds av de flesta, dock inte alla, assemblers och kompilare. För våra experiment har vi använt oss av en gammal assembler som vi använde för en assemblerkurs för så länge sedan som 1992 (började i nr 5/1992). Denna assembler finns fortfarande tillgänglig från oss, order **1661**.

Om det redan finns ett önskat program i Flashminnet hos DS89C420 så måste detta först tas bort med hjälp av *Erase* innan en ny applikation kan



Figur 5. Applikationsexemplet har laddats in i Flashminnet.

laddas. Om du klickar på *Load* kommer det fram en dialogruta där du kan välja önskad hexfil. Loader420 kommer att programmera denna fil in i mikrokontrollern och därefter kan objekt-koden exekveras genom att klicka på *Run*. Detta tvingar DTR att gå Låg, mikrokontrollern går ur monitor-mod och börjar köra den objekt-kod som den hittar i sitt Flashminne vid adress 0H (dvs resetvektorn).

## En enkel applikation: RAM-testning

Det utvecklingskort som vi diskuterar här har inga LED eller andra indikatorer som låter ett enkelt testprogram köra en funktionstest av hårdvaran. Det program som vi beskriver nedan skriver till det externa RAM-et, läser tillbaka den minnesplats som senast skrevs till och skriver adressen och dess innehåll tillbaka till din skärm. Listning 1 visar det enkla RAM-testprogram som också visar korrekt funktion av kommunikationen mellan processorn och den externa RAM-enheten.

Som vi redan sagt i del 1 av denna artikelserie så kan hastigheten hos DS89C420 ge upphov till timing-problem i läs eller skrivoperationer till eller från externa minnesenheter.

Lyckligtvis tillåter kontrollern att externa periferienheter kan adresseras med relativt låg hastighet. Tre bitar i CKCON-registret kan användas för att 'sträcka ut' en minnesåtkomstcykel, och med den, längden på signalerna RD WR, ALE och PSEN och därmed låta långsammare externa minnesenheter användas utan problem. När dessa bitar är rätt inställda hanteras allting automatiskt och är ingenting att bry sig om. Processen beskrivs ingående i en separat applikationsnot AN26, *Ultra High-Speed Microcontroller Memory Interfacing*, som du kan hitta på [http://dbserv.maxim-ic.com/app-notes.cfm?appnote\\_number=954](http://dbserv.maxim-ic.com/app-notes.cfm?appnote_number=954).

Med de flesta moderna RAM-IC som har åtkomsttider på 100 ns och mindre (som det vi föreslår för position IC5) så blir det inga problem.

Programexemplet läser och skriver i externt RAM genom att inte använda mycket mer än (den förväntade) instruktionen MOVX. Längden på programmet beror i huvudsak på rutiner för den seriella kommunikationen mellan DS89C420-chipet och Loader420 som körs i PC:n. Programmet kan delas in i två delar:

- konfigurera serieporten på DS89C420 så att den kan kommunicera med PC:n;
- översätta adresser och data till ASCII-format så att de verkar vettiga på en PC.

Båda styckena kod har tagits från det monitorprogram som skrivits av M. Ohsmann för

## Listning 1. Enkel RAM-test.

```
; ***** FILE RAMTEST.A51 *****
; Register addresses
DPL EQU 082H ; DPTR, 2 registers
DPH EQU 083H ;
PSW EQU 0D0H ; program status word
ACC EQU 0E0H ; accumulator
PCON EQU 087H
TCON EQU 088H
TMOD EQU 089H
TL1 EQU 08BH ; Timer1 registers
TH1 EQU 08DH
SCON EQU 098H
SBUF EQU 099H

DPS EQU 086H ; Data Pointer Select, SFR in DS89C420

; reload for Timer1, baudrate generator
V24SPD EQU 256-5 ; speed for V24

;
ORG 00H ; program runs from 00H (reset vector)
AJMP START
ORG 030H
CNT1 DS 1
ORG 200H
START MOV PSW,#0 ; reset register banks
MOV PCON,#080H ; SMOD=1
MOV TMOD,#22H ; modes are timer
MOV TH1,#V24SPD ; preload value
MOV TL1,#V24SPD
SETB TCON.6 ; start counter
MOV SCON,#052H ; mode 1 , Enable receiver=10H

MOV A,#0
MOV DPH,A
MOV DPL,A
;The Data Pointer Select bits are set to use both DPTR's
;bit 5: toggle select, auto toggle between DPTR's after MOVX
;bit 4: AID, auto increment DPTR after MOVX
;bit 0: SEL is set to select DPTR1, DPTR0 is already initialised
MOV A,DPS
ORL A,#00110001B
MOV DPS,A
;now initialise DPTR1
MOV A,#0
MOV DPH,A
MOV DPL,A

LOOP MOV A,DPH ;#0AAH
MOVBX @DPTR,A ; Write using DPTR1
MOV A,DPH
ACALL BYTE
MOV A,DPL
ACALL BYTE
MOV A,#':'
ACALL SND
MOVBX A,@DPTR ; Read using DPTR0
ACALL BYTE
```

den tidigare nämnda MCS51-kursen 1992. De rutiner som är nödvändiga för den andra funktionen kan hittas i slutet av listningen, med början vid etiketten BYTE.

### Seriell portkonfiguration

Att 89C420 är tolv gånger snabbare än sina föregångare (vid samma klockfrekvens) har ingen effekt på den hastighet med vilken de interna tidskretsarna (timers) klockas (med standardinställningar och efter en reset). Detta beror på att 12 prescalern används för, bland annat, bakåtkompabilitet med äldre processorer för seriella portkonfigurationsrutiner. Även om närvaron och användningen av prescalern inte är något nytt sedan trimmade 8051-or dykt upp, kan en kort förklaring vara i sin ordning för att få allt att fungera rätt.

Om vi använder de inställningar som definieras från etiketten START och framåt så kan förhållandet mellan kristallfrekvensen, baudrate och Reloadvärdet uttryckas som:

$$\text{Baudrate} = \text{kristallfrekvens} / ((12 \times 16) \times (256 - \text{Reload}))$$

Om vi antar att en kristallfrekvens på 27 MHz används och det krävs en baudrate på 28 800 bitar/s på DS89C420 (Loader420 inställning) så ser du att Reloadvärdet blir 5 (Listning 1, definition av V24SPD). Uppenbart kan du redigera denna rad om du använder en annan kristallfrekvens eller en annan baudrate i Loader420.

### En andra datapekare

Det kan möjligen vara lite för mycket av det goda i detta enkla applikationsprogram att vi använder den extra datapekaren som finns i DS89C420. Pekaren DPTR0, som finns tillgänglig i alla MCS51-kontroller, används för den adress som vi läser från, medan DPTR1 pekar på den adress som skall skrivas till. Dessutom inkrementeras den aktiva datapekaren automatiskt efter en MOVX-instruktion. Därefter väljs den alternativa datapekaren (Toggle Select biten). Det höga bytet hos DPTR1 skrivs in i det externa RAM-et. När adress 7FFFH har skrivits till och lästs tillbaka in i processorn har alla minnesplatser i IC5 hanterats och programmet går in i den ändlösa loop som kallas FOREVER.

## Sammanfattning

DS9C420 är (1) en superb mikrokontroller, (2) 100% kompatibel med de 'väl sedda' MCS51-kontrollers som vi kommit att tycka så mycket om, (3) tolv gånger så snabb som sina föregångare och (4) har ett internt Flash-minne för enkel omprogrammering. Dessa, och andra, egenskaper gör att det mikrokontrollerkort som vi beskriver här är eminent lämpligt att använda vid utveckling av nya, spännande applikationer för 8051-(lika) kontrollers.

Det antal ytterligare egenskaper som hittas hos DS89C420 är så många att de inte kan diskuteras inom det begränsade utrymme som vi har till förfogande. De av er som vill 'veta allt' uppmanar vi att hämta hem datablad, användarmanual och applikationsnoter från Maxim/Dallas Semiconductors websida.

(020102-2)

```

ACALL CRLF
MOV A, DPH
CJNE A, #80H, LOOP
FOREVER SJMP FOREVER

BYTE PUSH ACC ; send BYTE hexadecimal , destroy ACC only
SWAP A
LCALL NIBBLE
POP ACC
NIBBLE ANL A, #0FH
ADD A, #246
JC HEXOUT
ADD A, #58
SJMP SND
HEXOUT ADD A, #65
SJMP SND

;
CRLF MOV A, #13 ; send a CRLF
LCALL SND
MOV A, #10
LCALL SND
RET

;
SND JNB SCON.1, SND ; must not destroy any reg
CLR SCON.1
MOV SBUF, A
CJNE A, #10, OK2

WAITCR MOV CNT1, #100 ; wait for slow scrolling terminals
LOP1 MOV A, #255
LOP2 DJNZ ACC, LOP2
DJNZ CNT1, LOP1
MOV A, #10
OK2 RET
END

```

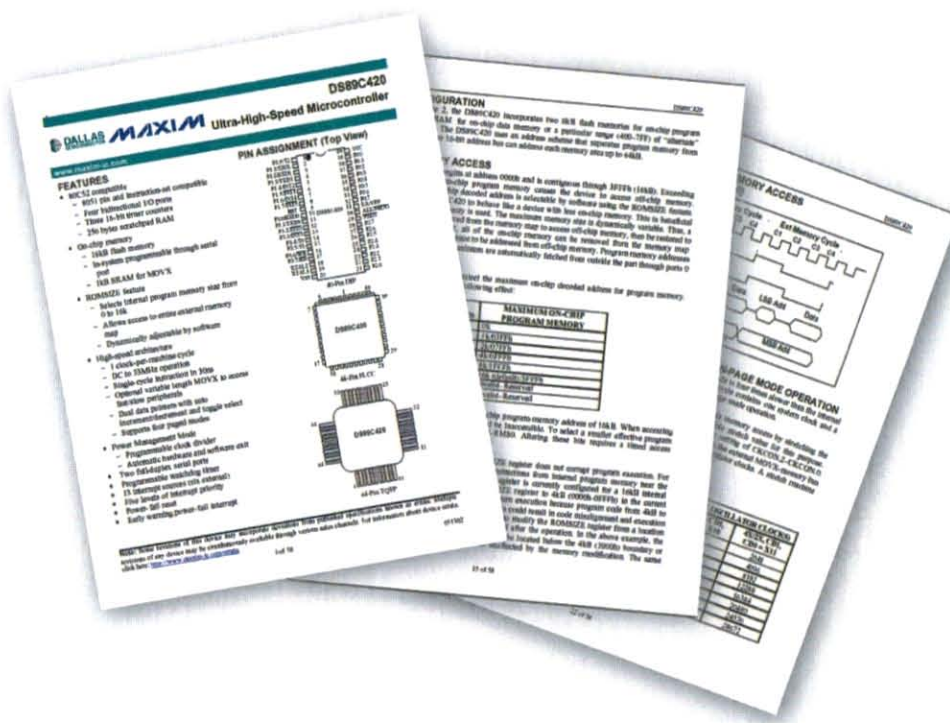
## Litteratur

DS89C420 datablad:

<http://pdfserv.maxim-ic.com/arpdf/DS89C420.pdf> DS89C420

Användarmanual:

[http://pdfserv.maxim-ic.com/arpdf/Design/89c420\\_userguide.pdf](http://pdfserv.maxim-ic.com/arpdf/Design/89c420_userguide.pdf)



# Digital HF-wattmeter med LC-display

för 1 kHz till 1 GHz

Design av Thomas Scherrer OZ2CPU

[www.webx.dk](http://www.webx.dk)

Alla radioamatörer vet hur viktigt det är med en noggrann effektmätare för HF. En wattmeter kan användas för att mäta förstärkning i förstärkare, bandbredd i filter, fältstyrka från antenner, sändareffekt, SWR, returförluster och mycket annat.



## Specifikationer

|                                   |                                                                                                           |
|-----------------------------------|-----------------------------------------------------------------------------------------------------------|
| Frekvenstäckning:                 | 1 kHz till 500 MHz (kalibrerad)<br>1 kHz till 1000 MHz (okalibrerad, endast för relativ effektmätning)    |
| Nominell ingångsimpedans:         | 50 $\Omega$                                                                                               |
| Ineffektområde:                   | -60 dBm till +30 dBm<br>(1 nanowatt till 1 watt).                                                         |
| Ineffektområde med 50-dB dämpare: | upp till 100 kwatt.                                                                                       |
| Dynamiskt område:                 | 90 dB med bra HF-skärmd låda.                                                                             |
| Upplösning:                       | 0.1 dBm (1 dBm på stapel)                                                                                 |
| Ingångens returförlust:           | @300kHz: -35dB<br>@100MHz: -27dB<br>@500MHz: -25dB<br>@ 300kHz: 1.036<br>@ 100MHz: 1.094<br>@500MHz: 1.12 |
| Ingångs-SWR:                      | $\pm 1$ dB från 1 MHz till 450 MHz.<br>$\pm 0.2$ dB vid varje kalibrerad frekvens.                        |
| Noggrannhet före kalibrering:     | 0 till 20 volt.                                                                                           |
| Efter kalibrering:                | 20 mV                                                                                                     |
| DC-spänningsmätning:              | $\pm 20$ mV                                                                                               |
| DC-spänningsupplösning:           | 9 till 20 VDC                                                                                             |
| DC-noggrannhet efter kalibrering: | utan LCD-belysning: 30 mA;<br>med normal LCD-belysning: 120 mA.                                           |
| Strömförsörjning:                 |                                                                                                           |
| Strömförbrukning:                 |                                                                                                           |

Denna HF-wattmeter använder sig av en AD8307 för att mäta effektnivå. Frontendkretsen hos AD8307 är både frekvenskompenserad och optimerad för returförluster för att ge en optimal ingångs-SWR över ett stort frekvensområde.

En förprogrammerad mikrokontroller av typ PIC16F876 med inbyggd 10-bitars analog-till-digitalomvandlare används för att konvertera den analoga utspänningen från AD8307 till digitala värden. Därefter används ett antal uppslagstabeller till att omvandla dBm-värdena till HF-spänning och HF-effekt (watt). Avläsningen av alla värden, inklusive en stapel, sker på en stor 20x2 LCD med bakgrundsbelysning. Det finns ock-

så en DC-voltmeter som kan spara min och maxvärden, plus många extra egenskaper, se mjukvarubeskrivningen längre fram i artikeln.

### Enheten decibelmilliwatt (dBm)

Inom HF-teknologin representerar 0 dBm 1 milliwatt in i 50  $\Omega$ . På samma sätt representerar, 0 dBW 1 watt in i samma impedans.

Så, +10 dBm = 10 mW; +20 dBm = 100 mW; +30 dBm = 0 dBW = 1 W och så vidare.

Termen 'dBm' används vid all professionell radioutveckling, reparation och service, och även av radioamatörer för att beskriva (relativa) effekt-

nivåer för HF. Vissa tekniker känner sig hemma med dBm-enheten, andra föredrar 'watt' och ytterligare några talar hellre om 'RMS-spänning'. Denna wattmeter visar alla tre enheterna samtidigt.

### Om AD8307

Blockskemat över den mycket populära monolitiska logaritmiska förstärkaren AD8307 från Analog Devices visas i figur 1. Detta är en relativt billig IC-krets och vi betalade lite drygt 100:- för en styck (13 euro), plus porto. Vi provade från början DIL-versionen av AD 8307. Även om den är enklare att löda och använda än den ytmonterade (SMD) versionen så gjorde de längre anslutningsbenen det omöjligt att använda kretsen vid frekvenser högre än 100 MHz. Vi utförde flera experiment med denna typ innan vi fann att SMD-typen kunde användas upp till 500 MHz. Återigen utfördes flera experiment, korten tillverkades, ingångsmotstånd och kondensatorer byttes ut för att optimera bygget. Vid frekvenser över 300 MHz skall ingångseffekten inte överskrida +20 dBm (100 mW) för att behålla noggrannheten. Detta är en dokumenterad svaghet hos AD8307. Dock inget stort problem om du är medveten om det, det gäller bara att använda rätt ingångsdämpning för att minska den egentliga HF-effekt som läggs till chipet till en nivå under 20 dBm och på så sätt garantera optimal noggrannhet.

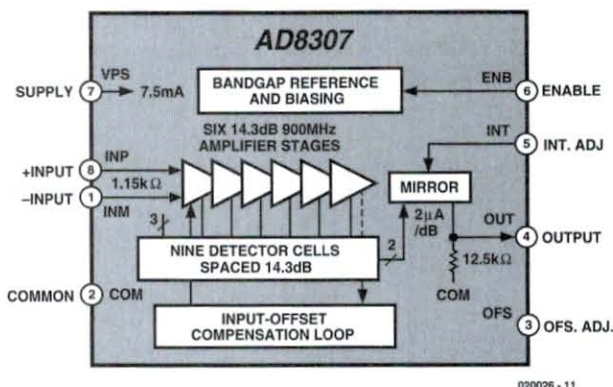
Författaren till denna artikel vill passa på att tacka alla de radioamatörer och tekniker som har bidragit med goda idéer, hjälpt till med mätningar och lånat ut dyrbar testutrustning och på så sätt gjort det möjligt att konstruera denna ingångskrets.

### Kretsbeskrivning

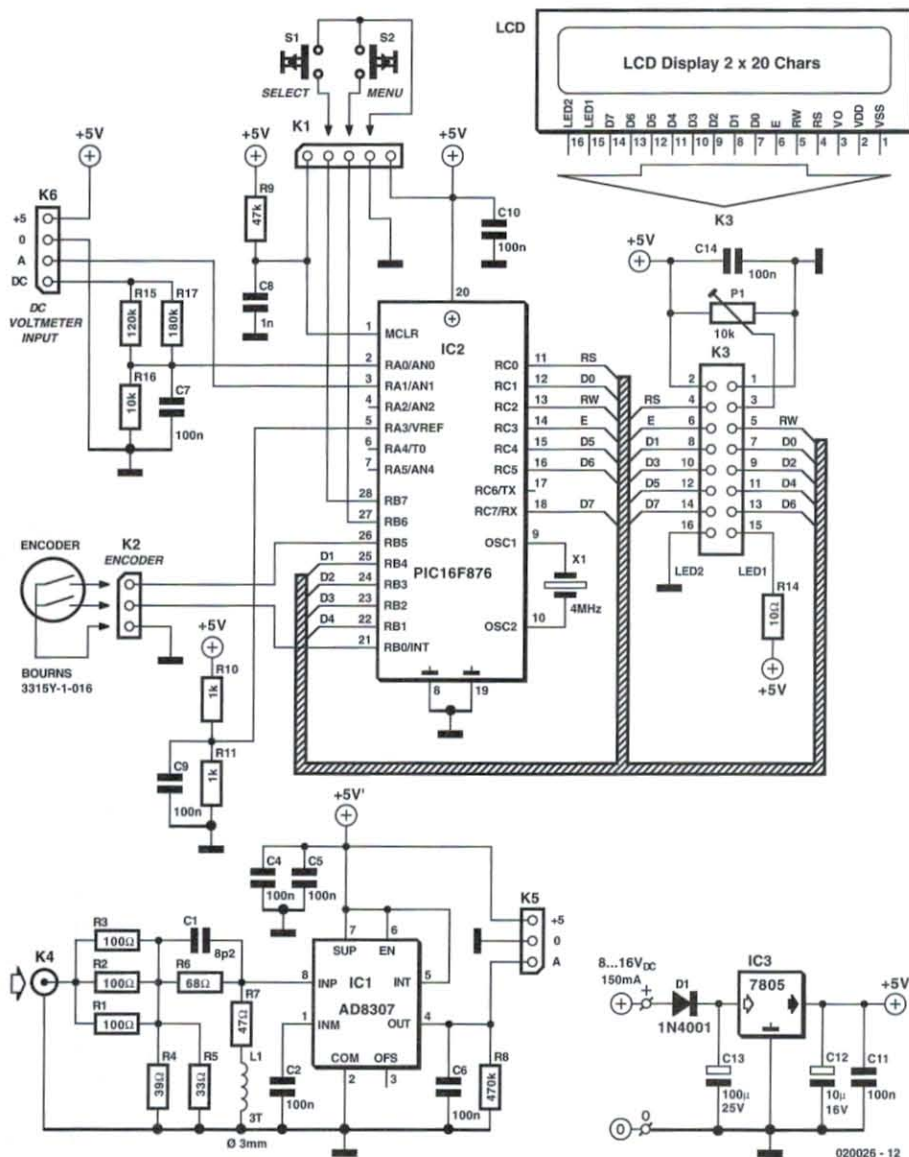
Kretsschemat över HF-wattmetern visas i figur 2. Den består av fyra sektioner, vilka kortfattat kommer att beskrivas nedan. På så sätt för vi lite mer utrymme att titta på den andra huvudkomponenten, PIC16F876.

HF-spänningsomvandlaren är ritad som en separat enhet runt AD8307. 'Log' ekvivalenten för den spänning som representerar den HF-effekt som läggs till kontakt K4 finns på K5 i form av en brant nivå mellan 0 och 2,5 V. Ingångens motståndsnät är dimensionerat för en 50  $\Omega$  ingångsimpedans, vilket är standard inom HF. Ingångsfiltret kan hantera effektnivåer på upp till 1 watt. C1 och L1 används för att ta bort strökapacitans och induktans och på så sätt optimera ståendevägförhållandet (SWR).

Den andra enheten är den digitala kontrollern runt IC2. Denna 'svarta låda' kör mjukvara



Figur 1. Blockskemat över AD8307.



**Figur 2. Kretsschemat över HF-wattmeter. De två huvudkomponenterna är AD8307 i ingångsomvandlaren och PIC16F876 i kontrollsektionen.**

som skrivits av konstruktören och som hanterar följande funktioner: omvandlar utgången från AD8307 till ett format som vi jordlingar kan läsa och förstå; läser användarkontrollerna (tryckknapparna S1 och S2 samt vridencodern på K2); driver en LCD så att den kan visa menyer, värden etc. PIC-en nollställs vid uppstart av R9-C8. Den klockas av en 4-MHz keramisk resonator. Den tredje enheten är LCD-n. Här använder vi en typ med två rader och tjugo tecken per rad. Trimpotentiometern P1 används för att justera kontrasten. Den fjärde enheten är nätdelen runt IC3. Detta är en helt konventionell design och torde inte behöva någon närmare diskussion. Ingångsspänningen kan tas från vilken nät-

adapter som helst som kan lämna en ström på runt 150 mA vid en spänning på 8 till 16 VDC.

## Om PIC16F876

Kraven på den digitala sektionen i detta projekt inkluderade en billig mikrokontroll med en 10-bitars analog-till-digital omvandlare (ADC), ett billigt och enkelt programmeringsinterface samt programminne av Flashtyp för enkel mjukvaruutveckling och avbuggning. Dessutom, fyra digitala ingångar, två för tryckknapparna, två för encodern, sju utgångar till LC-displayen i 4-bitarsmod, eller 11 utgångar i 8-

bitarsmod. Microchips 16F873 och 16F876 med 4 och 8 kByte Flash programminne visade sig vara ett alldeles utmärkt val. Eftersom de kostar ungefär lika mycket valde vi 8 kByte. 16F876 har fem analoga ingångar med 10-bitars upplösning, vilket motsvarar området 0-1023 i diskreta värden när ingångsspänningen går från 0 till 5 V. DC-signalen från AD8307 täcker 0 till 2,5 V för hela arbetsområdet. För att få full digital upplösning kan ADC-n inuti PIC-en använda en extern referens för fullskalespänningen. Denna funktion implementeras av R10 och R11 som skapar en 2,5-V referens från matningsspänningen på 5-V. Denna spänning är inte superkritisk, om den skulle vara lite fel så kan detta kalibreras i mjukvaran med en noll dBm offsetpunkt.

En bra och enkel programmerare för denna PIC har designats av Johann Aichinger och kallas PROPIC. Scheman och mjukvara kan laddas ner från

<http://jaichi.virtualave.net/>

Många andra PIC-programmerare stöder 16F876, bl.a. 'ICPROG' av Bonny Gijzen. Denna fungerar också bra och stöder i stort sett vilken IC som helst som kan programmeras. Du hittar den här:

[www.ic-prog.com](http://www.ic-prog.com)

Mjukvaran för wattmeters utvecklas med Microchips PIC-programmeringsats 'MPLAB' tillsammans med en extern C-kompilator som kallas PICC från Hi-Tech. Deras webbsida finns vid:

[www.htsoft.com](http://www.htsoft.com)

där du kan hämta en gratis demoverision. Om du behöver rekompilera mjukvaran för denna wattmeter så kan du ändra startskärmen så att den visar ditt namn eller din anropssignal eller liknande. Källkoden för wattmeters kan du hämta gratis från vår websida. Det relevanta filnumret är 020026-11 (Nummer 9/2002). För er som inte har tillgång till Internet så finns filerna att köpa på diskett från oss till självkostnadspris. Se Läsarservice.

## Själva bygget

Ett instrument som den här HF-wattmeters måste byggas med tanke på den mekaniska aspekten eftersom den inte kommer att

fungera korrekt om du slarvar med avskärmningen.

Till att börja med så skall det krets-kort som visas i figur 3 delas i två delar för att separera kontrollersektionen från ingångskortet.

### Ingångskort

Detta kort, figur 4, använder SMD (ytmonterade komponenter) eftersom små komponenter kan hantera högre frekvenser på ett bättre sätt. Läs följande avsnitt om det är första gången som du bygger ett kort med SMD.

Om du löder med din högra hand så lägg först en liten mängd tenn på det högra lödplåten.

Placera komponenten noga med din vänstra hand med hjälp av en pin-cett och löd fast den högra sidan medan du håller komponenten helt rakt. Vrid nu på kortet och löd fast det vänstra komponentbenet på den vänstra lödplåten.

För att löda fast AD8307 på kortet så förtenner du snabbt och noga löd-

plätt för ben 3 på kortet. Håll sedan IC-kretsen nedtryckt på plats och varm ben 3 samtidigt som du trycker ner det mot lödplåten med spetsen på lödpennan. Se till att IC:n sitter korrekt och löd fast de övriga benen.

På ingångskortet finns det gott om plats för SMD-komponenter av både 1206 och 0805 typer.

Spolen L1 består av 3 varv 0,5 mm lackerad koppartråd. Den inre diametern hos denna spole är 3 mm och varven skall ligga ca 0,5 mm isär.

Ingångskortet skall monteras direkt på kragen hos en BNC-kontakt med fyra gängade M2,5 hål. Vissa BNC-kontakter har en 3-4 mm lång teflonisolering runt mittstiftet. Ta bort denna isolering och kapa mittstiftet så att det blir ca 2 mm långt.

Kolla nu att det går att löda på kragens baksida. Om det inte gör det så måste du skrapa bort det skyddslager som eventuellt finns så att det går att löda. Montera nu ingångskortet, spara inte på lödtennet, och löd

fast det hela vägen.

### Kontrollerkortet

En tidig prototyp av detta kort visas i figur 5. Om PIC16F876 monteras i en IC-hållare så

## KOMPONENTLISTA

### Motstånd:

SMD-kapsel 1206 eller 0805:

R1, R2, R3 = 100Ω (R3 ovanpå R1/R2)

R4 = 39Ω

R5 = 33Ω

R6 = 68Ω

R7 = 47Ω

R8 = 470kΩ

R9 = 47kΩ

R10, R11 = 1kΩ

R15 = 120kΩ

R16 = 10kΩ

R17 = 180kΩ

R14 = 10Ω 1W

P1 = 10kΩ trimpot

### Kondensatorer

SMD-kapsel 1206 eller 0805:

C1 = 8pF2

C2-C7, C9, C10, C11, C14 = 100nF

C8 = 1nF

C12 = 10μF 16V radial

C13 = 100μF 25V radial

### Spole

L1 = 3 varv, 0,5mm diam., varven isär 0,5mm, intern diam. 3mm.

### Halvledare:

D1 = 1N4001

IC1 = AD8307AR (SMD)

IC2 = PIC16F876-04/SP, programmerad, ordernr 020026-41, se Läsaerservice

IC3 = 7805

### Övrigt:

K1 = 5-pol SIL-stiftlist

K2, K5 = 3-pol SIL-stiftlist

K3 = 16-pol SIL-stiftlist

K4 = BNC-kontakt med krage

K6 = 4-pol SIL-stiftlist

S1, S2 = tryckknapp, 1 slutande kontakt, för panelmontering

PC1, PC3, PC8, PC10, PC12 = lödstift

X1 = 4MHz keramisk resonator (3 bens)

LCD-modul med 2 rader om 20 tecken, t.ex., LM032L (PC2002LRS-BEA-C)

Vridencoder typ 3315Y-I-016 (Bourns)

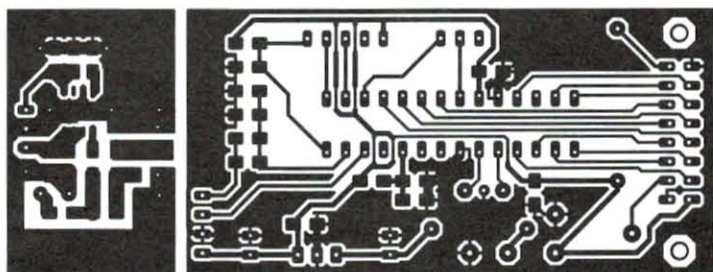
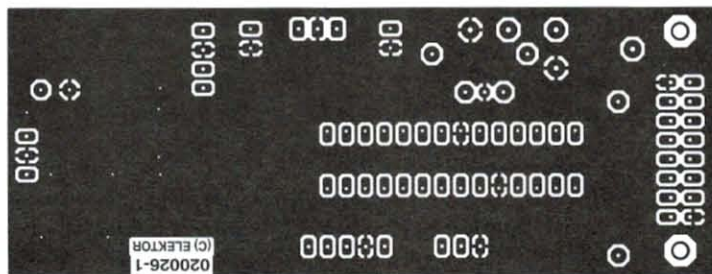
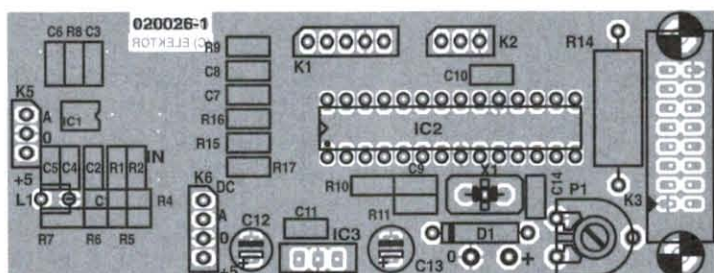
Kontakt för nätdapter, panelmonterad

IC-hållare, 28-bens, smal

Kretskort 020026-1 (se Läsaerservice)

Diskett med källkodsfiler 020026-11

(filerna finns också att hämta gratis från vår hemsida)



Figur 3. Kretskortslayouten för instrumentet. Kortet är dubbelsidigt, genompläterat och finns att köpa från oss, se Läsaerservice.

skall du använda en av hög kvalitet, dvs en hållare med svarvade ben. De av er som har tillgång till lämpliga programmeringsverktyg för PIC kan löda fast kretsen direkt på kontrollerkortet eftersom chipet kan uppdateras 'in-circuit' med ny mjukvara när sådan finns tillgänglig.

Den keramiska resonatorfrekvensen är 4 MHz och detta är inte kritiskt. Om du inte vill använda den alternativa seriella utgång som diskuteras i menyutan kan du använda en 3-benstyp med inbyggda kondensatorer. En normal 2-bensversion går också bra.

Kontrollerkortet monteras på undersidan av LCD-modulen med två distanser på 10 mm eller vad du nu har för hand. Du behöver fyra distanser för att montera LC-displayen på lådans frontpanel.

Motståndet R14 är ett 1-wattstyp som styr nivån på LCD-ns bakgrundsbelysning. Ett värde på 10 ohm ger normal belysning, men om du vill strömförsörja instrumentet med batterier så är det bättre att använda ett på 20 ohm för R14. Du kan också montera in en omkopplare så att du kan koppla bort belysningen när du kör på batterier.

Spänningsregulatorn 7805 kommer att bli ganska varm om ingångsspänningen överstiger 10 V. Du kan antingen böja ner regulatorn mot lådan och fästa den med en M3-skruv eller montera fast ett litet kylelement. Det räcker med den typen som du klämmer fast.

## Vridencoder och switchar

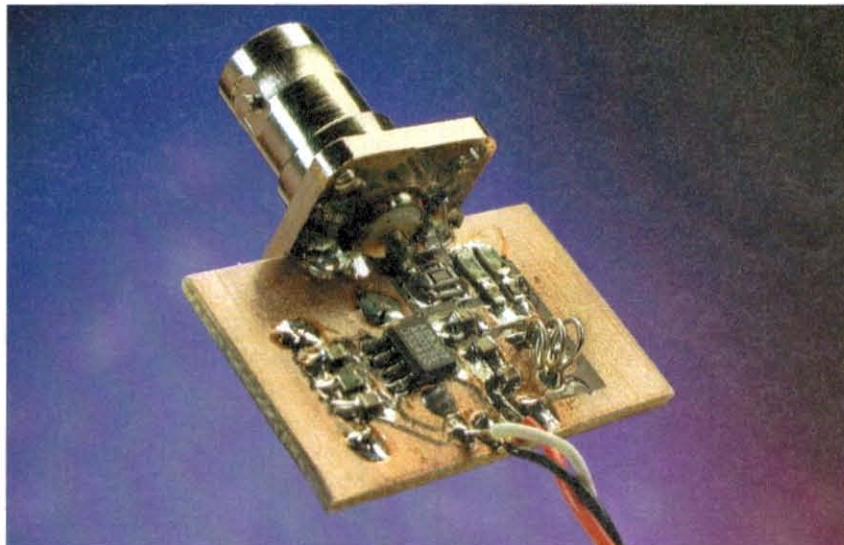
Vridencodern är en billig dubbelkontakt som lämnar Graykod. Denna encoder används för att välja olika alternativ från menyn och för att ändra olika inställningar, se avsnittet om mjukvaran längre fram. Om encodern går åt fel håll så växla bara trådarna vid de två anslutningarna vid datautgången.

De två switcharna S1 och S2 måste vara tryckknappar. S1 används för att plocka alternativ från menyerna och S2 används för att komma åt menyerna.

## Initialtest (hårdvara)

Vid denna punkt förutsätter vi att båda korten är färdigbyggda. Om PIC-kretsen sitter i en hållare så ta bort den och om den är fastlöd direkt på kortet så skall en variabel DC-nät-del användas för det test som beskrivs nedan.

Börja med bara kontrollerkortet, med 7805 monterad och eventuellt också PIC. Displayen och ingångskortet skall inte anslutas ännu. Innan du fortsätter måste du kolla att 7805-an fungerar korrekt. Mata kortet med 9 V, kolla att det finns +5 V vid lödplätten för ingångskortet intill C12. Kolla att det finns



Figur 4. Ingångskortet med BNC-kragen fastlöd utefter kanten.

+2,50 V på IC1, ben 5.

Kolla att det finns +5 V på IC1 vid ben 5. Koppla bort spänningen, anslut displayen, sätt in PIC-en i hållaren och sätt på spänningen igen.

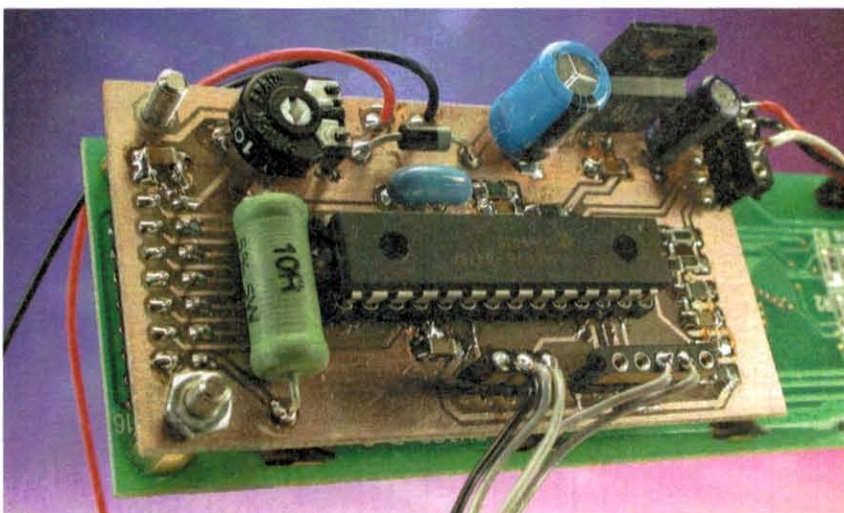
Vrid P1 moturs så att displayen får noll volt på justeringsbenet 3. Detta ger maximal kontrast så att du kan se att LCD-n lever. Justera nu P1 för bästa kontrast ur din synvinkel. Kolla att det finns grön bakgrundsbelysning och du kommer att se displayen skriva HF-wattmeterns välkomstmeddelande, se ruta. Rör vid ingång A med ett finger. Detta skall ge utslag på displayen. Stäng av strömmen.

Dubbelkolla all lödning på AD8307 och ingångskortet och anslut detta till huvudkortet med skärmad audio-

kabel eller liknande för att koppla signalen från ingångskortet till kontrollerkortet. Koppla på strömmen igen och kolla att strömförbrukningen ligger under 150 mA.

Om du inte har någon HF-testgenerator att kalibrera och testa med så sätt ett vanligt motståndsbän in i BNC-ingångskontakten och sänd med din VHF, UHF eller vad för radio (PMR) du nu har tillgänglig. Ju närmare du för radion mot wattmetern, medan du sänder naturligtvis, desto högre kommer avläsningen på instrumentet att bli.

När detta initialtest har utförts OK skall du använda en bra HF-signalgenerator (låna en om det behövs). Med denna kan du kalibrerar din nya digitala wattmeter och använda den



Figur 5. Kontrollerkortet sätts fast på LCD-modulens baksida.

## Att använda menyerna (mjukvara v. 1.03)

All intelligens för instrumentet befinner sig i den mjukvara som utvecklats av konstruktören och som är flashad in i PIC-mikrokontrollern. De av er som har tillgång till lämplig mjukvara och utrustning för PIC-programmering kan hämta källkodsfilerna gratis från vår websida ([www.alltomelektronik.com](http://www.alltomelektronik.com), Download, 9/2002, fil 020026-11). Färdigprogrammerad PIC finns också att köpa från oss, ordernummer 020026-41.

Instrumentets välkomstskärm skall se ut så här:



Startskärmen visar: dBm, status, HF-spänningsstapel, HF-effekt watt



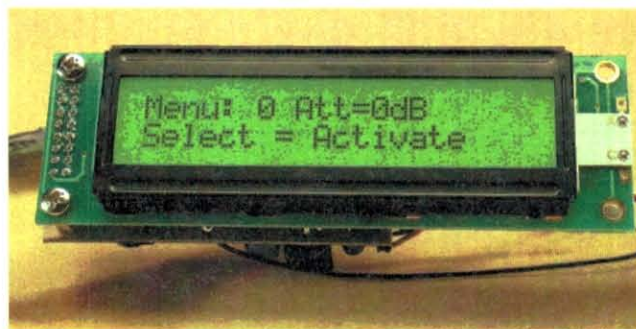
Om ingen dämpning används går dBm-visningen från -63 dBm (brusgolvet) till +30 dBm (1 W). Statusvisningen visar det valda frekvensbandet och dämpningsmod. Använd bandväljaren för att växla mellan LF, HF, VHF, UHF och SHF kalibreringsminnen. Kalibrera förslagsvis wattmeterns 0 dBm-referens vid: LF = 3.5MHz, HF = 14 MHz, VHF = 145 MHz, UHF = 430 MHz, SHF = 440 MHz. Du kan självklart kalibrera dina egna favoritfrekvenser för bäst prestanda. I effektmeters meny använder du SELECT-knappen för att komma in i RELATIVE mod. I detta mod visas dBm och stapel, när du



går in i detta mod nollas dB-visningen

### Menyn

För att gå in i menyn/inställningar använder du MENU-knappen. När du är i en meny använder du vridencodern för att få önskad inställning. Vid rätt inställningen använder du SELECT för att aktivera, detta visas också på displayen.



### Tillgängliga menyposter:

- 0: 0 dB, ingen dämpare monterad, 1 W max.
- 1: -10 dB dämpare monterad, 10 W max.
- 2: -20 dB dämpare monterad, 100 W max.
- 3: -30 dB dämpare monterad, 1 kW max.
- 4: -40 dB dämpare monterad, 10 kW max.
- 5: -50 dB dämpare monterad, 100 kW max.
- 6: DC-voltmeter, verklig, min. och max.
- 7: HF-effektmeter, startskärmen.
- 8: SSB PEP (peak envelope power) wattmeter, med peak hold och variabel avklingning.
- 9: Returförlust med SWR-visning, användbar med SWR-brygga.
- 10: Kalibrera 0 dBm i det valda bandet.
- 11: Läs alla kalibreringsvärden.
- 12: Nolla alla kalibreringsminnen.
- 13: Displayens uppdateringsfördröjning 2-80 ms, peak hold och avklingningshastighet.
- 14: Om Info, visar mjukvaruversion osv..

### DC-voltmeter

Ingenting brinner om ingångspolariteten vänds. På DC-voltmeter-skärmen visas verklig spänning, minimum och maximum. För att nollställa min och maxvisningen trycker du på SELECT. Voltmetern kan användas för att övervaka batterispänningen om du använder batterimatning, eller vad du nu vill mäta. Kom dock ihåg att ingångsimpedansen är ungefär 80 kΩ.

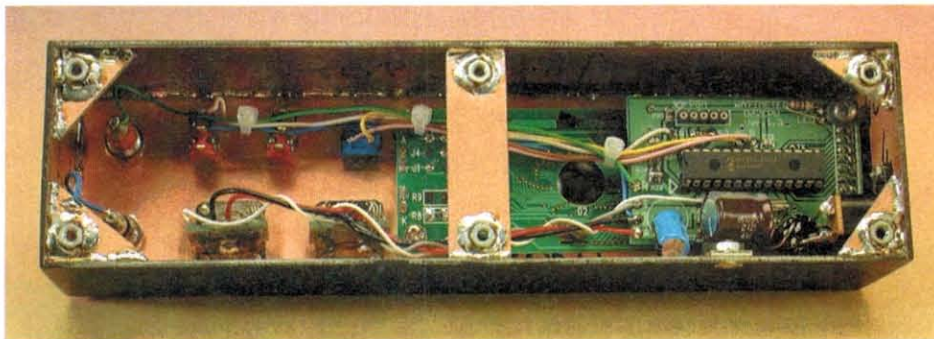
### Extra egenskaper

Några av er kanske tycker det skulle vara bra med en seriell utgång. PIC-en skickar ut en seriell dataström på ben 17 som kan omvandlas till RS232-nivåer med en MAX232-IC i sin normala konfiguration. Den dataström som produceras av PIC-en kan anslutas till en ledig serieport på din PC. Du kan läsa denna med vilket terminalprogram som helst (HyperTerminal t.ex.). Kommunikationsinställningarna skall vara: 38400 baud, 8 bitars ASCII, ingen paritet, 1 stoppbit, eller kort: 38K4 8 N 1.

för noggranna mätningar i din egen lya. Men först skall du bygga in elektroniken i en korrekt avskärmd låda. Kalibreringen skall utföras senare.

## Låda? Hemgjord eller..

Självklart vill du att lådan skall passa exakt för LC-displayen, BNC-kontakterna, switcharna, DC-kontakterna osv. Så varför inte tillverka en egen låda av enkelsidigt, kopparklätt, oetsat kretskortsmaterial? Detta är billigt, lätt att säga och borra i. Lägg ner lite extra tid på att göra öppningen för displayen snygg och prydlig. Det är lätt att löda ihop de olika sidostyckena med varandra och när du är färdig kan du spreja lådan svart och sedan använda vita gnuggbokstäver för markeringarna. Vi lade på ett lager klarlack till slut för att skydda texten. En sådan låda kostar inte mycket mer än tiden och arbetet. Vår prototyplåda visas i figur 6.



**Figur 6.** Förslagsvis kan en låda tillverkas av bitar av enkelsidiga kopparklädda kort som löds ihop utefter kanterna. Denna metod är populär bland radioamatörer.

## Kalibrering av HF-ingång och DC-ingång

HF-metern kalibreras med mjukvara genom att lägga 0 dBm till ingångsterminalen. Välj först motsvarande bandminne med hjälp av encodern, tryck sedan Menu, välj Calibrate 0 dBm och tryck på selectknappen.

Signalstyrkan 0 dBm lagras i wattmeters interna minne. Från och med nu kommer den att mäta inom  $\pm 0,2$  dB i det frekvensbandet. Detta upprepas nu för alla andra band som du behöver. Det finns fem bandminnen som kallas LF, HF, VHF, UHF och SHF. Kom ihåg att du kan kalibrera vid vilken frekvens du önskar, när som helst, om och om igen. DC-voltmetern kalibreras med hårdvara. Du behöver finavstämningsspänningsdelarens

toppmotstånd R15 parallellt med R17 och bottenmotståndet R16. Du behöver en noggrann digitalvoltmeter för DC och en variabel DC-nät-del.

Ändra wattmeters displayskärm till DC Voltmetermod med hjälp av Menu -> Dial -> Select.

Lägg 20.00 volt på DC-ingången och kolla på LCD-visningen. Om displayen visar ett värde som är lägre än 20.00 skall du montera ett 10 megohms motstånd parallellt med toppmotståndet R17. Om den visar mer än 20.00 montera det då till bottenmotståndet R16. Kom ihåg att koppla bort 20-V källan och strömförsörjningen till wattmetern när du löder inuti instrumentet. Om du kortsluter R15 eller R17 och det samtidigt finns 20 V på terminalen DC-in, kommer du att behöva en ny mikrokontroller!

## Varningar

HF-spänningar på några volt, speciellt från UHF-sändare, kan orsaka brännskador eller andra hudskador. **Vidror aldrig en terminal som bär uteffekt från en sändare.**

AD8307 kan inte överleva +5 V som av misstag läggs till dess ingångsben, så dubbelkolla dina ledningar från huvudkortet.

Kom ihåg det faktum att ingångskortet är designat för ineffektnivåer som inte får överstiga 1 watt.

Om du lägger på mer effekt kommer avläsningen på instrumentet faktiskt att gå ner! Om du är slarvig och lägger mer än 1 watt till instrumentet

gör du förmodligen slut på AD8307-an. Lita aldrig på att ingångsmotstånden kommer att brinna sönder först eftersom dessa kan hantera betydligt mer än de är specificerade för (även under flera minuter och då har din AD8307 gett upp andan för länge sedan). **Det finns alltså inget överbelastningsskydd.**

Om du vill testa en sändare så använd en dämpare, en riktningskopplare eller en avtappning. Konstruktören till denna wattmeter använder en hembyggd 30 dB dämpare som klarar att hantera 50 watt kontinuerlig HF-effekt. Dess frekvensåtergivning är rak från DC till 700 MHz. Ingen av hans VHF eller UHF-sändaren kan lämna mer än 50 watt så detta är inget problem. En HF-rigg kräver dock en 50 dB 'tap' typ dämpare som också är hembyggd. Denna tillåter mätningar upp till en kilowatt utan problem. Fler detaljer kan du hitta på konstruktörens hemsida.

## Sammanfattning

Konstruktören har använt denna wattmeter under mer än ett år för olika ändamål som involverat radiosignaler.

Den marginellt högre noggrannheten, större frekvensområdet och dynamiska området hos professionella HF-wattmetrar bär ett pris som är 20-40 gånger högre än kostnaden för det instrument som beskrivs här.

Som en framtida expansion kan två ingångskort monteras med en mjukvarurutin som subtraherar ingång B från ingång A för att visa fram- och reflekterad effekt tillsammans med beräknat SWR-värde. Just nu är detta dock bara en idé men det är skälet till att det finns två BNC-kontakter som kallas för ingång A och ingång B på konstruktörens HF-wattmeter.

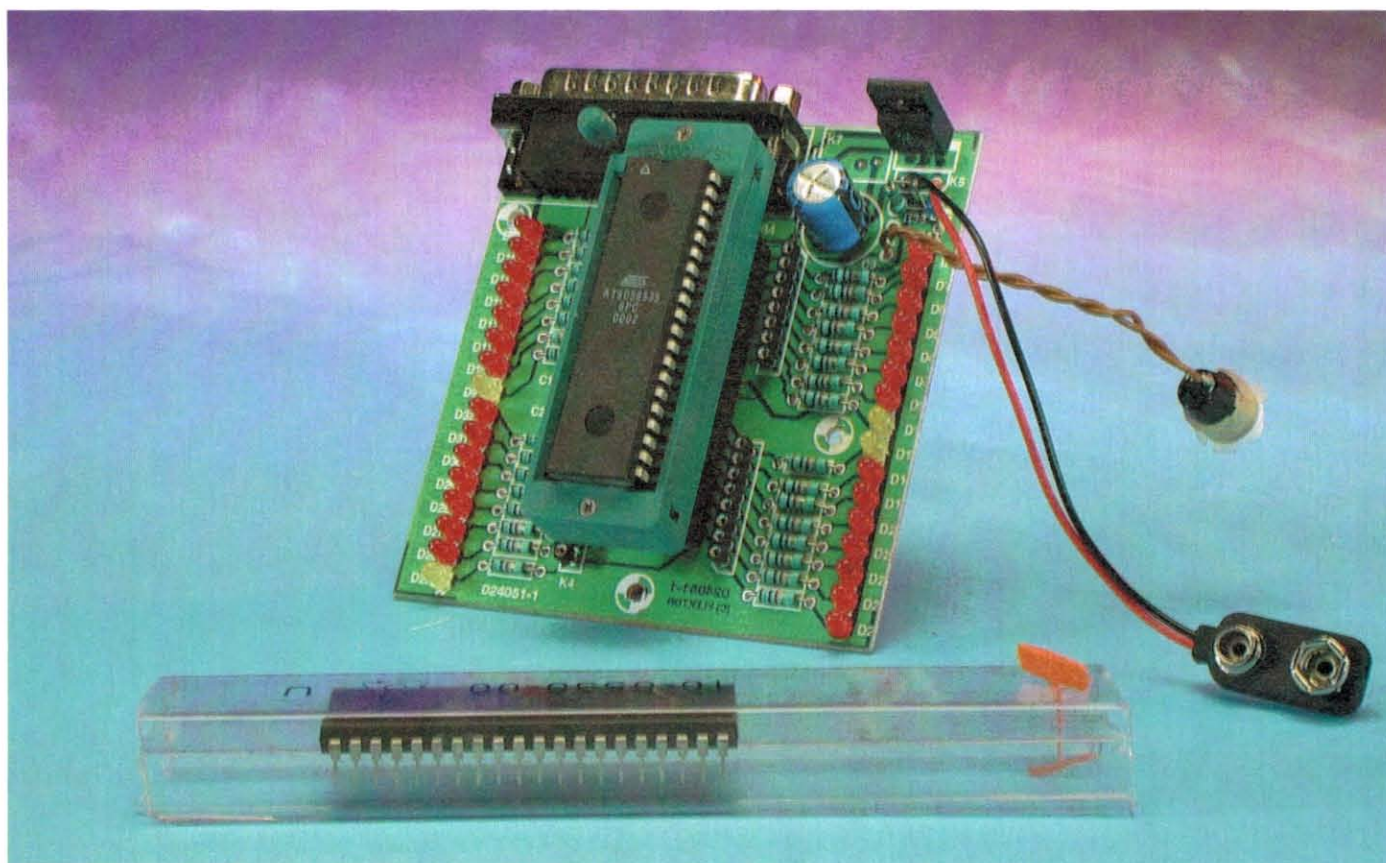
(020026-1)

# AT90S8535 programmerare

## lämplig för användning in-system

Design by M. van Houwelingen

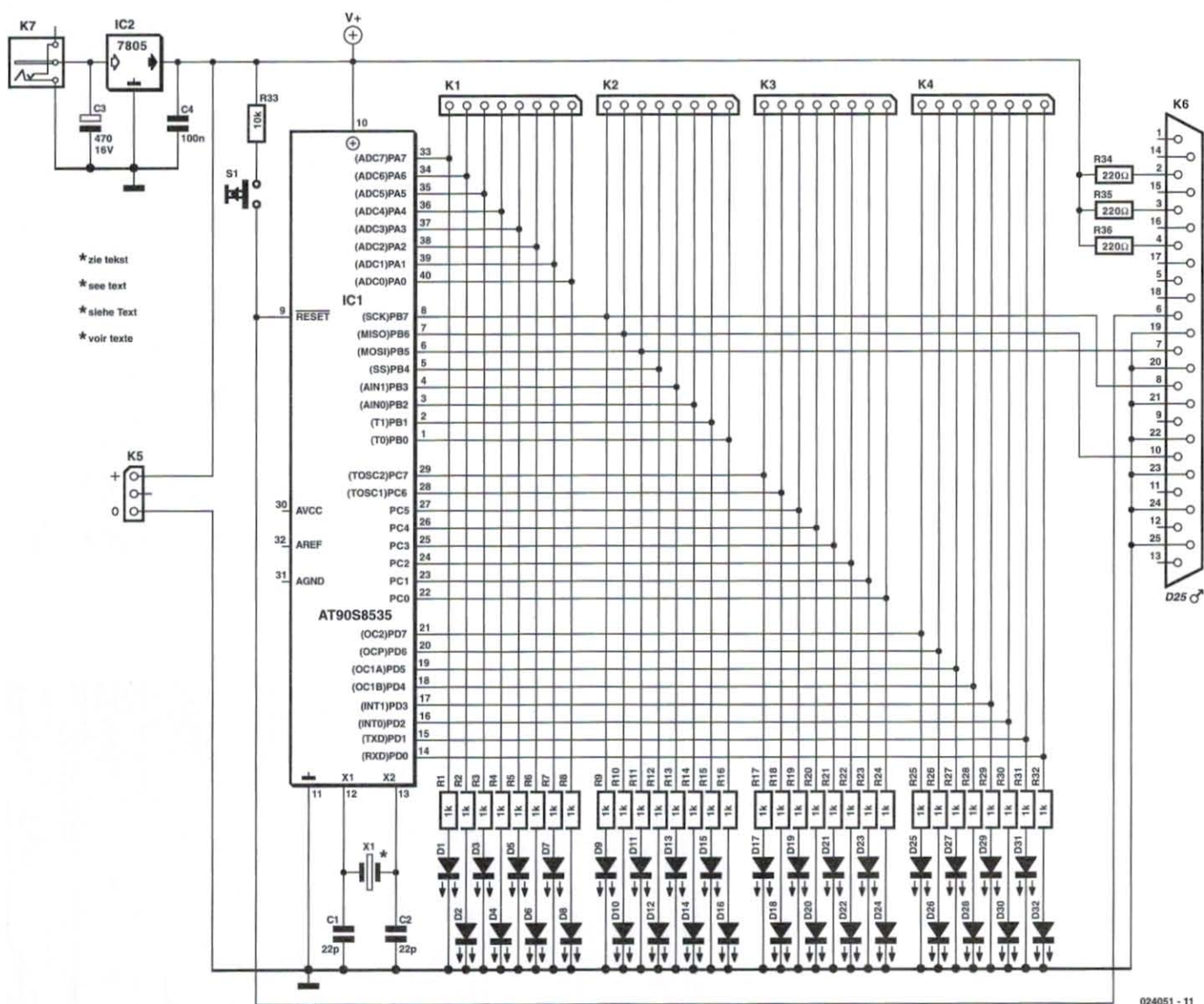
Denna programmerare för Atmels processorer är lätt att bygga och mycket billig. Inte av mycket mer än en ZIF-hållare, fyra kontakter och 32 LED. Laddarprogrammet kan hämtas gratis.



De av er som använder många olika typer av mikrokontroller har stött på det oundvikliga problemet att de alla behöver olika programmerare. Om man måste köpa dessa varje gång blir det snart ganska dyrt. Dessutom

har de flesta programmerare den nackdelen att mikrokontrollern först måste tas ut ur kretsen innan den kan programmeras. Genom att bygga programmeraren själv blir allt både enklare och billi-

gare. Den programmerare som vi beskriver här kan, tillsammans med den tillhörande (gratis) mjukvaran och en datakabel, programmera chips ur AT90S/LSxxxx serien från Atmel. Detta betyder att du inte längre behöver ta bort mikrokontrollern från kretsen för



**Figur 1. Kretsschemat över den kombinerade testare/programmeraren. Den externa matningsspänningen (via K7) behövs bara under testning.**

att programmera den. Dessutom är den mycket billig eftersom det lägsta kravet på hårdvara är den kabel som behövs för att överföra programmet från datorn till mikrokontrollern. Huvudfunktionen hos den krets som vi beskriver här är att göra testningen av program enklare. En blick på kretsschemat i figur 1 bekräftar att detta är en mycket enkel affär.

## Mjukvara

Den mjukvara som skickar hexfiler till mikrokontrollern baseras på det faktum att dessa Atmelchips kan programmeras seriellt. Detta sker via de MISO, MOSI och SCK-ben som hittas på alla chip ur AT90S/Lsxxxx serien. Mjukvaran använder också RESET-benet, GROUND och en matning via parallellporten.

För ytterligare information angående programmeringstekniken hänvisar vi till Atmels datablad.

Eftersom det bara är de ovan nämnda benen som behövs vid programmering så är det enkelt att förse den krets som innehåller en mikrokontroller med en extra kontakt för datakabeln. Chipet kan sedan enkelt programmeras 'in-system'.

En sak som man skall komma ihåg är att parallellporten endast kan lämna en begränsad mängd ström. Av detta skäl rekommenderar vi att mikroprocessorn matas från den inbyggda strömförsörjningen och inte via kabeln.

## Kretsen

Som man kan se av figur 1 består kretsen endast av en 40-polig ZIF-hållare (nollkraftshållare), fyra kontakter för anslutning till externa kretsar samt 32 lysdioder med seriemotstånd. Lysdioderna har lagts till så att statusen hos utgångarna kan inspekteras när du kör dina egna program. Det finns också en resetknapp såväl som en regulator på 5 V som tar sin spänning från en nätadapter (minimum 8 V). Matningen behövs bara när du kör ett program och inte under den egentliga programmeringen.

Frekvensen hos kristallen X1 skall matchas till mikrokontrollerns has-

tighet eftersom det finns 4 MHz och 8 MHz versioner. Det säkraste alternativet är att satsa på en 4 MHz kristall eftersom denna fungerar med båda typerna av mikrokontroller.

[www.student.tue.nl/E/m.j.v.houwelingen](http://www.student.tue.nl/E/m.j.v.houwelingen).

Programmet heter FBPRG16.EXE. Denna webbplats innehåller också annan mjukvara som kan visa sig användbar vid utvecklingen av dina egna program.

## Kretskort

Kretskortet för testaren/programmeraren visas i figur 2. Som du ser är det en mycket enkel krets som är mycket lätt att bygga. Se till att lysdioderna är av lågströmstypen eftersom värdet på seriemotstånden har beräknats efter dessa.

Om du vill använda kretsen för att programmera en mikrokontroller med färre ben måste du göra en adapter som passar in i den 40-poliga hållaren. På detta sätt blir det möjligt att programmera hela området i AT90S/LSxxxx serien. När du bygger adaptern skall du naturligtvis ansluta alla I/O-ben så att också de mindre chipen kan testas i denna krets

## Och slutligen

Laddarprogrammet är en gratis DOS version. Det kan hittas här:

## KOMPONENTLISTA

### Motstånd:

R1-R32 = 1k $\Omega$   
R33 = 10k $\Omega$   
R34,R35,R36 = 220 $\Omega$

### Kondensatorer:

C1,C2 = 22pF  
C3 = 470 $\mu$ F 16V radial  
C4 = 100nF

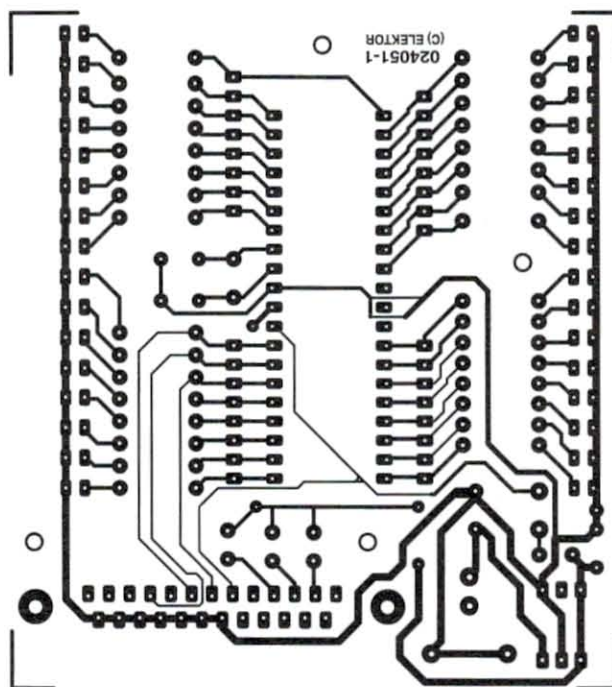
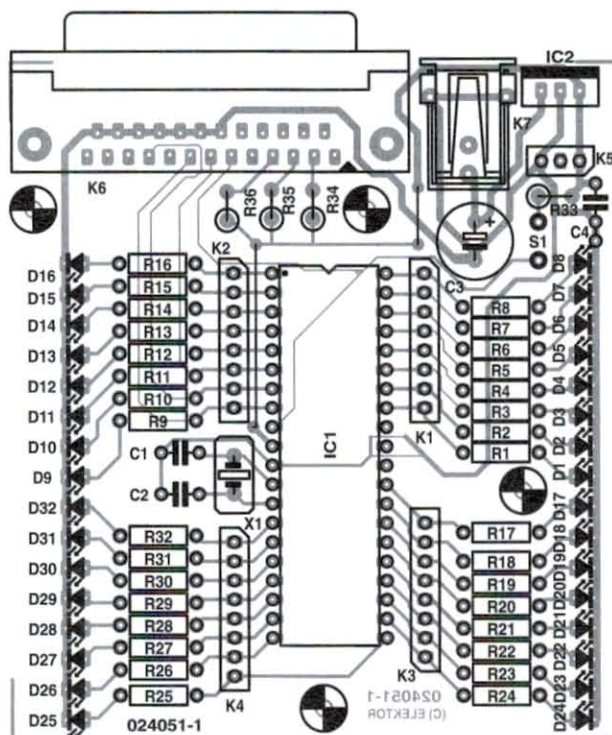
### Halvledare:

D1-D7,D9-D15,D17-D23,D25-D31 =  
LED, 3mm, röd (lågströms)  
D8,D16,D24,D32 = LED, 3mm, gul  
(lågströms)  
IC2 = 7805

### Övrigt:

IC1 = 40-pol ZIF-hållare  
K1-K4 = 8-pol header  
K5 = 3-pol header  
K6 = 25-pol sub-D plugg  
(hane),vinklade stift, för kretskort  
K7 = kontakt för nätdapter  
S1 = tryckknapp  
X1 = 4MHz eller 8MHz kristall (se text)  
Kretskort **024051-I** (se Läsar-service)

(024051-1)



Figur 2. Att bestycka kretskortet borde inte ta lång tid.