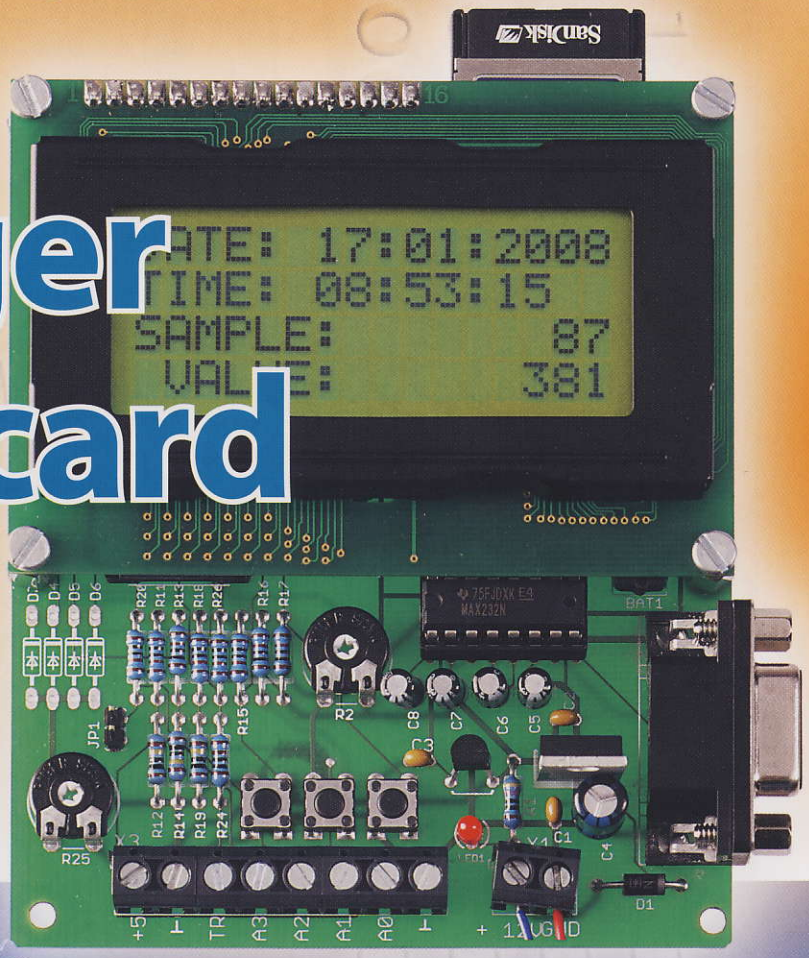
www.alltomelektronik.se 3/2008 SEK 65:00 NOK 69:00 EURO 9

elektor

electronics worldwide



Datalogger med SD-card

Har minne, kan logga



Mjukvara bygger hårdvara

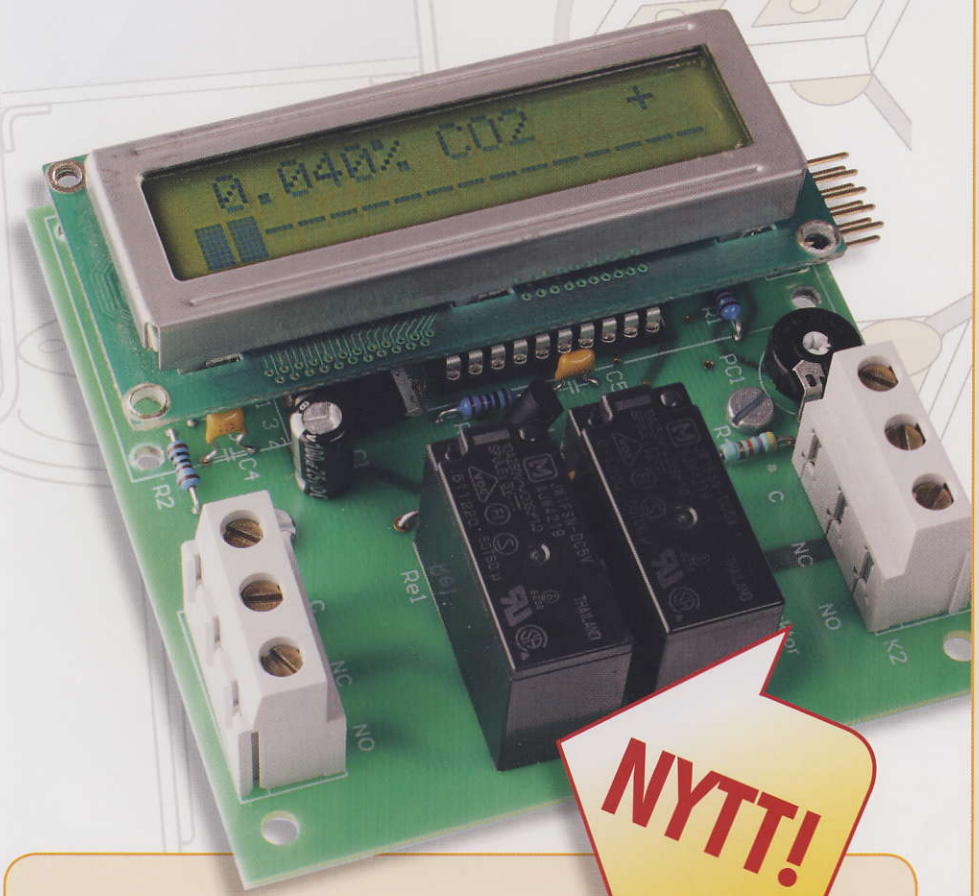
C-till-H, kompilatorer & FPGA

- LED-projektor; Tala som en robot
- Realtids operativsystem, RTOS
- Hemligheterna med I²C-bussen

INTERPRESS 0173-03
03
7 388017 306500
RETURVECKA 19

Vår styrka

En värld av elektronik från en enda shop!



NYTT!

Upptäck dålig luft innan du dåsar till

CO_2 mätning

(Januari 2008)

Koldioxid (CO_2) är inte bara ett hot mot miljön, det är också en viktig och ofta bortglömd faktor när man avgör luftkvalitén på jobbet eller hemma. För hög koncentration av CO_2 leder till trötthet, koncentrationsproblem och huvudvärk. Den CO_2-meter som vi beskriver här gör det enkelt att avgöra koncentrationen av koldioxid i luften. En mikrokontroller övervakar det uppmätta värdet och kan trigga ett larm eller starta en ventilationsfläkt om värdet överskrider en förinställd tröskel.

Komponensats, kretskort, sensorkretskort, ATtiny26 och display

Art.# 070802-71 • SEK 1625:00



Reflow lödkontroll

(6/2007)

I Elektors labb har behovet att löda SMD (ytmonterade komponenter) ökat stadigt under de senaste åren och detta gäller säkert också många av våra läsare. I nummer 3/2006 beskrev vi i detalj hur du kunde bygga din egen reflowugn med hjälp av en vanlig billig elektrisk ugn. Artikeln resulterade i många entusiastiska kommentarer från våra läsare, vilket bekräftade att det fanns ett stort intresse för projektet. I denna artikel presenterar vi en ny version av styrelektroniken för en gör-det-självugn.

Bestyckat kretskort med låda

Art. # 060234-91 • SEK 2350:00



USB Flash Board

(November 2007)

Flash mikrokontroller är lätta att programmera. Förr laddades programkoden normalt ned via ett serieinterface, men idag har många PC (speciellt bärbara) bara USB-portar. Detta universella Flashkort är en lösning på detta problem. Det är byggt runt en AT89C5131A som är en utbyggd mikrokontroller från 8051-familjen med en 80C52-kärna och en Full Speed USB-port. Som en bonus har IC-n ett komplett uppdateringsinterface för nedladdning av ny firmware. Atmel tillhandahåller också lämplig mjukvara i form av dess FLIP-program som finns tillgängligt gratis.

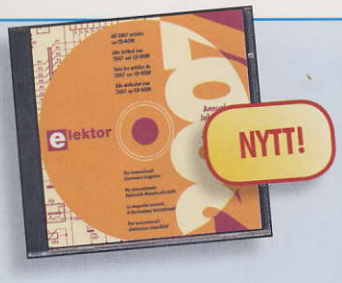
Byggsats inklusive kretskort och med alla komponenter

Art. # 070125-71 • SEK 525:00

Satser & moduler

Priser och beskrivningar kan komma att ändras.

CD-ROM



Alla artiklar publicerade 2007 i Elektor

Elektor 2007

Elektors årsvolym är bland de mest populära varorna i Elektors produktområde. Denna CD-ROM innehåller alla redaktionella artiklar som publicerats i Elektor under 2007. Med hjälp av det medföljande Acrobat Readerprogrammet presenteras artiklarna i samma layoutformat som i tidningen. Det finns en utbredd sökfunktion för att hitta nyckelord i en artikel. Engelska, tyska, franska och holländska.

978-90-5381-218-1 • SEK 275:00



Mer än 68,000 komponenter

Elektor's Components Database 4

Detta programpaket består av åtta databanker som täcker IC-kretsar, germanium och kiseltransistorer, FET-ar, dioder, tyristorer, triacar och optokopplare. ytterligare elva applikationer täcker beräkningarna av, till exempel, seriemotstådn för LED-ar, seriemotstånd för zenerdioder, spänningsregulatorer och AMV. Det finns en färgbandsavkodare för bestämning av motstånds- och spolvärden. ECD 4 ger omedelbar tillgång till fler än 68 000 komponenter. Alla datanaksapplikationer är fullt aktiva, vilket gör att du som användare kan lägga till, redigera och kompletta komponentdata. Denna CD-ROM är ett måste för alla som sysslar med elektronik.

ISBN 978-90-5381-159-7 • SEK 210:00



Silent alarm, poetry box, night buzzer and more!

PIC Microcontrollers

Denna byggbok täcker en hel serie med spännande och roliga projekt med en PIC mikrokontroller. Du kan bygga fler än 50 projekt för eget bruk. De tydliga förklaringarna, schemorna och bilderna på varje projekt på ett kopplingsdäck gör detta till en rolig aktivitet. Du kan också använda boken som en studieguide. Den tekniska bakgrundsinformationen för varje projekt förklarar varför projektet är uppkopplat som det är, inklusive användningen av datablad. Även när du byggt alla projekten kommer den fortfarande att vara en värdefull referensguide att ha intill din PC. All mjukvara som används i denna bok kan laddas ner helt gratis, inklusive all källkod, en programeditor och det öppna programmeringsspråket JAL.

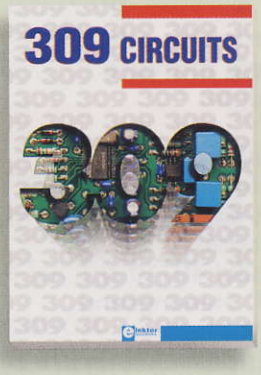
446 sidor • ISBN 978-0-905705-70-5 SEK 425:00

Mer information hittar du på vår wesida:

www.alltomelektronik.se

Allt om Elektronik
Box 178
444 22 Stenungsund
Tel: 0303-770 490
Fax: 0303-770 490
Email: order@alltomelektronik.se

elektor SHOP



Fullt genomarbetade elektronikprojekt

309 Circuits

I den här tionde utgåvan av den populära '30x Circuits' serien innehåller även den en omfattande varitet av kretsar, tips, tricks och designidéer för elektronik. Bland många övriga inspirerande ämnen är följande kategorier väl presenterade i denna bok: test & mätning; HF (radio); datorer och kringutrustning; audio & video; hobby & modeller; mikrokontroller; hem & trädgård; nätdelar och batteriladdare etcetera.

432 sidor • ISBN 978-0-905705-69-9 SEK 325:00



5.0, 6.0, VBA, .NET, 2005

Visual Basic for Electronics Engineering Applications

Denna bok vänder sig till er som vill styra befintlig eller självbyggd hårdvara från er dator. När du har bekantat dig med Visual Basic diskureras dess utvecklingsmiljö och verkstygsuppsättning i detalj. Varje ämne åtföljs av tydlig kod som är klar att köras och, där det är nödvändigt, schemor som får dig att komma igång med dina projekt på nolltid.

476 sidor • ISBN 978-0-905705-68-2
SEK 415:00

Böcker

elektor
electronics worldwide

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.se
http://
www.alltomelektronik.se

Prenumerationer 2007
PressData AB
Tel: 08-799 63 25
Fax: 08-98 45 72
Helår 545:-
Norge helår 545:- 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.se

Tryck
db grafiska ab
Örebro 2008

ISSN 0280-8765

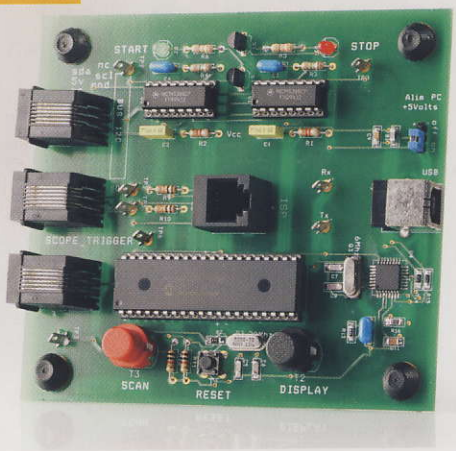
Samtliga artiklar, fotografier, krets-kortslayouter och annat är copyrightskyddat och får endast återges efter utgivarens skriftliga tillstånd. Detta gäller även delar av materialet. Allt om Elektronik har, så långt som är möjligt, kontrollerat samtliga byggprojekt. Dock tar vi inget ansvar för fel som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

Copyright:
Elektor International Media
Box 11
NL-6114-ZG Susteren
Holland

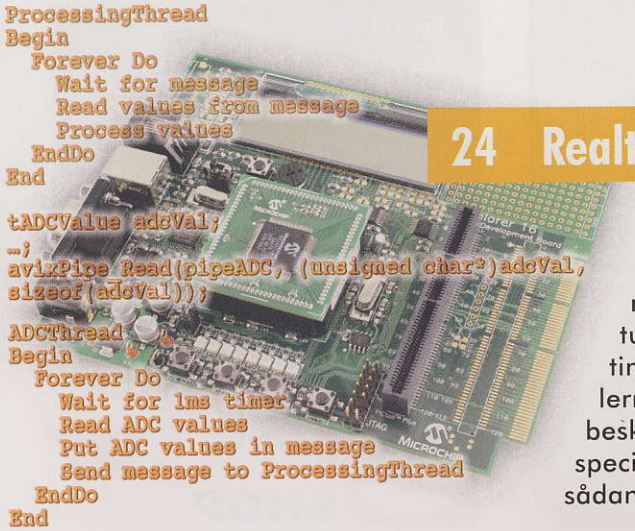
Datalogger "deLuxe"

Elektor har en lång historia när det gäller att publicera datainsamlingssystem och loggare. Den datalogger vi visar här är unik tack vare sin enkelhet och kompakta storlek; en mikrokontroller och en handfull vanliga komponenter är allt som behövs.

12 Hemligheterna hos I²C



Den I²C-analysator som beskrivs i denna artikel ansluts till I²C-bussen på en applikation och plockar fram dess start, stopp, adress, data och acknowledge signaler så att de kan undersökas. Den kan användas för felsökning eller för 'reverse engineering' på befintliga applikationer.

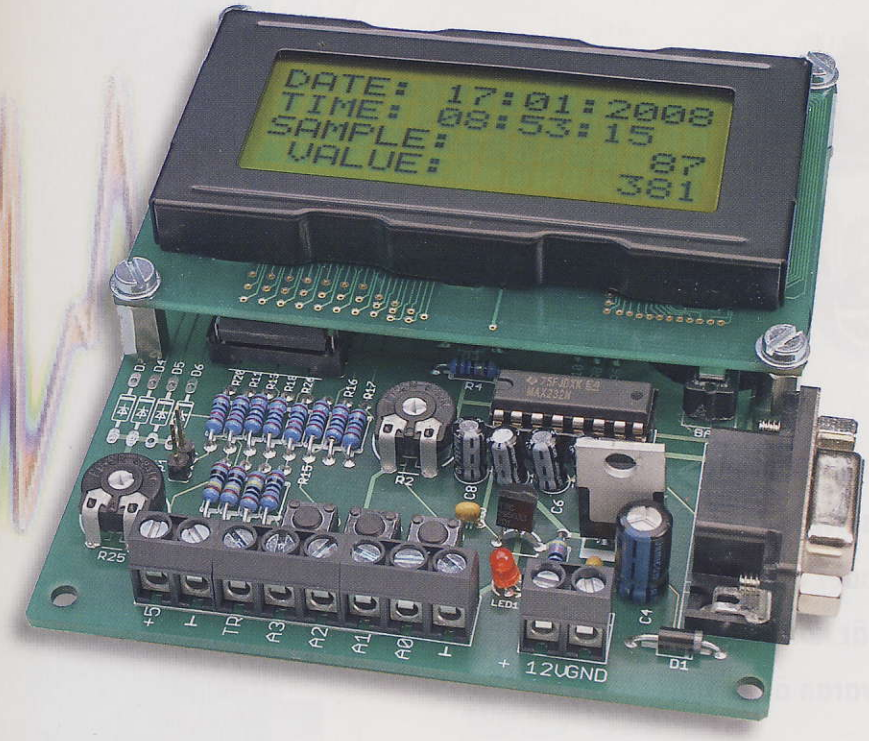


24 Realtidsberäkningar

Genom att använda ett RTOS (Real Time Operating System) kan strukturen hos mjukvaran och timingen hos mikrokontrollern förbättras betydligt. Här beskriver vi funktionen och de specifika egenskaperna hos ett sådant RTOS.

INNEHÅLL

Årgång 26
Mars 2008
nr. 227



projekt

6 Data Logger "deLuxe"
12 The Secrets of I²C
20 Robotröst
24 Realtidsberäkningar
58 Gör-det-själv LED-projektor
44 ECIO PLC

teknologi

40 EasyControl I/O
28 Från C till hårdvara



28 Från C till hårdvara

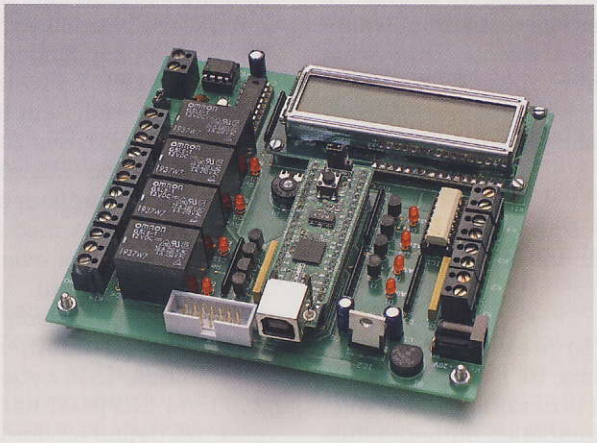
Trots den tekniska utvecklingen så utnyttjar de flesta nya apparater inte kapaciteten hos en FPGA. Huvudskälet är att det finns relativt få FPGA-experter. Den senaste utvecklingen inom EDA-området kan i stort sett eliminera detta problem.

info & marknad

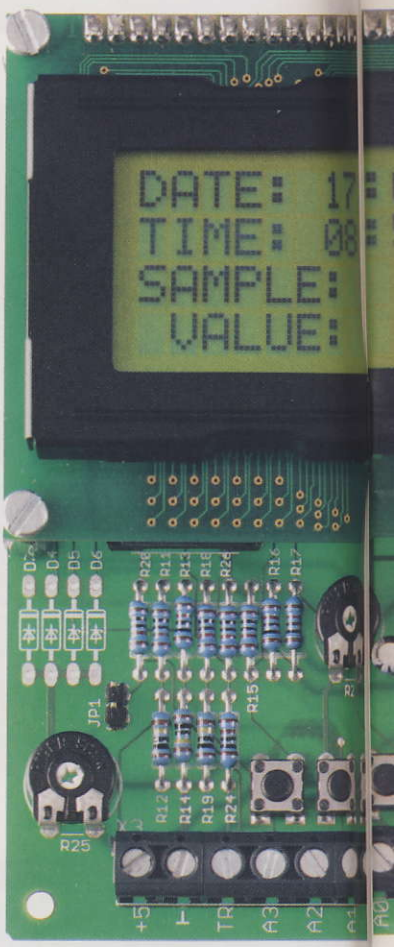
68 Elektor SHOP
50 Nästa nummer

44 ECIO PLC

En billig ECIO-modul fungerar som hjärna i ett PLC-kort som har reläer, optokopplare, CAN (!) anslutning och en LCD. All denna I/O-kapacitet tillsammans med Flowcode gör kortet till en universellt PLC (programmerbar logisk styrning) för ganska så komplexa styr- och automatiseringsprojekt.



Datalogger "deLuxe"



Vi har haft glädjen att föreslå olika datainsamlingsenheter under de senaste åren. Den som vi beskriver här är en trevlig övning i produktutveckling. Den använder ett SD-card som media för datalagringen. Hårdvarudesignen är kompakt och det gör egenskaperna hos firmware och mjukvaran ännu intressantare.

Loïc Marty

Denna datalogger används för att spara värdena från fyra analoga kanaler som lämnar spänningar från 0 V till 5 V på ett vanligt minneskort (SD – Secure Digital).
Elektor har en lång historia när det gäller att publicera datainsamlingssystem och loggar och det senaste projektet [1] är mycket populärt. Den datalogger som vi visar här är unik genom sin enkelhet och kompakta storlek, en mikrokontroller och en handfull vanliga komponenter är allt som behövs när det gäller hårdvaran.

Vår datainsamlingsenhet har flera olika arbetssätt:

1. Triggning 'on the fly' (intern trigger)... med andra ord, tryck på knappen!

2. Triggning från en extern signal. Denna kan komma, till exempel, från en sensor som arbetar med 12 V eller 24 V (till exempel en närhetsdetektor). Signalingången skyddas av en 5,1 V zenerdiod.

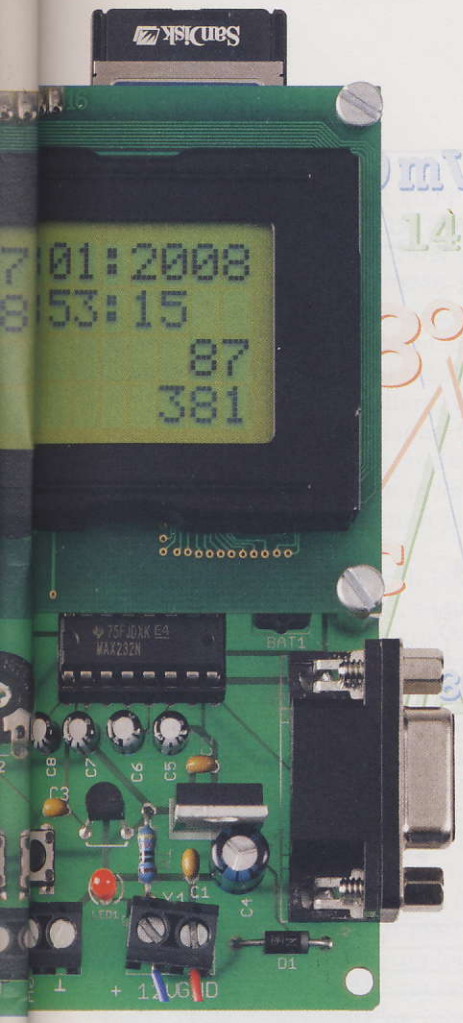
3. Spara data med 10-sekundersintervall.

4. Spara data med 1-minutsintervall (till exempel temperatursensor).

5. Spara data när signalen överskrider ett visst maximalt värde, i detta fall definierar kanal 3 referensnivån.

Elektroniken

Som schemat i figur 1 visar så är det ingenting speciellt eller komplicerat när det gäller elektroniken.
Kärnan i uppkopplingen är IC4, en högstanivås 18F452 PIC-mikrokontroller [2] som arbetar i sin maximala hastighet i HS-mod, dvs 20 MHz. Den har flera olika kringenheter som används här: A/D-omvandlare, en SPI-port (för att kommunicera med SD-kortet) och en RS232-port (för eventuell framtida mjukvaruexpansion).
Denna PIC-enhet, som är top-of-the-line i 18Fxx2-familjen, med 32 kB Flashminne, 1536 byte med RAM och 256 byte EEPROM har också en I^2C-port. Då denna använder samma pinnar som SPI-porten så visade det sig nödvändigt att använda två andra pinnar för kommunikationen med realtidsklockan (RTC), IC5. Här tar detta den fysiska formen av en PCF8583 [3], en IC som är mycket enkel att konfigurera. Några av dess nyckeldata listas separat i textrutan. Notera närvaron av ett CR2032 3-V litiumbatteri, detta används för att spara tids- och datuminformationen om den normala matningsspänningen skulle försvinna. D2 och D3, ett par dioder med låg framförspänning (Schottky) gör det möjligt att mata RTC-n permanent. Notera också att det finns en trimkondensator, C14, som används för att korrigera eventuell frekvensdrift i kristallen X1, som används som klocka (inte speciellt kritisk). RTC har en interruptutgång med open-drain med ett pull-upmotstånd, R23, vilket ger en excellent tidbas på 1 Hz, vilket förenklar tidshanteringen av 18F452.
LCD1, en stor display med 4 rader x 16 tecken driven i 4-bitarsmod, gör det möjligt att visualisera de olika data, datum, timma, inspelningsfilens nummer samt värdet hos den analoga kanalen AN0 (0 till 1024 givet att A/D-omvandlaren har en upplösning på 10 bitar).
Tre tryckknappar, S1, S2 och S3 används för att ställa in arbetssättet.
Notera också närvaron av en RS232-port som kan implementeras här. Framtida ändringar av firmeware behöver



Den effektiva enkelheten

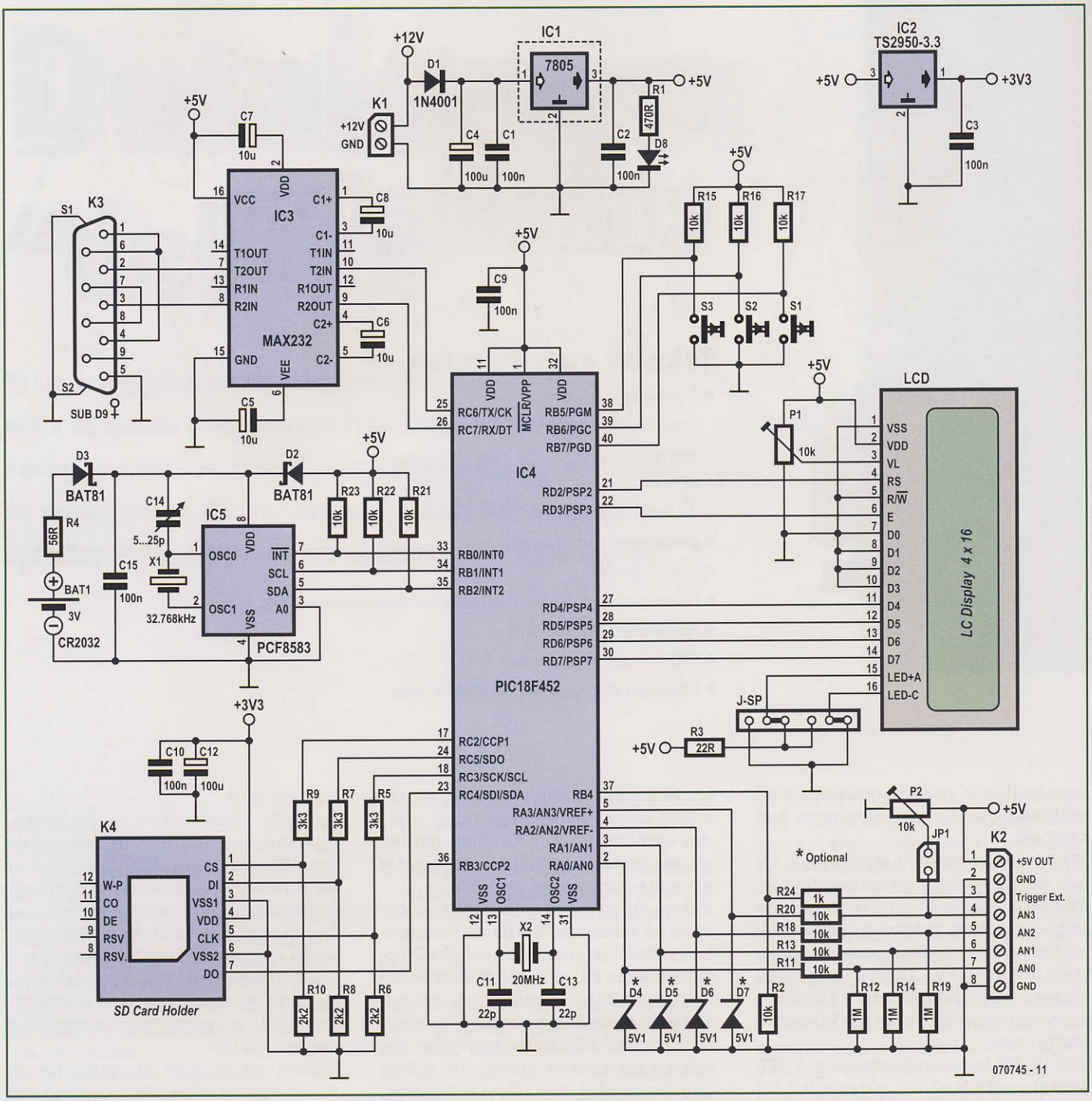
Teknisk specifikation

- 5 datainspelningssätt
  - direkt 'on the fly' (tryck på en knapp)
  - extern trigger
  - tid: varje 10 s, varje 60 s
  - vid överskridande av en förinställd nivå
- Data sparas i .txt filformat
- Upp till 99 filer
- Direkt läsning av SD-kort i ett ordbehandlingsprogram på PC-n.
- Formaterar FAT16 SD-kort
- RTC-krets för att tidsstämpla filerna
- Tillgänglig som byggsats från Elektor Shop

kanske denna port tillsammans med en bootloader för att ladda in ny kod i PIC-en.
SD-kortet kräver en matning på 3,3-V och denna spänning kommer från en TS2950-3.3 lågfallsregulator som bara behöver en liten spänningsskillnad för att lämna den nödvändiga spänningen (från 5 V till 3,3 V). Denna komponent klarar av den relativt stora strömförbrukningstopp som sker när SD-kortet startar upp.
PIC-till-SD kommunikationen är i SPI-format med 5 V på PIC-sidan och 3,3 V på minneskortssidan. I riktningen PIC till kort används motstånden R5-R10 för att erhålla den behövliga spänningsnivån på kortingångarna (CS, DATA IN, CLK). I den andra riktningen (DATA OUT till PIC) är 3,3 V nivån tillräcklig för att PIC-en skall känna igen de giltiga logiska tillstånden.
En anmärkning när det gäller de analoga och externa triggeringångarna (signalen som triggar registreringen av värdena i arbetssättet 'External trigger'), dessa linjer kan skyddas av 5,1 V zenerdioder, D4-D7, men de skall bara monteras om du vet hur du skall hantera det spänningsfall dessa orsakar. Eftersom de förskjuter mätningarna betyder det att du måste anpassa ingångssteget efter din applikation, till exempel genom att använda en spänningsdelare, för att mäta variationer över 5 V.
Notera att en bygel (jumper) kan monteras på kontakten JP1 för att lämna en fast spänning (justerbar med 10 kΩ potentiometern P2) till ingången RA3, denna jumper skall monteras när arbetssättet 'maximalt signalvärde' används (endast på AN0). Detta är mycket enkelt att justera i praktiken, det definierar den tröskel över vilken spänningstoppen tas med i beräkningen. Det är värt att notera att detta projekt ursprungligen designades med tanke på just detta.

Kretskortet

Detta projekt kan du köpa som byggsats från Elektor Shop, ordernummer **070745-71**. Figur 2 visar komponentplaceringen. Vill du etsa kortet själv finns layouten gratis på vår websida.
Med tanke på det låga antalet komponenter och storleken på kretskortet så borde bygget inte vara något problem. Kom ihåg att justera kontrasten på LCD-n med hjälp av 10 kΩ trimpotentiometern P1.
Bygget är enkelt eftersom det är vanliga komponenter som är lätta att få tag på, som PIC18F452. Displayen är en 4-raders/16-teckens med standardgrön färg, monterad med hjälp av fyra 10 mm plastdistanser. Den elektriska anslutningen görs via ett 16-poligt stifttag (hane) lött på kretskortet. Honkontakterna löds fast på displaysidan. Detta gör montering/demontering mycket enklare.
Notera att vi gjort utrymme för en RS232-sektion vilket gör uppsättningen ännu flexiblare. RS232-sektionen gör det möjligt att använda ett bootloader-alternativ (se textrutan med skärmbilder som handlar om detta).
Då det är svårt att få tag på SD-kort kontakter som reservdelar så kan du använda en från en slaktad extern kortläsare för PC (köp den absolut billigaste du hittar). Pinnarna för Lock/Unlock kontakten på SD-kortet är inte lödda (som pinnarna 8 och 9), PIC-programmet bryr sig inte om de data som dessa bär.
Om hårdvarudelen är enkel betyder det ofta att mjukvarudelen är betydligt mer komplicerad och så är fallet även här.



Figur 1. I hjärtat av kretsen en PIC18F452, omgiven av (medurs) LC-display, SD-kortkontakt, PCF8583 RTC och MAX232 nivåomvandlare.

Nyckelord

LC-display – LC-displayer (LCD) har ersatt de klumpiga effekthungriga, inflexibla LED-displayerna. Det finns många typer, med tecken (x rader om x tecken) eller med grafiska symboler. Vårt projekt använder en LCD med 4 rader om 16 tecken.

I²C-buss – Inter-Integrated-Circuit buss. Utvecklad av Philips under tidigt 1980-tal för hemautomatisering och hopkoppling av applikationer för konsumentelektronik. I²C-bussen omfattar tre linjer: SDA (Serial Data) för data, SCL (Serial Clock) för klockan och en jordlinje.

EEPROM – Electrically Erasable Programmable Read Only Memory (elektriskt raderbart läsminne). Denna typ av minne förlorar inte sina data vid ett strömavbrott.

FAT (16 eller 32) – Innan Windows ens existerade utvecklade och patenterade (delvis) Microsoft ett operativsystem för filer, för disketter såväl som för hårddiskar, som kallades för FAT (File Allocation Table). För mer information kolla på wikipedia.

IDE – Integrated Development Environment. Mer och mer komplex mjukvara behöver en centraliserad miljö från vilken man kan komma åt olika program.

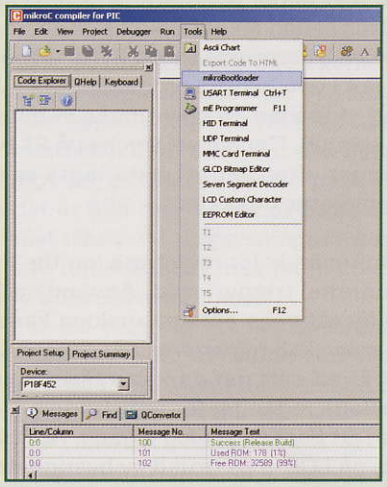
Mjukvaran

Källkodslistningen är på 500+ rader och kan inte tryckas här. Du kan hämta den från vår websida, www.alltomelektronik.se. Filnamn **070745-11.zip**. Denna arkivfil inkluderar också den .hexfil som du behöver programmera in i PIC18F452-an om du har möjligheter att programmera kontrollern. Notera att du också kan köpa denna färdigprogrammerad från oss, ordernummer är **070745-41**. I textrutan förklaras programmeringsproceduren med en bootloader.
Programmet skrevs i C och kompilerades med hjälp av MikroC, en utmärkt kompilator från Mikro Elektronika (bra teknisk support). Källkoden är ganska intuitiv, även för de som inte känner till detta språk särskilt bra (i C börjar du med funktionen main()). Det finns en gratisversion av denna kompilator som stöder kod mindre än 2 k. För detta projekt, som använder en hel del minne på grund av filhanteringen i FAT-mod, behöver du använda fullversionen. Priset är hyggligt med tanke på den tekniska kapaciteten hos denna kompilator, IDE inkluderat och speciellt användningen av de inbyggda egenskaperna. Dessa var ovärderliga och användes för att hantera filerna i FAT, deras attribut (datum/timma etc.) såväl som att hantera kommunikationen med RTC-n i implementerat I²C mod. Vi uppmuntrar er att upptäcka denna överraskande kraftfulla kompilator – se länken i slutet av artikel.
Meddelandena på LCD-skärmen talar för sig själva och använder etablerade termer som *Sample*, *Value* och *Save*.

Implementering

Låt oss börja med att beskriva de tre tryckknapparna:
S1: Används för att välja arbetssätt. Låter dig också ställa in timma och datum, i detta fall skall du hålla knappen

Bootloader



Den PIC18F452 mikrokontroller som används i detta projekt kan programmeras på vanligt sätt med hjälp av .HEX filen (logger.hex). För att göra det enklare att utveckla och speciellt avbugga en applikation, har artikelförfattaren inkluderat en bootloader i fimware.
Principen är enklare än vad den kanske verkar till att börja med. Bootloadern är ett litet program som startas så fort mikrokontrollern återställs (resets). Den scannar den seriella anslutningen och kollar efter ett svar (här ett 'r') på det meddelande den sänt (här ett 'g'). Om den tar emot den överenskomna responsen inom den tilldelade tiden (här 5 s) går den in i firmwares uppdateringsmod (med andra ord, den flashar den del av minnet som är tilldelat firmware med de data som finns på den seriella anslutningen). I annat fall sätter den pekaren på firmwareadressen som då tar över kontrollen.

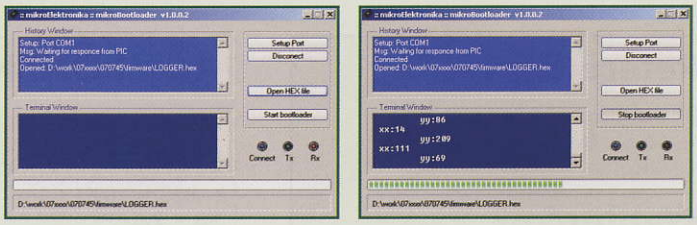
På detta sätt är det möjligt att återladda mjukvara in i mikrokontrollern vid varje uppstart så länge som den seriella anslutningen är korrekt uppsatt (57600 bps/1 stopp/8 databitar/ingen paritet/flödeskontroll med mjukvara) och om PC-värdens mjukvara är redo att kommunicera.

Nu skall vi se hur vi skall dra fördel av denna funktion. Först måste PIC-en programmeras med bootloadern på vanligt sätt för att vi skall kunna ha mjukvaran för bootloading.
Kör därefter mjukvaran (mikroBootloader) som finns under menyn [Tools] i [mikroC]. Var noga med att konfigurera serieanslutningen korrekt.

Anslut dataloggern och klicka på [connect] innan fördröjningen på 5 sekunder har förflutit. Därefter kan du koppla av, nu skall .HEX filen laddas från dataloggerns firmware med [Open HEX file] och klicka på [Start bootloader] för att flasha PIC-en.

Så fort denna operation är över skall du koppla bort och koppla in matningsspänningen till dataloggern [Hard Reset], vänta 5 sekunder och sedan skall mjukvaran köras. Glöm inte att sätta in SD-kortet innan du startar upp igen, återställningen sker direkt efter start, annars måste du återställa igen från 'off' för att kunna använda din modul.

Skärmbilderna visar arbetssekvensen när du använder bootloadern för att uppdatera mjukvaran, från det första steget till ... succé (självklart!).



MikroC – C-kompilator avsedd för PIC12, PIC16 och PIC18. En demoversion finns tillgänglig för nedladdning och enda begränsningen är att hexfilen bara kan vara max 2 kB. Inte så dåligt att börja med...

RTC – Real Time Clock, realtidsklocka. Denna IC kom ut tillsammans med de första mikroprocessorerna så att de fick en noggrann tid/datumstämpling. Du hittar dem, bland annat, på alla moderkort för PC-datorer.

SD (Card) – Secure Digital (Card). Utan tvekan det mest sålda flyttbara Flashminneskortet i världen. Dess format är en referens för många andra kort vars små dimensioner kräver en adapter, som microSD och miniSD.

SPI – Serial Programming Interface. Denna buss används för att etablera en synkroniserad seriell anslutning. Den har 4 lines:
SCLK – Serial CLocK (Master Output),
MOSI/SIMO – Master Output, Slave Input,
MISO/SOMI – Master Input, Slave Output
SS – Slave Select (aktivt låg). Används ofta för att programmera moderna processorer.

Wiki – en öppen och fri encyklopedi på nätet vid http://en.wikipedia.org/wiki/Main_Page eller http://sv.wiki.org för den svenska. Här hittar du information om i stort sett allting och du kan också bidraga själv...

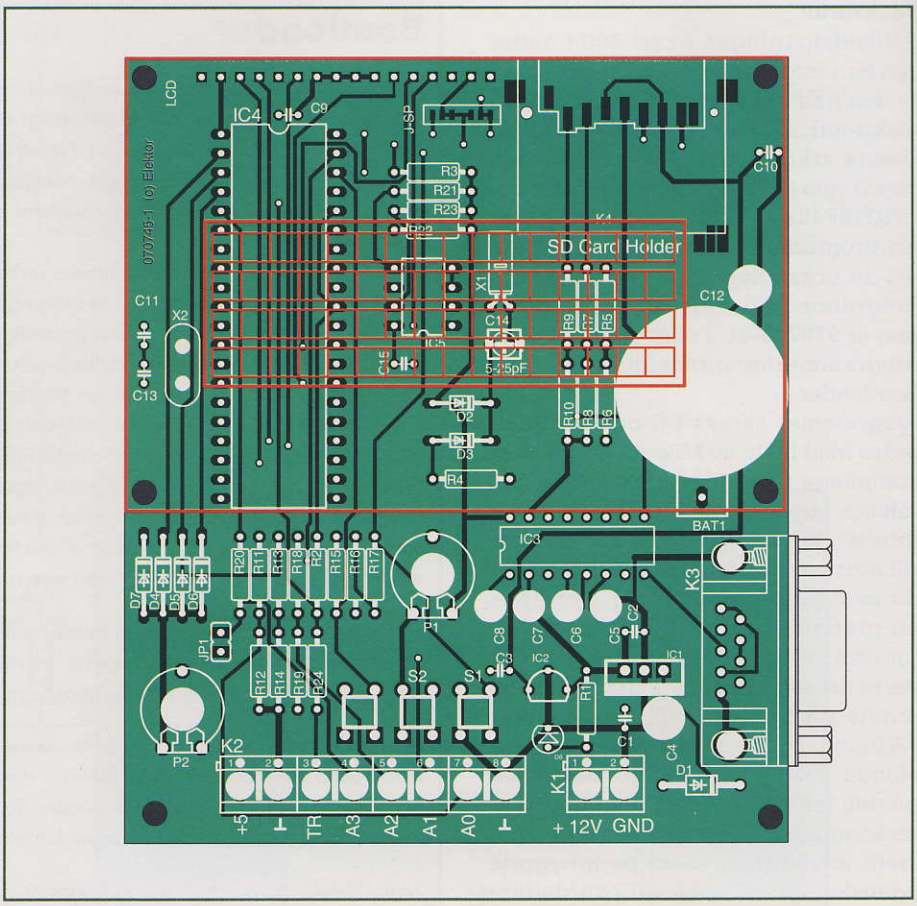
intryckt under uppstart. S2 och S3 används sedan för att öka/minska värdena och S1 används för att flytta till nästa fält i den logiska ordningen:
dag → månad → år → timma → minut → sekund. Genom att trycka på S1 två gånger efter det att data lagts in så memoreras tidklockan.

S2: Används för registrera 'on-the'fly' i 'internal trigger' mod. Används också för att välja antalet analoga kanaler som skall registreras när arbetssättet är ett sätt där du kan välja antalet analoga ingångar för registrering. Notera att bara en analog ingång, AN0, visas på LCD-n på grund av hastigheten på kodkörningen.

S3: Används för att välja en ny registreringsfil (inspelningsfil) i .txt format så att det är möjligt med direkt avläsning på en PC med SD-kort och en textredigerare. De inspelade värdena kan gå från 0 till 1024 (från 0 V till 5 V) med möjlighet att spela in från 1 till 4 kanaler samtidigt.

Till slut

Du har utan tvekan lagt märke till jumperfältet J-SP. Detta används för att vända polariteten hos spänningen till LCD-displayens bakgrundsbelysning om du använder en annan typ av LCD än den som anges i komponentlistan och som har andra egenskaper.
Några ord om SD-kortet, normalt använder SD sitt eget protokoll för att kommunicera med sin värd. Vid uppstart är det möjligt att kommunicera i seriellt mod (SPI) så länge som vissa förhållanden är uppfyllda. Detta är långsammare men enklare att hantera när man använder en mikrokontroller. Detta möjliggör interfacing med PIC 18F452 och, vad som är ännu mer be-



Figur 2. Komponentplaceringen på det kompakta kort som tagits fram för datainsamlingsprojektet.

Referens

[1] USB Datainsamlingskort, Allt om Elektronik 6/2007

Webblänkar

[2] www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1335&dDocName=en010296

[3] www.nxp.com/#/pip/cb=[type=product,path=50807/53497,final=PCF8583_5]|pip=[pip=PCF8583_5][0]

www.f1ubz.fr

www.mikroe.com (kompilator och RTC-diagram)

www.sdcard.org

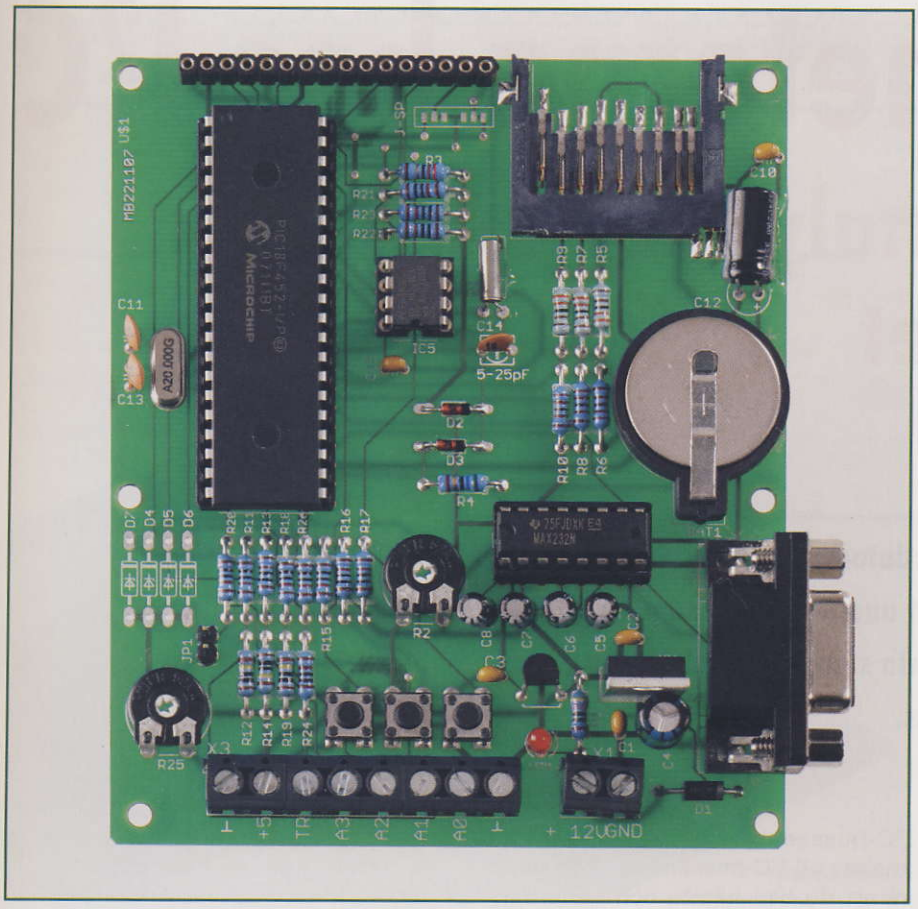
KOMPONENTLISTA

Motstånd
R1 = 470 Ω
R2,R11,R13,R15,R16,R17,R18,R20,R21,R22,R23 = 10kΩ
R3 = 22Ω
R4 = 56Ω
R5,R7,R9 = 3kΩ3
R6,R8,R10 = 2kΩ2
R12,R14,R19 = 1MΩ
R24 = 1kΩ
P1,P2 = 10kΩ trimpot

Kondensatorer
C1,C2,C3,C9,C10,C15 = 100nF
C4,C12 = 100µF
C5–C8 = 10µF
C11,C13 = 22pF
C14 = 5-25pF trimmer

Halvledare
D1 = 1N4001
D2,D3 = BAT81
D4-D7 = 5V1/400 mW zenerdiod
D8 = LED, 3 mm, röd
IC1 = 7805
IC2 = TS2950-3.3
IC3 = MAX232
IC4 = PIC18F452, programmerad, Elektor Shop # 070745-41
IC5 = PCF8583

Övrigt
S1,S2,S3 = miniatyrtryckknapp
JP1 = 2-pol SIL-stiftlist med jumper
K1 = 2-pol kopplingsplint för kretskort, raster 5 mm
K2 = 8-pol SIL-stiftlist
K3 = 9-pol sub-D sockel för kretskort (hona)
K4 = SD-kortkontakt
X1 = 32.768 kHz kristall
X2 = 20 MHz kHz kristall
BAT1 = CR2032 litiumbatteri
LCD = LCD display med 4 rader om 16 tecken, t.ex. DEM 16481
Kretskort 070745-1
Byggsats, Elektor Shop # 070745-71
Mjukvara (källkod, hexfil, PC-program), fil # 070745-11 från www.alltomelektronik.se



Figur 3. Vår färdigbyggda prototyp. LCD-displayen placeras snyggt och prydligt ovanpå huvudkortet. Kondensatorn för realtidsklockan i vår prototyp hade ett fast värde baserat på mätningar.

tydelsefullt, även med Built-ins från MikroC.
Ett 1 GB SD-kort är överraskande billigt idag och kan fås för under 100-lappen.

Kommer ni ihåg introduktions-priset för 128 MB SD-kort för fyra år sedan. Om vi inte minns fel rörde det sig om många hundralappar. 1-GB kortet fungerar perfekt i vår applikation.
Det är viktigt att kortet formatteras i FAT16-mod (dvs inte FAT32). Filerna i vilka data registreras börjar vid 1.txt och går upp till 9999.txt (räknas upp automatiskt). Självklart blir det nödvändigt att formatera kortet på nytt när det är fullt och du har överfört innehållet till din PC (upplysningsvis kan varje fil registrera ett max på ungefär 65000 diskreta samplingar).
Nu har du tillgång till ett sprillans nytt datainsamlingssystem. Använd det på bästa sätt. Vi ser fram emot din feedback.

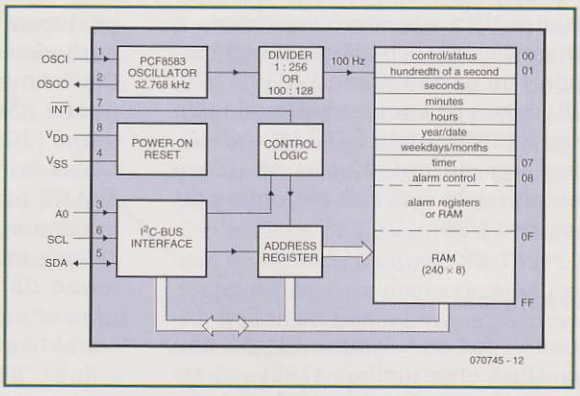
Loic.marty@neuf.fr (070745-I)

PCF8583

PCF8583 är en klocka/kalender-IC från Philips (nu NXP) med ett 2,048 bitars statiskt CMOS RAM arrangerade som 256 byte. Adress och dataöverföring sker via den 2-poliga dubbelriktade I²C-bussen. Adresslinje A0 används för att programmera hårdvaruadressen och tillåter därmed anslutning av två komponenter till bussen utan behov av ytterligare hårdvara.

Den inbyggda oscillatorn arbetar med 32.768 kHz, de åtta första byten av RAM utför klocka/kalender och räknefunktionerna. De nästa åtta byten kan programmeras som alarmregister eller användas som ledigt RAM-utrymme om det inte används på annat sätt. Resterande 240 byte är tillgänliga för användaren.

Notera att denna realtidsklocka (RTC) också kan synkroniseras med hjälp av en extern 50 Hz klocksignal.



Blockschemat över PCF8583 (Källa: NXP/Philips Semiconductors)

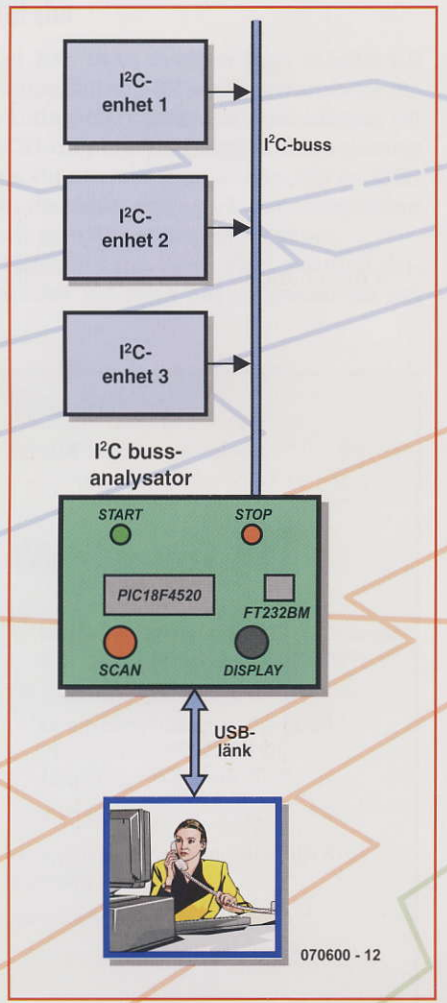
Hemligheterna hos I²C

En I²C-bussanalysator som stillar din nyfikenhet

Etienne Boyer

I denna artikel ger vi förslag på ett datorinstrument som är mycket värdefullt – för att inte säga ovärderligt – när det gäller att analysera vad som händer på I²C-bussen. Med detta kan du undersöka de mest intressanta signalerna på denna mycket vanliga, lättimplementerade anslutningsbuss.





Figur 1. Blockschemat över I²C-bussanalysatorn strippad till det allra nödvändigaste..

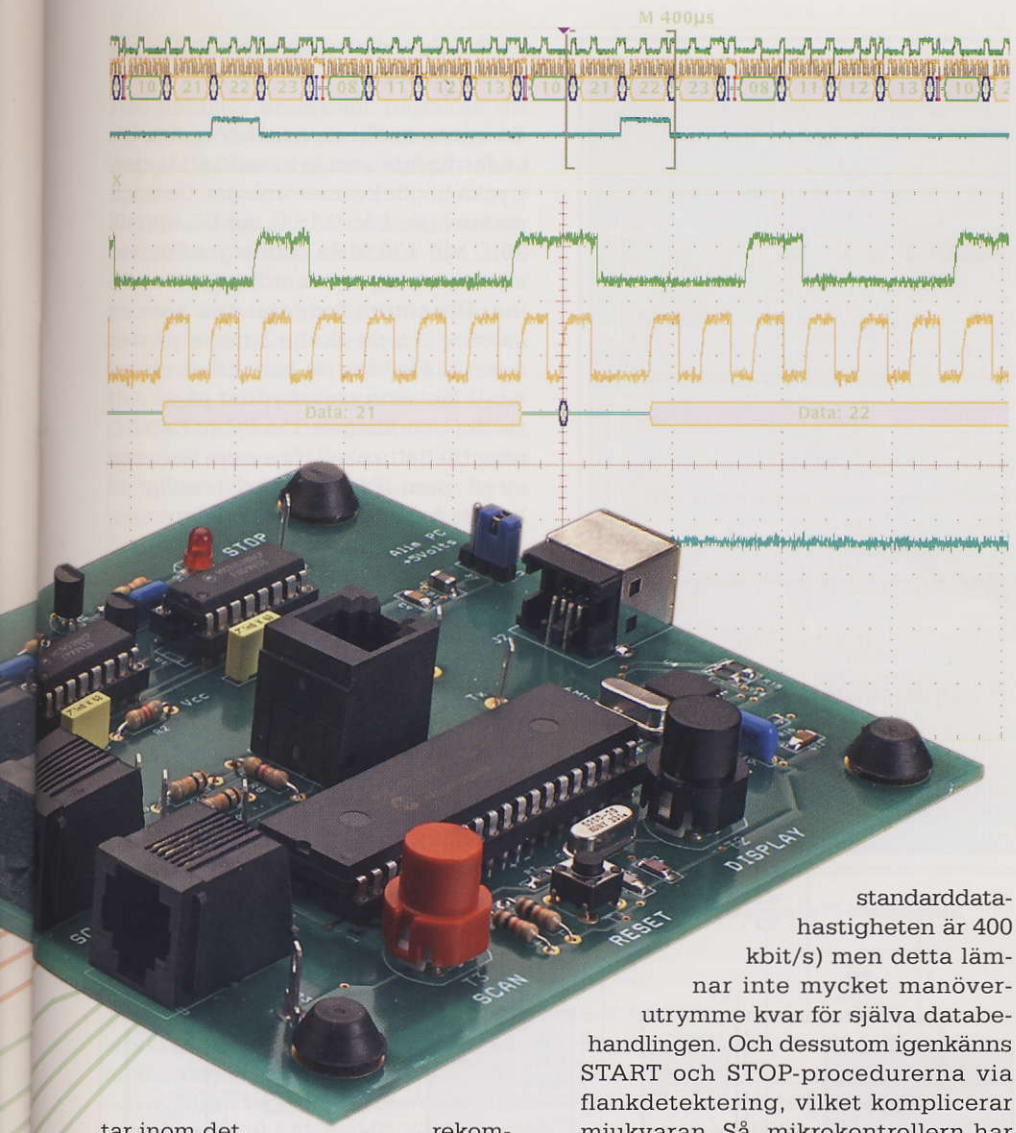
I²C-bussanalysatorn i denna artikel ansluts till I²C-bussen i en applikation så att du kan hämta och undersöka den karakteristiska informationen hos de signaler som bussen bär, speciellt då START, STOP, ADDRESS, DATA, och ACKNOWLEDGE signalerna.
Den kan användas för att felsöka en motsträvig applikation eller för 'reverse engineering' av en befintlig applikation. Enheten kommunicerar med en PC via en USB-länk konfigurerad som en virtuell port (COMx) och matas direkt från USB och behöver därför inte någon extern nätadapter (inte ens batterier).

Blockschemat

Du hittar blockschemat i figur 1. Som vi redan sagt monteras I²C-bussanalysatorn in mellan det föremål som skall undersökas och den övervakande PC-n. Upp till max tre I²C-moduler kan anslutas till analysatorn samtidigt.
Kretsens hjärta och hjärna är kombinerade i en enda PIC, en PIC18F4520, USB-länken på standardvis med hjälp av en FT232BM från FTDI [1] – en IC-krets som du redan stött på i flera Elektorprojekt som haft någonting att göra med USB.

Elektroniken

Innan vi tar en närmare titt på den egentliga elektroniken tänkte vi att det skulle vara en bra idé att lyfta fram vissa speciella punkter i kretsen med hjälp av en liten påminnare. Textrutan 'Hemligheterna hos I²C och dess buss' återger några av de viktigaste elementen hos I²C-specifikationen.
Men låt oss nu börja med vår krets. I I²C består START-proceduren av att detektera en negativ flank på SDA medan SLC-signalen är hög ('1'), utförd via monovippan IC2a hos den 4538 IC som är konfigurerad såsom visas i kretsen i figur 2.
Monovippans tidskonstant R2*C2 producerar en 8,2 µs puls som troget härmar I²C-bussens Startprocedur. Denna pulstid är ganska kompatibel med mikrokontrollerns reaktionstid men inte med det mänskliga ögats! För att kunna visa att det skett en Start med hjälp av en LED behövs det därför en andra monovippa. IC2b förlänger pulsen till runt 150 ms för att producera en synlig puls på den gröna lysdioden D2.
Detsamma gäller för STOP-proceduren, denna gång med hjälp av monovipporna i IC1 och den röda LED-en D1. Tidsdiagrammet i figur 3 visar de två START och STOP-signalerna som är genererade på detta sätt.
Notera att de två pulserna har olika längd då START-pulsen slutar tidigare eftersom signalen SLC går hög. I praktiken varar START-pulsen 3 µs och STOP-pulsen är längre än i teorin (8 µs) eftersom kretsen inte arbe-



Tekniska specar

- Analyserar 100 och 400 kbit/s I²C-buss
- Lagrar 620 kontinuerliga I²C-händelser
- Hårdvarudetektering av START och STOP. Visas med 2 LED
- USB-kommunikation med virtuell comport
- Matas med 5 V via USB-porten
- PIC programmerad i C (CCS-kompilator)
- Windows man/maskininterface i C++ Builder V5 (Borland)

tar inom det rekommenderade området (100 µs till 100 ms).

Om man tittar på kretsen i figur 4 så ser man att det inte finns så mycket elektronik, en PIC, en USB interface-IC från FTDI och ett par dubbla monostabila IC. Låt oss titta lite närmare på deras funktioner.

Kretsens hjärta bildas av en mikrokontroller från Microchip. PIC18F4520 [2] är en förbättrad version av 16F452 men fortfarande pinkompatibel, inte bara med den sistnämna utan också med den berömda 16F877. Dess funktion är att analysera, genom scanning, ingångslinjerna RC3 och RC4 hos PIC-en, direkt drivna av SCL (**S**erial **CL**ock) och SDA (**S**erial **DA**ta) signalerna. Tryckknappen S3 (SCAN) startar analysen. En närmare titt på kretsen leder ofelbart till frågan: har mikrokontrollern verkligen tid att scanna I²C-bussen?

Svaret är både ja och nej. Längden på instruktionen är 0,2 µs, vilket representerar ungefär 12 gånger längden på en bit (den maximala standarddatahastigheten är 400 kbit/s) men detta lämnar inte mycket manöverutrymme kvar för själva databehandlingen. Och dessutom igenkänns START och STOP-procedurerna via flankdetektering, vilket komplicerar mjukvaran. Så, mikrokontrollern har inte tid att scanna I²C-bussen. Därför finns det en hårdvarulösning som backar upp mjukvarulösningen genom att använda flankdetekterande monovippor (se detaljerna i textrutan 'Hemligheterna hos I²C och dess buss').

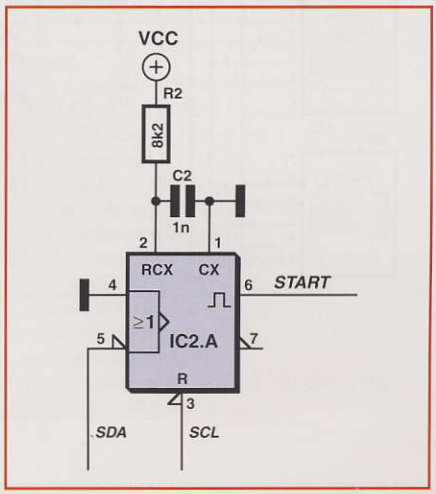
Nu när START-detekteringen har skett behöver vi för ögonblicket bara detektera varje SLC-klockpuls och sampla SDA-data. Så fort som signalerna har analyserats kommer mikrokontrollern att lagra varje händelse i minnet på samma sätt som en datalogger. De fyra händelserna och START, BYTE, ACKNOWLEDGE, och STOP. Minnet fylls upp med hastigheten av trafiken på bussen och när det är fullt överförs det via en serielänk till PC-n (*USB är per definition en seriell länk och inte en parallell länk*).

Om trafiken på bussen är för långsam kan vi använda knappen S2 (DISPLAY) för att skicka över minnesinnehållet till PC-n utan att behöva vänta på att minnet skall bli fullt.

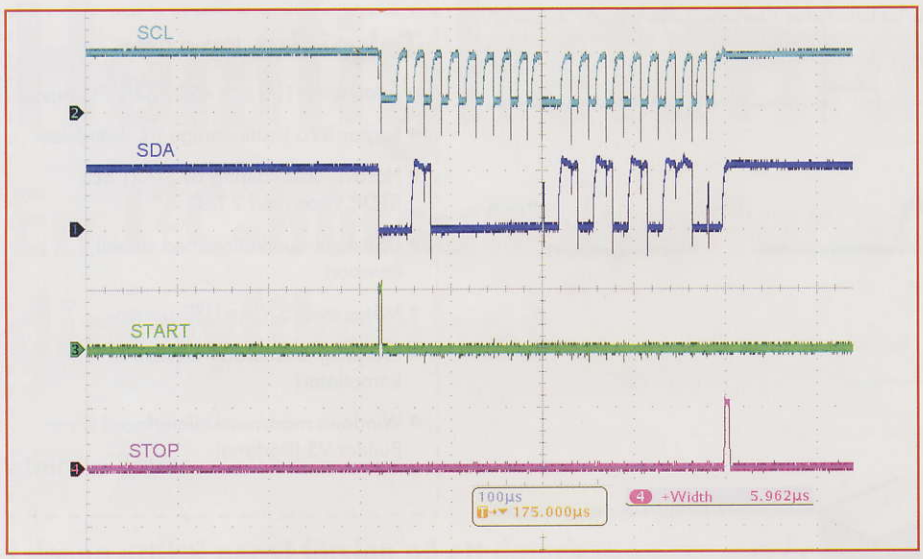
Kommunikationen med PC-n sker med hjälp av den nu standardiserade FT232 IC-n från FTDI, som använder USB-n i CDC-mod (**C**ommunication **D**evice **C**lass). Det finns bara några få komponenter runt denna IC, i huvudsak en 6 MHz kristall och dess två kollegor C7 och C7, samt en 4-polig USB-kontakt av typ B.

Ett par röda och gröna LED är associerade med händelserna STOP (D1) och START (D2).

Två motstånd på 27 Ω skyddar Data+ och Data- linjerna. IC-n FT232BM kopplar +5 V till motståndet R16, vilket indikerar två saker till värd-USBn (PC-n). Först av allt att det finns en kringutrustning inkopplad och att det gäl-



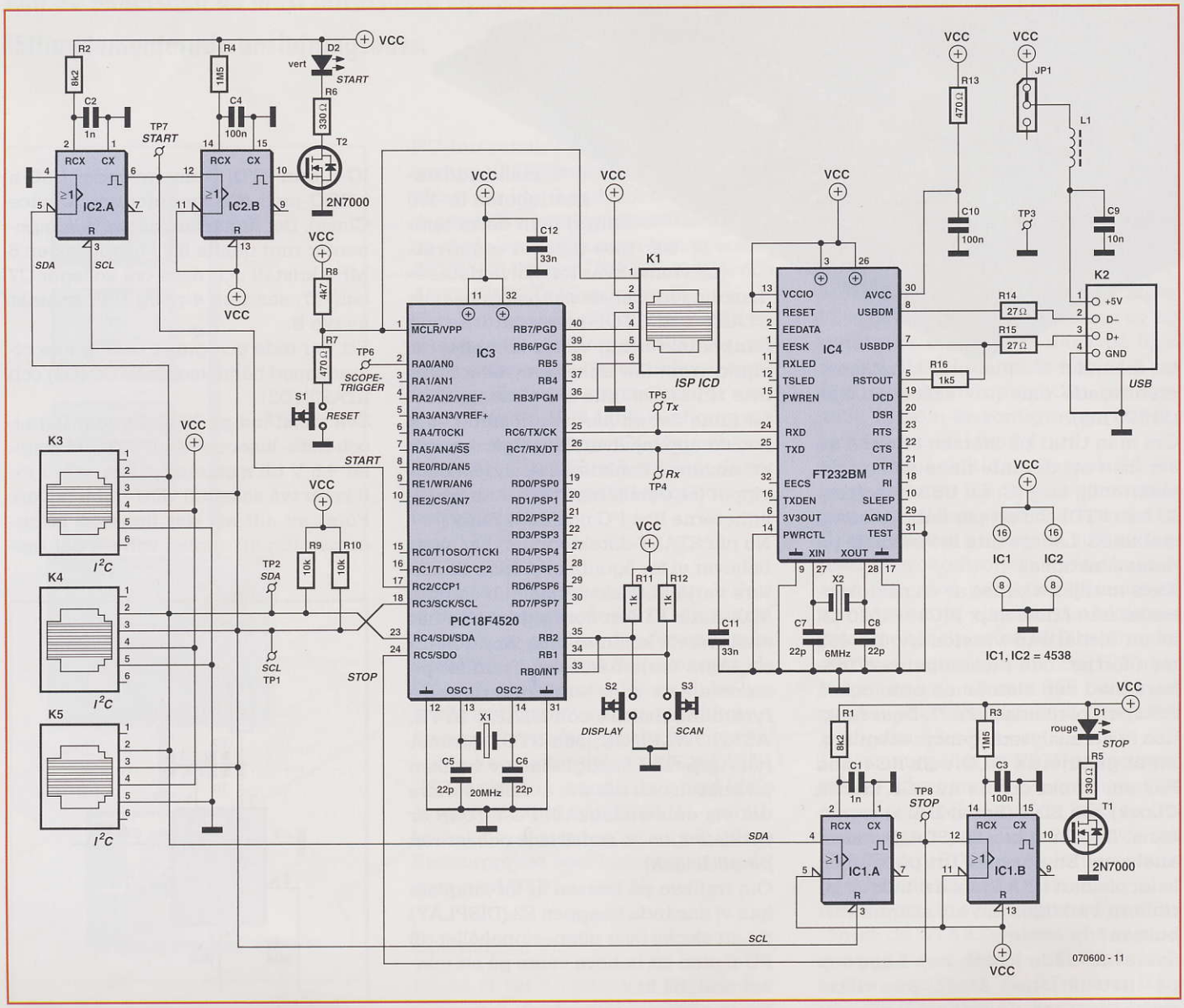
Figur 2. En monovippa triggad av en fallande flank.



Figur 3. Skapandet av Start och Stopp.

ler Full Speed mod eftersom R16 är anslutet till Data+.

En intressant komponent är spolen L1, en ferritpärla som är avsedd att undertrycka högfrekvensstörningar. Dess impedans går från 0,15 Ω vid DC upp till 70 Ω vid 100 MHz och avger därmed all elektromagnetisk störning som värme. Dessutom fungerar den som en säkring om du skulle komma på den missriktade idén att kortsluta kretsens +5-V matning och skyddar på så sätt strömförsörjningen från PC-n. I anslutning till detta skall du notera närvaron av jumpern JP1, som gör det möjligt att koppla bort PC-ns strömförsörjning.

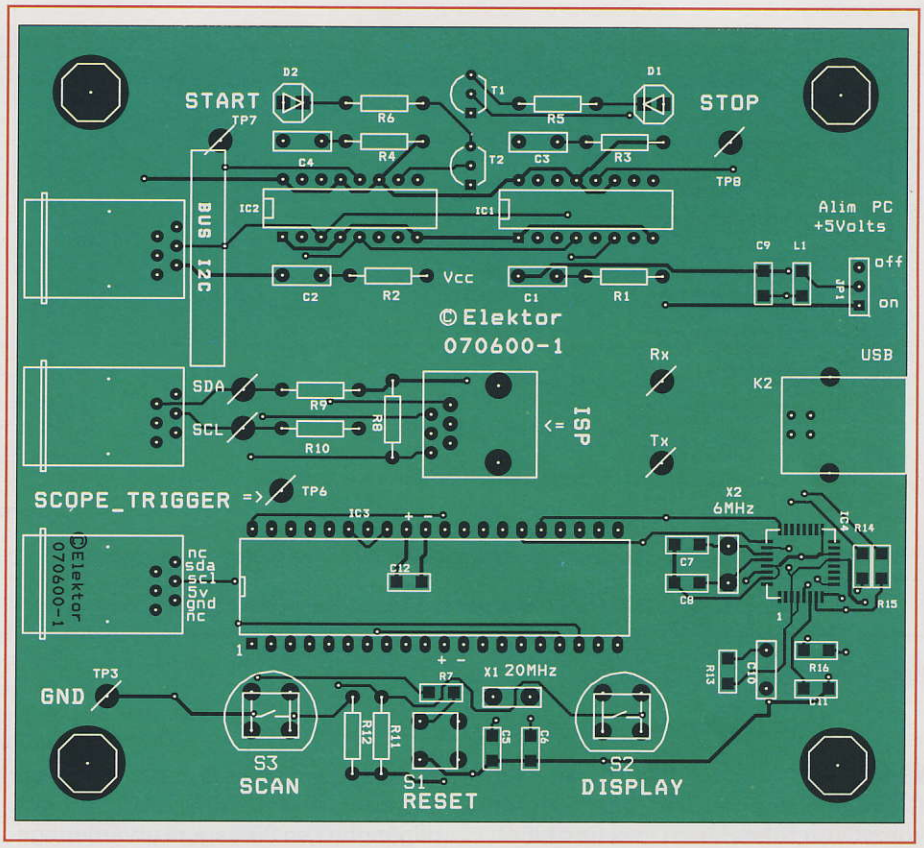


Figur 4. Elektroniken uppgår inte till mycket. Ett PIC/USB-chip omgivet av en handfull kontakter av olika typer...

Kretskortet

Det behöver knappast påpekas att en sådan här krets förtjänar ett kretskort. Komponentplaceringen visas i figur 5.
Det första steget, och också det svåraste, består i att montera IC4, den enda SMD-IC som används i detta projekt. Som alltid när det gäller ytmonterade komponenter (SMD) så kräver lödningen en stadig hand och tålamod. Lödplättarna för IC4-ans LQFP-32 kapsel har förlängts så att det skall vara enklare att löda denna komponent med en lödpenna. Börja med att vända kretsen rätt (pinne 1 är den som ligger omedelbart till vänster om den runda fördjupningen. På komponentplaceringen är positionen av ben 1 markerat med en liten '1'. Börja med att löda fast två diagonalt motsatta ben. Om nu resten av benen är korrekt anpassade till lödplättarna kan du snabbt löda hela kretsen med en finspetsad lödpenna och ett klent lödtenn. Använd ett förstoringsglas för att kolla kvalitén på lödställena och att det inte finns några kortslutningar. Sen kan du gå vidare med SMD-komponenterna, i 1206 kapslingar, och därefter de små hålkomponenterna, motstånd, kristall, kondensatorer, lysdioder och transistorer (tänk på polariteten). Därefter monterar du sockar (högkvalitets!) för IC1, IC2 och IC3 och avslutar med att montera de olika kontakterna, RJ-11 för K1, K3, K4, och K5, samt USB typ B för K2.
Allt som återstår är att montera tryckknapparna S2 och S2 samt resetknappen (S1).

Viktigt: om du matar kretsen via analysatorns USB-port, vilket är det vanliga fallet, måste du montera jumpern i 'on' läge på JP1.
Efter en sista koll på projektet för att se att det inte finns några fel eller kortslutningar så kommer då ögonblicket att ansluta kortet till en PC via en USB-kabel. Kolla att det finns matningsspänning till de korrekta punkterna på IC-hållarna med hjälp av en multimeter. Om allt är OK kan du koppla från analysatorn och montera IC1-IC3. Kola polariteten så att du vänder dem åt rätt håll. Kortet har ett antal testpunkter (TP1-TP8 på komponentplaceringen, testpunkterna TP1, TP2, TP4 och TP5 motsvarar SCL, SDA, Rx respektive TX-linjerna som, om du vill, kan förses med lödstift som i vår prototyp.
För testning kan kretsen matas med



Figur 5. Komponentplaceringen för I²C-bussanalysatorn.

5-V från I²C-busskontakten eller via kopplingstrådar som löds fast direkt på kortet.
Felsökning av hårdvarudelen är enklare eftersom monovipporna är oberoende av mjukvaran. När kretsen ansluts till en befintlig I²C-buss skall de fyra monovipporna reagera på ankomsten av en Start och Stop genom att tända den gröna respektive röda lysdioden. Därefter kan PIC-en programmeras.

Mjukvara

Mikrokontrollerns källprogram är skrivet i C och kompilerat med hjälp av

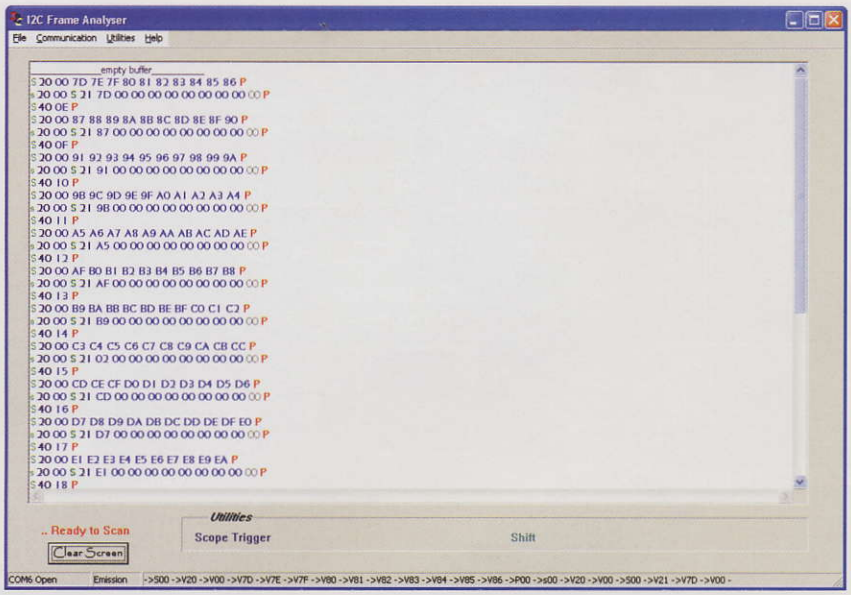
KOMPONENTLISTA

Motstånd
R1,R2 = 8kΩ2
R3,R4 = 1MΩ5
R5,R6 = 330Ω
R7,R13 = 470Ω
R8 = 4kΩ7
R9,R10 = 10kΩ
R11,R12 = 1kΩ
R14,R15 = 27Ω
R16 = 1kΩ5

Kondensatorer
C1,C2 = 1nF
C3,C4,C10 = 100nF
C5-C8 = 22pF
C9 = 10nF
C11,C12 = 33nF

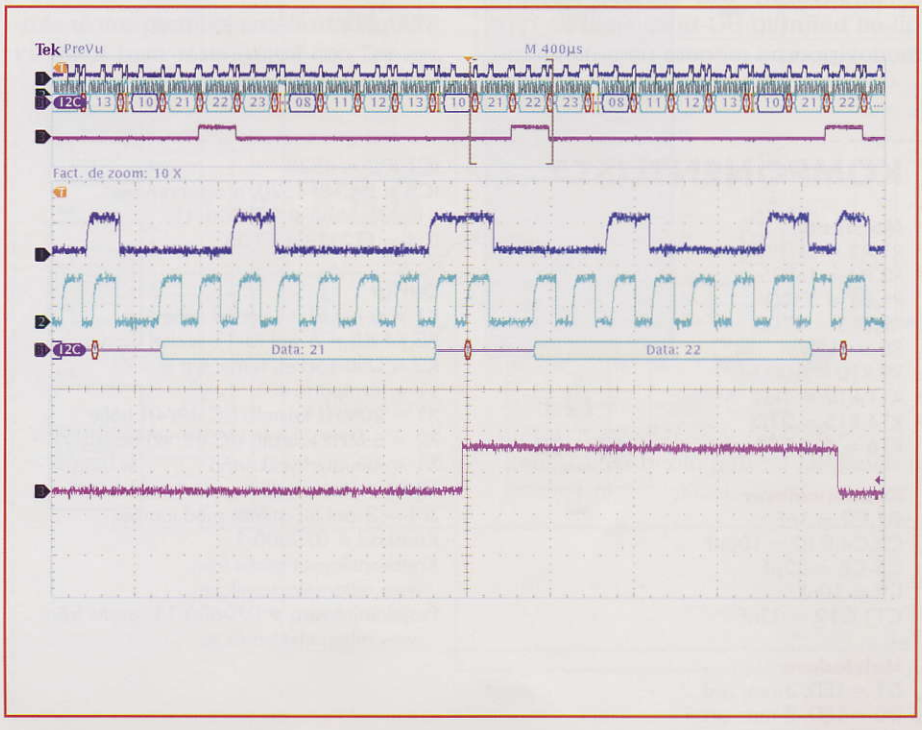
Halvledare
D1 = LED, 3mm, röd
D2 = LED, 3 mm, grön
T1,T2 = 2N7000
IC1,IC2 = 4538
IC3 = PIC18F4520, programmerad, Elektor shop # 070600-41
IC4 = FT232BM (FTDI)

Övrigt
K1 = 6-pol RJ-11 sockel (stående)
K3,K4,K5 = 6-pol RJ-11 sockel (liggande)
K2 = USB-sockel, hane, typ B
L1 = ferritpärla
X1 = 20MHz kristall, HC 49/4H hölje
X2 = 6 MHz kristall, HC 49/4H hölje
S1 = miniatyrtryckknapp
S2,S3 = 'D6' tryckknapp (röd och svart)
JP1 = 3-pol SIL-stiftlist med jumper
Kretskort # 070600-1
Kretskortslayout gratis från www.alltomelektronik.se
Projektmjukvara # 070600-11, gratis från www.alltomelektronik.se



Figur 6. Skärmdumpen för Monitor I2C programmet.

korskompilatorn PCH V4.010 från CCS (Custom Computer Services). Denna kompilator tolererar en viss flexiblitet i det akademiska C-språket och är på så sätt mycket lämplig för att programmeras av elektroniktekniker.
Utvecklingen utförs under MPLAB V.7.62. Programmet, som körs under Windows, är skrivet i C++ Builder V5 (Borland). Mjukvaran för Windows kan också kompileras under CodeGear 11, den senaste IDE från Borland, tillgänglig i en 30-dagars utvärderingsversion. Där har du det alltihop.
En ISP-kontakt, K1, sitter i mitten av kretsen för avbuggning (ICD = **I**n-**C**ircuit **D**ebugging) och *in situ* programmering (ISP = **I**n **S**ystem **P**rogramming) av mikrokontrollern, dvs du behöver inte ta bort mikrokontrollern



Figur 7. Skärmdumpen för Scope_trigger.

Snabbtitt på mjukvaran

När nu elektroniken är så enkel så måste det program som laddas in i mikrokontrollern vara kraftfullt. Du kan hämta det från vår websida och filnumret är 070600-11.zip. Låt oss ta en titt på några av dess praktiska aspekter.

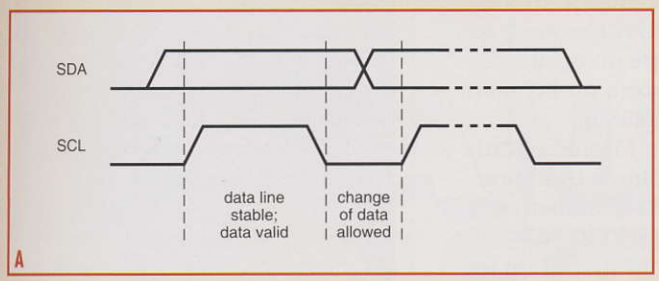
Man/maskininterfacet: installation
Denna applikation, skriven i C++ Builder V5.0 körs under Windows och är lätt att installera genom att kopiera den körbara filen **monitor_I2C.exe**. Denna applikation kräver att drivrutinerna finns installerade på PC-n. För att göra detta är det värt att konsultera FTDI-s websida [3] och tidigare Elektorartiklar om detta ämne.

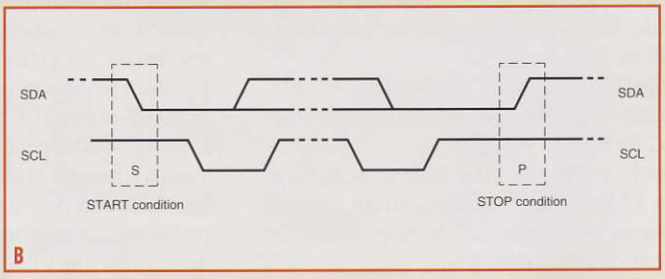
RS-232 konfiguration
Vid uppstart kommer det fram en dialogruta där du kan välja den Virtual Com Port via vilken USB-länken tar emot data skickade från PIC-en i form av en standard asynkron serielänk (överföringshastighet 128 000 baud). Därefter indikerar statusraden att serieporten har öppnats korrekt.

Visa resultaten
Scanningen startas genom att trycka på knappen S3 (SCAN). I²C-händelserna registreras sedan av kretsen och visas sedan enligt en färgkod som motsvarar I²C-händelsen. Skärmdumpen i figur 6 visar huvudskärmen när insamlingen är avslutad och du kommer att se START-erna i grönt, STOP-en i rött, adresserna i ultramarinblått och data i kungsblått. Denna display kommer dock att modifieras av värdet på acknowledgebiten, om den finns närvarande ser du dessa två i blått, men om den inte finns är dessa grå.
Statusraden visar också formatet hos de koder som sänds på den seriella länken mellan kretsen och PC-n (ASCII-kodning).
Exempel:
->S00 för START
->V20 för ett acknowledgedbyte med ett värde på 20 (HEX)
->v20 för ett unacknowledgedbyte med ett värde på 20 (HEX)
->P00 för STOP.

Synkroniseringsfunktion för oscilloskop
Denna egenskap låter oss konfigurera inställningen av ett konfigureringsbyte, som när det finns i 'framen' triggar en synkpuls på pinnen SCOPE_TRIG-

Hemligheterna hos I²C och dess buss





I²C (Inter Integrated Circuit) bussen skapades av Philips som på den tiden var en av de ledande tillverkarna av audioutrustning. Det huvudsakliga applikationområdet var hemautomation och hemelektronik under det tidiga 80-talet. Mikroprocessorer började hitta vägen in i TV-apparater och Philips behövde hitta en billig enkel teknik för att koppla ihop de olika elektroniska delsystemen i sådana apparater.

I²C-bussen är en synkron seriell buss med bara tre linjer. Data (SDA), Clock (SCL) och jord (används som referens).

Operationen av bussen baseras på ett koncept med Master (den komponent som hanterar kommunikationen, genererar klockan och sänder data) och Slaven (den komponent som tar emot data och bekräftar detta med en 'acknowledge' signal. Men låt oss inte bli lurade: trots sin enkelhet kan bussen hantera flera mikrokontroller utan några konflikter så länge som vissa regler åtlyds korrekt.

De fyra viktigaste situationerna i detta protokoll illustreras nedan.

1) Överföringen av en bit på I²C-bussen (Figur A).

Klockan har inte riktigt 'formen' av en realtidsklocka eftersom den kan ha olika puls/pausförhållanden (inom vissa gränser).

2) START och STOP-konditionerna (Figur B).

Vid starten av kommunikationen går SDA-linjen till '0' meden SCL-linjen är vid '1'. I slutet av kommunikationen, när SDA-linjen har gått tillbaka till '1' och SLC-linjen också är vid '1', har vi StopBit.

3) Överföring av data på I²C-bussen (Figure C).

När denna kondition definierats placerar mastern MSB på SDA-linjen. Denna bestyrker data genom att kort tvinga SCL-linjen 'hög'. Den fortsätter på samma sätt för de olika bitarna ner till LSB (minst signifikanta bit). När sändningen är över tvingar slaven SDA-linjen låg, detta är...

4) Bekräftelsesignalen (acknowledge) på I²C-bussen (Figur D).

Slavkomponenten lämnar denna signal för att visa att den har tagit emot alla data. Om allting är OK så tvingar den linjen till '0'.

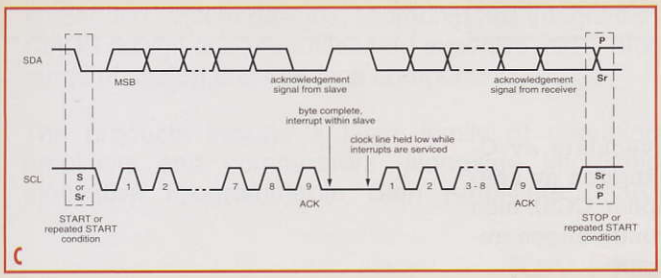
Trots sin enkelhet kan I²C-bussen hantera relativt komplexa operationer. De olika oscilloskopkurvorna i denna artikel illustrerar denna operation synnerligen väl.

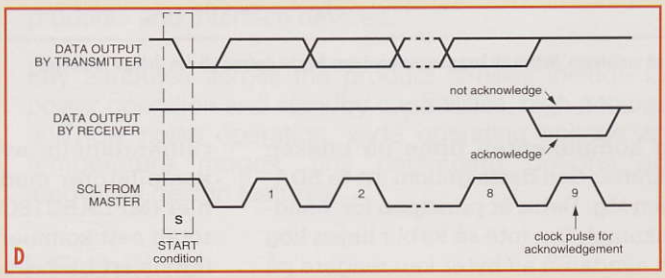
Det är värt att notera att en masterkomponent också kan ta emot data från en slav (mastermottagare).

Om du vill veta mer finns allt i I²C-busspecifikationen från 2000:

http://www.nxp.com/acrobat_download/literature/9398/39340011.pdf

(Illustrationer: Philips Semiconductors)





GER för att synkronisera med ett oscilloskop i externt triggermod. Detta gör det möjligt att visa formerna på SDA och SLC-signalerna i just det ögonblicket. Skärmdumpen i figur 7 visar, som exempel, SCOPE_TRIGGER- signalen för synkbyte 21 (HEX).

Den övre delen av kurvan visar tydligt hur varje gång som detta byte kommer så vippar signalen under längden av nästa byte. Den centrala sektionen visar de förstorade delen med en zoomfaktor på 10 och visar övergången hos SCOPE_TRIGGER-signalen i ögonblicket för den nionde SLC-klockpulsen. Detta ögonblick, som normalt motsvarar ett svar från mottagaren, låter oss läsa en '1' på SDA, vilket indikerar att det inte varit någon bekräftelse (acknowledgment): från detta ögonblick anses I²C-kretsen med adressen 10 (HEX), för vilken databyten 21 22 23 var avsedda, vara bortkopplad från bussen eller felaktig.

Fördröjning

En annan egenskap gör det möjligt att fördröja starten av registreringen av ett visst antal händelser, i detta fall är de ersatta med punkter på displayen.

I²C-summering

En inbyggd hjälpsida inkluderar en kort summering av några definitioner av händelser på I²C-bussen. Dessutom visar den inspelningen av riktiga signaler.

För att sända data över I²C-bussen

är det nödvändigt att övervaka två specifika konditioner, Start och Stopp. Startkonditionen motsvarar en fallande flank på SDA medan samtidigt SLC är hög.
Stoppkonditionen motsvarar en stigande flank på SDA medan samtidigt SLC är hög.
Därefter triggar åtta pulser som lämnas av klockan samplingen av de åtta bitarna i ett byte och börjar med MSB (mest signifikanta bit).
Den nionde klockpulsen triggar en respons, en bekräftelse från den mottagande komponenten av bytet.

använder standardkomponenter (som CMOS logik-IC PIC mikrokontroller, USB-interface) gör det möjligt att analysera de signaler som finns på en I²C-buss. En ytterligare utveckling av kretsen vore att montera en PC med en USB-stack (PIC18F4550).
Denna lösning skulle förenkla hårdvaran (genom att eliminera USB-interfacekretsen och 6 MHz kristallen) och förbättra hastigheten då PIC 18F4550 använder en PLL för att generera en klocka på 48 MHz. Nackdelen skulle vara en ytterligare komplexitet hos mjukvaran. Det finns bibliotek som

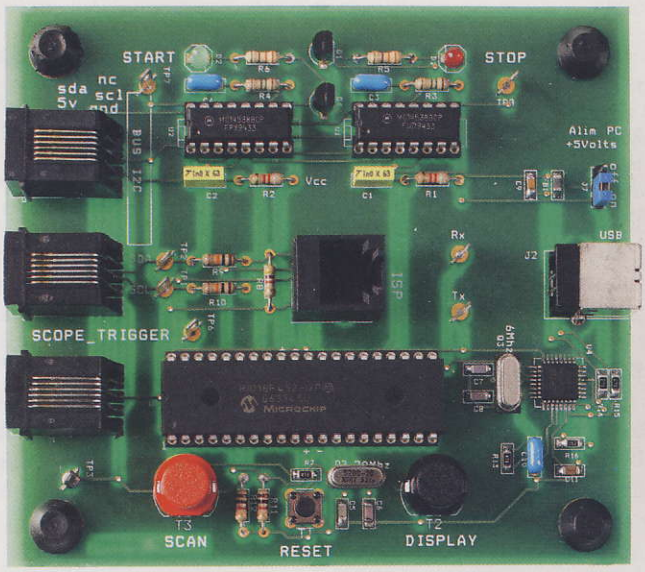


Bild på prototypen. Notera att komponentplaceringen kan ha ändrats till den slutliga versionen.

Om komponenten finns på bussen bekräftar den detta genom att ta SDA-linjen låg. Detta är principen för 'handskakning'. Om inte så förblir linjen hög och sändaren av bytet kan reagera på detta på lämpligt sätt.

Och vad mer?

Inte så mycket! Allt du behöver göra nu är att ansluta en applikations I²C-buss till I²C-bussanalysatorn, ansluta den senare till en PC, starta programmet **monitor_I2C.exe**, trycka på SCAN-knappen följt några sekunder senare med ett tryck på S2, DISPLAY, och sedan vänta på att de första data skall visas på skärmen.

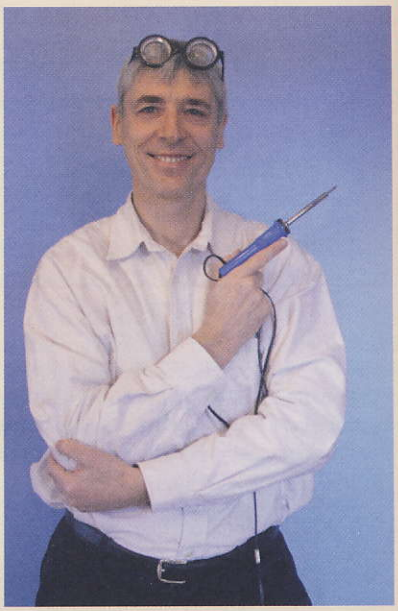
Sammanfattning

Denna enkel-att-använda krets som tillhandahålls av utvecklare av C-kompilatorer med mängden av källfiler (MPLABC18Compiler, CCS), men totalt sett kommer avbuggningen troligtvis att bli besvärligare.

Med detta praktiska verktyg kan du se vad som händer på bussen hos den datalogger som vi presenterar på annat ställe i tidningen då denna också har en I²C-buss som ansluter realtidsklockan till resten av systemet. God jakt!

(070600-I)

Författaren



Författaren läste till ingenjör vid INSA i Lyon och började sedan att undervisa.

Han har undervisat i elektronik vid tekniska högskolor under många år och upplevt och lärt vidare den fantastiska utvecklingen inom teknologin: diskreta komponenter (2N2626 unijunctiontransistorer), minne, mikroprocessorer, stordatorer och sedan utvecklingen av PC-datorer.

Utbildningen har också utvecklats och går idag mycket ut på att förstå de komplexa elektroniska systemen: installation, konfiguration och felsökning.

Weblänkar och bibliografi

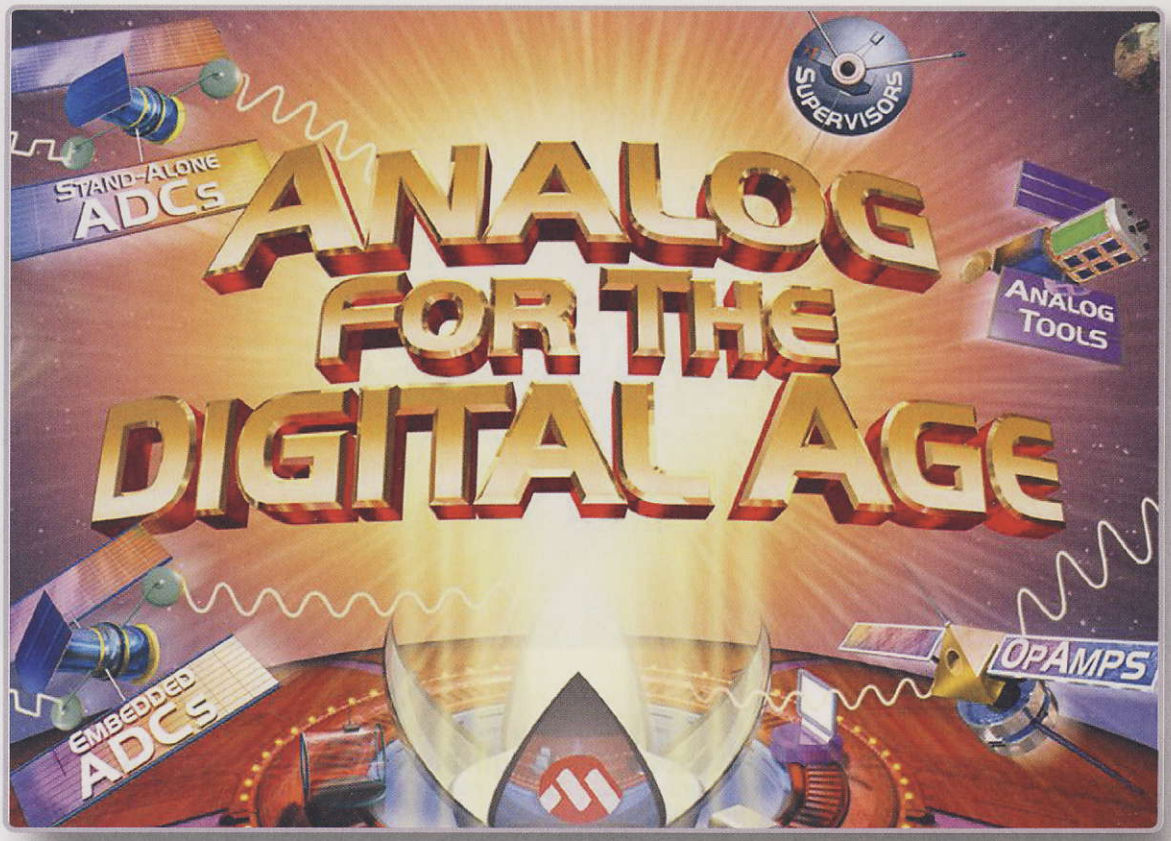
[1] FT232BM Datablad:
www.ftdichip.com/Documents/DataSheets/ds232b18.pdf

[2] PIC18F4520 Datablad:
ww1.microchip.com/downloads/en/DeviceDoc/39631a.pdf

[3] FTDI websida:
www.ftdichip.com

[4] CCS kompilator:
www.ccsinfo.com

Innovative Analog Components



Innovative Analog Components from the Leading Supplier of 8-Bit Microcontrollers

In response to ever increasing requirements to minimise power, space and noise, and maximise accuracy in embedded system designs, Microchip has integrated its CMOS design and manufacturing expertise into a broad range of analog and interface components.

The products include a large family of operational amplifiers and comparators, alongside Microchip's innovative Programmable Gain Amplifiers (PGAs); power management and conversion devices; thermal measurement and management devices; mixed signal products and interface devices.

Key attributes across the product families include low power operation and standby capabilities, high accuracy and low noise operation, wide operating voltage and temperature ranges, tiny packaging options and innovative design features.

Key Product Families:

- Temperature Sensors
- Voltage Regulators
- Charge Pumps and Switching Regulators
- System Supervisors
- Power MOSFET Drivers
- Battery Chargers
- Operational Amplifiers
- Programmable Gain Amplifiers
- Comparators
- A/D Converters
- D/A Converters
- Digital Potentiometers
- CAN Interfacing Products
- Infrared Interfacing Products

For more information go to: www.microchip.com/analog

microchip DIRECT
www.microchipdirect.com

Now Pb-free!
RoHS Compliant

MICROCHIP
www.microchip.com/analog

The Microchip name and logo are registered trademarks of Microchip Technology Inc. in the USA and other countries. All other trademarks and registered trademarks are the property of their respective owners.
© 2006 Microchip Technology Inc. All rights reserved. ME160Eng/12.06



Cylon Voi

Robotröst från en ATtiny mikrokontroller

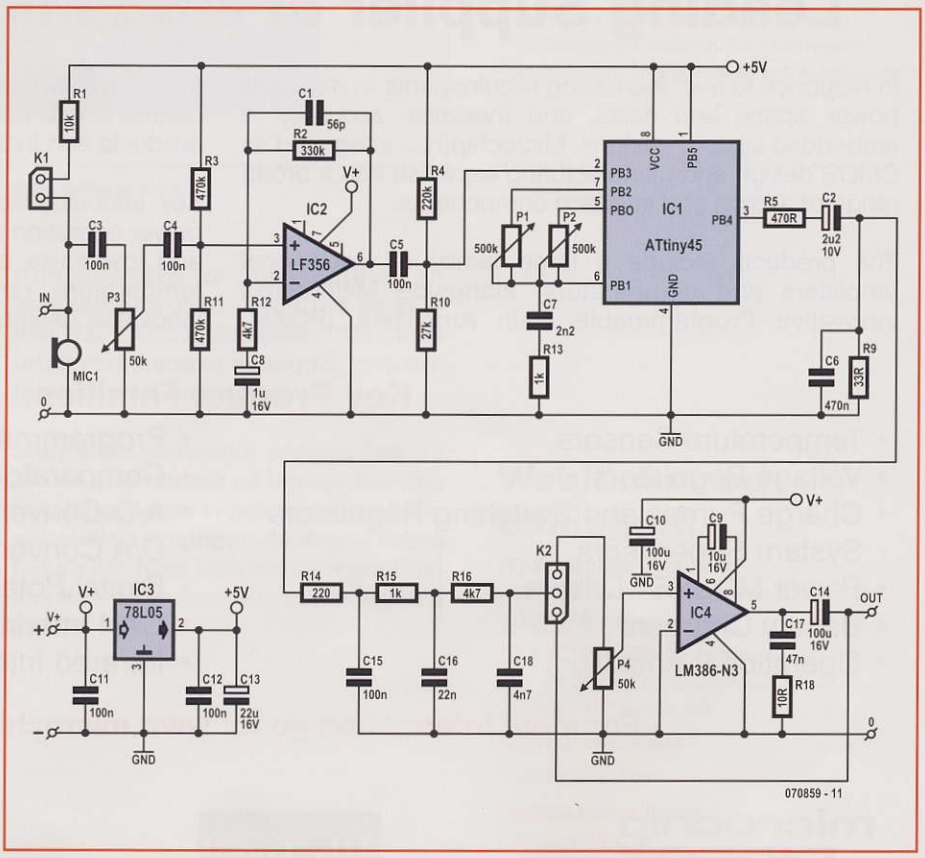
Gert Baars, g.baars13@chello.nl

Att ändra din röst så att du låter som en robot är en uppgift som är mycket lämplig för en mikrokontroller. Denna krets visar hur detta kan utföras med en enkel uppkoppling byggd runt en liten ATtiny45 mikrokontroller.

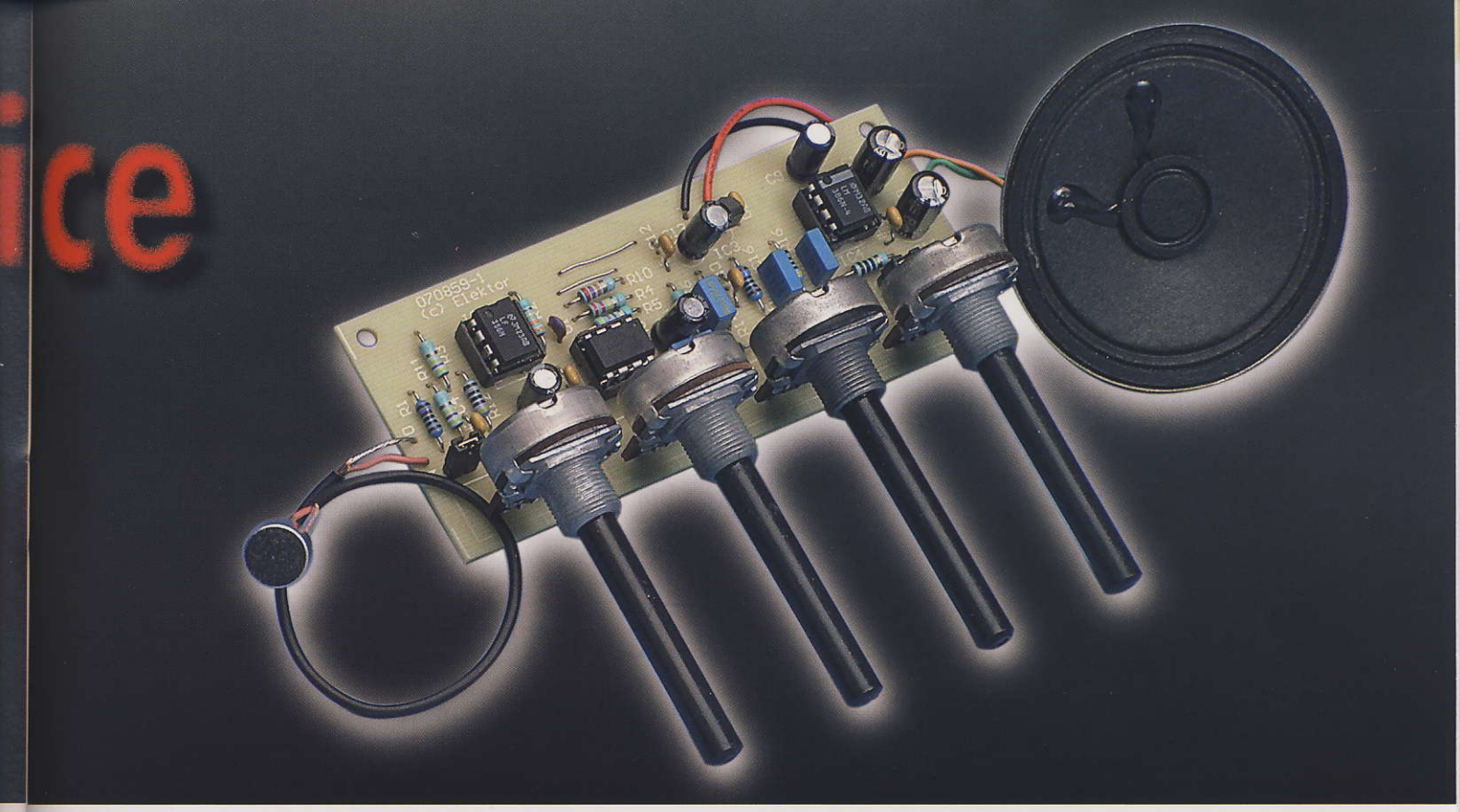
En röstförvrängare som emulerar en så kallad Cylonröst kan implementeras med hjälp av en liten mikrokontroller. De av er som har tittat på TV-serien Battlestar Galactica vet direkt vad det är vi pratar om. För de av er som inte är inne på sci-fi så är det en metallisk robotröst.
Kretsen kan användas som en liten rolig pryl, men designades ursprungligen för att visa hur en enkel digital krets kan användas för audiobehandling.

Hårdvara

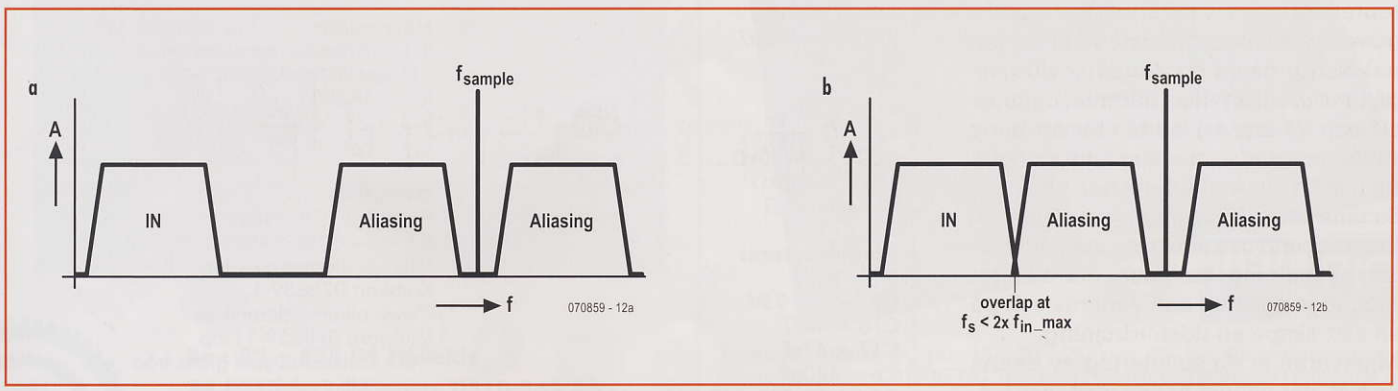
Hårdvaran (figur 1) består av en förförstärkare byggd runt operationsförstärkaren (opampen) IC2, som har en förstärkning på ca 70. Detta är tillräckligt för att förstärka mikrofonsignaler. R1 och K1 kan användas för att mata mikrofonen med en likspänning (DC), som krävs av elektretmikrofonen.
Förförstärkaren matar den förstärkta signalen till ADC-ingången på ATtiny 45 kontrollern som sedan behandlar den.
Eftersom klockan för kontrollern inte behöver vara alldeles exakt så har den inbyggda RC-oscillatorn använts som klockgenerator. Den har en utgångs-



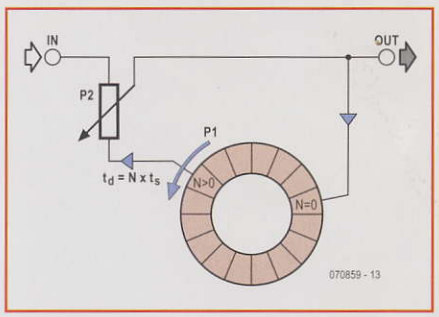
Figur 1. Röstförvrängaren består av en förförstärkare, en mikrokontroller och en effektförstärkare.



frekvens på ca 8 MHz, som dubblas till 16 MHz av en inbyggd PLL. Denna metod ser till att två ingångar förblir lediga på mikrokontrollern för anslutning av de två potentiometrarna. Dessa används för att ställa in den önskade 'Cylon' effekten.
Om två andra ADC-kanaler används för att läsa de två potentiometrarna skulle det betyda att audioingången skulle avbrytas av de interna MUX-arna medan dessa ingångar läses, vilket skulle påverka ljudkvalitén negativt. Det är därför att föredra att läsa potentiometrarna genom att tajma laddningen av en extern kondensator.
'Cylon' mjukvaran läser in audiosignalen med en samplingsfrekvens på ungefär 10 kHz. Enligt Nyquists samplingsteorem måste samplingsfrekvensen vara minst dubbelt så hög som den högsta frekvensen hos ingångssignalen. Av detta följer att den maximala ingångsfrekvensen är 5 kHz. Eftersom 'Cylon' kretsen är avsedd för talade signaler så är detta mer än tillräckligt.
Nyquists teorem är inte så svårt att förstå om vi först tänker på vad samplingen gör i frekvensdomänen. Under samplingsperioden kan man tänka sig ingångssignalen multiplicerad med '1' och under resten av perioden med '0'. Detta är i princip samma som 100% AM-modulering med en fyrkantvåg, vilket resulterar i ett AM-spektrum som visas i figurerna 2a och 2b. Från dessa kan du tydligt se att överlappning sker när samplingsfrekvensen är lägre än den dubbla högsta ingångsfrekvensen, vilket leder till distorsion. Detta visar hur viktigt det är med ett bra ingångsfilter före ADC-n. Detta så kallade anti-aliasingfilter spärrar komponenter som är högre i frekvens än halva samplingsfrekvensen. Då Cylonkretsen normalt arbetar med talsignaler, som för det mesta ligger under 3 kHz, finns det inget behov för ett komplicerat filter när man samplar vid 10 kHz.
Efter läsning av varje sampling utför mjukvaran vissa processer för att uppnå Cyloneffekten. Resultatet matas sedan till kontrollerns PWM-utgång. Denna genererar en fyrkantvåg med variabel pulsfaktor vid utgång PB4 hos ATtiny45. Efter integrering genom ett RC-filter (R14 till R16, C15, C16, C18) så återuppstår audiosignalen.
Denna signal matas sedan till audioförstärkaren IC4, som gör ljudet hörbart via en högtalare. Om du sätter jumpern på K2 i den andra positionen så går signalen förbi effetsteget och kan matas till linjeingången på en dator, till exempel.



Figur 2. Här ser man att överlappning sker när samplingsfrekvensen är längre än dubbla den högsta frekvensen i ingångssignalen.



Figur 3. Grafiskt representation av FIFOregistret, här konfigurerad som en ringbuffert.

Mjukvara

Mjukvaran måste utföra ett antal uppgifter. Den huvudsakliga är att läsa, behandla och skicka ut audiosignalen, men det finns också de två potentiometrarna som skall läsas. Eftersom ADC-n är fullt upptagen med att läsa audiosignalen så används en intern räknare och komparator för att läsa potentiometervärdena. Först av allt laddas C1 ur. Därefter startas en intern räknare och C1 laddas via potentiometern till ett viss värde då räknaren stoppas. Det resulterande räknarvärdet är då beroende på resistensen hos potentiometern.

För att förenkla det hela, Cyloneffekten uppnås genom att blanda den fördröjda versionen av ingångssignalen med den direkta ingångssignalen och fördröja den resulterande signalen igen osv.

Samma princip kan användas för att uppnå en ekoeffekt, men då måste fördröjningen av ingångssignalen vara minst 100 ms medan i Cylonversionen fördröjningen är mycket kortare. Den interna FIFO-buffert som används för att få fördröjningen är 200 byte lång så med en samplingsfrekvens på ungefär 10 kHz kan den maximala fördröjningen bara bli 200/10 kHz = 20 ms.

I figur 3 kan du se en grafisk representation av denna metod. Hjulet 'vrids' moturs med ett varv varje 20 ms och representerar en FIFO-buffert, konfigurerad som en ringbuffert. Ingångssignalen blandas med den fördröjda utgångssignalen (visas här med hjälp av en potentiometer) vilket skapar en ny utgångssignal som matas tillbaka igen till början av ringbufferten.

Samma effekt kan uppnås med hjälp av magnetband och separata inspelnings- och avspelningshuvuden, där de sistnämndas position kan varieras och på så sätt skapa en tidsfördröjning.

Mjukvaran är en simulering av denna metod, där minnet är lika med magnetbandet, en lagringsinstruktion är inspelningshuvudet och en laddningsinstruktion är avspelningshuvudet.

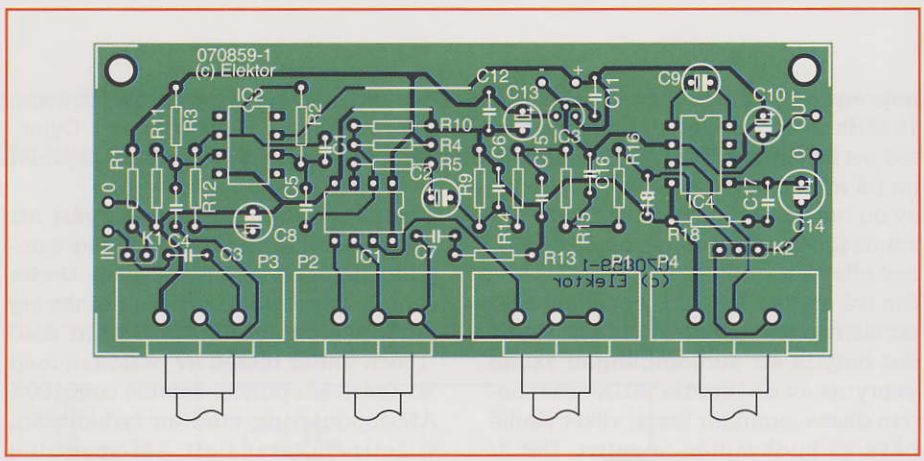
Mängden fördröjning ställs in med P1. Positionen hos P1 används av mjukvaran för att ställa in positionen för avtappningen i ringbufferten. P2 används för att ställa in styrkan på den fördröjda återkopplingssignalen. Med en större återkopplingssignal (feedback) blir fördröjningen mindre vilket ger en starkare effekt.

P1 används för att ställa in avtappningar på positioner mellan 1 och 200 i ringbufferten. Fördröjningen är därför variabel mellan 100 µs och 20 ms. Då den fördröjda signalen matas tillbaka i en loop kommer de dämpade resonanserna att vara märkbara om dämpningen är tillräckligt liten. Detta är den så kallade Cyloneffekten. Frekvensen hos dessa resonanser är 1/fördröjningstiden och kan därför ställas in mellan 50 Hz och 10 kHz. En inställning runt 250 låter ungefär likadant som de ur sprungliga Cylonrobotarna.

Själva bygget

Figur 4 visar layouten för röstförvrängaren. Att bestycka kortet är hur enkelt som helst då det finns gott om plats på kortet. Kontrollern kan ni köpa färdigprogrammerad från oss, ordernummer **070859-41**. Om du vill kan du programmera den själv (mjukvara 070859-11 men se till att använda korrekta inställningar för programmering av chipet (visas i den lilla textrutan).

Om du använder en elektretmikrofon måste du placera en jumper (kortslutningsbygel) över K1. Jumpern på K2 väljer typ av utgångssignal (pinnarna



Figur 4. Ett litet kretskort har tagits fram för röstförvrängaren, vilket gör bygget mycket enkelt.

KOMPONENTLISTA

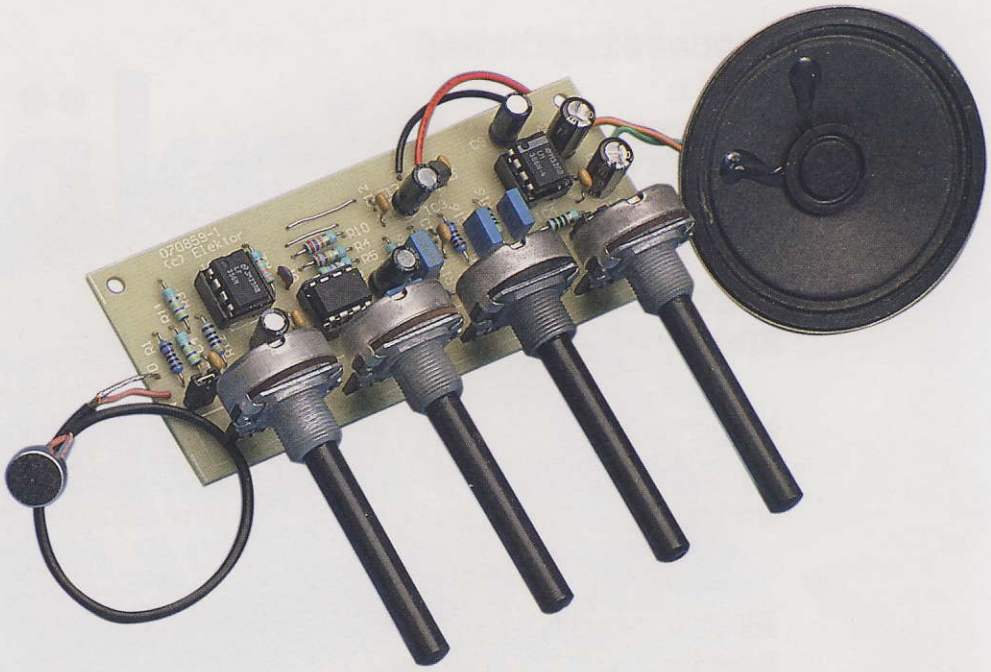
Motstånd
R18 = 10Ω
R9 = 33Ω
R14 = 220Ω
R5 = 470Ω
R13,R15 = 1kΩ
R12,R16 = 4kΩ7
R1 = 10kΩ
R10 = 27kΩ
R4 = 220kΩ
R2 = 330kΩ
R3, R11 = 470kΩ
P1,P2 = 500kΩ
P3,P4 = 50kΩ

Kondensatorer
C1 = 56pF
C2 = 2µF2 25V
C18 = 4nF7
C17 = 47nF
C6 = 470nF
C3,C4,C5,C11,C12,C15 = 100nF
C1 = 1µF 25V
C9 = 10µF 25V
C10,C14 = 100µF 25V
C13 = 22µF 25V
C7 = 2nF2
C16 = 22nF

Halvledare
IC1 = ATtiny45, programmerad, Elektor Shop # 070859-41
IC2 = LF356
IC3 = 78L05
IC4 = LM386-N3

Övrigt
K1 = 2-pol SIL stiftlist
K2 = 3-pol SIL stiftlist
MIC1 = elektretmikrofon
Kretskort 070859-1, www.alltomelektronik.se
Mjukvara 070859-11.zip och kretskortslayout gratis från www.alltomelektronik.se

2-3 använder utgångsförstärkaren, pinnarna 1-2 för linjeutgångssignalen). Kretsen behöver inte mycket ström, och denna dras i huvudsak av utgångsförstärkaren (mellan 25 mA och 150 mA). Du kan därför använda ett 9-V batteri så länge du inte använder kretsen under längre perioder. När du inte använder utsteget drar den ca 25 mA. Vi uppstart är det bäst att sätta effekten på minimum. När du hör en ren signal vet du att kretsen fungerar och nu kan du öka mängden effekt. För mycket effekt ger ingen audio alls (som du kan se av figur 3). Detta kan ställas in precist med P1 för att ge en bra Cyloneffekt tillsammans med ett tydligt tal.



(070859-I)

Inställningar för programmering av mikrokontrollern

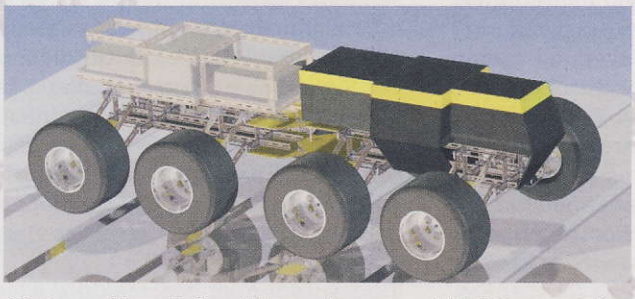
Fuses: - Brown-out detection disabled: BODLEVEL=111
- PLL clock: CKSEL=0001, SUT=11

P1 och P2 inställningar

- P1: Lägre värde → högre frekvens.
- P2: Högre värde → större effekt.

Annons

Bygg din modell med FAC



"Roboten Hugo" chassi är uppbyggt med FAC-Systemdelar

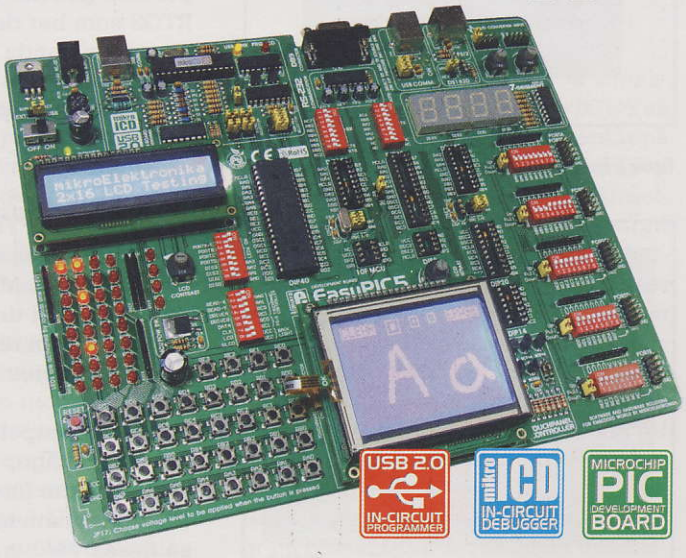
Byggsatsen

Universalbyggsatsen FAC är ett system av färdiga standarddelar, som tack vare en konsekvent genomförd moduluppbyggnad kan kombineras sinsemellan i nästan obegränsad omfattning. Med dessa standarddelar kan man bygga arbetande modeller och apparater, som åskådliggör idé och funktion utan ingående och tidsödande förklaringar. Byggsatsen används inom industri, mekanisk verkstadsindustri samt inom utbildningen av konstruktörer och ingenjörer från gymnasienivå och högre utbildningar. FAC-system har funnits på marknaden sedan 1952.

Box 82 • 533 04 Hällekis
Tel 0510-862 00 • Fax 0510-862 20
www.facsystem.se

mikroElektronika
UTVECKLINGSVERKTYG | KOMPILATOR | BÖCKER

EasyPIC5 with hardware In-Circuit Debugger



Systemet stöder 8-, 14-, 18-, 20-, 28- och 40- pin PIC mikrokontroller (kommer med PIC16F877A). mikroICD (In-circuit Debugger) möjliggör en mycket effektiv debugging och snabb utveckling av prototyper.

Besök vår hemsida http://www.mikroe.com för mer information

```
ProcessingThread
Begin
  Forever Do
    Wait for message
    Read values from message
    Process values
  EndDo
End

tADCValue adcVal;
…;
avixPipe_Read(pipeADC, (unsigned char*)adcVal, sizeof(adcVal

ADCThread
Begin
  Forever Do
    Wait for 1ms timer
    Read ADC values
    Put ADC values in message
    Send message to ProcessingThread
  EndDo
End
```

Realt

Hur fungera

Användningen av mikrokontroller inom elektroniska system ökar hela tiden. Vi har tidigare publicerat artiklar som beskrivit 16-bitars mikrokontroller från 24F, 24H och 33F familjerna från Microchip. Dessa moderna, kraftfulla mikrokontroller har många kringkretsar och en avancerad interruptstruktur. Att använda ett Real Time Operating System (RTOS) kan hjälpa till vid hanteringen av mjukvarustrukturen och timingegenskaperna hos ett system baserat på en så kraftfull mikrokontroller.

I denna artikel skall vi beskriva hur ett sådant RTOS fungerar. Det exempel som ges här baseras på AVIX, ett RTOS som har designats specifikt för de ovan nämnda mikrokontrollerfamiljerna. I slutet av denna artikel beskriver vi en demoapplikation för vårt Explorer-16 kort.

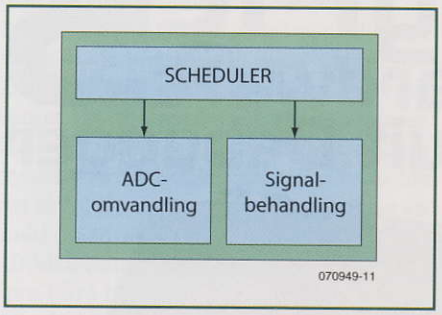


Figure 1. Typical software structure.

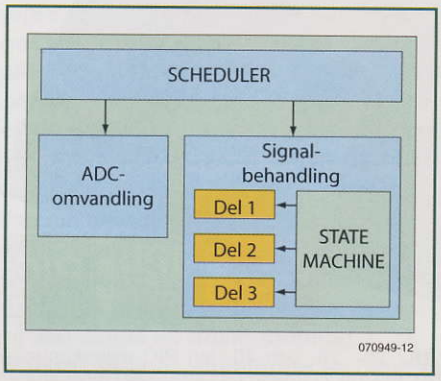


Figure 2. Complex software structure.

Typisk mjukvarustruktur

Mjukvara är vanligtvis designad att vara modulär. Med denna lösning är det viktigt att de olika modulerna får sin tur vid den rätta tidpunkten för att se till att systemet som helhet fungerar korrekt.

Som ett exempel kan vi ta det system som visas i figur 1. Här kan vi se att vi har en modul för att läsa analoga värden och en annan modul för att processa dessa värden. Dessa moduler körs under kontroll av en så kallad scheduler (schemaläggare). I detta fall är schedulern inget annat än en huvudfunktion som anropar de övriga funktionerna. I detta system skall de analoga värdena läsas en gång per millisekund och sedan skall signalbehandlingen avslutas innan början på nästa millisekund. Om det tar längre tid än det måste signalbehandligen 'delas upp' så att timingkraven kan tillfredsställas. Signalbehandlingsmodulen måste då själv hålla reda på vilken del som skall aktiveras varje gång den anropas och från ett utsidesperspektiv så utförs allting i rätt ordning i signalbehandlingsmodulen. Denna struktur visas i figur 2.

Tilldelningen av processorkapacitet till modulerna är nu uppdelad mellan Scheduler och State Machine. När ett stort antal moduler används blir denna mjukvarustruktur mer och mer komplex och otydlig. Den korrekta timingen beror på exekveringstiden hos de olika (del)modulerna. Denna exekveringstid kan i allmänhet endast bestämmas experimentellt. När förändringar görs i mjukvaran, när man byter till en annan mikrokontroller med annan hastighet eller när man använder annan kompilatoroptimering, så måste processen med att få den rätta timingen upprepas. Detta är mycket tidsödande och och väldigt öppet för fel.

Schemaläggningen av modulerna görs av programmeraren genom att dela

tidsberäkningar

ar ett realtidsoperativsystem (RTOS)?

val));

Leon van Snippenberg, AVIX-RT

Genom att använda ett Real Time Operating System (RTOS) kan strukturen på mjukvaran och timingegenskaperna hos mikrokontrollern förbättras markant. I denna artikel beskriver vi funktionen och de specifika egenskaperna hos ett sådant RTOS.

upp processorkapaciteten mellan modulerna genom att ändra koden. Detta är en statisk modell. Så fort koden körs i mikrokontrollern är denna timing fixerad och kan inte ändras dynamiskt. När schedulern har anropat en modul kan den bara fortsätta sitt jobb efter det att modulen är klar. Under hela den tid som modulen är aktiv så kan inte schedulern göra någonting. Detta är varför den ibland kallas för en kooperativ scheduler.

```
ADCThread
Begin
 Forever Do
  Wait for 1ms timer
  Read ADC values
  Put ADC values in message
  Send message to
ProcessingThread
 EndDo
End
```

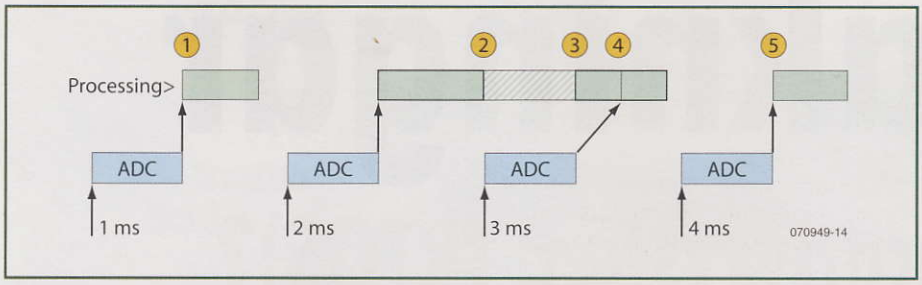
```
ProcessingThread
Begin
 Forever Do
  Wait for message
  Read values from message
  Process values
 EndDo
End
```

Figur 3. Grundsystem som använder ett RTOS.

Ett Real Time Operating System

Dessa problem kan man lösa genom att använda en så kallad 'pre-emptiv' scheduler, som också går under namnet RTOS. Med ett RTOS kan mjukvarustrukturen vara välorganiserad och timingen är förutsägbar. Detta är möjligt eftersom ett RTOS, till skillnad från en kooperativ scheduler, kan agera medan modulen körs. Strukturen hos en modul behöver således inte anpassas för att se till att RTOS får sin tur tillräckligt ofta. Om en händelse uppstår i systemet där RTOS måste agera, kan det avbryta den aktiva modulen för att kolla om en annan modul måste få gå före. Detta är en viktig fördel hos ett RTOS. Varje modul är skrivet som ett linjärt program utan att dela upp det i flera delar för scheduling. De moduler som körs under kontroll av ett RTOS kallas för trådar (threads). Varje tråd kodas som ett linjärt program och det indikeras inte någonstans i programkoden var RTOS kan avbryta det. När man använder ett RTOS tilldelas var och en av trådarna en prioritet och på basis av denna bestämmer RTOS vilken modul som skall aktiveras. Självklart erbjuder RTOS också möjlighet för trådarna att utväxla information med varandra (såsom meddelanden) och möjlighet att arbeta med tid (som timers). Med hjälp av ett RTOS kan systemet i figur 1 representeras i pseudokod såsom visas i figur 3. Här visas en viktig aspekt när det gäller operationen av ett RTOS. Båda trådarna väntar på en händelse. ADCThread väntar på en timer och ProcessingThread väntar på ett meddelande. Dessa väntfunktioner har implementerats av RTOS så att RTOS vet vilken tråd som väntar på vad. RTOS kommer att aktivera tråden i det ögonblick den önskade händelsen sker och den motsvarande tråd som har den högsta prioriteten. Väntfunktionerna är inte aktiva. Väntan betyder inte att köra genom en ändlös loop tills någonting sker. Väntan med hjälp av ett RTOS betyder att tråden stoppas totalt och inte använder några processorcykler alls. Under denna tid kommer RTOS att se till att en annan tråd är aktiv.

Hur fungerar det? Statusen hos ett program bestäms av det aktuella innehållet i processorregistren. Vid körning av koden kommer innehållet i registren att ändras hela tiden. När en händelse



Figur 4. Schedultrådar med ett RTOS.

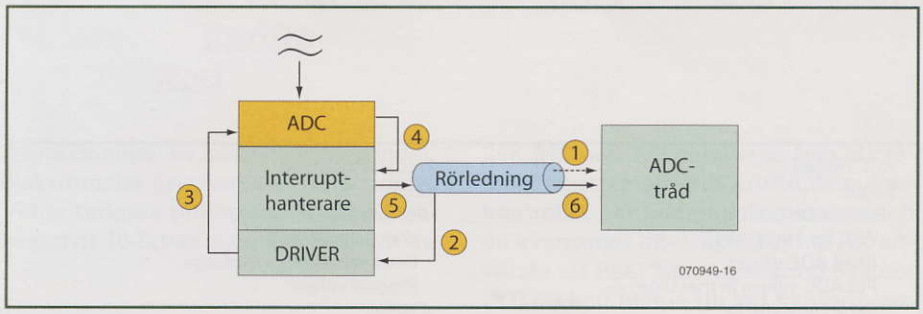
```
tADCValue adcVal;
...;
avixPipe_Read(pipeADC, (unsigned char*)adcVal, sizeof(adcVal));
```

Figur 5. ADC-omvandling med hjälp av ett rör (pipe).

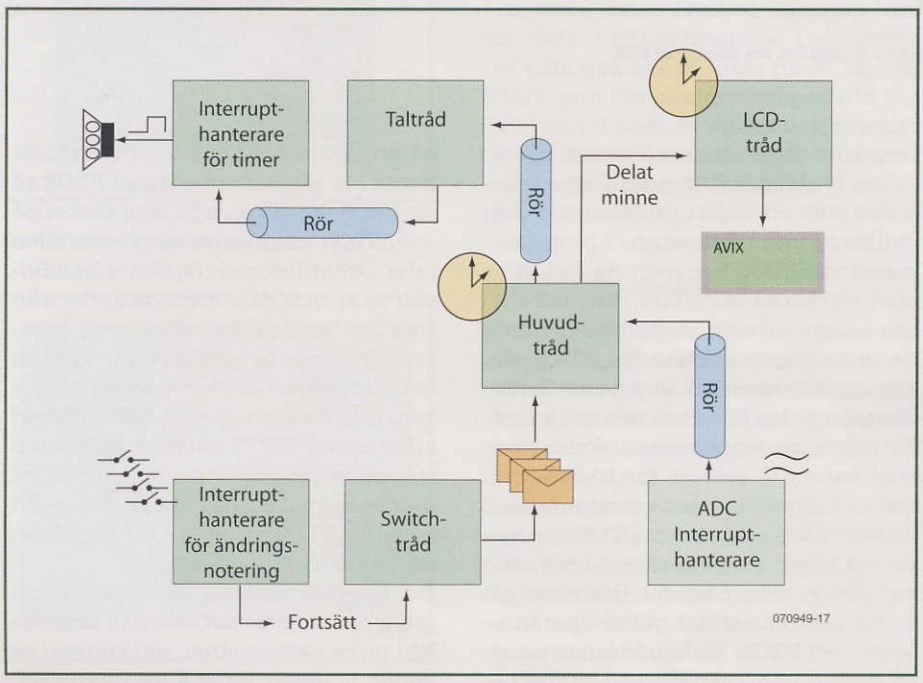
sker som resulterar i aktiveringen av en annan tråd kommer RTOS att skriva innehållet i alla registren till en minnessektion som är reserverad för just detta ändamål. Varje tråd har sin egen sektion. På detta sätt lagras tillståndet hos programmet som körs på ett säkert sätt.

Notera att ett av dessa register är programräknaren. Det är detta register som håller ordning på vad programkörningen håller på med. Eftersom detta register också sparas så sparas också trådens exekveringsprocess i avbrottsögonblicket.

RTOS kommer sedan att ladda processorregistren med innehållet i den tråd som skall aktiveras. Vilken tråd detta är bestäms av RTOS på basis av trådarnas prioritet. När som helst kan flera trådar aktiveras eftersom de önskade händelserna har inträffat för dessa trådar. RTOS väljer från denna grupp tråden med den högsta priori-teten. Eftersom ADCThread i systemet i figur 3 har en högre prioritet än ProcessingThread när timern löper ut kommer ADCThread att aktiveras oavsett vad ProcessingThread har för sig för ögonblicket. Detta löser därför det andra problemet med kooperativ scheduler. Utan att direkt kunna se detta i programkoden så har timingkraven uppfyllts. Eftersom ADCThread har en högre prioritet än ProcessingThread kommer avläsningen av de analoga värdena att ske i samma ögonblick som timern löper ut. När man behandlar data måste den totala processtiden för varje period hos ADCThread och ProcessingThread självklart **genomsnittligt** vara kortare än huvudperiodtiden på 1 ms. Om nu emellertid processen tar lite längre tid då och då så är inte detta något problem eftersom RTOS ser till att ADCThread aktiveras i rätt ögonblick. Detta visas med hjälp av figur 4. I botten av denna figur visas de ögonblick som timern för ADCThread löper ut. RTOS kommer att aktivera denna tråd och meddelandet som denna tråd sänder kommer följaktligen att aktivera ProcessingThread vid tid 1. I detta exempel tar nästa behandlingsperiod lite längre tid och vid tid 2 är ProcessingThread ännu inte klar. ADCThread aktiveras i alla fall eftersom den har högre prioritet. När ADCThread är klar kommer ProcessingThread att fortsätta vid tid 3. Meddelandet för 3 ms händelsen kommer därför att processas lite senare, nämligen vid tid 4. För att detta skall fungera är det nödvändigt att meddelandena placeras i kö, en funktion som erbjuds av i stort sett alla RTOS.



Figur 6: ADC interrupthanterare/trådintegrering.



Figur 7: AVIX-baserad talande termometer.

RTOS och interrupter (avbrott)

När man tittar på arkitekturen så kan man notera att samma sak skulle kunna uppnås utan RTOS men med interrupter i stället. Läsning och sändning av analoga värden kan också utföras av en interrupthanterare. Detta har

dock några nackdelar. 16-bitarskontrollerna från Microchip har en interruptprioritetsstruktur. Medan en interrupt behandlas spärras alla andra interrupter som har lägre prioritet. Om det inte finns något RTOS kommer interrupthanteraren att få utföra mer arbete och som ett resultat av detta kommer interrupter med lägre prioritet att få vänta längre vilket ökar risken för att en interrupt skall missas.
För det andra, analoga värden från interrupthanteraren måste göras tillgängliga för signalbehandlingsrutinen. Detta är en komplex procedur utan ett RTOS. Situationen att skriva till en minnesplats medan samma minnesplats läses av signalbehandlingsrutinen måste undvikas så att det inte finns någon risk att använda felaktiga värden. Detta kräver antingen en komplex mekanism eller – som ofta är fallet – så kopplas interruptrutinen bort av signalbehandlingsrutinen för att garantera att denna använder den rätta uppsättningen med värden. Detta ökar dock återigen risken för att interrupter skall missas.
Ett RTOS gör intert ett omfattande bruk av interrupter och tillhandahåller också säkra metoder att sända data säkert från interrupthanterarna till den tråd som är ansvarig för den fortsatta processen.
Interrupter är basen för de tidigare nämnda händelserna. Ett exempel på detta är timers. Praktiskt taget alla RTOS erbjuder timers så att en tråd kan vänta under en viss tid. Dessa RTOS-timers är mjukvarutimers. Dessa timers baseras på en av hårdvarutimerna i kontrollern. Denna hårdvarutimer genererar interrupter med fasta intervall. Vid varje interrupt kommer RTOS att uppdatera aktiva timers och när en löper ut aktiverar den processen som bestämmer om en annan tråd behöver köras. Så en hårdvaruinterrupt som ett resultat av en hårdvarutimer som löpt ut leder till en uppdatering av alla mjukvarutimers, vilket leder till den händelse vi nyligen diskuterat.
Interrupter används också vid styrning av kringutrustningen och dessa interrupter kan också leda till händelser som kan resulterar i att en tråd aktiveras. Ett RTOS erbjuder mekanismer för att göra detta, vilket tillåter programmering på en ganska hög nivå eftersom RTOS tar hand om komplexiteten av att hantera interrupter. Detta illustreras med hjälp av ett exempel som baseras på rörmekanismen, som också finns tillgänglig i AVIX.
Dessa rör (pipes) är FIFO-buffrar som låter trådar utväxla information med varandra eller med interrupthanterare. En tråd som vill läsa ett rör kommer att få vänta om den önskade mängden information inte finns tillgänglig (ännu). Vid skrivning kommer en tråd att få vänta om det nödvändiga tomma utrymmet inte finns tillgängligt (ännu). RTOS kommer att avaktivera tråden så att den inte använder processorkapacitet. Vi skall nu titta på hur ett rör kan användas för att läsa data från ADC-n. Detta sker från inuti en tråd och den kod som behövs för detta ses i figur 5. Tråden innehåller bara läsfunktionen och spärrar när önskade data inte finns tillgängliga. Vad händer i bakgrunden? Detta visas i figur 6.
Mjukvaruflödet är följande:
1. Tråden läser från röret. Tråden väntar eftersom rörledningen är tom.
2. Inifrån rörmekanismen anropas en DRIVER-funktion.
3. DRIVER aktiverar ADC-n. ADC-n körs autonomt och genererar en interrupt när den är klar.
4. Denna interrupt kommer att aktivera den interrupthanterare som motsvarar ADC-n. Denna hanterare kommer att läsa resultatet av omvandlingen och:
5. Skriva detta till röret. På detta sätt har de data som tråden önskar blivit tillgängliga och RTOS ser till att:
6. Data överförs till tråden som sedan kan behandla dessa.
Denna mekanism finns i det demoprogram som författaren skrev för denna artikel.

Ett system baserat på AVIX

Ett av demoprogrammen i Explorer-16 serien (i början på 2007) var en talande termometer. Denna använder ett stort antal kringkretsar som måste arbeta tillsammans. Strukturen hos detta program är dåligt organiserad och det är svårt att modifiera eller expandera det. Kompilering med den högsta nivån av optimering resulterar i att det inte fungerar mer eftersom timingen är helt annorlunda.
Detta program har skrivits om så att det kan arbeta under kontroll av AVIX för att demonstrera fördelarna med ett RTOS. Detta program kan laddas ner, inklusive en demoversion av AVIX, från www.avix-rt.com eller från Elektors websida www.alltomelektronik.se. Efter installation kan programmet byggas inifrån MPLAB och programmeras i en 24FJ128GA010-kontroller på utvecklingskortet för Explorer-16. Detta använder Microchip PICtail Plus ljudkortet (AC164125). Denna struktur kan ses i figur 7.
Demot inkluderar en omfattande beskrivning av de olika delar som programmet består av

(070949-I)

Webläkar:

www.avix-rt.com

Vi har några exemplar kvar av Explorer-16 om någon läsare är intresserad. Kolla på www.alltomelektronik.se

Författaren



Leon van Snippenberg skrev ett RTOS själv som kallas AVIX. Detta RTOS har speciellt utvecklats för PIC24 och dsPIC-serierna av mikrokontroller från Microchip. Hans företag AVIX-RT specialiserar sig på att tillhandahålla produkter och tjänster för utvecklingen av inbäddade system.



Från C

Med hjälp av

Paul Goossens

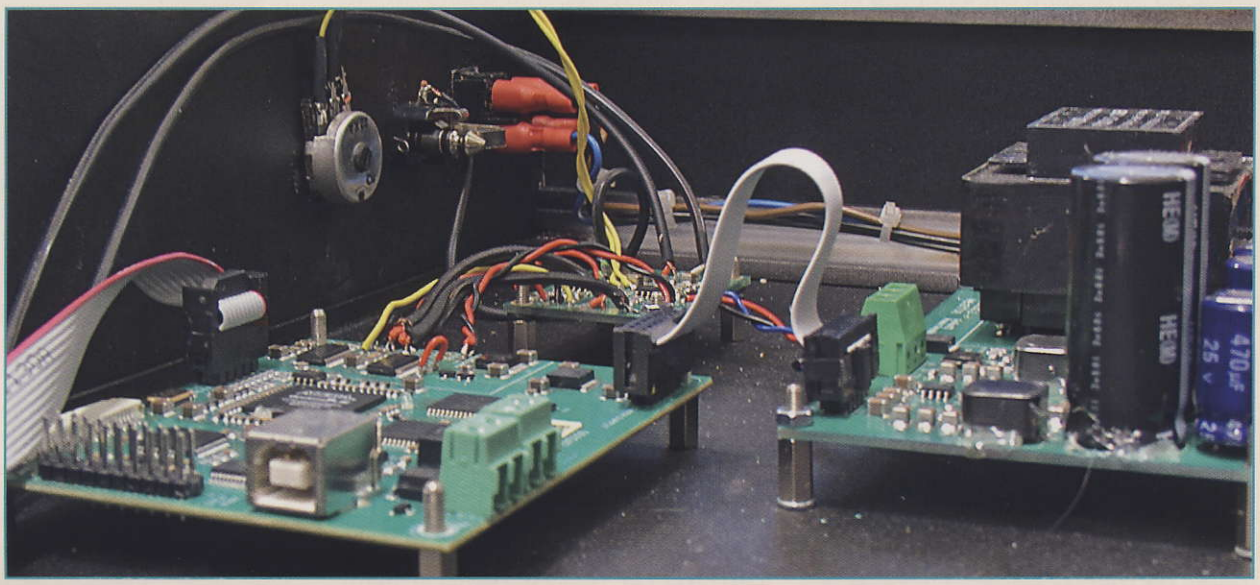
FPGA-teknologin har utvecklats snabbt på bara kort tid och FPGA kan nu användas i ett stort antal applikationer. Trots detta så drar nya apparater inte fördel av kapaciteten hos dessa IC-kretsar. Huvudskälet till detta är att det finns för få FPGA-experter. De senaste utvecklingarna inom EDA-området kan till stor del eliminera detta problem.

De flesta av apparaterna i vår vardagsomgivning har idag en mikrokontroller på insidan. Även de enklaste produkter omfattar ofta någon form av mikrokontroller. Detta har inte alltid varit fallet. Under mikrokontrollernas tidiga dagar var deras användning begränsad till en utvald grupp tekniker och de flesta elektroniska apparater implementerades fullt ut med analog elektronik.

Mystiska FPGA

För inte så länge sedan användes mikroprocessorer enbart i datorer. De ingenjörer som arbetade med dessa IC kallades för 'experter' och ockuperade ett speciellt område som var separerat från vardagselektroniken. Medföljande firmware var primärt programmerad i assemblerspråket. Designers som lutade mer åt det analoga hållet (vilket betydde den absoluta majoriteten av konstruktörerna) betraktade mikroprocessorer och mikrokontrollers som mystiska komponenter.

Samma situation existerar idag när det gäller FPGA. Även om de flesta konstruktörer har hört talas om FPGA och faktiskt är intresserade av dessa lovande IC-kretsar så fortsätter de att använda sina välbekanta mikrokontroller i sina designer. I många fall är också en mikrokontroller det mest logiska valet, men i fler och fler fall kunde en FPGA vara en mycket god ersättare för en mikrokontroller.



till hårdvara

FPGA och kompilatorer

Framsteg

Jämfört med mikrokontroller har FPGA ett rykte om sig att vara dyra och energihungriga. FPGA-tillverkarna har tagit åt sig detta genom att minska energiförbrukningen i sina IC-kretsar och utveckla billiga FPGA. En hel del har redan uppnåtts inom detta område och specifikationerna kommer bara att bli bättre i framtiden.

Om du kollar websidorna hos olika FPGA-tillverkare kommer du att se att varje tillverkare erbjuder billiga och energisnåla FPGA-familjer. Resultatet har blivit att valet mellan en mikrokontroller och en FPGA har blivit svårare om det enda kriteriet är låg effektförbrukning och lågt pris.

Ett annan stor invändning mot att använda FPGA är att det finns relativt få konstruktörer som klarar av att utveckla FPGA-baserade designer.

Det mest vanliga programmeringsspråket för att programmera en mikrokontroller är C, som arbetar helt och hållet i enlighet med den sekventiella principen. I motsats till detta är innehållet i en FPGA primärt designat med hjälp av VHDL eller Verilog. Dessa två språk hör inte hemma i den sekventiell-logiska världen. Att växla från C till ett av dessa språk kräver ett helt annorlunda tänkande.

Detta skapar en ganska stor utmaning för konstruktörerna.

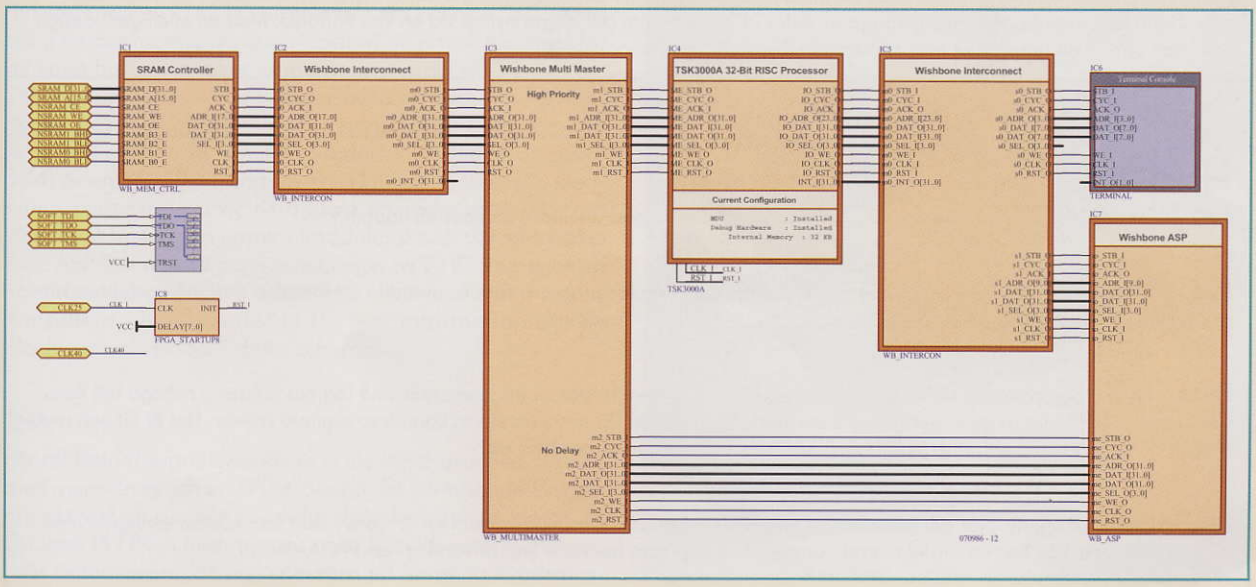
Om du vill bli någorlunda hemma på ett språk som VHDL så måste du definitivt investera en hel del tid för att få kunskaper och erfarenhet. Detta betyder att dina första få FPGA-baserade designer kommer att ta mycket längre tid än en likadan design baserad på en mikrokontroller. För många företag är detta ett tillräckligt skäl för att använda en mikrokontroller som en standardlösning.

Två världar

Även om beslutet tas att använda en FPGA så åtföljs den i många fall också av en mikrokontroller. Detta kan vara en extern mikrokontroller eller en mikrokontroller implementerad i FPGA.

Detta gör det möjligt att kombinera det bästa från två världar. Med denna lösning kan en stor del av designen fortfarande utvecklas på traditionellt sätt. FPGA används för en specifik del av designen medan resten av jobbet hanteras med firmware på det vanliga sättet.

Färdiggjorda block av 'intellektuell egendom' (IP) sätts ihop för att forma en arbetsbas. Ibland adderas en bit ny design (utvecklad med VHDL, Verilog etc.) till mixen. I många EDA-miljöer kan allting plockas ihop med hjälp av ett grafiskt interface utan att man behöver skriva en enda rad av VHDL



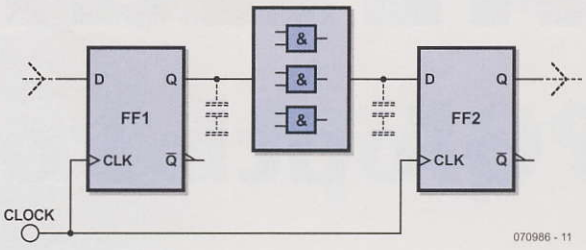
Figur 1.
Den interna strukturen hos FPGA-designen.

Statisk kontra dynamisk strömförbrukning

Den ström som förbrukas av en IC kan delas upp i två delar. En del är den ström som alltid flyter så länge som det finns en matningsspänning. Denna ström består av olika läckströmmar såväl som strömmar som är nödvändiga för olika uppgifter, som att behålla innehållet i ett statiskt minne. Denna ström är den så kallade 'statiska strömmen'.

Den andra delen av strömmen kallas för den 'dynamiska strömmen'. Denna ström resulterar från nivåväxlingen hos de olika signalerna inuti IC-kretsen.

Som du vet så består en digital IC (och således också en FPGA) av en samling enkla digitala kretsar. Var och en av dessa kretsar har en utgång. I en digital krets kan varje utgång ha ett av två tillstånd som kallas '1' och '0' eller 'hög' och 'låg'. Utgångarna är anslutna via ledare i IC-n till en eller flera ingångar hos andra logiska kretsar i IC-n.



Från fysiken vet vi att alla ledare har en viss kapacitans i förhållande till andra ledare i dess närhet. Dessa oavsiktliga parasitiska kapacitanser är ofta källor till problem för både elektronikingenjörer och hobbyister och de kan också orsaka problem även i IC-kretsar. Varje gång en utgång växlar tillstånd så ändras spänningsnivån på utgångssignalerna.

Då utgången har en viss kapacitans i förhållande till jord och andra ledare så måste det flyta en ström för att ladda eller urladda denna kapacitans och således ändra spänningsnivån.

Om en utgång stiger från 0 V till +5 V måste utgångskretsen lämna en ström som laddar den parasitiska kapacitansen tills den har nått nivån på 5 V. Om denna utgång växlar från +5 V till 0 V måste laddningen på kapacitansen laddas ur till jord för att gå tillbaka till spänningsnivån på 0 V.

Den mängd laddning som är nödvändig för att ladda kapacitansen beror på två faktorer. Den första faktorn är skillnaden mellan spänningsnivåerna '0' och '1'. Om matningsspänningen minskas så minskas också mängden laddning som behövs. Den andra faktorn är storleken på den parasitiska kapacitansen. Denna kapacitans kan minskas genom att modifiera den interna strukturen i IC-kretsen och välja rätt material. Även här innebär en minskning av kapacitansen en minskning av den mängd laddning som är nödvändig för att växla spänningsnivån.

Synkron logik

I en synkron design reagerar IC-n på hastigheten hos en klocksignal. Utgångarna hos vipporna kan växla när det kommer en klockpuls. Om det följer kombinationskretsar efter vipporna så kommer dessa att ha olika ingångar. Som ett resultat kan deras utgångsnivåer också ändras.

Efter en liten tidsfördröjning har alla utgångar hos vipporna och kombinationskretsarna nått sina sluttillstånd. Dessa utgångar kan bara ändras när nästa klockpuls kommer.

Från Hz till ström

Allt detta betyder att den dynamiska strömförbrukningen kan minskas genom att minska klockfrekvensen. Det betyder att det blir färre klockpulser per sekund och detta kan direkt översättas till en minskning av den dynamiska strömmen.

Den dynamiska strömmen kan minskas ytterligare genom att se till att så få utgångar som möjligt byter tillstånd på varje klockpuls. Detta kan uppnås genom att stänga av delar i IC-n som inte gör något nyttigt vid en viss tidpunkt. Med en intelligent design kan man uppnå ett betydande energisparande med hjälp av denna teknik.

Ordlista

ASP – Application Specific Processor. Digital krets som är designad att utföra en speciell uppgift.

EDA – Electronic Design Automation. En gemensam beteckning för program som är avsedda att förenkla designen och utvecklingen av elektroniska apparater.

FPGA – Field Programmable Gate Array. En integrerad krets som består av ett stort antal små logiska kretsar (i många fall flera tiotusental). Med hjälp av en programmerare kan de länkas ihop för att bilda stora, komplexa digitala kretsar. Det är till och med möjligt att bygga kompletta mikrokontroller, grafikkort och liknande på detta sätt.

IP – Intellectual Property. När det gäller FPGA så betyder detta mjukvarukod (såsom en 'IP core') som kan köpas, vanligtvis med en licens. Denna kod har en fördefinierad komplex funktion som bekvämt kan användas i en FPGA-design.

eller Verilogkod. Dessa metoder betyder att konstruktörerna inte behöver investera alltför mycket tid på VHDL-design.

Från C till H

Den senaste utvecklingen, som vi för bekvämlighets skull kan kalla 'från C till H' alltså från C till hårdvara, är nästa steg i VHDL-fri design av FPGA-kretsar.
Flera EDA-utvecklare har utvecklat metoder att designa digitala kretsar utan att använda VHDL. Den mest anmärkningsvärda metoden handlar om direkt omvandling av standard ANCI C-källkod till en digital design. Det finns redan kompilatorer tillgängliga som som kan omvandla ett specifikt subset av C-språket till en digital krets. De senaste kompilatorerna kan hantera hela omfattningen av standard C (ANSI C).
Två leveratörer har nu utvecklat kompilatorer som kan göra detta. FPGA-tillverkaren Altera med sin C2H-kompilator och EDA-utvecklaren Altium med sin CHC-kompilator. Det förväntas att andra leverantörer också snabbt kommer att erbjuda sina egna C till H kompilatorer.

FPGA för massorna

Styrkan hos dessa kompilatorer är att rutiner som utvecklats på det gamla välkända sättet nu kan dra fördelar av kraften hos FPGA.
Mikrokontrollers är genom sin natur utvecklade att vara så universella som möjligt. Detta gör att de kan utföra en mängd olika uppgifter. Det är också ett av skälen till den enorma succé som mikrokontroller har haft, men detta är också en av dess svagheter. Mikrokontroller kan göra allting, men de är inte specialister på någonting.
FPGA, liksom mikrokontroller, kan också göra allting – men på en annan nivå. Du kan bygga din egen digitala krets (inkludera en mikrokontroller om du vill) i en FPGA, men du kan också göra dem till en specialiserad bit hårdvara som excellerar i en specifik uppgift. Dessa hårdvarubitar kan utvecklas i C genom att använda dessa 'C till H' kompilatorer. Det betyder att även programmerare som inte har någon erfarenhet av VHDL kan dra fördel av kraften i en FPGA.

I praktiken

Hur fungerar då detta i praktiken? I vårt labb kan vi använda Altium Designer, som från version 6.8 och framåt är försedd med en C till H kompilator som kallas 'CHC'. Med hjälp av denna mjukvara använde vi ett enkelt exempel för att kolla hur olika saker fungerar rent praktiskt.
Vårt exempel (figur 1) använder bara standard IP-komponenter från Altium. Med hjälp av dessa komponenter kan du skapa ett inbäddat system på nästan nolltid. Processorn (IC4) är en TSK3000, som är en 32-bitars processor. Denna processor ansluts via flera block till andra moduler, som ett externt minne, en terminalmodulator och vår stjärnspelare: ASP. Det resultat som produceras av CHC-kompilatorn hamnar i detta block.
Konsultera Altiums websida för mer information om de block som användes i detta schema.

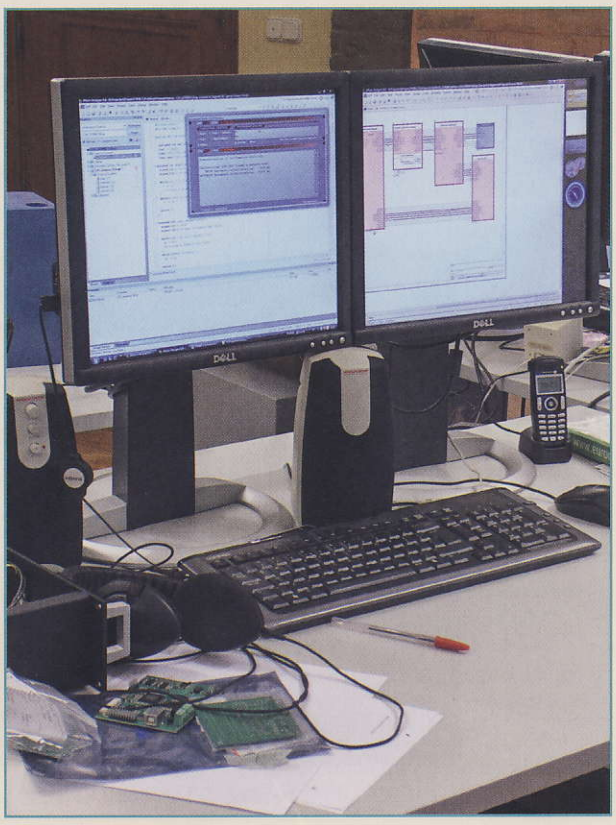
Test

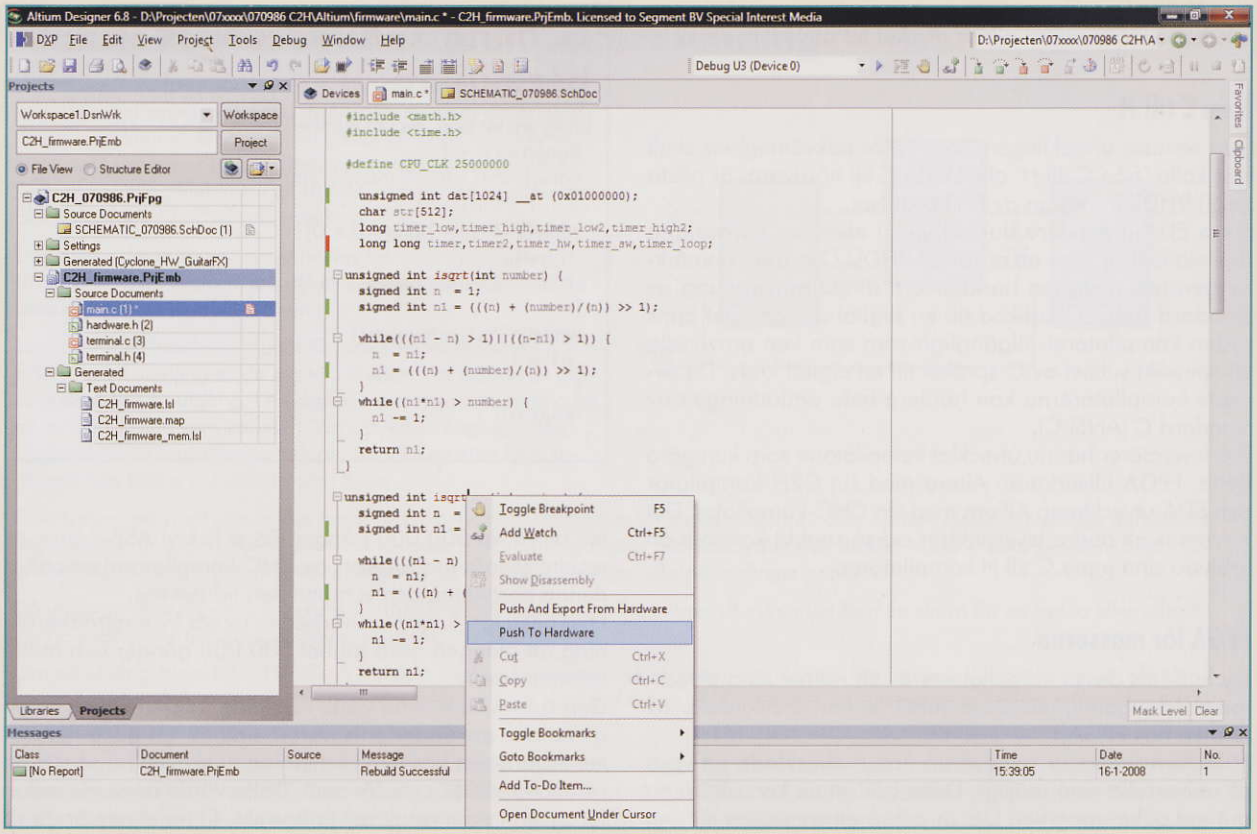
För att testa designen behövde vi lite hårdvara tillsammans med mjukvaran och en FPGA-design. Ett kommande Elektorprojekt var en perfekt kandidat. Hårdvaran inkluderar en Cyclone III FPGA tillsammans med 256 kB externt minne.
I vår testfirmware får vi processorn att beräkna kvadratroten ur ett tal 500 000 gånger. Först fick vi ASP-n (som är resultatet som producerats av CHC-kompilatorn) att utföra denna beräkning och vi mätte den tid det tog.
Efter detta lät vi mjukvaruversionen av vår kvadratrotsberäkning att återigen göra jobbet 500 000 gånger och mätte resultatet igen.
Den rutin vi använde visas i Listning 1. Den första delen av rutinen använder en smart metod för att gissa resultatet. Algoritmen hos denna gissning är skriven på så sätt att resultatet alltid är lite för stort. Detta värde används sedan för att kolla om resultatet är korrekt. Om det inte är de så dekrementeras (minskas) det förutsagda resultatet med 1. Denna procedur upprepas tills det rätt resultat uppnås.
En trevlig sak med denna algoritm är att den använder två loopar (slingor): Skiftinstruktioner och multiplicerainstruktioner. Detta betyder att den innehåller en trevlig blandning av vanligt använda operationer.

Listning 1: Kvadratrotsberäkning

```
unsigned int isqrt_sw (int number) {
 signed int n  = 1;
 signed int n1 = (((n) + (number)/(n)) >> 1);

 while(((n1 - n) > 1)||((n-n1) > 1)) {
  n  = n1;
  n1 = (((n) + (number)/(n)) >> 1);
 }
 while((n1*n1) > number) {
  n1 -= 1;
 }
 return n1;
}
```





Figur 2.
Det kunde knappast vara enklare!

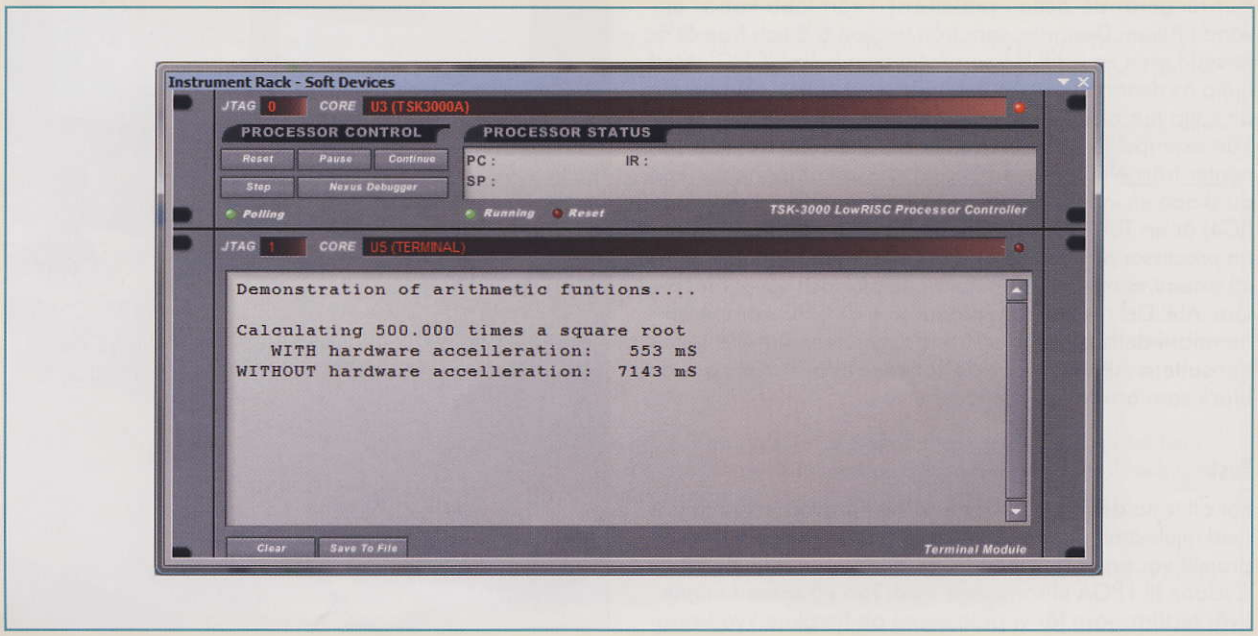
Exportera till hårdvara

Vi använde denna rution (kallad ***isqrt_sw***) för att mäta tiden som behövs för att utföra denna beräkning. För att testa CHC-kompilatorn gjorde vi en identisk kopia av denna rutin. Vi kallade den ***isqrt***.

Nästa steg var att använda utvecklingsmiljön för att få denna rutin omvandlad till hårdvara av CHC-kompilatorn. Detta är en mycket enkel process. Först placerar du pekaren (kursorn) på rutinens namn. Om du högerklickar på musen får du upp en liten meny. Välj 'Push and export hardware' i denna meny (se figur 2). Detta får CHC-kompilatorn att implementera den valda rutinen i hårdvaran. Dessutom får denna aktion varje anrop till firmware i denna rutin att ersättas med ett anrop till ASP. Allt detta med bara ett enda musklick.

Resultat

Efter kompilering och programmering av FPGA-n visades utgången på en PC med hjälp av en terminalemulator, figur 5. Här kan du se att hårdvaruversionen tar lite mer än en halv sekund för att utföra kvadratrotskalkyleringen 500 000 gånger. Mjukvaruversionen tar nästan 13 gånger



Figur 3.
Testresultaten.

längre tid på sig i detta fall. Det trevligaste av allt i detta exempel är att vi (avsiktligt) inte skrev en enda rad av VHDL. Denna teknik låter programmerare dra fördel av styrkan hos en FPGA utan att behöva lämna den familjära C-miljön.

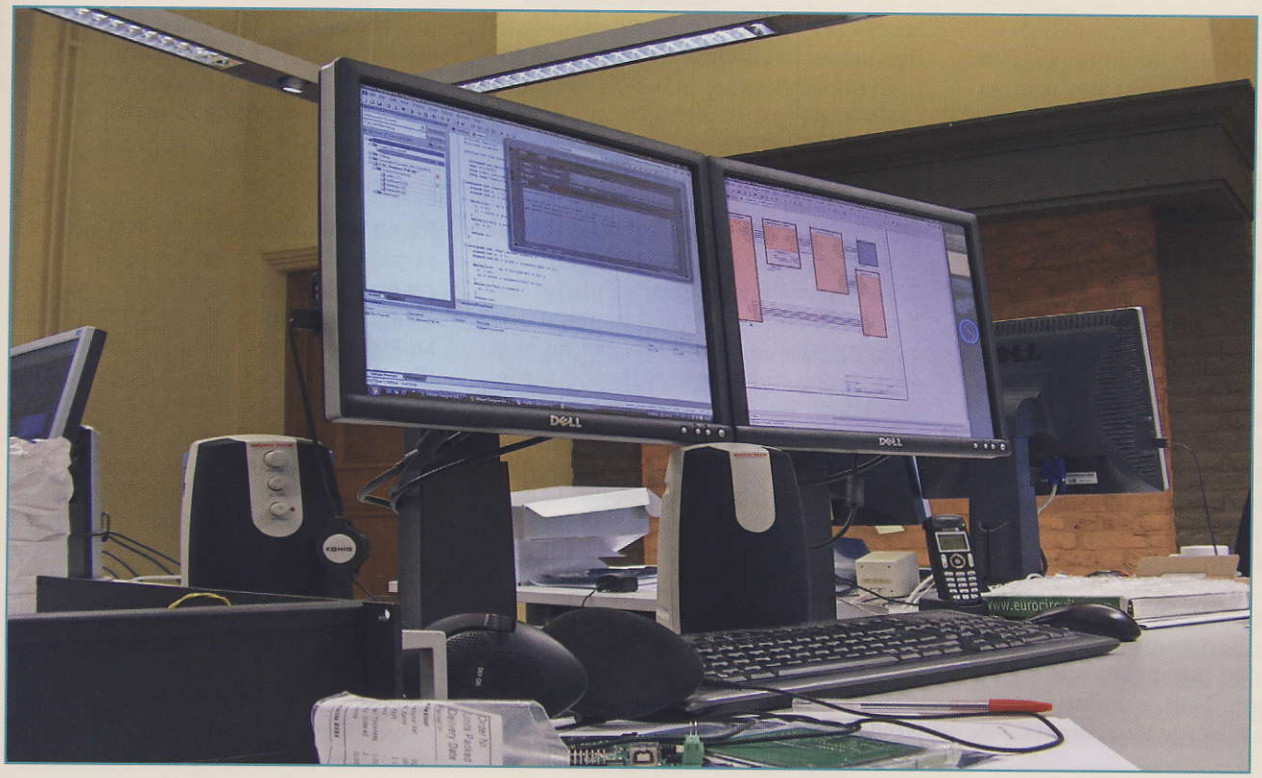
Applikationer

ASP-n kan man se som en hjälpprocessor av gör-det-själv-typ. ASP-n kan anpassas till individuella applikationer såsom en videoaccelerator. I detta fall kan hastigheten hos rutinerna för att rita linjer, cirklar och ytor ökas genom att implementera dem i en ASP.
Det är också möjligt att ha rutiner i ASP som körs parallellt med rutinerna i mjukvaran. Detta kräver dock modifiering av mjukvaran. Fördelen med detta är att det ytterligare ökar processorkraften.
Det tidigare nämnda exemplet med en videoaccelerator är en lämplig kandidat för denna lösning. Processorn behöver inte vänta tills videoacceleratorn har ritat linjen, cirkeln eller något annat objekt utan kan spendera sin tid på andra uppgifter.

FPGA tillgängliga för programmerare som inte har erfarenhet av FPGA, VHDL och annat relaterat till detta. Den minskade utvecklingstiden och möjligheten att återanvända befintliga, pålitliga rutiner i FPGA representerar också ett stort steg framåt när det gäller att acceptera FPGA-teknologin i ingenjörsvärlden.

I kombination med en tekniska utvecklingen vid FPGA-tillverkning betyder detta att FPGA blir ett ännu starkare konkurrerande alternativ till konventionella inbäddade mikrokontroller.

(070986-1)



Låg effekt

En inte fullt så uppenbar fördel är att effektförbrukningen kan minskas om man använder en ASP. Klockfrekvensen kan minskas genom att använda en ASP för att öka processkraften. Den hastighetsökning som man erhåller av att använda en ASP kan man helt eller delvis byta bort mot en lägre klockhastighet.
Den dynamiska strömförbrukningen i FPGA minskar i direkt proportion till klockhastigheten. Resultatet blir således en lägre effektförbrukning. Detta är primärt viktigt i bärbara apparater och i miljövänliga produkter.

Slutligen

Den nya utvecklingen av C-till-H kompilatorer har gjort

C-till-H för PC?

FPGA används redan i några superdatorer för att ge processorkretsarna en hjälpande hand när det gäller att utföra krävande beräkningsuppgifter.

Det är definitivt tänkbart att FPGA i framtiden också hittar vägen till moderkortet i en vanlig standard-PC. För att detta skall bli möjligt är det nödvändigt att skapa en standard som specificerar hur processorn skall kommunicera med FPGA. Ännu viktigare är att ha en standardteknologi för att omvandla mjukvarurutinen till hårdvaruversioner som är lämpliga att implementera i FPGA. De nya C-till-H kompilatorerna är ett steg i denna riktning.

Och vem vet vilka hastighetsförbättringar denna teknik kan erbjuda framtida PC-datorer.

DIY LED-projektor

DLP-projektor med effekt-LED

Jeroen Domburg

Även om en kväll på bio kan vara en trevlig upplevelse så finns det även några nackdelar. Du kan inte pausa filmen för att gå på toaletten och det är inte alltid som sätena är så bekväma. Om du har riktig otur så får du dela upplevelsen med en hop stojiga ungar som inte kan sitta still. En hemmaprojektor är ett bra alternativ. En stor trevlig display, du kan pausa när du vill och du kan också välja ditt eget sällskap. Det är bara synd att lamporna slits ut så fort...

Det finns definitivt en hel del fördelar med att titta på film hemma. Bjud in några goda vänner, ta ett par drinkar och du kommer att få en trevlig stund. Huvudskillnaden är naturligtvis att medan bion erbjuder en jättestor skärm så har man en (relativt sett) liten TV hemma. Och även om stora LCD och plasmaskärmar blir allt billigare så ligger dock de största utanför de flestas budget.
En apparat som kan producera stora bilder hemma är projektorn. Rikta den mot en vit vägg eller projektorduk, anslut den till en laptop eller DVD-spelare och du får en högkvalitets display som är väl över en meter i storlek.

Priser

Projektorer är inte speciellt dyra längre och en begagnad kan du få för några tusenlappar och inte heller nya kostar längre skjortan. Men, och detta glömmar man ofta bort, de kan vara ganska kostsamma att underhålla. Den inbyggda lampan har en livslängd på bara några få tusen timmar och kan kosta upp till 6-7000:- att ersätta. Detta gör det till en mycket dyrbar fritid om du ofta tittar på film. Och det får dig också att tänka två gånger innan du startar projektorn för att kolla nyheterna eller spela ett spel på den stora skärmen.
En ny utveckling är att använda lysdioder (LED) i stället för en lampa i några nya projektorer. Dessa varar mycket längre och har den ytterligare fördelen att de inte blir så varma och kan ha tystare fläktar för kylningen.
Nackdelen med dessa projektorer är att det inte tillverkas så många av dem ännu och de är ganska dyra. Du hittar idag ingen under 6000:- såvitt vi vet.
Under de senaste åren har många förbättringar skett inom LED-teknologin. Detta har också upptäckts av många hobbyelektroniker. Vissa elektronikbutiker och webbutiker lagerför LED som ställer glödlampor helt i skuggan. Detta ger oss alla fördelarna: En projektor som är tyst, inte kostar alltför mycket och där man inte behöver ersätta lampan vid regelbundna intervall.

Lite teori

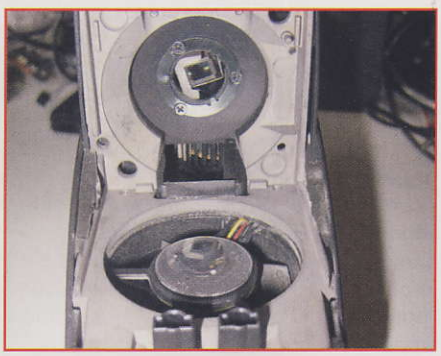
Innan vi börjar med projektet behöver vi gå igenom lite teori först. Hur fungerar en genomsnittlig DLP (Digital Light Processor) konsumentprojektor? En DLP-projektor är byggd runt en DMD (Digital Mirror Device) som är ett chip med tusentals mikroskopiskt små elektroniskt aktiverade speglar på sin yta. Varje spegel motsvarar en pixel. Genom att vinkla dessa speglar kan en infallande ljusstråle skickas vidare eller spärras. Detta betyder att pixlarna bara kan vara 'till' eller 'från'. Styrelektroniken kan generera mellanliggande nivåer genom att vrida speglarna tiotusentals gånger per sekund.
Men du behöver mer än bara en DMD för att göra en projektor. Det behövs



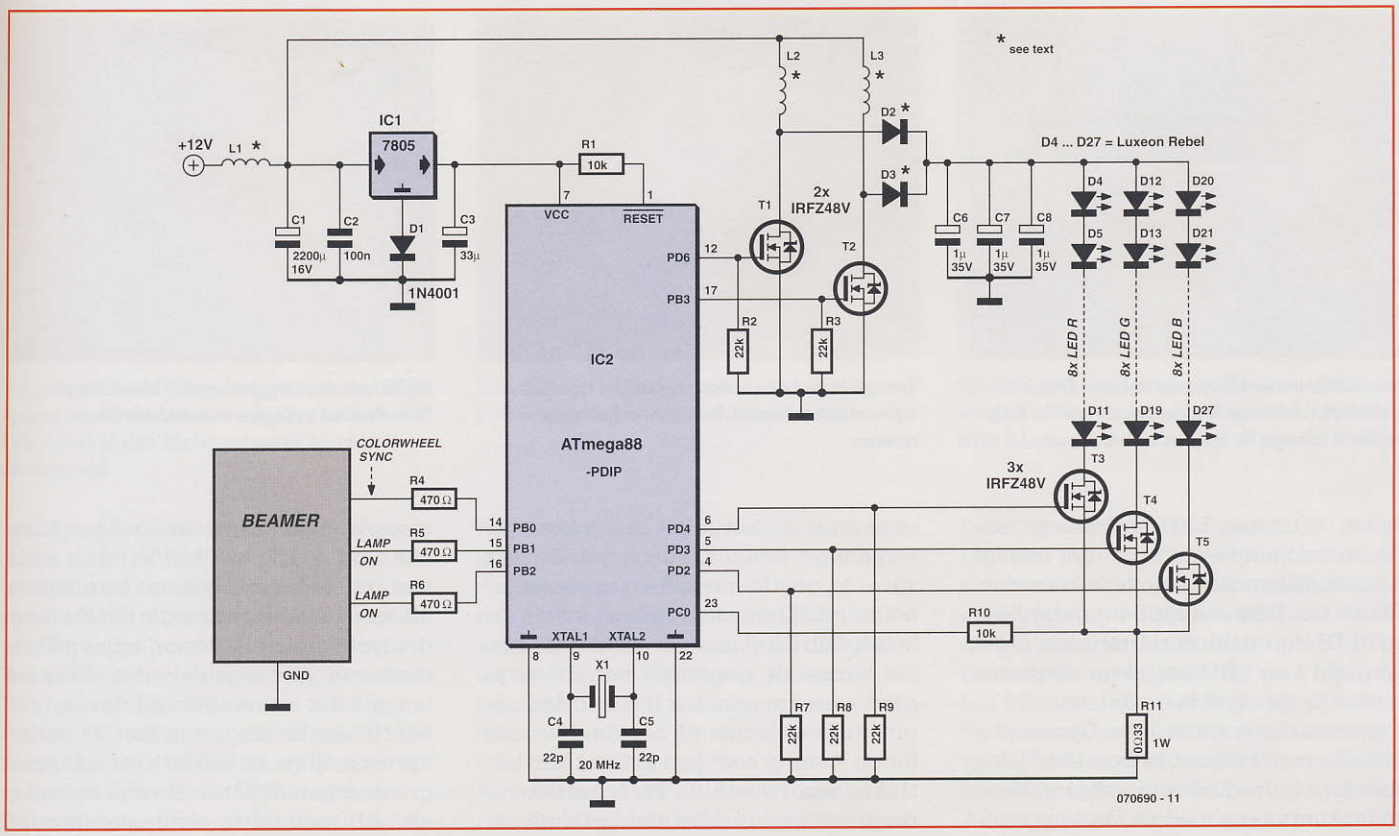
Detta är den projektor vi skall modifiera. Det är en ganska kompakt modell, vilket tyvärr betyder att det inte finns mycket utrymme kvar för vår egen krets.



Den borttagna, allt för dyrbara lampan, som vi skall ersätta med en billigare lösning.



Här sitter lampan. I botten kan du se en av de tre fläktar som kyler lampan. Den stora runda öppningen går till färghjulet och ljustunneln.



Figur 1. Kretsen är byggd runt en mikrokontroller, som så ofta är fallet nuförtiden. Denna kontroller tar också hand om boostomvandlingen.

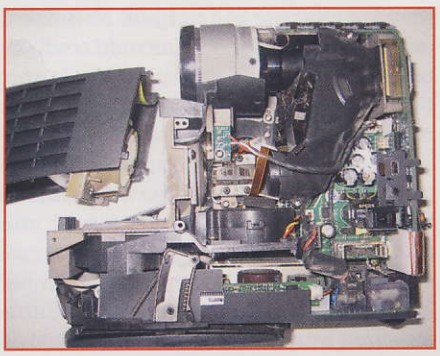
också ett stort antal andra komponenter. Normalt genereras det ljus som behövs för displayen av en dyrbar gasurladdningslampa. Ljuset går först genom ett färghjul. Detta är ett hjul med ett antal färgfilter, vanligtvis röda, gröna och blå. Filtren placeras framför ljuskällan med mycket hög hastighet. Ofta snurrar detta hjul med 100 varv per sekund.

Efter färghjulet kommer en ljustunnel som består av fyra sammanfogade speglar, vilket skapar en typ av rektangulär 'tunnel' av speglar. Denna samlar in och homogeniserar ljuset. Ljuset går sedan genom en lins som fokuserar bilden på DMD-n. Elektroniken som styr DMD-n håller reda på positionen hos färghjulet och justerar positionen av speglarna i enlighet med detta. Det reflekterade ljuset lämnar projektorn via en uppsättning linser som fokuserar bilden på filmduken.

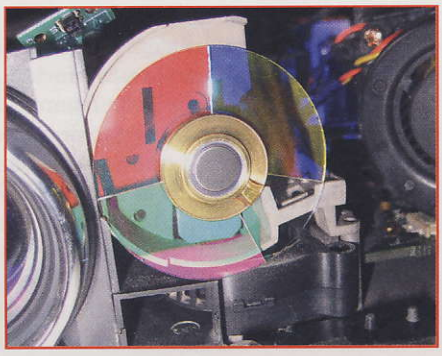
Denna metod har den nackdelen att den skapar vad som kallas för 'regnbågseffekten'. Denna effekt kan ses om du snabbt rör ögat över den visade bilden. Utefter sidorna på bilden kan du då se primärfärgerna.

Det är naturligtvis fullt möjligt att ersätta gasurladdningslampan med en uppsättning vita LED, men det finns en bättre lösning. Om vita LED används kan du i genomsnitt bara använda en tredjedel av det genererade ljuset. Varje segment hos färghjulet släpper bara igenom en av de tre primärfärgerna. Detta är ganska opraktiskt eftersom LED redan är ljussvagare än en 150-W gasurladdningslampa.

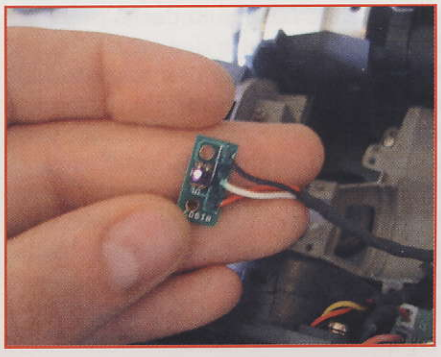
Lyckligtvis kan vi gå runt detta. Vi kan simulera färghjulet genom att använda separata röda, gröna och blåa LED. På detta sätt kan det ljus som DMD-n förväntar sig genereras direkt av lysdioderna. Nu kanske du tror att vi behöver tre gånger så många LED för att få samma ljusstyrka, men så är inte



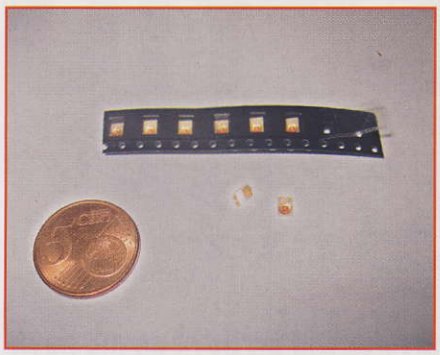
Ljuset färdas från lampan genom färghjulet (bakom det lilla gröna kretskortet) genom ljustunnel till linserna och speglarna i de svarta delarna. Den guldfärgade DMD-n syns överst till höger.



Färghjulet. Som du kan se består det av röda, gröna och blåa segment plus ett vitt segment för ljusstyrkan. Överst till höger finns kretskortet för positionsbestämningen.



Detekteringskretskortet. Den infraröda LED-en och fototransistorn går inte att skilja på. Det normalt osynliga ljuset från lysdioden plockas dock upp av kameran så att du snabbt kan se vilken som är vilken.



Luxeon LED är inte särskilt stora som du kan se. Detta är praktiskt då vi behöver en liten men intensiv ljuskälla. Det gör emellertid lödningen lite trixigare.



Kylningen för lysdioderna. Värmespridaren har tagits bort och exponerat Peltierelementet. Det är här som lysdioderna skall monteras.



Här har lysdioderna snyggt och prydligt limmats fast på Peltierelementet med hjälp av värmeledande lim.

fallet. Eftersom LED-en bara är tända en tredjedel av tiden är det möjligt att öka strömmen genom dem med en faktor tre. Denna metod fungerar bara med DLP-projektorer. Det finns inget färghjul i en LCD-projektor så denna metod fungerar inte med dessa.
Projektet börjar nu ta form. Öppna projektorn, ta ut lampan, ta bort färghjulet, montera in lysdioderna, programmera mikrokontrollern med en färghjulsemulator och det är allt. Då uppstår frågan, kan det verkligen vara så enkelt? Det finns bara ett sätt att ta reda på detta och det är att försöka.

Vad vi behöver

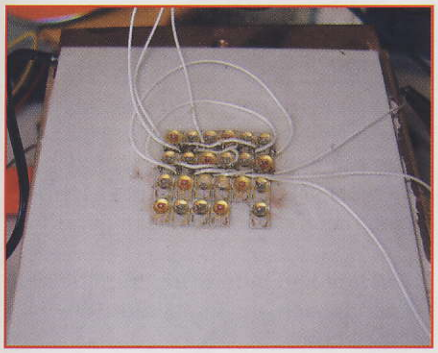
För att få igång detta projekt behöver vi först en projektor. Efter att ha kollat på Internet hittade vi en trevlig DLP-projektor, en iPaq MP3800 tillverkad av Compaq (nu en del av HP). Denna projektor har den fördelen att den är mycket liten och har en upplösning på 1024x768 pixlar, vilket ger en mycket skarp bild. Kritiken mot den var att den väsnades för mycket och att lampans livslängd inte var den bästa utan kunde vara så kort som 500 timmar. I vårt fall spelar detta ingen roll. När projektorn konverteras till att användas med LED så är dessa problem lösta.
Medan vi ändå beställer via Internet så kan vi lika gärna skaffa de övriga komponenterna också. De LED som behövs är (röda, gröna och blåa) Luxeon Rebels. Detta är ytmonterade (SMD) LED med en yta ungefär lika stor som tändsatsen på en tändsticka och kan dra upp till 700 mA då de producerar 100 lumen med ljus. De är inte alltför dyra heller. En sats om 24 av dessa kan man få för runt 50-60 euro (har inte hittat något svenskt pris) och detta är då 2400 lumen av rå LED-styrka.
Eftersom dessa LED är effektenheter genererar de betydligt mer värme än en vanlig 5-mm röd indikator-LED. Vad värre är, om du inte vill att utgångsljuset från LED-en skall falla så måste de kylas. Och då vi inte vet exakt hur mycket värme de genererar och då varje grad varmare minskar livslängden några hundra timmar så bestämde vi oss för en lösning som kan kanske kan betrakta som 'overkill'. Från överklockningssektionen i datorvärlden skaffade vi en kylelement-fläkt-Peltier kombination som är avsedd för en Pentium 4 processor. Denna enhet kan kyla sin yta under fryspunkten och kan avleda upp till 130 W värme. Och då de masstillverkas är de inte dyra. De kostar några hundra kronor och då får du i allmänhet också med en kontrollpanel med en (återanvändbar) blå LCD-display.
Och när vi ändå köper överklock-ningsprodukter så kan vi också köpa ett par tuber med 'termiskt lim'. Detta är ett tvåkomponentslim som leder värme bra. Det är avsett för att limma kylelement på IC-kretsar men vi använder det för att limma lysdioderna på kylelementet.

Nu kan vi börja

Hur skall vi nu få dessa delar till någonting som fungerar? Först måste vi kolla om vi kan få lysdioderna att producera de tusentals lumen som de är märkta med. Lysdioderna måste också monteras på ett kylelement annars överhettas de. Vi såg direkt att detta under bli ett problem, lysdioderna är SMD-komponenter som har anslutningarna på undersidan. Om vi skulle montera dem på ett kylelement skulle detta uppenbarligen producera en kortslutning.
Nu finns det tack och lov en enkel lösning på detta. När kylelementets metallplatta skruvas loss så exponeras Peltierelementet. Utsidan av detta är tillverkat av ett icke-ledande keramiskt material som inte orsakar någon kortslutning. Dock, när lysdioderna satts fast på elementet går det inte längre att komma åt anslutningarna. Nu finns det lyckligtvis två 'banor' uppe på lysdioderna. När skyddslagret skrapats bort gå det lätt att löda på dessa.
Nu till anslutningarna. Det är naturligtvis möjligt att ansluta ett strömbegränsningsmotstånd till varje individuell LED men detta skulle medföra 24 motstånd som alla konsumerar effekt och genererar värme. En bättre lösning är att seriekoppla alla LED av samma färg. Detta kräver självklart en högre spänning men strömmen behöver bara begränsas på ett ställe. LED-kedjorna görs genom att löda tunna trådar ovanpå lysdioderna.
För att testa lysdioderna så anslöts alla kedjorna till ett bänkaggegat via ett strömbegränsande motstånd. Var försiktig när du tänder dem då tvåtusen lumen är mycket starkt!

Reverse engineering

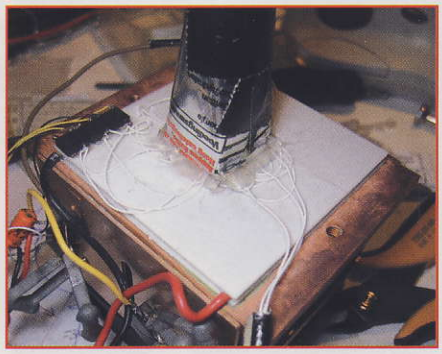
När vi nu har tagit hand om själva belysningen så skall vi ta en närmare titt på projektorn. Med sidostycket borttaget är det lätt att se alla delarna, lampan, färghjulet, ljustunneln och resten är lätt att känna igen. Noterbart är att lampan måste kylas med tre fläktar, vadå brusigt?
För att kolla att vår idé fungerar så måste vi ta bort färghjulet och emulera detta på något sätt. Färghjulet är ett typexempel på de man hittar i projektorer. Motorn som används för att vrida runt hjulet är en stegmotor även om återkopplingen för bestämningen av hjulets position görs optiskt. Det finns en svart punkt på hjulet som detekteras med en IR LED och en fototransistor. Detta är mycket användbart! Allt vi behöver göra när vi tagit bort färghjulet är att emulera den signal som nor-



Den första halvan av lysdioderna har kopplats ihop med hjälp av denna tunna kynartråd. Använd aldrig superlim för att hålla trådarna på plats. Alla reparationer på lösa trådar blir då extremt svårt.



Endast hälften av lysdioderna är anslutna och kameran kan just precis hantera ljusstyrkan. Första testet är en succé!

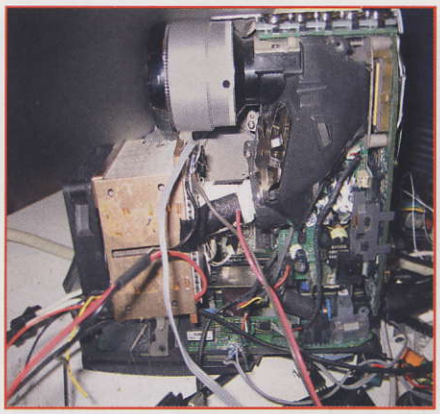


Den första snabba testuppsättningen. Alla LED är anslutna och några 'snabba-och-smutsiga' matningar har kopplats in. Höljet förhindrar oönskat ljusläckage.

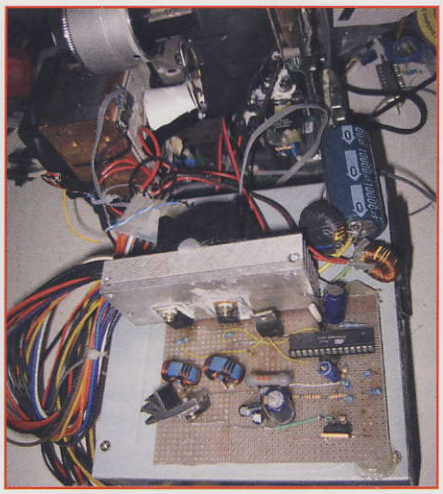
malt genereras av fototransistorn och mata den till DMD-elektroniken. En snabbtitt på oscilloskopet visar att detta är en 100 Hz signal, vilket betyder att färghjulet gör ett varv var 10 ms.
Det är mycket enkelt att programmera färghjulsemuleringen i en mikrokontroller. Här använde vi oss av en ATMega88. Tre MOSFET-ar adderades för att tända och släcka LED-kedjorna.
Om man tittar på färghjulet ser man att den första $^2/_7$ är blå, därefter $^2/_7$ röda, sen $^2/_7$ gröna och slutligen $^1/_7$ vita. Det är enkelt att skriva ett program som emulerar detta.
För att frigöra lite utrymme för kylelementet och lysdioderna var vi tvungna att ta bort fläktarna. Det gav direkt upphov till ett annat problem. Trasig (eller borttagen) fläkt betyder att en ickemodifierad projektor överhettas och vägrar att fungera. Nu är det ganska lätt att lura i projektorn att fläktarna fortfarande finns och att de fungerar. Alla fläktar har en gul ledning som dras till jord när fläkten är igång. Allt som behöver göras är att jorda denna anslutning på projektorkretskortet och elektroniken kommer att vara lycklig.
Projektorn har också en annan skyddsmekanism. Skulle logiken som driver lampan inte indikera inom några sekunder att lampan har tänts korrekt så vägrar elektroniken att fungera. Denna signal är också lätt att hitta. I denna projektor är drivkortet för lampan galvaniskt isolerat från resten av projektorn med två optokopplare. En indikerar att lampan skall vara tänd och den andra returnerar lampstatusen. Dessa optokopplare kan tas bort (avlödas). Den första ersätts med en LED så att vi kan se när logiken vill tända lampan. Den andra ersätts med en (normalt sluten) tryckknapp så att vi manuellt kan simulera 'allt klart' från lamplogiken.
Nu handlar det bara om att koppla ihop allting och kolla om konceptet är genomförbart. Kylelementet med lysdioderna placeras inuti projektorn och ljustunneln, som tagits bort tillsammans med fläktarna och färghjulet ersätts med vår egen ljustunnel av tunn plåt. Anslut strömmen, starta projektorn och...
Wow, en bild!
Firmware behövde lite trimning innan alla färgerna kom ut korrekt, men vi hade en bild. Den var inte så ljusstark som med originallampan, men detta var väntat. Nu över till de övriga delarna.

Strömförsörjningen

Bortsett från 12 V för fläkten och 5 V för Peltierelementet så behöver projektorn också en matning för lysdioderna som ju kom från ett bänknätaggregat. Detta måste självklart bytas ut mot en riktig nätdel när projektorn skall användas på normalt sätt utanför labbet. I vårt fall skulle det också öka projektorns ljusstyrka. Bänknätaggregatet skulle ge 60 watt bara för lysdioderna och det var mer än vad vårt kunde leverera.
För att hålla allting flyttbart så lade vi till en kraftfull boostomvandlare till kretsen (se figur 1) som omvandlar 12-V matningen till en 2,1 A strömkälla för lysdioderna. Den har till och med implementeras som en dubbel boostomvandlare. L2 och L2 'laddas' växelvis via T1 och T2. D2 och D3 skickar strömmen vidare till kondensatorbanken, C6-C8. ATMega88-an mäter spänningen över shuntmotståndet R11 för att kontrollera spänningen över och strömmen genom lysdioderna.
Kondensatorbanken kan kanske verka lite konstig till att börja med. Skälet till att använda tre kondensatorer med lägre värde beror på den ekvivalenta serieresistensen (ESR). En kondensator har en inre resistens som 'normalt' inte har någon större påverkan. Men i denna krets laddas och urladdas kondensatorerna i banken tiotusentals gånger i sekunden med höga strömmar. Och då får man definitivt räkna med ESR. En singelkondensator får man skrapa ner från taket efter en kort stund (vilket författaren vet av bitter erfarenhet), trots det faktum att vi aldrig överskred kondensatorernas spänningsgräns. Att koppla tre kondensatorer parallellt sprider lasten och minskar den totala serieresistensen.
Det totala värdet på kondensatorbanken får inte heller vara för stort. När du växlar mellan olika färgade LED med lägre framspänningsfall så vill du inte att kondensatorbanken ska dumpa överspänningen in i LED-arna från en högkapacitiv källa. Detta kan till och med få någonting att fara i luften. En mindre bank har den nackdelen att den producerar ett visst rippel på matningsspänningen, men detta sker vid en frekvens på flera tiotals kHz och spelar ingen roll vid drivning av LED.
Matningsspänningen för mikrokontrollern är satt till 5,6 V med hjälp av en 7805 och en diod. Detta är 0,1 V över den rekommenderade maxspänningen, men i praktiken är detta inget problem. För delen med en något högre matningsspänning är att gate hos MOSFET-arna kan drivas lite hårdare. På detta sätt kan vi vara säkra på att de antingen är hårt till eller hårt från. Med de strömnivåer som flyter i denna krets kommer du att finna att bara delvis ledande MOSFET kommer att skapa viss rök från kretsen. Detta är också skälet till de extra motstånden från gate hos MOSFET-arna till jord. Utan dessa så skulle ett problem som tvingade mikroprocessorn att lägga sina utgångar i tri-statemod, snabbt få kislet i MOSFET-arna att övergå till sitt gasliknande tillstånd.
LED-drivarkretsen behöver fortfarande



Så här ser insidan av projektorn ut till slut, lampan, fläktarna, färghjulet och ljustunneln har ersatts med lysdioder och associerade komponenter.



Och så här ser det färdiga projektet ut. Det ser ganska ovårdat ut och genom att använda ett ATX-nätaggregat så har vi inte precis hållit nere storleken. Men det är i alla fall ett bra bevis på att vi lyckades och om allting monteras i en större låda så kan ingen i alla fall se det.

ca 100 W. Adderat till detta är matningen av Peltierelementet. En billig lösning är ett standard ATX-nätaggregat för en PC. Detta kan vanligtvis lämna över 10 A vid 12 V, medan 5 V utgången är lämplig för Peltierelementet. L1 och C1 har lagts till för att skydda nätdelen från störningar orsakade av de strömtoppar som dras av boostomvandlaren. L1 är en avstörningsdrossel från skrotlådan och C1 är en stor elektrolytkondensator.

De stora strömmarna och förlusterna i de olika komponenterna i boostomvandlaren betyder att de måste väljas med omsorg. Både MOSFET-arna och spolarna måste kunna hantera mycket höga strömmar. Men det är ganska lätt att hitta högströms MOSFET. IRFZ48V gör jobbet bra och är lätt att få tag på. Spolarna L2 och L3 kan vara svårare att få tag på. För prototypen användes två spolar från ett gammal P4 moderkort. De består av tre parallella lindningar av 1 mm² koppartråd som lindats runt ringkärnan fyra gånger.

För D2 och D3 i prototypen användes CTG24S ultrasnabba dioder från Sanken. De togs från en gammal nätdel och har en TO-220 kapsling, vilket betyder att de enkelt kan monteras på ett kylelement. Men även vanliga Schottky effektdioder borde vara lämpliga.

Medan vi ändå är inne på kylelement så rekommenderar vi att MOSFET-arna också förses med små kylelement.

Slutresultatet

Är idén genomförbar och fungerar den? Enligt författaren så gör den det. Denna LED-modifiering är perfekt för att omvandla en gammal projektor till en billig 'filmvisare'. Ytan som du projicerar på skall vara ganska mörk, men då får du också en bra synupplevelse. Att bygga in detta projekt i en sprillans ny projektor är dock inget vi rekommenderar.

Finns det utrymme för förbättringar? Troligen. Det sker ny utveckling hela tiden inom LED-världen. Och varje extra lumen som kan dras ur lysdioderna ger en ljusstarkare bild.

Dessutom är det optiska systemet i experimentprojetorn inte perfekt. Det är inte allt ljus från lysdioderna som når DMD-n via tunneln, en del läcker ut på vägen. När ljuset fokuseras för mycket får du färgade fläckar på displayen. Och optiska komponenter är svårare att få tag på än de elektroniska delarna.

Firmware kan laddas ner från Elektors websida och du kan modifiera den efter behov.

(070690-I)

Weblänk

http://www.alltomelektronik.se

KRETSKORTSFABRIKEN



Peter

Hej, här arbetar jag och klipper till dina kretskort som du vill ha dem.

Från 0,50:-/ cm²

Glasfiberlaminat 1.55 mm 35 µm Cu
max 200 x 200 mm *utan* fotoresist 0,50:-/cm²

Glasfiberlaminat 1 mm 35 µm Cu
max 200 x 200 mm *med* fotoresist 1,25:-/cm²

Glasfiberlaminat *dubbelsidigt* 1 mm 35 µm Cu
max 200 x 200 mm *med* fotoresist 1,50:-/cm²

Framkallning och etsning av dina egna *ej dubbelsidiga* layouter max 160 x 160 mm
150:-/kort

Borrning 0,5 mm 0,7 mm 1,0 mm
max 160 x 160 mm 50:-/kort

JPS Tel 0522-174 90 Öppet även kvällstid

PCB-POOL®

SERVICING YOUR COMPLETE PROTOTYPE NEEDS

- Prototypes at a fraction of the cost
- Tooling and setup included
- Any contour
- Fr4 1.6mm, 35µm Cu
- Industry standard quality
- Follow up series runs
- CAM / CAD consulting

PRICE EXAMPLE:
1 EUROCARD DS/PTH
+ Tooling
+ Photoplots
+ VAT
=

€49

FREE with first order (while stocks last)



may differ slightly to model shown

Beta LAYOUT

Tel.: +353 (0)61 701170
Fax: +353 (0)61 701164
E-Mail: sales@beta-layout.com

ROHS / WEEE konform

Industry Quality LEAD FREE FR4, DS, DK, HAL

Simply send your files and order ONLINE:

WWW.PCB-POOL.COM

p-cad TARGET Protel EDWin orcad GraphiCode PROTEUS Electronics Easy-PC

PROTEUS

The Complete Electronics Design System



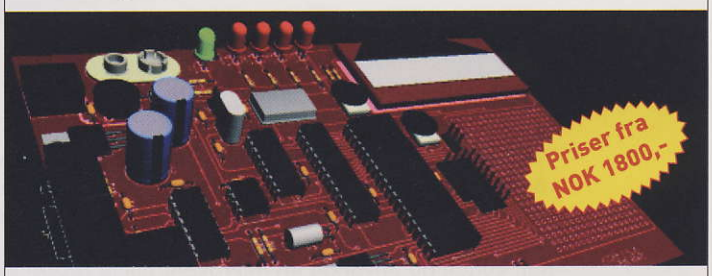
Proteus VSM

Co-simulation of microprocessor software within a mixed mode SPICE simulator.

- Available for PIC, 8051, AVR, HC11, ARM7/LPC2000 and Basic Stamp processors.
- See your code interact with simulated hardware in real-time.
- Interactive peripheral models for displays, keypads, etc.
- Over 8000 analogue and digital device models.
- Extensive single step and debugging facilities including system wide diagnostics.
- Works with popular compilers and assemblers.

Proteus PCB Design

Professional schematic capture and PCB design software with automatic component placement, track routing and design validation.



- Includes full feature schematic capture environment.
- Create PCBs automatically or manually from a schematic.
- Extensive support for power planes.
- Outputs to industry standard CADCAM formats.
- Integrated 3D Viewer provides board visualisation during design.
- Interactive SPICE circuit simulator included with all versions .

Proteus VSM is highly recommended for designers frequently working on circuits containing digital as well as analogue electronics.
– Elektor Electronics Review

PEDTEC

www.pedtec.no

Official Labcenter Distributor

labcenter Electronics

LabEquipment.nu

Vår bästsäljare:

I2C/SPI via USB!

Embedded Systems Interface



Protocol Analyzer

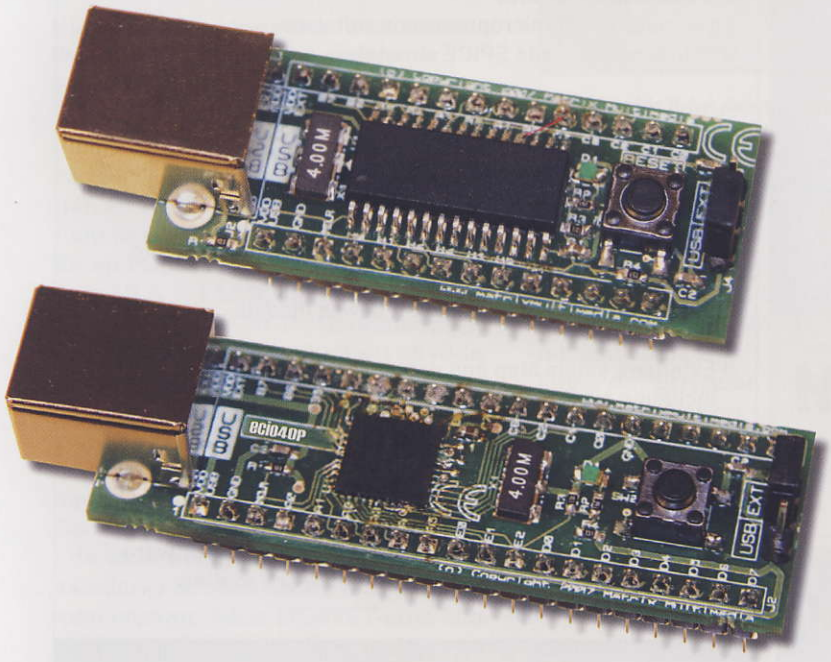
BEAGLE I2C/SPI

AARDVARK I2C/SPI

www.labequipment.nu www.beta-phase.com

EasyControl I/O

ECIO: ett superbilligt USB-aktiverat PIC18 utvecklingssystem



John Dobson

Många läsare som är bekanta med PIC mikrokontroller vet att flera av dessa fantastiska komponenter redan har en inbyggd USB-port. Så varför behöver du då en speciell programmerare för en PIC om den redan har en USB-port. Svaret är att det behöver du inte. Läs vidare...

Förr i tiden var livet enklare. Alla datorer som man hade råd med hade en parallellport och en serieport. Du kunde sätta samman en enkel krets för att programmera en mikrokontroller med hjälp av PC-ns parallellport och sedan kunde du använda en mikrokontrollers USART (universal serial asynchronous receiver transmitter — ber om ursäkt för det) för kommunikationen tillbaka till PC-n via serieporten. Sedan kom USB för att 'göra livet enklare' för oss. Och det faktum att moderna datorer idag inte har varken parallella eller seriella portar betyder att du måste köpa en USB mikrokontrollerprogrammerare. Tack och lov så håller mikrokontrollerna nu på att komma i kapp, många mikrokontroller har en inbyggd USB-port som implicerar att det återigen är möjligt att plugga mikrokontrollern rakt in i PC-n. Nåja, nästan i alla fall... Då det inte är möjligt att skapa en komplett mikrokontrollerprogrammerare med enbart USB-linjerna så är det möjligt att göra det näst bästa, att utveckla ett stycke mjukvara som kallas för 'bootloader' och som finns inuti en USB-aktiverad mikrokontroller och som gör att mikrokontrollern kan ladda in ett program via USB-ledarna.

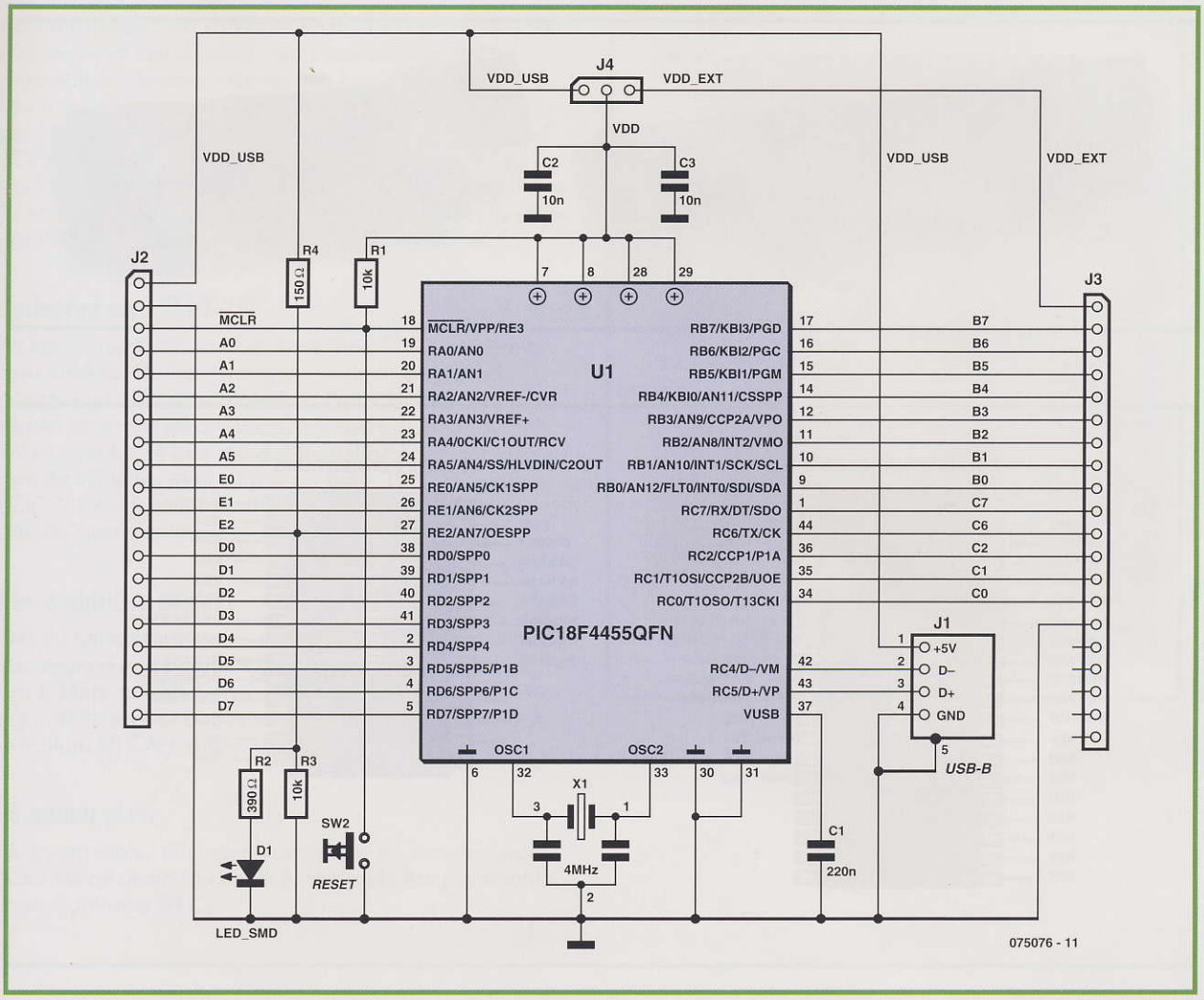
Kretsdetaljer

Inom produktområdet PIC-mikrokontroller finns det nu flera mikrokontroller i 18-serien som har USB-portar integrerade i chipet, speciellt 18F4455 40- eller 44-pinnars komponenterna och 18F2455 28-pinnars komponent. Att koppla upp detta i en krets kan vara mycket enkelt. Slit dig från datorspelet, stäng av TV-n, ta PIC-en, lägg till en 4 MHz oscillator till resonatorpinnarna, anslut de två PIC USB-linjerna till PC-ns USB-linjer, lägg till en resetswitch och starta upp.

När det gäller strömförsörjningen har du två alternativ. Du kan ta 5 V från en extern spänningskälla eller så kan du använda PC-ns egen kraftkälla på USB-porten. En liten varning när det gäller detta, du är inte garanterad att mer än 100 mA finns tillgängligt på din USB-port eftersom tillgängligheten beror på hur många enheter som är inpluggade. I de flesta fall kommer det dock att finnas runt 250 mA.

Kretsschemat för ECIO40P med en 44-pinnars PIC18F 4455 visas i figur 1. LED D1 indikerar när USB-kontakten pluggats in och med jumpern kan du välja om strömmen skall tas från PC-n eller från en extern källa.

Kretsen för den mindre (28-pinnars) enheten (ECIO28P) är mycket lik — du får bara färre I/O (ingångs/utgångs) linjer. Och där har du den, **EasyControl I/O** eller kort, ECIO.



Figur 1.
Komplett kretsschema för 18F4455 ECIO-versionen.

Bootloaderdetaljer

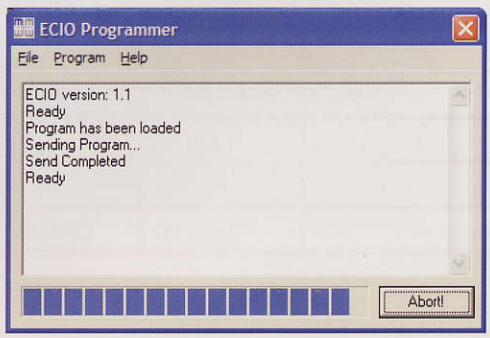
Bootloadermjukvaran för ett system som ECIO måste vara enkel men effektiv. Det bestämdes att bara en switch för både Reset och Programmering skulle vara bekvämt. Mjukvaran har skrivits på så sätt att när Resetswitchen hålls in så tittar bootloadermjukvaran efter om någon USB-kontakt har pluggats in. Om den upptäcker att det finns en USB-enhet implementerar bootloadermjukvaran ett nedladdningsprogram som tillåter att PC-n kommunicerrar med PIC-en och skickar över ett nytt program till den. Om ingen USB upptäcks stängs bootloadern och användarprogrammet körs. Och det kan eventuellt vara ditt eget program!

Förutom bootloadermjukvaran behövs det en Windows-baserad applikation för att tillåta användare att sända hexkoder till mikrokontrollern. Du kan se detta i figur 2. I linje med att Elektor finns över hela världen så finns ECIO-mjukvaran tillgänglig på fem språk, engelska, franska, holländska, spanska och tyska. Med denna kan du lokalisera en hexfil i datorn och ladda ner denna till enheten.

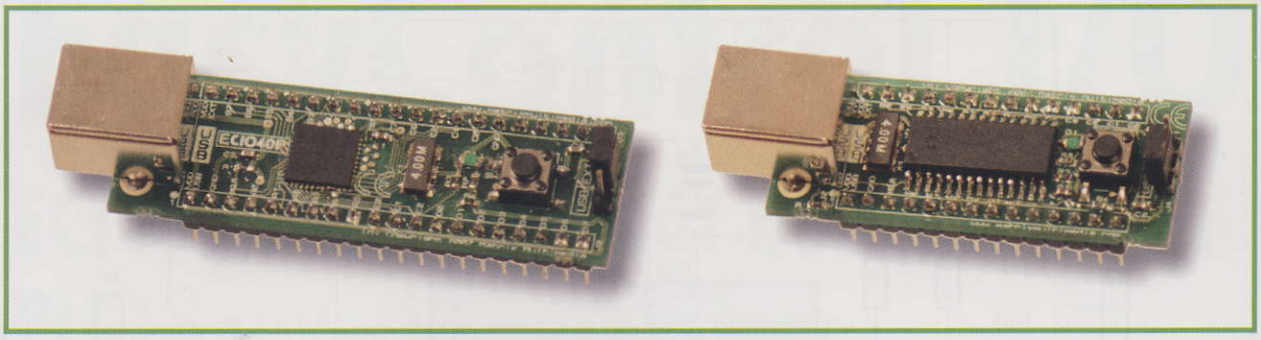
Plocka ihop allting

För bekvämlighets skull har vi designat och byggt ihop ett litet kretskort som inkluderar PIC-mikroenheten, en resonator, en USB-kontakt och en resetswitch. För var och en av de 28-pinnars och 40-pinnarskomponenterna i figur 3 ger detta en 'färdig att användas' komponent som kan pluggas direkt in i ett applikationskort eller ett kopplingsdäck.

För båda ECIO-enheter finns det också ett **E-blockkompatibelt applikationskort** tillgängligt – detta gör att du kan dra fördel av det stora urval med E-block som nu finns tillgängligt.



Figur 2.
Windows nedladdningsmjukvara för ECIO.



Figur 3.
Bild på ECIO40P och ECIO28P.

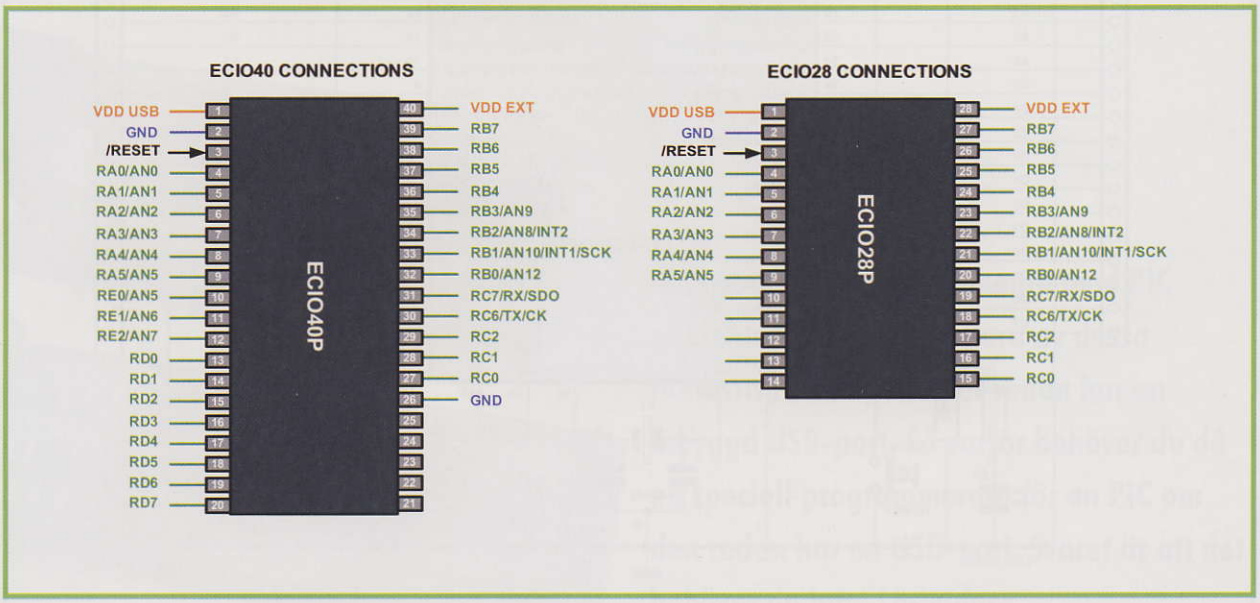


Figure 4.
Pinkonfigurationen för ECIO40P och ECIO28P.

Läsarerbjudande

Denna månad kan vi erbjuda våra läsare ECIO-kort så billigt som 350:-/st.

Vi erbjuder också en sats med ett ECIO-40P, applikationskort, LED-kort, switchkort, LCD-kort, reläkort och optoisolatorkort samt Flowcode Professional med 30% rabatt i förhållande till normalpriset. Denna sats är idealisk för er som vill utveckla en Programmable Logic Controller baserad på en ECIO.

Komponent(er)	Pris inkl. moms.	Pris exkl. moms.
ECIO28P (1 st)	350:00	280:00
ECIO28P (4 st)	1100:00	880:00
ECIO40P (1 st)	450:00	360:00
ECIO40P (4 st)	1500:00	1200:00
E-block adapterkort	450:00	360:00
E-block ECIO Startsats	3600:00	2880:00

Porto tillkommer.

Schemorna i figur 4 visar anslutningarna för 28-pinnars (ECIO28P) och 40-pinnars (ECIO40P) DIL-kapslarna. Notera att detta är förenklade representationer av komplexa chip – se databladen över PIC-mikroenheter för kompletta detaljer. Som du kan se har 40-pinnars ECIO sex bitar från port A, en full port B, fem bitar från port C, en full port D och tre bitar från port E. Tyvärr är mycket av port C inte tillgängligt, den interna USB-modulen använder bitarna 3, 4 och 5 hos denna port. 28-pinnars ECIO-enheten är mycket lik, bara med färre I/O-linjer. Valet mellan 28-pinnar och 40-pinnar är ... ditt.

Utvecklingsmjukvara

ECIO- enheter kan användas med all mjukvara som genererar hexkod för 18-serien av PIC-mikrokontroller.

Så, om du har en kompilator eller assembler som är kompatibel med 18-serien av Microchips PIC-mikrokontroller så har du allt som behövs för att komma igång. Notera att bootloadern finns i PIC-mikron mellan platserna 0x000 och 0x7FF och din kod måste därför börja vid plats 0x800 (detta kan vanligtvis utföras med ett enkelt kommando till Linkerverktyget).
Om du inte har någon mjukvara så skall du inte förtvivlas, en gratisversion av Flowcode finns tillgänglig med båda ECIO-enheterna som låter dig skapa ganska så funktionella program. Denna gratisversion av Flowcode är densamma som den fulla versionen men med vissa

begränsningar. Den kan bara kompilera upp till 2k kod, inkluderar ett litet dsubset med komponenter, tillåter maximalt 64 flowchart ikoner och kan inte användas för kommersiella ändamål. Naturligtvis finns det en uppgradering till version 3.2 av Flowcode (kompatibel med ECIO40) så att du kan komma åt alla egenskaperna, inkluderande CAN-buss, Bluetooth, TCP/IP och mycket mer. Ni som har Flowcode 3 och senare kan få en gratis uppgradering som är kompatibel med ECIO-familjen.

Prototypa med ECIO-28P

Att koppla upp och köra en krets med ECIO kunde inte vara enklare. Du kan antingen använda ett E-block adapterkort eller ett prototypkort (figur 5). När du använder ett prototypkort (kopplingsdäck) pluggar du helt enkelt in enheten i en standardpanel och ansluter kretsen som du vill testa. Med ett pris på bara 350:- för varje ECIO28P är detta förmodligen den billigaste vägen till USB PIC-programmering i världen.

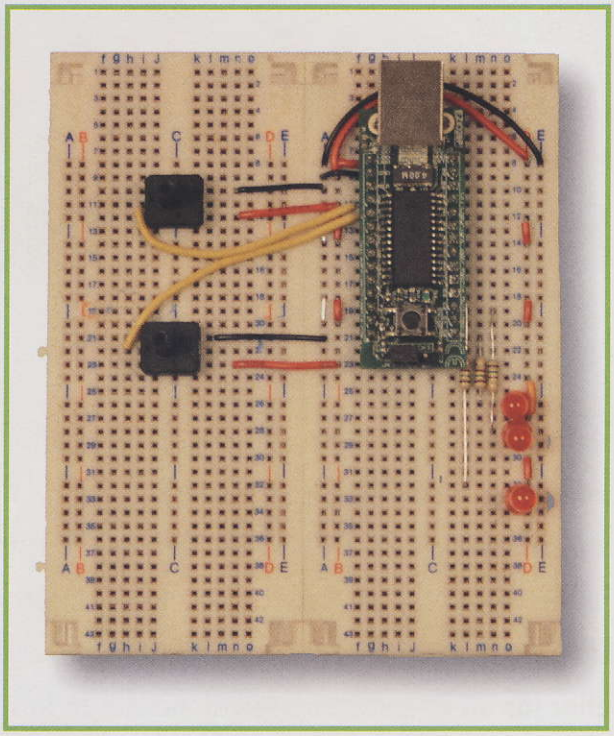
Mer ambitiösa projekt

Om du har större projekt, eller om du vill kunna ansluta mer avancerade system tillsammans, kan du använda den E-block adapterpanel som visas i figur 6. Detta låter dig ansluta en stor uppsättning kort från enkla LED och switchkort till CAN-subsystem etc.

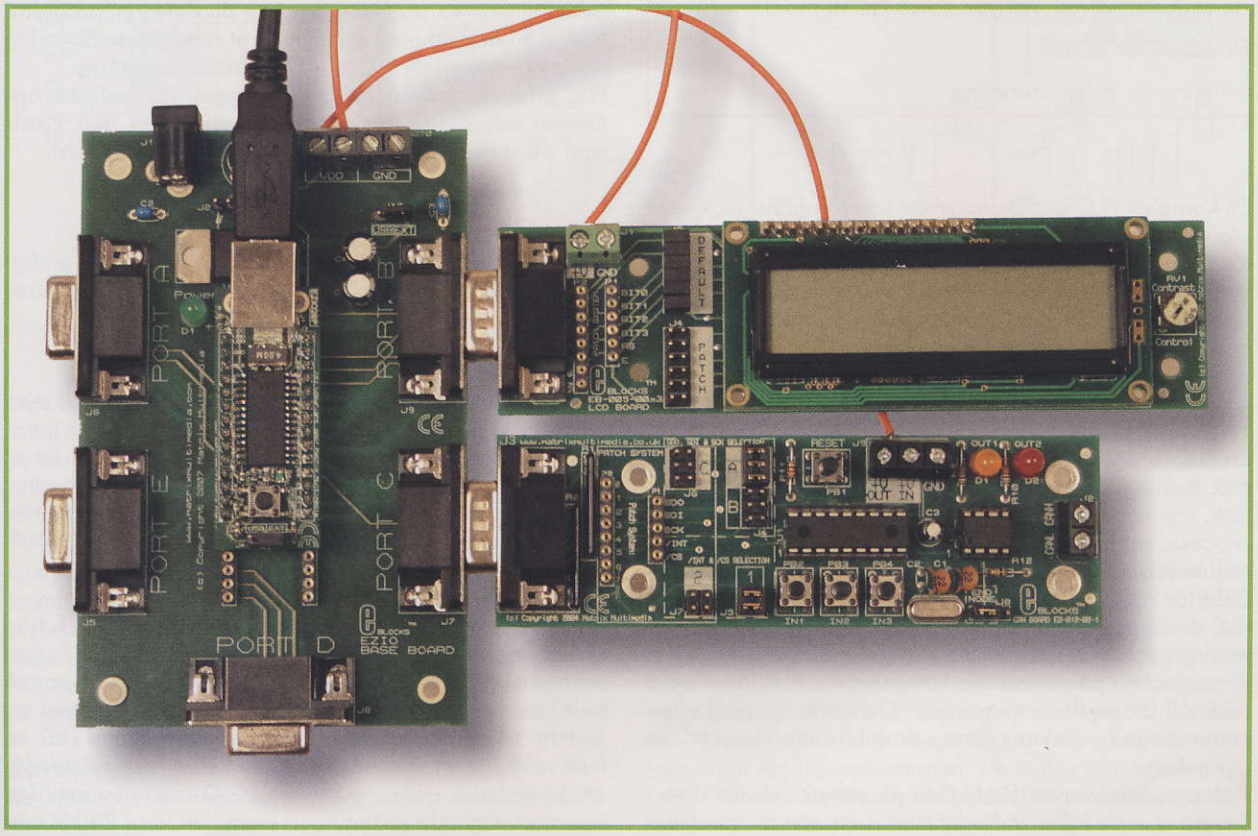
På annan plats

På annan plats i tidningen visar vi hur du kan använda ECIO för att utveckla en fullt fungerande Programmable Logic Controller (PLC).

(075076-I)



Figur 5.
Användningen av ECIO för projektutveckling på ett kopplingsdäck.



Figur 6.
En ECIO med adapterkort och E-block.

ECIO PLC

Billigt, gör-det-själv och med CAN!

Ben Rowland (Matrix Multimedia) & Luc Lemmens (Elektors Lab)

Här är den första 'real-life' applikationen för de ECIO-moduler som finns på annan plats i tidningen. En ECIO fungerar som hjärnan i ett PLC-kort som har reläer, optokopplare, CAN (!) anslutning och en LCD. All denna I/O-kapacitet tillsammans med Flowcode gör att kortet kan fungera som ett universellt kraftfullt PLC för ganska komplex styr- och automationsprojekt. LCD-modulen används för att visa ASCII-tecken vid felsökning under mjukvaruutvecklingen eller för att övervaka systemet när det är igång.

ECIO PLC egenskaper

- CAN-bussanslutning
- 4 optiskt isolerade ingångar
- 4 relädrivna utgångar
- 2x16 teckens alfanumerisk LCD
- Flowcode-programmerad men tar också PIC18 hexfiler
- Gratis Flowcode för ECIO
- USB-anslutning för programmering

Ett komplett PLC-applikationskort som omfattar isolerade utgångar, ingångar, LC-display och CAN-bussanslutning... låter bra! Men du vill säkert också höra att det är billigt, baseras på en ECIO-modul och är lätt att programmera med hjälp av Flowcode, den grafiska metoden för att behärska PIC (och även andra mikrokontroller).

PLC

En PLC (programmable logic control, programmerbar logisk styrning) är en enhet som typiskt används som ett centralt, intelligent styrelement i en flödesdesignad industriell process, vanligtvis för massproduktion eller kvalitetskontroll och godssortering. Styrning av transportband är typiskt för PLC.
PLC existerar kommersiellt och deras (högnivås) programmeringsspråk har i stor utsträckning standardiserats. Tyvärr är prislappen på nästan alla kommersiella PLC långt utom räckhåll för de flesta entusiaster. Och detta är synd eftersom många av Elektors läsare är väl insatta i industriella styrsystem.
Kommandosekvenser för PLC är på sätt och vis likt datorprogram — de följer en fördefinierad sekvens av händelser med alla alternativ för villkorade loopar, händelsetimers, logiska villkor, händelseräknare, analoga värden (temperatur, vätskenivå, tryck) och enkla resultatloggar.
Om du har en lite komplex process som som du skulle vilja styra med hjälp av elektronik på en maskin så är PLC ett bra val eftersom det är elektriskt robust och flexibelt när det gäller anslutningen till den 'riktiga världen' med hjälp av reläer och optokopplare.
För att kunna programmera en PLC (dvs definiera händelseföljden i en process) så måste du skriva ett program. När detta är avbuggat och simulerat i en PC kan du ladda ner programmet till PLC-n och sen hålla tummarna.
PLC-program kan redigeras, avbuggas, expanderas, optimeras och naturligtvis sparas och användas igen. Precis som vilket PC eller mikrokontrollerprogram som helst.

Kretsbeskrivning

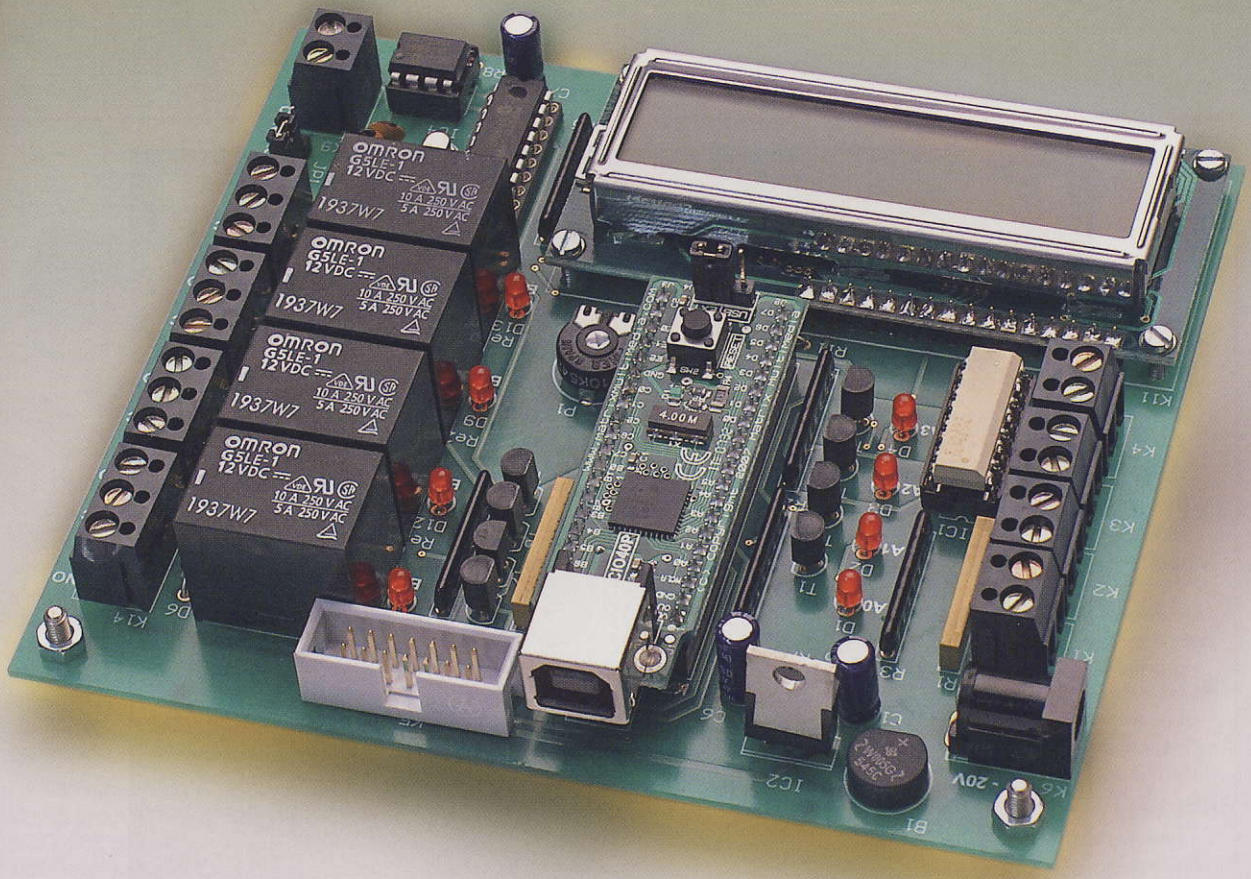
Kretsschemat för ECIO PLC visas i figur 1. Kretsen omfattar ett antal distinkta element som kommer att diskuteras nedan.

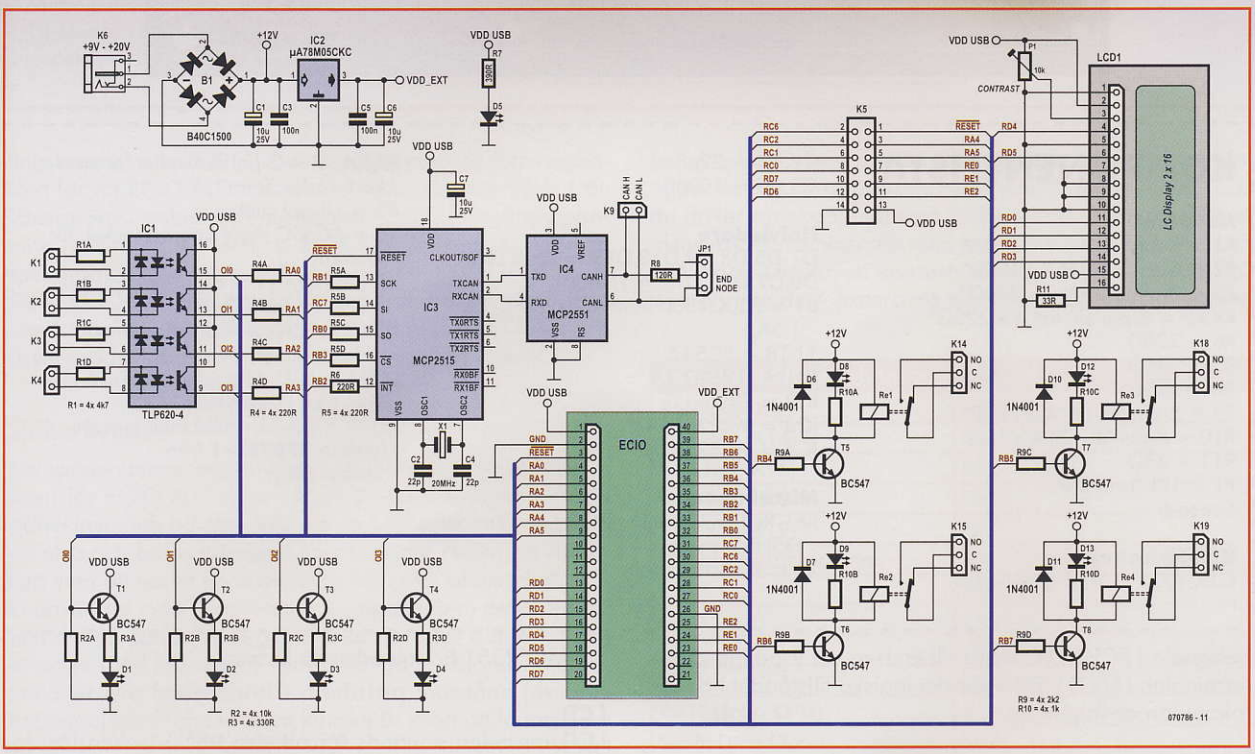
ECIO

ECIO-modulen är kortets hjärna. Den används för att styra alla kringkretsarna på kortet. De ECIO-moduler som introduceras i detta nummer representerar ett ultrabilligt sätt att träda in i programmeringsvärlden för PIC mikrokontroller. ECIO finns att köpa från oss med en bra mängdrabatt. Här använder vi den 40-pinnarsversion som kallas ECIO-40P.

Reläer

De reläer som PLC-n använder för att switcha till och från elektriska enheter styrs via ECIO-pinnarna RB4-RB7. Om man skickar ut en logiskt Låg signal på dessa pinnar så switchas motsvarande relä från och en Hög kommer att switcha till reläspolen. Om ett relä aktiveras tänd LED-en intill relät för att visa att det är till. Reläerna används för att ge elektrisk isolering mellan ECIO-modulen och den externa switchade spänningen (som kan vara 230 V nät-



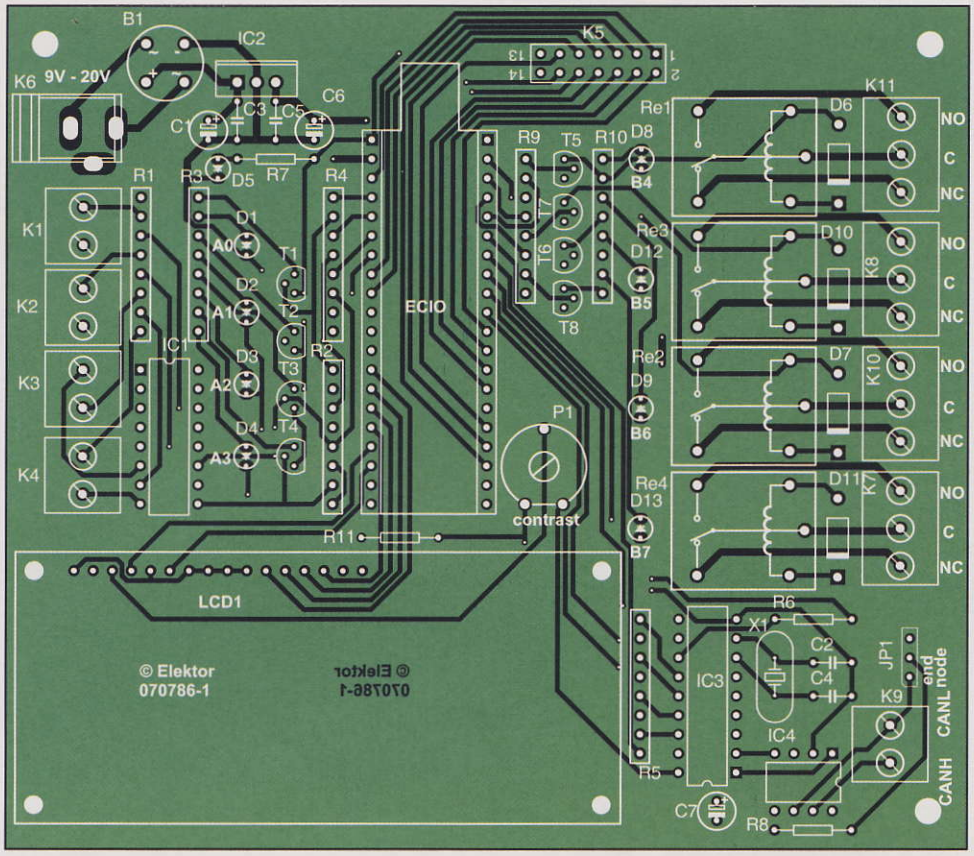


Figur 1.
Kretsschemat över ECIO PLC-kortet. Minimal hårdvara — stor potential när det gäller I/O.

spänning). Detta betyder att enheter som körs från höga spänningar som 48 VDC eller 230 VAC kan styras säkert via en ECIO-modul.
Varje uppsättning reläkontakter (NO,normalt öppen och NC, normalt sluten) dras ut till tre stift på en kopplingsplint på kortet — mittstiftet är polen.

Optokopplare

De optoisolerade ingångarna är speciella i det att de inte har någon plus (+) eller minus (-) ingångsterminal. Inuti IC1 är de två dioderna på varje ingång lysdioder så polariteten hos den styrspänning som du vill lägga till ECIO PLC spelar ingen roll! Optokopplarnas utgångar är anslutna till ECIO-pinnarna RA0-RA3. Dessa ingångspinnar kommer att vara vid logiskt noll vid ingen ingångsspänning och logiskt 1 för en ingångsspänning på 3,5 V eller mer. Optokopplarna används för att skapa ett isolerande lager mot ingångsspänningen. De kallas därför också för optoisolatorer. Det betyder att relativt höga spänningar kan användas för att styra ECIO-modulen. De signaler du vill



Figur 2.
Det obestcyckade ECIO PLC-kortet finns att köpa från oss

KOMPONENTLISTA

Motstånd
R1 = 8-pin SIL-nät 4 x 4kΩ7*
R2 = 8-pin SIL-nät 4 x 10kΩ*
R3 = 8-pin SIL-nät 4 x 330Ω*
R4,R5 = 8-pin SIL-nät 4 x 220Ω*
R6 = 220Ω
R7 = 390Ω
R8 = 120Ω
R9 = 8-pin SIL-nät 4 x 2kΩ2*
R10 = 8-pin SIL-nät 4 x 1kΩ*
R11 = 33Ω
P1 = 10kΩ trimpot
* se text

Kondensatorer
C1,C6,C7 = 10µF 25V radial
C2,C4 = 22pF
C3,C5 = 100nF

Halvledare
D1-D5,D8,D9,D12,D13 = 3mm LED
D6,D7,D10,D11 = 1N4001
B1 = B80C1500 (runt hölje; $80V_{piv}$ @ $1.5A_p$)
T1-T8 = BC547
IC1 = TLP620-4
IC2 = 7805
IC3 = MCP2515-I/P
IC4 = MCP2551-I/P

Miscellaneous
Re1-Re4 = 12V relä, SPDT, t.ex. Omron G5LE-1
X1 = 20MHz kristall
K1,K4,K9 = 2-pol skruvplint för kretskort, raster 5mm
K5 = 14-pol stifttag
K6 = AC/DC låg-V adaptersockel, för kretskort
K7,K8,K10,K11 = 3-pol skruvplint för kretskort, raster 5mm
ECIO = ECIO-40P processormodul (Elektor Shop)
LCD1 = LCD, alfanumerisk, 2x16 tecken, t.ex. Displaytech 162
JP1 = 3-pol SIL stiftlist med jumper
Kretskort **070786-1** från Elektor Shop

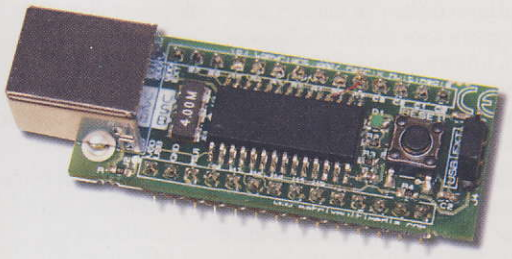
behandla i ECIO PLC läggs till kretsen via 2-poliga skruvterminaler. LED D1-D4 visar det logiska tillståndet hos optokopplarnas ingångar.

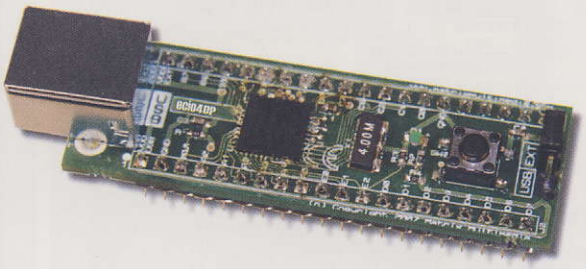
CAN
CAN (controller area network) interfacet används för att addera ECIO PLC till ett CAN-nätverk. CAN är 'inne' och få PLC som vi har sett erbjuder denna möjlighet. Här använder vi ett MCP2515 CAN-kontrollerchip och en MCP2551 linjedrivare. CAN-kontrollerchipet ansluts till ECIO periferipinnar som är inställda för SPI-kommunikation. Det finns också en interruptpinne, som ansluts till ECIO-pinne RB2 och en chip selectpinne som ansluts till ECIO-pinne RB3. CAN-kontrollern har sin egen 20 MHz klocka som den tar från en kristall, X1.
Den egentliga anslutningen av CAN-bussen till ECIO sker via K9, en 2-polig skruvplint. Jumpern JP1 skall installeras endast om MCP2551 är slutnoden för bussen.

LCD
LCD-modulen används för att visa ASCII-tecken för användaren som hjälp vid felsökning under mjukvarans utveckling eller för övervakning av systemet när det är i full gång med att kontrollera och styra händelserna i den automatiserade processen. LCD-n ansluts till ECIO-pinnarna RD0-RD5 där de fyra databitarna tar upp bitarna 0-3, RS-biten tar upp bit 4 och Enablebiten tar upp bit 5. P1 är kontrastjusteringen för LCD-n.

Strömförsörjningen
Inget speciellt när det gäller detta — den vanliga 7805 spänningsregulatorn (IC2) och en tradionell uppsättning avkopp-

ECIO – billigaste USB PIC-programmeraren som finns



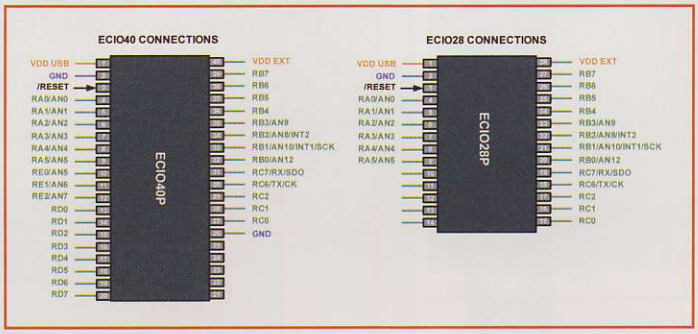


ECIO-familjen av USB programmerbara mikrokontroller är ett enkelt sätt att anpassa mikrokontrollerteknik i dina egna projekt. Enheten uppför sig precis som en normal mikrokontroller – men när du pluggar in USB-kontakten och trycker på resetknappen kan du skicka ett nytt program till enheten. Detta gör ECIO till ett av de billigaste USB-kompatibla PIC-programmerarna i världen.

För närvarande finns det två produkter: ECIO-28P och ECIO-40P. Dessa enheter är baserade på komponenter ur PICmikro 18-serien, 18F2455 respektive 18F4455.

ECIO-mokrokontrollerna är förprogrammerade med ett bootloaderprogram som gör att du kan sända ett nytt program till mikrokontrollern via USB i princip hur många gånger som helst. ECIO är kompatibel med hexkod från vilken lämplig kompilator som helst. ECIO är direkt kompatibel med Flowcode — ett grafiskt programmeringsspråk som kraftigt underlättar kodgenereringsprocessen — men kan också användas med vilken C-kompilator som helst eller med Microchips MPLAB.

ECIO stöds av ett stort urval läro- och utvecklingsverktyg inkluderande (gratis) Flowcode och billiga E-block.



lingskondensatorer. En likriktarbrygga används före regulatorn för att ECIO PLC kortet skall kunna matas både från växelströms- och likströmsadaptrar med en utgångsspänning mellan ca 9 V och ca 20 V. För DC-anslutningen så spelar polariteten ingen roll.
Detta ECIO PLC-kort har en hel del anslutningar till världen utanför och för att inte tappa bort alla dessa I/O-linjer och associerade enheter så har vi summerat dem i tabell 1.

Själva bygget

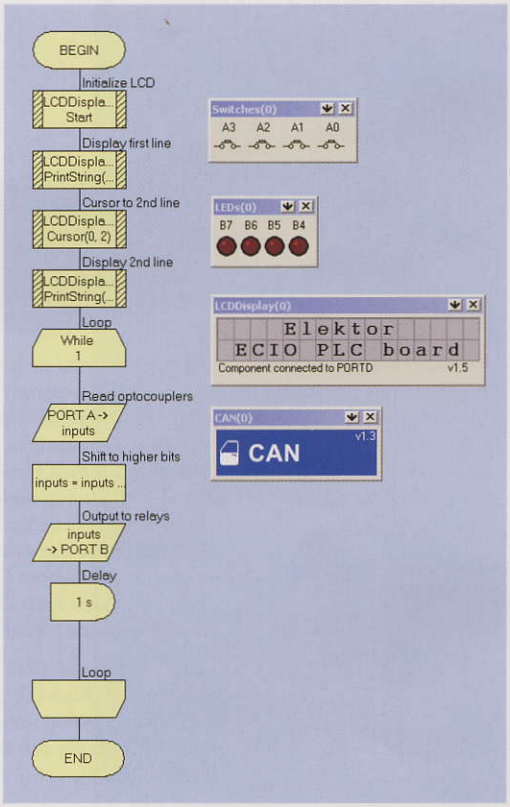
Komponentplaceringsritningen för det kretskort som tagits fram för ECIO PLC visas i figur 2. Som vanligt så måste själva layouten hämtas från vår websida för er som gärna vill etsa och borra era egna kort hemma. Ni övriga vill säkert veta att kortet finns att köpa från oss, obestyckat.
Vi använder inga ytmonterade komponenter i detta projekt och det är därför lättbyggt om du bara tar det lungt och är noga med lödningen. Det 8-poliga motståndsnätet innehåller fyra **individuellt anslutna** motstånd (dvs inte hopkopplade i toppen) som kanske är svårt att få tag på. I så fall kan du använda fyra individuella motstånd i stället som du monterar stående sida vid sida.
Inspektera nu kortet noga innan du sätter i IC-kretsarna och strömsätter det för första gången.

ECIO hårdvarutestning — I/O och CAN

För att hjälpa dig komma igång har vi skrivit ett enkelt och instruktivt ECIO testprogram och Flowcodelistningen visas i figur 3. Ladda ner detta, ladda in det i Flowcode, kör simuleringen, kompilera och blås in det i ECIO-modulen.
Detta program inkluderar en allmän PIC setuprutin som utför all port och I/O-pinkonfigurering på ECIO. Till exempel sätts ECIO-linjerna RB4-RB7 upp som utgångslinjer (för relädrivarna). Testprogrammet är enkelt att använda. Om du driver optokopplaren på RA0 kommer relät på RB4 att aktiveras. På samma sätt med kombinationen RA1/RB5 och så vidare. Displayen kommer att visa 'Elektor ECIO PLC board'. Inte det? Justera då kontrasten på LCD-n. Varken

Tabell 1. ECIO-anslutningar

Reläutgång	**ECIO I/O-pinne**
Relä Re1 (K14)	RB4
Relä Re2 (K15)	RB5
Relä Re3 (K18)	RB6
Relä Re3 (K19)	RB7
Optokopplaringångar	**ECIO I/O-pinne / LED**
K1	RA0 / D1
K2	RA1 / D2
K3	RA2 / D3
K4	RA3 / D4
CAN (från CPU)	**ECIO I/O-pinne**
Serial Data Out	RC7
Serial Data In	RB0
Serial Clock	RB1
/Interrupt	RB2
/Chip Select	RB3
LCD	**ECIO I/O-pinne**
D0	RD0
D1	RD1
D2	RD2
D3	RD3
RS	RD4



Figur 3.
Detta Flowcodeprogram kör ett hårdvarutest på ECIO PLC.

familjemedlemmar eller PIC-fantaster blir väl alltför imponerade av detta, men om det fungerar så är det en hel del PIC-kod som körs och du kan då säkert utgå från att hårdvaran fungerar som den skall.
Den CAN-test som diskuteras i textrutan är också inkluderat i gratismjukvaran för detta projekt. De två testprogrammen är mycket lärorika och är definitivt värda att titta på även om du inte bygger projektet. Tips: Använd den fria Flowcodeversionen till att börja med. Filen heter **070786-11**.

PLCprogrammering

All ECIO PLC programmering påminner mycket om E-block och Flowcode och har du erfarenhet av dessa kan du skatta dig lycklig. Om inte så finns det en mängd information ute på Internet [1], i tidigare utgåvor av Elektor och i själva Flowcodepaketet. Bäst av allt, Flowcode för ECIO är **gratis** [1].
Att skriva ett processkontrollprogram för ECIO PLC understryker allt vad Flowcode står för. I stället för att oroa dig för syntaxfel och PIC assemblerkod så arbetar du på en ganska hög nivå, sätter upp ett komplett flödesschema för programmet och låter Flowcode sköta all kompilering, felsökning och kodnedladdning till ECIO-modulen. Naturligtvis kan du simulera ditt PLC-program och det finns därför en stor chans till omedelbar succé när ECIO PLC-kortet ansluts till den egentliga världen.
Mer avancerade användare vill kanske förlita sig på sina egna metoder när det gäller att producera PIC18-kod med hjälp av C++ kompilatorer eller liknande. ECIO accepterar också ren hexkod från från alla er assemblerkodare där ute. Används helt enkelt de gratis USB I/O-drivarna för att ansluta ECIO till din PC.

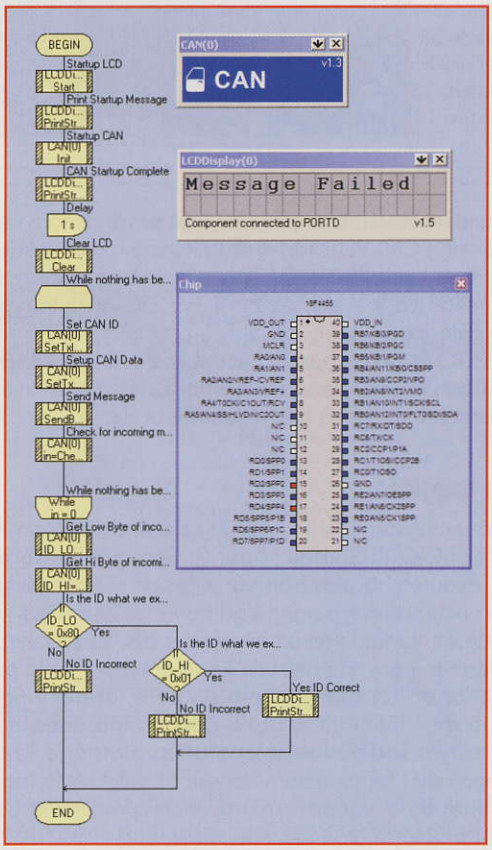
(070786-I)

CAN-interfacetestning

CAN är i huvudsak bra på att skicka komplexa meddelandestrukturer mellan ett antal ECU-er (mikrokontroller). Det testprogram som skrivits för Flowcode tittar helt enkelt efter ett eko från ett utgående CAN-meddelande. Detta kan användas för att se hur många noder det finns på nätverket eller användas för att beräkna avståndet mellan noder baserat på ekotiden etc.

I grunden sänder ECIO-testprogrammet ut ett specifikt CAN-meddelande med en standard ID på 12. När Multiprogrammet tar emot ett CAN-meddelande kollar det ID och om detta är lika med 12 återsänder det meddelande-ID 12. ECIO fortsätter att skicka ut detta meddelande-ID tills det får tillbaka ett CAN-meddelande. Så fort det har fått tillbaka ett meddelande kollar det ID och bekräftar eller förnekar sedan att ekot var korrekt.

För detta test behöver du ytterligare ett ECIO PLC-kort eller en E-block Multiprogrammerare ansluten till en E-block CAN-modul. ECIO PLC-kortet sänder flera meddelanden över en CAN-buss. Det andra kommer att lyssna tills ett fördefinierat meddelande kommer och svarar då. När CAN-sändaren startas visar displayen 'Startup' följt av 'Done' en bit senare när CAN-kontrollern initialiseras. Om CAN-anslutningen är korrekt visas 'Message returned' annars 'Message failed'.



Weblänk

[1] http://www.matrixmultimedia.com/ECIO-X.php

Produktlista

Nummer 3/2008

Datalogger deLuxe
070745-1 ..... Kretskort, obestyckat 245:00
070745-41 .... Programmerad kontroller 300:00
070745-71 .... Komponentsats (kretskort, kontroller och display) 1100:00

Hemligheten med I2C
070600-1 ..... Kretskort, obestyckat 205:00
070600-41 .... Programmerad kontroller 300:00

Cylon Voice
070859-41 .... Programmerad kontroller 75:00

ECIO PLC
070786-1 ..... Kretskort, obestyckat 245:00
070786-71 .... Byggsats (kretskort, ECIO-modul och alla komponenter) 1200:00

Nummer 2/2008

LEDBUS-system
070459-1 ..... Kretskort, effektmodul obestyckat ..... se www.alltomelektronik.se
070459-2 ..... Kretskort, huvudmodulmodul obestyckat ..... se www.alltomelektronik.se
070459-41 .... Programmerad mikro, effektmodul ..... se www.alltomelektronik.se
070459-42 .... Programmerad mikro, huvudmodul ..... se www.alltomelektronik.se

Stämningsljus med RGB
070892-1 ..... Kretskort, obestyckat ..... se www.alltomelektronik.se
070892-2 ..... Kretskort, obestyckat ..... se www.alltomelektronik.se
070892-3 ..... Kretskort, obestyckat ..... se www.alltomelektronik.se
070892-41 .... Programmerad mikrokontroller ..... se www.alltomelektronik.se

Surround Light för PC
070491-1 ..... Kretskort, obestyckat 315:00
070491-2 ..... Kretskort, obestyckat 75:00

TV Surround Light
070487-1 ..... Kretskort, obestyckat 325:00
070487-41 .... Programmerad mikrokontroller 195:00
070487-42 .... Programmerad mikrokontroller 165:00
070487-81 .... Mjukvara på CD-ROM 85:00

CAN Explorer
060201-1 ..... Kretskort, MCP2515 och MCP2551SN ..... se www.alltomelektronik.se
060201-W ..... Testing & Error Sources manual ..... se.www.alltomelektronik.se

Ringblixt med LED
070612-1 ..... Kretskort, obestyckat ..... www.ThePCBShop.com
070612-41 .... Programmerad kontroller 165:00
070612-81 .... Mjukvara på CD-ROM 85:00

Nummer 1/2008

CO_2-mätning
070802-1 ..... Kretskort, obestyckat 220:00
070802-41 .... ATtiny26, programmerad 115:00
070802-71 .... Byggsats med komponenter, kretskort, sensorkretskort, ATtiny26 och display 1625:00
070802-81 .... Mjukvara på CD-ROM 85:00

Anti-standbyswitch
070797-1 ..... Kretskort, obestyckat 215:00
070797-41 .... ATtiny25, programmerad 85:00

DC-effektmeter
070559-1 ..... Kretskort, obestyckat 145:00
070559-41 .... ATmega8-16P, programmerad 135:00

Nummer 6/2007

USB datainsamlingskort
070148-1 ..... Kretskort, obestyckat 135:00
070148-41 .... PIC18F4550 DIP40, programmerad 215:00
070148-81 .... Mjukvara på CD-ROM 85:00

Hörlursförstärkare med 3D-ljud
070393-1 ..... Kretskort, obestyckat ..... www.thePCBshop.com

Billig värmestyrning
060325-1 ..... Kretskort, obestyckat 135:00
060325-41 .... Programmerad kontroller ATmega32-16 CPU 225:00

USB Flashkort
070125-1 ..... Kretskort, obestyckat 245:00
070125-71 .... Byggsats, kretskort och samtliga komponenter 525:00
070125-81 .... Mjukvara på CD-ROM 85:00

Priser och beskrivningar kan komma att ändras.

Bästsäljare

Böcker

1 **PIC Microcontrollers**
ISBN 978-0-905705-5 ..... SEK 425:00

2 **Visual Basic** for Electronics Engineering Applications
ISBN 978-0-905705-68-2 ..... SEK 415:00

3 **Microcontroller Basics**
ISBN 978-0-905705-67-5 ..... SEK 285:00

4 **PC Interfaces under Windows**
ISBN 978-0-905705-65-1 ..... SEK 390:00

5 **Designing Audio Circuits**
ISBN 978-0-905705-50-7 ..... SEK 310:00

CD-ROM

1 **Ethernet Toolbox**
ISBN 978-90-5381-214-3 ..... SEK 275:00

2 **USB Toolbox**
ISBN 978-90-5381-212-9 ..... SEK 275:00

3 **Home Automation**
ISBN 978-90-5381-195-5 ..... SEK 210:00

4 **Elektor 2006**
ISBN 978-90-5381-207-5 ..... SEK 250:00

5 **Robotics**
ISBN 978-90-5381-179-5 ..... SEK 210:00

Satser & moduler

1 **Fristående OBD-2 läsare**
Art. # 070038-72 ..... SEK 925:00

2 **Digital Inspector**
Art. # 060092-71 ..... SEK 1100:00

3 **ElekTrack**
Art.# 040161-91 ..... SEK 3750:00

4 **Mjukvarudefinierad radio**
Art. # 070039-91 ..... SEK 1175:00

5 **Tilt-Gamepad**
Art. # 070233-91 ..... SEK 275:00

Beställ online på
www.alltomelektronik.se
eller via telefon eller fax!
Det går även bra med brev!

elektor SHOP

Allt om Elektronik
Box 178
444 22 Stenungsund
Tel. 0303-770 490
Fax 0303-770 490
Email: order@alltomelektronik.se

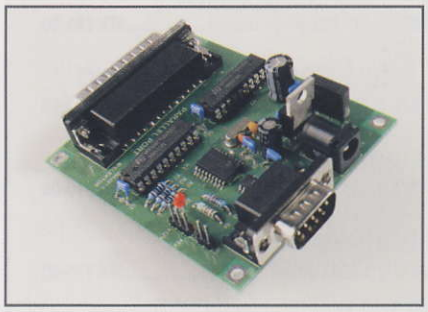
Surroundljus

Philips förser vissa av sina senaste platt-TV monster med ett färgat bakgrundsljussystem som reagerar för innehållet i bilden. Tyvärr finns detta system inte tillgängligt med andra fabrikat och inte heller för datorskärmar. I vårt februarinummer presenterar vi två system för hemmabygge som gör att liknande effekt kan uppnås. En krets är en komplett analog design, det andra digitaliserar först ingångssignalen och utför sedan all signalbehandling på digitalt vis. Projektet låter dig eftermontera en skärmkontrollerad bakgrundsbelysning till din TV eller till din datorskärm.



LED-ringblixt

Du behöver en ringblixt när du tar makrobilder med en digitalkamera. Denna enhet sprider ljuser jämt över det föremål som skall fotograferas och du slipper därmed alla fula skuggor. Professionella ringblixtar kostar en hel del och Elektor har därför utvecklat en ringblixt baserad på lysdioder och med en tillhörande drivenhet.



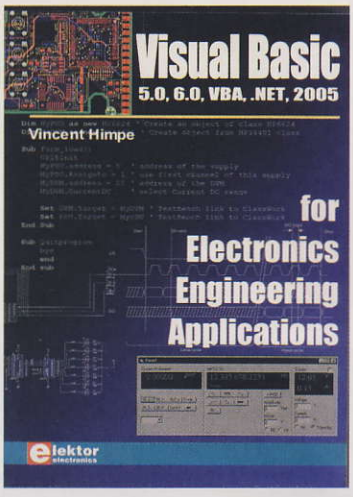
CAN Explorer

En CAN-buss är inte bara till för fordon och industriella automationssystem utan kan också användas hemma. Tyvärr är konfigurationen av CAN-bussen ingen enkel uppgift och det är var vår kontroller kommer in och etablerar en länk mellan CAN-enheter och den trogna PC-n. Med den resulterande uppkopplingen kan du, tillsammans med gratis mjukvara, ta kontrollen över en CAN-buss och testa funktionerna hos de anslutna CAN-enheterna.

Nästa nummer av Elektor utkommer den 26 februari i butik och några dagar tidigare till våra prenumeranter. Vi förbehåller oss rätten att ändra innehållet av tekniska, eller andra, skäl.

Visual Basic 5.0, 6.0 VBA, NET, 2005 for Electronics Engineering Applications.

PC-n har för länge sedan vuxit ifrån att bara vara en dator och har i stället förvandlats till en allt-möjligt-maskin. Denna bok riktar sig till er som vill kunna styra befintliga, eller hembyggda, apparater med hjälp av en dator. Genom att används Visual Basic som ett Rapid Application Tool tar vi med er på en resa bortom kontakterna på PCV-ns baksida. När du väl bekantat dig med Visual Basic, dess utvecklingsmiljö och de verktygsuppsättning som erbjuds, så går vi i detalj igenom ämnen som seriekommunikation, skrivarportar, bitbanging, protokollemulering, ISA, USB och Ethernetinterfacing och fjärrstyrning av testinstrument över GPIB-bussen. Varje ämne åtföljs av en tydlig, körbar kod och, där det är nödvändigt, kretsschemor som gör att du kan få igång dina projekt på nolltid. Denna bok visar dig avancerade saker som att använda Debug för att hitta hårdvaruadresser, sätta upp fjärrkommunikation med hjälp av TCP/IP och UDP-socklar och även hur du skriver en egen Internetserver. Eller hur du ansluter ditt eget hårdvarublock över USB eller Ethernet och styr det från Visual Basic. Även inter-programkommunikation, DDE och det nya grafikinterfacet hos Windows XP gås också igenom. Alla exemplen är klara att kompileras med hjälp av Visual Basic 5.0, 6.0, .NET eller 2005. Detaljerad information ges också om skillnaderna mellan vad som kan kallas Visual Basic Classic och Visual Basic .NET / 2005. 476 sidor, ISBN 0-905705-68-8. Engelska.
Pris 415:- inkl. moms. Porto tillkommer.
Allt om Elektronik, Box 178, 444 22 Stenungsund
Tel/Fax: 0303-770 490, www.alltomelektronik.se, order@alltomelektronik.se



Microcontroller Basics

Mikrokontroller har blivit en ovärderlig del av modern elektronik.Med dessa är det möjligt att utföra uppgifter som man tidigare bara kunde drömma om. Det finns därför alla skäl till att lära sig mer om dem, men då uppstår frågan var man hittar en bra introduktion till denn fascinerande teknologi.
Svaret är enkelt: Denna bok, Microcontroller Basics, kombinerad med det 89S8252 Flashkort som publicerats i Allt om Elektronik. Boken erbjuder dock mer än bara en introduktion. Den för-klarar tydligt teknologin med hjälp av olika mikrokontrollerprojekt och program skrivna i flera olika programmeringsspråk. Tre mikro-kontroller ur 8051-familjen används för programexemplen, från den enkla 89C2051 till AN2131, som är designad att stödja USB-applikationer. Programmeringsverktygen inkluderar assemblers, Basic-52 och BASCOM-51 samt flera C-kompilerare. Varje läsare kan således hitta den programmeringsmiljö som passar honom eller henne bäst. Vid läsning av boken ökas gradvis din kompetens att omvandla dina egna idéer till mikrokontrollerprojekt.



Alla programexemplen kan hämtas från vår eller författarens hemsida. Därmed kan du alltid hitta de senaste versionerna.

230 sidor, ISBN 0-905705-67-X. Engelska.
Pris 285:- inkl. moms. Porto tillkommer.

Allt om Elektronik, Box 178, 444 22 Stenungsund
Tel/Fax: 0303-770 490, www.alltomelektronik.se, order@alltomelektronik.se

LAWICEL
www.lawicel-shop.se

www.lawicel-shop.se

Embedded Solutions

CAN (Controller Area Network)

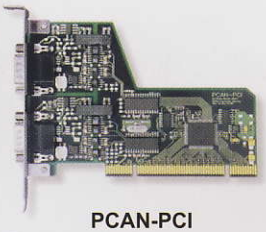
För enkla och kostnadseffektiva integreringar till 2.0A and 2.0B CAN nätverk.
Hanterar hastigheter upp till 1 Mbit/s med full busslast. 1-2 kanaler, opto isolation som option.
Drivrutiner, exempelkod och ett enkelt program för att skicka och ta emot CAN inkluderat.



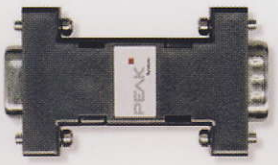
PCAN-USB
Finns i 2 versioner (med eller utan opto isolering)



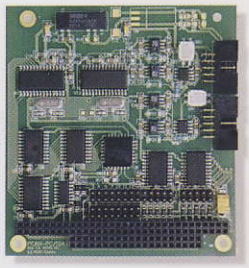
PCAN-LWL
Kör CAN på fiber i krävande miljöer eller för att verifiera/ testa CAN produkter (CE märkning)



PCAN-PCI
Finns med 1-2 kanaler (med eller utan opto isolering)



PCAN-Optoadapter
Opto adater för CAN.
För att separera två CAN nätverk galvaniskt



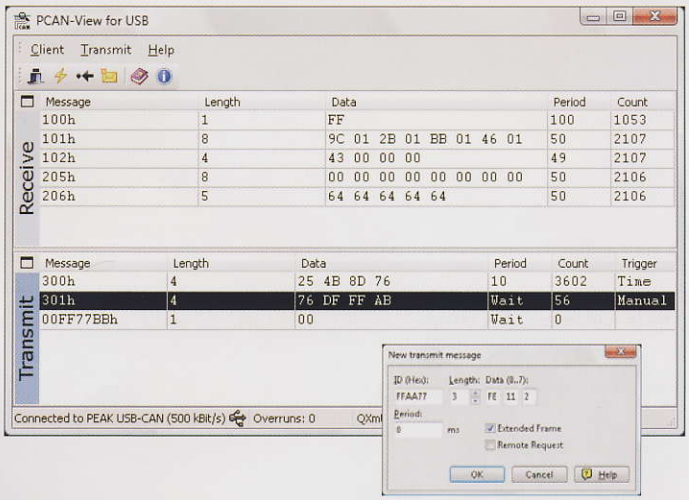
PCAN-PC/104
Finns med 1-2 kanaler (med eller utan opto isolering)



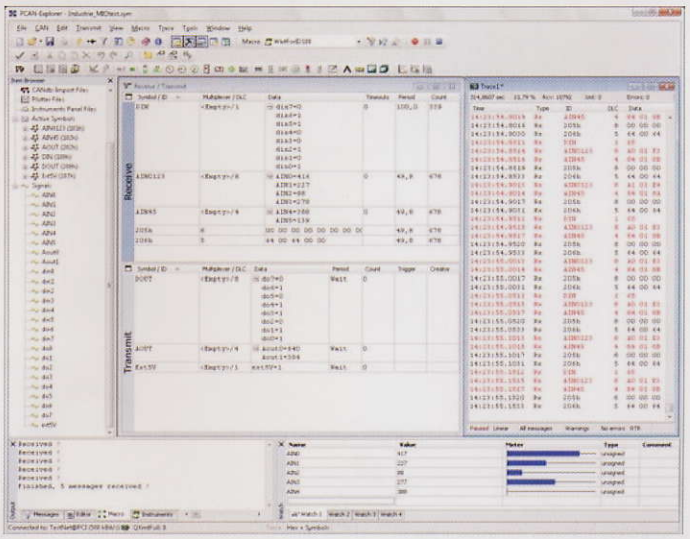
PCAN-Router
Två CAN kanaler som kan ställas in för att filtrera, behandla CAN meddelanden



PCAN-PC (PC CARD)
Finns med 1-2 kanaler (med eller utan opto isolering)



PCAN-View
Inkluderad med alla adaptrar (CAN, PCI, ISA, PC104 osv.).
Ett enkelt verktyg för att logga och sända CAN meddelanden



PCAN-Explorer
Ett avancerat verktyg för att logga CAN meddelanden, tolka CAN i riktiga ingenjörsdata istället för råa CAN meddelanden

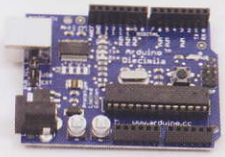
Ett urval av våra CAN produkter, fler finns i vår webshop

LAWICEL AB
Box 3, Industrigatan 6
282 21 TYRINGE

Tel. 0451-59877
Fax. 0451-59878
sales@lawicel.se

CAN - USB - ZigBee - TCP/IP
Bluetooth - RFID - LCD - Linux
AVR - ARM - MSP430 - PIC

Byggsatser - Komponenter - Tillbehör




Arduino
- AVR baserade moduler
- komplett utvecklingsmiljö i open source

Arduino är en open-source plattform för styrning och reglering via en mikroprocessor (Atmel AVR). Programmering sker i ett högnivåspråk (C). Arduino han känna av sensorer och knappar och styra t.ex motorer och belysningar. I plattformen ingår en utvecklingsmiljö för att enkelt kunna skriva styrprogram för korten. Projekten kan vara stand-alone, dvs de kör helt på Arduino-hårdvaran, eller så kan de kommunicera med en PC via serieport, USB eller Bluetooth. Besök www.arduino.cc för utförlig information, programvara och tutorials.

Beskrivning	Art.nr	Pris
Arduino Diecimila	*12200020*	**299**:-
Arduino BT (bluetooth)	*12200010*	**1150:-**
Arduino mini	*12200030*	**250**:-
Arduino USB extension (USB-serial)	*12200040*	**130**:-
Xbee shield trådlös modul (zigbee)	*12200090*	**544**:-





Smarta moduler
- Dimension Engineering

Beskrivning	Art.nr	Pris
Accelerometer 2g X-Y	*41002702*	**238**:-
Accelerometer 5g X-Y	*41002703*	**238**:-
Accelerometer 2g X-Y-Z	*41002701*	**348**:-
AnyVolt micro (3-14V in/3-14V ut)	*41002704*	**225**:-
Kopplingskort för spänningsregulatorer	*41002705*	**98**:-

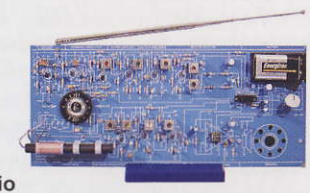



Klocka för oscilloskop
En AVR-baserad klocka som visar tiden både analogt och digitalt på ditt oscilloskop. Ansluts till oscilloskopets X-Y ingångar med vanliga 10x probar. 150-250 Hz refresh rates. Tiden och övriga inställningar kan ställas med strömställare på kortet. Ingång för extern klocksignal, 1 pps (GPS). Flyttar automatsikt bilden på skärmen för att förhindra att bilden bränner in. RS-232 analsutning för bl.a uppgradering av firmware och extern styrning.
Kretskortet är uppbyggt kring en AVR ATMega168, TLC7528 DAC samt en PCF8563 realtidsklocka med batteribackup.
Matningsspänning: 5V. Byggsats.
41002365 **229:-**

Jumper wire - 0.64mm
Labsladd med 0.64mm hona i båda ändar. Lämplig för att koppla samman 0.64 mm stiftlister.

41002665	13cm	**3:-**
41002666	23cm	**4:-**





AM/FM Radio
En pedagogiskt uppbyggd radio för AM och FM. Varje steg i mottagaren är tydligt utmärkt på kretskortet, och den 52-sidiga manualen (engelska) ger en genomgående beskrivning av funktionen. Byggsats.
10110108 **379:-**



Apparatlådor
Ett prisvärt urval av apparatlådor till ditt bygge. Plast och metall - ca 50 olika varianter. Besök vår hemsida för detaljer.

Plastlåda - utförande i slät svart plast med över och underdel. De större lådorna har upphöjningar för kretskort.

Typ	*Yttermått b x d x h*	*Art.nr*	*Pris*
FB01	60x65x25 mm	12109001	15:-
FB02	36x118x21 mm	12109002	18:-
FB03	69x105x38 mm	12109003	22:-
FB04	89x34x145 mm	12109004	28:-
FB05	103x150x67 mm	12109005	33:-
FB09	130x195x66 mm	12109009	55:-
FB11	32x47x25 mm	12109011	10:-
FB17	63x115x23 mm	12109017	29:-
FB19	27x48x10 mm	12109019	19:-
FB20	50x76x20 mm	12109020	33:-



Instrumentlåda i plast, med överdel, underdel, frontpanel och bakpanel.

Typ	*Yttermått b x d x h*	*Art.nr*	*Pris*
FB06	215x168x78 mm	12109006	139:-
FB10	146x157x67 mm	12109010	89:-
FB16	140x110x42 mm	12109016	65:-

Lödstation 48W med temperaturjustering
En lödstation med lödpenna, lämplig för elektronikarbeten. Temperatur inställbar 150-450 C. 220V AC. Jordad spets.
41000870 **268:-**

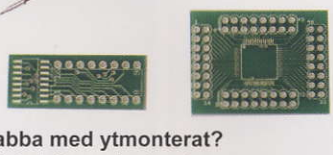




Oscilloskop
10 MHz, X-Y Ett bra oscilloscop för hobbybruk.
41001068 **1.395:-**



Lödstation - mini 14W
En temperaturkontrollerad lödstation med miniatyrlödpenna, lämplig för små lödställen och handlödning av SMD komponenter. Temperatur inställbar upp till 375 C. ESD-safe. 220V AC.
41000422 **395:-**



Labba med ytmonterat? Då behövs adapters!
En serie adapterkort så att ytmonterade kretsar kan monteras på kopplingsdäck och experimentkort avsedda för hålmonterat. Den ytmonterade kretsen lödd fast på adapterkortet, och via stiftlist ansluts adapterkortet till det hålmonterade kortet.

41000907	SO-DIL8	**10:-**
41000908	SO-DIL14	**14:-**
41000909	SO-DIL16	**15:-**
41000910	SO-DIL20	**18:-**
41000911	QFP44 - 0.80 mm	**20:-**
41000912	QFP64 - 0.50 mm	**23:-**
41000913	QFP80 - 0.50 mm	**25:-**
41002360	Stiftlist m svarvade ben, 40-pol brytbar	**18:-**

electro:kit

Tel: 040-298760

www.electrokit.se

Moms ingår. Frakt tillkommer - från 29:-.
Se hemsida för detaljer.
Reservation för ev fel o ändringar.



Robotbyggsatser
Ny serie robotbyggsatser från Lynxmotion.

Lynx 5 combo kit	*41002411*	**2690:-**
Lynx 6 combo kit	*41002412*	**3590:-**
Johnny 5 combo kit	*41002498*	**10690:-**

- se hemsidan för utförlig beskrivning

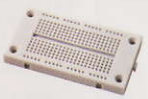
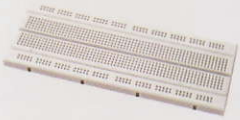
I vårt sortiment finner du över 250 olika byggsatser
DIY, Velleman, Future Kit, GTI, Elenco
För Dig som tycker att elektronik är kul!



Ringsignaldetektor
Känner av när telefonen ringer, och då blinkar en ultrastark LED. Dessutom aktiveras en reläutgång som du kan ansluta valfri utrustning till. Byggsats inkl låda.
12008086 **159:-**



Oscilloskop AT-7328
2 kanaler, 20 MHz, X-Y
Ett bra standardoscilloskop med två kanaler (alt/chop), X/Y, flera triggmöjligheter mm.
41001297 **3.395:-**

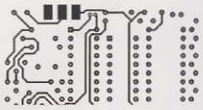
Kopplingsdäck
Lödfria kopplingsdäck för experimentuppkopplingar.

270 anslutningar	*10160270*	**29:-**
840 anslutningar	*10160840*	**59:-**



Trådbyglar
Färdigskalade trådbyglar lämpliga för kopplingsdäck. Totalt 140 st i olika längder, 3.5--125mm. Levereras i praktisk plastask.
41002700 **33:-**

SUCCÉ



PCB-service
Nu kan du beställa dina egna mönsterkort via oss. De tillverkas i modern svensk fabrik - och leveranstiden är endast 5-7 arbetsdagar (10-12 för 4-lager).
Standardutförande är FR4 (glasfiber) 1.6 mm, CU 18/18 (35/35) inkl blyfri varmförtenning, grön lödmask, vitt komonenttryck på en sida, UL-märke och eltest. Du kan beställa så lite som ett kort, och vill du ha flera är följandepriset mycket förmånligt.

Se www.electrokit.se för vidare information och priskalkyl. Begär gärna offert för större antal.