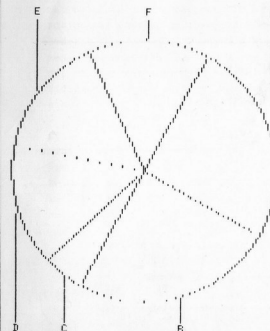


Class PR ex

PROGRAM BITEN 92-4

A: GOD 25.0%
B: JUL 25.0%
C: OCH 5.0%
D: GOTT 15.0%
E: NYTT 15.0%
F: AR 15.0%



Innehåll

Egen modul	3
Bättre rengöring av skrivhuvudet	3
Utmaningen	4
Cirkeldiagram 4; Registerblockhantering	
6; Flera pgm på ett kort 6; INV Pgm 20	
8; Problem 9	
Graphics Mode och Fast Mode - Plot 60-II	10
Tips om hembyggen	12
Periodiska systemet	14
... stryk i Nim för sista gången	17
Van der Waals gaslag	18
Budget	20
Avancerade programmeringsmetoder - 3	22
TI Programmer	23
Beräkning av kontrollsiffra	24
Bokföring	26
Kortslutet batteri	27
Hexkoder	28
Utländska klubbar	29
Romerska siffror	30



Dyre medlem!

Så är vi igen vid slutet av en årgång. Programbitens fjärde. Fr o m den femte årgången kommer vi att ha en ny redaktör, den hittillsvarande programföredragaren Bo Nordin, som jag härmed (jag förutsätter årmötets medgivande!) hälsar välkommen till hans nya syssla och lämnar över ansvaret för medlemsbladet till.

Dessvärre har Björn Gustavsson, vår förtruffade Utmaningsredaktör, sagt att detta nummers Utmaning är hans sista. Där är det värre med ersättare, men lät oss ändå hoppas att någon intresserad medlem nu rycker in och tar hand om denna hittills utomordentliga avdelning.

Den 29 januari har vi årsmöte, och då får vi diskutera utgivningsplanerna för medlemsbladet. Tyvärr bryter redaktionen faktiskt mot årsmötetsbeslutet i detta nummer som bara har 32 sidor mot normalt 40, men jag anser att det inte varit möjligt att åstadkomma mer om tidningen skulle ut till jul. Jag vill därför föreslå att årsmötet uttryckligen ger den nye redaktören frihet att bestämma medlemsbladets sidantal. Det är ju inte särskilt angenämt för den som frivilligt åtar sig att vara redaktör att ha en piska över sig i form av ett årsmötetsbeslut att medlemsbladet skall ha 40 sidor. Ibland är tillgången på material bättre, ibland sämre, och redaktören måste få känna medlemmarnas förtroende att han gör vad han kan inom ramen för tillgängligt material och sin egen tillgängliga tid.

Och här kommer ju den synpunkt in som många gånger fört framförts i den här spalten: Vi är en förening och inget tidskrifts-förlag. I en förening byter man erfarenheter medan tidskriftsköparen eller prenumeranten inte har något ansvar utöver betalningen. Givetvis kan vi på andra sidan inte ta in bidrag från alla medlemmar, inte ens om vi saknat det utländska materialet, men det är känslan av förening jag vill betona och att man alltså inte kan "kräva" stort sidantal om man inte själv är aktiv med bidrag till medlemsbladet.

Å andra sidan är utsikterna inför 1983 spännande. TI-88 tycks visserligen knappt överleva födelsen, men i stället verkar det finnas en basiräknare med många intressanta drag som väntar på att introduceras. Om den motsvarar förväntningarna bör den säkert passa in på en sådan användargrupp som vi och alltså rimligen avhandlas i Programbiten. F n finns inga nya uppgifter åt få men vi hoppas återkomma i nästa nummer.

Teckningstävlingen som utlystes i förra numret har vi inte hunnit få några bidrag till så därför har vi tillgripit ett annat "(h)jultmotiv" (oj!). Ursprungligen planerade vi att ha flera spel som skulle ha passat i ett julnummer, t ex "Lärare i Black Jack" där dock ett fel upptäckts trots att den programmerats av de berömda bröderna R' C' Snow, ett damspel som ännu är under utveckling och ett spel "Minifält" där man med "minidetektor" och logik skulle ta sig över en spelplan med där exekveringstiden per drag var ca 45 s medan det alldeles bestämt skulle gå att få ner den till ett par sekunder. Alla de här bidragen får alltså stå över. Trots det hoppas vi här erbjuda ett intressant nummer bl a med hjälp av vår nye holländska medlem Robert Prins.

Ett tack till alla medarbetare under 1982 och särskilt till Björn Gustavsson som lämnat bidrag i absolut särklass både i fråga om antal och kvalitet. Vi hoppas få läsa nya bidrag av Er under 1983.

God jul och Gott Nytt År!

Lars Hedlund

Lars Hedlund ordf

FÖRENINGENS

TILLBEHÖRSFÖRSÄLJNING

Följande finns att köpa för medlemmar genom att motsvarande belopp insätts på postgiro 430 01 59 - 3:

Programbiten, årgång 1982	100,-
"-"- 1981	80,-
"-"- 1980	60,-
"-"- 1978/79	60,-
"-"- fyra årgångar 1978/79 - 1982	250,-
Patenthandlingar TI-59	25,-
Katalog med belgiska och engelska program	20,-
Astronomical Formulae for Calculators (viss leveranstid)	55,-
40 st lomma magnetkort och plånbok	130,-
(OBS HÖJT PRIS!)	10,-
Tom magnetkortspånbok	10,-
Föreningens programmeringsblanketter, olika typer, block om 50 blanketter (se Pb 79-364)	11,-
Sourcebook är slutsåld.	

Årsmöte

Lördagen den 29 januari 1983 kl. 13.00

i "Dagens Träffcentrum" vid Sandsborgs T-banestation vid Stockholms T-linje 18. Ingång i stora hallen, vår lokal ligger i tr upp.

På dagordningen står vanliga årsmötessändren, dvs ansvarsfrihet och styrelseval och val av ny redaktör. Dessutom bör vi diskutera utgivningen av medlemsbladet under 1983.

En motion har inlämnats med förslag om att programbiblioteket skall ersättas med en annonsavdelning i Programbiten. Styrelsen föreslår avslag med hänsyn till att möjlighet till "engångsinförande" finns för den som själv vill marknadsföra sina program.

Ärende som medlem önskar få behandlat på årsmötet skall vara styrelsen tillhanda senast den 11 januari 1983.

Efter de formella förhandlingarna fortsätter vi med

Medlemsträff

i samma lokaler fram till kl. 18.00. Ev.chans i samma lokal medlemmar även från utom Skandinavien.

BÄTTRERENGÖRING AV SKRIVHUVUDET

AV MIKAEL NORDLIN

Det belgiska skrivarpappret "TP 30250" till FC-100 har fått många lovord för sin fina svärta, som ger en god kontrast. Med rätta, men en egenhet tror jag mig ha upptäckt: det smutsar ner skrivhuvudet mer än det tidigare (franska) pappret, med samma beteckning.

Efter listning av några långa program blir utskriften allt blekare och otidigare, vissa punkter i teckenmatrisen försvinner helt och hållet. Rengöring av skrivhuvudet kan därför vara nödvändigt betydligt oftare än i instruktionsboken föreskrivna en gång per rulle.

Olika program har presenterats i Programbiten för att bistå rengöring av skrivhuvudet. Här följer ett tips, som ger en utomordentligt effektiv rengöring på 0 programsteg. Man förfär sålunda: På apoteket inköpes en vätska benämnd "Iso-

Egen modul!

Den här rubriken var en följetong i många nummer, men i de senaste har den inte förekommit så ofta och i så fall varit kortfattad.

Vad som nu skulle kunna påverka saken är att man i kretsarna kring TI PPC Notes i USA anser TI-88 definitivt borta. Palmer Hanson (den nye redaktören) berättar i ett brev att han fått en TI-affisch från Dave Leising (en av medlemmarna). Affischen föreställde en TI-88 och i displayen hade det stått "May I help you?". Dave hade skrivit dit "I do not exist".

Palmer Hanson tar därför upp frågan om det inte är dags att ta upp vårt gamla modulprojekt igen. Det behövs bara 250 intresserade (varav förstås en del som är så intresserade att de vill jobba med projektet - inte bara köpa modulen!) och för den som gillar programmerbara räknare som TI-59 mer än basic-räknare är oenigheten projektet intressant. En hel del förarbete har ju ändå gjorts.

Om Palmer får respons från sin förening kan alltså något börja hända. På tar förstås igen gärna emot idéer och synpunkter i ämnet samt meddelanden om intresse.

Robert Prins meddelar f o att han har Funderingar på hur en Eprom skulle kunna göras större än 10.000 steg, ändå upp till tio block om 10.000 steg! Han har vissa kontakter med de avancerade hårdvarupojkarna och alldeles säkert funderas det en hel del runt de här frågorna. ■

propanol". Denna lämpar sig utmärkt för rengöring även av bandspelarlathuvudet, i och med att den löser upp föroreningar utan att angripa vare sig metall eller gummi. "Audiofiler" har använt den länge i det sammanhanget, varför den borde lämpa sig utmärkt för "hackers" med FC-100, då materialet är de samma. Man dränker därför in en rengöringsremsa, som medföljer pappret, med isopropanol och för in den på papperets slitt i skrivarens pappersframmatning. Sedan trycker man in Adv på skrivaren och håller den intryckt tills hela remsan gått igenom. Om skrivhuvudet därefter får stå och torka några minuter i uppfällt läge, slipper man misfärgning av skrivpappret från isopropanolen. Därefter är skrivarens utskrift som ny! Jag har prövat metoden ett bra tag, och inte kunnat upptäcka några negativa biverkningar. Fritt fram således. ■

UTMANINGEN!

Redaktörens adress:
Björn Gustavsson
Nordlandervägen 12 A
770 00 SNEDEBJÄCKEN

lista över skrivor och procental. - Om du råkar läsa in fel block ("2.2") är ingen skada skedd. Varje block har ID-nummer och programmet kommer att visa "2.1" igen och du kan läsa in rätt block.

10. När listan är utskrivnen visas "2.2". Läs in block "2.2" och utskriften startar. (När block 2.1 har lästs i steg 9 kan du direkt sätta in block 2.2 i kortläsaren.) Du har nu åtminstone en timme på dig för att dricka en kopp kaffe eller varför inte stada bort alla gamla skrivarensor som ligger och dräller i hela rummet?

11. Om den färdiga utskriften är antingen för stor eller för liten kan du helt enkelt mata in antal resor på nytt och trycka A. Programmet frågar då efter block 2.1 och efter att ha skrivit listan efter block 2.2. Därefter fås en ny utskrift.

12. För att skriva ut ett helt nytt cirkeldiagram, tryck RST och gå tillbaka till steg 4.

Figur 1 visar hur ett cirkeldiagram kan se ut.

R: GOD 25.0%
S: JUL 25.0%
C: OCH 5.0%
T: GOTT 25.0%
E: NYTT 15.0%
F: RÅK 15.0%

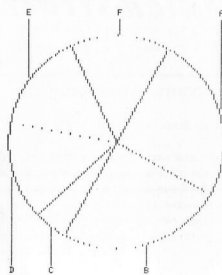
I förra Utmaningen utlovade jag en utökad version av Robert Prins cirkeldiagramprogram. Här är den!

Programmet använder både Graphics och Fast Mode. Fast mode initieras med STF h2 sekvensen och med väl vald initieringskonstant: 1 "X" x² 1/x så att man kan återvända till normal mode utan att TRACE ställs in på skrivaren. (Se PB 82-1 s 18-20.)

Bruksanvisning för pgm 1:

1. Se till att standardmodulen finns på plats.
2. Läs in block 1 & 3.
3. Skapa de två hexoderna med följande sekvens: 10 Op 17 CLR OT6 P16 9N SBR 045 P/R LRN IN (dessa skapar h25) SST (16 gånger till steg 032) IN (dessa skapar h12) LRN RST CLR.
4. Mata in antal resor som skall användas och tryck A. Om du har väldigt gott om tid kan du välja A fler än 2 resor.
5. Mata in antal sektorer i cirkeldiagrammet och tryck R/S. Högsta tillåtna antal är 20.
6. Mata in sektorn för den första sektorn och tryck xit. (Detta steg kan hoppas över om du inte vill mata in någon skrivid. T-registret är nollställt.)
7. Mata in procenttalet för sektorn och tryck R/S. Procenttalet kommer att avrundas till närmaste 0.5%.
8. Upprepa steg 6 och 7 för alla sektorer.
9. När alla sektorer är inmatade visas "2.1". Läs in blocket märkt "2.1". Programmet startar automatiskt och kommer att först testa alla vinklar (procental) och sedan att skriva en

Figur 1. (Robert Prins.) Cirkeldiagram. Inmatning: Andra uppdatering till 4 Op 17 och mata in steg 000-827 enligt den första listningen. Spela in block 2 på "baksliden" (kortare andra spår) av ett magnetkort och märk det "2.1". Andra uppdatering till 6 Op 17 och spela in block 1 & 3 på samma magnetkort. Andra uppdatering in en gång till 4 Op 17 och mata in steg 240-478 enligt den andra listningen. Spela in block 2 på det återstående spåret ("framsliden") av det första kortet och märk det spåret "2.1". Du har nu två inspelade magnetkort: det ena märkt "2.1" och "2.2" inspelat i 4 Op 17 och det andra märkt "1" och "2" inspelat i 6 Op 17.



000 01 1	160 52 17	240 89 4	320 05 05	400 77 GE	480 43 RCL	560 07 07
001 76 LBL	081 05 5	161 02 2	241 32 17	321 05 0	401 40 INE	481 01 1
002 40 09	082 45 0	162 02 2	242 07 07	322 03 03	402 22 INV	482 01 01
003 40 09	083 45 0	163 02 2	243 07 07	323 03 03	403 22 INV	483 01 01
004 40 09	084 45 0	164 02 2	244 07 07	324 03 03	404 22 INV	484 01 01
005 40 09	085 45 0	165 02 2	245 07 07	325 03 03	405 22 INV	485 01 01
006 40 09	086 45 0	166 02 2	246 07 07	326 03 03	406 22 INV	486 01 01
007 12 18	087 01 01	167 01 01	247 32 52	327 34 +/-	407 01 1	487 02 17
008 01 18	088 01 01	168 01 01	248 01 01	328 34 +/-	408 01 1	488 02 17
009 00 00	089 00 00	169 00 00	249 01 01	329 34 +/-	409 01 1	489 02 17
010 18 18	090 01 01	170 01 01	250 01 01	330 34 +/-	410 01 1	490 02 17
011 76 LBL	091 42 10	171 05 5	251 68 17	331 33 34	411 33 34	491 15 15
012 13 13	092 42 10	172 05 5	252 68 17	332 33 34	412 33 34	492 15 15
013 25 RCL	093 42 10	173 42 10	253 07 07	333 43 RCL	413 43 RCL	493 22 10
014 40 09	094 42 10	174 42 10	254 07 07	334 43 RCL	414 43 RCL	494 22 10
015 05 05	095 01 1	175 09 9	255 01 01	335 33 34	415 38 38	495 40 10
016 76 09	096 01 1	176 09 9	256 03 03	336 34 +/-	416 42 10	496 15 15
017 90 90	097 42 10	177 42 10	257 53 53	337 01 1	417 17 17	497 53 53
018 12 18	098 42 10	178 42 10	258 07 07	338 34 +/-	418 33 34	498 15 15
019 40 09	099 53 53	179 01 1	259 02 02	339 77 GE	419 54 54	499 53 53
020 40 09	100 42 10	180 12 18	260 03 03	340 40 40	420 77 GE	500 46 46
021 30 10	101 00 00	181 69 69	261 53 53	341 15 15	421 40 40	501 65 65
022 41 07	102 00 00	182 69 69	262 02 02	342 43 RCL	422 15 15	502 04 04
023 53 53	103 01 1	183 02 02	263 06 06	343 17 17	423 22 INV	503 85 85
024 32 10	104 09 9	184 12 18	264 09 09	344 43 RCL	424 17 17	504 85 85
025 93 93	105 54 54	185 13 13	265 42 10	345 04 04	425 03 03	505 75 75
026 93 93	106 27 27	186 13 13	266 54 54	346 15 15	426 03 03	506 54 54
027 34 17	107 05 05	187 05 05	267 05 05	347 77 GE	427 43 43	507 42 10
028 34 17	108 01 01	188 01 01	268 05 05	348 18 18	428 03 03	508 22 10
029 35 17	109 46 46	189 35 35	269 43 RCL	349 33 34	429 43 RCL	509 54 54
030 35 17	110 02 02	190 35 35	270 43 RCL	350 33 34	430 43 RCL	510 54 54
031 35 17	111 02 02	191 35 35	271 43 RCL	351 02 02	431 10 10	511 02 02
032 35 17	112 02 02	192 35 35	272 43 RCL	352 02 02	432 10 10	512 02 02
033 03 03	113 05 05	193 42 10	273 34 34	353 05 05	433 43 RCL	513 54 54
034 03 03	114 05 05	194 42 10	274 34 34	354 05 05	434 43 RCL	514 54 54
035 97 52	115 33 33	195 00 00	275 43 RCL	355 01 1	435 05 05	515 54 54
036 74 74	116 33 33	196 00 00	276 17 17	356 42 10	436 69 69	516 94 +/-
037 74 74	117 33 33	197 12 18	277 43 RCL	357 15 15	437 10 10	517 43 43
038 00 00	118 65 65	198 13 13	278 04 04	358 43 RCL	438 54 54	518 07 07
039 00 00	119 65 65	199 13 13	279 04 04	359 43 RCL	439 54 54	519 07 07
040 49 09	120 95 95	200 04 04	280 01 1	360 42 10	440 67 67	520 65 65
041 17 17	121 95 95	201 04 04	281 01 1	361 42 10	441 67 67	521 65 65
042 47 17	122 00 00	202 05 05	282 44 44	362 01 1	442 15 15	522 54 54
043 04 04	123 00 00	203 05 05	283 44 44	363 01 1	443 15 15	523 54 54
044 00 00	124 25 25	204 00 00	284 43 RCL	364 42 10	444 05 05	524 59 59
045 49 49	125 25 25	205 00 00	285 43 RCL	365 42 10	445 05 05	525 59 59
046 47 05	126 25 25	206 96 96	286 94 +/-	366 80 80	446 46 46	526 01 01
047 01 1	127 01 01	207 96 96	287 94 +/-	367 80 80	447 17 17	527 01 01
048 93 93	128 00 00	208 42 10	288 04 04	368 65 65	448 61 61	528 97 97
049 93 93	129 00 00	209 42 10	289 04 04	369 65 65	449 61 61	529 97 97
050 05 05	130 28 28	210 02 02	290 05 05	370 05 05	450 76 76	530 02 02
051 05 05	131 28 28	211 02 02	291 05 05	371 05 05	451 76 76	531 02 02
052 42 10	132 69 69	212 13 13	292 44 44	372 52 52	452 18 18	532 48 48
053 42 10	133 69 69	213 13 13	293 44 44	373 52 52	453 18 18	533 48 48
054 02 2	134 73 73	214 05 05	294 43 RCL	374 07 07	454 67 67	534 05 05
055 02 2	135 73 73	215 05 05	295 43 RCL	375 07 07	455 67 67	535 05 05
056 01 1	136 97 97	216 12 12	296 40 40	376 37 37	456 80 80	536 32 32
057 42 10	137 18 18	217 69 69	297 18 18	377 37 37	457 80 80	537 32 32
058 02 02	138 01 01	218 00 00	298 23 23	378 23 23	458 69 69	538 22 10
059 02 02	139 01 01	219 13 13	299 09 09	379 23 23	459 69 69	539 22 10
060 00 00	140 02 02	220 02 02	300 42 10	380 03 03	460 00 00	540 05 05
061 02 02	141 02 02	221 42 10	301 03 03	381 99 99	461 48 48	541 48 48
062 42 10	142 58 58	222 05 05	302 30 30	382 54 54	462 10 10	542 39 39
063 24 24	143 58 58	223 01 1	303 39 39	383 39 39	463 09 09	543 39 39
064 06 06	144 58 58	224 01 1	304 39 39	384 39 39	464 09 09	544 39 39
065 06 06	145 58 58	225 01 1	305 39 39	385 39 39	465 09 09	545 39 39
066 42 10	146 58 58	226 01 1	306 39 39	386 39 39	466 09 09	546 39 39
067 42 10	147 58 58	227 43 RCL	307 43 RCL	387 54 54	467 11 11	547 39 39
068 42 10	148 58 58	228 05 05	308 05 05	388 05 05	468 11 11	548 39 39
069 42 10	149 58 58	229 44 44	309 44 44	389 17 17	469 03 03	549 07 07
070 49 49	150 61 61	230 98 98	310 03 03	390 67 67	470 48 48	550 52 52
071 49 49	151 61 61	231 98 98	311 03 03	391 67 67	471 48 48	551 52 52
072 49 49	152 61 61	232 98 98	312 03 03	392 67 67	472 48 48	552 52 52
073 49 49	153 61 61	233 98 98	313 03 03	393 67 67	473 48 48	553 52 52
074 49 49	154 61 61	234 98 98	314 03 03	394 67 67	474 48 48	554 52 52
075 02 02	155 93 93	235 75 75	315 75 75	395 46 46	475 52 52	555 33 33
076 49 49	156 93 93	236 98 98	316 15 15	396 46 46	476 52 52	556 33 33
077 49 49	157 93 93	237 98 98	317 15 15	397 46 46	477 52 52	557 33 33
078 49 49	158 93 93	238 98 98	318 15 15	398 46 46	478 52 52	558 33 33
079 04 04	159 96 96	239 98 98	319 43 RCL	399 54 54	479 75 75	559 43 RCL

Ovan: Pgm 1, listning 1.

Medan: Pgm 1, listning 2.

Se vidare text längst ned på noterade sida.

NÅGOT OM PROGRAMMETS UPPBYGGNAD

Programmet består av följande block:

Block 1 (steg 000 - 239): Används för initiering och inmatning av procenttal.

Block 2.1 (steg 240 - 478): Kontrollerar alla inmatade procenttal och skriver ut listan över namn och procenttal.

Block 2.2: Början av cirkeldiagramskriften.

Block 3 (steg 480 - 627): Resten av utskrifts-rutinen.

Procenttalen omvandlas omedelbart till vinklar vid inmatningen genom en enkel multiplikation med 2. Vinklarna fås då i nygrader (gon). Det är den första gången jag har sett någon ha någon nytta av nygrader. När block 2.1 läses in testas alla vinklar. Vinklar från 90 till 1710 och 290 till 310 (1 nygrader) är otillåtna, eftersom Plot 60 inte kan plotta horisontella linjer. När en sådan vinkel hittas roteras samtliga vinklar 8 nygrader och testas igen. Om det efter 50 rotationer, dvs ett helt varv, fortfarande finns otillåtna vinklar stoppas programmet och niorna blinkar.

REGISTERBLOCKHANTERING (25)

Problem 25 gick ut på att göra subrutiner för registerblockhantering. Erik Kempf, Odense, Danmark har skickat in två lösningar. Den första återges här som pgn 3. Subrutinen är mycket allmän. Genom att endast ändra ett steg i den kan man a) Radera ett registerblock, b) kopiera registerblock med nollställning av ursprungsblock, c) kopiera registerblock, d) Skifta två registerblock, c och d efterfrågas i problemformulering; a och b kommer på köpet.

Bruksanvisning:

Programmet hanterar registerblock $R_m - R_n$ och $R_m - R_n$ där $n < 0$. Användaren ansvarar själv för att registerblocken inte överlappar varandra. p skall lagras i R_0 och vid anrop skall m finnas i T-registret och n i displayen.

a) Nollställning av registerblock $R_m - R_n$: Sätt p q högsta registernummer, t ex 100.

b) Kopiering av registerblock $R_m - R_n$ till $R_m - R_n$ med nollställning av $R_m - R_n$: Inga särskilda åtgärder.

c) Kopiering av $R_m - R_n$ till $R_p - R_q$: Byt ut steg 11 till RCLInd.

d) Byta innehåll mellan $R_m - R_n$ och $R_p - R_q$: Byt ut steg 10 till Nop. (Steg 11 får inte ha ändrats enligt c ovan. Ändra i så fall tillbaka till ExciInd på steg 11.)

Om vi i ett program behöver både operationerna c och d kan man antingen ha två versioner av subrutinen i programmet eller också skriva om subrutinen så att man kan välja operation genom att ställa en flagga. Den första metoden ger ett snabbare program vilket tar upp mer programminne, medan den andra metoden ger ett något långsammare program som tar upp mindre programminne. (Eriks andra version valde operation med hjälp av två flaggor.)

```
000 74 LBL 000 00 00 016 48 EXC 024 18 18
001 11 R 009 32 XIT 017 00 000 025 77 77
002 65 HIR 010 00 0 018 32 XIT 026 11 11
003 09 09 011 63 EXC 019 72 9T 027 24 24
004 74 RCL 012 00 00 020 00 00 028 00 0
005 00 00 013 50 00 021 60 00 029 92 PTH
006 32 XIT 014 20 20 022 00 00 030 00 0
007 48 EXC 015 32 XIT 023 82 HIR
```

Figm 2. (Erik Kempf.) Registerblockhantering.

FLERA PROGRAM PÅ ETT KORT (33)

Till problem 33 som gällde att lagra flera program på ett kort har en lösning från Lars Nilsson inkommit. Men eftersom hans system kräver ett extra magnetkort för att välja program och skrivare är nödvändig för att definiera startadresserna tar jag här i stället med ett enklare program för jag själv gjort.

För att lagra programmen läser man först in pgn 3. Man matar sedan in sina egna program på stegen 089-399, ut att använda Lbl A - Lbl E, Lbl A' - Lbl C' eller Lbl E' (Lbl D' får användas.) Maximalt fyra program får matas in.

```
000 69 DP 022 45 45 044 23 23 046 14 2
001 99 99 023 59 10T 045 00 0 047 81 05
002 32 PTH 024 94 4+ 046 10 E' 048 25 25
003 74 LBL 025 03 00 047 34 1E 049 30 30
004 71 SBR 026 00 00 048 34 PTH 050 15 1E
005 49 DP 027 45 10T 049 10 E' 051 81 05
006 16 16 028 03 3 050 18 C' 052 64 24
007 79 7 029 42 S10 051 81 05 053 18 C'
008 06 6 030 28 L05 052 64 24 054 14 A'
009 01 01 031 44 PTH 053 18 C' 055 81 05
010 42 S10 032 00 00 054 11 A 056 18 C'
011 00 00 033 44 PTH 055 81 05 057 81 05
012 00 0 034 60 40 056 18 C' 058 18 C'
013 63 DP 035 05 05 057 81 05 059 18 C'
014 17 17 036 74 98+ 058 18 C' 060 50 50
015 42 S10 037 00 00 059 18 C' 061 50 50
016 42 S10 038 05 30 060 50 50 062 10 E'
017 42 S10 039 05 30 061 50 50 063 81 05
018 00 00 040 73 RCL 062 10 E' 064 69 69
019 42 40 041 00 00 063 81 05 065 18 C'
020 00 00 042 00 00 064 69 69 066 18 C'
```

Figm 3. (Rad.) Flera program på ett kort. Programul.

Pgm 5 är ett exempel på två program som har matats in. Det första (steg 085-154) beräknar fakultet och det andra räknar med bråk (steg 157-237) och är hämtat ur PB 80-4. (För den som inte har det numret: Bråk matas in som täljare x:t nämnare. A' förkortar ett bråk. A matar in första bråket i en beräkning. B adderar bråket i X- och T-registret med tidigare inmatat bråk. C subtraherar. D multipliserar. E dividerar.)

I listningen har även markerats startadresser för de användardefinierade lablarna. Dessa sammanställs till följande tabell:

```
E' 157 085
A 214 152
B 223 -
C 220 -
D 233 -
E 234 -
A' 167 -
B' -
C' - 131
```

Efter att ha spelat in pgn 5 läser man in pgn 4: definitionsprogram. Bruksanvisningen för det programmet är följande:

1. Initiera med E. 10 visas.
2. Mata in adressen för labeln vars kod visas i displayen och tryck R/S. Om labeln inte finns behöver du bara trycka R/S.
3. Upprepa detta steg tills du har definierat alla lablar.
4. För att definiera lablarna för nästa program: Tryck A och gå till steg 2.
5. För att spela in på magnetkort när alla program är definierade: Tryck B och stoppa in kortet.

Lablarna i pgn 5 definieras på följande sätt:

```
000 74 LBL 0205 09 9 052 35 35 078 12 B
001 15 E 027 42 S10 053 16 LBL 079 14 A'
002 25 CLR 028 00 00 054 16 A' 080 49 DP
003 74 LBL 029 42 S10 055 43 RCL 081 34
004 69 DP 030 01 01 056 05 05 082 43 RCL
005 17 17 031 66 4 057 32 XIT 083 04 04
006 47 CMS 032 09 9 058 43 RCL 084 22 HIR
007 42 S10 033 42 S10 059 02 02 085 28 L05
008 05 5 034 02 02 060 77 7E 086 35 1X
009 03 3 035 43 RCL 061 00 00 087 45 A'
010 22 HIR 036 05 45 062 65 05 088 33 A'
011 38 L05 037 64 PTH 063 42 S10 089 44 A'
012 35 1X 038 02 02 064 05 05 090 95 +
013 02 02 039 42 S10 065 42 S10 091 22 S10
014 03 03 040 21 21 066 03 03 092 60 60
015 01 01 041 09 DP 067 43 RCL 093 22 S10
016 11 A 042 01 01 068 02 02 094 05 05
017 05 05 043 09 DP 069 03 03 095 06 6
018 24 24 044 34 PTH 070 32 32 096 49 DP
019 03 03 045 05 05 071 87 882 097 17 17
020 00 00 046 02 02 072 00 00 098 02 02
021 47 47 047 69 07 073 00 00 099 56 56T
022 67 CMS 048 32 32 074 67 67 100 00 0
023 42 S10 049 00 00 075 00 00 101 92 PTH
024 26 26 050 00 00 076 22 HIR
```

Figm 4. (Rad.) Flera program på ett kort. Definitionsprogram.

Mata in/Tryck med Display Kommentar

```
E 10. Initiering
157 R/S 11. Address Lbl E'
214 R/S 12. Address Lbl A
223 R/S 13.
220 R/S 14.
233 R/S 15.
234 R/S 16.
167 R/S 17. Sista adressen
A 10. Nästa program
85 R/S 11. Address Lbl E'
152 R/S 12.
R/S 13. Lbl B odefinierad
R/S 14.
R/S 15.
R/S 16.
R/S 17.
R/S 18.
131 R/S 0. Sista labeln
B (Sätt in kort) Spela in
```

Nu kan programmen provas. Läs in block 1 (innehållande pgn 3 & 5) och block 2 (inspelat av definitionsprogrammet). Inläsningen måste göras i normaluppladdning. För att välja ut det första programmet trycker man SBR SBR. Detta kommer också att medföra att programnavsett som definierades som Lbl E' kommer att köras. Vi får alltså utskriften "FAKULTET" och sedan "N7?". Nu kan ett tal matas in och R/S tryckas så beräknas fakultet.

För att plocka fram bråkräkningsprogrammet trycker man SBR SBR igen. Lbl A - Lbl A' kan nu användas för bråkräkning. Trycker man SBR SBR ännu en gång får man tillbaka fakultetsprogrammet.

```
#085 69 DP 125 65 DP 165 03 03 205 49 PTH
086 00 00 126 05 05 166 95 + 206 32 02
087 98 ADV 127 22 HIR 167 58 PTH 207 43 RCL
088 02 128 58 PTH 168 00 00 208 32 02
089 02 129 00 00 169 42 STD 209 00 00
090 03 1 130 00 00 170 21 01 210 43 RCL
091 09 0 131 00 00 171 01 01 211 00 00
092 02 2 132 42 STD 172 42 STD 212 22 HIR
093 06 4 133 00 00 173 06 4 213 00 00
094 04 4 134 43 RCL 174 42 STD 214 42 STD
095 01 1 135 00 00 175 01 1 215 00 00
096 02 2 136 45 + 176 35 CLR 216 32 XIT
097 01 01 137 32 XIT 177 32 XIT 217 00 00
098 69 DP 138 00 00 178 45 + 218 24 04
099 61 01 139 01 01 179 45 + 219 45 PTH
100 03 3 140 24 34 180 48 EXC 220 32 PTH
101 07 7 141 03 3 181 00 00 221 84 +
102 01 1 142 01 1 182 95 + 222 32 XIT
103 07 7 143 02 07 183 43 RCL 223 43 PTH
104 03 3 144 03 3 184 00 00 224 24 04
105 07 7 145 49 DP 185 54 1 225 32 XIT
106 00 0 146 04 04 186 94 + 226 45 +
107 00 0 147 01 01 187 52 HIR 227 43 RCL
108 00 0 148 05 + 188 22 HIR 228 03 03
109 00 0 149 49 DP 189 52 HIR 229 00 00
110 69 DP 150 06 06 190 05 + 230 61 STD
111 02 02 151 02 02 191 82 HIR 231 81 01
112 69 DP 152 61 STD 192 12 12 232 59 +
113 02 02 153 01 01 193 95 + 233 61 STD
114 86 ADV 154 12 12 194 22 HIR 234 45 +
115 03 3 155 07 07 195 87 82 235 61 STD
116 01 1 156 03 03 196 01 01 236 01 01
117 02 2 157 07 07 197 78 78 237 00 00
118 09 PTH 158 43 RCL 198 00 00 238 49 PTH
119 00 00 159 00 00 199 00 00 239 49 PTH
120 69 DP 160 92 HIR 200 49 PTH
121 69 DP 161 95 + 201 32 HIR
122 00 00 162 42 STD 202 42 STD
123 69 DP 163 45 + 203 01 01
124 01 01 164 43 RCL 204 22 HIR
```

Figm 5. (Rad.) Flera program på ett kort. Exempel.

©

Graphics Mode och FAST MODE

PLOT 60 - II

AV ROBERT PRINS; ÖVERSÄTTNING OCH BEARBETNING: LARS HEDLUND

När Graphics Mode och Michael Sperbers Plot 60-program kom var ju detta en stor sensation. Nu är det dags för förbättringar och här kommer vi att beskriva De Bilt, Holland, förbättrat upplösningen ytterligare något och dessutom infört en variant med Fast Mode. (På annan plats i detta nummer finns en elegant tillämpning - cirkeldiagram i Fast Mode.) Så här skriver Robert själv i översättningen:

Plot 60 Typ II

Plot 60 Typ II är en revidering av Michael Sperbers ursprungliga program. Det placerar h25 på steg 016 vilket medger att "n" kan användas i st f "m". Eftersom h25 på 016 avbryter utskriften snabbare än på 024 blir också upplösningen bättre. Du märker också att programmet använder parenteser vilka emellertid blir utan nytta då vilande operationer alltid störs av CLR på steg 013. Att ta bort dem skulle göra programmet ännu kortare. Plot 60 Typ II har en viktig förändring framför Plot 60: subrutinstacken är alltid tom efter körning men inte genom RST. Det betyder att programmet skulle kunna sättas in i en modul! (?)

Och nu till instruktionerna.

1 Skapa h25 på steg 016: 10 Op 17 CLR
GTO 016 Pgm 19 SBR 045 P/R LEN Ins LEN
RST CLR 6 Op 17

2 Lågg in funktionerna med början på steg 220. Precis som i Plot 60 finns x-värdet i displayen och R05. Sluta varje funktion med B, utom den sista som skall sluta med GTO 061.

3 Mata in parametrarna:
Antal punkter (punktrader) att plotta n A
Ymin R/S
Ymax R/S
Xmin R/S
Xmax R/S
X0 R/S
AX R/S
Antal remors
Sedan T matats in startar programmet.

Två ställen i programmet är kanske av intresse: Användningen av decimalmaska och Int för att skriva koordinater för J, L och på steg 164-172 och att jag inte använde "D+S HIR 18 Int" på steg 192-200. Den "normala" metoden är mycket snabbare!

En sista anmärkning: Steg 016-023 har efter initiering ändrats till "h25 RTN INV Fix CMS GTO 0 25" ("A. Alla normala Plot-60-regler gäller för detta program!

Och nu över till något som är litet svårare: Plot 60 typ II i Fast Mode.

För att kunna skriva programmet måste jag lösa problemen med att gå in i och ut ur Fast Mode utan att hamma i Trace-läge samt att lämna ihop funktionerna. Det sistnämnda löstes ganska lätt. Sluta varje funktion så här: f(n) x:qq GTO ... där f(n) är det beräknade funktionsvärdet för x=n, ... det första steget i skrivkodensomvandlaren och det första steget i nästa funktion.

Nu till det svårare problem: Att gå in i och ut ur Fast Mode.

Int i FM: Det enda sättet som går att använda i ett program är att utnyttja h12.

Än en gång: Alla Plot 60-regler gäller med ett tillägg: R14 får inte användas till funktionerna.

3 Mata in parametrarna:
Antal punkter (punktrader) att plotta n A
Ymin R/S
Ymax R/S
Xmin R/S
Xmax R/S
X0 R/S
AX R/S
Antal remors
Sedan T matats in startar plottningen automatiskt.

På grund av den extra tiden för att anropa funktionerna är programmet bara ca 1 1/2 gång så snabbt som Plot 60 typ II utan Fast Mode.

Ut ur FM: RST Det skulle emellertid ge problem med RTN på steg 000. En möjlig lösning vore att byta ut detta RTN mot GTOInd - i varje fall inte särskilt snyggt.

Lyckligtvis finns det en mycket bättre metod, som har beskrivits av Patrick Acosta i V6NSP4 av TI PPC Notes. Anropa FM-programmet som subrutin och se till att första siffran i initieringskonstanten är en etta så att enbart flagga 4 ställs vilket tillåter en att gå ut ur FM utan att hamma i Trace-läge. Några försök enligt "trial and error" gav sekvensen " $1 \sqrt{x}^2 1/x$ " (bättre än " $5 \sqrt{x}^2 1/x$ " som återvänder i Trace) vilken är en perfekt initieringskonstant.

När de här problemen är lösta kan man nästan gissa hur programmet blir. Det tog mig ungefär 60 minuter att skriva och avlusa det.

Nedan: Plot 60 typ II (normal mode).

T: Plot 60 typ II i Fast Mode.

Stegen fr o m 220 resp. 236 innehåller de funktioner som skall plottas.

```
0000 92 RTN 060 12 B 120 17 17 180 55 +
0001 76 LBL 061 00 0 121 29 CP 181 43 RCL
0002 11 R 022 48 EOC 122 22 INV 182 07 OF
0003 61 GTO 065 09 09 123 77 GE 183 22 INV
0004 00 00 064 69 DP 124 02 02 184 28 LCO
0005 18 18 065 01 01 125 17 17 185 52 EE
0006 76 LBL 066 00 0 126 33 CP 186 23 CP
0007 12 B 067 48 EOC 127 33 CP 187 22 INV
0008 61 GTO 068 10 10 128 53 INV 188 07 OF
0009 01 01 069 60 00 129 46 IHS 189 42 STD
0010 11 R 070 60 02 02 130 00 0 190 53 OF
0011 76 LBL 071 48 EOC 131 00 0 191 54 +
0012 13 13 072 48 EOC 132 05 192 53 OF
0013 02 02 073 48 EOC 133 05 193 53 OF
0014 69 DP 074 69 DP 134 75 + 194 26 INV
0015 76 LBL 075 00 0 135 39 INV 195 33 CP
0016 74 SHL 076 00 0 136 42 STD 196 01 1
0017 40 70 077 48 EOC 137 13 197 00 0
0018 12 B 078 12 15 138 54 + 198 00 0
0019 00 0 079 00 0 139 54 + 199 00 0
0020 17 R 080 04 04 140 05 5 200 59 IHT
0021 30 TRM 081 43 RCL 141 33 STD 201 59 IHT
0022 61 GTO 082 04 04 142 07 07 202 22 INV
0023 51 083 44 SUM 143 07 07 203 57 OF
0024 42 STD 084 05 05 144 59 IHT 204 02 02
0025 06 085 13 145 54 + 205 17 17
0026 92 RTN 086 97 92 146 44 SUM 206 53 +
0027 42 STD 087 00 00 147 07 07 207 43 RCL
0028 01 01 088 00 00 148 54 + 208 07 OF
0029 22 INV 089 54 54 209 65 +
0030 44 SUM 090 98 98 210 53 +
0031 02 02 091 12 12 211 13 13
0032 92 RTN 092 08 08 212 52 EE
0033 44 SUM 093 99 PRT 213 59 IHT
0034 02 02 094 43 RCL 214 67 67 214 54 +
0035 92 RTN 095 02 02 125 01 01 215 4 88H
0036 42 STD 096 44 SUM 156 67 67 216 13 13
0037 03 03 097 01 01 157 32 R/T 217 43 RCL
0038 92 RTN 098 98 98 158 02 02 218 05 05
0039 42 STD 099 74 74 159 56 56 219 07 07
0040 00 00 100 00 0 160 22 INV
0041 92 RTN 101 161 16 16 219 92 RTN
0042 00 00 102 00 0 162 41 SUM
0043 08 08 103 00 0 163 70 70 220 29 CLR
0044 22 INV 104 22 R/TN 164 55 55 221 29 CLR
0045 49 PRT 105 53 13 165 05 5 222 39 PRT
0046 76 LBL 106 10 10 166 07 07 223 07 OF
0047 43 RCL 107 46 IHS 167 02 2 224 85 +
0048 06 108 7 7 168 02 2 225 02 2
0049 42 STD 109 43 RCL 169 53 53 226 07 OF
0050 00 110 01 01 170 53 53 227 07 OF
0051 43 RCL 111 54 + 171 07 7 228 95 +
0052 03 03 112 59 59 172 00 0 229 08 CDS
0053 42 STD 113 01 1 173 52 EE 230 12 B
0054 05 05 114 25 R/T 231 93 PRT
0055 43 RCL 115 43 SUM 232 02 2
0056 02 02 116 02 13 233 04 +
0057 61 GTO 117 54 + 175 53 + 234 00 0
0058 02 02 118 07 07 235 73 RCL
0059 20 20 119 02 02 179 19 13
```

Instruktionerna är

1 Skapa h25 och h12: 10 Op 17 CLR GTO 016
Pgm 19 SBR 045 P/R
LEN Ins (skapar h25)
SST (16 ggr, jag
föredrar att stanna
i ROM)
Ins (skapar h12)
LEN RST CLR 6 Op 17

2 Mata in funktionerna med början på steg 236. Som vanligt ligger x i displayen och i R05. Sluta alla funktionerna utom den sista så här: f(n) x: ppp GTO 119 där f(n) är det funktionsvärdet som beräknats för aktuell x och ppp på numret på steget efter GTO 119. Sluta den sista funktionen så här: f(n) x: t 94 GTO 119. Ett exempel i listningen visar hur det fungerar.

```
0000 92 RTN 070 05 05 140 53 + 210 45 x
0001 76 LBL 071 12 B 120 17 17 180 55 + 211 41 R
0002 11 R 072 13 C 142 53 + 212 00 0
0003 61 GTO 073 87 82 143 05 5 213 00 0
0004 00 00 074 00 00 144 05 + 214 54 +
0005 41 41 075 00 00 145 04 + 215 59 IHT
0006 76 LBL 076 73 73 146 85 + 216 29 CP
0007 12 B 077 98 80V 147 09 9 217 07 OF
0008 61 GTO 078 43 RCL 148 75 75 218 67 E9
0009 00 00 079 08 08 149 59 IHT 219 02 02
0010 11 18 080 99 PRT 150 42 STD 220 32 32
0011 76 LBL 081 43 RCL 151 13 13 221 93 +
0012 13 C 082 02 02 152 54 + 222 43 RCL
0013 25 CLR 083 44 SUM 153 05 5 223 07 OF
0014 69 DP 084 01 01 154 05 5 224 65 +
0015 05 085 98 98 155 42 STD 225 43 RCL
0016 74 SHL 086 97 92 156 07 07 226 17 13
0017 40 087 02 02 157 32 R/T 227 43 RCL
0018 12 B 088 00 00 158 54 + 228 59 IHT
0019 00 00 089 65 65 159 26 INV 229 54 +
0020 30 TRM 090 43 RCL 160 33 STD 230 59 IHT
0021 30 TRM 091 92 RTN 161 94 + 231 74 SHL
0022 61 GTO 092 04 04 162 07 07 232 13 13
0023 51 093 44 SUM 163 07 07 233 05 05
0024 00 094 00 00 164 54 + 234 00 0
0025 93 + 095 69 DP 165 65 + 235 14 14
0026 06 096 13 166 54 + 236 19 19
0027 33 097 00 0 167 00 0 237 02 2
0028 10 098 10 10 168 02 02 238 04 +
0029 10 10 099 10 10 169 59 IHT 239 29 CP
0030 57 100 09 09 170 47 EOC 240 61 GTO
0031 71 71 101 02 02 171 01 01 241 02 2
0032 25 102 00 0 172 62 62 242 19 19
0033 54 + 103 48 EOC 173 32 R/T 243 03 3
0034 46 PRT 104 11 11 174 02 2 244 61 GTO
0035 50 105 10 10 175 22 INV 245 01 01
0036 94 + 106 03 03 176 67 EOC 246 19 19
0037 93 107 00 0 177 01 01 247 39 CDS
0038 00 108 00 00 178 05 5 248 32 R/T
0039 47 CDS 109 12 12 179 02 2 249 02 2
0040 42 STD 110 10 10 180 05 5 250 05 5
0041 06 111 04 04 181 53 + 247 01 01
0042 92 RTN 112 43 RCL 182 02 02 248 61 GTO
0043 42 STD 113 04 04 183 07 07 249 01 01
0044 01 114 02 02 184 07 07 250 19 19
0045 22 INV 115 05 05 185 04 + 251 01 01
0046 02 116 16 16 186 07 07 252 01 1
0047 02 117 16 16 187 07 07 253 01 1
0048 92 RTN 118 14 14 188 52 EE 254 00 0
0049 00 119 14 14 189 54 + 255 00 0
0050 02 120 53 + 190 44 SUM 256 39 CDS
0051 92 RTN 121 92 R/T 191 13 13 257 32 RCL
0052 42 STD 122 75 + 192 53 + 258 02 2
0053 03 123 02 2 193 07 07 259 06 6
0054 92 RTN 124 01 01 194 13 13 260 04 4
0055 04 125 54 + 195 43 RCL 261 01 01
0056 04 126 55 + 196 43 RCL 262 01 01
0057 92 RTN 127 07 07 197 07 07 263 19 19
0058 42 STD 128 32 32 198 22 INV 264 85 +
0059 08 129 08 129 28 LCO 265 02 2
0060 22 INV 130 02 02 200 52 STD 266 04 4
0061 93 PRT 131 54 + 199 43 RCL 267 08 CDS
0062 02 132 77 GE 202 22 INV 268 95 +
0063 02 133 02 13 203 04 + 269 39 CDS
0064 06 134 32 32 204 42 STD 270 32 R/T
0065 43 RCL 135 02 02 205 43 RCL 271 49 IHT
0066 00 136 22 INV 206 54 + 272 04 4
0067 43 RCL 137 07 07 207 43 RCL 273 61 GTO
0068 03 138 02 02 208 22 INV 274 01 01
0069 42 STD 139 32 32 209 59 IHT 275 19 19
```

TIPS OM HEMBYGGEN

av: Bo Nordlin

I Programbiten har vi oftast skrivit om hur man kan utvidga TI 58/59 genom användbara programmeringsknep: jag tänker på Fast och Graphics Mode, "decimalpunktsknepet", skrivkodsomvandlare, "hur man påverkar ett program under körning" etc. Den andra sidan: hur man utvidgar räknares förmåga genom att bygga om den, har vi nästan aldrig nämnt. Redaktörerna för CHIP-Special Taschenrechner har gått den andra vägen. På 164 sidor med ett 40-tal artiklar skriver dom nästan enbart om hur man kan bygga om TI 58/59.

Här nedan skall jag kortfattat sammanfatta det väsentligaste ur denna skrift. Låt mig dock först slå fast att denna tidning ger, förutom en mängd högtintressanta byggbeskrivningar, en förståelse för 58/59:ans konstruktion och uppbyggnad, som ligger på en klart djupare nivå än Programbitens "Djupdykningar i TI-59". Tidningen innehåller också ett par artiklar rörande programmering: t.ex. en jämförelse mellan AOS och EFN, en beskrivning över hur man omvandlar ett PASCAL-program till TI-språk. En del program ingår även: två spelprogram till HP-41, ett stjärnpositionsprogram (som är exaktare än "Planet-lägen", publicerat i PB 80-1) samt två såmre program: "Bräkräkning" och "Hushållsbudget". Trots att tidningen innehåller många fotografier och klara krets-korts-scheman är det nödvändigt med god kunskap i både tyska och elektronik, för att kunna tillgodogöra sig innehållet.

Programmerbar pipton

Denna krets gör det möjligt att i ett program styra en tonalstrare. Varje gång programmet kör funktionen DEL (kod 56) ljuder tonen, det det går att välja en annan funktion, exempelvis 2nd (kod 21), 2nd INV (kod 27), NOP (kod 68) eller ADV, Print eller List; de sistnämnda fungerar dock enbart då skrivare ej är ansluten. Hela kretsen kan byggas in i en TI 58 (på magnetkortsläsarens plats), i en 59:a måste tonalstraren placeras externt.

Musikinterface

Den består bl.a. av sex stycken fototransistorer som placeras över displayens sex sista siffror. De "känner av" om siffran

är en etta eller åtta: den informationen omvandlas till en analog signal, som styr en tongsensor. Det innebär att man slipper göra ingrepp i 58/59. Ett program som underlättar inmatandet av melodier är också publicerat.

Timer

Med hjälp av någon av dessa kretsar är det möjligt att med räknares styra på- och av-sättandet av elektriska apparater, såsom lampor, element osv.

Det finns fyra huvudtyper av timers:

De elektromagnetiska, som aktiveras av radiovågor(!). När räknares kör ett program skickar den ut en svag störningssignal, som kan uppfattas på mellanvågsbandet. Det fungerar faktiskt; prova genom att ställa räknares i närheten av antennen till din AM-radio, som du har ställt in på 720 KHz. Tryck R/S och du hör en svag periodiskt ljud, tryck R/S på nytt och ljudet upphör. Den här sortens timer fungerar alltså som ett slags AM-radio.

De induktiva, displayen på 58/59 avkänns över 200 gånger per sekund när den är i "standby"-läge (dvs. då den inte kör ett program). Det ger en växelström på mer än 200 Hz. Om en spole anbringas i närheten av displayen induceras en svag ström till spolen; den strömmen styr timern.

De optiska, finns i två varianter: den enkla, som aktiveras när displayen tänds (antigen genom R/S eller Pause). Den mer avancerade kretsen tar även hänsyn till vad som visas i displayen; därigenom är det möjligt att styra upp till 20 kanaler. Dvs. lampa i tänds 18:30, elementet sätts på 18:52, kaffebruggaren startar 07:00 osv.

De elektroniska, kopplas in i räknares. Räknares har en inbyggd klocka som synkroniserar alla händelser i räknares. Ju högre frekvens på klockan, desto snabbare blir räknares, men den drar även mer ström vid högre klockfrekvens; för att

öka batterilivslängden sänks frekvensen när programexekveringen är avslutad. Denna förändring registreras av kretsen, som då slår på strömmen.

Alfanumeriskt tangentbord

Tangentbordet, som får sin strömförsörjning från kalkylatorn, gör att man kan glömma alla skrivkoder. Under tangentbordet finns en buffert som lagrar skrivkoderna för fem tecken, när bufferten är fylld startas automatiskt ett program i räknares, som lagrar skrivkoden i ett register.

Joystick

En s.k. Joystick, dvs. en styrspak som främst används för att styra exempelvis "rymdraketer" i spelprogram, kan anbringas på räknares. När joysticken trycks åt vänster sker ett subrutin-anrop till label A, vid tryck åt höger anropas label B.

Interface mellan TI 58/59 och mikrodatorer

Interface kopplas in vid skrivrututgången till PC-100. Det gör det möjligt att lagra program till kalkylatorn på kassetband eller floppy-discs. Det innebär även att 58/59 kan användas som en portabel terminal, från vilken man kan överföra data till datorn vid ett senare tillfälle.

Interface mellan 58/59 och en X/Y-plottter

Även detta interface kopplas in vid skrivrututgången, vilket innebär att man slipper göra lödningar inne i räknares eller skrivaren. Detta projekt ger en billig styransordning till X/Y-plotttern.

Tillbaka matning av rader på PC-100

Tidigare har det endast varit möjligt att skjuta fram rader med Adv, nu går motsatsen att göra. För att ändra riktning på pappersremsan utförs kommandot Pause. Detta underlättar vid all sorts plottning. Man kan först skriva ut X- och Y-axeln, sedan automatiskt dra tillbaka remsan till början och där påbörja plottningen.

TV-interface

En TV-skärm kan ibland ersätta skrivarens funktion, speciellt med dagens priser på

skrivarpapper. Med detta interface kan en vanlig TV anslutas till skrivrututgången. På TV-skärmen kan man sedan få programlistningar, alfanumerisk text, plottning etc. Kort sagt: alla skrivarens möjligheter.

Minnesexpansion på TI 58C

En fmedubbling av minnet blir resultatet av denna expansion. Minnet delas upp i tio block om vardera 256 bytes. Block 1-9 består av 240 programsteg och två minnen (R₂₈ och R₂₉). Block 0 består av 28 minnen (R₂₀ till R₂₇), dessa register kan användas av program i samtliga de övriga blocken. R₀₀ är ett pekregister som pekar på det block som för tillfället används, för att byta från exempelvis block 2 till block 3 gör man helt enkelt Op 20 eller 3 STO 00. Byta mellan block är möjligt eftersom kalkylatorn hämtar åtta programsteg i taget från programminnet, dessa steg lagras i taget i en buffert. Om man då bland dessa åtta steg har sekvensen Op 20 (byte av block) CNO A, kommer räknares att hoppa till label A i det nya blocket. Om 58:an tog fram ett steg i taget ur programminnet skulle det hela bli mycket mer komplicerat: om Op 20 ligger på steg 100 skulle programmet hoppa in på steg 101 i det nya blocket. Därigenom skulle det bli svårkontrollerat, som tur är arbetar inte 58:an inte på det viset.

Tidigare har vi lärt oss att bara fantasin sätter gränserna för de program vi kan göra. Nu är det uppenbart att även möjligheterna till ombyggnad är obegränsade.

Vill du köpa häftet? Skriv då till CHIP-Leserservice, Vogel-Verlag, Postfach 6740, D-8700 WÜRZBURG, Västtyskland, och uppge att du önskar beställa en CHIP-Special Taschenrechner. Efter två veckor kommer häftet med en räkning på DM 33:50, vilket blir ca 100 kr.

Jag vill be om ursäkt för det myckna användandet av engelska ord. Tyvärr finns det ofta inte några bra motsvarigheter i det svenska språket, svenskan har inte kunnat hålla jämna steg med den tekniska utvecklingen.

Periodiska systemet

PROGRAM: LEM MATTESSON. TEXT: GÖRAN AHLSTRÖM

Nedanstående program publicerades ursprungligen i TI PPC NOTES 17M/SP18-20 (Fästelse i V7N6P9) och skrevs av Lem Mattesson i USA. Det är avsett för TI-59 med skrivare och ryms på tre magnetkort. (Program + två datakort).

Programmet innehåller data för 107 ämnen och kan alltså endast användas i kombination med skrivare. Utskriften har LiTH olika utseende beroende på om det ämne vars data man vill åt är stabilt eller instabilt (radioaktivt). Se f.ö. nedanstående tabell.

STABILA ÄMNEN

atomnummer

symbol

atommassa

valens 1)

medelant neutr

ant stabila

isotoper

proc förekomst
av den vanligaste
isotopen

massan 2)

ant neutroner 2)

ant protoner

INSTABILA ÄMNEN

atomnummer

symbol

atommassa

valens 1)

medelant neutr

ant naturl

förekommande
isotoper 3)

halv.tid 4)

massan 4)

ant neutr 4)

ant protoner

- 1) Antal elektroner i det yttersta ofyllda skalet. 0 innebär att skalet är fyllt.
- 2) För den vanligaste isotopen.
- 3) För konstgjorda ämnen antalet isotoper med halveringstid med samma enhet (år, månader, dagar etc.).
- 4) För den mest långlivade isotopen.

Alla data är hämtade ur 60:e utgåvan (1979-80) av "Handbook of Chemistry and Physics". Atommassan, vilken är massan av 6-DECE3 atomer av ämnets normala blandning har i vissa fall rundats av till fyra signifikanta siffror för att passa utseendeformatet. Den procentuella

förekomsten har i flera fall rundats av till ett heltal. En skrift av 100 % är sann om antalet isotoper endast är en. Allt över 99.5 % har rundats av.

Det finns två sätt att hitta ett ämne i Allt mata in atomnumret (är snabbast) eller att mata in skrivkoden för symbolen. När det sökta ämnet har hittats skrivs dess data ut på skrivaren.

ANVÄNDARINSTRUKTIONER

Programmet består av tre kort. Huvudprogrammet (block 1 & 2) har 479 steg. De två datakort A och B i innehåller vardera 54 minnen med data ett minne per ämne. Kort A innehåller ämnena 1 - 54 och kort B ämnena 55 - 107. Det sparar tid om man vet vilket kort det sökta ämnet finns på.

- 1) Läs in programkortet (block 1 & 2) och ett av datakort A eller B (block 3 & 4) i normaluppdelning.
 - 2) a) Mata in atomnummer och tryck A. eller b) Mata in skrivkoden för symbolen och tryck B.
- Oberoende av vilken av 2a) eller 2b) som utförts så skrivs data för ämnet ut.

Om kort B är inläst och det sökta ämnet är mellan 1 och 54 kommer "READ CARD A/R/S" att skrivas ut varefter programmet stannar med "3" i displayen. Läs då in block 3 på kort A. Tryck på CLR och läs in block 4 på samma kort. Därefter trycker man på R/S varvid papperet matas fram en rad och programmet stannar med det atomnummer eller den skrivkod som matats in på displayen. Tryck sedan på A eller B varefter alla data erhålles som tidigare. Ett motsvarande meddelande skrivs ut för kort B om kort B är inläst. För instabila element vars halveringstid uttrycks i år anges denna i tiopotensform med en siffra och exponent.

PROGRAMBESKRIVNING

Alla data för varje ämne lagras i ett enda minne. Stabila ämnen lagras som positiva tal och instabila ämnen som negativa tal. De senare har desutom ett sinneslänat något olikartat utseende. Så fort ett negativt minnesinnehåll har upptäckts vid sökningen

kommer flagga 8 att ställas. För ämnena 1 - 54 (kort A) är minnesnumret samma som ämnets medan för ämnena 55 - 107 (kort B) är minnesnumret lika med ämnets nummer minus 54. Flaggas i resp 2 är ställd beroende på om kort A eller B är inläst. Skrivkoden för varje ämnes symbol är lagrad som ett heltal i varje minne. Detta tillåter sökning efter symbol i stället för atomnummer. Programmet startar sökningen med minnet och söker sedan i fallande ordning (DSI D) till till motsvarande tal påträffats. Det tar ungefär en halv sekund att kontrollera ett minne så det är en långsamt metod eftersom man då oftast inte vet var ämnet finns.

Det snabba sättet att söka är genom att mata in atomnumret. Detta lagras i R 00 och om det är 54 eller mindre och flagga 1 är ställd erhålles data direkt genom RCL IND. Om atomnumret är 55 eller högre och flagga 2 är ställd subtraheras först 54 från R 00.

Om felaktigt flagga 2 är ställd kommer programmet att skriva ut "READ CARD A/R/S" resp "READ CARD B/R/S". Allt man behöver göra är att läsa in motsvarande kort samt trycka på R/S. Detta nötslås även om flaggan och tar fram den tidigare inmatade talet. Tryck sedan på A eller B som valnigt.

Efter som upp till fyra positioner används till skrivkoden finns det bara nio kvar för alla data. För stabila ämnen består de fyra första decimalerna av dess "tal" av de fyra mest signifikanta siffrorna i atommassan.

För ämnena 1 - 4 ställs flagga 3 för att ge tre decimaler medan för ämnena 5 - 42 flagga 4 ger två decimaler. Ämnena 43 och uppåt har bara en decimal.

Decimal fem innehåller ämnets valens medan den sjätte innehåller det tal som måste adderas till eller subtraheras från atommassans heltalsdel för att ge massan av den vanligaste isotopen. Tre har adderats till det verkliga talet för att ge ett tal som alltid är positivt. Den sjunde decimalen innehåller antalet stabila isotoper minus ett. Detta tillåter intervallt CLSD att lagras som en enda siffra. Decimalerna åtta och nio innehåller den procentuella förekomsten av den vanligaste isotopen för ämnet ifråga. Efter som denna riktigtvis ej kan vara noll tolkas en nolla som 100 %.

När kort B är inläst orsakar flagga 2 att programmet hoppar över den del som involverar flaggorna 3 och 4, varför alla atommassor har endast en decimal.

För instabila ämnen lagras data på ett något anordnat sätt. De tre första decimalerna är atommassan avrundad till ett heltal, vilket tillåter den fjärde decimalen att användas för att koda halveringstidens enhet. 0 betyder 3 dagar, 1 betyder 1 månad, 2 betyder 1 år, 3 betyder 100 år, 4 betyder 1000 år, 5 betyder 10000 år, 6 betyder 100000 år, 7 betyder 1000000 år, 8 betyder 10000000 år, 9 betyder 100000000 år. Efter att det plöskats fram placeras detta tal i R 59 varefter STF IND 59 ställer den flagga som styr

utskriften av enheten.

Det innebär inga konflikter att använda flaggorna 3 och 4 här eftersom de enda instabila ämnena på kort A alla har en halveringstid i år och om kort B är inläst så ställs aldrig flaggorna 3 och 4.

Decimalerna fem, sex och sju har samma betydelse som tidigare medan åtta och nio innehåller halveringstidens numeriska värde i form av ett tal med en decimal för sekunder i o m sekunder. För R lagras halveringstiden som ett ensiffrigt heltal plus exponent vid tiopotensform. V x s det antal nollor som ska adderas i valnigt notation.

KOMMENTARER

Vad ska man nu ha det här programmet till?

Det finns ju periodiska system att tillgå om man behöver.

Svaret är att de periodiska system som finns att tillgå är oftast inte lika informationrika som detta program samt att de desutom brukar ha en viss vana att försvinna när man som bäst behöver dem (undrar om Murphys lag spelar en viss roll här?). Alternativet är att sätta sig och bläddra i "Handbook", men detta är för det första jobbigt och för det andra så kräver det en viss utbildning på tabellverket ifråga.

I valet mellan att använda programmet och att bläddra i "Handbook" väljer Steinert jag att använda programmet. En nackdel är ju att man måste ha skrivaren med sig, men den är ju ändå tiden lättare än det ska fee kg tunga tabellverket.

Fördelaktigt är att programmet är så otroligt lätt skött, vilket kan tyvärr ej kan säga om alla program man ser.

En något oönskad del är i stället uppgiften om att leta protoner. Denna uppgift har man ju redan i och med att atomnumret skrivs ut bredvid symbolen.

För att ge ett slutord måste jag dock säga att programmet är väl värt att lägga in på magnetkort för framtida bruk. Det kanske inte kommer att användas dagligen av andra än kemister, men när man en dag behöver just de här uppgifterna så finns de lätt tillgängliga.

Om någon är intresserad av den nämnda "Handbook of Chemistry and Physics" så rekommenderas att denna tar kontakt med IFM, Institutionen för Fysik och Mätteknik vid LiTH (Tekniska Högskolan i Linköping). Förra året kunde den beställas utlagan (6 ölar 1980-81) köpas från USA via IFM för ca 160 kr.

Översättning och granskning Göran Ahlström 82 HL

17

Van der Waals gaslag

AV GÖRAN AHLSTRÖM

Detta program är tänkt att användas av alla som har behov av att någon gång räkna LiHte noggrannare med gaser eller vid de tillfällen då den allmänna gaslagen inte är tillräckligt exakt.

TEORI

Den allmänna teorin för gaser kan sammanfattas i den s k allmänna gaslagen som har följande utseende:

$$pV=nRT \quad (1)$$

och där p = trycket
 V = volymen
 n = antal mol gas
 R = allmänna gaskonstanten
 T = temperaturen

Vanligtvis ger denna formel ett tillräckligt noggrant svar för de flesta tillstånd, men det finns vissa tillfällen då den ger helt felaktiga värden. Detta gäller då främst vid de tillfällen då gasen är på väg att förändras till vätska i kondensationen. För att räkna bot på dessa missförhållanden konstruerade keelsten Van der Waal en egen gaslag och den har därefter upptäckts efter honom. Han tar där in två nya faktorer i beräkningarna, konstanterna a och b , vilka tar hänsyn till molekylernas relativa storlek samt deras inbördes attraktionskraft. Som en följd av detta är de specifika för varje ämne och finns därför tabellerade i bl a Handbook of Chemistry and Physics. Formelns utseende är nedanstående:

$$\left(p + \frac{a}{V^2} \right) (V - nb) = nRT \quad (2)$$

Som synes ger detta ett linjärt uttryck när man vill lösa den för p eller T , medan den för n och V ger upphov till en tredjegrads-ekvation. Detta faktum har gjort att man i det längsta dragit sig för att tillgripa den eftersom det helt enkelt har kostat mer än det smakat att använda den.

I och med datorteknikens intåg har dessa skillnader reducerats mer och mer och har numera inte någon större betydelse eftersom det finns biblioteksrutiner för att lösa ekvationen numer. Nedanstående program är ett försök till just en sådan rutin för TI-58/59.

För den som är intresserad av en tydligare genomgång av bakgrunden eller betydelsen av de inledande termerna hänvisas till någon bok om exempelvis fysikalisk kemi eller termodynamik.

PROGRAMBESKRIVNING

Givet tre av fyra parametrar samt konstanterna a och b beräknar programmet den saknade. För p och T beräknas värdet exakt (i ena gällande noggrannhet) medan för n och V Newton-Raphsons metod används för att approximera en rot. För den skall beräknas först en begynnelseapproximation med hjälp av den allmänna gaslagen och därefter används denna som startvärde. Avbrottsvillkor är när fem korrekta decimaler erhållits.

På grund av att tredjegradsdelen inte löses exakt kan det hända att programmet misslyckas i rotuppsökningen. Eftersom det oftast är en mycket snabb konvergens kan man vara säkra på att problem uppstått om man efter flera minuter ännu inte fått något svar. Det återstår då endast att lösa ekvationen exakt med t ex det tidigare publicerade programmet för tredjegradsens lösning med reella koefficienter.

Vid användande av programmet är det viktigt att tillse att alla inmatade data är i samma enhet, eftersom svaret annars blir felaktigt. Som standard har jag använt SI-enheter vilket har medfört att värdet på R är 8.314 J/mol K. Eftersom värdena på a och b i handboken är angivna i enheterna kvadratmeter och/kvadratmol resp liter/mol måste man antingen räkna om värdena på a och b eller mata in ett annat värde på R .

I detta fall ska i så fall värdet 0.08206 användas i stället för standardvärdet 8.314. En genomgående använder ett annat måttssystem än det standardiserade bör du mata in motsvarande värde på R i programmet på steg TI-16. Givetvis kan man även ändra storleken på avbrottsvillkoret.

Lbl A är en initieringsrutin för konstanterna a , b och R .

Lbl A' och B' är interna subrutiner.

Lbl B används för att lagra/beräkna p .

Lbl C används för att lagra/beräkna V .

Lbl D används för att lagra/beräkna n .

Lbl E används för att lagra/beräkna T .

För att lagra ett värde med hjälp av B-matar man bara in det samt trycker ner mot-svarande tangent. Det inmatade värdet kommer då att ligga kvar tills dess man ändrar det. För att beräkna en viss storhet trycker man på CLR och därefter på någon av B-E, men på CLR i displayen kommer nämligen beräkningen att startas då programmet testat displayen mot noll som första åtgärd.

Kort beskrivning:

Antal programtegl: 305
Antal labels: 7
SBR-nivåer: 1
Använda minnen: 00-12 samt T-registret
Absolut adressering: Ja

ANVANDARINSTRUKTIONER

1. Läs in programmet.
 2. Mata in a och tryck A.
 3. Mata in b och tryck B/S.
Programmet stannar då med det inmatade värdet på R i displayen. Om ett annat värde skall användas, mata då in det i R DD nu. OBS att det kommer att raderas och att S-i-värdet mata in vid ny initiering.
 4. Mata in tre av fyra i feljande i tur och ordning: p , V , n eller T och tryck B, C, D eller E.
 5. Tryck CLR och därefter den sökta storhetens tangent varvid den kommer att beräknas.
- Programmet kan användas flera gånger för samma gas eftersom indata ej förslärs mellan körningarna (ett inmatat värde ligger kvar tills man matar in ett nytt).

EXEMPEL PÅ KÖRNING

Antag att man vill beräkna volymen för en mol setan vid NT standardtillståndet, tryck i stäm och temp 025.

MATA IN: TRYCK MED: DISPLAY

2.253	A	2.253	(a)
0.04278	B/S	8.3143	(b)
0.08206	STO 00	0.08206	(c)
1	B	1.	(d)
1	D	1.	(e)
298	E	298.	(f)
	CLR C	24.4044	(g)

Kommentarer:

- a) Värdet på a , hämtat ur Handboken.
- b) Värdet på b , hämtat ur Handboken.
- c) 8.3143 är det inmatade värdet på R i SI-enheter.
- d) Trycket i atm.
- e) Mängden i mol.
- f) Temperatur i 298 K (+25 C)
- g) Det beräknade värde på volymen. Beräkningstid ca 10 s.

000 75 +	040 45 CL	080 43 KCL	120 25 ETH	160 17 B'	200 08 08	240 29 CF	280 43 KCL
001 33 -	041 43 KCL	081 03 03	121 74 LB	161 25	201 43 KCL	241 50	281 01 01
002 24 CE	042 11 11	082 25	122 12 B	162 43 KCL	202 03 03	242 02 02	282 03 03
003 45 +	043 85 KCL	083 43 KCL	123 23 CF	163 03 03	203 23 23	243 47 47	283 43 KCL
004 33 KCL	044 43 KCL	084 02 02	124 01 01	164 03 KCL	204 03 KCL	244 03 KCL	284 03 KCL
005 45 +	045 09 09	085 75 +	125 01 01	165 43 KCL	205 43 KCL	245 03 03	285 95 +
006 43 KCL	046 43 KCL	086 43 KCL	126 01 01	166 43 KCL	206 43 KCL	246 43 KCL	286 43 KCL
007 07 07	047 43 KCL	087 05 +	127 42 STD	167 95 +	207 95 +	247 95 +	287 10 10
008 43 KCL	048 01 01	088 43 KCL	128 01 01	168 43 KCL	208 43 KCL	248 43 KCL	288 43 KCL
009 43 KCL	049 01 01	089 05 05	129 25 ETH	169 74 LB	209 09 09	249 55 +	289 01 01
010 43 KCL	050 43 KCL	090 43 KCL	130 01 01	170 43 KCL	210 03 03	250 03 03	290 03 03
011 45 +	051 43 KCL	091 95 ETH	131 94 +	171 29 CF	211 03 03	251 02 02	291 43 KCL
012 43 KCL	052 43 KCL	092 43 KCL	132 01 01	172 43 KCL	212 03 03	252 03 03	292 03 03
013 11 11	053 50 10	093 17 B'	133 43 KCL	173 01 01	213 33 33	253 95 +	293 55 +
014 43 KCL	054 43 KCL	094 43 KCL	134 03 03	174 74 LB	214 43 KCL	254 43 KCL	294 43 KCL
015 85 +	055 43 KCL	095 43 KCL	135 45 +	175 42 STD	215 43 KCL	255 42 STD	295 00 00
016 43 KCL	056 43 KCL	096 43 KCL	136 12 12	176 03 KCL	216 03 KCL	256 03 KCL	296 03 KCL
017 09 09	057 50 10	097 75 +	137 04 04	177 25 ETH	217 95 +	257 43 KCL	297 43 KCL
018 43 KCL	058 43 KCL	098 43 KCL	138 45 +	178 43 KCL	218 43 KCL	258 43 KCL	298 43 KCL
019 43 KCL	059 50 10	099 03 03	139 43 KCL	179 01 01	219 42 STD	259 55 +	299 55 +
020 11 11	060 43 KCL	100 43 KCL	140 00 00	180 43 KCL	220 10 10	260 43 KCL	300 04 04
021 85 +	061 06 6	101 43 KCL	141 95 +	181 07 07	221 43 KCL	261 02 02	301 11 11
022 43 KCL	062 43 KCL	102 43 KCL	142 12 12	182 43 KCL	222 03 03	262 03 03	302 03 03
023 10 10	063 22 INW	103 94	143 95 +	183 43 KCL	223 45 KCL	263 42 STD	303 00 00
024 54 +	064 43 KCL	104 43 KCL	144 03 KCL	184 43 KCL	224 03 KCL	264 03 KCL	304 00 00
025 95 +	065 77 CE	105 74 LB	145 74 LB	185 45 KCL	225 00 00	265 43 KCL	305 43 KCL
026 43 KCL	066 43 KCL	106 43 KCL	146 43 KCL	186 43 KCL	226 43 KCL	266 43 KCL	306 43 KCL
027 03 03	067 75 +	107 42 STD	147 29 CF	187 06 06	227 43 KCL	267 45 KCL	307 43 KCL
028 43 KCL	068 43 KCL	108 43 KCL	148 43 KCL	188 43 KCL	228 43 KCL	268 43 KCL	308 43 KCL
029 43 KCL	069 12 12	109 31 31	149 01 01	189 43 KCL	229 55 +	269 43 KCL	309 43 KCL
030 07 07	070 42 STD	110 43 KCL	150 43 KCL	190 43 KCL	230 43 KCL	270 43 KCL	310 43 KCL
031 11 11	071 11 11	111 06 06	151 03 03	191 43 KCL	231 01 01	271 01 01	311 01 01
032 43 KCL	072 00 00	112 03 03	152 43 KCL	192 43 KCL	232 43 KCL	272 43 KCL	312 43 KCL
033 43 KCL	073 43 KCL	113 03 03	153 43 KCL	193 43 KCL	233 43 KCL	273 43 KCL	313 43 KCL
034 43 KCL	074 43 KCL	114 03 03	154 43 KCL	194 45 KCL	234 11 11	274 43 KCL	314 43 KCL
035 85 +	075 85 +	115 85 +	155 85 +	195 43 KCL	235 43 KCL	275 43 KCL	315 43 KCL
036 43 KCL	076 43 KCL	116 43 KCL	156 43 KCL	196 04 04	236 00 00	276 95 +	316 95 +
037 43 KCL	077 43 KCL	117 43 KCL	157 43 KCL	197 43 KCL	237 43 KCL	277 43 KCL	317 43 KCL
038 43 KCL	078 74 LB	118 42 STD	158 95 +	198 94 +	238 74 LB	278 42 STD	318 42 STD
039 08 08	079 16 A'	119 00 00	159 45 +	199 42 STD	239 14 B	279 09 09	319 09 09

17 ► NU HAR JAG TAGIT STRYK I NIM...

	4 2 1
	0
	0
	0
1	1
Summa	1

Maskinen avulater spelet genom att ta den sista stic-ken i hög 3.

Den första specialmaskinen som kunde spela Nim byggdes av Edward U. Condon som var chef för den fysika-liska institutionen vid Washingtonuniversitetet i St. Louis. Den patenterades 1940 under namnet Nimatron och ställdes ut på världsmutställningen i New York. Nimatronen vägde ett ton och var utrustad med drygt 4000 rellier. Den klarade av spel med 4 högar med upp till 7 stickor i varje hög.

Året därpå konstruerade Raymond M. Redheffer, biträdande professor i matematik vid Californiavarse-ritet i Los Angeles en betydligt förbättrad Nimmelspe-les maskin. Den hade samma kapacitet som Nimatron men n vägde bara 24 kilo och använde roterande strömbrytare i stället för rellier.

Längre fram konstruerades en robot som kallades Nim-rod. Den ställdes ut på Berlin-mässan 1951. Enligt en redogörelse av A. M. Turing var maskinen så populär i Berlin att det blev nödvändigt att kalla på polis för att hålla ett öga på maskinen. Maskinen blev ändå me-rra populär sedan den besegrat ekonomiminister Ludwig Erhard i tre raka partier. Erhard var som bekant näm-nen bakom det "tyska undret".

Min källa är en bok av Martin Gardner som heter Rolig matematik och kom ut 1960. Materialet hade desfer-förinnan publicerats i Scientific American. Boken som är förstå den av en serie på tre är en guldkgruva för den som är intresserad av matematiska gåtor. Den kan läsas på närmsta folkbibliotek. ■

BUDGET

AV MIKAEL NORDLIN

Detta korta, lilla program (160 pgmsteg) är betydligt mera kraftfullt än ett normalt program med samma längd. Detta beror på två saker: programmet anropar modulprogram (M/U 10: PGM 02, PGM 07) som subrutiner och utnyttjar TI-59:s hela arbetsminne genom minnesuppdelning 159.99 (10 OP 17), vilket givetvis sker automatiskt. Anrop av modulprogram i huvudprogram har ju både för- och nackdelar; fördelen som expanderat arbetsminne och snabbare programsekververing i modul anser jag övervinna nackdelar som tillgång till modulen i fråga och nödvändigheten av att rätt modul sitter i. Har övriga världen kommit till annan slutsats än jag, av den låga frekvens av publicerade program med modulanrop att döma? Möjligheten till modulanrop är ju ändå TI-59:s stora styrka, och var ju åtminstone tidigare unikt för apparaten i fråga. Detta program utnyttjar Maths/Utilities-modulen, vilket torde vara den mest allsidigt användbara och mest spridda modulen (bortsett från Master Library förstås). Förhoppningsvis har detta program därför inte enbart akademiskt intresse, utan finner också sina användare.

Vad gör då programmet? Det är avsett att användas i löpande bokföring, för att när som helst ange hur verksamheten förhåller sig till budget. Det medger arbete med 30 konton i taget, vart och ett av dessa kan ökas eller minskas och jämföras med budget. Denna kontogrupp om 30 bokföringskonton summeras automatiskt efter varje gång som man har arbetat färdigt med programmet. Den uppdaterade kontoplanen lagras på magnetkort (4 sidor för varje kontogrupp om 30 konton), för att sparas till nästa gång programmet ska användas. Kontona adresseras i stigande ordning med steg ett från och med ett. Första gången programmet används läses ingående balans för varje konto i tur och ordning in i minnet genom att först trycka A för att initiera budgetinmatning eller först trycka C för att initiera inmatning av saldo för varje konto, därefter (med värdet, som ska matas in i minnet, i displayen) R/S. Kontoadressering sker automatiskt i stigande ordning med början vid konto ett.

När 30 konton finns inmatade skrivs "OVERFLOW" ut med PC-100, som måste vara ansluten. Budget ändras ju normalt inte under verksamhetsåret, varför den har tilldelats en andrafunktion (A²). För kontering, d.v.s. förändring av enskilda konto, initieras saldokolumnen återigen med C, därefter anges kontots nummer i displayen varefter D trycks ned. PC-100 skriver då ut kontots nummer samt tidigare saldo. Ökning/minskning matas in i displayen (vid minskning värdet följt av +/-) varefter R/S trycks ned. Förändringen skrivs ut, adderas till kontot varefter nytt saldo skrivs ut. Detta upprepas från D för varje ny kontering. När alla konteringar för denna gång är gjorda ska data lagras på magnetkort och eventuellt ny kontoplan skrivas ut. Detta sker genom att tangenten E, betecknad "SAVE", trycks ned. Denna operation innebär att saldo- och budgetkolumnerna adderas, därefter divideras varje kontos saldo med kontots budget, och resultatet multipliceras med 100 för att anges i procent. Detta värde, "Index", lagras i en egen kolumn, varje konto för sig. Kven summa saldo och summa budget för kontogruppern divideras och uttrycks i procent, varför man även får hela kontogrupperns utfall i förhållande till budget, sist i indexkolumnen. När detta arbete, som tar sin tid, är färdigt skriver PC-100 "INSERT CARD 4". När denna upmaning har följts skrivs "INSERT CARD 3" ut, o.s.v. tills alla fyra block har skrivits, och minnesinnehållet överförs till magnetkort (två st.). När sista kortsidan har förts in i magnetkortsprungan skriver printern "PRESS RST R/S". Denna upmaning kan ignoreras om man inte önskar den uppdaterade kontoplanen utskrivin. Följer man upmaningen skrivs "RESULT", varefter följer "ROW 1" där saldokolumnen listas, sedan följer "ROW 2" med budgetkolumnen, och sist "ROW 3" med indexkolumnen.

Dessa tre kolumner kan säras åt och placeras bredvid varandra för överskådlighetens skull. Hur det kan se ut visar bilden med exemplet, hämtat från en bostadsrättsförening. I denna bostadsrätts-

förening, där jag är ordförande, har jag använt programmet på kostnadsidan enbart, och har tagit med endast de kostnader som går att påverka. Som sådant har programmet visat sig vara till stor hjälp i "kostnadsjakten" under verksamhetsåret, och inte minst i arbetet inför 1983 års budget.

Till nästa år hade jag tänkt att använda programmet även på intäktsidan. Om kontona ska utvidgas på detta vis med ytterligare 30 (intäkts-)konton, är det lämpligast att numrera dessa från 31 - 60. Detta går alldeles utmärkt, i programmet för intäktskontogruppern (på två separata magnetkort) ändras endast "Nop" i programmet till "3". Sålunda kan programmet "skräddarsys" till att passa just det företag eller den förening där det ska användas. Vill man använda en tredje kontogrupp ändrar man "Nop" till 6 o.s.v. Vill man ha fler än tre kontogrupper behöver man förstås ta bort någon "prompt" som man bedömer som onödig ("RESULT" t. ex.) eller på annat sätt spara in ett programsteg för att få

plats med det tvåstelliga tal som ska in i programmet i stället för "Nop". Det torde inte vara omöjligt med "tätare" programmering för att spara in ett programsteg någonstans. Märk dock att programmet innehåller direktadresser. Den direktadress som finns i sekvensen "PGM 02 SBR 078" behöver självklart inte ändras, den är ju ett exempel på direktadressering i modul (den subrutinen står inte medtagen i programboken som medföljer modulen). Rent generellt kan sägas följande om användande av modulanrop som subrutiner i huvudprogram: Kolla först vad den önskade modultitulen gör, ibland är det inte särskilt imponerande. "PGM 07 A" t. ex. lagrar endast konstanten i R02, varför det ofta kan vara en fördel att titta på modultitlarna genom att ladda arbetsminnet med modulprogrammet (OP 09). På det sättet kan man också hitta andra nyttiga rutiner genom direktadressering, sekvensen "PGM 02 SBR 078" upptäcktes på det viset. Nog sagt, mycket nöje!

BUDGETUPPFÖLJNING - MODIFIERAD KONTOPLAN, HSB (BOSTADSÄRTSFÖRENING)

	SALDO	BUDGET	INDEX
1. Underhåll, mark löpande	(32100) 34310.81	160000.00	21.44
2. Underhåll, byggnader löpande	(32300) 84741.27	80000.00	105.23
3. Underhåll, elinstallatörer	(32311) 15066.28	25000.00	60.27
4. Underhåll, hissar	(32315) 70798.00	65000.00	108.92
5. Underhåll, garage	(32321) 145482.34	300000.00	48.83
6. Underhåll, VVS	(32326) 2271.32	20000.00	11.36
7. Underhåll, tvättutrustning	(32331) 12362.13	25000.00	49.45
8. Underhåll, byggnader periodiskt	(32500) 813854.33	805800.00	101.00
9. Underhåll, P-platser	(32700) 9123.53	50000.00	18.25
10. Underhåll, lokaler	(32800) 16971.97	10000.00	169.72
11. Sakföreskringar, självrisker	(33200) 83365.00	130200.00	64.03
12. Hyresföretfall, lokaler-P-platser	(33301) 201761.79	436400.00	69.12
13. Arvden HSB, HSB:s konster	(34100) 192780.80	218400.00	88.27
14. Hyreskostnader, föreningslokaler	(34500) 13703.70	4900.00	279.67
15. Byggnadskostnader	(34701) 46788.30	50700.00	92.28
16. Löver, extra anställda	(35111) 10913.58	10000.00	109.14
17. Arvden, förtroendevalda	(35112) 13700.00	19000.00	72.11
18. Personföreskringar	(35120) 6623.00	8000.00	82.79
19. Splittring	(36100) 327911.60	335000.00	97.88
20. Stökning	(36111) 370422.00	438000.00	84.57
21. Sotning	(36112) 15016.50	25000.00	60.07
22. Fastighetsaskötsel	(36900) 42207.02	625600.00	67.47
23. Fastighetsförvaltning	(36910) 123272.00	208900.00	59.01
24. Värme	(37100) 2410711.50	4362900.00	55.25
25. Vatten	(38100) 411043.00	555700.00	73.97
26. El	(38200) 775550.45	1069900.00	72.49
27. Porto, HSB	(39400) 12261.50	12000.00	102.18
28. Fritidsverksamhet	(39714) 24396.73	43100.00	56.60
29. Studier	(39800) 2090.00	10000.00	20.90
30. Övrigt	(39930) 20386.94	30000.00	67.96
Summa, ovan redovisade konton:	13579004.55	20269400.00	66.09

Avancerade programmeringsmetoder -3

av BJÖRN GUSTAVSSON

På grund av tidsbrist har jag inte skrivit den artikel om sortering som jag tidigare utlovat. Beträffande artikelserien "Avancerade programmeringsmetoder" kan flera artiklar komma som jag får tid eller lust att skriva något. Andra författare som har artiklar inom det här ämnesområdet kan få publicera dessa under rubriken "Avancerade programmeringsmetoder".

Bär kommer nu en översättning av en del av kapitel 1.4.1 i *The Art of Computer Programming*, volym 1, av D.E. Knuth. Det är något av det bästa jag har läst om programutveckling.

I texten nämns några "MIXAL" och "PL/MIX". Dessa är två assemblerpråk för den mytiska datorn "MIX". Knuth skriver följande om MIX: "På många ställen i den här boken hänvisas till MIX: 'då många knappar'. Mänskens som ut använder är en mytisk dator kallad 'MIX'. MIX är väldigt lik nästan alla datorer som finns nu, förutom att den, kanske, är trevligare. MIX:s språk har utformats för att vara tillräckligt kraftfullt så att korta program kan skrivas för de flesta algoritmerna, men samtidigt tillräckligt enkelt så att dess operationer lätt kan läsas."

MIXAL beskrivs i kapitel 1, medan PL/MIX kommer att beskrivas i kapitel 9 (Dessa är utgivna).

Vi avslutar den här avdelningen med en diskussion om hur vi kan bära oss till att skriva ett långt och komplicerat program. Hur kan vi bestämma oss för vilken sorts subrutiner som behövs och vilka anropssvensor som bör användas? En främgång metod för att bestämma det är en iterativ procedur:

Steg 0 (En första idé). Först bestäms vi vagt den allmänna angreppsmetoden som skall användas i programmet.

Steg 1 (En grov skiss av programmet). Vi börjar nu skriva programmet utifrån nivåer, i något lämpligt språk. En någorlunda systematisk metod för att göra det har beskrivits väldigt bra av E.W. Dijkstra, *Structured Programming* (Academic Press, 1972), kapitel 1, och av N. Wirth, *Communications of the Association for Computing Machinery* 14 (1971), 221-227. Vi kan börja med att bryta ned hela programmet till ett litet antal delar, som för tillfället kan betraktas som subrutiner fastän de endast anropas en gång.

Dessa delar förflyter successivt till mindre och mindre delar som på sin tur kan betraktas som enkla och enkla uppgifter. När något inträffar som troligen kommer att inträffa någon annanstans eller som redan har inträffat någon annanstans definierar vi en subrutin (en riktig sådan) som utför uppgiften. Vi skriver inte subrutinen, utan bara förskärter att skriva huvudprogrammet, förutsättande att subrutinen har utfört sin uppgift. När slutligen huvudprogrammet har skissats ut vi

hand om subrutinerna i tur och ordning, under det att vi försöker behandla de mest komplicerade subrutinerna först och sedan deras subrutiner, osv. På det sättet får vi en hel lista med subrutiner. Varje subrutins faktiska funktion har troligen redan skrivits flera gånger så att de första delarna av skissen är inkorrekt. Men det är inget problem eftersom det endast är en skiss. Om varje subrutin vet vi nu någorlunda hur den skall anropas och hur allmänheten den skall vara. Det innebär sig ofta att göra varje subrutin lite mer allmännyttig.

Steg 2 (Första fungerande program). Detta steg utförs i motsatt riktning jämfört med steg 1. Vi skriver nu i ett datorspråk, t.ex. MIXAL eller PL/MIX. Den här gången skrivs subrutinerna på den lägsta nivån först och huvudprogrammet sist. I så stor utsträckning som möjligt bör vi försöka att inte skriva några instruktioner som anropar en subrutin som ännu inte har skrivits. (I steg 1 eftersträvades motsatsen: att inte bry sig om den subrutinen förrän alla anrop till den hade skrivits.)

Allt eftersom fler och fler subrutiner skrivs under den här processen, ökar vårt självförtroende efter hand eftersom vi allt mer tydligt utökar förmågan hos maskinen vi programmerar. Efter att varje enskild subrutin är skriven bör vi omedelbart göra i ordning en fullständig beskrivning om vad den gör och hur den anropas. Det är också viktigt att inte använda samma temporära minnescell för flera subrutiner. Det kan mycket väl vara katastrofalt om alla subrutiner använder cellen TEMP, även om det är beaktat att ströva att det inte vi gör skissen i steg 1. Ett självklart sätt att lösa problemet består i att låta varje subrutin endast använda sina egna temporära minnesceller, men om det innebär för mycket slöseri med minne så fungerar systemet med att numera cellerna TEMP1, TEMP2, ... ganska. Numreringen inom en subrutin startar med TEMP1, där jag är något högre än det högsta tal som används av någon av subrutinerna, subrutinerna.

Steg 3 (Omprövning). Resultatet från steg 2 bör vara väldigt nära ett fungerande program, men det kan fortfarande finnas några detaljer. Ett bra sätt är att byta riktning igen och för varje subrutin studera alla anrop som görs till den. Det är mycket möjligt att subrutinen borde utökas för att göra en del väsentligt förekommande åtgärder som utförs av den anropande rutinen alldeles före eller efter anropet. Kanske bör flera subrutiner slås ihop till en - eller en subrutin kanske bara anropas en gång (eller om vi har tur kanske en ändrad anropas) och bör inte alls vara en subrutin.

Vid det här tillfället är det ofta en god idé att kassera allt och börja om från steg 1. Detta är inte avsiktligt, kanske innefattar en anmärkning. Tiden som använts för att komma så här långt är inte bortkastad för vi har lärt oss en hel del om att skriva ett sådant program. Det är också viktigt att vara säker på flera förbättringar som kan göras i programets organisation. Det finns inget anledning att vara rädd för att gå tillbaka till steg 1 - det är mycket lättare att utföra ovanstående steg igen när ett program redan har gjorts. Dessutom kommer vi troligen

TI Programmer

av Björn Gustavsson

TI Programmen kan för 4-5 år sedan. (Numera finns också en LCD-version.) Den är inte programmerbar som namnet kanske kan antyda. Namnet syftar i stället på den av de avancerade för programmerare av datorer. Den gör nämligen omvandling mellan de decimala, oktala och hexadecimala talsystemen vilket är mycket användbart vid speciellt assemblerprogrammering.

Den gamla TI Programmer liknar utvändigt TI-30 och har liksom den strömsparret. När den är inställd i decimal läge fungerar den som vanliga räknare i läppraskissen med de fyra räknesätten. ADS saknas, men man kan specificera ordningen att utföra beräkningarna i med parenteser upp till fyra vilande operationer. Några möjligheter att använda tiopotensnotation finns inte. Det finns ett register som man kommer åt med tangenterna STO, RCL och SHD. Dessutom finns konstant.

Intressantare blir det när man går över till oktalt eller hexadecimalt läge. Man använder tangenterna DEC, HEX och OCT för att byta talsystem. Vid byte till ett nytt talsystem omvandlas talet i displayen till motsvarande tal i det nya talsystemet (endast heltal kan omvandlas). För att man skall veta vilket talsystem som är inställt visas "n" HEX är inställt och "d" OCT är inställt.

Exempel: För att omvandla talet 2512 (251 decimal) till hexadecimalt trycker man DEC (endast om decimal läge inte redan är inställt) 2512 HEX och får resultatet "FB". På grund av att displayen är av vanlig sjusegmentstyp kan de hexadecimala siffrorna "B" och "D" inte visas på vanligt sätt utan de visas i stället som "b" resp "d".

Man kan göra beräkningar med tal i alla tre baserna och byta talsystem mitt i beräkningen. Exempel: Beräkna 652800 (65280 decimal) * 127 (127 hexadecimalt) och uttryck resultatet.

att spara lika mycket tid för "debugging" som det tar att skriva ett program. Det är därför bäst att programmera som någon som skrivs har en stor del av sin främring att tacka det faktum att ingen av de viktigaste delarna av arbetet oavsiktligt förlorats och författarna måste börja på igen.

A andra sidan så när man troligen aldrig den punkt när det komplicerat program inte kan betraktas ytterligare som ett program, så steg 1 och 2 bör inte uppmärksam i oändlighet. Se den ytterligare diskussionen i kapitel 9. När betydelsefulla förbättringar har skedd, gick det göras är det väl värt den extra tid som är nödvändig för att börja om, men slutligen är man på punkt med ständigt minskande vinster.

Steg 4 (Debugging). Efter en sluttlig inövning av programmet, kanske innefattande tilldelning av minnesutrymme och andra saker, bör man i steg 4 börja att titta på programmet i sin helhet. Detta kan göras för hand eller naturligtvis på en dator. Författaren har funnit det vara till stor hjälp vid det här steget att använda så kallade "trace" (trace) varje instruktion de första två gångerna

test decimalt. Tryck DEC 65280 + HEX d7 = och Fd7 visas. Tryck DEC för att visa svaret (65495) decimalt.

I de oktala och hexadecimala talsystemen finns ytterligare funktioner tillgängliga. "1% C" (ett vanligare namn är NOT) beräknar ett komplementet för talet i displayen, vilket innebär att alla bitar (binära siffror) i talet inverteras (ettor blir nollor och nollor blir ettor). "+/-" beräknar tvåkomplementet för talet i displayen (ettkomplementet ökat med ett), som är det vanligaste sättet för att representera negativa tal i datorer.

Följande logiska operationer finns: AND, OR, XOR och BNF, de tre första har samma funktion som en i mikroprocessor. SHF motsvarar en mikroprocessors skiftinstruktioner och skiftar alltså ett tal ett antal bitar åt höger eller vänster. Man anger vänterskift med ett positivt tal och högerskift med ett negativt tal. Exempel: HEX AF SHF 3 = skiftar det hexadecimala talet AF 3 bitar åt vänster och ger 57B. HEX SHF +/- skiftar talet AF 3 bitar åt höger och ger 15.

TI Programmer saknar ett binärt läge, men det är ändå relativt lätt att omvandla binära tal. Vid inmatning av binära tal kan man gruppera siffrorna i grupper om 3 och i så fall omvandla dessa till en oktala siffra (från 0 till 7) och mata in i oktalt läge. För att underlätta detta finns under varje siffer-tangent motsvarande binära tal tryckt. Vid omvandling till binära tal omvandlar man första talet till oktalt och gör sedan om varje oktala siffra till 3 binära siffror.

Med ett enkelt knep beskrivet i instruktionsboken kan man även omvandla tal med decimaler för att t ex få reda på hur pi uttrycks hexadecimalt (5,243876).

den utförs Det är viktigt att tänka över programmet bakomliggande idéer och kontrollera att allt verkligen uppfyller sig som väntat.

Debugging är en konst som behöver studeras mycket mer, och metoden att angripa den är i hög grad beroende av de hjälpmedel som finns till varje datorsystem. En bra början till effektiv debugging är ofta att förbereda tillräckligt många testfall, vilket diskuteras i kapitel 9. De mest effektiva debugging-metoderna är att skriva ett program som konstruerade och inbyggda i själva programmet, sådana av de bästa programmerare använder nästan hälften av sina program till att underlätta debugging av den andra hälften. Den första hälften, som vanligen består av rakt på sak skrivna rutiner som visar relevant information i en lätt läsbart form, tas bort på slutet, men slutresultatet är en förvärande produktivitetssökning.

En annan god vara består i att föra anteckningar över alla gånger. Även om detta troligen är ganska pinsamt, så är sådan information ovärderlig för den som forskar i debugging, och det kommer också att hjälpa en att lära sig hur man gör färre antal fel i fortsättningen.

Beräkning av KONTROLLSIFFRAN

och "extra utmaning"

Den extra "utmaningsuppgiften" på s. 12 i PB 82-2 gav inte någon större genklang, kanske beroende på att jag skrev att jag lyckats lösa uppgiften på 26 steg vilket nog var ganska bra. Till TI PPC Notes kom det in ett trettiotal lösningar, varav 13 publicerades, men ingen av dem var kortare än 30 steg. Men så kommer Hans Lindgren i Lund med den här eleganta lösningen på bara 19 steg (!). Och vad som är intressantare, han fortsätter själv resonemanget och går över till våra statliga svenska metoder för beräkning av kontrollsiffran (personnummer och postgironummer t ex) och kommer också där ner till ett mycket kort program - 39 eller 40 steg! (Jag hade själv inte observerat sammanhanget mellan "utmaningsuppgiften" och svenska kontrollsiffror!) Efter Lars Kristianssons program för kontrollsiffran i PB 81-4 kom det i våras ett brev från Erik Sörensen i Hjörring i Danmark med en ganska lång utredning och ett snabbt program - 5,6 s. Av uttrymmesskäl fick det emellertid stå över, men nu tar vi in det tillsammans med Hans Lindgrens båda program.

Så här skriver Hans om "utmaningsuppgiften":

Angående LHis "extra utmaningsuppgift" i PB 82-2. Slå upp sidan 12 och se på de fem första K-talen i varje kolumn. För varje nytt K-tal bär de med 8, om man enbart ser på entalsiffran efter summering. Slått upp funktionen $y \times 8$. Gör man vidare till K-tal nummer sex märker man att ökningen bara blir 7 från föregående K-tal. Funktionen modifieras till $y \times 8 - 0,1x$. En avrundningstecknet förs på, vilket ger $y \times 8 - 0,1x + 0,5$. Efter testning av y (tex med $x=24$ och $x=25$) får man till sist en korrekt funktion där $y \times 8 - 0,1x + 0,5 - 0,17,9x + 0,4$. För snabbare hantering i programmet använder jag $y = 0,79x + 0,04$ där K-talet återfinns som första decimalen i y (som att tidigare ha varit entalsiffran). Programmet blir sedan väldigt enkelt.

```
000 76 LBL      010 95 =
001 11 A        011 22 INV
002 65 x        012 29 INT
003 93 x        013 65 x
004 07 7        014 01 1
005 09 9        015 00 0
006 85 +        016 95 +
007 93 x        017 59 INT
008 00 0        018 92 RTN
009 04 4
```

Hans Lindgren

Och därefter en översättning (fråmt erforderlig för att få rätt spaltbredd, min bästa Erik!!) av Erik Sörensens bidrag om kontrollsiffran enligt modul-10-metoden:

Modul 10-kontroll kan utföras utan att användaren behöver jämföra kontrollsiffror. Numret $abcdef-ghij$ är fel, om summan $2a + b + 2c + d + 2e + f + 2g + h + 2i + j$ inte har 0 som sista siffra. 2n (med streck under) betyder tvåvarsumman av 2n.

Ett prov med 630307-5052 (exemplet i PB 81-4 s. 10) ger $1+2+3+0+3+0+7+1+0+2 = 20$. Då summan 20 har 0 som sista siffra är kontrollsiffran rätt.

Man kan slipa de besvärliga tvåvarsummeberäkningarna som ju ger längre och långsammare program genom att man räknar med "pseudotvåvarsummor" genom att addera 5 till varannan siffra eller riktigare att addera talet 505050-5050: 630307-5052 + 505050-5050 = 1135358-0102. Nu adderas varannan siffra av resultatet: $1+3+3+8+1+2 = 18$. De överhoppade siffrorna adderas också och multipliceras med 2: $1+5+5+0+0 = 11$ och $11 \times 2 = 22$. Sista 2 + 18 ger 40. Slutresultat är noll - kontrollsiffran var alltså riktig.

Om man emellertid adderar 505050-5050 på en gång på det här sättet så blir det komplikationer om endera eller båda av f och h är 9. Om 630309-5951 testas efter denna metod får man 34 som inte slutar på 0 fastän numret är riktigt.

I programmen härnäst undvikes denna svårighet genom att talet 5 adderas till varje särskild siffra. Användaren matar in personnumret med decimalkomma på streckets plats och trycker A. Om numret är riktigt visas 0 i displayen. Annars visas det inmatade talet blinkande så man direkt kan se om inmatningen varit felaktig.

Det kortare (vänstra) programets körtid är 5,6 s. Subrutinen B behåller två siffror och genomsnittet före gånger (steg 029 - 033). Eftersom programmet bara har 56 steg lämpar det sig särskilt väl för TI-58.

Ett program utan subrutin klarar körningen på 4,4 s. Det användes likadant, men felaktiga nummer ges utan blink. Om man kör det i fast mode går körtiden ner till 2,4 s. Siffrorna tas fram i följande ordning: fedcba-ghij. Talet 5 adderas inte till siffrorna c och e så dessa kan vara högst 1 resp. 3. Man initierar fast mode genom att låsa kortet och trycka RST R/S. Därefter läses kortet igen. Mata in personnumret och tryck R/S (inte A i fast mode!).

Ett av följande nummer är fel: 170312,0558; 501229,4327; 090718,3734 och 351030,3997. Decimalkomma har använts i stället för streck eftersom talen ju måste matas in så.

```
000 76 LBL      000 00 0      060 01 1
001 12 8        001 00 0      061 95 =
002 01 1        002 00 0      062 59 INT
003 00 0        003 00 0      063 44 SUM
004 85 +        004 00 0      064 01 1
005 93 x        005 36 PGM      065 93 LCL
006 05 x        006 02 02      066 00 00
007 93 x        007 71 SBR      067 22 INV
008 45 x        008 02 02      068 59 INT
009 59 INT      009 29 29      069 95 +
010 44 SUM      010 02 02      070 95 =
011 01 01      011 00 0      071 95 =
012 01 01      012 00 0      072 95 =
013 00 0        013 28 FIC      073 65 x
014 00 0        014 00 0      074 95 =
015 59 INT      015 00 00      075 44 SUM
016 44 SUM      016 14 14      076 95 +
017 02 02      017 91 R/S      077 01 1
018 92 RTN      018 14 14      078 59 INT
019 76 LBL      019 11 R      079 65 x
020 11 A        020 00 0      080 95 =
021 47 CHS      021 00 00      081 44 SUM
022 42 STD      022 42 STD      082 02 02
023 00 00      023 42 STD      083 01 1
024 95 +        024 01 01      084 00 00
025 05 x        025 65 x      085 95 +
026 22 INV      026 93 93      086 53 x
027 28 LDG      027 01 1      087 05 =
028 35 +        028 65 x      088 95 +
029 12 8        029 59 INT      089 65 x
030 12 8        030 42 STD      090 59 INT
031 12 8        031 02 02      091 44 SUM
032 12 8        032 95 x      092 00 01
033 12 8        033 01 1      093 01 1
034 01 1        034 00 0      094 00 0
035 85 +        035 39 INT      095 85 =
036 43 RCL      036 44 SUM      096 59 INT
037 01 01      037 01 01      097 44 SUM
038 25 x        038 92 92      098 02 02
039 01 1        039 01 1      099 44 SUM
040 00 0        040 00 0      100 01 01
041 85 +        041 59 INT      101 01 1
042 43 RCL      042 44 SUM      102 01 01
043 05 x        043 02 02      103 01 1
044 55 02      044 93 93      104 95 =
045 00 0        045 1 1      105 02 02
046 95 x        046 65 x      106 02 02
047 59 INT      047 59 INT      107 12 INV
048 44 SUM      048 44 SUM      108 05 =
049 44 CP      049 44 CP      109 59 INT
050 67 EQ      050 93 1 1      110 22 INV
051 00 00      051 00 00      111 59 INT
052 18 18      052 85 +      112 29 CP
053 43 RCL      053 05 x      113 47 01
054 00 00      054 95 +      114 00 00
055 13 C        055 44 SUM      116 61 STD
056 00 00      056 21 21      117 00 00
057 00 00      057 65 x      118 14 14
058 93 x        058 93 x
```

Och så till sist Hans Lindgrens allmänna program för kontrollsiffror enligt modul-10-metoden:

Som ett komplement till LHis "extra utmaningsuppgift" följer här ett program som utnyttjar mitt första. Jag utnyttjar här ett K-talet för 1234567 är lika med entalsiffran i summan av K-talen för 01, 23, 45 och 67. Programmet biter av två siffror åt gången och exekveringen ligger på ca 6 sekunder för tiotusiffriga tal. Vill man minska tiden, så kan P-R ändras till HIR 08.

```
000 76 LBL      015 00 0      030 65 x
001 11 A        016 85 +      031 82 HIR
002 32 xIT      017 93 x      032 68 68
003 25 CLR      018 04 4      033 82 HIR
004 37 P-R      019 95 =      034 18 18
005 32 xIT      020 59 INT      035 22 INV
006 65 x        021 82 HIR      036 59 INT
007 93 x        022 38 38      037 95 =
008 00 0        023 25 CLR      038 92 RTN
009 01 1        024 22 INV
010 65 x        025 77 CC
011 59 INT      026 00 00
012 32 xIT      027 05 05
013 07 7        028 01 1
014 09 9        029 00 0
```

PS! Glöm inte ändra adressen på steg 026-027 om du ändrar i programmet.

Hans Lindgren

Johannes Sjögren har sänt in nedanstående program för kontrollsiffran på TI-57. Blir kanske även detta en utmaning för någon att göra någon annan variant för den räknaren?

Bruksanvisning:

1. Mata in de 6 första siffrorna (före strecket) och tryck SBR 4.

2. Mata in återstående siffror (ej kontrollsiffran) och tryck SBR 4. Efter en stund visas kontrollsiffran.

```
00 86 4 LBL 4      25 65 -
01 32 2 STD 2      26 86 1 LBL 1
02 03 3          27 01 1
03 32 0 STD 0      28 00 0
04 00 0          29 -39 2 INV PRJ 2
05 32 5 STD 5      30 -33 2 RCL 2
06 61 1 SBR 1      31 -49 1 SBR 1
07 61 0 SBR 0      32 -34 2 INV SUM 2
08 61 1 SBR 1      33 -42 EE
09 34 5 SUM 5      34 01 1
10 61 1 SBR 1      35 -42 INV EE
11 61 0 SBR 0      36 85 =
12 33 7 RCL 7      37 -61 1 INV SBR
13 32 2 STD 2      38 86 0 LBL 0
14 86 3 LBL 3      39 38 2 EXC 2
15 61 1 SBR 1      40 32 6 STD 6
16 34 5 SUM 5      41 02 2
17 61 1 SBR 1      42 39 2 PRJ 2
18 61 0 SBR 0      43 61 1 SBR 1
19 56 DSZ         44 34 5 SUM 5
20 51 3 STD 3      45 33 6 RCL 6
21 33 5 RCL 5      46 38 2 EXC 2
22 32 2 STD 2      47 34 5 SUM 5
23 01 1          48 -61 1 INV SBR
24 00 0          49 00 0
```

TI SERVICE FLYTTAT TILLBAKA

TI Service är nu tillbaka igen i huvudkontorets byggnad med egen ingång från N. Hamnvägen. Telefon 63 07 10 har varit bortkopplad under flyttningen men bör fungera när detta nummer av PB är ute.

AV LARS WINTER

1	BOKFÖRING			7 Op 17	3
BAL.m.List	2debet	✓	BEL.u.koll		
BALANS	DEBIT	KREDIT	BELÖPP	SALDO	

Vid bokföring kan det vara svårt att få
allt att stämma både vägrätt och lodrätt.
Detta är ett program som är mera lagt på
att vara idiotsäkert än snabbt.
Programmet är gjort för skrivare, men
funnar även utan. Uppdelning 7 Op 17.

1. Numrera kontona från 1 till 29, debetkonton = udda nr, kreditkonton = jämna nr.
2. Mata in ingående balans i R01 t.o.m. R29.
3. Mata in kontonamnen i R31 t.o.m. R59.
4. Kolla balansen, tryck A (med lista) eller A (utan lista). Om summa debet inte = summa kredit skrivs 70FF ut och displayen blinkar.
5. Mata in numret på första postens kreditkonton, tryck C, resp. debetkonton, tryck B. Apparaten accepterar endast udda debetnr, resp. jämna kreditnr. Debetkontonnamnet skrivs ut till vänster.
6. Mata in beloppet för första posten, tryck D. Apparaten blinkar fram debet- kredit-kontonumret, kreditkontonamnet skrivs ut till höger vid beloppet.
7. Mata in ev. nya nr. för nästa posta kreditnr., tryck C, resp. debetnr, tryck B. Örfärdighet nr. behövs inte matas in.
8. Mata in beloppet för nästa post, tryck D (eller R/S).

9. Kollas saldo når på ønsket, t.ex. postgirobesked eller med månadsavskiftet (antag att här kontonr. 3 och 4 tryck 3 eller 4 E.

10. Saldo för nästa konto: tryck R/S.

11. Kolla balansen ibland: tryck A.

12. Utgående balans: tryck R (eller A).

13. Nu kommer det svårare: Avgör för varje konto om det ska in på resultaträkningen eller på balansräkningen!

Summa res. kredit - summa res. debet = Årets vinst debet.

Summa bal. debet - summa bal. kredit = Årets vinst kredit.

Årets vinst debet skall vara = Årets vinst kredit. annars har ett fel begått!

Programmet upptar block 1, steg 000-239.
Kontonamnen upptar block 3, reg. 31-59.
Balansen upptar block 4, reg. 1-29.
Programsteg 240-399, block 2, är lediga
(för t.ex. programmerad inmatning).
Lbl R/S får inte användas, eftersom den
ska ge felindikering.
De 2 Nop-arna efter Lbl A kan utbytas mot
valfri Fix, t.ex. Fix 2.

```

REG:      0 Räkneverk
udda      1-29 Debet-konton
jämna     2-28 Kredit-konton
          30
udda      31-59 Kontonamn debet
jämna     32-58 Kontonamn kredit
          60 Använt
          61 Ind. debet-konto
          62 Ind. kredit-konto
          63 Ind. kontonamn debet
          64 Ind. kontonamn kredit
          65 Summa kredit
66-69

```

26.136.36.12.		EX. RØD SØDKÆRNE:		TRYCK		PG	
26.136.36.12.	31				PRD70	2000	PG
32.136.36.12.	32						PG
33.220.000.00.	33	1982.	03	01			PG
34.136.36.12.	34						PG
42.137.31.26.	35			02	BNK	2000	PG
43.136.36.12.	36						PG
44.136.36.12.	37	14000.	04	04		211000.	PG
45.136.36.12.	38						PG
56.36.23.13.60.	39					0.	BET
57.36.23.13.60.	40					0.	PG
58.36.23.13.60.	41					0.	PG
59.36.23.13.60.	42					0.	PG
33.32.23.37.12.	43			08		1983.84	PG
33.32.23.37.12.	44			09			PG
33.32.23.37.12.	45			10	TRYCK	10000.	PG
33.32.23.37.12.	46			11			PG
33.32.23.37.12.	47			12	PRD70	2000.	PG
33.32.23.37.12.	48			13			PG
33.32.23.37.12.	49			14	SKATT	1000.	PG
33.32.23.37.12.	50			15			PG
33.32.23.37.12.	51			16	MYSKR	2000.	PG
33.32.23.37.12.	52			17			PG
33.32.23.37.12.	53			18		1000.	PG
33.32.23.37.12.	54			19			PG
33.32.23.37.12.	55			20			BAL. 01
33.32.23.37.12.	56			21		76000.	PG
33.32.23.37.12.	57			22		76000.	PG
33.32.23.37.12.	58			23		76000.	PG
33.32.23.37.12.	59			24	RES.	13000.	PG
33.32.23.37.12.	60			25		8000.	PG
33.32.23.37.12.	61			26		1000.	PG
33.32.23.37.12.	62			27		1000.	PG
33.32.23.37.12.	63			28		1000.	PG
33.32.23.37.12.	64			29		1000.	PG
33.32.23.37.12.	65			30		1000.	PG
33.32.23.37.12.	66			31		1000.	PG
33.32.23.37.12.	67			32		1000.	PG
33.32.23.37.12.	68			33		1000.	PG
33.32.23.37.12.	69			34		1000.	PG
33.32.23.37.12.	70			35		1000.	PG
33.32.23.37.12.	71			36		1000.	PG
33.32.23.37.12.	72			37		1000.	PG
33.32.23.37.12.	73			38		1000.	PG
33.32.23.37.12.	74			39		1000.	PG
33.32.23.37.12.	75			40		1000.	PG
33.32.23.37.12.	76			41		1000.	PG
33.32.23.37.12.	77			42		1000.	PG
33.32.23.37.12.	78			43		1000.	PG
33.32.23.37.12.	79			44		1000.	PG
33.32.23.37.12.	80			45		1000.	PG
33.32.23.37.12.	81			46		1000.	PG
33.32.23.37.12.	82			47		1000.	PG
33.32.23.37.12.	83			48		1000.	PG
33.32.23.37.12.	84			49		1000.	PG
33.32.23.37.12.	85			50		1000.	PG
33.32.23.37.12.	86			51		1000.	PG
33.32.23.37.12.	87			52		1000.	PG
33.32.23.37.12.	88			53		1000.	PG
33.32.23.37.12.	89			54		1000.	PG
33.32.23.37.12.	90			55		1000.	PG
33.32.23.37.12.	91			56		1000.	PG
33.32.23.37.12.	92			57		1000.	PG
33.32.23.37.12.	93			58		1000.	PG
33.32.23.37.12.	94			59		1000.	PG
33.32.23.37.12.	95			60		1000.	PG
33.32.23.37.12.	96			61		1000.	PG
33.32.23.37.12.	97			62		1000.	PG
33.32.23.37.12.	98			63		1000.	PG
33.32.23.37.12.	99			64		1000.	PG
33.32.23.37.12.	100			65		1000.	PG
33.32.23.37.12.	101			66		1000.	PG
33.32.23.37.12.	102			67		1000.	PG
33.32.23.37.12.	103			68		1000.	PG
33.32.23.37.12.	104			69		1000.	PG
33.32.23.37.12.	105			70		1000.	PG
33.32.23.37.12.	106			71		1000.	PG
33.32.23.37.12.	107			72		1000.	PG
33.32.23.37.12.	108			73		1000.	PG
33.32.23.37.12.	109			74		1000.	PG
33.32.23.37.12.	110			75		1000.	PG
33.32.23.37.12.	111			76		1000.	PG
33.32.23.37.12.	112			77		1000.	PG
33.32.23.37.12.	113			78		1000.	PG
33.32.23.37.12.	114			79		1000.	PG
33.32.23.37.12.	115			80		1000.	PG
33.32.23.37.12.	116			81		1000.	PG
33.32.23.37.12.	117			82		1000.	PG
33.32.23.37.12.	118			83		1000.	PG
33.32.23.37.12.	119			84		1000.	PG
33.32.23.37.12.	120			85		1000.	PG
33.32.23.37.12.	121			86		1000.	PG
33.32.23.37.12.	122			87		1000.	PG
33.32.23.37.12.	123			88		1000.	PG
33.32.23.37.12.	124			89		1000.	PG
33.32.23.37.12.	125			90		1000.	PG
33.32.23.37.12.	126			91		1000.	PG
33.32.23.37.12.	127			92		1000.	PG
33.32.23.37.12.	128			93		1000.	PG
33.32.23.37.12.	129			94		1000.	PG
33.32.23.37.12.	130			95		1000.	PG
33.32.23.37.12.	131			96		1000.	PG
33.32.23.37.12.	132			97		1000.	PG
33.32.23.37.12.	133			98		1000.	PG
33.32.23.37.12.	134			99		1000.	PG
33.32.23.37.12.	135			100		1000.	PG
33.32.23.37.12.	136			101		1000.	PG
33.32.23.37.12.	137			102		1000.	PG
33.32.23.37.12.	138			103		1000.	PG
33.32.23.37.12.	139			104		1000.	PG
33.32.23.37.12.	140			105		1000.	PG
33.32.23.37.12.	141			106		1000.	PG
33.32.23.37.12.	142			107		1000.	PG
33.32.23.37.12.	143			108		1000.	PG
33.32.23.37.12.	144			109		1000.	PG
33.32.23.37.12.	145			110		1000.	PG
33.32.23.37.12.	146			111		1000.	PG
33.32.23.37.12.	147			112		1000.	PG
33.32.23.37.12.	148			113		1000.	PG
33.32.23.37.12.	149			114		1000.	PG
33.32.23.37.12.	150			115		1000.	PG
33.32.23.37.12.	151			116		1000.	PG
33.32.23.37.12.	152			117		1000.	PG
33.32.23.37.12.	153			118		1000.	PG
33.32.23.37.12.	154			119		1000.	PG
33.32.23.37.12.	155			120		1000.	PG
33.32.23.37.12.	156			121		1000.	PG
33.32.23.37.12.	157			122		1000.	PG
33.32.23.37.12.	158			123		1000.	PG
33.32.23.37.12.	159			124		1000.	PG
33.32.23.37.12.	160			125		1000.	PG
33.32.23.37.12.	161			126		1000.	PG
33.32.23.37.12.	162			127		1000.	PG
33.32.23.37.12.	163			128		1000.	PG
33.32.23.37.12.	164			129		1000.	PG
33.32.23.37.12.	165			130		1000.	PG
33.32.23.37.12.	166			131		1000.	PG
33.32.23.37.12.	167			132		1000.	PG
33.32.23.37.12.	168			133		1000.	PG
33.32.23.37.12.	169			134		1000.	PG
33.32.23.37.12.	170			135		1000.	PG
33.32.23.37.12.	171			136		1000.	PG
33.32.23.37.12.	172			137		1000.	PG
33.32.23.37.12.	173			138		1000.	PG
33.32.23.37.12.	174			139		1000.	PG
33.32.23.37.12.	175			140		1000.	PG
33.32.23.37.12.	176			141		1000.	PG
33.32.23.37.12.	177			142		1000.	PG
33.32.23.37.12.	178			143		1000.	PG
33.32.23.37.12.	179			144		1000.	PG
33.32.23.37.12.	180			145		1000.	PG
33.32.23.37.12.	181			146		1000.	PG
33.32.23.37.12.	182			147		1000.	PG
33.32.23.37.12.	183			148		1000.	PG
33.32.23.37.12.	184			149		1000.	PG
33.32.23.37.12.	185			150		1000.	PG
33.32.23.37.12.	186			151		1000.	PG
33.32.23.37.12.	187			152		1000.	PG
33.32.23.37.12.	188			153		1000.	PG
33.32.23.37.12.	189			154		1000.	PG
33.32.23.37.12.	190			155		1000.	PG
33.32.23.37.12.	191			156		1000.	PG
33.32.23.37.12.	192			157		1000.	PG
33.32.23.37.12.	193			158		1000.	PG
33.32.23.37.12.	194			159		1000.	PG
33.32.23.37.12.	195			160		1000.	PG
33.32.23.37.12.	196			161		1000.	PG
33.32.23.37.12.	197			162		1000.	PG
33.32.23.37.12.	198			163		1000.	PG
33.32.23.37.12.	199			164		1000.	PG
33.32.23.37.12.	200			165		1000.	PG
33.32.23.37.12.	201			166		1000.	PG
33.32.23.37.12.	202			167		1000.	PG
33.32.23.37.12.	203			168		1000.	PG
33.32.23.37.12.	204			169		1000.	PG
33.32.23.37.12.	205			170		1000.	PG
33.32.23.37.12.	206			171		1000.	PG
33.32.23.37.12.	207			172		1000.	PG
33.32.23.37.12.	208			173		1000.	PG
33.32.23.37.12.	209			174		1000.	PG
33.32.23.37.12.	210			175		1000.	PG
33.32.23.37.12.	211			176		1000.	PG
33.32.23.37.12.	212			177		1000.	PG
33.32.23.37.12.	213			178		1000.	PG
33.32.23.37.12.	214			179		1000.	PG
33.32.23.37.12.	215			180		1000.	PG
33.32.23.37.12.	216			181		1000.	PG
33.32.23.37.12.	217			182		1000.	PG
33.32.23.37.12.	218			183		1000.	PG
33.32.23.37.12.	219			184		1000.	PG
33.32.23.37.12.	220			185		1000.	PG
33.32.23							

I TI-PPC har Palmer Hanson Jr och Bob Fruit behandlat frågan om man kan använda en reservkraftkälla till TI-59 exempelvis ett 6 volts batteri.

Ett mera vanligt förekommande problem med 59an är att en av de tre cellerna i ackumulatören (BP-1A) får en intern kortslutning som gör att den inte fungerar normalt. Det är ett ganska vanligt fel i ackumulatörer av nickel-kadmium typ. Detta fel leder till att laddning med laddningsadaptern eller i PC-100 inte kan upphäva kortslutningen. (Maurice E. T. Swinnen konstruerade därför en enkel krets (se nedan) som avhjälper felet)

Spänningen från en "fulladdad" felaktig ackumulator ligger något under 3 volt.
Den låga spänningen leder till följande:

a/ När räkneren används med bara ackumulatören ansluten, lyser displayen en aning svagt. Beräkningar utförs korrekt. Läsning och skrivning av magnetkort fungerar inte, om de ens går att utföra. Vid läsning av magnetkort visar displayen varierande resultat, exempelvis blinkande 2 siffriga tal.

b/ Med ackumulatören i skrivarens laddningsfack och räknaren ansluten på PC-100, fungerar räknaren normalt.

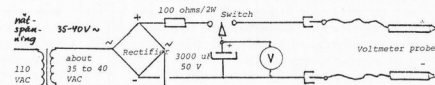
c/ Med den defekta ackumulatoren i räknaren och laddningsadaptern ansluten uppför sig räknaren som om enbart ackumulatoren var isatt.

Sedan Palmer Hanson Jr ursprungligen identifierade detta problem har han hjälpt flera vars räknares inte kunde skriva eller läsa magnetkort.

The Zapper av Maurice E. T. Swinnen

Overseas: Jan T. Jacobson

När jag, i samband med ovanstående problem, varit irriterad flera gånger, började jag undersöka vad som var orsaken till problemen. När jag plockade isär en defekt ackumulator fann jag en dendrit (ett litet kristallkorn). Det visade sig att dendriten var ledande och alltså orsaken till kortslutningen av ackumulatören. Genom att ge ackumulatören en stöt på ungefär 50 volt upplöstes dendriten. Så föddes "The Zapper".



Bilden ovan visar vad som behövs. En kraftkälla, antingen en man sätter ihop från en kaserad transformator och en likriktare, eller en existerande kraftkälla som kan ge ungefär 50 volt vid några få milliamperer. En elektrolyt-kondensator på 30000 30000 uf vid 50 volt kan lagra tillräckligt med energi för att få fart på den mest förkalkade ackumulatör.

En voltmeter är praktisk att använda för att se när kondensatorn är laddad och klar för nästa stöt. Den bör ha omkring 100.000 Ohm per volt inre impedans för att inte belasta den uppladdade kondensatorn för mycket.

Med hjälp av en strömbrytare kan man ladda upp kondensatorn långsamt genom att slå den mot vänster eller ge en av ackumulatorns tre celler en stöt genom att slå strömbrytaren mot höger.

För att kunna göra det borrar man 3 hål, 2 mm i diameter, på var sida av ackumulatören, MEN BARA GENOM PLASTEN. Detta för att komma åt ändarna, av metall, på varje cell. Voltmeters mäteladdar är praktiska för att ge cellerna en stöt genom de små hålen.

Ibland kan det vara nödvändigt att ge flera stötar. Ett bra tillvägagångssätt är att ge en krånglande akkumulator några stötar och sedan ladda den i PC-100 omkring en timme. Om den fortfarande inte går att ladda, ge den en stöt till. Jag har lyckats med ungefär 90 % av de över 40 akkumulatorer som jag provat metoden på.

PROGRAMMERBAR MODUL FÖR TI-59

Utom vår danske vän som knåpat ihop den modul som vi nämnde i PB 82-3 finns det en amerikan som tidigare gjort en Eprom för HP-41 som för någon månad utvecklat en också för TI-59. Vi har fått en (visserligen osäker) adress och återkommer kanske med mer upplysningar.

HEXKODER

AV ROBERT PRINS

ÖVERSÄTTNING OCH BEARBETNING: LARS HEDLUND

I "Hexkoden" i 82-3 skrev vi om att Robert Prins utredat hur man skulle kunna kontrollera de nya stegen efter en hexkod med den "modifika metoden". Här är Roberts egen beskrivning.

Hur kontrollerar man stegen efter en hexkod? Ja, jag vet inte det exakta svaret, men jag kan kontrollera de sex stegen närmast efter den skapade hexkoden. Jag tar ett exempel från ett eget program för alfabetiseringslistning (endast 0p-skivind!) Registrernumret och utskriften kommer på samma rad. Programmet kan lista alla register (inkl 00 som ej förlores!) i två steg (00-89 och 90-99) men även enskilda register. Programmet arbetar i Fast Mode ("TFM 96.2" = transparent Fast Mode med h12 på steg 096 skapad med 2 ins.). Jag vill ha följande efter steg 096: Nop 0 x: t10 G10. Först satte jag in den efter initieringen, men det är inte särskilt användarvänligt och så experimenterade jag litet och här är resultatet:

Jag startade med stegen i ①. Prt finns bara för att markera första steg i nästa otkod. Jag skapade h12 så här: 9 Op 17 CLR G0 096 Pgm 19 SBR 045 PR LRN Ins Lrn RST CLR. Här efter blev listningen enligt ②. Jag antog att precis kommande adresser till nollorna i steg 097-103 drossla som ... (redogjordes i början av artikel "Hexkoden"; red. anm.). Därför provade jag listning ③ och fick efter initiering listning ④. Uppenbarligen sker överföring av hundratalssiffror ("ett i minne"):

097 RST + B* + Nop (51+17-68)
098 G7 + B* + 0 (83+17-100)
099 48 + Op* + x: t + 1 (48+94+1=133) - x²
100 + + 76 + G0 + 1 (85+76+1=162) + Pgm
Varför "4" i ②/105 ändras till 54? Fråga inte mig, jag vet inte.

Naturligtvis blev mitt nästa försök stegen i ⑤. Som synes fick jag då rätta stegen i ⑥.

```

① 095 86 STF ② 095 86 STF ③ 095 86 STF ④ 095 86 STF
096 12 12 096 12 12 096 12 12 096 12 12
097 00 0 097 14 097 51 RST 097 48 NOP
098 17 8* 098 17 8* 098 83 G0 098 00 0
099 00 0 099 54 099 48 099 48 099 48
100 00 0 100 76 76 100 85 + 100 62 R*
101 00 0 101 00 0 101 02 02 101 02 02
102 00 0 102 23 LCL 102 83 G0 102 06 6*
103 00 0 103 00 0 103 00 0 103 00 0
104 00 0 104 00 0 104 99 FRT 104 100
105 04 4 105 04 4 105 04 4 105 99 FRT

```

```

⑤ 095 86 STF ⑥ 095 86 STF
096 12 12 096 12 12 096 12 12 096 12 12
097 51 RST 097 51 RST 097 51 RST 097 51 RST
098 83 G0 098 83 G0 098 83 G0 098 83 G0
099 48 099 48 099 48 099 48 099 48 099 48
100 00 0 100 01 01 100 01 01 100 01 01
101 00 0 101 00 0 101 00 0 101 00 0
102 00 0 102 00 0 102 00 0 102 00 0
103 00 0 103 91 RST 103 91 RST 103 91 RST 103 91 RST
104 00 0 104 00 0 104 00 0 104 00 0
105 99 FRT 105 99 FRT 105 99 FRT 105 99 FRT

```

- Av dessa exempel kan man få följande riktlinjer:
 - Sätt den kod som skall omvandlas på rätt steg och låt den följas av nollor.
 - Ge det erforderliga antalet ins som behövs för att skapa hexkoden.
 - Se vilka kodar som kommer på de följande sex stegen (Jag rekommenderar starkt bara sex steg eftersom tidigare versioner av mitt registerlistningsprogram alltid kraschade då jag försökte knappa in steg 097-105 manuellt.)
 - De sökta koderna skall vid addition till dessa ge de önskade koderna; ev. hundratalssiffror förs över till nästa kod.

Det var allt! (?)

- Så här använder man registerlistningsprogrammet:
- För R00 - R99. Läs in motsvarande kortslod.
 - Läs in alfabetiseringsprogrammet i normaluppladdning (6 op 17) och skapa h12 på steg 096: 9 Op 17 CLR G0 096 Pgm 19 SBR 045 PR LRN Ins Lrn RST CLR.
 - a) Ett enda register: Mata in registrernumret och tryck A.
 - b) En grupp av register från LL till HH: Mata in LL, hh och tryck A.
 - 3 För R90 - R99. Tvångsläs kortslod 1 med "-3" i displayen. Göd sedan som under 2a eller 2b men tryck i stället B.

```

000 00 0 060 12 8 120 82 HIR 180 03 3
001 92 R* 061 53 4 121 13 12 181 00 0
002 29 CP 062 46 INS 122 69 DP 182 06 6
003 01 1 063 79 5 123 04 04 183 00 0
004 52 EE 064 32 X:IT 124 82 HIR 184 00 0
005 01 1 065 06 6 125 04 04 185 33 32
006 03 3 066 00 0 126 69 DP 186 22 INV
007 82 HIR 067 00 0 127 20 20 187 07 FFF
008 07 07 068 22 INV 128 93 0 188 01 01
009 93 0 069 93 0 129 0 0 189 01 01
010 01 1 070 69 DP 130 49 FFF 190 96 96
011 00 0 071 10 10 131 49 FFF 191 52 52
012 82 HIR 072 85 4 132 00 0 192 93 93
013 84 64 073 93 1 133 03 3 193 82 HIR
014 75 7 074 06 6 134 04 4 194 54 54
015 93 075 54 3 135 03 3 195 82 HIR
016 82 HIR 076 61 G10 136 44 S0H 196 57 57
017 14 14 077 00 0 137 43 R55 197 43 R55
018 75 7 078 82 82 138 61 G10 198 00 00
019 99 INV 079 76 LBL 139 00 0 199 22 INV
020 85 4 080 11 8 140 02 02 200 59 INV
021 82 HIR 081 29 CP 141 82 HIR 201 52 HIR
022 54 54 082 48 EXC 142 06 06 202 02 2
023 9 9 083 00 0 143 00 0 203 22 INV
024 01 1 084 82 HIR 144 00 0 204 27 ENG
025 82 HIR 085 03 3 145 01 1 205 42 STD
026 00 0 086 00 0 146 00 0 206 00 00
027 82 HIR 087 69 DP 147 01 1 207 69 DP
028 00 0 088 00 0 148 00 0 208 00 00
029 82 HIR 089 93 17 149 82 HIR 209 32 X:IT
030 01 1 090 47 47 150 00 0 210 00 00
031 54 7 091 34 7 151 82 HIR 211 00 00
032 82 HIR 092 82 HIR 152 14 14 212 00 00
033 37 37 093 35 35 153 22 INV 213 59 INV
034 95 95 094 40 BEC 154 47 47 214 52 HIR
035 59 59 095 86 STF 155 00 0 215 02 2
036 95 95 096 12 12 156 02 02 216 22 INV
037 05 5 097 51 RST 157 82 HIR 217 57 ENG
038 54 3 098 83 G0 158 09 09 218 77 GE
039 82 HIR 099 47 47 159 43 RCL 219 01 01
040 37 37 100 84 DP 160 00 0 220 17 17
041 02 2 101 00 0 161 22 INV 221 06 6*
042 00 0 102 85 4 162 02 02 222 00 00
043 67 67 103 00 0 163 52 HIR 223 17 17
044 00 0 104 02 02 164 04 04 224 00 00
045 52 52 105 01 01 165 82 HIR 225 69 DP
046 82 HIR 106 11 11 166 04 04 226 36 36
047 14 14 107 86 STF 167 01 1 227 82 HIR
048 22 INV 108 01 01 168 82 HIR 228 13 13
049 16 16 109 43 RCL 169 54 54 229 42 STD
050 00 0 110 00 0 170 02 2 230 00 00
051 15 15 111 59 INV 171 02 2 231 00 00
052 54 4 112 42 42 172 44 S0H 232 00 00
053 22 INV 113 01 01 173 00 00
054 27 27 114 76 76 174 24 24 233 00 00
055 82 HIR 115 73 RCL 175 32 X:IT
056 17 17 116 00 00 176 16 16 234 00 00
057 83 G0 117 61 G10 177 00 00
058 00 0 118 01 01 178 02 02 235 00 00
059 76 LBL 119 24 24 179 53 53

```

UTLÄNDSKA KLUBBAR

av Björn Gustavsson

I PB 82-3 s 27 skrev jag om utländska TI-användarklubbar. Paul Schlyter tycker att jag också borde ha nämnt HP-användarklubben

PPC

2545 W. Camden Place
Santa Ana, CA 92704
USA

Medlemskap kostar \$37/år inklusive flygporto för medlemmar i Europa.

Klubben ger ut tidskriften PPC Journal, som tidningsformatet består av 65 Notes. De första nummerarna av HP-65 (den första programbara räknaren), och också om SR-52 när den kom. Sedan beslöts att 65 Notes endast skulle handla om HP-räknare och Richard Vanderburgh bildade då SR-52 Users Club (se Forra PB). Det går fortfarande att köpa gamla nummer av 65 Notes. Skicka ett SASE (Self-Addressed and Stamped Envelope) till PPC för mer information.

Nyligen har PPC av skatte tekniska skäl börjat skiva om TI-räknare igen. TI-avdelningen heter LBN och har hittills redigerats av Maurice Swinnen (om någon annan tar hand om det vet vi ännu inte). Det mesta materialet är hämtat ur TI PPC Notes, men det finns också andra intressanta artiklar ej hämtade därifrån: Syntetisk programmering på TI-55 (i Sverige kallad TI-51-III), skrivet av en HP-41C-användare! Ett program som översätter ett TI-59-program till ett HP-41C-program (alltså en omvandling RPN-simulator), gjord av Dejan Ristanović.

Vi har emellertid också fått kontakt med Innu en TI-59-förening, den tyska TI-58/59 Software Club. Den leds av Peter G. Poloczec med adressen Kaibacher Hauptstrasse 71, D-6000 FRANKFURT MAIN 56, Västtyskland. Årsavgiften är 30 DM, och då får man 5-6 nr av klubbladet, tillsammans ca 100 sidor. Klubben förmedlar också program för 5 DM/st. Vi har kommit överens med Peter om tidsnings- och programutbyte och kan alltså lämna mer upplysningar i nästa nummer.

FELTRYCK I MANUALEN FÖR STANDARDBIBLIOTEKET ML-20

En medlen som försökt använda formulera på s. 76 för programmering på dator fann att parenteserna i "TAC-TOR" för jan och febr, sista termen, inte stämde. En häpandesamt felaktig "1" (sist) och denna terms uppgivna är exakt lika termen två rader längre ner.

Stjärnskytte

Till det omvända stjärnskytte som föreslogs i PB 82-3 s 12 (man börjar med ett svart hål i mitten omgivet av stjärnor och skuter tillis man får en stjärna i mitten omgivet av svarta hål), har Gunnar SJUNNERYD, Mora, hittat följande lösning: 1, 7, 9, 5, 3.

Björn Gustavsson

P.S. Jan Björklund, Kungälv, har med papper och penna kommit fram till följande lösningar:

Från ett ensamt svart hål i mitten till en ensam stjärna i mitten (som ovan): 1, 3, 7, 5, 9.

Skjuta bort alla stjärnor: 5, 2, 8, 4, 6.

HOPTRYCKT TEXT PÅ RENSA

Som bekant händer det att remsan inte matas fram ordentligt när den första raden skrivs sedan remsan rivits av. Björn Gustavsson har ju rekommenderat i Utmaningen att man börjar alla program med ett "Adv". Det gör faktiskt TI Service också när vi hör med dem. Vad som sker är att remsmatningen kopplas ur om man drar i remsan. Olika maskiner har olika glapp och i kombination med olika avrinningsmekanis drabbas alltså några inte alla med andra inte klarar sig utan "Adv".

21 BUDGET

```

000 13 0 040 01 1 080 36 PGM 120 02 02
001 01 1 041 95 4 081 07 07 121 71 SEP
002 94 4 042 42 STD 082 11 8 122 00 00
003 85 4 043 02 02 083 03 3 123 87 87
004 03 3 044 03 3 084 36 PGM 124 36 PGM
005 07 07 045 02 02 085 07 07 125 02 02
006 12 8 046 36 PGM 086 12 8 126 13 13
007 47 852 047 91 8 087 36 PGM 127 91 8
008 00 0 048 07 07 088 07 07 128 76 LBL
009 00 0 049 02 02 089 31 SEP 129 31 SEP
010 01 01 050 59 PFT 090 50 50 130 86 STF
011 06 6 051 61 G10 091 01 1 131 05 05
012 22 INV 052 44 44 092 00 0 132 03 3
013 58 FIF 053 44 44 093 00 0 133 01 1
014 04 04 054 12 12 094 02 02 134 02 02
015 17 17 055 15 15 095 02 02 135 76 LBL
016 25 CLR 056 13 13 096 03 03 136 16 16
017 92 RST 057 01 1 097 07 07 137 08 08
018 36 PGM 058 36 PGM 098 07 07 138 06 06
019 07 07 059 07 07 099 11 1 139 00 00
020 13 0 060 15 15 100 36 PGM 140 02 02
021 99 PFT 061 42 STD 101 07 07 141 03 03
022 77 RCL 062 07 07 102 71 SEP 142 42 STD
023 04 04 063 22 INV 103 07 07 143 01 01
024 76 76 064 07 07 104 57 57 144 32 32
025 00 00 065 05 05 105 00 00 145 95 95
026 72 72 066 06 06 106 11 11 146 95 95
027 00 00 067 09 09 107 00 00 147 42 STD
028 02 02 068 02 2 108 42 STD 148 00 00
029 03 3 069 03 3 109 43 RCL 149 00 00
030 93 93 070 09 09 110 43 RCL 150 02 2
031 03 3 071 02 02 111 02 02 151 02 02
032 14 14 072 42 STD 112 36 PGM 152 04 04
033 04 04 073 42 STD 113 02 02 153 02 02
034 93 99 PFT 074 86 STF 114 16 16 154 01 1
035 42 42 075 82 HIR 115 06 06 155 02 02
036 43 RCL 076 01 1 116 06 06 156 06 06
037 87 87 077 00 0 117 00 00 157 07 07
038 75 75 078 00 0 118 10 10 158 00 00
039 02 2 079 02 2 119 36 PGM 159 14 14

```


ROMERSKA

SIFFROR

Här kommer ett program för att skriva romerska siffror. Det är taget ur 52-Notes, så det har redan några år på nacken, men det är roligt att titta närmare på och kan möjligen vara en utmaning för våra läsare att förbättra.

Programmet klarar tal upp till 3.999.999 och arbetar med de "moderna" romerska beteckningarna: I=1, V=5, X=10, L=50, C=100, D=500, M=1.000, V=5.000, X=10.000, L=50.000, D=100.000, V=500.000 och M=1.000.000. Om antalet tecken överskrider 20 trycks två rader.

Det finns som bekant andra romerska skrivsätt också, t ex (med parenteser för fram- och bakvänt C) (I)=1.000, ((I))=10.000, (((I)))=100.000 etc eller VIII för IX, IXX för XIX etc. Men det är kanske att vara verkligt finsmakare att ta med det i programmet?

Knappta in program och data och spela in tre sidor i normalupplösning. Obs att indelningen "7 op 17" tillfälligt behövs för att ladda de två sista minnesregistren.

31
32
1.
10000.
33
1000000.
34
100000000.
35
1.1
36
11.1
37
11.3
12.2
38
2.1
40
2.1
41
211.3
42
211.4
43
13.2
44
24.
45
22.
46
44.
47
25.
48
17.
49
16.
50
30.
51
42.2
52
44.2
53
27.2
54
15.2
55
16.2
56
30.2
57
3732320014.
58
2422000000.
59
413161721.
60
243117600.
61

MANUSSTOPP FÖR PB 83-1
DEN 1 MARS 1983.

MEDLEMANSANNONSER

JULKAPP?? TI 58 C säljes.

Jan-Erik Öberg, tel 0457-165 33 efter kl. 18.

KÖPES: PC 100 C

Jan Wikström, Ring helg dagar 0953/100 96

Yrkesmässig programförsäljning

BYGNADSTEKNIK

Kontinuerliga balkar, ramar, takstolar och fackverk, dimensionering av betong, stål och trä, strömsar, pålgrupper, grundplattor, plattor, 88K-program. Modular Structural Engineering och Baustatik II.

Carl-Adolf Granholm
Göteborgs gata
411 03 GÖTEBORG
tel 031 480 475

Lars Hedlund
Årstavägen 27
121 68 JOHANNESTAD
tel 08/91 38 97

GEODESI

Imåttning, utsättning, avskärning, tolkning, polygon-
tåg, avvägning, arealberäkning, läktemåttning, knut-
punktsutjämnning, administrativa program för databanken.

HÄNKDATA AB
Mäster Samuelsgatan 42
Box 7198
103 88 STOCKHOLM
tel 08/22 58 50

PROGRAMBIBLIOTEKET

Adress: Bo Nordlin, Stora Björnsens gata 70, 136 64 HANDEN.

BESKRIVNING AV PROGRAM

Den del av mallen som tas med i tidningen innehåller följande uppgifter: programbeskrivning, hårdvara nödvändig för utnyttjande av programmet, programmerarnas namn och datum för insändandet. Dessutom ger det åtta-siffriga identifikationsnumret, t ex 75593007, ytterligare upplysningar. De två första siffrorna betecknar ämnesområde enligt tabellen nedan. Nästa två siffror betecknar programmerarens namn eller avsett för. Den femte siffran betecknar källan:

- 1 = TISOT
- 2 = BTUC
- 3 = Föreningen Programbiten
- 8 = PPX

De tre sista siffrorna är ett löpnummer som programförmedlaren tilldelar.

INSÄNDANDE AV PROGRAM

Skriv programdokumentation på skrivmaskin eller läsligt för hand med svart skrift. För att göra hållslagnin möjlig skall marginalerna på samtliga papper vara minst 2 cm breda (undantag: monterade programlistningar). Dokumentationen skall som minimikrav bestå av: ifylld mall, bruksanvisning för körning och monterad listning. Skicka med en inspelad version av programmet om du har möjlighet. Vi returnerar magnetkort eller kassetter.

För varje program som antas få du ur Programbiblioteket välja antingen: 4 st svenska eller engelska program

utan magnetkort eller 3 st svenska TI-59-program med magnetkort eller 3 st belgiska program utan magnetkort. I stället för program kan du få ett gratis medlemskap när du har fått sammanlagt 6 program antagna.

Någon hittar ett fel i ditt program sedan det tagits in i biblioteket får du en uppmärksam att revidera det. Det reviderade programmet ersätter sedan det felaktiga programmet i biblioteket. Försök till revidering av ditt program av en annan programmerare betraktas som ett förslag och kan tas in i biblioteket som en revidering endast efter ditt godkännande.

BESTÄLLNING AV PROGRAM

Beställning till program sker genom inbetalning av aktuellt belopp till förenings postgiro 430 01 59-3. För varje beställning uttas en beställningsavgift på 10 kr. Endast svenska och engelska program lagras. Övriga beställs en gång i månaden från den utländska klubben.

Blanketten "Användarsynpunkter" följer med varje program. Fyll i den och skicka den till programförmedlaren om du har synpunkter på programmet. Synpunkterna kommer att vidarebefordras till programmeraren.

PRISLISTA

Svenska och engelska (BTUC) program utan magnetkort 10,-
Svenska TI-59-program med magnetkort 15,-
Belgiska program (TISOT) utan magnetkort 15,-
Katalog med belgiska och engelska program 20,-

BUSINESS	STATISTICS & PROBABILITY	NATURAL SCIENCES	ENGINEERING	SOCIAL & BEHAVIORAL SCIENCES
01 Management Accounting	20 Regression/Curve Fit	40 Physics	60 Aeronautical Engineering	80 Political Science
02 Manufacturing Engineering	21 Analysis of Variance	41 Chemistry	61 Chemical Engineering	81 Sociology
03 Inventory Control	22 Statistical Testing	42 Biology	62 Civil Engineering	82 Psychology/Physiology
04 Marketing/Sales	23 Statistical Inference	43 Agriculture	63 Computer Science	83 Law Enforcement
05 Personnel	24 Stochastic Processes	44 Forestry	64 Electrical Engineering	84 Social & Behavioral Sciences (Other)
06 Transportation	25 Probability Theory	45 Ecology	65 Electronic Engineering	
07 Insurance	26 Probability Distributions	46 Geology/Resources	66 Mechanical Engineering	
08 Real Estate	27 Quality Assurance	47 Oceanography	67 Nuclear Engineering	
09 Business (General)	28 Reliability/Maintainability	48 Anthropology	68 System Engineering	
	29 Statistics & Probability (G-need)	49 Natural Sciences (Other)	69 Engineering (General)	
FINANCE	MATHEMATICS	LIFE SCIENCES	TECHNICAL	NATURAL RESOURCES
10 Accounting	30 Linear Algebra/Matrices	50 Clinical/Diagnostic	70 Ceramics	85 Lumber/Forest Products
11 Auditing	31 Complex Variables	51 Virology/Immunology	71 Architecture	86 Oil/Gas/Energy
12 Banking	32 Nonlinear Systems	52 Biochemistry	72 Ceramics	87 Food Resources
13 Consumer Finance	33 Numerical Integration	53 Genetics	73 Heating, Air Cond., Cooling	88 Water Resources
14 Personal Finance	34 Numerical Equations	54 Pharmacology	74 Optics	89 Natural Resources (Other)
15 Economics	35 Number Systems	55 Optoacoustics/Optics	75 Programming	
16 Leasing	36 Operations Research	56 Nutrition/Food Science	76 Surveying	
17 Tax Planning/Preparation	37 Mathematics (General)	57 Life Sciences (General)	77 Astronomy	
18 Securities			78 Archaeology	
19 Finance (General)			79 Technical (Other)	



Texas Instruments hemdator.

Texas Instruments hemdator är en riktig dator. Som hela familjen kan arbeta med, räkna på, lära av eller spela med.

Det finns mer än 1000 färdiga program från hela världen - från rymdspel och schack till budgetering av familjens ekonomi. Lägg bara in programmen och du är klar att börja.

När du själv vill programmera ger den utförliga handboken dig instruktioner, steg för steg. Spännande och inte alls svårt.

Texas Instruments hemdator har hela 16 Kbyte användarminne (RAM) redan från början men stannar inte vid detta - utvecklingsmöjligheterna är mycket stora. Du kan tex ansluta en talenhet som ger din dator röst.

Kom in och prova Texas Instruments hemdator - den är mer än värd sitt pris.



**TEXAS
INSTRUMENTS**