G A R B

NR 1/78



INNEHÅLLSFÖRTECKNING

Denna innehållsförteckning 2
Spalten 2
Ordföranden talar 3
Lysoteket 3
Lysator, en kort återblick
av Ingemar Ingemarsson 4
Lysators Styrelse 5
Varian 5
Jubileum 6
9900 7
LISP, en introduktion
av Anders Ström 8
Terminlysbuggar 12
Modifieringar av LYS 16 13
So Kleves minnestestare 14
LYS-BASICENS felutskrifter 17
Addendum till senaste
mjukvaruutskicket 18

Detta nummers förstasida visar mödorna vid transport av datorer.

Tidningen GARB utges av datorföreningen Lysator, Linköpings Tekniska Högskola

S P A L T E N

UH??

Efter många långa nätters idogt misshandlande av en (många) dataterminaler (som sig bör) är den klar - BLASKAN.

En liten förklaring kanske skulle kunna vara på sin plats.

För ett större antal (!) Lysatormöten sedan bildades en PR - grupp som bla. skulle undersöka om Lysator klarar av att författa och distribuera en liten tidskrift - info blad. PR - gruppen, nu bestående av Björn Lindskog och Lars Dahlberg, gör därför ett försök med detta nummer.

Under en längre tid nar det dumpit ner brev och andra påminnelser om de externa medlemarnas existens. Detta nummer av GARB riktar sig därför till de externa medlemarna och deras frågor.

Eftersom vi - Editorerna är teknologer på D -linjen och följdaktligen borde ägna en del tid åt att studera, så är vi tacksamma för alla bidrag till blaskans författande och konstruerande. Bidrag kan utföras som programmeringstips (allmänt eller om LYS - 16),frågor (dessa kan vi nämligen delegera till våra experter ?). Om du nu har förstått ovanstående så sätter du dig genast ner och skriver till oss!!! Huvudsaken är att vi får HJÄLP!!

Nåväl, vi har ingen ambition att ge ut GARB efter något regelbundet schema, dock ska vi så fort vi nar fått ihop material, försöka trycka ett nummer.

Som en intressant inriktning kan nämnas LYSOTEKET, ett program forum för LYSanvändare, senare även för användare av andra system, (se separat artikel).

Många lyckliga timmar framför terminalen önskar

/Lars Dahlberg/

Editor

ORDF. TALAR

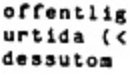
Lysator nar som alla vet firat 5-års jubileum i vår, bla med ett dundrande bakelse kalas men också genom att omstrukturera styrelsen. Vi har hälsat de nya tiderna, dvs alla "nya" datorer vi fått igång, med att utse systemchefer. Den gamla modellen innebar att vi hade soft- resp hard-ware chefer, vilket inte var särskilt praktiskt. Efter en del interna diskussioner kom vi fram till det nuvarande systemet. De nya cheferna utsågs på årsmötet 780308 och stadgeändringen som möjliggjorde detta spikades 780403 (Lysatormässigt om något!).

För övrigt kan nämnas att av de tre LYSar och sex Termilysar vi är i full färd med att spika inop är 1+2 färdiga, med det normala antalet buggar. Vi ska så småningom skaffa en floppydisc (är det beslutat) och med lite tur ordentligt med minne till en av LYSarna.

En stor nyhet är att D21an fungerar! Det är möjligt att köra ALGOL-GENIUS kompileringar och vad annat skoj man kan hitta på (läs: musik). CPU 16K*24 minne och åtta bandstationer går. Den andra stora reliken på Lysator (Collosso, skivminne på >300Mbyte) ska utlokaliseras till en plats i offentligheten som ett exempel på urtida (<1960) datateknik. Vi har dessutom gjort ett nyförvärv på museei fronten: en hålkortssorterare. Den funkar men alla medlemmar gillar inte ideen med hålkort? Till sist ett sorgens ord: vi sitter inte säkert i våra lokaler, illvilliga krafter vill kasta ut oss!

Tecknat vid gott mod

ORDF.



LYSOTEKET!

Vi är nu många som äger den eminenta datorn LYS-16. Det är nu dags att bilda LYSOTEKET, ett forum för insamling och tillhandahållande av program som medlemmarna själva bidrar med. I skrivande stund har detta inte diskuterats (idé'n kom till på 5 minuter en fredagnatt). Meningen är att medlemmarna skall kunna bidra, byta och köpa program.

Förslag till mediaform är den välbekanta kasetten och inom en snar framtid även disketter och remsor. Även 1-tumsband skulle kunna tänkas (program till D21, hi).

Till en början begränsas LYSOTEKET till LYSATORNS datorinnenav: LYS-16, DATASAAB D21, D5/20, D5/30, Varian 520i, TMS 9900 samt mikroprogram till 2900 och om möjligt även dito till IMP-16 (levererade i CROM). Det diskuteras lite om LSI-11 inom en fraktion av LYSATOR också.

Eftersom majoriteten av LYSATOR (?) har en LYS-16 börjar vi lämpligen att bygga upp LYSOTEKET med sådana program.

Vi hoppas att gensvaret blir största möjliga, tänk vilket jättebibliotek det skulle kunna bli. Alla program som fungerar (mer eller mindre) kommer in i LYSOTEKET, även program av typen STARTREK.

Alla synpunkter är välkomna, även förslag till namn om LYSOTEKET verkar trist. (Se även ledaren.)

Förnoppningsvis tänker vi alla i 16 bitar, så minsta ordlängd i LYSOTEKET blir - 16 bitar. (Synpunkter? Skriv!)

Avslutningsvis: Givetvis kommer även LYSATOR att bidra med många program till LYSOTEKET.

Med många hälsningar från Eder sekreterare tillika systemchef Varian.

/Jan Axing/

(Obs! 4-bitarsfantast.)

LYSATOR

Hur kom det sig, vad blev det och vad skall det bli?

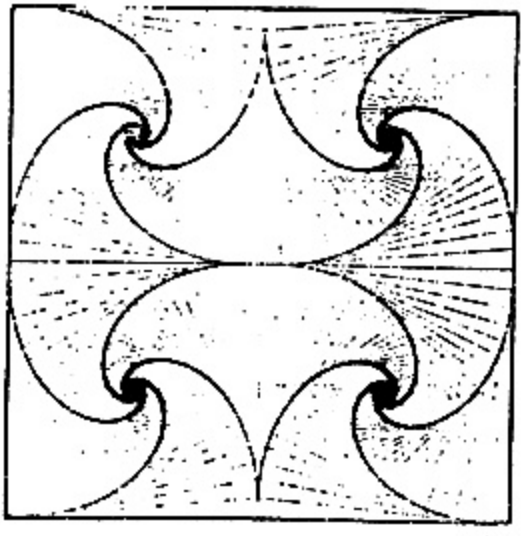
Av Ingemar Ingemarsson.

Eftersom jag en gång tog initiativet till LYSATOR skall jag försöka erinra mig hur det började. Hur uppstår ideer? Svårt att svara på, oftast kryper de fram lika slumpartat men regelbundet som blommorna om våren. En av knopparna ledde till att jag pratade med Kjell Stålbom, som då var teknolog på Y-linjen, om att starta någon slags datorförening. För att göra skäl för namnet behövde vi något som liknade en dator. Därför ringde jag till Gunnar Lindström som då var chef för en känd datorfabrik här i staden. Han var helt med på noterna och lovade att leta i gömmorna efter prylar som inte användes. Resultatet blev över förväntan, man nittade en gammal men (nästan) ny och bra kärra som vi fick. Ett villkor var att Datasaab frånsade sig allt ansvar för prylarna och deras funktion efter leveransen. Detta hindrade dock inte Kjell och hans förhoppningsfulla kompisar att åka till Datasaab och "hjälpa till att bära".

Hur det bar (!) sig så återvände de med inte mindre än tre (3) datorer, eller rättare, delar av datorer. De behövde kompletteras och då kom jag att tänka på statskontoret som handlar med alla datorer inom statsförvaltningen. Där borde finnas en massa gammalt skräp, tänkte jag, och misstog mig inte. Vänliga människor hjälpte oss med att nitta en D21 i Göteborg som höll på att bytas ut. En lastbil hyrdes och lastades med prylar som införlivades med LYSATORs nu ganska omfångsrika samling. Om jag inte missminner mig, fick vi betala 1 krona, dvs det belopp som grejorna var nedskrivna till!

Den stora samlingen hårdvara verkade till en början lite handlingsförlamande. I vilken ände skulle vi börja? Så småningom fick entusiasterna maskineriet så pass, att det en luciadag kunde framföra Sankta Lucia. Föreningen växte dock i de till en början lite för stora kläderna och kunde bygga vidare på flera vilda ideer. En sådan klämktes av Robert Forchheimer och blev LYS 16. Vi vet alla hur det gick.

Vad ska man önska sig av framtiden för LYSATOR? Det känns onaturligt för mig att försöka styra verksamheten genom att tycka till från en konstlat uppnådd position. Det enda jag noppas är att föreningen lika entusiastiskt som hittills ska ta sig an nya ideer som skjuter upp, även om de från början verkar lika gräs.



LYSATORs styrelse ser för närvarande ut som:

Ordförande	Sören Tirfing	17 09 75
Driftchef	Tomas Pettersson	15 96 79
Administrativ chef	Olov Fanlander	13 38 76
Kassör	Anders Thulin	
Sekreterare	Jan Axing	17 26 38
Editor	Lars Dahlberg	17 15 50
2nd editor	Björn Lindskog	17 45 29
Systemchefer:		
D21	Mats Lengren	14 06 86
D5/30	Tomas Magnusson	15 16 79
LYS-16	Göran Karlsson	17 15 81
Varian	Jan Axing	17 26 38
Projekt (2901)	Hans Norberg	17 06 47
La Cimbali	Robert Forchheimer	17 46 93



V A R I A N

Vad säga om Varian? Just nu är den lite krasslig, men det verkar lösa sig. Hittils har jag funnit 9 pajade chipa, mest 7401-or.

Remsläsaren är just nu det enda i 'systemet' som funkar, stansen lider av sitt mekaniska fel ännu.

När Varian kommer igång skall den troligtvis användas som kanalprocessor till diskarna. Vi får se hur det går. Svårast blir att skriva ett OS för detta som får plats i dess futtiga 8 kbyte minne. För närvarande håller jag på med en editor till assemblern. Denna är skriven i HP- språket (sk. Basic.) då assemblern har en dålig sådan. Dessutom tar en bra editor i Varian för stor plats. På Önskelistan står det mera minne, mera minne.....

FORTRAN rots the brain.
-- John McQuillan

Beträffande mjukvara i övrigt finns en remskonverterare avsedd att ta hand om remsor från Hans Norbergs HP-program som underlättar produktion av Algolgeniusprogram till vår eminenta D21:a. (Lång mening.) Det finns tyvärr några buggar i Hans program, men tala med honom.

Nåja, lillen kommer väl så småningom igång igen noppas jag.
Eder systemchef:
/JAN AXING/



LYSATOR 5 ÅR

Den 4 april firade Lysator sitt femårs jubileum. Vi är ju som bekant sveriges äldsta datorförening, med anor ända från 1973. Mycket har förändrats sedan dess men en sak upptar fortfarande medlemmarnas (och även andras) intresse: D21.

Lysator bildades ju mer eller mindre i samband med att en D21 skänktes av SAAB. (Se Ingemar Ingemarssons historik) Runt denna D21 kretsade det huvudsakliga datorintresset. Senare kom LYS-16 systemet in i bilden och D21 kom mer eller mindre att falla i glömska. Men några oförskräckta entusiaster kunde inte slita sig från denna relik utan jobbade kontinuerligt med den, även efter det att CPU:n navererade för något år sedan.

I slutet av förra året lyckades vi komma över en annan D21, som under vådliga strapatser flyttades från Stockholm ned till Linköping. Den monterades upp, och fungerade! Lagom till 5-års jubileet hade Lysator fått en ny D21. Cirkeln slöts.

Jubileet hade för övrigt noggrant förberetts. En speciell kak-kommite utsågs i god tid med uppgift att konstruera en speciell Lysatorbakelse och även stå för övrig förplägnad vid det högtidliga tillfället.

Eftersom ett femårsjubileum inte inträffar varje år, beslöts att även icke Lysatormedlemmar skulle få ta del av högtidligheterna. Sålunda kunde de anställda vid Institutionen För Systemteknik, å vilkens domäner Lysators lokal är belägen, samt datalogerna att vid normal eftermiddagsfikatid finna kaffet redan klart och detsamma utbjudet



tillsammans med högst eminenta specialkomponerade bakelser.

Tyvärr möttes denna mycket givmilda gest med idel misstänksamhet. Muttranden av typen "Vad är dom nu ute efter?" hördes från flera håll. Förutom bakelseätning och kaffedrickande ägnades jubileumsdagen åt mer Lysator-mässiga sysselsättningar såsom LYS-16 byggande och naturligtvis Algol-Genius kompileringar på D21.

De flesta jubileumsdeltagarna uppskattade tilldragelsen och uttryckte önskemål om fler bakelser. Trots detta ser vi med tillförsikt fram mot nästa jämna femtal och noppas att vi också då har en fungerande D21.

9900

Under den senaste tiden har det bland LYSATOR-medlemmarna vid LiTH visats ett nyvaknat intresse för mikrodatorer. Detta har bla tagit sig uttryck i byggandet av tre LYS-16 och ett flertal Termilysar för föreningens räkning. Från vissa medlemmar har också framförts önskemål om lämpliga hemdatorsystem.

Dessa skulle vara flexibla och kunna byggas ut steg för steg från ett minimalsystem till en mycket sofistikerad hemdator. Pga våra mycket goda erfarenheter från LYS-16 har kravet på en 16 bitars maskin förts fram. Ett annat krav är att kostnaden för startkonfigurationen ligger inom rimliga gränser.

Efter en del sonderingar kom vi fram till att en TMS9900 skulle passa bäst som CPU. Det är en mycket populär och relativt billig 16 bits processor. Efter ytterligare sonderingar fann vi att det fanns åtminstone fyra system (i Sverige !) baserade på 9900, färdiga eller under utveckling.

Anders Rundgren har tex i en serie i Modern Elektronik presenterat ett helt modul-uppbyggt 9900-system, kallat MD99.

Chalmers Dataförening (CD) håller just på att färdigställa ett CPU-kort av stand-alone typ som också kan anslutas till en yttre buss. Detta kort ska enligt uppgift innehålla, förutom CPU och kringkretsar, en UART, 4Kord RAM och plats för 2Kord PROM (4st 2708). Kortet bjuder dessutom på skrivskydd för minnet i 32ords inkrement samt refresh logik. En annan godbit är ett minneskort med 4Kord dynamiskt minne. Minneskapslarna, som är av Mosteks tillverkning, är helt pinn-kompatibla med firmans 16K-kapslar. Detta medför att et 4Kord kort lätt kan ändras till et 16K do när 16K kapslerna fått et mer attraktivt pris.

Många av de Linköpingsbaserade Lysator medlemmarna som inte redan har en hemdator kommer förmodligen att skaffa ett sådant CD system.

CDs nyupprättade programbank för microprocesorer gör det dessutom högst troligt att det kommer att finnas ett starkt mjukvarustöd tillgängligt.

Vissa tecken tyder dock på att CD inte har för avsikt att utveckla fler hårdvarumoduler än de som är under produktion. Det innebär ju trots allt ganska mycket arbete att konstruera datorer och dess kringutrustning.

Lysator planerar därför att inom en inte alltför avlägsen framtid konstruera bla floppyinterface och videoram till systemet.

Om du är intresserad kontaxta Lysator.

Det kan för övrigt nämnas att när detta skrevs så planerade CD att sälja kompletta komponent satser (inkl kort) för ca 1700:-.



LISP - hur fungerar det ?

Av Anders Ström

Låt oss omedelbart titta på ett exempel.

Om du skriver in följande till en LISP interpretator

(TIMES PI (PLUS 12 4 -11))

skulle den villigt lämna svaret

15.70

under förutsättning att PI har värdet 3.14.

Redan detta enkla exempel visar mycket som är typiskt för LISP. Du skriver in ett uttryck som omedelbart beräknas (evalueras), varefter resultatet skrivs ut.

Ett program i LISP lagras direkt i LISPs grundläggande datastrukturer, nämligen atomer och listor. Denna ekvivalens mellan program och data, gör LISP utmärkt för att operera på LISP program. Man kan på ett par sidor definiera en LISP interpretator i LISP.

En atom är entydigt definerad av sitt namn. Dvs om två atomer skrivs ut på samma sätt är det samma atom. Exempel på atomer är PI, PLUS, 2 och 1-FOO. Till varje atom kan vi associera ett godtyckligt antal egenskaper. Två av dessa egenskaper är så speciella att man bör minnas dem, nämligen atomens värde och dess funktions definition.

En lista börjar med en vänsterparantes och innehåller sen ett godtyckligt antal element, som vart och ett kan vara en lista eller en atom, och avslutas med en högerparantes.

När ett uttryck skall evalueras tittar interpretatorn först på om det är en atom eller lista som ska evalueras. Om det är en atom returneras värdet av atomen. Om det är en lista tolkas första elementet i listan, som vi förutsätter är en atom med funktions egenskap, som namnet på en funktion, och resten av listan som argumenten till denna funktion. På detta sätt kan vi ge godtyckligt många argument till en funktion, t.ex. PLUS kan ta hur många argument som helst. Nästa steg i evalueringen blir nu att evaluera argumenten till funktionen. Dessa evalueras på samma sätt, dvs interpretatorn anropar sig själv. Till sist applicerar vi funktionen till de aktuella argumenten. <1>

Ovan gjorde vi en enkel numerisk beräkning, men LISP är inte bara till för att göra numeriska beräkningar, utan framför allt till för att hantera symboler. Vi kan till exempel hålla ordning på vilka som är våra vänner genom att skriva

Forts.

(SETQ VÄNNER ´(LISA PELLE DATORN))

Funktionen SETQ sätter den variabel som är första argument till värdet av andra argumentet.

Vad gör nu blippen (´) i uttrycket?

Vi vill ju att VÄNNER skall få värdet (LISA PELLE DATORN), och inte det som lämnas tillbaka om vi beräknar uttrycket (LISA är en odefinierad funktion).

Blippen (eller "quote" som varje garvad LISP programmerare säger) betyder helt enkelt att vi inte ska beräkna utrycket för att få värdet, utan att vi ska ta uttrycket i sig självt som värde. <2>

Om vi senare vill ha reda på vilka som är våra vänner skriver vi

VÄNNER

varvid svaret blir

(LISA PELLE DATORN)

Låt mig nu ge två exempel som klart visar skillnaden mellan att använda quote och inte.

Först med quote

(SETQ KOMPISAR ´VÄNNER)

VÄNNER

Här sätts KOMPISAR till VÄNNER.

Och utan quote

(SETQ KOMPISAR VÄNNER)

(LISA PELLE DATORN)

Här sätter vi alltså KOMPISAR till (LISA PELLE DATORN), dvs värdet av VÄNNER.

Vi kan också hålla reda på våra fiender, vi skriver

(SETQ FIENDER ´(OLLE TROLLET SPÖKET))

Om vi vi en dag upptäcker att TROLLET är vår vän och inte en fiende, så måste vi kunna ändra på det.

Vi skriver då

(SETQ VÄNNER (CONS ´TROLLET VÄNNER))
(TROLLET LISA PELLE DATORN)



Mats.

```
(SETQ FIENDER (REMOVE 'TROLLET FIENDER))
  (OLLE SPÖKET)
```

CONS adderar det element som ges som första argument till listan som ges som andra argument. REMOVE plockar bort alla förekomster av första argumentet i listan som ges som andra argument.

Om vi har många vänner och många fiender, och vi hela tiden ändrar oss vad gäller vem som tillhör vilken grupp, så blir det ganska jobbigt att skriva in allt detta varje gång. Vi kan då definiera en funktion för att förenkla det.

```
(DE GÖR-TILL-VÄN (NAMN)
        (SETQ VÄNNER (CONS NAMN VÄNNER))
        (SETQ FIENDER (REMOVE NAMN FIENDER)))
```

Då kan vi istället skriva

```
(GÖR-TILL-VÄN 'TROLLET)
```

Funktionen DE som används för att definera andra funktioner kan beskrivas med
(DE <namn> <argument> <form-1> ... <form-n>)
Där ord inom vinkelparanteser beskriver vad som ska placeras i dess ställe.

Värdet av en funktion är samma som värdet av den sista formen (<form-n>) i funktionen.

Naturligtvis har även funktionen DE ett värde, nämligen den funktion man just har definerat.

Nu hoppas jag att alla läsare har undrat
"Skulle man inte evaluera argumenten till en funktion, hur går det inop med anropet till DE ovan?"

Helt rätt, jag har inte berättat allt ännu. Genom att använda funktionen DF istället för DE när vi definerar en funktion, kan vi få special funktioner (så kallade FEXPRs till skillnad från EXPRs) som inte ska ha sina argument evaluerade.

DE och DF är exempel på FEXPRs.

Även funktionen SETQ är en FEXPR, den ska ju inte evaluera sitt första argument. Det finns en funktion SET, som fungerar som SETQ med den skillnaden att den evaluerar båda sina argument. Vi kan nu definera SETQ med hjälp av SET.

```
(DF SETQ (X Y) (SET X (EVAL Y)))
```

Där EVAL är ett explicit anrop till LISP interpretatorn.

Mycket mer finns att säga om LISP, men press-stop avlägsnar sig alltmer, så någon gång måste det ta slut. Men om inte himlen rasar ner över oss är det möjligt att det blir någon mer artikel

Mats.

om LISP i senare nummer av GARB.

För den som inte kan hålla sig rekomenderas Patrick Winstons bok "Artificial Intelligence", den lär finnas på biblioteket. (Jag har stulit en del exempel från den boken.)

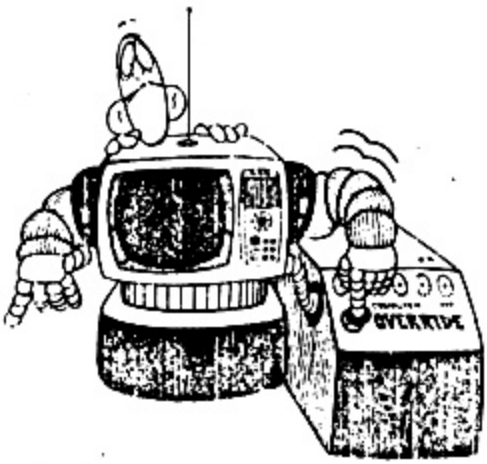
<1>

Det vållar ofta nybörjaren problem att hålla reda på när man ska använda quote och inte. Detta är särskilt markant om personen i fråga har använt andra programmerings språk tidigare. Det beror på att man där ofta fuskar bort problemet genom att helt enkelt bara tillåta den ena varianten.

Om vi tittar på PASCAL och utgår från deklarationen
type färg = (röd, grön, blå);
sedan kan man inte använda röd som en variabel, för hur ska man kunna veta vad x:=röd betyder. Menar man variabeln röd eller röd som ingående i typen färg.

<2>

I LISP nöjer vi oss med funktioner medan man i andra språk använder använder sig av funktioner, procedurer och operatorer. En funktion lämnar alltid ett värde, även om det många gånger slängs bort omeddelbart.
@
e





ON A CLEAR DISK YOU CAN SEEK FOREVER.
-- ED MC CLUSKY

WRITING A PROGRAM IS NOTHING BUT DEBUGGING A BLANK PAGE.
-- ANONYMOUS, BUT FRUSTRATED, STANFORD PROGRAMMER

T E R M I L Y S B U G G A R

Torbjörn Linder har i ett brev till Lysator beskrivit några fel som uppträtt i hans Termilys. Här följer hans frågor, vilka har besvarats av Olof Fanlander, med nopp om att detta kan vara till hjälp för andra Termilysbyggare.

1. Radslut ger ej radbyte vid skrivning såvida inte raden ifråga inletts med ett LF.

2. Vid stort format blir rad 0 (eller 32) förvanskad vid skrivning.

3. Erase screen (form Feed) fungerar inte:

a. radering sker inte

b. FF no 1 ger en displayundertryckning vid litet format t o m position 50 och vid stort format t o m position 25. Nästa inskrivna tecken eller FF no 2 ger tillbaka den gamla skärmbilden, nu berikad med en rad "skräptecken" som namnar ganska godtyckligt på skärmen, strax före cursorn.

4. Vid växling från grafisk mode till alfanumerisk mode namnar en hakparentes på skärmen.

Olofs förklaringar och recept till lösning av problemen är följande:

1. Carry från kolumndelen till rad-delen av skrivadressräknaren [illegible] då 5a5 = 5H12 går från 1 [illegible] 0. Som framgår av schema II är denna signal inverterad med 6E5/6 vars utgång är kapacitivt lastad, detta för att de bägge eller-ingångar som känner dessa båda signaler skall under denna övergång känna två nollor under kort tid. För NOR-grinden 6D6 gäller då att de båda andra ingångarna 1 och 2 skall vara 0 vilket de troligen är. Den troligaste orsaken till den uteblivna carrypulsen är förmodligen att den ingående kondensatorn C5 är för liten, men den bör å andra sidan inte väljas för stor p g a att NOT-grinden annars tröttas ut. Lämpligt värde bör vara c:a 330-470 pF. F ö bör under detta förlopp även ingångnen 2E11 vara noll. Mät och upptäck!

2. Att tecknen förvanskas beror förmodligen på att de skrivs in fel (Detta kan avgöras om samma tecken ligger kvar vid växling till liten stil) Anledningen är troligen att skrivpulsen till minnena blir i kortaste laget. Genom att jämföra ASCII-koden för vad som avsågs och vad det blev kan man enkelt avgöra vilken bitposition som går snett. Ram-kapslarna är placerade på moderkortet så att bit 7 är längst åt vänster och bit 0 åt höger, två och två. (2*8=16) Av dessa båda positioner är den nägra för rad 0-15 och följaktligen den vänstra för 16-31. Kortet skall hållas så att Ram-kapslarna är uppåt. Härmed skall den felande Ram-kapseln kunna identifieras och denna kan då växlas med någon av de båda i bit 7 som ju bara används i grafisk mode där skrivning normalt bara sker i litet format. Alternativt byt till snabbare 2102.

Mats.



3. Undertryckningen uppstår genom att FF-vippan 1B2/3 aldrig återställs genom att ingen kvittering kommer fram till 1B1/2C1. Det enda jag kan hänvisa till är att följa signalvägarna som de beskrivits i funktionsbeskrivningen. Ex vis skall tågpulsgeneratorn leverera skurar av 64 pulser som avslutas med attt latchen 4C1-6 växlar. Allt sker i takt med CPSY-pulsen. Efter att detta utförts 32 - 63 gånger skall punkten 4G5 gå från 1 till 0 vilket skall avsluta förloppet. MÄT!

4. Detta är helt normalt och beror på att alla tecken skrivs in i grafisk mode, d v s även omkopplingen till AN, vilket är just det tecken som syns.

Sammankoppling av minnes kontaktraden 1a, 1b och GND stiftet för Reset har visat sig ge en minskad störkänslighet hos minneskorten.

Ett bra sätt att förstärka jorden till kapslarna, är ca 6 mm breda enkelsidiga koppar - laminatremsor. Tillkapade längder av dessa läggs mellan sockelarnas ben och löds (via trådbitar) till jordledningen och till kapslarnas jordstift. Kontrollera NOGA att inga andra sockelben ligger mot laminatremsan. Denna förstärkning bör göras på kapselraderna mellan RALU - kapslarna och minneskortshållarna, samt raden med bl a 8434A. 8334A - radens laminatremsa förbinds med jordledningarna i båda ändar.

Avkopplingskondensatorer för 5 V bör även lödas in enligt följande:

a/ 100 nF mellan kapslarna:
7408 A - D 3st
8123 A - D 3st
74175 A - D 3st
74H00B och 7404B
7404B och 74H00B

b/ 4.7 uF mellan kapslarna:
74H04 A - D 3st
8095 A - D 3st
74H00B och 7432B
7432A och 7408F

Avkoppla även -12 V vid mosdrivarna, med en tantal på 10 uF, 35V.

/BO KLEVE/



I gather it chou

```
                        .TITLE PRNTST
                    ;PROGRAM FOR TESTING THE MEMORY WITH
                    ;PSEUDORANDOM-NUMBERS
                    ;COPYRIGHT      BO KLEVE, LYSATOR
0000 0000 A                 R0=0
0000 0001 A                 R1=1
0000 0002 A                 R2=2
0000 0003 A                 R3=3
0000 0004 A                 BIT1=4
0000 000B A                 NEG=11
0000 0002 A                 SEL=2
0000 0007 A                 FLAG=7
0000 FF80 A                 ENTER=X'FF80
0000 FFA4 A                 READWD=X'FFA4
0000 FFA4 A                 OUTC=X'FFA4
0000                    ;
0000                    ;
0000                        .ASECT
0000 FE80 A                 .=X'FE80
FE80 895E A         INIT:   LD      R2,TXTAD
FE81 2D32 A                 JSR     @PRIAD
FE82 03A4 A                 JSRI    READWD
FE83 21FC A         SOH:    JMP     INIT
FE84 0380 A         ETX:    JSRI    ENTER
FE85 3781 A         NORMAL: RCPY    R1,R3
FE86 4D01 A         START:  LI      R1,1
FE87 2923 A         PASS:   JSR     PRN             ;GET A RANDOM IN R1
FE88 4100 A                 PUSH    R1
FE89 8929 A                 LD      R2,PSZ
FE8A 2920 A         STORE:  JSR     PRN             ;A RANDOM IN R1
FE8B 4400 A                 PULL    R0
FE8C 4000 A                 PUSH    R0
FE8D 4300 A                 PUSH    R3
FE8E 2918 A                 JSR     MADDR           ;AN ADDRESS IN R3
FE8F 4700 A                 ST      R1,(R3)
FE90 4700 A                 PULL    R3
FE91 4AFF A                 AISZ    R2,-1
FE92 21F7 A                 JMP     STORE
FE93 891F A                 LD      R2,PSZ
FE94 4500 A                 PULL    R1
FE95 4100 A                 PUSH    R1
FE96 2914 A         VERIFY: JSR     PRN             ;GET A RANDOM IN R1
FE97 4400 A                 PULL    R0
FE98 4000 A                 PUSH    R0
FE99 4300 A                 PUSH    R3
FE9A 290C A                 JSR     MADDR           ;AN ADDRESS IN R3
FE9B F700 A                 SKNE    R1,(R3)
FE9C 2101 A                 JMP     .+2
FE9D 2917 A                 JSR     ERROR
FE9E
FE9E 4700 A                 PULL    R3
```



```
 52.  FE9F 0AFF A            AISZ    R2,-1
 53.  FEA0 21F5 A            JMP     VERIFY
 54.  FEA1 4C40 A            LI      R0,X'40
 55.  FEA2 4900 A            PUSH    R3
 56.  FEA3 03A9 A            JSRI    OUTC
 57.  FEA4 4700 A            PULL    R3
 58.  FEA5 4500 A            PULL    R1
 59.  FEA6 21E0 A            JMP     PASS
 60.  FEA7                ;
 61.  FEA7                ;
 62.  FEA7 3882 A  MADDR:    RXOR    R2,R0
 63.  FEA8 610A A            AND     R0,PSZ
 64.  FEA9 3300 A            RADD    R0,R3
 65.  FEAA 0200 A            RTS     0
 66.  FEAB                ;
 67.  FEAB                ;
 68.  FEAB 3481 A  PRN:      RCPY    R1,R0
 69.  FEAC 3100 A            RADD    R0,R1
 70.  FEAD 6104 A            AND     R0,MASK
 71.  FEAE 4800 A            AISZ    R0,0
 72.  FEAF F102 A            SKNE    R0,MASK
 73.  FEB0 4901 A            AISZ    R1,1
 74.  FEB1 0200 A            RTS     0
 75.  FEB2 8400 A  MASK:     .WORD   X'8400
 76.  FEB3 0FFF A  PSZ:      .WORD   X'0FFF
 77.  FEB4 FF6D A  PRIAD:    .WORD   X'FF6D
 78.  FEB5                ;
 79.  FEB5                ;
 80.  FEB5 4200 A  ERROR:    PUSH    R2
 81.  FEB6 4300 A            PUSH    R3
 82.  FEB7 4EFA A            LI      R2,X'FA
 83.  FEB8 2DFB A            JSR     @PRIAD
 84.  FEB9 4700 A            PULL    R3
 85.  FEBA 4600 A            PULL    R2
 86.  FEBB 4100 A            PUSH    R1
 87.  FEBC 290F A            JSR     OUTWRD
 88.  FEBD 5700 A            XCHRS   R3
 89.  FEBE 290D A            JSR     OUTWRD
 90.  FEBF 5700 A            XCHRS   R3
 91.  FEC0 4500 A            PULL    R1
 92.  FEC1 4300 A            PUSH    R3
 93.  FEC2 4100 A            PUSH    R1
 94.  FEC3 8F00 A            LD      R3,(R3)
 95.  FEC4 2907 A            JSR     OUTWRD
 96.  FEC5 4500 A            PULL    R1
 97.  FEC6 4100 A            PUSH    R1
 98.  FEC7 3782 A            RXOR    R1,R3
 99.  FEC8 2903 A            JSR     OUTWRD
100.  FEC9 4500 A            PULL    R1
101.  FECA 4700 A            PULL    R3
102.  FECB 0200 A            RTS     0
103.  FECC                ;
```

BE PATIENT; REMEMBER, HALF THE PEOPLE IN THE WORLD
ARE BELOW AVERAGE.
-- MARVIN GETTNER



```
104.  FECC                ;
105.  FECC 4004 A   OUTWRD: LI      R1,4
106.  FECD 0A80 A           PFLG    SKL
107.  FECE 5804 A   X2A:     ROL     R1,4      ;HEX TO ASCII
108.  FECF 4C00 A           LI      R0,X'0F
109.  FED0 3C83 A           BAND    R3,R0
110.  FED1 48F6 A           AISZ    R0,-10
111.  FED2 1801 A           BOC     NEG,DEC
112.  FED3 4807 A           AISZ    R0,7
113.  FED4 483A A   DEC:     AISZ    R0,X'3A
114.  FED5 4300 A           PUSH    R3
115.  FED6 03A9 A           JSRI    OUTC
116.  FED7 4700 A           PULL    R3
117.  FED8 497F A           AISZ    R1,-1
118.  FED9 21F4 A           JMP     X2A
119.  FEDA 4C20 A           LI      R0,' '/256
120.  FEDB 4300 A           PUSH    R3
121.  FEDC 03A9 A           JSRI    OUTC
122.  FEDD 4700 A           PULL    R3
123.  FEDE 0200 A           RTS     0
124.  FEDF                ;
125.  FEDF                ;
126.  FEDF FEE0 A   TXTAD:  .WORD TEXT
127.  FEE0 0D0A A   TEXT:   .WORD X'0D0A
128.  FEE1 494E A           .ASCII 'INPUT STARTADDRESS'
      FEE2 5055 A
      FEE3 5420 A
      FEE4 5354 A
      FEE5 4152 A
      FEE6 5441 A
      FEE7 4444 A
      FEE8 5245 A
      FEE9 5353 A
129.  FEEA 2003 A           .WORD X'2003
130.  FEEB FE80 A           .END INIT

******      NO ERRORS IN ASSEMBLY   ******
```

PRNTST testar RAM-minnet med pseudoslumptal enligt en metod angiven i Popular Electronics. För varje genomlöpt testcykel skrivs ett "@" ut. Upptäckta fel skrivs ut som:

Adress, skrivet data, läst data samt EXOR av skrivet och läst data.

EXOR-utskriften ger besked om vilka bitar som skiljer sig åt. Rutinen använder sig ej av några RAM-positioner varför den med fördel kan läggas i PROM och användas för test av hela minnet.

Konstanten PSZ på FEB3 anger minnesstorleken:
07FF för 2K
0FFF för 4K
1FFF för 8K
3FFF för 16K

ERROR MESSAGES TO
LYD-16 BASIC,VERSION 1.

26,10,81.



ERROR NR:	ERROR MESSAGE:
1	TOO LARGE LINE NUMBER
2	CHARACTER AFTER STATEMENT END
3	MISMATCH SYNTAX/PROGRAMTEXT
4	MISSING OR ILLEGAL 'TO'
5	SYNTAX OR ILLEGAL STEP
6	MISSING OR ILLEGAL RELATIONOPERATOR
7	MISSING OR IF ... THEN
8	UNDEFINED OR ILLEGAL
9	LOAD PROGRAM AREA IS FULL
10	TOO MANY DIMENSIONS
11	... NOT ALLOWED
12	...
13	... VARIABLE
14	... EXPRESSION
15	... NOT DIMENSIONED
16	...
17	...
18	...
19	...
20	...
21	...
22	...
23	...
24	...
25	...
26	...
100	...
101	...
102	...
103	...
104	...
105	...
106	...
107	...
201	...
202	...
203	...
204	...
205	...
206	...
207	...
208	...
209	...
210	...
211	...
212	...
401	...
501	...
999	...

SYSTEMKIT

BASIC
(erfordrar minst 8K)
Start : 092F
Omstart : 0033
Program : 0010 - 022F
därefter buffertar.
OBS! Källtexten till BASIC ligger på baksidan av kassetten. Dock ej på *EDITOR - forsat utan dumpat rutt* upp och ner. Om du vill se vad som står på skriv ett program som läser från band och skär på skärmen. (Lämplig övningsuppgift).

4K assembler (FASM)
Start : 0000
Omstart : 0001
Programarea 0000 - 0990
Programmet använder samma buffert som 4K EDITOR'n. På samma sätt som i denna används följdaktigen cellerna 10 och 11 för att ange början och slut på *bufferten*. OBS! Vid kopiering av dessa program således cell 11 utelämnas så att innehållet ej ofrivilligt påverkas när swapping sker mellan programmen.

RUN
Start : 0500
Omstart : Varför?
Basarean (0 - 5F)
Programarea 0400 - 0510

Relokerbara program :
STORE och *SUPERDUMP*

STORE
(Görans special)
Start : 0200
Prog. area 0200 - 03A7
Anm: Detta program är EJ exakt lika *med den källtext* som är medsänd. Bl.a. så kvadras inte de första 16 cellerna av denna version av STORE. I övrigt är funktionen helt ekvivalent.

LUFFARSCHACK
Programmet *erfordrar* 8K minne.
Start : 0600
Basarea + (600-659)
Bildarea : 86A - 9E5
Prog. area 0B2A - 0FFF

STORE. ASM
.ASM innebär källtext enligt allmän konvention. Använd denna fil för test av Editorn och assemblern.

4K EDITOR
Programarea : 0010 - 0990
därefter Editorbuffert. Aktuellt slut på buffert finns i cell 0011
Start /Återstart i cell 0200
Beroende på om bufferten är tom eller ej, sker initiering. I övrig samma kommandon som i editordelen i LYSASM. OBS!! Vid kopiering av programmet se kommentar till 4K - Assembler'n.

LYSASM
Programarean : 0000 - 16C8
därefter editor buffert.
Start : 0FC7
Omstart : 0000
(0F8d efter GO)
Erfordrar minst 8K.

Vissa buggar har upptäckts i programmet. 1. LL fungerar ej. Obs! LF Rekvotet bortslopat, använd LST. 2. XE kräver nedtryckning av godtycklig tangent (för frammatning av osynliga sidor) 3. XO dito. 4. Felaktigt slag mellan tecken och records vid dumpning och av. läsning. Felet beror till ator del på ofullkomligheter i rela - logiken på c: a bandspelarkortet. Rekomenderad lösning är att ej använda relästyrningen. (gör enklare handhavande, enligt vår erfarenhet) eller att sätta in en fördröjning i bandspelaren, så att korta frislag ej påverkar denna.
/Robert Forsmeimer/

